UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

USING DEEP LEARNING TO IMPROVE PREDICTION AND UNDERSTANDING OF HIGH-IMPACT WEATHER

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

RYAN LAGERQUIST
Norman, Oklahoma
2020

USING DEEP LEARNING TO IMPROVE PREDICTION AND UNDERSTANDING OF HIGH-IMPACT WEATHER

A DISSERTATION APPROVED FOR THE
SCHOOL OF METEOROLOGY

BY THE COMMITTEE CONSISTING OF

Dr. Amy McGovern, Chair

Dr. Michael Richman

Dr. Jeff Basara

Dr. Jason Furtado

Dr. Andrew H. Fagg

Dr. Justin Metcalf

© Copyright by RYAN LAGERQUIST 2020
All Rights Reserved.

Dedication

I dedicate this dissertation to Hypatia, Eratosthenes, and Enheduanna.



Hypatia.



Eratosthenes and Enheduanna.

Acknowledgments

Work presented in this dissertation was funded by the National Science Foundation, under grant EAGER NSF AGS 1802267, and the National Oceanic and Atmospheric Administration (NOAA) Office of Oceanic and Atmospheric Research, under NOAA – University of Oklahoma (OU) Cooperative Agreement #NA16OAR4320115. NOAA is part of the United States Department of Commerce. Computing was done at the OU Supercomputing Center for Education and Research (OSCER), provided by OU, and Cheyenne (doi:10.5065/D6RX99HX), provided by the National Center for Atmospheric Research (NCAR) Computational and Information Systems Laboratory. NCAR is sponsored by the National Science Foundation. Unless otherwise noted, all plots in this dissertation were generated with matplotlib version 3.1.1 (Hunter 2007). Weather Prediction Center (WPC) surface bulletins were provided by Alan Robson at the WPC.

I have been helped along the way, both personally and professionally, by so many people that it would be impossible to thank everyone individually. However, I would like to thank my mentors, who have gone above and beyond. The first is my committee chair and supervisor, Amy McGovern. Over the last six years, Amy has always been available to talk about science and life in general, both of which can be difficult during graduate school. She has encouraged me to maintain work-life balance and supported my professional development in a myriad of ways. She has nominated and encouraged me to serve on American Meteorological Society committees, encouraged me to take internships and sought out potential internships, encouraged me to collaborate with others and sought out potential collaborators, and also helped me find a post-doctoral position. Amy is a one-of-a-kind supervisor who works tirelessly to support her students and promote them in the science community at large. The second is David John Gagne II, who sponsored my visit to NCAR in summer 2019 and has always been a willing teacher and brainstorming partner. The third is Brian Williams, who supervised my internship at Google in 2017. Brian always made me feel welcome in his home and office, taught me much of what I know

about good coding practice, and let me live at his house and pet his cats for my last two weeks in California. Fourth, I would like to thank Cameron Homeyer and Thea Sandmael for creating and providing the GridRad data in this study, as well as their creative input on the tornado project. Finally, I would like to thank John Allen for approaching me with the idea to create a climatology of fronts, providing the ERA5 data used to do so, and providing creative input on the climatology project.

I would also like to thank my friends, but I don't even know where to start. I have struggled with mental health throughout graduate school, and I have been supported by so many people, most of whom probably don't know how much positive impact they have had. Some of them have been my closest friends, and some of them have just been caring colleagues. The atmospheric-science community is amazing, and I'm immensely privileged to be a part of it and know all of you. To everyone who has reached out and supported me during difficult times, it means more than you know, and I hope to pay it forward. And yes, I'm talking about this in my official dissertation, because mental health needs to be destigmatized. Create a safe space where your friends and colleagues can discuss their mental health.

Finally, I would like to thank the Toronto Maple Leafs, 2019-2020 Stanley Cup champions, for inspiring me on the ice.

Table of Contents

Dedication iv

Acknowledgments vi

List of Tables xi

List of Figures xiii

Abstract xviii

1 Introduction 1

2 Literature Review 7
- 2.1 Tornadoes . . . 7
  - 2.1.1 Tornadogenesis Mechanisms . . . 7
  - 2.1.2 Tornado Prediction . . . 18
- 2.2 Synoptic-scale Fronts . . . 26
  - 2.2.1 Physical Description . . . 26
  - 2.2.2 Climatology . . . 31

3 Data Description and Pre-processing 37
- 3.1 Tornado Prediction . . . 37
  - 3.1.1 Data Description . . . 39
  - 3.1.2 Echo Classification . . . 44
  - 3.1.3 Storm Detection . . . 46
  - 3.1.4 Storm-tracking . . . 49
  - 3.1.5 Tornado Attribution . . . 53
  - 3.1.6 Creation of Predictors . . . 54
- 3.2 Front Detection . . . 60

4 Convolutional Neural Networks 68

5 Hyperparameter Experiments 81
- 5.1 Tornado Prediction (finding the best MYRORSS and GridRad models) . . . 83
- 5.2 Front Detection . . . 91
  - 5.2.1 Finding the Best CNN . . . 92
  - 5.2.2 Finding the Best Object-conversion Method . . . 96

6 Performance Evaluation for CNNs **101**
6.1 Tornado Prediction (best MYRORSS and GridRad models) . . . 102
6.2 Front Detection . . . 123
6.2.1 Best CNN . . . 123
6.2.2 Best Object-conversion Method . . . 133

7 Interpretation of CNNs **141**
7.1 Methods . . . 142
7.1.1 The Permutation Test . . . 142
7.1.2 Saliency Maps . . . 144
7.1.3 Class-activation Maps (CAM) . . . 147
7.1.4 Backward Optimization (BWO) . . . 149
7.2 Tornado Prediction . . . 151
7.2.1 GridRad Model . . . 152
7.2.2 MYRORSS Model . . . 164
7.3 Front Detection . . . 174

8 Climatology of Warm and Cold fronts **189**
8.1 Methods . . . 189
8.2 Overall Averages . . . 191
8.2.1 Cold-front Frequency . . . 191
8.2.2 Warm-front Frequency . . . 193
8.2.3 Front Length . . . 197
8.3 ENSO-relative Variability . . . 197
8.3.1 Winter . . . 200
8.3.2 Spring . . . 205
8.3.3 Summer and Fall . . . 208
8.4 Long-term Trends . . . 209
8.5 Sensitivity Tests . . . 214

9 Conclusions **218**

Reference List **224**

Appendices **243**

A Estimation of Storm Velocity **244**

B Additional Data **246**

C Algorithms for Permutation Test **255**

D Direct Comparison of MYRORSS and GridRad Interpretation Results **259**
D.1 Methods . . . 259
D.2 Results . . . 260

E Supercomputer Configurations for Training CNNs **269**

List of Tables

3.1 Summary of tornado-predictors 38
3.2 Coefficients for estimation of melting level 46
3.3 Training, validation, and testing periods for tornado prediction 59
3.4 List of ERA5 variables used 60
3.5 Class frequencies for front detection 65
3.6 Training, validation, and testing periods for front detection 65

5.1 Experimental hyperparameters for tornado prediction 88
5.2 Contingency table for binary classification 91
5.3 Experimental hyperparameters for front-detection CNN 93
5.4 Experimental hyperparameters for object conversion 99

6.1 Definitions of extreme cases for tornado prediction 118
6.2 Definitions of extreme cases for front detection 129

7.1 Pearson correlations among top GridRad variables 154

8.1 Definition and sample size for each ENSO phase 190

B.1 List of GridRad days and domains, part 1 246
B.2 List of GridRad days and domains, part 2 247
B.3 List of GridRad days and domains, part 3 248
B.4 List of GridRad days and domains, part 4 249

C.1 Algorithm for single-pass forward permutation test 255
C.2 Algorithm for multi-pass forward permutation test 256
C.3 Algorithm for single-pass backward permutation test 257
C.4 Algorithm for multi-pass backward permutation test 258

D.1 Matched extreme cases for tornado prediction 263

List of Figures

2.1 Schematic of linear model for mid-level mesocyclogenesis 12
2.2 Schematic of mesocyclogenesis and tornadogenesis 15
2.3 Schematic of Norwegian cyclone model 29
2.4 Schematic of global circulation 32
2.5 Schematics of El Niño and La Niña 34

3.1 Overview of 5-minute GridRad data 41
3.2 Side-by-side comparison of MYRORSS and GridRad data 42
3.3 Flow chart for echo classification 45
3.4 Storm detection 48
3.5 Flow chart for prelim_tracking algorithm 51
3.6 Spaghetti plot of storm tracks 52
3.7 Flow chart for tornado attribution 55
3.8 Land boundary of the CONUS, eroded by 10 km 56
3.9 Predictors for matched MYRORSS and GridRad examples 57
3.10 Short time series of WPC fronts 62
3.11 Overview of input data for front detection 63
3.12 Masking of grid cells for CNN development 66
3.13 Patch for front detection 67

4.1 Example of CNN architecture 70
4.2 Schematic for univariate convolution in two spatial dimensions 71
4.3 Schematic for multivariate convolution in two spatial dimensions 72
4.4 ReLU with various negative slopes 74
4.5 Convolutional block with one convolutional layer 76
4.6 Schematic for maximum-pooling in two spatial dimensions 78

4.7 Schematic for dense layers 79

5.1 Theoretical schematic of overfitting 82
5.2 Architecture of one GridRad model 85
5.3 Architecture of one MYRORSS model 86
5.4 Architecture of sounding-feature-detector 87
5.5 Example of data augmentation 90
5.6 Object-conversion procedure for fronts 98

6.1 AUC and CSI for GridRad models on validation data 103
6.2 AUC and CSI for MYRORSS models on validation data 104
6.3 Performance of MYRORSS and GridRad models on testing data 106
6.4 Performance of MYRORSS and GridRad models on strong tornadoes only . 108
6.5 Same as Figure 6.3 but only for matched examples in testing data 109
6.6 Monthly and hourly performance of the GridRad model on testing data . . . 111
6.7 Monthly and hourly performance of the MYRORSS model on testing data . 112
6.8 Regional performance of the GridRad model on testing data 114
6.9 Regional performance of the MYRORSS model on testing data 115
6.10 Regional performance of the GridRad model on testing data, smoothed . . . 116
6.11 Regional performance of the MYRORSS model on testing data, smoothed . 117
6.12 Radar fields for extreme GridRad cases 121
6.13 Radar fields for extreme MYRORSS cases 122
6.14 Soundings for extreme GridRad cases 124
6.15 Soundings for extreme MYRORSS cases 125
6.16 AUC for front-detection models on validation data 127
6.17 ROC curve on validation data for selected front-detection model 128
6.18 Spatial distribution of extreme cases for selected front-detection model . . . 130
6.19 Extreme warm-frontal cases for selected front-detection model 132
6.20 Extreme cold-frontal cases for selected front-detection model 134

6.21 Best correct nulls for selected front-detection model 135
6.22 CSI for various front-probability thresholds on validation data 137
6.23 Weighted CSI for various front-probability thresholds on validation data . . 138
6.24 Performance diagram for front detection on testing data 139
6.25 Regional performance of front detection on testing data 140

7.1 Results of permutation test for GridRad model 153
7.2 Saliency maps for extreme GridRad cases 156
7.3 Class-activation maps for extreme GridRad cases 158
7.4 Backward optimization for best hits in GridRad model 159
7.5 Unconstrained backward optimization for best hits in GridRad model . . . 160
7.6 Backward optimization for worst misses in GridRad model 162
7.7 Sanity checks for saliency maps on best hits in GridRad model 163
7.8 Results of permutation test for MYRORSS model 166
7.9 Pearson correlations in MYRORSS data 167
7.10 Saliency maps for extreme MYRORSS cases 169
7.11 Class-activation maps for extreme MYRORSS cases 170
7.12 Class-activation maps for best correct nulls in MYRORSS model 171
7.13 Backward optimization for best hits in MYRORSS model 173
7.14 Backward optimization for worst misses in MYRORSS model 175
7.15 Sanity checks for saliency maps on best hits in MYRORSS model 176
7.16 Results of permutation test for front-detection model 178
7.17 Pearson correlations among predictors for front detection 179
7.18 Class-activation maps for extreme warm fronts 181
7.19 Class-activation maps for extreme cold fronts 183
7.20 Class-activation maps for best correct nulls 184
7.21 Backward optimization on best hits for cold fronts 185
7.22 Backward optimization on worst misses for cold fronts 187

7.23 Sanity checks for saliency maps on best hits for cold fronts 188

8.1 Frequency of cold fronts in each season 192
8.2 Frequency of warm fronts in each season 194
8.3 Examples of warm fronts in eastern tropical Pacific 195
8.4 Average cold-front length in each season 198
8.5 Average warm-front length in each season 199
8.6 Front frequency by ENSO phase during winter 201
8.7 Examples of warm fronts near Hawaii 204
8.8 Front frequency by ENSO phase during spring 207
8.9 Front frequency by ENSO phase during summer 210
8.10 Front frequency by ENSO phase during fall 211
8.11 Linear trend in front frequency during each season 212
8.12 Front frequency during strong winter El Niño for different minimum lengths 216
8.13 Front frequency during strong winter El Niño for different minimum CNN probabilities 217

A.1 Predecessors for estimation of storm velocity 245

B.1 POD, FAR, and frequency bias for GridRad models on validation data . . . 249
B.2 POD, FAR, and frequency bias for MYRORSS models on validation data . 250
B.3 Sanity checks for saliency maps on worst misses in GridRad model 251
B.4 Sanity checks for saliency maps on best correct nulls in GridRad model . . 252
B.5 Sanity checks for saliency maps on worst misses in MYRORSS model . . . 253
B.6 Sanity checks for saliency maps on best correct nulls in MYRORSS model . 254

D.1 Performance of chosen upscaled-GridRad model on testing data 261
D.2 Results of permutation test for upscaled-GridRad model 262
D.3 Saliency maps created by MYRORSS model for matched extreme cases . . 265

D.4 Saliency maps created by upscaled-GridRad model for matched extreme cases . . . 266
D.5 Class-activation maps created by MYRORSS model for matched extreme cases . . . 267
D.6 Class-activation maps created by upscaled-GridRad model for matched extreme cases . . . 268

Abstract

This dissertation describes the application of convolutional neural networks (CNN), a type of deep-learning method, to high-impact weather. CNNs are specially designed to learn directly from spatial grids, which improves both skill and interpretability.

Specifically, I develop and test CNNs for two tasks. The first is tornado prediction, where the two CNNs predict next-hour tornado occurrence for a given storm, using datasets similar to those used by forecasters in real-time operations. The tornado models achieve an area under the receiver-operating-characteristic curve (AUC) of 0.94 and critical success index (CSI) of $\gtrsim 0.30$. This is competitive with a machine-learning (ML) model currently used in operations, which suggests that the CNNs would also be suitable for operations. Novel ML-interpretation methods highlight the importance of a deep reflectivity core and strong mesocyclone, as well as low-level instability and wind shear in the surrounding environment. Also, interpretation methods suggest that a rear-flank downdraft with too much precipitation and negative buoyancy can lead to tornadogenesis failure, which corroborates some previous literature.

The second task is front detection, where the CNN draws warm and cold fronts in reanalysis data. When evaluated on human-identified fronts, the CNN achieves a CSI of 0.29 and frequency bias of 2.03. Although this is a high frequency bias, visual analysis suggests that the humans often miss fronts, so this "overdetection" is considered acceptable. I use the CNN-detected fronts to create a 40-year climatology over North America and the surrounding oceans. To my knowledge, this is the only climatology that separates fronts by type (warm or cold), one of only two climatologies that investigate variability with respect to the El Niño – Southern Oscillation (ENSO) over a continent-scale or larger domain, and one of only three that investigate long-term trends over a continent-scale or larger domain. I find that fronts are most common in the mid-latitude cyclone track, which migrates poleward from winter to summer, equatorward during El Niño, and poleward during La Niña. Also, I find that the cyclone track appears to be migrating poleward as a consequence of

global warming. These results are broadly consistent with previous climatologies, although there are some discrepancies and interesting results that warrant future investigation.

To summarize, my contributions to atmospheric and data science include creating tornado models that perform competitively with the state of the art, using novel interpretation methods to understand physical relationships learned by the tornado models, creating a front-detection model that appears to overcome some deficiencies of human labels, using the front-detection model to create a climatology, and identifying similarities and discrepancies to the few pre-existing climatologies. Overall, this dissertation demonstrates that deep learning can be used to advance both the prediction and understanding of high-impact weather.

Chapter 1

Introduction

Machine learning (ML) has been used widely in atmospheric science, with some examples given in Section 2.1.2. In particular, convolutional neural networks (CNN) have recently gained popularity in the field. CNNs are specially designed to learn from spatial grids and often contain many layers, which qualifies them as a deep-learning method (Section 1.1.4 of Chollet 2018). In traditional ML, spatial grids must be transformed into scalar features, which become the direct inputs to the model. Examples are principal components, spatial statistics (such as means and standard deviations), and raw grid-point values (where each value in the grid is treated as a scalar feature, with no regard to spatial structure). Inevitably, since the transformation to scalar features is done prior to ML, it does not optimally exploit the spatial information available. In contrast, spatial grids are fed directly into a CNN, which simultaneously learns to transform the grids into features and the features into predictions. This generally improves skill (Krizhevsky et al. 2012; Dieleman et al. 2015; Silver et al. 2016) and interpretability, the latter because the model's behaviour can be projected onto the input space and viewed as an image, which is a natural way for humans to process data.

The following is a summary of CNN applications in atmospheric science. Liu et al. (2016) used a small CNN (two convolutional layers and two pooling layers, or 2C + 2D; layer types are explained in Chapter 4) to detect tropical cyclones, atmospheric rivers, and fronts in gridded reanalysis data. They used patch classification, where each grid cell is classified independently, instead of semantic segmentation, where all grid cells are classified at once. For nearly balanced testing sets – event frequencies[1] of 50% for tropical cyclones, 48.9% for atmospheric rivers, and 46.3% for fronts – they obtained accuracies

[1] In this context, event frequency is the percentage of patches for which the center grid cell is part of an object of the given type.

of 99%, 90%, and 89.4% respectively. Their ground truth (labels used to train the CNN) was provided by a combination of numerical algorithms and manual analysis. Racah et al. (2017) performed similar work but with semantic segmentation instead of patch classification. Differences in performance are unclear, because Racah et al. (2017) used a different evaluation score (intersection over union) than Liu et al. (2016). Kurth et al. (2018) built on the work of Racah et al. (2017) by training a much deeper CNN (over 50 layers) on 27 360 graphical processing units (GPU), achieving 1.13×10^{18} floating-point operations per second (FLOPS) and winning the 2018 Gordon Bell prize for computing. Lagerquist et al. (2019, henceforth L19) used a medium-sized CNN (6C + 2D) for patch classification of warm and cold fronts. Their ground truth was provided by meteorologists at the Weather Prediction Center (WPC), and their Figure 13 suggests that the CNN strongly outperforms numerical frontal analysis (defined in Section 2.2.2), which has been commonly used for front detection since the 1960s.

Outside of feature detection, Wang et al. (2016) used a small CNN (2C + 1D) to estimate sea-ice concentration from synthetic-aperture radar imagery. They achieved reliable estimates for most concentration values, as shown in their Figure 4. Wimmers et al. (2019) used a large CNN (15C + 3D) to estimate tropical-cyclone intensity (maximum sustained wind speed) from microwave imagery taken by satellites. They achieved a root mean squared error of $\sim$5 m s^{-1}, which is nearly state-of-the-art performance. Gagne et al. (2019) employed a small CNN (3C + 1D) to predict the occurrence of large hail (diameter $\geq$ 25.4 mm) for a simulated storm at one-hour lead time. The CNN, trained with full 2-D grids, outperformed a logistic-regression model trained with spatial statistics or principal components. Also, they found that certain neurons in the CNN appear to encode well known storm types: supercells, bow-echo storms, and pulse storms. Motivated by this discovery and recent work in ML interpretation, McGovern et al. (2019) used several interpretation methods to understand the physical relationships learned by CNNs that predict large hail and tornadoes. Reichstein et al. (2019) and Gil et al. (2019) have called

for a vast expansion of efforts to incorporate deep learning into the geosciences, including model interpretation, which should lead to more trustworthy and widely adopted models and possibly new knowledge discovery.

The remainder of this chapter summarizes my results, then my overall contributions to atmospheric and data science. Work presented in this dissertation is published in McGovern et al. (2019), L19, and McGovern et al. (2020). Also, two papers featuring work in this dissertation have been conditionally accepted (Lagerquist et al. 2020a,b).

I develop and test CNNs for two tasks, using the Python package Keras (Chollet et al. 2020). The first is tornado prediction, where the CNN predicts next-hour tornado occurrence for a given storm, using datasets similar to those available to forecasters in real-time. I compare results to the ProbSevere model discussed in Section 2.1.2, which is currently used in real-time. The second task is front detection, where the CNN draws warm and cold fronts in gridded reanalysis data. I use the CNN to create a 40-year climatology over North America and the surrounding oceans, which I compare to the previous climatologies discussed in Section 2.2. Also, I use several interpretation methods to understand the physical relationships learned by the CNNs. The remainder of this chapter summarizes results.

I predict tornadoes with two CNNs: one trained with radar data from MYRORSS, the other trained with radar data from GridRad. When evaluated on the testing data, both models achieve an area under the receiver-operating-characteristic curve (AUC) of 0.94 and critical success index (CSI) of $\gtrsim 0.30$. This is competitive with an operational model called ProbSevere, which suggests that the CNNs would also be suitable for operations. Also, the results for strong (EF-2+) tornadoes are slightly better, which suggests that model skill improves with tornado strength. For both the MYRORSS and GridRad models, I examine four sets of storms, representing the best and worst model predictions. These are the best hits (tornadic storms with high forecast probabilities), worst false alarms (non-tornadic storms with high forecast probabilities), worst misses (tornadic storms with low forecast probabilities), and best correct nulls (non-tornadic storms with low forecast probabilities).

The best hits and worst false alarms are supercells with deep reflectivity cores and deep, strongly rotating mesocyclones; the worst misses are mostly cells in quasi-linear convective systems (QLCS); and the best correct nulls are mostly non-dominant cells in mesoscale convective systems (MCS). For both the MYRORSS and GridRad models, radar images are very similar between the best hits and worst false alarms, which suggests that these datasets have trouble discriminating between strong tornadic and non-tornadic supercells. Human meteorologists have the same problem, as most of the worst false alarms are associated with National Weather Service tornado warnings. Interpretation methods highlight the importance of a deep reflectivity core and strong mesocyclone, as well as low-level instability and wind shear in the surrounding environment. Also, interpretation methods suggest that a precipitation-loaded rear-flank downdraft, with too much negative buoyancy, can lead to tornadogenesis failure.

For front detection, I train the CNN to reproduce fronts drawn by meteorologists at the Weather Prediction Center (WPC). The CNN performs best when trained with predictors (thermal and wind fields) at two vertical levels, which highlights the importance of 3-D structure. When evaluated on testing data with a 100-km matching distance, the CNN achieves a probability of detection (POD) of 0.69, false-alarm ratio (FAR) of 0.66, CSI of 0.29, and frequency bias of 2.03. Although the FAR and bias are high, visual analysis suggests that the WPC often misses fronts, so this "overdetection" is considered acceptable. Common failure modes are WPC fronts in anticyclones and deformation zones, featuring weak thermal advection, which causes the CNN to not identify these fronts. However, an interpretation method called class-activation maps suggests that the CNN can "correct" these labels by highlighting a nearby area with greater thermal advection that appears more front-like.

The CNN-based climatology is used to analyze overall averages, variability with respect to the El Niño – Southern Oscillation (ENSO), and long-term trends. Overall averages suggest that fronts are most common in the mid-latitude cyclone track, which migrates about

10° poleward from winter to summer, due to the global annual heating cycle. Averages also show many regional effects, such as cold-air damming in the lee of the Appalachians and fronts near land-sea boundaries. ENSO effects are greatest in the winter and spring, and greater for El Niño than La Niña, consistent with previous literature on ENSO teleconnections in general. El Niño (La Niña) leads to a southward (northward) displacement of the mid-latitude cyclone track, due to a southward (northward) displacement of the subtropical Pacific jet. During winter especially, El Niño also causes an eastward extension of the subtropical Pacific jet into the Atlantic, increasing cyclone and front activity in the southern continental United States. ENSO effects in the summer and fall are mostly not statistically significant, but they are consistent in suggesting a southward (northward) shift during El Niño (La Niña). As for long-term trends, the mid-latitude cyclone track appears to have migrated poleward over the last 40 years, consistent with previous studies. These results are broadly consistent with the few pre-existing climatologies, although there are some discrepancies and interesting results that should be targeted for future investigation.

The following is a summary of my contributions to atmospheric and data science. I create tornado-prediction models that perform competitively with a model currently used in operations. I use novel interpretation methods to identify relationships learned by these models, many of which corroborate previous literature. Also, I create a front-detection model that appears to overcome some deficiencies of the human labels used to train it. I thoroughly quantify the performance of the model, establishing a baseline for future studies.[2] I use the model to create a climatology and compare this climatology to the few pre-existing ones, identifying many similarities and also discrepancies that require future investigation. Thus, for two very different weather phenomena, I demonstrate that deep learning can advance both prediction and understanding.

[2]Previous studies in manual and numerical frontal analysis contain no objective evaluation, because they have no separate "ground truth" dataset. Also, as discussed at the beginning of Section 6.2, previous studies using CNNs for front detection report few details on their performance.

The remainder of this dissertation is organized as follows. Chapter 2 is the literature review, covering the physical mechanisms behind tornadoes and fronts, as well as previous efforts in predicting tornadoes and determining the climatology of fronts. Chapter 3 describes the input data and pre-processing for both prediction tasks; Chapter 4 describes the inner workings of CNNs; Chapter 5 describes experiments used to find the best CNN for each task; Chapter 6 evaluates performance of the CNNs; Chapter 7 discusses physical relationships learned by the CNNs; Chapter 8 presents and analyzes the CNN-derived climatology of fronts; and Chapter 9 summarizes and discusses opportunities for future work.

Chapter 2

Literature Review

This chapter reviews the literature on tornadoes and fronts.

2.1 Tornadoes

This section discusses the mechanisms responsible for tornadogenesis and recent work in tornado prediction.

2.1.1 Tornadogenesis Mechanisms

Tornadoes are one of the costliest weather disasters, in terms of both life and property, in the United States (Insurance Information Institute 2019). According to the American Meteorological Society Glossary (AMS 2013), a tornado is a vortex connected to the surface and a cumuliform cloud, often visible as a funnel cloud. Traditionally, tornadoes have been classified into types I and II (Davies-Jones et al. 2001, henceforth DJ01). However, Agee and Jones (2009, henceforth AJ09) proposed a scheme with three types, which I will adopt here.[1] Type-I tornadoes form in a mesocyclone, the rotating core of a supercell thunderstorm; type-II tornadoes form in a quasi-linear convective system (QLCS); and type-III tornadoes are the miscellaneous category. These include gustnadoes, which form along the storm's outflow boundary far from any mesocyclone (Agee and Jones 2009); cold-core funnels, which typically form in a moderate thunderstorm associated with a deep extratropical cyclone (Davies 2006); and landspouts, which are often associated with a background[2]

[1] AJ09's type I corresponds to DJ01's type I, and AJ09's type III corresponds to DJ01's type II. The difference between the two schemes is that AJ09 allow for QLCS tornadoes, their type II.

[2] In this section, "background" refers to a property of the near-storm environment that existed before the storm and does not account for modification by the storm.

wind-shift line, like those that frequently occur along the Front Range of Colorado (Caruso and Davies 2005). Cold-core funnels and landspouts are caused mainly by tilting and stretching of background vorticity, rather than storm-generated vorticity. Because almost all strong tornadoes (rated EF-2 or higher) are type I (Table 2 of Smith et al. 2012), the rest of this review focuses on type I, with a brief discussion of QLCS tornadoes at the end.

Supercells, the storms responsible for type-I tornadoes, can form in a wide variety of large-scale environments. Thus, it is difficult to characterize a "typical" supercell environment. Instead, forecasters and researchers typically focus on the ingredients necessary for supercells: instability, moisture, trigger, and wind shear. The first three ingredients are required for any thunderstorm, while wind shear is required specifically for supercells. Unless otherwise noted, the following discussion of these ingredients is based on Section 8.3 of Wallace and Hobbs (2006).

"Instability" refers to static instability in the vertical dimension, which is determined by the thermal profile (temperature and humidity). The concept is related to buoyancy and lapse rates. Buoyancy[3] is typically defined as $g\frac{T_p - T_e}{T_p}$, where T_p is the temperature of an air parcel (Kelvins or K); T_e is the temperature of the surrounding environment (K); and g is the gravitational constant ($\sim$9.81 m s^{-2}). When $T_p > T_e$, buoyancy is positive, representing an upward force on the air parcel. As the parcel rises, its pressure decreases, due to a decrease in the weight of the overlying air column. Assuming that the motion is adiabatic (does not involve an exchange of heat with the environment), this decrease in pressure causes the parcel to expand and cool. If the parcel is not saturated with water vapour, it cools at the dry adiabatic lapse rate (DALR), about 9.8 K km^{-1}. Meanwhile, the environmental lapse rate (ELR) is the rate at which the environment cools with height. If

[3] A more accurate version of buoyancy replaces temperature with virtual temperature, which accounts for density variations due to humidity. Virtual temperature is defined as $\frac{T}{1-(1-\varepsilon)\frac{e}{p}}$, where T is temperature (K); $\varepsilon \approx 0.622$ is the dry-air gas constant over the water-vapour gas constant; e is the partial pressure of water vapour (or just "vapour pressure"; Pa); and p is the total air pressure (Pa). Virtual temperature increases with e, so more humid air, like warmer air, has a higher virtual temperature.

ELR $>$ DALR, the parcel's temperature surplus ($T_p - T_e$) will increase as it rises, leading to an increase in buoyancy and upward acceleration (neglecting other forces). If ELR $<$ DALR, the parcel's temperature surplus will decrease, eventually becoming zero. At this height, called the parcel's equilibrium level, buoyancy also becomes zero and the parcel undergoes Brunt-Väisälä oscillations rather than continued upward motion.

But there is one problem: superadiabatic lapse rates (ELR $>$ DALR) generally do not exist in the atmosphere, so unsaturated air parcels rarely achieve positive buoyancy. This is one reason that moisture, the second supercell ingredient, is required. As a saturated parcel rises, its saturation vapour pressure (which is an increasing function of temperature) decreases. This causes the parcel to become supersaturated, which forces water vapour to condense into the liquid phase. This releases latent heat into the parcel, which somewhat offsets the cooling caused by adiabatic expansion. Thus, a saturated parcel cools less quickly than an unsaturated parcel – in other words, the saturated adiabatic lapse rate (SALR) is less than the DALR. A common situation is that SALR $<$ ELR $<$ DALR, meaning that the atmosphere is unstable only to saturated ascent, or "conditionally unstable." Thus, saturated parcels often maintain their temperature surplus ($T_p - T_e$) and positive buoyancy as they rise over great depths, sometimes up to the tropopause. Much of the water that condenses out during this process, falls to the surface as precipitation.

The third supercell ingredient is a trigger, or mechanism of forced (non-buoyant) ascent. This is often required to lift unsaturated air until it becomes saturated and achieves positive buoyancy. Two special waypoints along this ascent are the lifting condensation level (LCL) and level of free convection (LFC). The LCL is the height at which adiabatic cooling causes the parcel to become saturated, and the LFC is the height at which the parcel first achieves a temperature surplus ($T_p > T_e$), leading to positive buoyancy. There are many possible triggering mechanisms, such as lifting over a warm front, cold front, or dryline; lifting over an outflow boundary from another storm; upslope flow over a mountain; sea-breeze convergence over the Florida peninsula; etc.

The fourth ingredient, wind shear, allows the storm to develop a mesocyclone and thereby become a supercell. A typical mesocyclone has a width of ∼5 km, height of ∼10 km, and maximum vertical vorticity of ∼0.01 s^{-1} (DJ01). Mesocyclones begin in the mid-troposphere, around 5 km above ground level (AGL), and may eventually build towards the surface. The next few paragraphs describe mid-level and then low-level mesocyclogenesis, based on DJ01 unless otherwise noted. The review of mid-level mesocyclogenesis is especially reliant on DJ01, as there is wide agreement in the literature on this process (see review papers by Markowski and Richardson 2009, 2014).

"Wind shear" refers to vertical shear of the background horizontal wind, expressed as $\frac{\partial \bar{u}}{\partial z}$ for the zonal component and $\frac{\partial \bar{v}}{\partial z}$ for the meridional component.[4] Wind shear generates horizontal vorticity, and mid-level mesocyclogenesis occurs via tilting of this horizontal vorticity into the vertical dimension. The Earth's vorticity is neglected, because it is typically a small fraction of the wind-shear-generated vorticity. For example, assuming unidirectional shear from 0 m s^{-1} at the surface to 50 m s^{-1} at 10 km AGL, the resulting vorticity is 50×10^{-4} s^{-1}, while the Earth's vertical vorticity varies from zero at the equator to 1.46×10^{-4} s^{-1} at the poles. Much of mid-level mesocyclogenesis can be explained by a linear model, including only linear terms in the Boussinesq equations for dry inviscid motion (Equations 1-5 of Davies-Jones 1984, henceforth DJ84). A "linear" term is one that includes only interaction of the storm with the background environment, whereas a non-linear term includes intra-storm interactions or interactions between the storm and the storm-modified (not background) environment. The Boussinesq approximation ignores density variations in all terms not multiplied by the gravitational constant g. An important result of these equations (Equation 11 of DJ84) is reiterated below:

$$\zeta' = \frac{\partial \bar{u}}{\partial z}\frac{\partial h'}{\partial y} - \frac{\partial \bar{v}}{\partial z}\frac{\partial h'}{\partial x}, \tag{2.1}$$

[4]Background quantities will be denoted with overbars, and perturbation quantities (deviations from the background, caused by the storm) will be denoted with primes, such as u' for perturbation zonal wind.

where $\frac{\partial \bar{u}}{\partial z}$ and $\frac{\partial \bar{v}}{\partial z}$ represent the aforementioned wind shear (m s^{-1}); h' is the vertical displacement of an air parcel from its background level (m); and ζ' is the resulting vertical vorticity (s^{-1}), where positive values are cyclonic. Assuming that the wind shear is entirely westerly[5] ($\frac{\partial \bar{u}}{\partial z} > 0$ and $\frac{\partial \bar{v}}{\partial z} = 0$), ζ' is positive south of the updraft and negative north of the updraft. Thus, tilting of environmental vortex tubes leads to counter-rotating vortices, with a cyclonic vortex to the south and anticyclonic vortex to the north. If the vortices reach sufficient intensity, they are called a mesocyclone and mesoanticyclone, respectively.

However, neither vortex is collocated with the updraft (Figure 2.1b), so this schematic does not explain why supercell updrafts rotate as a whole. To this end, DJ84 separated the background horizontal vorticity into streamwise and crosswise components, oriented parallel and perpendicular to the background wind, respectively. Streamwise vorticity is created by directional wind shear, while crosswise vorticity is created by unidirectional wind shear (henceforth, speed shear). DJ84 derived an approximation (their Equation 27) for the correlation between ζ' and w', where w' is the perturbation vertical velocity or updraft velocity. This correlation is proportional to the background streamwise vorticity and thus increases with directional wind shear. When the correlation is large enough, the updraft and mesocyclone become collocated enough that the updraft rotates as a whole, with the mesoanticyclone in a region of weaker updraft or downdraft (Figure 2.1c).

Thus, the linear model partly explains why cyclonic supercells are more common than anticyclonic supercells. However, the linear model does not account for some key observations, especially storm-splitting. The remaining discussion of mid-level mesocyclogenesis uses the non-linear model, which includes non-linear terms in the aforementioned Boussinesq equations.

Under speed shear only (the "straight hodograph" case in DJ01), two non-linear effects cause the updraft to split. First, precipitation accumulates in the updraft and generates a downward drag force, causing the air to descend as a downdraft. Second, vorticity in the

[5]With no loss of generality, because if the wind shear were not westerly, a coordinate transformation could be applied to make it point in the $+x$-direction nonetheless.

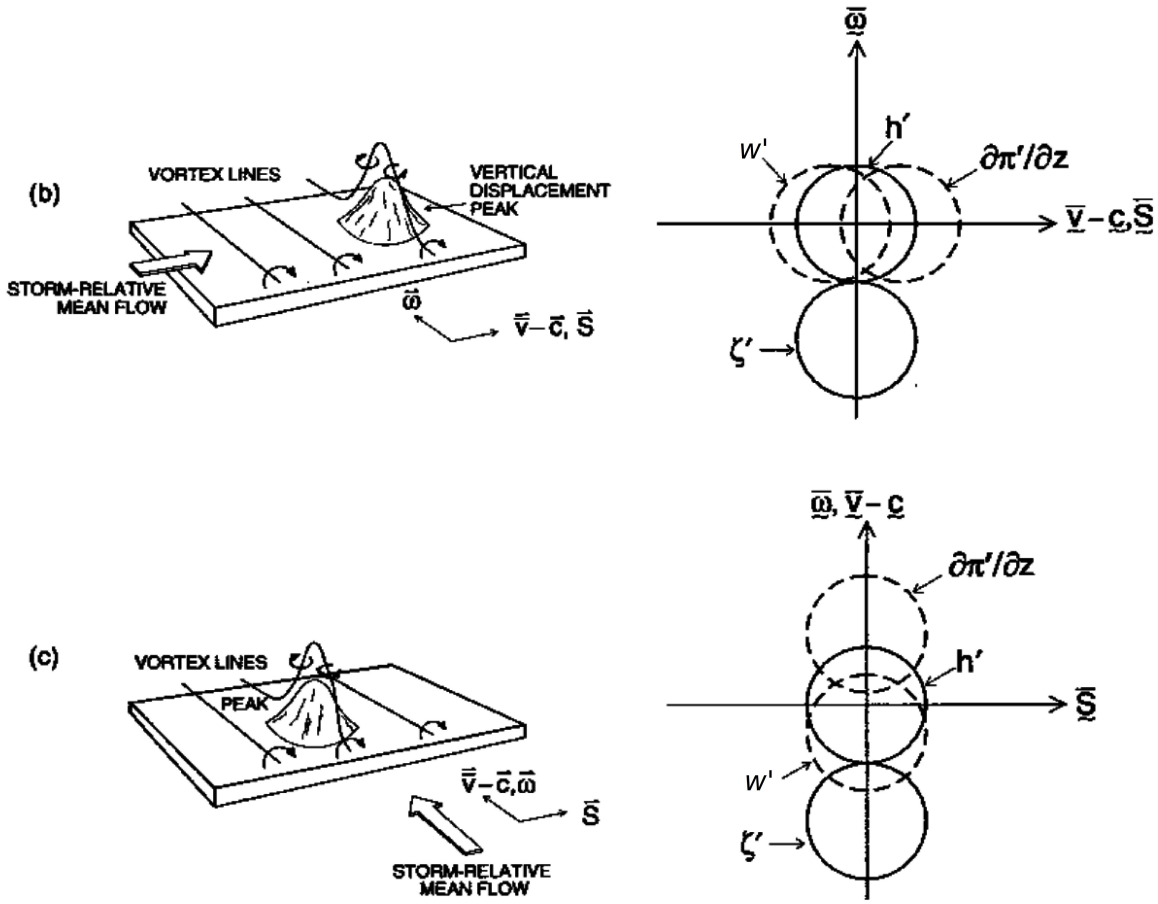


Figure 2.1: Schematic of linear model for mid-level mesocyclogenesis (Figure 5.7b-c in DJ01; panel a is omitted). In the diagrams on the left, $\vec{v}$ is the background wind; $\vec{c}$ is storm motion; $\vec{\omega}$ is background vorticity; and $\vec{S}$ is vertical wind shear. In the graphs on the right, w' is perturbation vertical velocity; h' is perturbation vertical displacement; ζ' is perturbation vertical vorticity; and $\frac{\partial \pi'}{\partial z}$ is the vertical perturbation pressure gradient. The w' circle is the updraft, and the ζ' circle is the mesocyclone. The mesoanticyclone is not shown. The center of each circle is the location of the maximum. Overlapping/touching/disjoint circles have positive/zero/negative correlation. [b] Under speed shear. The environmental wind shear is entirely westerly, leading to only crosswise environmental vorticity. [c] Under directional shear. The environmental wind veers with height, leading to streamwise environmental vorticity. Note: the figure in DJ01 labels circles as ω' instead of w', but their caption makes clear that this is an error.

mesocyclone and mesoanticyclone generates negative perturbation pressure (according to Equation 12 of Markowski 2002). Since both vortices are still confined to the mid-levels, this corresponds to an upward perturbation-pressure-gradient force (PPGF) between the surface and mid-levels. This allows the vortices to "suck" air up from the surface, encouraging new updraft growth on the vortex flanks opposite the center (where precipitation-loading has created a downdraft). This pattern of updraft growth and decay causes the southern storm to turn southward and the northern storm to turn northward, deviating from their initial eastward motion (aligned with the background wind). In other words, the cyclonic storm turns right and the anticyclonic storm turns left, consistent with real-world observations. Since the storms are no longer moving with the background, storm-relative wind deviates from Earth-relative wind. In a storm-relative reference frame, this leads to veering wind for the cyclonic storm and backing wind for the anticyclonic storm.[6] Veering and backing wind generate environmental streamwise and antistreamwise vorticity, respectively, which the updrafts tilt into cyclonic and anticyclonic vorticity. Thus, deviant storm motion allows both the mesocyclone and mesoanticyclone to intensify. However, surface friction generally causes veering in the Ekman layer (the lowest $\sim$1 km above the surface), which opposes backing near the anticyclonic storm. This is one reason that cyclonic supercells are more common than anticyclonic ones.

Supercells in directional shear are often seen as first-order modifications of this conceptual model in speed shear. In general, as directional shear (of the Earth-relative, not storm-relative, wind) increases, approaching the "circle hodograph" case discussed in DJ01, storm-splitting becomes less likely as non-linear effects become less important. Also, the mid-level mesocyclone becomes more collocated with the updraft, since streamwise vorticity (and thus correlation between ζ' and w') increases with directional shear.

Not all mid-level mesocyclones develop a low-level mesocyclone. Decades of research (Rotunno and Klemp 1985; Davies-Jones and Brooks 1993; DJ01; Markowski and

[6] "Veering" means that the wind turns clockwise with height; "backing" means that the wind turns counterclockwise with height.

Richardson 2009, 2014) suggest that a downdraft is required for low-level mesocyclogenesis, which is required for tornadogenesis. Without a downdraft, vertical vorticity would be generated only in the updraft, which contains quickly rising air, so vertical vorticity would not reach mesocyclone strength near the ground (Figure 2.2a). In the downdraft, vertical vorticity may be generated either baroclinically or barotropically. In the baroclinic mechanism, the mesocyclone draws rain around its rear side, which falls into dry (unsaturated) air outside the updraft. Both precipitation drag and negative buoyancy, the latter due to evaporative cooling, impart downward acceleration to the air, causing it to descend as a rear-flank downdraft (RFD). Air in the RFD is colder than its surroundings, creating a baroclinic zone and solenoidal circulation ("warm" and "cool" arrows in Figure 2.2a). This generates horizontal vorticity, which is tilted into positive vertical (cyclonic) vorticity on the left side of the RFD. When this air exits the RFD at the surface, it flows into the updraft, where it is "sucked up" via the aforementioned PPGF. This suction stretches the vortex tubes, amplifying low-level vertical vorticity to mesocyclone strength. Conversely, in the barotropic mechanism, low-level mesocyclogenesis occurs via inward (towards the rotation axis) and downward transport of angular momentum from aloft. The relative importance of the baroclinic and barotropic mechanisms was debated for decades (Brandes 1984; DJ01; Markowski et al. 2002; Markowski 2002; Markowski and Richardson 2009, 2014), but recent work (see Section 1 of Davies-Jones 2017) suggests that the baroclinic mechanism is dominant. However, the coldest downdrafts, which generate the most baroclinic vorticity, are often too negatively buoyant to be ingested into the updraft after exiting the downdraft. Thus, stronger tornadoes are often associated with warmer RFDs, which generate less baroclinic vorticity (Markowski et al. 2002; Markowski and Richardson 2009).

The low- and mid-level mesocyclones form a continuous vortex in the updraft, which can be intensified to tornado strength via low-level convergence (Figure 2.2b). Specifically, as air approaches the rotation axis, conservation of angular momentum dictates that its tangential velocity must increase. Tornadogenesis can occur in two ways, labeled modes 1

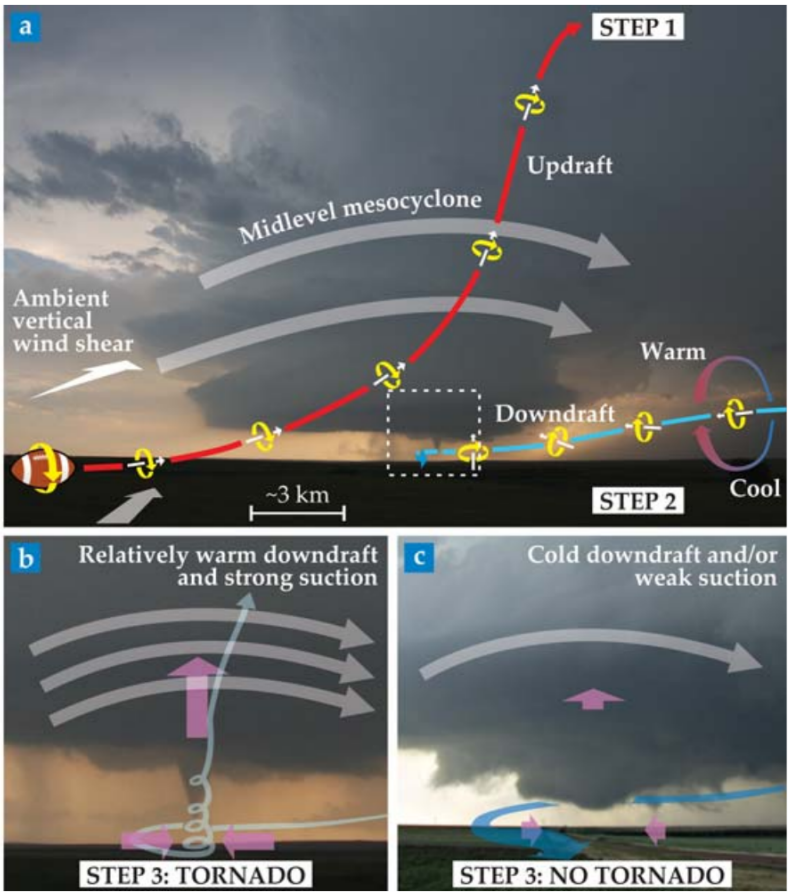


Figure 2.2: Schematic of mesocyclogenesis and tornadogenesis (Figure 2 in Markowski and Richardson 2014). [a] Mesocyclogenesis. Step 1 is mid-level mesocyclogenesis, and step 2 is low-level mesocyclogenesis. Thick grey arrows show the lower- and upper-level winds; white arrows show the orientation of the vorticity vector; and yellow arrows show the sense of rotation. Red and blue lines show the trajectories of updraft and downdraft parcels, respectively. The dashed white box is the area shown in panels b-c. [b] Mode-2 tornadogenesis, where air approaches the axis of rotation faster at the surface than aloft. Thick grey arrows show the upper-level wind; the thin grey curve shows the trajectory of an air parcel that enters the tornado; pink lines show contraction of the vortex by low-level convergence and stretching by upward motion. [c] Tornadogenesis failure. Thick grey and pink arrows are as in panel b. The blue arrow shows the trajectory of downdraft parcels, which are too negatively buoyant to be ingested into the updraft after exiting the downdraft.

and 2 (not to be confused with types I, II, and III). In mode 1, air approaches the rotation axis faster aloft. The mesocyclone is in cyclostrophic balance (where the horizontal pressure-gradient force balances the centrifugal force), making it stable to radial displacements. In other words, the mesocyclone resists horizontal motion into and out of the mesocyclone, causing it to act like a solid barrier. This allows the vortex to build downward via the dynamic pipe effect (DPE), eventually reaching the surface. In mode 2, air approaches the rotation axis faster at the surface than aloft, so the tornadic vortex begins at the surface. However, most mesocyclones, even low-level mesocyclones, do not produce tornadoes (Trapp et al. 2005a). In general, these cases of "tornadogenesis failure" may occur when the RFD is too cold and undercuts the mesocyclone (Figure 2.2c; also see Trapp 1999). Also discussed in Trapp (1999), mode-1 tornadogenesis might fail when the low-level atmosphere is too stable to allow downward growth of the tornadic vortex via the DPE, and mode-2 tornadogenesis might fail when surface friction weakens low-level convergence. However, recent work (*e.g.*, Roberts et al. 2020) suggests that some surface friction is necessary for tornadogenesis. Thus, like baroclinic vorticity generation in the RFD, surface friction is likely another "Goldilocks problem."

The foregoing discussion addresses only type-I tornadoes, which form in supercell mesocyclones. However, type-II tornadoes, which form in QLCSs, account for $\sim$18% of tornadoes in the United States (Trapp et al. 2005b). QLCS tornadoes often form in mesovortices, reviewed by Schenkman and Xue (2016, henceforth SX16). Morphologically, the main difference between supercell mesocyclones and QLCS mesovortices is that the latter exist only in the low levels (estimated as 0-3 km AGL in Mahale et al. 2012). Once a supercell mesocyclone or QLCS mesovortex is in place, the mechanisms for tornadogenesis appear to be similar (SX16). Thus, the rest of this discussion focuses on genesis of the mesovortex itself, based on the review by SX16.

Three models have been proposed for mesovortex genesis. In the first (Trapp and Weisman 2003, henceforth TW03), pairs of counter-rotating vortices develop in the precipitation-loaded downdraft behind the leading convective line, due to tilting of baroclinic vorticity generated in the downdraft (Figure 8a of SX16). Thus, the TW03 model for mesovortex genesis is similar to the widely accepted model for low-level mesocyclogenesis in supercells, with two key differences. One is that the mesovortex is shallower, because the downdraft is associated with the rear-inflow jet, which is less upright than a supercell downdraft. Also, the stretching of planetary vorticity is important in the TW03 model, because unlike in supercells, its magnitude is not dwarfed by other vertical-vorticity-generation processes.[7] Because planetary vorticity is cyclonic, the cyclonic vortex strengthens while the anticyclonic one dissipates. In the second model (Atkins and St. Laurent 2009), baroclinic vorticity generated along the gust front is tilted by an enhanced updraft associated with a gust-front bulge (Figure 9a of SX16). This leads to counter-rotating vortices, and as in the TW03 model, the cyclonic vortex strengthens while the anticyclonic one dissipates. In the third model (Xu et al. 2015), surface friction generates horizontal vorticity, which is tilted into the vertical dimension by both the updraft and downdraft (Figure 11 of SX16). However, Xu et al. (2015) did not explicitly calculate the frictional contribution to vertical vorticity, instead treating it as the residual in their vorticity budget. Due to this and a paucity of other simulations involving surface friction, its importance for mesovortex genesis is unclear (SX16).

QLCS tornadoes are a major forecasting challenge (Brotzge et al. 2013; Anderson-Frey et al. 2016) for several reasons. They are less understood than supercell tornadoes; the pre-existing mesovortices exist only at low levels, so may not be adequately sampled by radar; and the average time between mesovortex genesis and tornadogenesis is only $\sim$12 minutes (SX16), compared to a much longer lead time for mid-level mesocyclogenesis.

[7] As noted in SX16, all three models agree that planetary vorticity is important for mesovortex genesis.

2.1.2 Tornado Prediction

The National Weather Service (NWS) is responsible for issuing tornado warnings and generally issues warnings at lead times up to 30 minutes (Brooks and Correia 2018), with durations up to 60 minutes (Harrison and Karstens 2017). Although the skill of NWS tornado warnings has generally improved over time, critical success index (CSI) and lead time have stagnated in the last decade (Brooks and Correia 2018). During this time, the amount of data available to forecasters has exploded – including dual-polarization radar observations, high-resolution satellite observations, and output from physical models. However, none of these datasets explicitly resolve tornadoes, so forecasters must still synthesize and translate the data into useful information, which can lead to cognitive overload (Wilson et al. 2017). This can be alleviated by explicit tornado-modeling and post-processing methods, the latter of which synthesize multi-source data into explicit tornado predictions (Karstens et al. 2018).

For non-convection-allowing models (*i.e.*, physical models that do not explicitly simulate thunderstorms), many studies have used severe-weather parameters (SWP) to predict severe thunderstorms at lead times up to several days. SWPs, which are computed from the model output, represent the meso- or synoptic-scale (depending on model resolution) environment in which storms are expected to occur. One application is using SWPs to discriminate between tornadic and primarily non-tornadic (PNT) outbreaks. According to the AMS Glossary (AMS 2012b), a tornadic outbreak includes "[m]ultiple tornadoes associated with a particular synoptic-scale system." While there is no widely accepted definition for a PNT outbreak, it is generally considered an outbreak of primarily damaging wind and hail, with few tornado reports, associated with one synoptic-scale system.

Before Shafer et al. (2009, henceforth S09), most efforts to discriminate between tornadic and PNT outbreaks used only a few case studies. S09 used 50 tornadic outbreaks (with ≥ 7 tornadoes each) and 50 PNT outbreaks (with < 7 tornadoes each). They used the Weather Research and Forecasting (WRF) model to simulate outbreaks at lead times of 1, 2,

and 3 days. The model contained five nested grids centered on the outbreak, with horizontal grid spacing decreasing from 162 km for the outermost grid to 2 km for the innermost grid. The model was initialized with the National Centers for Environmental Prediction – National Center for Atmospheric Research (NCEP-NCAR) reanalysis, which has 2.5° (synoptic-scale) horizontal spacing. For each WRF run, S09 computed the following SWP fields: lifting condensation level (LCL); bulk wind shear from 0-1, 0-3, and 0-6 km AGL; storm-relative helicity (SRH) from 0-1 and 0-3 km AGL; energy-helicity index (EHI) from 0-1 and 0-3 km AGL; bulk-Richardson-number (BRN) shear; and storm-relative flow at 2 km AGL. SRH is the height-integrated product of vertical vorticity and storm-relative wind velocity; EHI is the product of SRH and convective available potential energy (CAPE); and BRN shear is approximately the storm-relative inflow speed from 0-0.5 km AGL. To use the SWP fields as predictors, S09 subjectively examined their spatial fields for each WRF run. For each SWP field, they created a predictor variable x_j. They set $x_j = 1$ if the SWP field reached[8] the threshold over an area comparable[9] to that of the outbreak; otherwise, they set $x_j = 0$. Finally, they summed the predictors to yield a single index for tornado potential: very high ($\sum_j x_j \geq 8$), high ($\sum_j x_j \in [6, 8]$), medium ($\sum_j x_j \in [4, 6]$), and low ($\sum_j x_j \leq 4$). This index discriminated skillfully between tornadic and PNT outbreaks at all lead times.

S09 also examined the discriminative skill of each SWP. They found that the best predictors are deep-layer wind shear, low-level wind shear, and LCL. The probability of a tornadic outbreak increases, increases, and decreases with these predictors, respectively. As discussed in Section 2.1.1, deep-layer shear is important for mid-level mesocyclogenesis, while low-level shear is important for low-level mesocyclogenesis and tornadogenesis. Lower LCLs are conducive to tornadogenesis, because (a) the potential tornado has less

[8]For all SWPs other than LCL, they used a minimum threshold, assuming that tornado potential increases monotonically with the SWP value. For LCL, they used a maximum threshold, assuming that tornado potential decreases monotonically with the LCL (*i.e.*, tornado potential increases as the LCL and cloud base, which are approximately collocated, descend towards the surface).

[9]At least slightly less than.

distance to cover between the surface and cloud base; (b) lower LCLs typically coincide with greater low-level relative humidity (RH). Air with greater RH is closer to saturation and therefore has less potential for evaporative cooling. This may lead to warmer (less negatively buoyant) air in the RFD, making the RFD less likely to undercut the mesocyclone and cause tornadogenesis failure (Section 2.1.1).

Although their method had skill at all lead times, S09 identified three main shortcomings. First, it produced false negatives (misidentified tornadic outbreaks as PNT outbreaks) in low-shear, high-CAPE environments. Storms in these environments occasionally produce tornadoes, because high CAPE allows for very strong updrafts, which produce strong upward PPGFs and convergence near the surface. This convergence leads to strong upward motion and vortex-stretching (important for tornadogenesis), as well as strong low-level inflow, which may prevent the downdraft-induced outflow from surging ahead and undercutting the storm (DJ01). Second, the S09 method had strong seasonal dependence. It performed better in the summer, when most outbreaks are PNT, and worse in the spring and fall, when there is a more balanced distribution of tornadic and PNT outbreaks. Third, because the WRF model was initialized with reanalysis data (which include observations collected after the forecast time), the S09 method is a hindcast rather than a forecast. There is no guarantee that the method would perform well in forecast mode. This shortcoming applies to the three follow-up studies discussed below and is stressed by the respective authors.

The first follow-up study is Mercer et al. (2009, henceforth M09), who used ML, rather than thresholding, as a prediction method. Specifically, for each lead time, they used principal components of the SWP fields to train two ML models – logistic regression and a support-vector machine (SVM) – to discriminate between the two outbreak types. Like S09, they found that the best predictors include wind shear or storm-relative wind. They also found that parameters quantifying thermodynamic instability, like CAPE and convective inhibition (CIN), are poor predictors. This is consistent with the results of Schneider

and Dean (2008), who found that different severe-weather phenomena tend to occur with similar CAPE values. Also, for each lead time, M09 determined the best predictor set for both logistic regression and SVM. For the 1-day lead time, the only SWP in both predictor sets was 0–1-km EHI, which includes CAPE. This suggests that even though CAPE is a poor predictor on its own, it is useful when combined with wind data. However, for the 3-day lead time, 0–1-km EHI was excluded from both predictor sets. This is likely because in physical models (like the WRF, from which 0–1-km EHI was computed), the quality of near-surface fields degrades more quickly with lead time than does the quality of upper-level fields. This is because near-surface fields are more strongly influenced by the surface and boundary layers, which rely on parameterizations that make simplistic assumptions like Monin-Obukhov similarity theory (McCandless et al. 2018).

Second, Shafer et al. (2010b, henceforth S10b) accounted for the seasonal-dependence problem discussed in S09. They used two sets of outbreaks: original (as in S09 and M09) and revised. To create the revised set, they replaced summer PNT outbreaks with ones that occur in the spring and fall, when tornadic and PNT outbreaks are more difficult to discriminate. They used a similar prediction method to M09, except with only SVM and not linear regression. As expected, the SVM performed significantly worse on the revised dataset than on the original, but it still provided skillful discrimination at all lead times (1, 2, and 3 days). In addition to SWP fields, S10b used mass fields as predictors: namely sea-level pressure, 700-mb geopotential height (GPH), and 850-mb GPH. As in M09, CAPE and CIN were poor predictors. The best predictors were low-level SRH, 2-km storm-relative flow, and the three mass fields. S10b found that tornadic outbreaks are favoured by a strong low-level cyclone downstream ($\sim$east) of a mid-level (500-mb) trough. In the warm sector of the low-level cyclone, positive thermal advection (warming and moistening) is collocated with cyclonic-vorticity advection from the mid-level trough. According to quasi-geostrophic theory (Bluestein 1993), both processes lead to strengthening of the cyclone (decreasing pressure and increasing upward velocity).

Third, Mercer et al. (2012, henceforth M12) used compositing, clustering, and ML to discriminate between tornadic and PNT outbreaks. In terms of physical interpretation, they generally corroborated the findings of S10b. Also, they found that tornadic outbreaks are favoured by more amplified troughs (with stronger vorticity and meridional flow) at both 500 and 850 mb, while PNT outbreaks occur in many different 500-mb flow regimes, which average out to approximately zonal flow.

In the vein of S09, M09, S10b, and M12, some studies have used non-convection-allowing models to discriminate between major and minor severe-thunderstorm outbreaks. These include Shafer et al. (2010a, henceforth S10a) and Shafer et al. (2012, henceforth S12). Most major outbreaks are tornadic, while most minor outbreaks are PNT, which motivates the ensuing discussion of major versus minor outbreaks in a dissertation about tornadoes. Before S10a and S12, most efforts to discriminate between major and minor outbreaks used only a few case studies. S10a used 840 outbreaks, consisting of the top 30 for each year from 1979 to 2006. Most of these outbreaks ($\sim$86%) were classified as minor, making the classes unbalanced. The SWP field chosen by S10a was the significant-tornado parameter (STP): an index derived from CAPE, low-level SRH, deep-layer shear, and the LCL. They used constrained and unconstrained versions of the STP; the constrained version included only grid cells over land (rather than water) with CAPE $\geq$ 1000 J kg^{-1} and 0–1-km SRH $\geq$ 100 J kg^{-1}. Their prediction methods were simple thresholding and four ML models: linear discriminant analysis, quadratic discriminant analysis, decision trees, and SVM. The one predictor for each method was either (a) number of grid cells with constrained STP $\geq$ 1 or (b) hypothetical storm distance traveled through the area with unconstrained STP $\geq$ 1.

All methods performed skillfully, with no significant differences among them. However, S10a identified notable failure modes. Misses (major outbreaks misclassified as minor) typically occurred for PNT outbreaks over a small area, while false alarms typically occurred with mid-level flow parallel to a surface front. When the flow was parallel to a

cold front or dryline, storms moved along the narrow corridor of favourable SWP values (just ahead of the front), leading to a QLCS. As discussed in Section 2.1.1, QLCS tornadoes are difficult to predict and QLCS storms generally do not produce tornadoes (which characterize most major outbreaks). When the flow was parallel to a warm or stationary front, the area of favourable SWP values (in the warm sector) was large, leading to clustered rather than discrete storms, which generally do not produce tornadoes, especially strong tornadoes (Table 2 of Smith et al. 2012).

S12 extended the work of S10a in three ways. First, S12 used many SWP fields to create predictors, not only the STP. Second, S10a considered only the areal coverage of a thresholded SWP field (*i.e.*, STP $\geq$ 1), whereas S12 considered the areal coverage of its intersection with severe-weather reports. This helped to exclude irrelevant regions with favourable SWP values (*e.g.*, those occurring away from the outbreak, where storms did not initiate). Third, S12 trained ML models with either one or two predictors each, whereas S10a used only one predictor. However, S12 found that the second predictor did not significantly improve performance. In terms of physical interpretation, S10a and S12 found similar results. However, since S12 used multiple SWP fields, they were able to find that both STP and 0–1-km EHI are significantly better predictors than the rest. Also, like the aforementioned studies on discriminating between tornadic and PNT outbreaks, S12 found that CAPE alone is not a good predictor.

More recent work on explicit tornado prediction largely falls under the Warn-on-Forecast initiative (WoF; Stensrud et al. 2009, 2013). The main goal of WoF is to shift the current warning paradigm, from extrapolation based on current observations (warn on detection) to the use of short-range simulations from convection-allowing models (CAM). This includes creating probabilistic tornado forecasts at 0–1-hour lead time. CAMs typically have 1–4-km horizontal grid spacing, which allows them to resolve thunderstorms but not tornadoes. However, CAMs do resolve mid-level and sometimes low-level mesocyclones. For example, Yussouf et al. (2015) ran a 3-km CAM ensemble to simulate the 27 Apr 2011 tornado

outbreak in Alabama. Storms with the strongest low-level rotation, which tend to be isolated, correctly maintained supercell characteristics throughout the one-hour simulation. However, the ensemble produced many spurious small storm cells and struggled to correctly simulate cells with strong between-storm interactions. In a similar study, Wheatley et al. (2015) ran a 3-km CAM ensemble for six tornado days in spring 2013-2014, producing strong low-level rotation 30 minutes before the first tornado in most storms, including the one QLCS tornado in the dataset. A longer-term goal of WoF is to "[establish] the feasibility of explicit ensemble probabilistic prediction of tornadoes" (Snook et al. 2019). With this aim, Snook et al. (2019) ran a 50-metre CAM ensemble for one storm, with all members correctly producing a tornado and 4 of 10 members correctly producing EF5-strength winds. The main shortcoming of 3-km models is their inability to explicitly resolve tornadoes, while 50-m models are too computationally expensive to run in real-time (and may continue to be for decades).

For non-tornado-resolving CAMs, storm surrogates are often used to relate resolved quantities to tornado occurrence. The most popular surrogate is updraft helicity (UH; Kain et al. 2008), which is the height-integrated product of vertical velocity and vertical vorticity. To create a surrogate severe probability forecast (SSPF; Sobash et al. 2011), UH is thresholded to create a binary mask (0 or 1 at each grid point), which is then smoothed via kernel density estimation (KDE). The UH threshold and smoothing radius are chosen to maximize predictive skill, defined by how well the forecast matches "ground truth," which may consist of tornado reports or a proxy such as radar-derived rotation tracks. The main disadvantage of tornado reports is underreporting bias, discussed in Section 3.1.1; the main disadvantage of rotation tracks is that the radar network generally does not have sufficient resolution to track tornadoes, and so is tracking mesocyclones instead.

Sobash et al. (2011) was the first study to use SSPF, predicting the probability of any severe weather.[10] In later work, Sobash et al. (2016b) applied SSPF to a CAM ensemble,

[10]Hereafter defined as a tornado, hail with diameter $\geq$ 25.4 mm, **or** wind gust $\geq$ 25.7 m s^{-1}.

outperforming the same technique applied to a deterministic CAM. Sobash et al. (2016a) was the first study to use SSPF to discriminate between tornadic and non-tornadic storms, obtaining skillful next-day predictions at the larger smoothing radii ($\geq$ 160 km). Gallo et al. (2016) used a similar approach but incorporated near-storm environment (NSE) variables, such as the CAPE and STP, which greatly reduced overprediction of tornadoes. While the aforementioned studies focused on 1–2-day lead times, SSPF is run annually on a 3-km CAM ensemble in the Hazardous Weather Testbed (HWT; Clark et al. 2012; Gallo et al. 2017), where the focus is on 0–3-hour lead times, and skillfully predicts radar-derived rotation tracks (Skinner et al. 2018).

Another widely used post-processing approach is ML, of which some examples are cited in the above discussion on discriminating between different outbreak types. Further examples are discussed below. One of the earliest efforts using ML for tornado prediction was the NSSL Severe Weather Potential algorithm (SWP; Kitzmiller et al. 1995), a linear-regression model that predicted any severe weather in the next 20 minutes. Marzban and Stumpf (1996) used neural networks to predict tornadogenesis for a given mesocyclone in the next 20 minutes. Lakshmanan et al. (2005) and Adrianto et al. (2009) used fuzzy logic and support-vector machines, respectively, to produce a spatiotemporal tornado-probability grid for the next 30 minutes. Due to computational limitations at the time, the aforementioned studies used only radar data as predictors. Gagne et al. (2012) trained a spatiotemporal relational random forest (SRRF) with radar data, surface observations, and NSE variables from a reanalysis, to predict tornado probability for a given supercell. They found that many of the best predictors came from the surface and NSE datasets.

Trafalis et al. (2013, henceforth T13) built on the work of Marzban and Stumpf (1996), with the goal of finding a better solution to the rare-event problem. Specifically, T13 aimed to find a better way to discriminate between tornadic and non-tornadic mesocyclones in a dataset where only 7.1% are tornadic. They applied three ML models – logistic regression, SVMs, and random forests – to radar data and reanalysis-derived NSE data. To convert

probabilistic predictions (the direct output of the ML models) to deterministic ones, the common practice is to use a probability threshold p^*, such that the deterministic prediction is "yes" if probability $\geq p^*$ and "no" otherwise. Another common practice is to set p^* = 0.5. However, T13 found that the best performance was obtained by setting $p^* < 0.5$, since the models had learned to predict lower probabilities for the minority (tornadic) class. They called this the "midpoint-threshold adjustment." Also, T13 found that the best predictors (tornado probability increasing with each) are related to deep-layer shear, relative humidity, downdraft CAPE, and low-level rotation. This corroborates results from earlier observational and physical-modeling studies (Section 2.1.1), as well as studies that used non-convection-allowing models to discriminate between outbreak types (S09, M09, S10b, S10a, M12, S12).

In the work most related to my own, Cintineo et al. (2014, 2018) developed a quasi-operational algorithm called ProbSevere, which uses naïve Bayes to forecast any severe weather for a given storm. The predictors are based on radar, satellite, and lightning data, as well as NSE variables from the Rapid Refresh model. ProbSevere has run in the HWT for several years, receiving favourable feedback from forecasters. It has improved upon the median lead time of NWS tornado and severe-thunderstorm warnings but at the cost of a decrease in CSI (Cintineo et al. 2018).

2.2 Synoptic-scale Fronts

This section provides a physical description of synoptic-scale fronts and reviews work in determining their climatology.

2.2.1 Physical Description

Synoptic-scale fronts are often associated with extreme weather, including heavy precipitation (Catto and Pfahl 2013; Catto et al. 2015) and severe thunderstorms (Fawbush and Miller 1954; Miller 1959; Garner 2013). According to the AMS Glossary (AMS 2012a),

a front is a quasi-vertical transition zone between air masses with different densities. This definition assumes that density within each air mass is nearly horizontally homogeneous, with a sharp gradient in the transition zone. For synoptic-scale fronts, the transition zone has a width on the order of 100 km (though it may be as narrow as $\sim$1 km; Shapiro and Keyser 1990) and length of a few hundred to a few thousand kilometres. In other words, "synoptic-scale" fronts are actually mesoscale in the cross-front direction. Three simplifications are often made in defining synoptic-scale fronts (henceforth, fronts). First, although the technical definition involves density, fronts are almost always defined by a thermal field such as (potential) temperature, wet-bulb (potential) temperature, or equivalent (potential) temperature. See Table 1 of Hewson (1998). Second, the transition zone is usually treated as 2-D, ignoring the vertical component. Third, the transition zone is often treated as a zeroth- or first-order discontinuity (Figure 13 of Schultz and Vaughan 2011, henceforth SV11), allowing the front to be drawn as a 1-D curve.

Air masses are created by prolonged residence in a source region, where their thermal properties are determined by the radiative balance. For example, around the winter pole, the atmosphere receives no solar radiation and continually emits longwave radiation to space, leading to strong cooling. The result is a continental polar air mass. Meanwhile, over the tropical oceans, the atmosphere receives ample solar radiation, as well as longwave radiation and water vapour from the underlying surface, leading to strong warming and moistening. The result is a maritime tropical air mass. Air masses are often characterized by their wet-bulb potential temperature (Hewson 1936; Low and Hudak 1997), because it is conserved during psuedo-adiabatic processes, which occur frequently within air masses but not across their frontal interfaces.

The leading conceptual models for fronts, also the leading conceptual models for mid-latitude cyclones, are the Norwegian (Bjerknes and Solberg 1922, henceforth BS22) and Shapiro-Keyser (1990, henceforth SK90) models. In the Norwegian model, illustrated in Figure 2.3, the cyclone begins with a small-amplitude disturbance along the polar front.

The polar front is a low-level baroclinic zone associated with the polar jet stream (Figure 10.5 of SK90), and in the Norwegian model this front is initially stationary. It is now understood that the initial disturbance is often positive vorticity advection in the upper levels, downstream of an upper-level trough. Regardless, the initial disturbance develops into a cyclone, with equatorward cold advection on the west side and poleward warm advection on the east side. The cold front rotates around the low-pressure center faster than the warm front, allowing it to catch up to the warm front and form an occlusion. This process begins at the center, where the distance between the two fronts is smallest, and spreads radially outward, forming an occluded front. Along the occluded front, cold air from behind the cold front meets cool air from ahead of the warm front, forcing air from the warm sector (between the warm and cold fronts) upward due to its lower density. This upward displacement of warm air destroys low-level baroclinicity, which is the potential energy that fuels the cyclone, so occlusion begins the cyclone's dissipating stage.

Although the Norwegian model is still widely used in textbooks and research (SV11), many of its tenets have been questioned. This led in part to the Shapiro-Keyser model, illustrated in Figure 10.27 of SV90. The initial stage is similar to the Norwegian model, featuring a small-amplitude disturbance in a low-level baroclinic zone. Due to the relative orientation of the isentropes and axes of dilatation (Figure 7 of SV11), the cold front becomes meridionally oriented and moves eastward, while the warm front becomes zonally oriented and moves westward. This prevents the cold front from catching up to the warm front and producing a Norwegian-type occlusion. Also, a frontal fracture develops between the two, where the isentropes are nearly perpendicular to the axes of dilatation (northeast of the "L" in Figure 7c of SV11), which acts to separate the isentropes and weaken the potential-temperature gradient. Air parcels in the warm-frontal zone are transported around the cyclone center and inward, causing the warm front to "bend back." This process continues until the warm front wraps fully around the center, forming a warm seclusion, a mesoscale pocket of warm air surrounded by colder air.

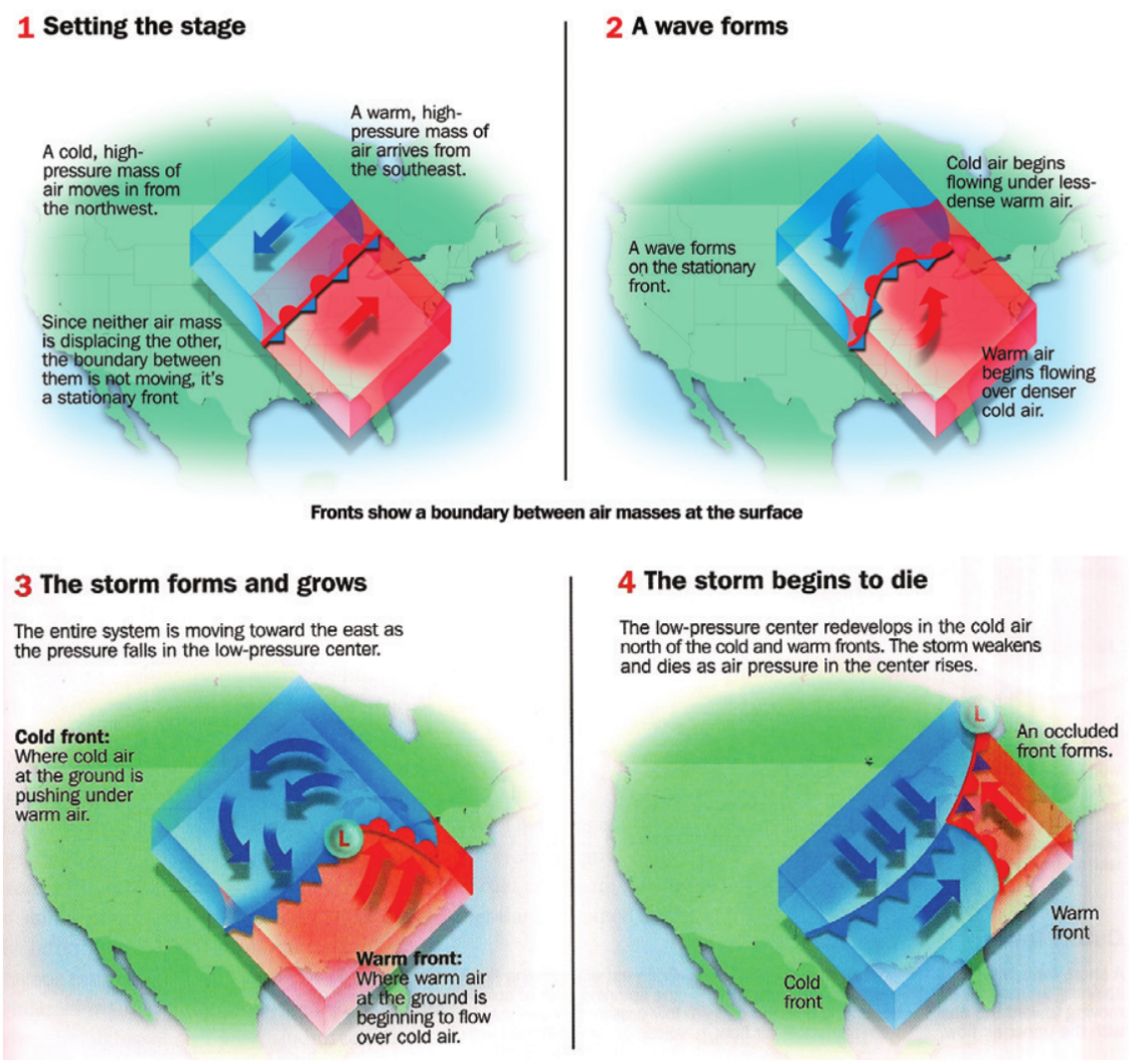


Figure 2.3: Schematic of Norwegian cyclone model (Figure 1 in SV11).

In the Norwegian model, occlusion signifies the end of the deepening stage and the beginning of cyclolysis, whereas in the Shapiro-Keyser model the cyclone may continue to deepen after occlusion.[11] The main cause of this discrepancy is that the Norwegian model treats the atmosphere as two slabs (lower and upper level), whereas the Shapiro-Keyser model treats the atmosphere as a continuous fluid. Thus, in the Norwegian model lifting of occluded warm air destroys low-level baroclinicity, but in the Shapiro-Keyser model low-level baroclinicity may persist for some time. The main deficiency of the Shapiro-Keyser model is that it does not account for cyclolysis. In general, since BS22, cyclolysis has received much less attention than cyclogenesis. However, cyclolysis may occur when the cyclone moves into a region of anticyclonic upper-level vorticity or when abundant cold advection destroys low-level baroclinicity (Morris and Smith 2001).

The most appropriate cyclone model depends on several factors, such as underlying topography and upper-level flow. Schultz et al. (1998) investigated the influence of confluent and diffluent upper-level flow on cyclone structure. Confluent flow features strong zonal wind and weak meridional wind, typical of the entrance region of a jet stream, while diffluent flow features weak zonal wind and strong meridional wind, typical of a jet-exit region. Schultz et al. (1998) found that diffluent flow favours Norwegian-type cyclones and confluent flow favours Shapiro-Keyser-type cyclones. However, as a cyclone moves through different upper-level flow regimes, its structure may change along a continuum between the two. This suggests that the Norwegian and Shapiro-Keyser models are opposite ends of a spectrum, but other models have been proposed (*e.g.*, Hobbs et al. 1996) and may not fall along this spectrum.

Not all fronts are associated with mid-latitude cyclones. Common examples are back-door fronts, associated with anticyclones (Bosart et al. 1973), and stationary fronts, often associated with topographic boundaries such as mountain ranges and coastlines. Both back-door and stationary fronts may feature warm or cold advection, and stationary fronts

[11] SV11 expanded the definition of "occlusion" to include both the Norwegian type and the Shapiro-Keyser warm seclusion; I will do the same.

may temporarily become mobile during periods of cross-boundary flow. The structure and evolution of non-cyclonic fronts have received much less attention, as fronts (Berry et al. 2011b) and associated extreme weather (*e.g.*, Catto and Pfahl 2013; Catto et al. 2015) appear to be most common in the mid-latitude cyclone track.

2.2.2 Climatology

The mid-latitude cyclone track is determined by the positions of the subtropical and polar jet streams (Figure 2.4). The subtropical jet stream is associated with the Hadley cell, which features low-level convergence and ascent near the equator, descent in the subtropics (around 30 °N), equatorward flow at the low levels, and poleward flow at the upper levels.[12] As upper-level air travels poleward, it approaches the Earth's axis of rotation, so conservation of angular momentum dictates that its speed increase. This leads to a speed maximum at the poleward edge of the Hadley cell (around 30 °N), which is the subtropical jet. Conversely, the polar jet is caused mainly by temperature gradients. The polar cell features low-level divergence and descent at the north pole, ascent at its equatorward edge (around 60 °N), and equatorward flow at the low levels. Meanwhile, the Ferrel cell (or "mid-latitude cell" in Figure 2.4), which is a thermally indirect circulation (forced by the thermally direct Hadley and polar cells), features poleward flow at the low levels, which meets equatorward flow from the polar cell around 60 °N. This meeting of warm and cold air (the polar front) leads to a sharp southward-directed thickness gradient. Due to the thermal-wind balance, this leads to westerly wind shear and westerly wind at the upper levels, which is the polar jet.

Thermal wind is defined by

$$(\Delta u_g, \Delta v_g) = \frac{g}{f}(-\frac{\partial H}{\partial y}, \frac{\partial H}{\partial x}), \tag{2.2}$$

[12]For the sake of simplicity, this discussion focuses on the northern hemisphere. However, a nearly symmetric pattern occurs in the southern hemisphere.

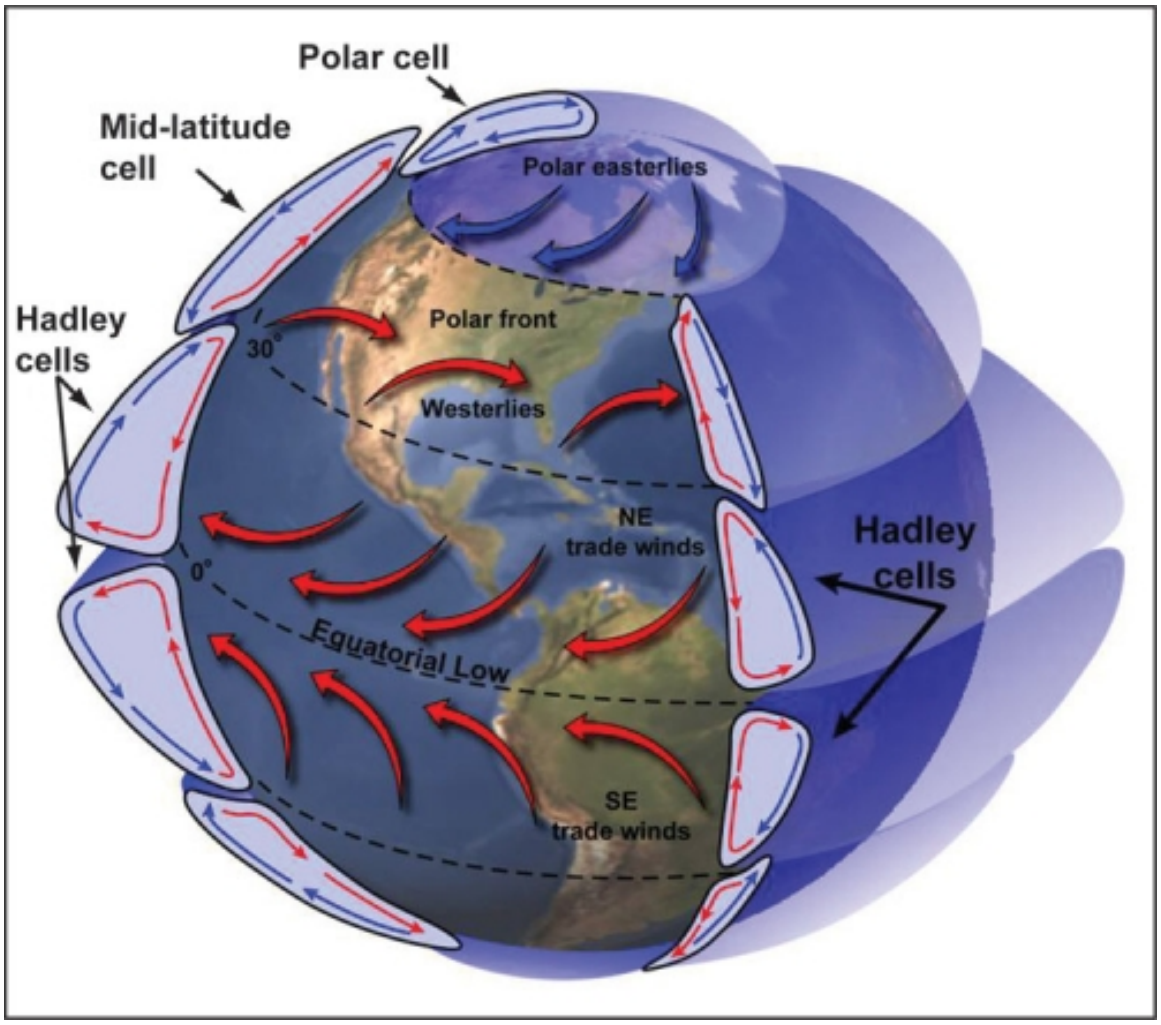


Figure 2.4: Schematic of global circulation (source: Harvard University 2020). Thick arrows embedded in the Earth's surface (blue in the polar easterlies, red elsewhere) show climatological wind near the surface, and thin arrows above the Earth's surface (both red and blue) show climatological vertical circulation cells.

where u_g and v_g are the zonal and meridional components of the geostrophic wind (m s^{-1}); Δu_g and Δv_g are changes in these components over a given layer (top minus bottom); H is atmospheric thickness over the same layer (metres); g is the gravitational constant ($\sim$9.81 m s^{-2}); and f is the Coriolis parameter (s^{-1}). If the thickness gradient points southward, $\frac{\partial H}{\partial y} < 0$ and $\frac{\partial H}{\partial x} = 0$, leading to $\Delta u_g > 0$ and $\Delta v_g = 0$. In other words, the wind shear is entirely westerly.

The simplistic picture in Figure 2.4 suggests that the jet streams are stationary and perfectly zonal. However, due mainly to Rossby waves, they undulate and occasionally parts of them merge. Mid-latitude cyclones are typically associated with the two jets, due to ample low-level baroclinicity and favourable upper-level dynamics. Thus, the area between the two jets comprises the mid-latitude cyclone track. Extratropical cyclones are more common here than anywhere else on Earth, shown in Figure 2 of Eichler and Higgins (2006, henceforth EH06) and Figure 2 of Hodges et al. (2011). Also, the mid-latitude cyclone track shifts poleward during the summer and equatorward during the winter, due to the global annual heating cycle. In the winter, solar radiance is maximized in the southern tropics and decreases towards the north pole, while day length also decreases towards the north pole. These effects are additive and cause a strong meridional temperature gradient, which prevents warm (cold) air from reaching far north (south), thus shifting the maximum baroclinicity southward. In the summer, solar radiance is maximized in the northern tropics and decreases towards the north pole, but day length increases towards the north pole. These effects are opposing and cause a weaker meridional temperature gradient, which allows warm (cold) air to reach farther north (south), thus shifting the maximum baroclinicity northward. This seasonal migration is shown for cyclones in Figure 2 of EH06 and for fronts in Figure 3 of Berry et al. (2011b, henceforth B11b).

In addition to seasonal variability, the mid-latitude cyclone track is affected by interannual climate variability and long-term change (external forcing). One of the leading modes

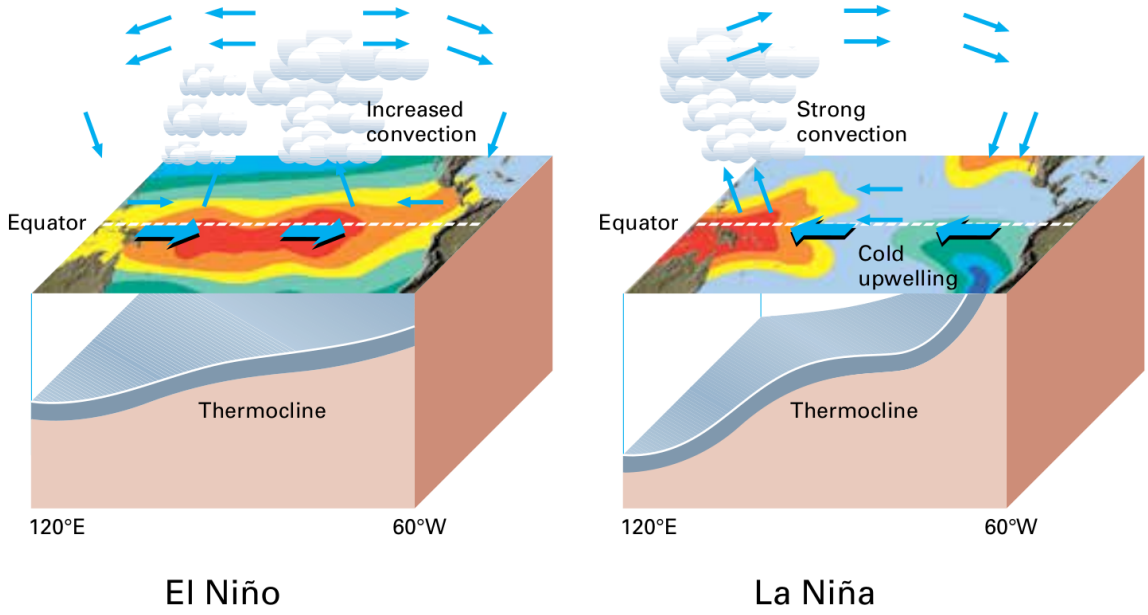


Figure 2.5: Schematics of El Niño and La Niña. Shading on the Earth's surface represents sea-surface temperature (source: World Meteorological Organization 2014), with blue colder and red warmer.

of interannual variability is the El Niño – Southern Oscillation (ENSO; Figure 2.5), characterized by sea-surface temperature (SST) anomalies in the equatorial Pacific (EqP). El Niño occurs when the easterly trade winds weaken, allowing warm surface water in the western EqP to move eastward, down the gradient in sea-surface height that was initially caused by the trade winds "piling up" water in the western EqP. This enhances deep convection in the eastern EqP while suppressing it in the western EqP, which causes the Hadley cell in the eastern EqP to strengthen and contract (Seager et al. 2003; Schmidt and Grise 2017). This leads to a southward shift of the subtropical jet and mid-latitude cyclone track, shown in Figure 13 of Schemm et al. (2018, henceforth S18). After several months to years, the El Niño phase ends and is often followed by La Niña.[13] La Niña features anomalously strong easterly trade winds, leading to strong upwelling and negative SST anomalies in the eastern

[13]Recent work (Lin and Qian 2019) suggests that this transition is caused by an eastward-moving wave of cold water just below the surface, originating in the western EqP.

EqP, which suppresses deep convection. This weakens and expands the Hadley cell (Seager et al. 2003; Schmidt and Grise 2017), which shifts the subtropical jet and mid-latitude cyclone track northward (Figure 13 of S18). The effects of both El Niño and La Niña are shown in the cyclone climatology of EH06 (their Figure 8) and front climatology of Rudeva and Simmonds (2015, henceforth RS15) (their Figure 6b).

In the long term, global warming could have two major effects on mid-latitude cyclones. First, some research suggests that the Hadley cell will expand towards the poles (Davis and Rosenlof 2012; Lucas et al. 2014; Schmidt and Grise 2017), which would lead to a poleward shift in the subtropical jet and cyclone track. Second, Arctic amplification will lead to more warming at high latitudes (Serreze and Barry 2011), reducing low-level baroclinicity at high latitudes. This could reduce the frequency and intensity of cyclones at the poleward margin of the cyclone track. To my knowledge, only two studies have investigated long-term trends in cyclones or fronts over a large (near-hemispheric to global) domain: Berry et al. (2011a, henceforth B11a) investigated fronts only, and RS15 investigated both cyclones and fronts. RS15 (their Figure 1) found a decrease in fronts over the Arctic but with no corresponding decrease in cyclones, as well as a northward shift in winter fronts over the Pacific. B11a (their Figure 2) also found a decrease in fronts over the Arctic, as well as a northward shift over the central and eastern Pacific, but an overall decrease in the Atlantic rather than a northward shift.

The aforementioned climatologies (B11a, B11b, and RS15) used numerical frontal analysis (NFA; Hewson 1998) to detect fronts. NFA is a class of algorithms that apply programmed rules to gridded data. The algorithm typically involves one locating variable, used to identify candidate fronts, and several masking variables, used to eliminate spurious candidates. The two most common algorithms are the thermal method (Hewson 1998) and wind-shift method (Simmonds et al. 2012). In the thermal method, the locating variable is based on derivatives of the thermal field τ, such as the thermal front parameter, defined as $\text{TFP} = -\vec{\nabla}\|\vec{\nabla}\tau\| \cdot \hat{\nabla}\tau$. Ridge lines of TFP (maxima) define the warm edge of the frontal

zone, and trough lines (minima) define the cold edge. Meanwhile, the wind-shift method, developed for the southern hemisphere, classifies a grid cell as frontal if within the last six hours: (a) wind direction has changed from northwesterly to southwesterly; (b) meridional wind v has increased by at least 2 m s^{-1}. The wind-shift method can be modified for the northern hemisphere. B11a and B11b used the thermal method, while RS15 used the wind-shift method.

Despite the benefits of automation, NFA suffers from noisy detection fields and a lack of flexibility. Noisy fields are caused by computing derivatives (especially higher-order derivatives) on a finite grid, while lack of flexibility is caused by the simplicity of the programmed rules. Because NFA is an expert system, these rules must be programmed *a priori* and cannot be learned from the data. To illustrate NFA's lack of flexibility, Schemm et al. (2015) found that the thermal method rarely detects anticyclonic fronts, while the wind-shift method rarely detects warm fronts. These issues have spurred recent efforts to use machine learning, which are reviewed in Chapter 1.

Chapter 3

Data Description and Pre-processing

This chapter describes the input data for tornado prediction and front detection, as well as pre-processing methods applied to the data before ML. Unless otherwise noted, all pre-processing methods are my own work.

3.1 Tornado Prediction

I use three datasets to create predictors: the Multi-year Reanalysis of Remotely Sensed Storms (MYRORSS; Ortega et al. 2012), Gridded NEXRAD WSR-88D Radar (GridRad; Homeyer and Bowman 2017), and the Rapid Refresh[1] physical model (RAP; Benjamin et al. 2016). Characteristics of the three datasets are summarized in Table 3.1. I use MYRORSS and GridRad to create a storm-centered radar image, representing the storm at forecast time, and the RAP to create a proximity sounding, representing the environment in which the storm will evolve over the next hour. I use National Weather Service tornado reports (NWS 2016) to determine when, if at all, the storm is tornadic.

Section 3.1.1 describes these datasets in more detail, and the remaining subsections describe pre-processing methods. Sections 3.1.4 and 3.1.5 describe storm-tracking and tornado attribution, which are needed to create labels, indicating whether or not a storm is tornadic in the next hour. Sections 3.1.2 and 3.1.3 describe echo classification and storm detection, which are needed to create the storm objects used by the tracking algorithm. Appendix A describes the estimation of storm velocity, which is used by both the tracking and attribution algorithms. Finally, Section 3.1.6 describes the creation of predictors, which are storm-centered radar images and proximity soundings.

[1]For initialization times before 1 May 2012, I use the Rapid Update Cycle (RUC; Benjamin et al. 2004), which is the predecessor to the RAP.

Table 3.1: Summary of tornado-predictors. "ASL" = above sea level; "AGL" = above ground level.

Dataset	**Time Period**	**Time Step**	**Horizontal Spacing**	**Vertical Levels**
MYRORSS reflectivity	2000-11	5 min	0.01°	0.25, 0.50, …, 3.00 km ASL and 3.5, 4.0, …, 9.0 km ASL and 10, 11, …, 20 km ASL
MYRORSS shear	2000-11	5 min	0.005°	Low level (0-2 km AGL) Mid level (3-6 km AGL)
GridRad	Selected days in 2011-18	5 min	0.0208°	0.5, 1.0, …, 7.0 km ASL and 8, 9, …, 22 km ASL
RAP	May 2012 – present	1 hour	13 or 20 km	100, 125, …, 1000 mb
RUC	Apr 2002 – Apr 2012	1 hour	13 or 20 km	100, 125, …, 1000 mb

3.1.1 Data Description

MYRORSS contains quality-controlled, merged data from all Weather Surveillance Radar 1988 Doppler (WSR-88D; Crum and Alberty 1993) sites in the continental United States (CONUS). Each radar scans a different part of the atmosphere, and where multiple radars scan the same point, they generally have differing resolution and errors. Merging data from all radars allows the data to be represented on a common grid, and the merging algorithm includes quality-control measures that cannot be applied to single-radar data. The merging algorithm is part of the Warning Decision Support System with Integrated Information (WDSS-II; Lakshmanan et al. 2007), a software package for the visualization, analysis, and forecasting of thunderstorms and their attendant hazards. MYRORSS is similar to the Multi-radar Multi-sensor dataset (MRMS; Smith et al. 2016), which is used by the National Weather Service to make warning decisions. The main difference is that MYRORSS is a reanalysis and MRMS is a real-time product, so MYRORSS has undergone further quality control. Ideally, I would train models with MRMS data, which would allow a more direct evaluation of their performance in a real-time setting. However, the MRMS archive is shorter ($\sim$4 years) and has a lot of missing data.

At each 5-minute time step, MYRORSS contains a 3-D reflectivity grid, plus 2-D grids of low-level and mid-level azimuthal shear. Reflectivity (Z_H) is the sixth moment of the particle-size distribution (PSD), so it is proportional to the sixth power of the diameter of hydrometeors (rain drops, snow flakes, hail stones, etc.). The unit is dBZ, which is the base-10 logarithm of mm^6 m^{-3} (sixth power of hydrometeor diameter per unit volume of air). Azimuthal shear, with units of s^{-1}, is the azimuthal shear of radial velocity (Mahalik et al. 2019). Low-level shear is the maximum from 0-2 km above ground level (AGL), and mid-level shear is the maximum from 3-6 km AGL. Azimuthal shear is computed for each individual radar, then the values are merged – rather than merging radial velocities, then computing azimuthal shear on the merged grid. Merging radial velocities is impossible,

because the radial direction corresponds to a different Earth-relative direction at each grid point.

GridRad also contains merged data from WSR-88D radars. The main differences between MYRORSS and GridRad are different merging algorithms, time periods, and variables (Table 3.1). I have obtained 5-minute GridRad data for 154 convective days[2], summarized in Figure 3.1. A convective day, defined by the Storm Prediction Center, runs from 1200-1200 UTC. For example, convective day "April 27" runs from 1200 UTC 27 Apr – 1200 UTC 28 Apr. These days represent a variety of scenarios, including large and small tornado outbreaks in all seasons. The spatial domain for each day is different and generally not CONUS-wide (Figure 3.1c). However, the domain usually covers a large portion of the CONUS, including most tornadoes on the given day and many non-tornadic storms. Tables B.1-B.4 contain an exhaustive list of the convective days and domains.

At each 5-minute time step, GridRad includes four variables on the 3-D grid: reflectivity, spectrum width, vorticity, and divergence. Spectrum width (σ_v; units of m s^{-1}) is the standard deviation of radial velocities within a grid cell, which generally increases with mean velocity and turbulence. Vorticity (ζ; units of s^{-1}) is the rotational component of the horizontal wind, defined strictly as $\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$, where x and y are the horizontal coordinates; u is the x-component of the wind vector; and v is the y-component. Divergence (δ; units of s^{-1}) is the outward flux of the horizontal wind, defined strictly as $\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$. However, only one wind component, radial velocity, is measured by radar. Thus, in GridRad, vorticity is computed as 2 $\times$ azimuthal shear (the azimuthal derivative of radial velocity), while divergence is 2 $\times$ radial shear. Like azimuthal shear in MYRORSS, vorticity and divergence are computed for each individual radar first, then merged across radars. Figure 3.2 shows an example of MYRORSS of GridRad data over their full domains at one time step.

[2]Hourly data are available publicly via the Research Data Archive (RDA; Bowman and Homeyer 2017), and 5-minute data are available upon request from Cameron Homeyer. 5-minute data will eventually be publicly available on the RDA as well.

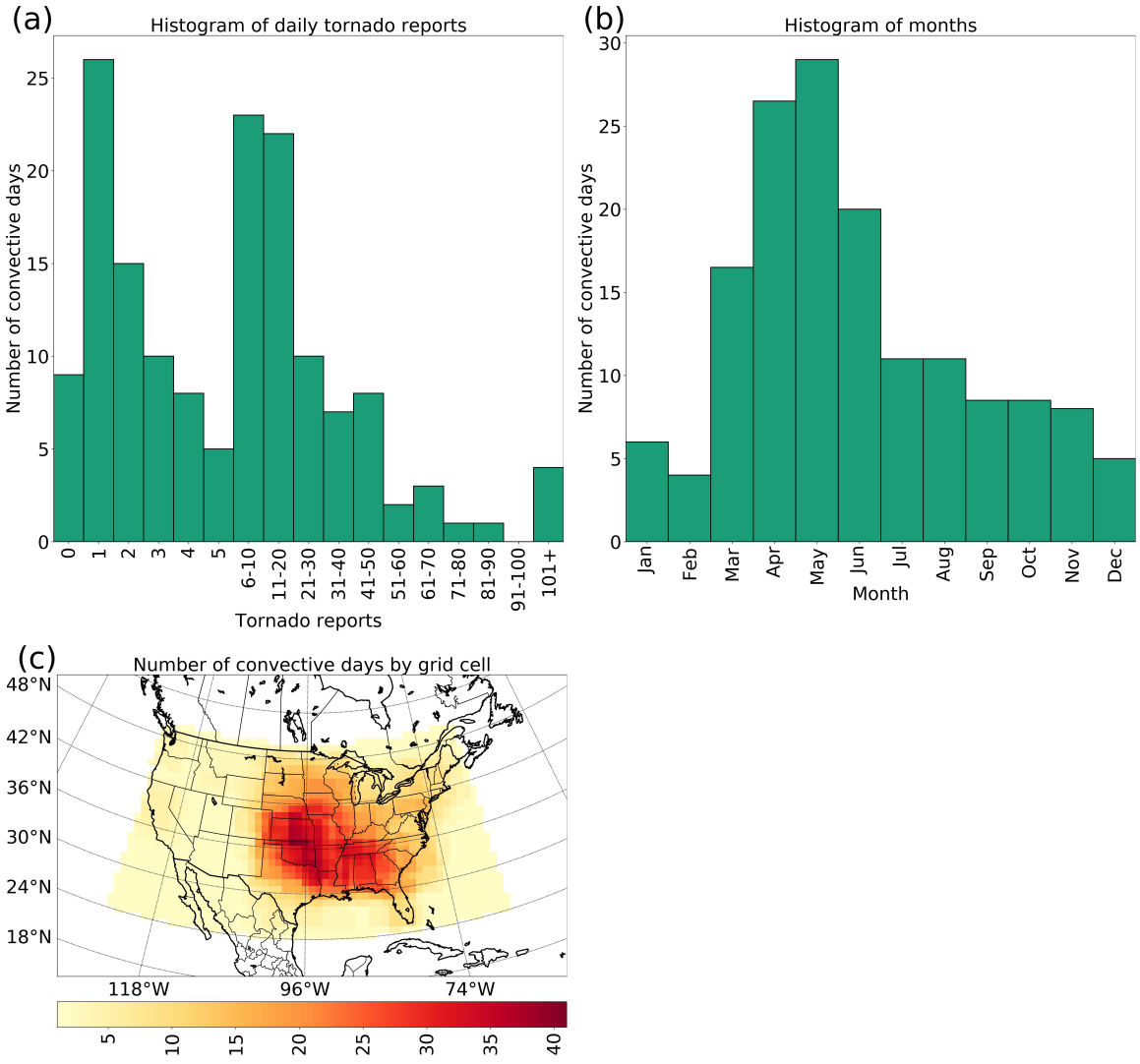


Figure 3.1: Overview of 5-minute GridRad data. [a] Convective days binned by number of tornado reports. [b] Convective days binned by month. [c] Number of convective days for which each grid cell is in the domain.

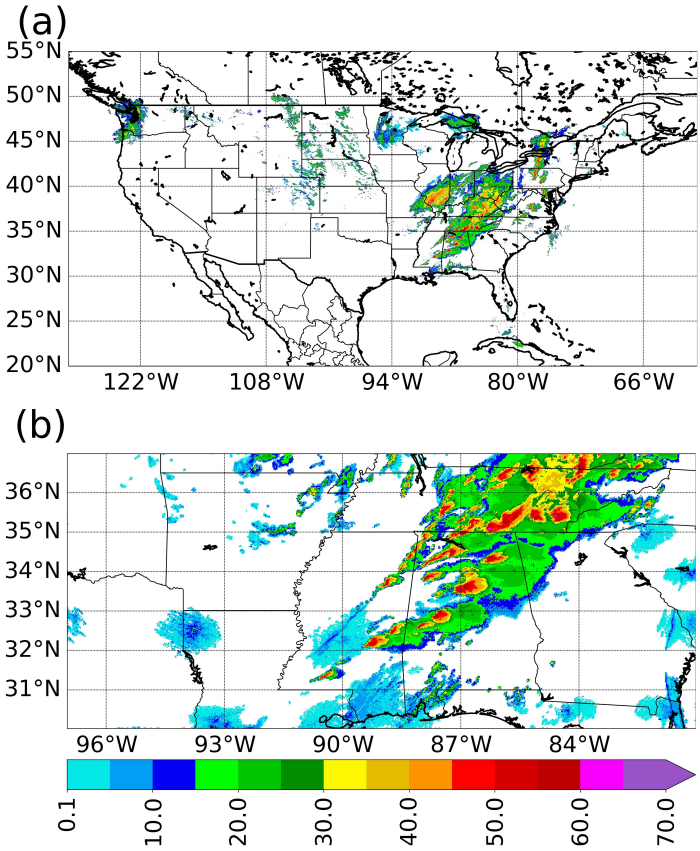


Figure 3.2: Side-by-side comparison of MYRORSS and GridRad data. [a] Column-maximum reflectivity (dBZ) for the full MYRORSS domain at 2300 UTC 27 Apr 2011. [b] Same but for GridRad, showing the full GridRad domain for convective day 27 Apr 2011.

The RAP is a non-hydrostatic mesoscale model with 13- or 20-km grid spacing and covers much of North America, including the full CONUS. The RAP is run every hour and produces forecasts at one-hour time steps, at 37 pressure levels spaced equally from 100 to 1000 mb. I use RAP-simulated instead of observed soundings, because the latter are few and far between (the CONUS has only 92 upper-air sites, which launch soundings only once every 12 hours). I use the RAP instead of another physical model, because the RAP has a long and mostly complete archive[3] and is commonly used in convective meteorology, including as the background field for the Storm Prediction Center mesoanalysis (SPC 2019). The disadvantage of operational models like the RAP is that their physics and dynamics change frequently (*e.g.*, Table 1 of Benjamin et al. 2016); these inhomogeneities can negatively affect the performance of ML. However, my goal in this work is to create a ML system that could be operationalized, so it is important to use an operational model despite this disadvantage.

NWS tornado reports are collected from local residents, and most reports are corroborated by sending a team of NWS employees to conduct a damage survey. Thus, the dataset contains very few false positives. However, the dataset contains many false negatives, especially in sparsely populated areas and at night (Doswell et al. 1999). To alleviate this population bias, some researchers use other datasets as ground truth, such as radar-derived rotation tracks (*e.g.*, Skinner et al. 2018). These tracks can directly resolve mesocyclones but not tornadoes, which is a major disadvantage, because most mesocyclones do not produce tornadoes (Trapp et al. 2005a) and some tornadoes are not associated with mesocyclones. Thus, I use NWS reports despite this advantage. To alleviate population bias, I also evaluate the models without EF-0 and EF-1 tornadoes (Section 6.1). There is little to no population bias for stronger tornadoes, because they generally have longer tracks and cause enough damage that they are reported (Anderson et al. 2007; Elsner et al. 2013).

[3] `https://www.ncei.noaa.gov/thredds/catalog.html`

3.1.2 Echo Classification

The echo-classification algorithm classifies each horizontal grid point as convective or non-convective, so that non-convective points can be masked out during storm detection. I use a modified version of Storm Labeling in 3 Dimensions (SL3D; Starzec et al. 2017). The modified algorithm, shown in Figure 3.3, is applied separately at each time step in each dataset (MYRORSS and GridRad). The inputs are reflectivity fields at heights of 1, 2, …, 12 km above sea level (ASL). i is the latitude index for the grid; j is the longitude index; and k is the height index.

The order of steps 1-3 is irrelevant, as long as they are followed by step 4 and then step 5. Steps 4 and 5 effectively smooth the field of convective and non-convective labels. All thresholds in my algorithm (*e.g.*, 25 dBZ in step 1, 45 dBZ in step 2) are as in the original SL3D. The main difference between my algorithm and the original SL3D is the use of a climatological melting level in step 2. Also, SL3D classifies echoes into five categories, whereas this algorithm considers only two: convective and non-convective.

Peakedness, used in step 1, is defined by

$$p_{i,j,k} = Z_{i,j,k} - \overline{Z}_{i,j,k}, \tag{3.1}$$

where $Z_{i,j,k}$ is the actual reflectivity at point (i, j, k); $\overline{Z}_{i,j,k}$ is the reflectivity after applying a horizontal median filter with 12-km radius; and $p_{i,j,k}$ is the peakedness. All variables have units of dBZ. The peakedness threshold, also used in step 1, is defined by

$$p^*_{i,j,k} = \max\left(4 \text{ dBZ}, 10 \text{ dBZ} - \frac{Z^2_{i,j,k}}{337.5 \text{ dBZ}}\right). \tag{3.2}$$

The melting level in step 2 is estimated by

$$h_{ML} = \beta_0 + \beta_1 \phi, \tag{3.3}$$

where β_0 is the intercept (metres ASL); β_1 is the slope (metres ASL per °N); ϕ is the latitude (°N); and h_{ML} is the melting level (metres ASL). β_0 and β_1 are different for each month (Table 3.2) and were determined by Cameron Homeyer (personal correspondence),

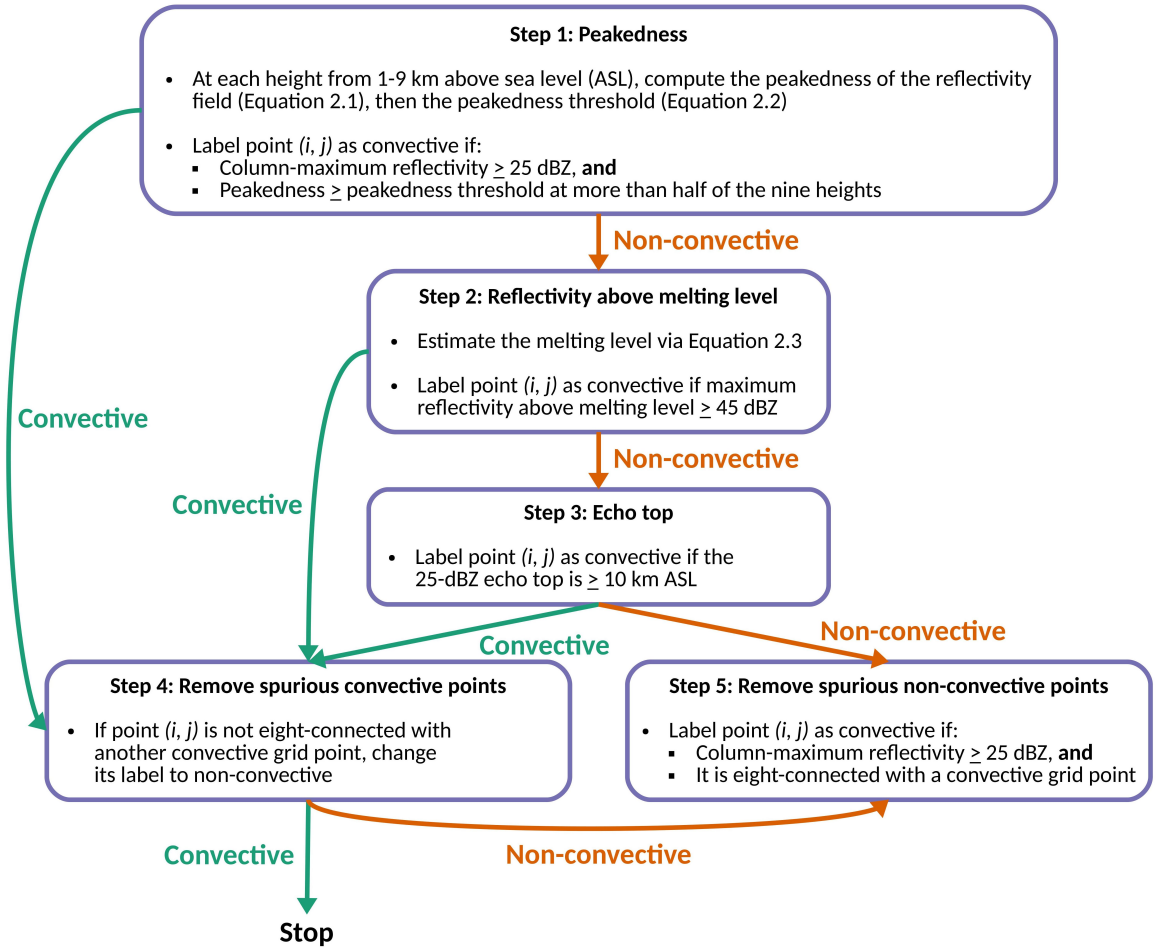


Figure 3.3: Flow chart for echo classification. At the beginning, all horizontal grid points are labeled non-convective by default. Green arrows point to the next step if horizontal grid point (i, j) is currently labeled convective; orange arrows point to the next step if (i, j) is currently labeled non-convective. Two grid cells are "eight-connected" if they share an edge or corner.

based on a 2001-2010 radiosonde climatology over the United States. I use this estimate rather than interpolating from the RAP analysis (which would be more accurate), because doing this for every grid point and every time step would require too much computing time.

Table 3.2: Coefficients for estimation of melting level (Equation 3.3).

Month	**Intercept, β_0 (metres ASL)**	**Slope, β_1 (metres ASL per °N)**
Jan	7072	-124
Feb	7896	-152
Mar	8558	-160
Apr	7988	-128
May	7464	-100
Jun	6728	-65
Jul	6080	-39
Aug	6270	-44
Sep	6786	-67
Oct	8670	-137
Nov	8892	-160
Dec	7936	-147

3.1.3 Storm Detection

The storm-detection algorithm outlines storms in the radar grid at each time step, so that they can be tracked over time. I use a modified version of the algorithm discussed in Section 3c of Homeyer et al. (2017). The input is a horizontal grid of 40-dBZ echo tops (ET_{40}), computed from a 3-D reflectivity grid provided by either MYRORSS or GridRad. I use ET_{40} because the maxima for two storm cells are usually well separated. For commonly

used input fields, such as column-maximum reflectivity, maxima are often indistinct, causing multiple storm cells to be lumped into the same object (*e.g.*, Figure 2 of Jergensen et al. 2020, storm A in Figure 5 of Cintineo et al. 2018). The detection algorithm, described below, is applied separately to each time step in each dataset (MYRORSS and GridRad). Steps 1 and 3-5 are additions to the original algorithm. Thresholds in steps 2 and 6 are from the original algorithm.

1. Mask out non-convective grid points (Figure 3.4a-b), as determined by the echo-classification algorithm in Section 3.1.2. This is done by setting ET_{40} to 0 km ASL.

2. Smooth the ET_{40} field, using a Gaussian filter with *e*-folding radius of 0.024°. Find local maxima in the smoothed field, using a 0.12° × 0.12° window. Keep only local maxima with $ET_{40} \geq 4$ km ASL in the unsmoothed field. All subsequent steps use the unsmoothed ET_{40} field.

3. Convert each local maximum from a point (storm center) to a polygon (storm boundary), as shown in Figure 3.4c. The polygon is the closed contour of $ET_{40} \geq 4$ km ASL that contains the original storm center. If one grid point is inside two or more storm boundaries, the tie is broken by assigning it to the storm with the nearest center. This prevents storm boundaries from overlapping.

4. For each storm, replace the original storm center with the center of the polygon defining the storm boundary (Figure 3.4d).

5. Remove objects with less than 5 grid points inside the polygon (Figure 3.4e).

6. Remove redundant objects. Two objects are considered redundant, and often part of the same storm, if their centers are separated by less than 0.1° latitude or 11.112 km. In this case the object with the smaller ET_{40} maximum is removed. This process is repeated until there is no pair of redundant objects. This step has no effect on Figure 3.4, because after removing objects with < 5 grid points, there are no redundant objects.

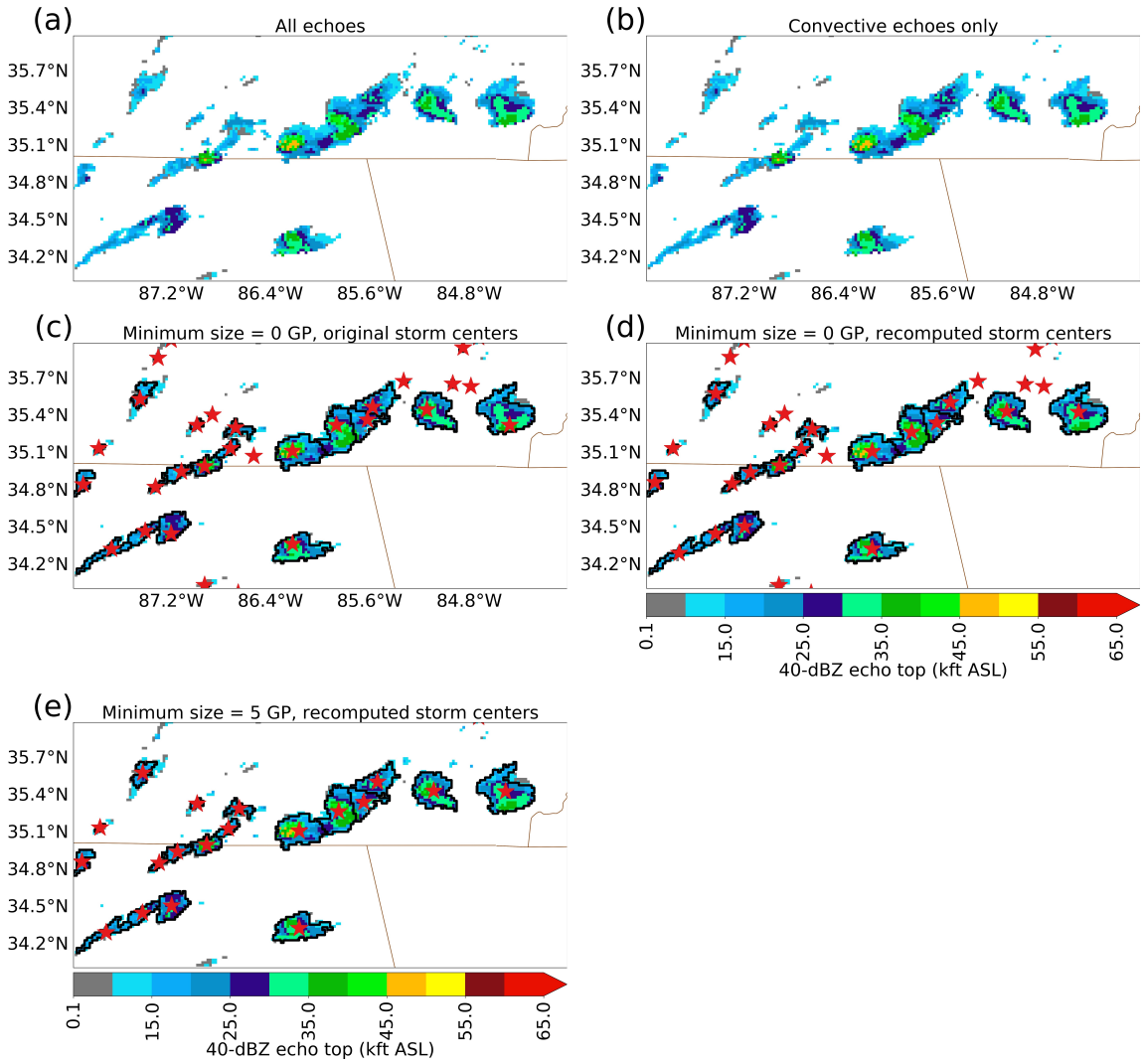


Figure 3.4: Storm detection for part of GridRad domain at 2300 UTC 27 Apr 2011. "GP" = grid points, and "kft ASL" = kilofeet above sea level. [a] 40-dBZ echo top for all echoes. [b] 40-dBZ echo top for convective echoes only. Differences between panels a and b are most evident in the northeast quadrant of the domain. Panels c-e also show convective echoes only. [c] Storm boundaries (black) and centers (red stars) with no minimum size, before recomputing centers. [d] Same but after recomputing storm centers. [e] Same but after removing objects with < 5 grid points.

Step 3 is added to the original algorithm, which contains storm centers but not boundaries, because the boundaries are used to compute distances for tornado attribution (Section 3.1.5). Step 4 facilitates tracking, because the ET_{40} maximum often "jumps" erratically between time steps while the polygon center is more stable. Lastly, step 5 generally removes non-thunderstorms (stratiform and non-meteorological echoes that are misdiagnosed by echo classification), but sometimes it removes weak thunderstorms. This trade-off is acceptable, because (a) the number of non-storms is much greater; (b) the weak storms removed are often at the beginning or end of their life cycle but are still detected in their mature stage, when they are larger than 5 grid points. Note that the 5-grid-point threshold corresponds to different physical areas in MYRORSS (0.01° spacing) and GridRad (0.0208° spacing). I tried defining the threshold by physical area, but the resulting number of grid points was too large for MYRORSS, causing too many MYRORSS objects to be removed, or too small for GridRad, causing too few GridRad objects to be removed. I have found that the 5-grid-point threshold is a good discriminator between storms and non-storms for both datasets.

3.1.4 Storm-tracking

Storms are tracked over time via two algorithms, preliminary and final tracking. The idea of two-stage tracking is motivated by WDSS-II, which uses segmotion (Lakshmanan and Smith 2010) followed by besttrack (Lakshmanan et al. 2015a). I will call my algorithms prelim_tracking and final_tracking. The prelim_tracking algorithm (Figure 3.5) is based on Section 3c of Homeyer et al. (2017), the main difference being that my algorithm allows storms to split and merge. The algorithm is applied separately to each pair of consecutive time steps, t_1 and t_2. Step 2 (linkage without extrapolation) serves as a "second chance" to link storms, which is useful because step 1 uses first-order backward differences to estimate storm velocity, which are often noisy (Appendix A). Steps 1 and 2 both ensure that no more than two storms at t_1 can be linked to the same storm at t_2. Otherwise, more than

two storms could merge into one over five minutes – which rarely, if ever, occurs in the real atmosphere. Step 3 (pruning) ensures that no more than two storms at t_2 can be linked to the same storm at t_1. Otherwise, one storm could split into more than two storms over five minutes, which is also implausible. The 9-km distance threshold in steps 1 and 2 was chosen subjectively. Smaller thresholds often split one storm into several tracks, and larger thresholds often connected several storms into the same track.

The main shortcoming of the prelim_tracking algorithm is that it often "drops" a storm track for one time step, thus splitting one actual track into two tracks with a 10-minute gap. One common reason is that the storm drops below the detection threshold ($ET_{40} \geq 4$ km ASL) for one time step, leaving no storm object to track. Another is that, even though step 4 of the detection algorithm partly alleviates this, the storm center "jumps" erratically between time steps, causing the distance between successive centers to exceed the 9-km threshold.

The result is often two nearly collinear storm tracks with a small gap in the middle, like a piece of string that has been cut in half. The final_tracking algorithm joins such pairs of tracks. The final_tracking algorithm is the same as prelim_tracking (Figure 3.5), with four exceptions. First, the difference between t_1 and t_2 is two time steps (ten minutes). Second, final_tracking is applied only to pairs of preliminary tracks in which one ends at t_1 and the other begins at t_2. Third, velocity estimates used in final_tracking are third-order, rather than first-order, backward differences. Computing higher-order differences requires more computing time per track, but final_tracking considers only a small number of preliminary tracks, which makes this computation feasible. Fourth, final_tracking skips step 2 (linkage without extrapolation), because third-order velocity estimates are much less noisy than

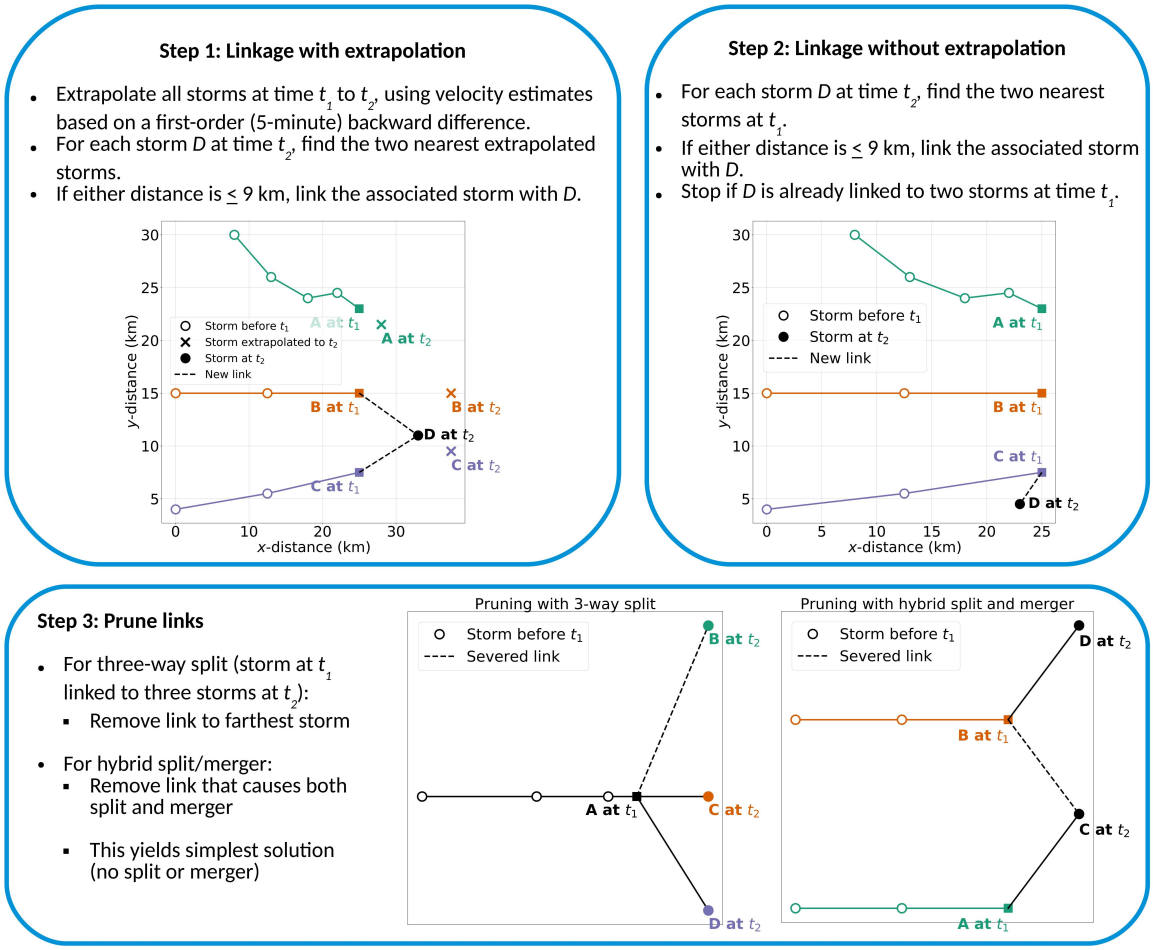


Figure 3.5: Flow chart for prelim_tracking algorithm. In step 1, the extrapolated locations of storms B and C are within 9 km of storm D, so B and C are linked to D. Thus, storms B and C merge into D between times t_1 and t_2. In step 2, storm C at time t_1 is within 9 km of storm D at t_2, so the two are linked. In the three-way split shown for step 3, storm A is initially linked to three storms at the next time step; only the links to the two nearest storms are kept. In the hybrid split and merger shown for step 3, storm C at time t_2 is involved in both a split (storm B at t_1 into storms C and D at t_2) and a merger (storms A and B at t_1 into storm C at t_2). The linkage between storms B and C is severed, leading to the simplest solution (no splits or mergers).

first-order estimates, which obviates the need for step 2 as a "second chance." Figure 3.6 shows sample output from the final_tracking algorithm.[4,5]

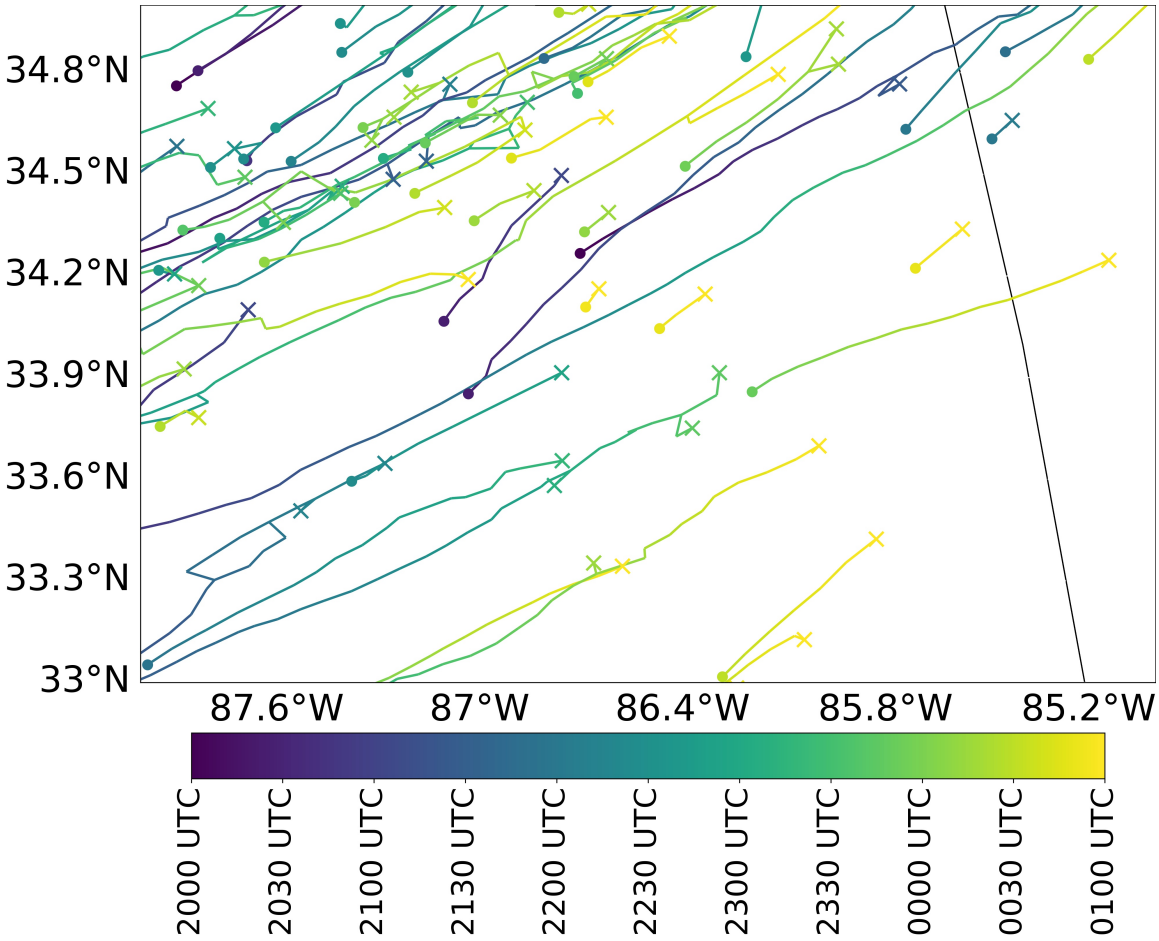


Figure 3.6: Spaghetti plot of storm tracks for part of GridRad domain over a 5-hour period. Circles are start points, and 'X's are end points.

[4]The following link shows animated output for the GridRad data on 26-27 Apr 2011: `https://github.com/thunderhoser/GewitterGefahr/blob/dissertation_gifs/gewittergefahr/dissertation/supfig01_gridrad_tracking.gif`

[5]The following link shows animated output for the MYRORSS data on 26-27 Apr 2011: `https://github.com/thunderhoser/GewitterGefahr/blob/dissertation_gifs/gewittergefahr/dissertation/supfig02_myrorss_tracking.gif`

3.1.5 Tornado Attribution

The procedure shown in Figure 3.7 is repeated for each tornado. It is more complex than most spatial linkage procedures, because the storm tracks include splits and mergers. Before step 1, the tornado is interpolated linearly to 1-minute intervals between its start and end locations (the only locations included in NWS tornado reports), yielding the "tornado points" mentioned in Figure 3.7. In step 1, each storm is interpolated to the time of tornado point p, so that tornado-to-storm distances can be calculated. This interpolation uses the previous and next locations of each storm – by definition, at times t_1 and t_2 respectively. The three cases for storm interpolation are explained below for clarity.

1. If the storm undergoes a merger (shown in Figure 3.7) between t_1 and t_2, interpolate from the midpoint of the two storm centers at t_1 to the actual storm center at t_2. Keep the storm boundary from t_2.

2. If the storm undergoes a split (shown in Figure 3.7) between t_1 and t_2, interpolate from the actual storm center at t_1 to the midpoint of the two storm centers at t_2. Keep the storm boundary from t_1.

3. If the storm neither merges nor splits (not shown in Figure 3.7), interpolate between the two actual storm centers. Keep the storm boundary from the time nearest the tornado point.

When a tornado is initially linked to a storm (step 2), it is linked to only one object. However, to label storm object s for machine learning, we need to know if s is responsible for a tornado within the next hour. This requires linking the tornado to other objects in the same track (successors and predecessors), which is done by steps 3-5. The definition of "simple predecessor," used in step 5, is explained below for clarity.

1. If storm C splits into storms D and E, then either D or E produces a tornado, the tornado is also attributed to C. See example 2 in Figure 3.7.

2. If storm A splits into B and C, then C splits into D and E, then D or E produces a tornado, the tornado is **not** attributed to A. I assume that storm characteristics change enough over two splits that storm A only negligibly impacts the tornado potential of storm D or E. See example 2 in Figure 3.7.

3. If storms X and Y merge into Z, then Z produces a tornado, the tornado is attributed to **neither** X nor Y. There is no obvious way to determine which of X and Y is primarily responsible for the characteristics of Z that led to tornadogenesis. See example 1 in Figure 3.7.

After the procedure shown in Figure 3.7, any storm object linked to a tornado in the next hour is labeled "yes"; all others are labeled "no." Among the storm objects labeled "no," two types are removed from the dataset:

- Storms that merge into a tornadic successor (like X and Y in rule 3 above), because it is unclear if either storm is responsible for tornadogenesis.

- Storms with a successor in the next hour that passes within 10 km (the maximum distance for tornado attribution) of the land boundary of the CONUS (Figure 3.8). NWS storm reports are not collected outside this area, so there is no way to determine if a storm over the ocean or Canada is tornadic.

3.1.6 Creation of Predictors

Each storm object is one case (or "example") for machine learning. The predictors are a storm-centered radar image and proximity sounding. The horizontal center of the storm is the horizontal center of the radar image. For GridRad, the image is 32 rows $\times$ 32 columns $\times$ 12 heights and contains all four variables (Figure 3.9a-d). The grid is equidistant with 1.5-km horizontal spacing, 1.0-km vertical spacing, and rotated so that storm motion points to the right (in the $+x$-direction). Storm motion is estimated by a third-order (15-minute)

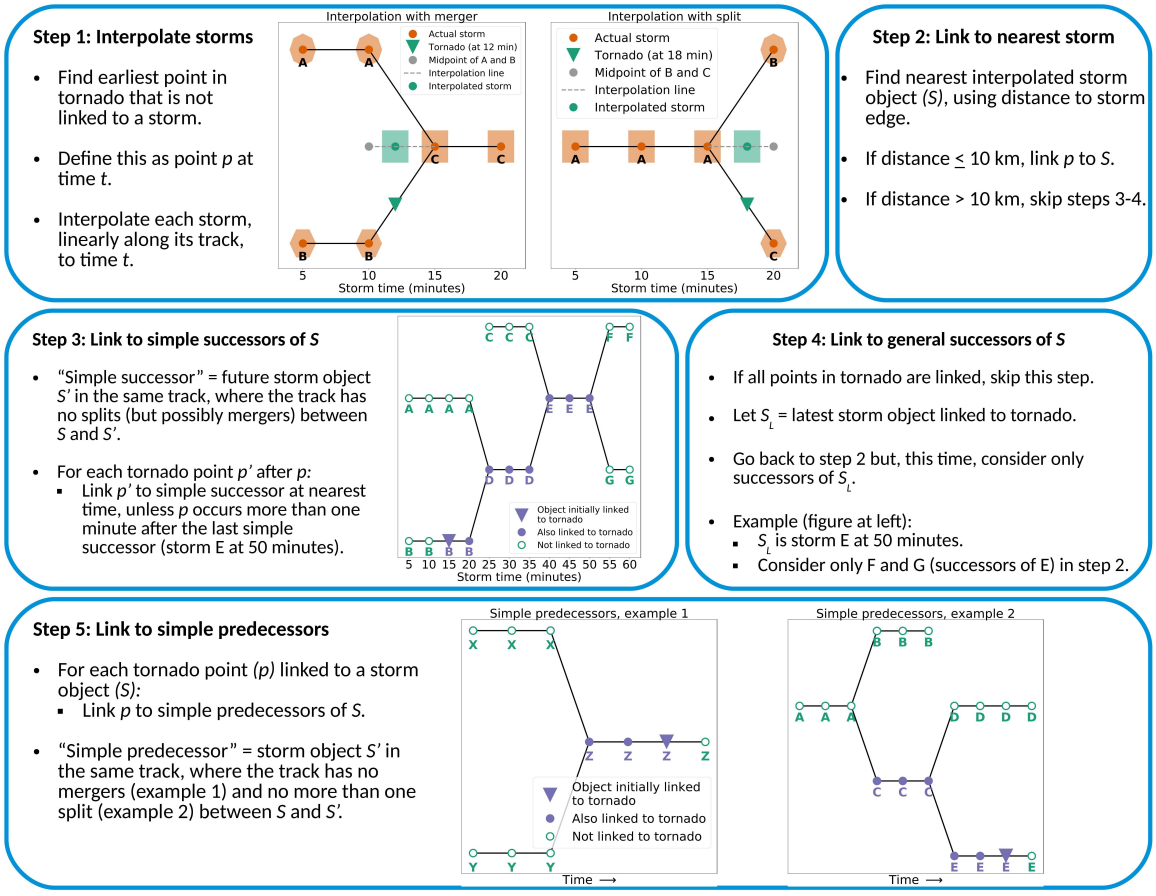


Figure 3.7: Flow chart for tornado attribution. In step 1, each storm is interpolated to the time of the earliest unlinked point in the tornado (p). The three cases for interpolation are explained in the main text. In step 2, p is linked to the nearest interpolated storm object (S), as long as S is close enough. In step 3, p is also linked to simple successors of S. In step 4, if any point in the tornado is still unlinked, the algorithm returns to step 2 but considers only successors of the latest storm object linked to the tornado. In step 5, any tornado point already linked to a storm object is linked to simple predecessors of the given object.

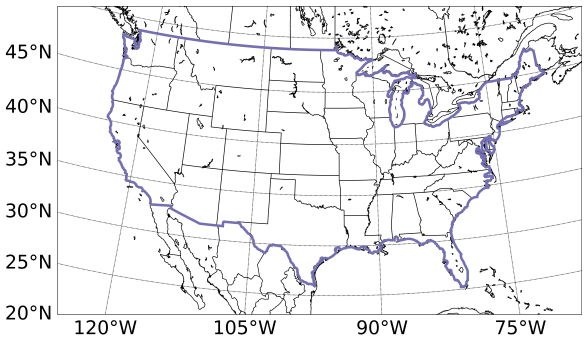


Figure 3.8: Land boundary of the CONUS, eroded by 10 km (purple line). Any storm object with a successor passing outside the purple line, is removed from the dataset.

backward difference. Bicubic interpolation is used to convert data from the native latitude-longitude grid to the new equidistant grid. The 1.5-km spacing is chosen so that resolution is not lost during interpolation. The native grid has 0.0208° spacing, which is 2.32-km meridional spacing everywhere, but zonal spacing decreases towards the north pole. The northern extent of the dataset is 49 °N, and the zonal grid spacing here is 1.52 km, so 1.5-km spacing preserves all relevant details. Also, while heights in the native grid are sea-level-relative, heights in the storm-centered grid are ground-relative (specifically, relative to surface elevation at the horizontal storm center). Otherwise, for storms at high elevation, the lowest heights would be underground and therefore have missing radar data, which would degrade performance of the CNNs. Data are interpolated linearly to the new heights.

For MYRORSS, the storm-centered radar image uses two different grids. The reflectivity grid (Figure 3.9f) is $64 \times 64 \times 12$ with 0.75-km horizontal spacing and 1.0-km vertical spacing, while the azimuthal-shear grid (Figure 3.9g-h) is 128×128 with 0.375-km spacing. As for GridRad, the new horizontal spacings are chosen so that resolution is not lost during interpolation from the native grid. The reflectivity and shear spacings in the native grid are 0.01° and 0.005°, which correspond to 0.73- and 0.36-km zonal spacings at 49 °N. All storm-centered radar images – GridRad, MYRORSS reflectivity, and MYRORSS azimuthal shear – have a horizontal extent of 48×48 km.

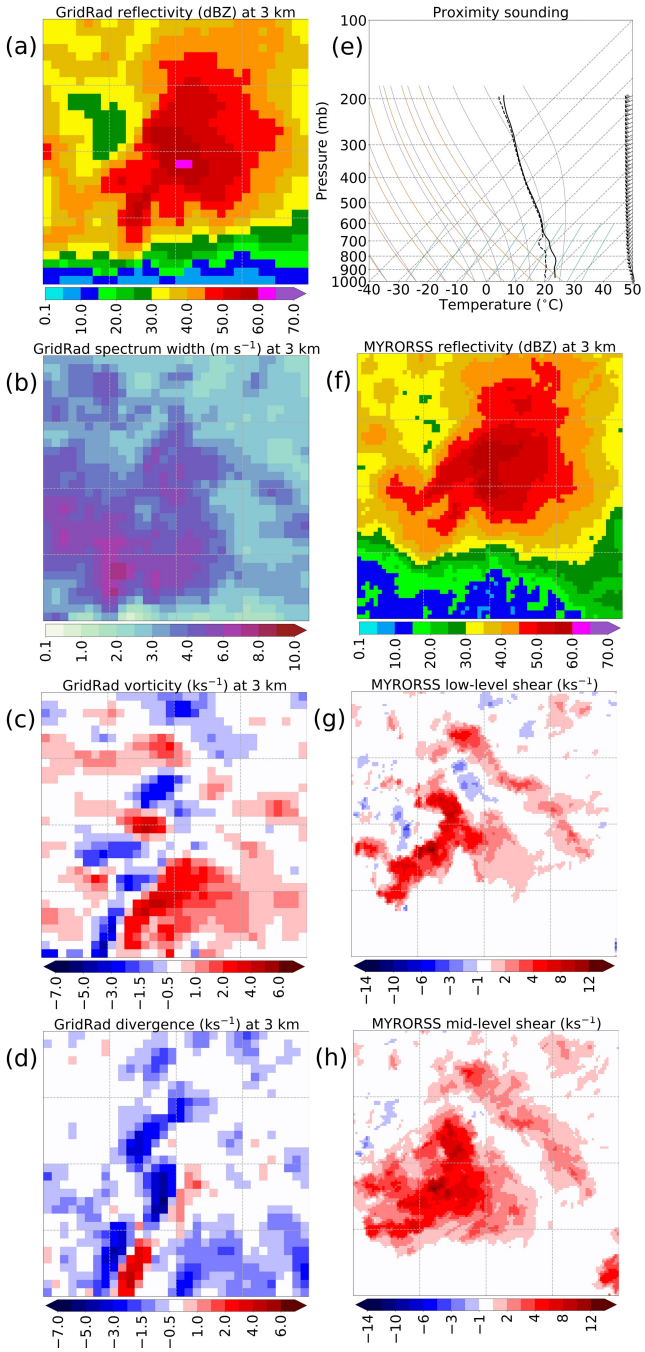


Figure 3.9: Predictors for matched MYRORSS and GridRad examples at 2340 UTC 27 Apr 2011. [a-d] Radar predictors at 3 km AGL for GridRad example. Storm motion points to the right. [e] Proximity sounding, which is the same for both examples, because the RAP grid point nearest the storm center is the same for both examples. [f-h] Radar predictors for MYRORSS example. Storm motion points to the right.

To create proximity soundings (Figure 3.9e), I use the following procedure for each example. Let the example be storm S at forecast time t. Note that proximity soundings are created separately for MYRORSS and GridRad examples.

1. Extrapolate S 30 minutes ahead (the median of the 0–1-hour prediction window) along its current motion vector. Again, storm motion is estimated by a third-order backward difference.

2. Take the sounding from the latest RAP analysis before t and the nearest grid point to the horizontal center of the extrapolated storm. Thus, the "interpolation" method used is previous-neighbour in time and nearest-neighbour in space. Using a higher-order method, such as linear or cubic, sometimes generates physical inconsistencies such as large supersaturations, because each variable in the sounding is interpolated separately. This method preserves the entire sounding from one grid point at one time step, which prevents such inconsistencies. Also, by using RAP data from at least 30 minutes before the extrapolated storm, convective contamination (where the sounding for storm S is influenced by storm S) is usually prevented.

Native RAP variables needed to form a sounding are u-wind, v-wind, temperature, and relative humidity. However, to train CNNs, I replace temperature with virtual potential temperature (θ_v) and specific humidity. *A priori*, these thermodynamic predictors should be useful, because the vertical profile of θ_v (Kelvins) alone determines static stability, while specific humidity (g kg^{-1}) is the total mass concentration of water vapour, which is important along with relative humidity. I interpolate all five variables – u-wind, v-wind, relative humidity, specific humidity, and θ_v – linearly to heights of 0, 0.25, …, 12 km AGL. I use ground-relative heights here for the same reason as in storm-centered radar images – to prevent the use of underground height levels.

Both MYRORSS and GridRad examples are split into training, validation, and testing data (Table 3.3). There is a one-week gap between each pair of consecutive datasets, which

ensures that the storms and synoptic patterns in one dataset are not temporally autocorrelated with those in another. The role of training data is to fit weights in the CNNs, called parameters; the role of validation data is to fit hyperparameters, defined as ML settings that must be chosen before training and cannot be fit during training; and the role of testing data is to evaluate the final model on data independent of the training and validation.

Table 3.3: Training, validation, and testing periods for tornado prediction.

Dataset	**Time Period**
MYRORSS training	Jan 1 2005 – Dec 24 2008
MYRORSS validation	Jan 1 2009 – Dec 24 2010
MYRORSS testing	2011
GridRad testing	Jan 1 2011 – Dec 24 2011
GridRad training	Jan 1 2012 – Dec 24 2014
GridRad validation	2015-2018

Finally, predictors are normalized by

$$z = \frac{x - \bar{x}}{s}, \tag{3.4}$$

where x is the original value; $\bar{x}$ is the mean over training data; s is the standard deviation over training data; and z is the normalized value. The equation is applied independently to each variable in each dataset: reflectivity, low-level azimuthal shear, mid-level azimuthal shear, and the five sounding variables in MYRORSS; as well as the four radar variables and five sounding variables in GridRad. It is crucial that $\bar{x}$ and s be computed with training data only. If validation/testing data were used, the normalized training data would contain information from the validation/testing data and the three sets would no longer be independent. Normalization ensures that all predictors have equal variance (1.0) in the training data, which prevents the CNN from unduly focusing on predictors with higher variance,

which is often due to physical units. For example, in the GridRad training data, reflectivity has a variance of 212 dBZ^2 while vorticity has a variance of 3.0×10^{-7} s^{-2}.

3.2 Front Detection

Predictors come from the ERA5 reanalysis (Hersbach 2016; Hersbach and Dee 2016), and labeled fronts come from Weather Prediction Center (WPC) surface bulletins (NWS 2007). For this project I use a reanalysis instead of an operational model, because the goal is to create a climatology, for which temporal homogeneity is important. To this end, the ERA5 uses Climate Model Intercomparison Project 5 (CMIP5; Taylor et al. 2012) analyses of major climate forcings and sources of internal variability, such as sea-surface temperature, sea ice, greenhouse gases, and volcanic emissions (Hersbach and Dee 2016). Also, recent evaluations of the ERA5 (Gossart et al. 2019; Tarek et al. 2019) suggest that it is appropriate for climatological studies. The ERA5 produces output at one-hour time steps, going back to 1979, on a global 0.281° grid. I use thermal, wind, and pressure fields at several vertical levels to train CNNs (Table 3.4).

Table 3.4: List of ERA5 variables used. "theta" is potential temperature.

Variable Name	Abbreviation	Vertical Levels
Zonal wind	u	Surface, 1000 mb, 950 mb, 900 mb, 850 mb
Meridional wind	v	Same
Temperature	T	Same
Specific humidity	q_v	Same
Wet-bulb theta	θ_w	Same
Pressure	p	Surface
Geopotential height	Z	1000 mb, 950 mb, 900 mb, 850 mb

WPC bulletins are drawn every three hours by meteorologists and include polylines defining warm and cold fronts (Figure 3.10). The bulletins also contain occluded and stationary fronts, which are not used in this work. Most of these polylines – 93% of warm fronts and 97% of cold fronts – have an arc length of at least 200 km, the minimum synoptic length scale (Orlanski 1975). Also, shorter fronts are often very close to another front (*e.g.*, the cold front in the southeastern CONUS in Figure 3.10a-d), with the summed length exceeding 200 km. Thus, WPC fronts can be considered synoptic-scale, fitting the goal of the climatology. However, because the fronts are drawn manually, they suffer from inconsistencies, the most obvious of which are dramatic changes between two time steps. For example, the cold front through Wyoming in Figure 3.10c-f changes orientation by almost 90° in 3 hours, with little to no apparent change in the underlying thermal and wind fields.

One could argue for using NFA labels instead of human labels, because they are consistent and based directly on *a priori* scientific reasoning. However, humans excel at pattern recognition (Mattson 2014), and since the WPC labels come from trained meteorologists, they are ultimately based on scientific reasoning as well. Also, NFA often has strong systematic biases (see discussion of Schemm et al. 2015 at the end of Section 2.2), which could be easily overfit by a CNN. The WPC labels do not appear to contain such biases, probably because each bulletin is a consensus among many meteorologists and the set of meteorologists for each bulletin is different. Thus, I believe that training a CNN with WPC labels is the best possible approach, because it leverages the pattern-recognition ability and scientific knowledge of humans.

Pre-processing methods, described below, are applied separately to each time step.

1. Interpolate ERA5 data from the native latitude-longitude grid to a 32-km equidistant grid (Figure 3.11a-b).

2. Rotate wind fields from Earth-relative to grid-relative coordinates.

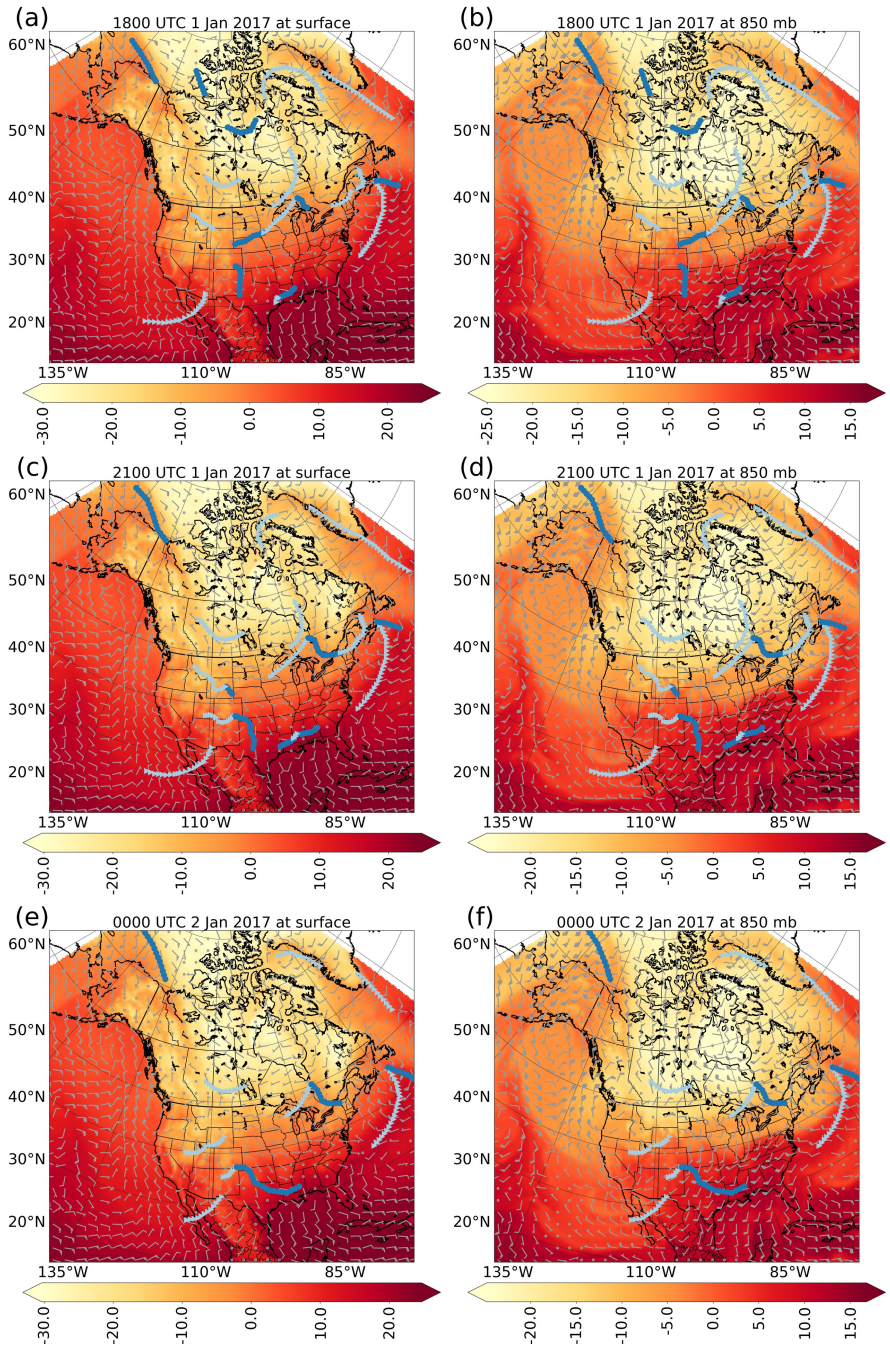


Figure 3.10: Short time series of WPC fronts. Heat maps are θ_w, and grey lines are wind barbs, from the ERA5 reanalysis. Dark blue circles are warm fronts, and light blue triangles are cold fronts, from the WPC bulletins. [a-b] Surface and 850-mb fields, with fronts overlain, at 1800 UTC 1 Jan 2017. [c-d] Same but at 2100 UTC. [e-f] Same but at 0000 UTC 2 Jan.

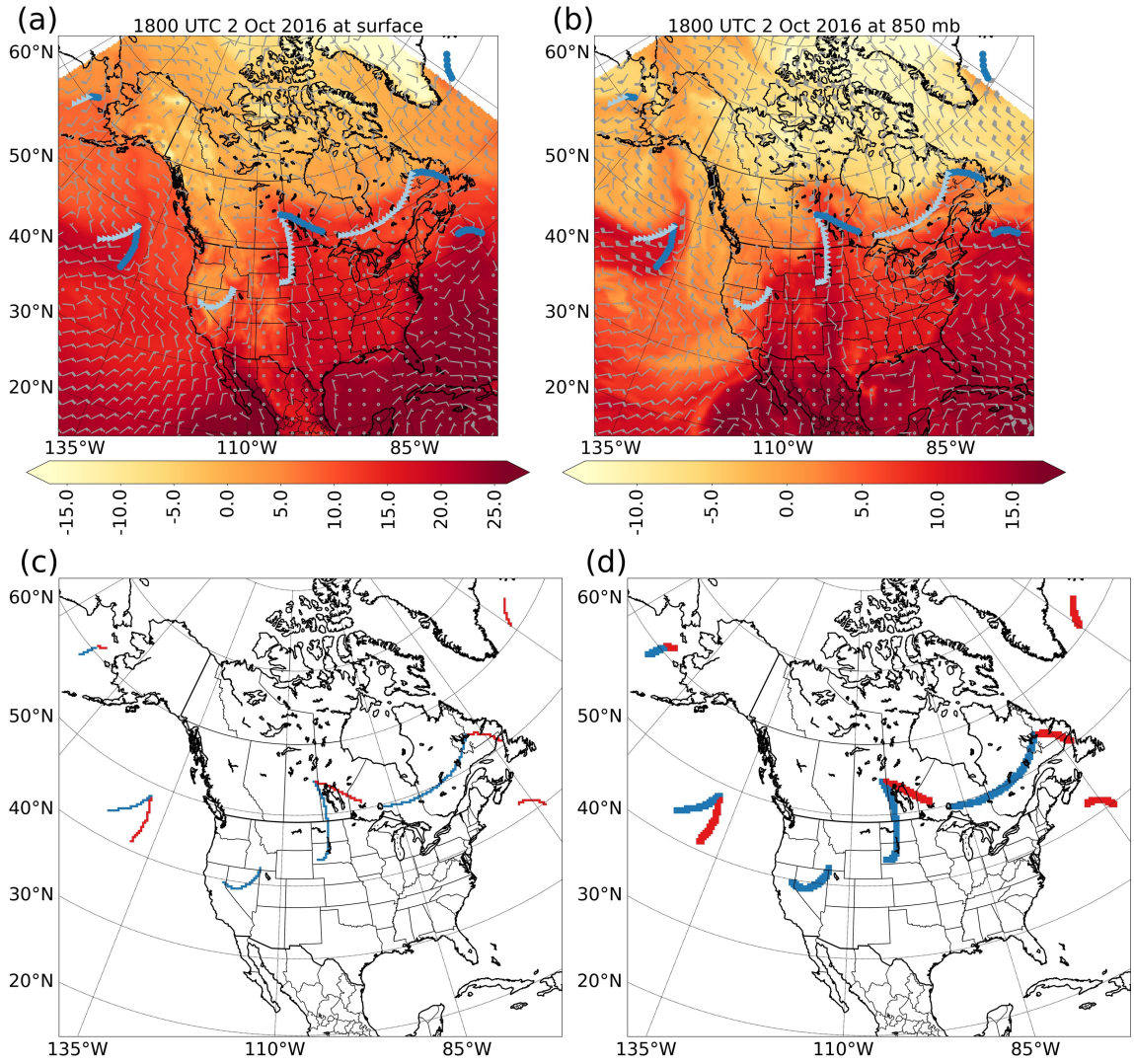


Figure 3.11: Overview of input data for front detection. Formatting of panels a-b is explained in Figure 3.10. [a] ERA5 predictors at the surface. [b] ERA5 predictors at 850 mb. [c] Gridded WPC fronts (blue for cold, red for warm) before dilation. [d] Gridded WPC fronts after dilation.

3. Convert WPC fronts from polylines to gridded fields, on the same 32-km grid used for predictors (Figure 3.11c). Each grid cell is labeled WF if intersected by a warm-frontal polyline, CF if intersected by a cold-frontal polyline, and NF otherwise.

4. Dilate WPC fronts (Figure 3.11d). Specifically, expand each WF or CF grid cell to a 3×3 region. This effectively turns WPC fronts from 1-D lines into 2-D regions, which is closer to physical reality. Ties (where dilation causes a grid cell to be labeled both WF and CF) are handled by keeping the CF label, the more common one in the dataset. The effect of dilation on class frequencies is shown in Table 3.5.

5. Mask grid cells where the WPC does not label fronts (Figure 3.12).

6. Create patches (Figure 3.13). Each patch contains the predictors on a 33×33 grid and the post-dilation label (NF, WF, or CF) at the center grid cell. At each time step there is one patch centered on each unmasked grid cell.

7. Split patches into training, validation, and testing datasets (Table 3.6). As for tornado prediction, there is a one-week gap between each pair of consecutive datasets, to minimize autocorrelation.

8. Normalize predictors. Equation 3.4 is applied separately to each variable at each vertical level (Table 3.4). As for tornado prediction, normalization parameters ($\bar{x}$ and s) are based on the training data only.

Step 1 is required because CNNs assume equidistant grids. In a latitude-longitude grid, this would cause an overestimation of thermal gradients near the equator (where zonal grid spacing is large) and underestimation of thermal gradients near the pole (where zonal grid spacing is small), leading to overdetection of fronts near the equator and underdetection near the pole. Step 2 ensures that wind vectors and gradients are in the same coordinates, which makes their relative orientation easier for the CNNs to represent. This relative orientation is important, because fronts generally have large cross-front advection of thermal

quantities. In reference to step 3, NF grid cells include those intersected by an occluded or stationary front. Thus, it would be more semantically accurate to label these grid cells as "other" rather than "no front." Grid cells masked in step 5 cannot be used as the center of a training, validation, or testing patch – *i.e.*, cannot be used to train or evaluate a CNN – because the true label at these locations is unknown. Lastly, in reference to step 6, previous experiments have shown that 33 $\times$ 33 (1056 $\times$ 1056 km) is a near-optimal patch size, a trade-off between too little and too much surrounding information.

Table 3.5: Class frequencies for front detection. Each value is the average number of unmasked grid cells (Figure 3.12c) intersected by a front of the given type over the training, validation, and testing periods.

Class	**Before Dilation**	**After Dilation**
No front	99.21%	97.57%
Warm front	0.20%	0.62%
Cold front	0.59%	1.82%

Table 3.6: Training, validation, and testing periods for front detection.

Dataset	**Time Period**
Training	1500 UTC 5 Nov 2008 – 2100 UTC 24 Dec 2014
Validation	0000 UTC 1 Jan 2015 – 2100 UTC 24 Dec 2016
Testing	0000 UTC 1 Jan 2017 – 2100 UTC 31 Dec 2017

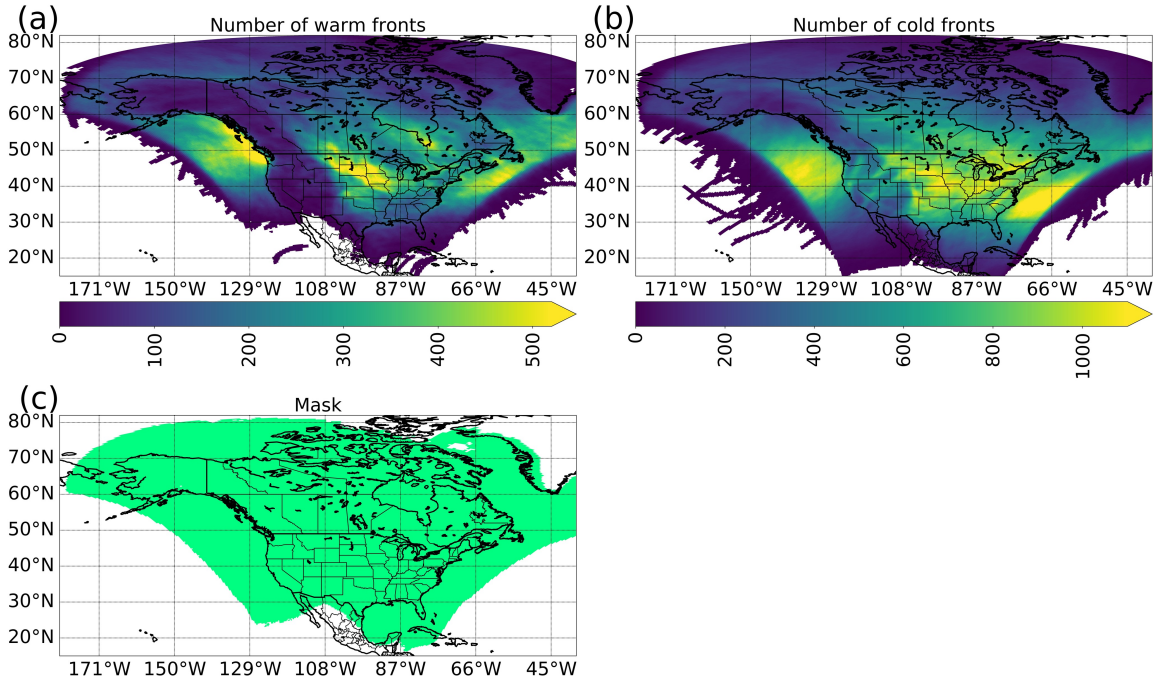


Figure 3.12: Masking of grid cells for CNN development. [a] Number of warm fronts in WPC dataset after dilation. [b] Number of cold fronts after dilation. [c] The resulting mask, based on which grid cells have at least 100 total fronts (sum of panels a and b). Grid cells outside the green area are masked and cannot be used as the center of a training, validation, or testing patch.

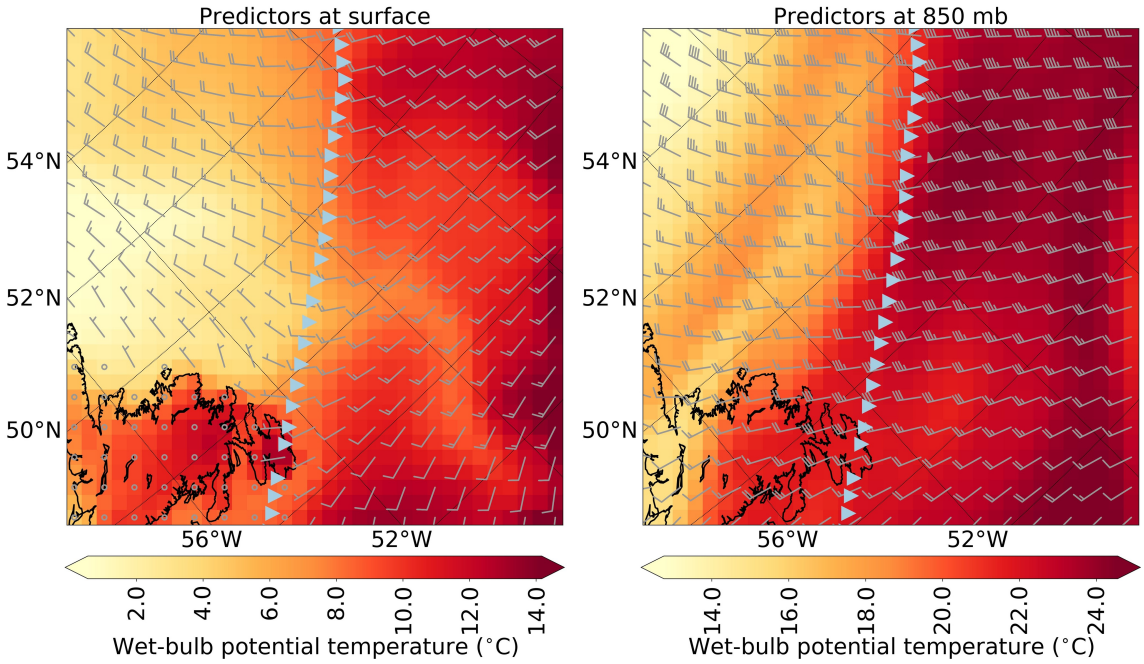


Figure 3.13: Patch for front detection. Formatting is explained in Figure 3.10. A cold front passes through the center grid cell (both before and after dilation), so the label is CF. Predictors may be at vertical levels other than the surface and 850 mb. Predictors may include multiple scalar fields and may or may not include θ_w.

Chapter 4

Convolutional Neural Networks

CNNs are briefly introduced at the beginning of Chapter 1. To reiterate, CNNs are a deep-learning method specially designed to learn from spatial grids. Although they were developed in the 1980s (Fukushima and Miyake 1982), CNNs did not become widely used until the 2010s (*e.g.*, Krizhevsky et al. 2012), for three reasons.

First, most people did not have access to the computing power needed to train CNNs. Whereas traditional neural networks (Chapter 6 of Goodfellow et al. 2016) typically have hundreds or thousands of weights to fit, CNNs often have millions. Two developments have made CNNs more computationally tractable: exponential growth of computing power and the adoption of graphical processing units (GPU) for CNN-training (*e.g.*, Kurth et al. 2018). GPUs are designed to perform matrix multiplication very quickly, and the convolutions inside a CNN are matrix multiplications.

Second, because CNNs have so many weights, the ratio of training examples to weights is often small, which can lead to overfitting. This problem has been alleviated by larger datasets and new regularization methods such as dropout, discussed later in this section, and data augmentation, discussed in Section 5.1.

Third, CNNs often have many layers, which can lead to vanishing gradients. CNNs are trained via gradient descent (Section 6.5 of Goodfellow et al. 2016), which computes the gradient of each model weight with respect to the loss $\left(\frac{\partial w}{\partial \varepsilon}\right)$. For a weight in the output layer, this calculation can be done directly. For a weight in any other layer ($\mathscr{L}$), this calculation requires backpropagating the loss through all layers between $\mathscr{L}$ and the output.[1]

[1]For example, if there is one layer ($\mathscr{M}$) between $\mathscr{L}$ and the output, computing $\frac{\partial w}{\partial \varepsilon}$ requires computing $\frac{\partial w}{\partial u}\frac{\partial u}{\partial \varepsilon}$ for all weights u in layer $\mathscr{M}$ that are influenced by weight w. The number of gradients in the product is $N+1$, where N is the number of layers between $\mathscr{L}$ and the output.

Due to the chain rule for differentiation, backpropagating through each layer involves multiplying by another gradient; these gradients are often $\ll 1$. Thus, backpropagating through many layers may yield a product small enough that rounding error becomes dominant. This causes the gradient estimates and ensuing weight updates to be essentially random, which prevents the weights from converging to an optimal solution. The vanishing-gradient problem has been alleviated by the rectified linear activation function and batch normalization, discussed later in this section.

The main components of a CNN are convolutional and pooling layers, which transform the inputs into abstractions called "feature maps," and dense layers, which transform the feature maps into predictions. The convolutional and pooling layers are spatially aware, while the dense layers are spatially agnostic. Figure 4.1 shows the architecture of a typical CNN, or how the layers fit together. A series of convolutional layers terminated by a pooling layer is called a "convolutional block"; there are two in Figure 4.1. Dense layers, also called "fully connected" layers, are the type used in a traditional neural network. Thus, a CNN can be viewed as a feature-detector (the convolutional and pooling layers) attached to a traditional neural network, which simultaneously learns the best transformation from inputs to features and the best transformation from features to predictions. This ability to learn the feature representation is a key advantage of CNNs over traditional ML, where the feature representation must be determined *a priori* and therefore is likely not the optimal representation for the prediction task.

A convolutional layer uses convolutional filters to transform the input maps. Convolutional filters have been used in image-processing for decades to perform operations such as blurring, sharpening, and edge detection (Chapter 5 of Shapiro and Stockman 2000). In these applications the filter weights are fixed, but in a CNN the weights are learned over time. Convolutional layers often increase the number of channels (*e.g.*, Figure 4.1), which increases the number of abstractions stored inside the CNN. For example, in Figure 4.1, the

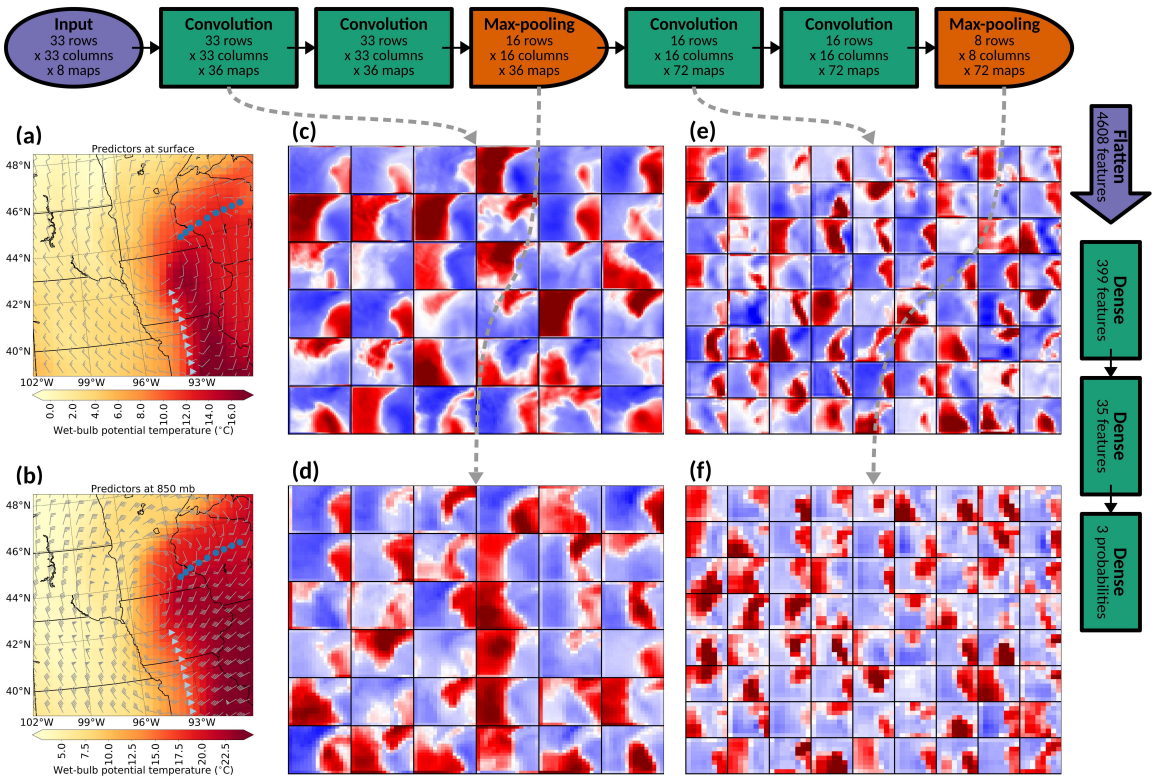


Figure 4.1: Example of CNN architecture. The input is a 33 × 33 grid with 8 predictors (T, q_v, u, and v at the surface and 850 mb). [a-b] Predictors, formatted as in Figure 3.10. [c] Feature maps produced by the first convolutional layer, after activation and batch normalization. Red values are positive, and blue values are negative. [d] Feature maps produced by the first pooling layer. [e] Feature maps produced by the third convolutional layer. [f] Feature maps produced by the second pooling layer. The flattening layer transforms maps from the last pooling layer into a vector of length 4608 (8 × 8 × 72), and the dense layers transform this vector into representations of exponentially decreasing length (4608 → 399 → 35 → 3), terminating with the prediction vector. The prediction vector contains probabilities of the three classes: no front, warm front, and cold front. This model contains 1.94 million adjustable weights, 1.85 million of which are in the dense layers.

input maps have 8 channels (all weather variables) and output maps from the first convolutional layer have 36 channels, each a transformation of the 8 original channels. Convolution is a spatial and multivariate operation, so it is performed over all spatial dimensions and all channels. Thus, each channel after the input layer can contain information from all original weather variables. Figure 4.2 is a schematic of univariate (single-channel) convolution[2], and Figure 4.3 is a schematic of multivariate convolution.[3] When the filter runs off the edge of the grid, the missing input values are assumed to be zero, which is the mean predictor value after normalization (Equation 3.4).

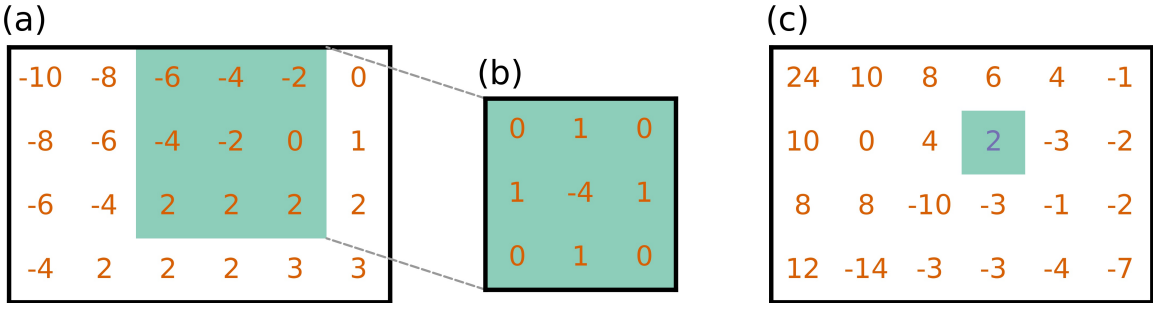


Figure 4.2: Schematic for univariate convolution in two spatial dimensions. [a] Input map. [b] Convolutional filter. Filter weights are set to values known to be useful for edge detection. In a real CNN these weights are learned during training. [c] Feature map produced by convolution. At the filter position shown, the highlighted value in c is the sum of element-wise products of highlighted values in a and b.

Convolution is defined mathematically by

$$\mathbf{X}_i^{(k)} = \sum_{j=1}^{N} \mathbf{W}_i^{(j,k)} * \mathbf{X}_{i-1}^{(j)} + b_i^{(k)}, \tag{4.1}$$

[2]See the following link for an animation, where the filter is moved to every position in the 2-D grid: `https://github.com/thunderhoser/GewitterGefahr/blob/dissertation_gifs/gewittergefahr/dissertation/supfig03_univariate_conv_animation.gif`

[3]See the following link for an animation, where the filter is moved to every position in the 2-D grid: `https://github.com/thunderhoser/GewitterGefahr/blob/dissertation_gifs/gewittergefahr/dissertation/supfig04_multivariate_conv_animation.gif`

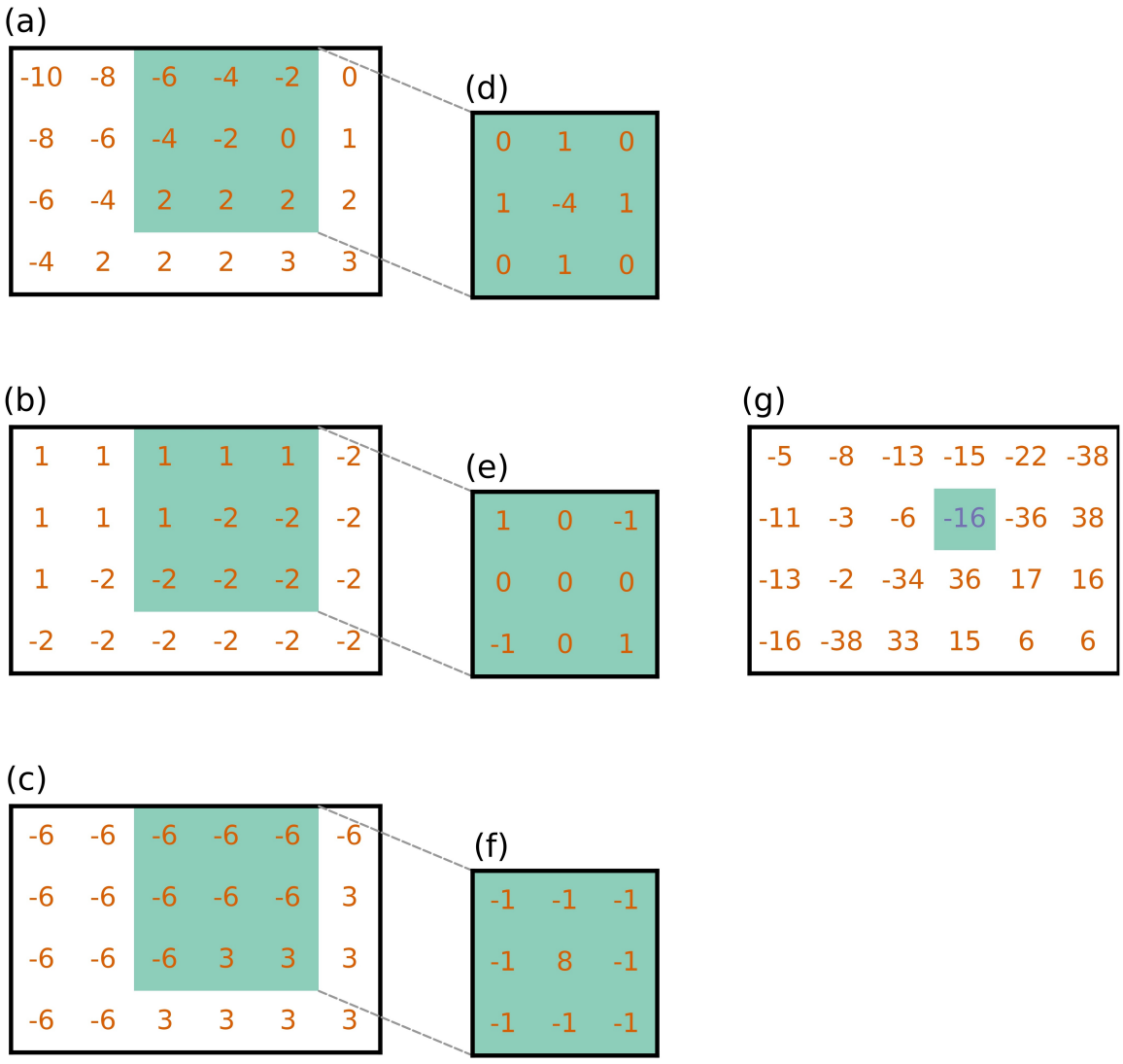


Figure 4.3: Schematic for multivariate convolution in two spatial dimensions. [a-c] Input maps, one for each variable. [d-f] The convolutional filter, one panel for each input variable. [g] Feature map produced by convolution. At the filter position shown, the highlighted value in g is the sum of elementwise products of highlighted values in a-f.

where $\mathbf{X}_{i-1}^{(j)}$ and $\mathbf{X}_{i}^{(k)}$ are feature maps; $\mathbf{W}_{i}^{(j,k)}$ is the convolutional filter; $b_i^{(k)}$ is the bias; and $*$ is the convolution operator. More specifically, $\mathbf{X}_{i-1}^{(j)}$ is the j^{th} channel in the feature map for the $(i-1)^{\text{th}}$ layer; $\mathbf{X}_{i}^{(k)}$ is the k^{th} channel in the feature map for the i^{th} layer; $\mathbf{W}_{i}^{(j,k)}$ is the filter connecting these two channels; $b_i^{(k)}$ is the bias for the k^{th} channel in the i^{th} layer; and N is the number of channels in the $(i-1)^{\text{th}}$ layer.

The feature maps $\mathbf{X}_{i-1}^{(j)}$ and $\mathbf{X}_{i}^{(k)}$ have the same dimensions, while the filter $\mathbf{W}_{i}^{(j,k)}$ typically has smaller dimensions. For example, in Figures 4.2 and 4.3, the feature maps are 4 $\times$ 6 grid cells and the filter is 3 $\times$ 3. Thus, the filter can detect features no larger than 3 $\times$ 3 grid cells. The convolution operator ($*$) is illustrated for univariate convolution (N = 1) in Figure 4.2 and for multivariate convolution ($N > 1$) in Figure 4.3. At each position of the filter $\mathbf{W}_{i}^{(j,k)}$, the convolution operator computes the sum of elementwise products of the filter weights and values in the input map $\mathbf{X}_{i-1}^{(j)}$ that fall inside the filter. For simplicity, both Figures 4.2 and 4.3 assume that the bias $b_i^{(k)}$ is zero.

Each convolutional layer includes operations after the convolution itself: activation and sometimes batch normalization (Ioffe and Szegedy 2015). Activation is a non-linear function applied elementwise to the output maps. Without this non-linearity, the CNN can learn only linear relationships, because convolution is a linear operation and a series of linear operations is still linear. All CNNs in this study use a popular activation function called the rectified linear unit (ReLU; Nair and Hinton 2010; Maas et al. 2013), plotted in Figure 4.4 and defined by

$$f(x) = \max(\beta x, x). \tag{4.2}$$

β, a hyperparameter in the range $[0, 1)$, is the slope for negative feature values. When β = 0, the function is called "strict ReLU"; when $\beta > 0$, the function is "leaky ReLU."

Strict ReLU is popular because it alleviates the vanishing-gradient problem. For positive inputs, strict ReLU preserves the magnitude of the feature value and thus preserves the magnitude of loss gradients ($\frac{\partial w}{\partial \varepsilon}$), which decreases the risk of many gradients multiplying to near zero. For negative inputs, strict ReLU decreases the magnitude to exactly

zero, which prevents the associated weights from being updated at all. However, this is a problem for weights consistently associated with negative feature values, which are never updated and therefore become "dead neurons" (Dubey and Jain 2019). Dead neurons are prevented by leaky ReLU, which preserves some information from negative feature values. However, leaky ReLU (used in this work) diminishes the magnitude of negative features, which reintroduces the possibility of vanishing gradients. Thus, batch normalization may be applied after the activation function. Batch normalization transforms each element of the feature maps (*i.e.*, each variable at each grid point) to approximately a Gaussian distribution, with a mean of 0.0 and standard deviation of 1.0.[4] Very small values are rare in such a distribution, which makes very small loss gradients rare, which decreases the risk of vanishing gradients.

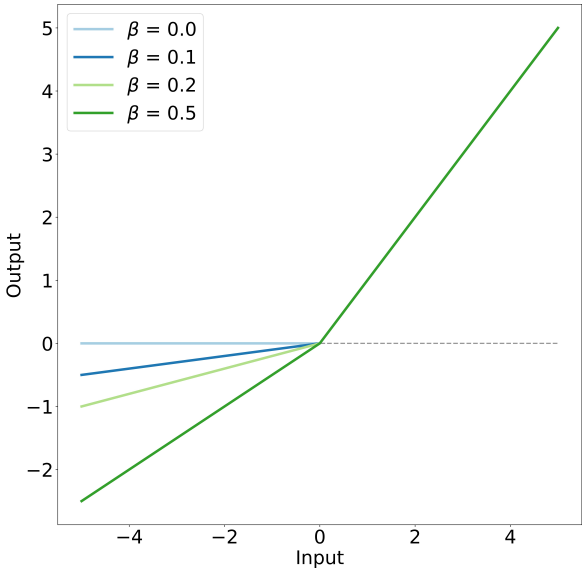


Figure 4.4: ReLU with various negative slopes.

Figure 4.5 shows the interplay of convolution, activation, and batch normalization. All convolutional filters in a network have different weights – the $\mathbf{W}_i^{(j,k)}$ and $b_i^{(k)}$ in Equation

[4]Parameters of the original distribution, which are used to transform to the Gaussian distribution, are estimated independently for each batch of training examples. For information on batch sizes used in this study, see Sections 5.1 and 5.2.

4.1 – which is ensured by random initialization. This allows different filters to detect different features. For example, in Figure 4.1, some filters respond strongly to thermal gradients while others respond strongly to wind shifts. As mentioned above, the weights are updated during training via gradient descent. Specifically, for each batch of examples presented to the network, the following rule is applied[5]:

$$\begin{aligned} \mathbf{W}_i^{(j,k)} &\leftarrow \mathbf{W}_i^{(j,k)} - \alpha \frac{\partial \varepsilon}{\partial \mathbf{W}_i^{(j,k)}} \\ \text{and } b_i^{(k)} &\leftarrow b_i^{(k)} - \alpha \frac{\partial \varepsilon}{\partial b_i^{(k)}} \end{aligned} . \tag{4.3}$$

ε is the loss, and α is the learning rate, a small positive number. By adjusting weights in the direction opposite the loss gradient, Equation 4.3 attempts to minimize ε.

This work uses the cross-entropy loss function, which is common for classification tasks, defined by

$$\varepsilon = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y_{ik} \log_2(p_{ik}). \tag{4.4}$$

N is the number of examples; K is the number of classes; p_{ik} is the model-predicted probability that the i^{th} example belongs to the k^{th} class; and y_{ik} is the true label, which is 1 if the i^{th} example belongs to the k^{th} class and 0 otherwise. Conceptually, cross-entropy is the number of bits needed to describe the difference between two data distributions: that of the predictions and that of the true labels. Thus, the best possible cross-entropy is zero. Cross-entropy ranges from $[0, \infty)$.

A pooling layer downsamples feature maps, using either a maximum or mean filter. Each channel is downsampled separately. Following common practice, this work uses a maximum filter with a downsampling factor of 2 and stride length of 2 (Figure 4.6),

[5]The update rule defined in Equation 4.3 is for "vanilla" backpropagation, the simplest method. For all models in this study, I use the Adam optimizer, which has a more complicated update rule (Algorithm 1 of Kingma and Ba 2017). The Adam optimizer generally leads to faster convergence of weights inside the CNN, in both the ML literature and my previous experience.

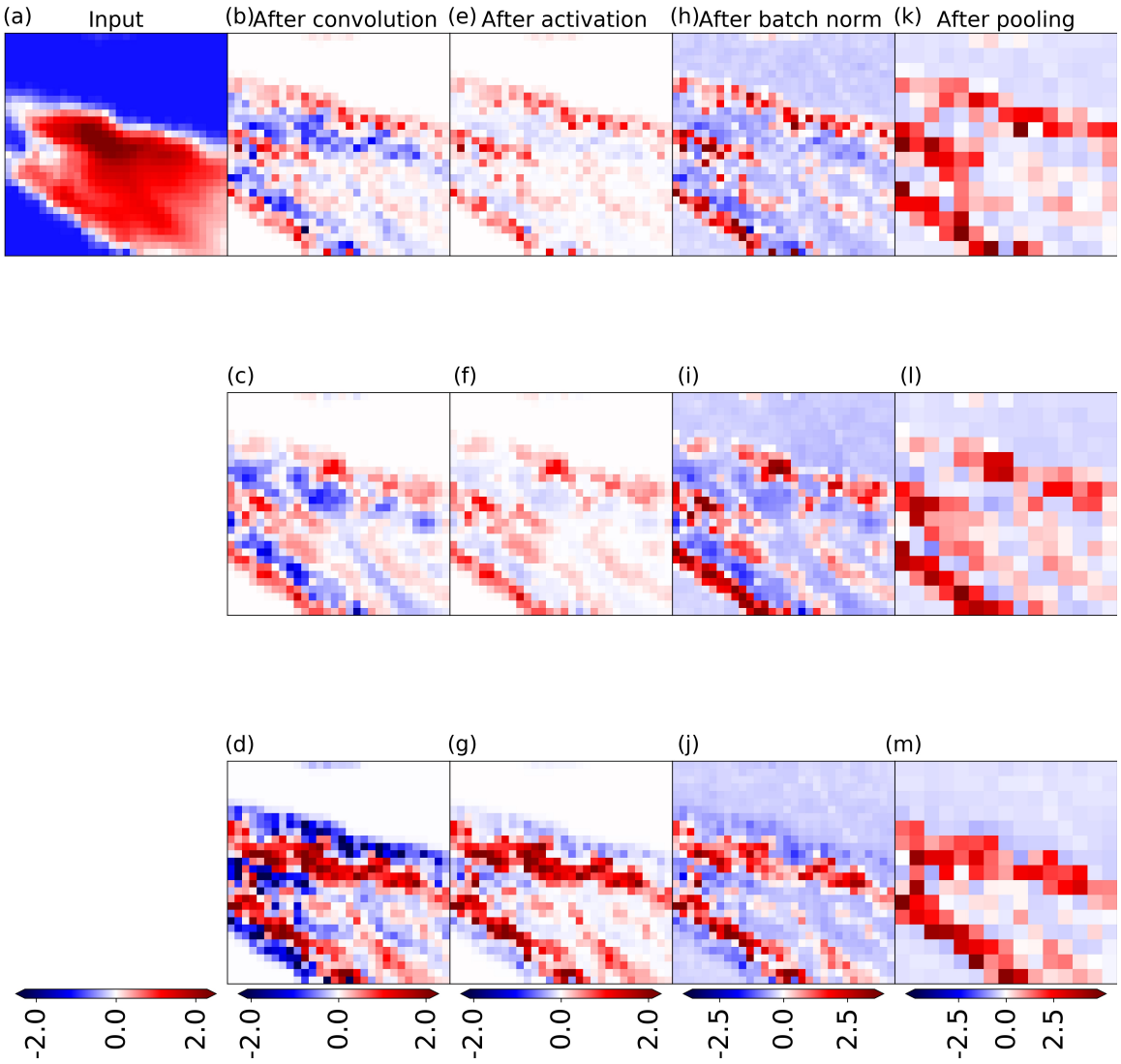


Figure 4.5: Convolutional block with one convolutional layer. If the block had two convolutional layers, batch normalization would be followed by another round of convolution, then activation, then batch normalization. There is one input channel and three output channels. [a] Input map. [b-d] Feature maps produced by convolution. Each panel is one channel. [e-g] Feature maps after leaky ReLU with slope of 0.2. Note that leaky ReLU leaves positive values untouched but reduces the magnitude of negative values. [h-j] Feature maps after batch normalization. [k-m] Feature maps after maximum-pooling with downsampling factor of 2.

which halves the spatial resolution (doubles the grid spacing).[6] For example, in Figure 4.1, pooling layers increase the grid spacing from 32 to 64 to 128 km, which allows deeper convolutional layers to learn larger-scale features. This, combined with the fact that feature maps in deeper layers have passed through more convolutions and non-linear activations, allows deeper layers to learn higher-level abstractions.

The dense layers (Figure 4.7) are spatially agnostic, so feature maps from the last pooling layer are flattened into a 1-D vector before they are passed to the dense layers (Figure 4.1).[7] Dense layers transform this vector into representations of varying (usually decreasing) length, terminating with the prediction vector. This transformation is a non-spatial version of the linear transformation performed by convolutional layers (Equation 4.1). All dense layers except the last follow this transformation with ReLU activation and batch normalization, like the convolutional layers. The last dense layer for tornado prediction uses the sigmoid activation function (Section 6.2.2.2 of Goodfellow et al. 2016), which forces the output to range from $[0, 1]$, allowing it to be interpreted as a probability. The last dense layer for front detection uses the softmax activation function (Section 6.2.2.3 of Goodfellow et al. 2016), which forces all outputs to range from $[0, 1]$ and forces their sum to equal 1.0, allowing them to be interpreted as probabilities of mutually exclusive and collectively exhaustive classes. The last dense layer does not use batch normalization, because this would force the outputs to a Gaussian distribution, which permits negative values and is therefore invalid for probabilities. Weights in the dense layers are learned via gradient descent, simultaneously with weights in the convolutional layers.

[6] See the following link for an animation, where the pooling window is moved to every position in the 2-D grid: `https://github.com/thunderhoser/GewitterGefahr/blob/dissertation_gifs/gewittergefahr/dissertation/supfig05_pooling_animation.gif`

[7] Space is implicitly represented in the dense layers, because features created by the convolutional and pooling layers, which are ultimately passed to the dense layers, are based on spatial maps. Thus, rather than spatially aware vs. spatially agnostic, the convolutional/pooling and dense layers, respectively, may be regarded as spatially explicit vs. spatially implicit.

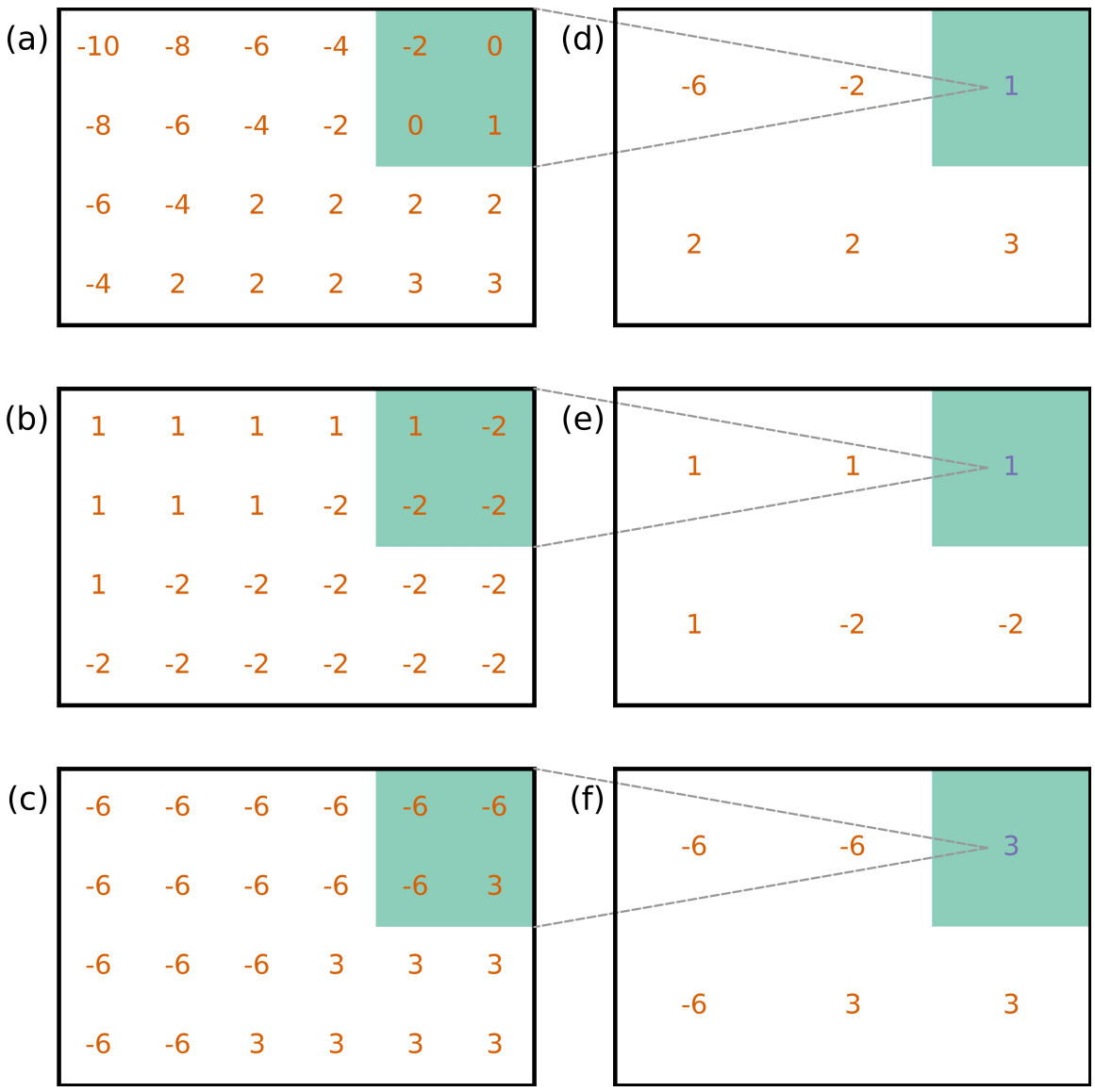


Figure 4.6: Schematic for maximum-pooling in two spatial dimensions. [a-c] Input maps, one for each variable. [d-f] Output maps, one for each variable. At each position of the pooling window, the highlighted value in (d, e, f) is the maximum of highlighted values in (a, b, c) respectively.

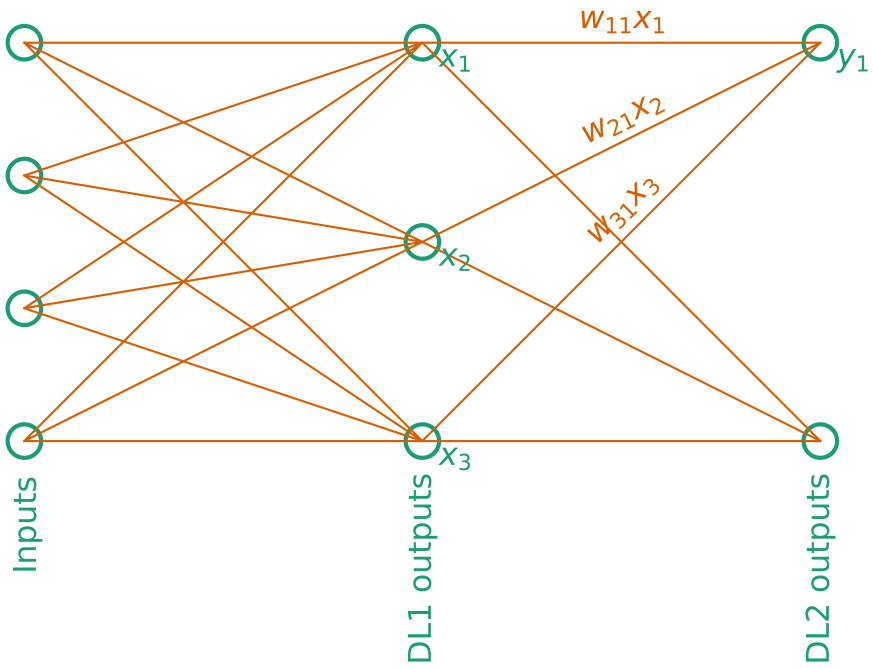


Figure 4.7: Schematic for dense layers. "DL1" and "DL2" are the first and second layers, respectively. The dense layers transform the feature vector from a length-4 representation to length-3, then length-2. In a CNN, inputs would be provided by the flattening layer and outputs from DL2 would be the predictions. Each line represents a weight, connecting two neurons in adjacent layers. $y_1 = b_1 + w_{11}x_1 + w_{21}x_2 + w_{31}x_3$, where b_1 is the bias for the first neuron in DL2; w_{i1} is the weight connecting the first neuron in DL2 to the i^{th} neuron in DL1; x_i is the post-activation feature value for the i^{th} neuron in DL1; and y_1 is the pre-activation feature value for the first neuron in DL2.

All CNNs in this work are trained with two types of regularization: a L_2 penalty (Section 4.4.2 of Chollet 2018) for the convolutional layers and dropout (Hinton et al. 2012) for the dense layers. In general, regularization encourages a simpler model, which reduces the risk of overfitting the training data. The L_2 penalty changes the loss function from Equation 4.4 to

$$L = \varepsilon + \lambda(\text{SSW}), \tag{4.5}$$

where ε is cross-entropy; SSW is the sum of squared weights in all convolutional layers; and λ, a hyperparameter, is the strength of the L_2 penalty. The L_2 penalty encourages the model to create smaller weights in general, creating large weights only when they are especially important for making skillful predictions. Meanwhile, dropout randomly omits fraction d of dense-layer weights when predicting each training example, where d is a hyperparameter. This forces redundancies in the behaviour of weights within the same layer (because for a given training example, any weight may be dropped out, forcing the other weights to "fill in"), which reduces overfitting.

Chapter 5

Hyperparameter Experiments

This chapter describes experiments used to find the best CNN hyperparameters for tornado prediction and front detection. The hyperparameters selected are those that yield the best performance on validation data, using a score deemed appropriate for the prediction task. Each experiment takes the form of a grid search (Section 11.4.3 of Goodfellow et al. 2016), which has the following procedure.

1. Decide the set of experimental hyperparameters.
2. Decide the values that will be attempted for each experimental hyperparameter.
3. For each possible set of hyperparameter values, train a model on the training data and evaluate it on the validation data.
4. Select the model that performs best on the validation data.
5. Evaluate the selected model on testing data (only once, at the end of the experiment, so that the selection is not influenced by testing data). This provides an independent assessment of model performance, based on data used for neither training nor model selection.

The disadvantage of grid search is that the number of possible combinations increases very quickly with the number of hyperparameters (combinatorial explosion). This can be alleviated by a random search (Section 11.4.4 of Goodfellow et al. 2016), where only some possible combinations are attempted. However, grid search often illuminates trends (*e.g.*, "model performance improves sharply as hyperparameter ζ increases from 0.0 to 0.2, then declines gradually as ζ increases beyond 0.2") that would not be discovered by random search, because random search samples the hyperparameter space too sparsely. Such trends

can provide insight into the model's inner workings and provide useful guidelines for future experiments.

CNN-training is subdivided into stages called epochs. In each epoch multiple batches of training examples are presented to the CNN, and the weights are updated after each batch as in Equation 4.3. At the end of each epoch, the loss is computed for both training and validation data, which can be used to diagnose overfitting. For example, in Figure 5.1, validation loss converges to a minimum at the 50th epoch and then overfitting begins.

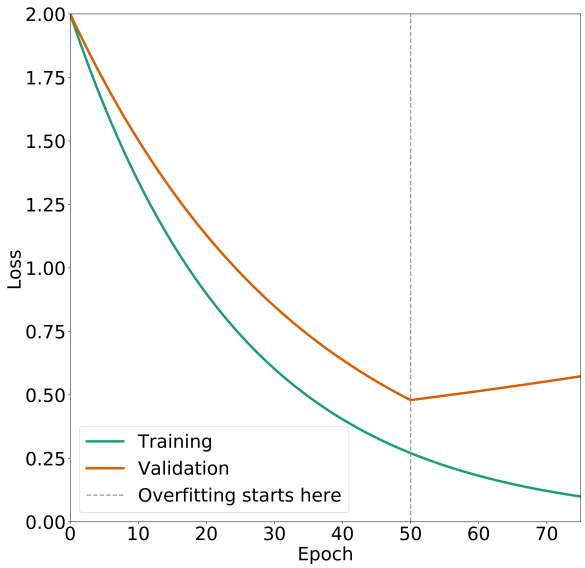


Figure 5.1: Theoretical schematic of overfitting. Overfitting occurs when the model learns relationships that allow for skillful prediction of the training data but not the validation data, causing the validation loss to increase over time while the training loss decreases.

For both tornado prediction and front detection, the CNNs are trained for 100 epochs, with 32 batches per epoch and $\sim$1000 examples per batch. These numbers are chosen to be large enough that validation loss always converges before training is complete. This occurs for every CNN in the following experiments, usually well before the 50th epoch, at which point they begin to overfit. Thus, after training is complete, weights are restored to the epoch with minimum validation loss (*e.g.*, the 50th epoch in Figure 5.1).

All CNNs use a slope of 0.2 for the leaky-ReLU activation function (Figure 4.4), which is the default in the Keras library. In my experience, performance is not highly sensitive to this value. Also, all CNNs use convolutional filters with three grid cells (the smallest value possible) in each spatial dimension, as in Figures 4.2 and 4.3. This allows the first convolutional layer to detect features at the finest scale possible, while pooling allows deeper convolutional layers to detect larger-scale features. In my experience, larger filters are detrimental to predictive skill. The supercomputer configurations I use for training CNNs are discussed in Appendix E.

5.1 Tornado Prediction (finding the best MYRORSS and GridRad models)

In this experiment a different CNN architecture is used for each dataset. GridRad contains 3-D radar images, so the associated CNNs, henceforth called "GridRad models," perform 3-D convolution (Figure 5.2). However, MYRORSS contains 3-D reflectivity images and 2-D shear images with different horizontal resolutions, which presents a challenge. To combine reflectivity and shear data in the same CNN, I upsample reflectivity to the same horizontal resolution as shear (from 0.75-km to 0.375-km spacing), using bicubic interpolation, then stack the 12 horizontal reflectivity maps with the two shear maps. The resulting image has two spatial dimensions, so the MYRORSS models perform 2-D convolution (Figure 5.3). In a previous experiment (not shown), I tried using 3-D convolution for the reflectivity images and 2-D convolution for the shear images. However, the approach shown in Figure 5.3 led to better predictions (differences of 0.02-0.03 in area under the receiver-operating-characteristic curve and critical success index).

In addition to radar images, the CNNs are trained with proximity soundings, which have one spatial dimension. Thus, each CNN contains two feature-detectors: one for radar images, which performs 2-D or 3-D convolution, and one for soundings, which performs 1-D convolution. The architecture of the sounding-feature-detector, shown in Figure 5.4, is

the same for both MYRORSS and GridRad models. Outputs from the two feature-detectors are flattened and concatenated, then passed through the dense layers at the end of the CNN, as shown in Figures 5.2 and 5.3. I do not experiment with the simple architectures (alternating convolutional and pooling layers) of these feature-detectors, because I have found that more complicated architectures lead to a large increase in computing time and no improvement in predictions. Including both feature-detectors, the GridRad model shown contains 0.46 million adjustable weights (0.15 million in the dense layers), while the MYRORSS model shown contains 0.77 million (0.34 million in the dense layers).

Examples in each training batch are drawn randomly from different time steps, which maximizes diversity within the batch – *i.e.*, ensures that most examples come from different storm cells and different synoptic situations – which reduces overfitting. Also, training data are undersampled so that each batch contains an equal number of tornadic and non-tornadic examples. Without undersampling, the CNNs would have little incentive to predict probabilities $\gg 0$, because the small fraction of tornadic examples would have little effect on the loss. Undersampling is used only for training. When a CNN is evaluated on the validation or testing data, undersampling is not used, so the evaluation scores fully reflect the difficulty of predicting a rare event.

The experiment, conducted separately for the MYRORSS and GridRad models, involves four hyperparameters: number of dense layers, dropout rate, L_2 weight, and data augmentation (Table 5.1). The number of dense layers affects performance strongly, because it is the main control on the number of model weights. The remaining architecture is fixed at one convolutional layer per block (Figures 5.2-5.4), and the number of blocks is chosen to yield a final domain size of 4-6 grid cells in each direction. I have found that smaller domains (*e.g.*, 2 grid cells per direction) do not contain enough spatial information to make skillful predictions, while larger domains (*e.g.*, 10 grid cells per direction) lead to too many weights in the dense layers, which makes training more computationally expensive and increases the risk of overfitting. The last three hyperparameters also control

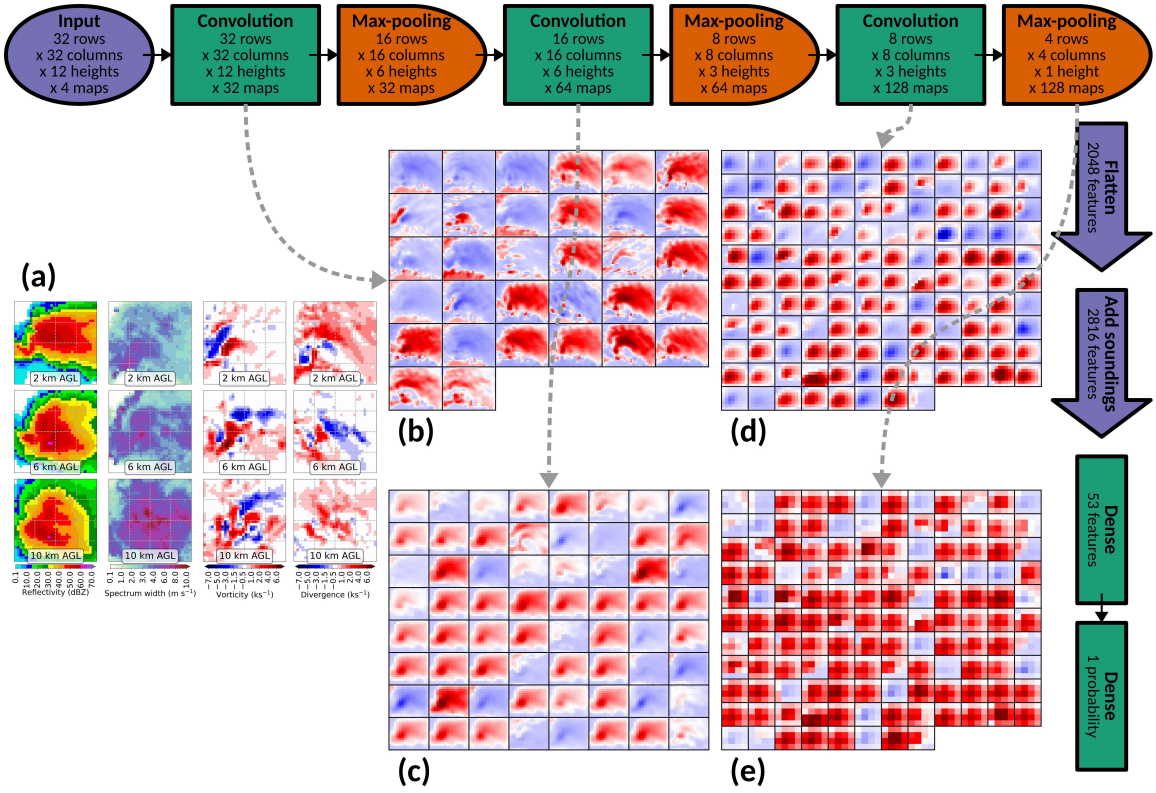


Figure 5.2: Architecture of one GridRad model. The architecture of other GridRad models differs only in the number of dense layers. [a] Radar predictors, consisting of a 32 $\times$ 32 $\times$ 12 grid with 4 maps. Only 3 of the 12 heights are shown, for the sake of brevity. There are also sounding predictors, shown in Figure 5.4. [b] Feature maps produced by the first convolutional layer, after activation and batch normalization. Red values are positive, and blue values are negative. [c] Feature maps produced by the second convolutional layer. [d] Feature maps produced by the last convolutional layer. [e] Feature maps produced by the last pooling layer. The flattening layer transforms these maps into a vector of length 2048 ($4 \times 4 \times 1 \times 128$), which is concatenated with sounding features produced by the layers shown in Figure 5.4. The dense layers transform this concatenated vector into representations of exponentially decreasing length ($2816 \rightarrow 53 \rightarrow 1$), and the final output is next-hour tornado probability.

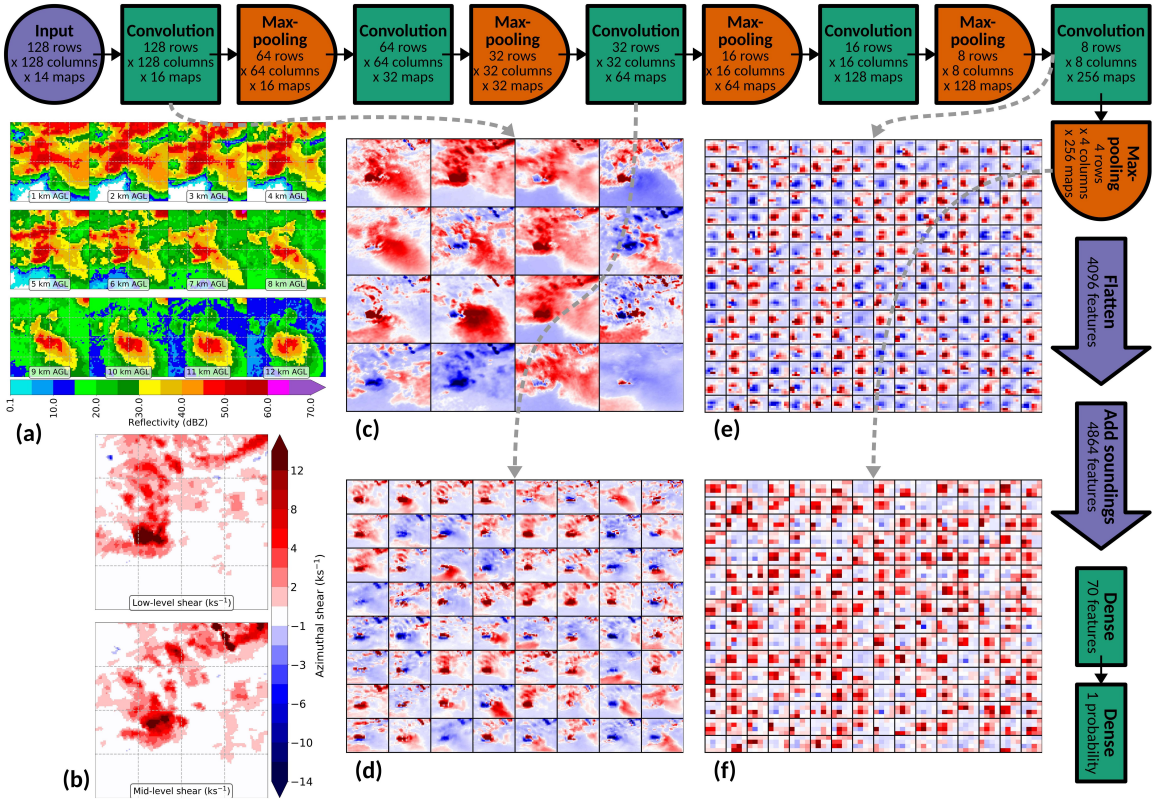


Figure 5.3: Architecture of one MYRORSS model. The architecture of other MYRORSS models differs only in the number of dense layers. [a-b] Radar predictors, consisting of a 128 $\times$ 128 grid with 14 maps. [c] Feature maps produced by the first convolutional layer, after activation and batch normalization. Red values are positive, and blue values are negative. [d] Feature maps produced by the third convolutional layer. [e] Feature maps produced by the last convolutional layer. [f] Feature maps produced by the last pooling layer. The flattening layer transforms these maps into a vector of length 4096 (4 $\times$ 4 $\times$ 256), which is concatenated with sounding features produced by the layers shown in Figure 5.4. The dense layers transform this concatenated vector into one value, next-hour tornado probability.

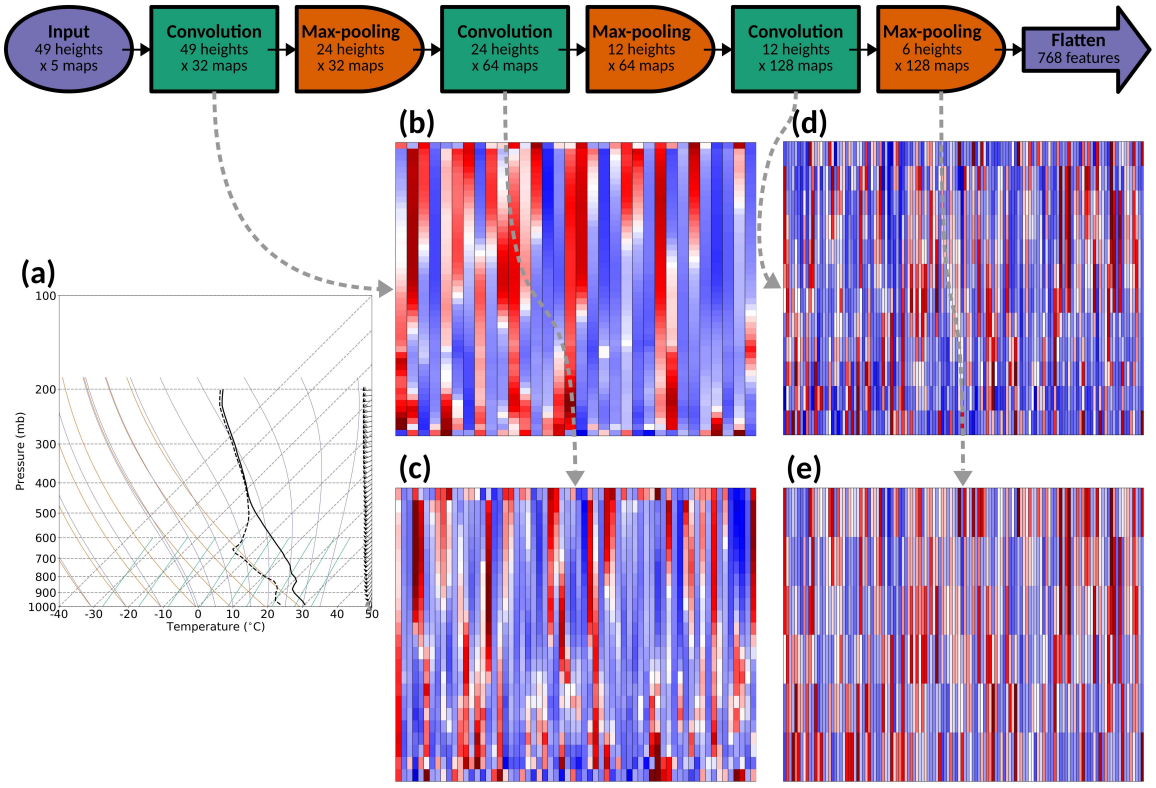


Figure 5.4: Architecture of sounding-feature-detector. [a] Input sounding, shown as a skew-T log-p diagram. The five variables fed to the CNN are u-wind, v-wind, relative humidity, specific humidity, and θ_v. [b] Feature maps produced by the first convolutional layer, after activation and batch normalization. Red values are positive, and blue values are negative. Each column is one channel, and each row is one height. [c] Feature maps produced by the second convolutional layer. [d] Feature maps produced by the last convolutional layer. [e] Feature maps produced by the last pooling layer. The flattening layer transforms these maps into a vector of length 768 (6×128), which is concatenated with radar features from either GridRad (Figure 5.2) or MYRORSS (Figure 5.3) data.

overfitting, which is a serious problem for tornado prediction, due to a large amount of uncertainty in the predictors (radar and NWP data) and labels (tornado reports). Dropout (Chapter 4) is used for all dense layers except the last, because I have found that using dropout in the last layer severely degrades performance. L_2 weight, or λ in Equation 4.5, controls the strength of L_2 regularization.

Table 5.1: Experimental hyperparameters for tornado prediction.

Hyperparameter	**Values**
Dropout rate	0.250, 0.375, 0.500, 0.625, 0.750
L_2 weight	10^{-3}, $10^{-2.5}$, 10^{-2}, $10^{-1.5}$, 10^{-1}
Data augmentation	On, off
Number of dense layers	1, 2

Data augmentation (Section 5.2.5 of Chollet 2018) is the practice of applying small perturbations to the predictors while assuming that the label (tornadic or not) remains the same. This reduces overfitting by preventing the CNN from learning relationships that are too specific. I apply 17 perturbations to the radar image for each example (Figure 5.5):

- Horizontal rotations (about the z-axis) of +15°, -15°, +30°, and -30°
- Horizontal translations of three grid cells in eight directions spaced equally from 0-315°
- Five additions of Gaussian noise, each with a standard deviation of 0.1. Since all predictors are normalized to the same scale (Equation 3.4), the magnitude of this effect is the same for all predictors.

The perturbations are applied separately, turning one example into 18 (the original example plus 17 perturbed ones). When data augmentation is turned on, each training batch contains 64 real examples (32 tornadic and 32 non-tornadic), perturbed 17 times each to create

a batch of 1152. When data augmentation is turned off, each training batch contains 1152 real examples (576 tornadic and 576 non-tornadic). The exact perturbations were decided by a previous experiment (not shown), which also attempted rotations up to $\pm 180^\circ$ and mirroring about the x- or y-axis. These perturbations degraded performance on the validation data, possibly because they changed the implied direction of storm motion (towards the right or positive x-direction) too drastically. Like dropout, data augmentation is used only during training. Thus, predictions for validation and testing data are based on real, not perturbed, examples.

After training, four scores are computed for each CNN on the validation data: probability of detection (POD), false-alarm ratio (FAR), frequency bias, and critical success index (CSI). The scores are defined as

$$\begin{aligned} \text{POD} &= \frac{a}{a+c}, \\ \text{FAR} &= \frac{b}{a+b}, \\ \text{bias} &= \frac{a+b}{a+c}, \\ \text{and CSI} &= \frac{a}{a+b+c}. \end{aligned} \tag{5.1}$$

a, b, c, and d are elements in the contingency table, defined in Table 5.2. POD is the fraction of tornadic examples that are correctly forecast and ranges from $[0,1]$, with 1.0 being the perfect score. FAR is the fraction of tornadic forecasts that are incorrect and ranges from $[0,1]$, with 0.0 being the perfect score. Frequency bias is the ratio of forecast to actual tornadic examples and ranges from $[0,\infty)$, with 1.0 being the perfect score. CSI is the accuracy excluding correct nulls and ranges from $[0,1]$, with 1.0 being the perfect score.

The contingency table is based on deterministic predictions (yes or no), while the CNNs output probabilities. The conversion from probabilistic to deterministic is done with a threshold, p^*. If probability $\geq p^*$, the deterministic prediction is yes; otherwise, the deterministic prediction is no. The threshold used is that which maximizes CSI on the validation data, which cannot be "played" (trivially optimized) as easily as the other scores. POD can

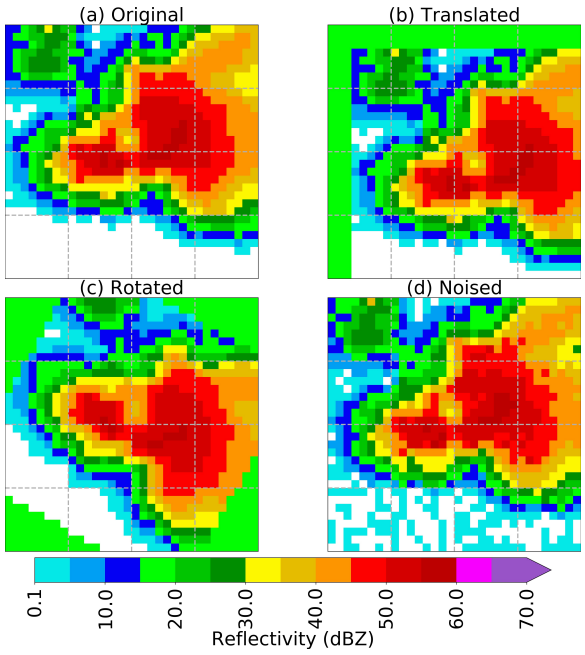


Figure 5.5: Example of data augmentation. This figure shows 3 of the 17 perturbations applied to a GridRad example. Only reflectivity at 3 km AGL is plotted, but the same perturbations are applied to each radar variable at each height. [a] The original reflectivity field. [b] The reflectivity field after translation by three grid cells to the bottom-right. Translation leaves missing values at the top-left, which are replaced with the mean reflectivity over training data (15.85 dBZ), the same value used for normalization in Equation 3.4. [c] The reflectivity field after rotation by 30° clockwise. Again, missing values caused by rotation are replaced with 15.85 dBZ. [d] The reflectivity field after adding Gaussian noise with a standard deviation of 0.1 normalized units (1.46 dBZ).

Table 5.2: Contingency table for binary classification. a is the number of hits (true positives); b is the number of false alarms (false positives); c is the number of misses (false negatives); and d is the number of correct nulls (true negatives).

Forecast	**Observation**	
	Yes	No
Yes	a	b
No	c	d

be played by always predicting yes; FAR can be played by always predicting no; and frequency bias can be played over a long time period by predicting the climatological number of tornadoes, regardless of whether or not they are predicted for the correct storms.

However, the final ranking of CNNs is based on area under the receiver-operating-characteristic curve (area under the ROC curve or AUC; Metz 1978), defined by

$$\text{AUC} = \int_{\text{POFD}=0}^{\text{POFD}=1} (\text{POD})\, d(\text{POFD}). \tag{5.2}$$

POFD is probability of false detection ($\frac{b}{b+d}$), or the fraction of non-tornadic examples that are incorrectly forecast. Because AUC integrates over all probability thresholds, it also cannot be easily played.

Results of this experiment are presented in Section 6.1.

5.2 Front Detection

This section describes an experiment used to find the best CNN, then an experiment to find the best hyperparameters for converting CNN probabilities into frontal zones.

5.2.1 Finding the Best CNN

For front detection, the CNNs perform patch classification, where the center grid cell of each 33 $\times$ 33 patch (Figure 3.13) is labeled independently. The alternative is semantic segmentation, where all grid cells are labeled at the same time. The advantage of semantic segmentation is computational efficiency: the predictors at each grid cell are used only once, rather than 1089 (33 $\times$ 33) times. To label the full grid for the climatology (Chapter 8), the patch-classification models developed herein take 10-30 minutes on one computing core (depending on the number of model weights), whereas a semantic-segmentation model would take $<$ 1 minute. The advantage of patch classification is that it allows the training data to be undersampled spatially. In a spatial sense, fronts are a rare event, covering an average of 2.43% of grid cells (Table 3.5) at a given time. For semantic segmentation, where the full grid must be labeled at once, the models thus have little incentive to produce front probabilities $\gg 0$. However, for patch classification, patches can be undersampled so that the training data are more balanced. Specifically, I undersample so that each training batch contains an equal number of frontal and non-frontal patches (256 warm fronts, 256 cold fronts, 512 non-fronts). As for tornado prediction, I draw examples in each batch randomly from different time steps, which discourages overfitting by reducing within-batch autocorrelation. Also, I do not undersample validation or testing data.

The experiment involves four hyperparameters (Table 5.3). The first is the set of predictor variables, which always includes both wind components and at least one thermal variable. Wind is always included because it is crucial for distinguishing stationary fronts, such as those often associated with orographic and land-ocean boundaries, from mobile fronts. Thermal variables may include T and q_v, the fundamental quantities for temperature and moisture. Thermal variables may also include θ_w, which is often used for front detection (*e.g.*, B11a, B11b, Catto and Pfahl 2013) because it is conserved during psuedo-adiabatic processes. Lastly, geopotential height is sometimes used as a predictor, because fronts are often associated with height troughs.

Table 5.3: Experimental hyperparameters for front-detection CNN. Each predictor variable is included at all vertical levels. Geopotential height (Z) is replaced with pressure at the surface level.

Hyperparameter	**Values**
Predictor variables (other than u and v at all vertical levels, which are always included)	θ_w
	T and q_v
	θ_w, T, and q_v
	θ_w and Z
	T, q_v, and Z
	θ_w, T, q_v, and Z
Vertical levels	Surface
	1000 mb
	Surface and 1000 mb
	Surface and 950 mb
	Surface and 900 mb
	Surface and 850 mb
Number of convolutional blocks	2, 3
Number of convolutional layers per block	1, 2, 3, 4

The second hyperparameter is the set of vertical levels for the predictors. In previous work (L19) each CNN was trained with only one level, but fronts are 3-D phenomena, so I hypothesize that training with multiple levels will improve performance. Ideally, I would train with full 3-D images, containing many vertical levels, and use 3-D convolution. However, this would be computationally intractable, as it takes approximately one week to download and process ERA5 data for the full climatological period (1979-2018) at one level. The attempted vertical levels (Table 5.3) are chosen to determine if there is benefit to training with multiple levels and, if so, which level adds the most benefit beyond the surface. Every CNN includes the surface or 1000-mb level, because (a) the eventual goal is to create a climatology of surface fronts; (b) the WPC bulletins nominally contain surface fronts; (c) despite this, L19 found that 1000 mb is the best single level.

The third and fourth hyperparameters are the number of convolutional blocks and number of convolutional layers per block. The number of blocks is chosen to yield a final domain size of 4×4 or 8×8, which is a middle ground between preserving too little and too much spatial information, as discussed in Section 5.1. The number of layers per block controls the number of transformations – one linear convolution and one non-linear activation per layer – at each spatial scale, which controls the complexity of features that can be learned. Unlike the tornado experiment, this experiment does not include the number of dense layers. In previous experiments on front detection, I have found that performance generally improves with the number of dense layers. However, adding a fourth layer yields little improvement and sharply increases the number of model weights and training time. Thus, all CNNs for front detection have three dense layers, as in Figure 4.1.

Also unlike the tornado experiment, this experiment does not include hyperparameters for regularization, because I have found that overfitting is a lesser concern for front detection. I speculate that this is because (a) there is more training data for front detection; (b) there are more positive training examples[1] for front detection; (c) fronts occur at every time

[1] A "positive example" is one in any class except the null class – in this case, a frontal or tornadic example.

step in the dataset, whereas tornadoes occur at only a small fraction of time steps, leading to more diversity among positive examples for front detection. The CNNs for front detection use L_2 regularization with a weight of 0.001, dropout with a rate of 0.5, and no data augmentation. All attempts to use data augmentation for front detection have degraded performance on the validation data, possibly because the training set is large enough to adequately sample the predictor space without augmentation. As for tornado prediction (Section 5.1), dropout is used for all dense layers except the last.

After training, CNNs are ranked by area under the ROC curve (AUC). Each ROC curve is computed for 1 million patches selected randomly from 1000 time steps in the validation data. The 1 million patches are selected once, so all CNNs are validated on the same data. As for tornado prediction, AUC is defined by Equation 5.2. However, because tornado prediction is a binary problem and front detection is multiclass, the POD and POFD in Equation 5.2 are defined differently. Specifically, the POD and POFD for front detection are defined by

$$\begin{aligned} \text{POD} &= \frac{n_W^W + n_C^C}{(n_N^W + n_W^W + n_C^W) + (n_N^C + n_W^C + n_C^C)} = \frac{n_W^W + n_C^C}{n^W + n^C} \\ \text{and POFD} &= \frac{n_W^N + n_C^N}{n_N^N + n_W^N + n_C^N} = \frac{n_W^N + n_C^N}{n^N}, \end{aligned} \tag{5.3}$$

where n_i is the number of patches for which class i is predicted; n^j is the number for which class j is observed; and n_i^j is the number for which class i is predicted and class j is observed. The classes are N for no front, W for warm front, and C for cold front. Conceptually, POD is the fraction of frontal patches that are correctly labeled, while POFD is the fraction of non-frontal patches that are incorrectly labeled. As for tornado prediction, POD ranges from $[0,1]$ with a perfect score of 1.0, while POFD ranges from $[0,1]$ with a perfect score of 0.0.

As for tornado prediction, Equations 5.3 require deterministic labels rather than probabilities. Deterministic labels are created by applying a threshold, p^*, to the sum of warm- and cold-front probabilities, $p_W + p_C$. If $p_W + p_C \geq p^*$, the deterministic label is cold front if $p_C \geq p_W$ and warm front otherwise; if $p_W + p_C < p^*$, the label is no front.

Visually, the main difference between ROC curves for tornado prediction and front detection is the top-right point, which is lower for front detection (*c.f.*, Figures 6.3 and 6.17). The top-right point corresponds to a probability threshold of 0.0, which means that only positive classes are predicted. For tornado prediction there is only one positive class, so all tornadoes are correctly predicted and POD = 1.0. However, for front detection there are two positive classes (warm and cold front), so even when only the positive classes are predicted, a warm front may be confused for a cold front and vice-versa. Mathematically, POD simplifies to $\frac{n_W^W + n_C^C}{(n_W^W + n_C^W) + (n_W^C + n_C^C)}$, which is generally less than 1.0. Thus, the top-right point in ROC curves for front detection is lower, which causes AUC to be lower.

Results of this experiment are presented in Section 6.2.

5.2.2 Finding the Best Object-conversion Method

The eventual climatology should reflect the fact that fronts are spatial objects. However, the raw output of the CNN consists of three probabilities at each ERA5 grid cell. The following method, applied separately at each time step, is used to convert CNN probabilities to objects (frontal zones).

1. Turn CNN probabilities into deterministic labels. This is done by applying the following rule to each grid point:

$$\begin{aligned} \text{if } p_C \geq p_C^* \text{ and } p_C \geq p_W &: \hat{y} = \text{C}, \\ \text{else if } p_W \geq p_W^* \text{ and } p_C < p_W &: \hat{y} = \text{W}, \\ \text{else} &: \hat{y} = \text{N}. \end{aligned} \tag{5.4}$$

   p_N, p_W, and p_C are the probabilities of no front, warm front, and cold front, respectively; p_W^* and p_C^* are thresholds; and $\hat{y}$ is the resulting deterministic label. See Figure 5.6a-b.

2. Find frontal zones. A frontal zone is a set of eight-connected[2] grid cells with the same label, either warm or cold. See Figure 5.6c.

3. Eliminate spurious frontal zones. A spurious zone is a small zone (major axis < 200 km) that is not within 200 km of a large zone of the same type (warm or cold). When a small zone is close to a large zone, they are often part of the same frontal zone but separated by a small gap of weaker thermal gradient or advection. The 200-km threshold is the minimum synoptic length scale (Orlanski 1975). In Figure 5.6d, the small cold-frontal zone off the coast of California is eliminated while the small cold-frontal zone in Saskatchewan is retained, because the latter is within 200 km of a large cold-frontal zone.

The above method involves two hyperparameters, the thresholds p_W^* and p_C^*. I perform a grid search, over the values shown in Table 5.4, to find the best thresholds for the best CNN from Section 5.2.1. Each pair of thresholds is evaluated on the same 1000 time steps selected randomly from the validation period.

Ideally, I would run this experiment for all CNNs, to find the best combination of CNN and probability thresholds. However, this would be computationally intractable. In the CNN experiment (Section 5.2.1) each model is applied to 1 million validation patches, but in this experiment the model must be applied to full grids, to create probability grids like that shown in Figure 5.6b. Each grid contains 45 893 patches.[3] Thus, over 1000 time steps the model must be applied to 45.893 million patches, which takes $\sim$200 core-hours of computing time. With parallelization across multiple cores, this is feasible for one model but not for all 288 models.

[2]Two grid cells are "eight-connected" if they share an edge or corner. Conversely, two grid cells are "four-connected" only if they share an edge.

[3]This is the number of grid points in a 200-km buffer around the unmasked grid points shown in Figure 3.12c. A 200-km buffer is needed to avoid edge effects, because this is the minimum length used to define frontal zones.

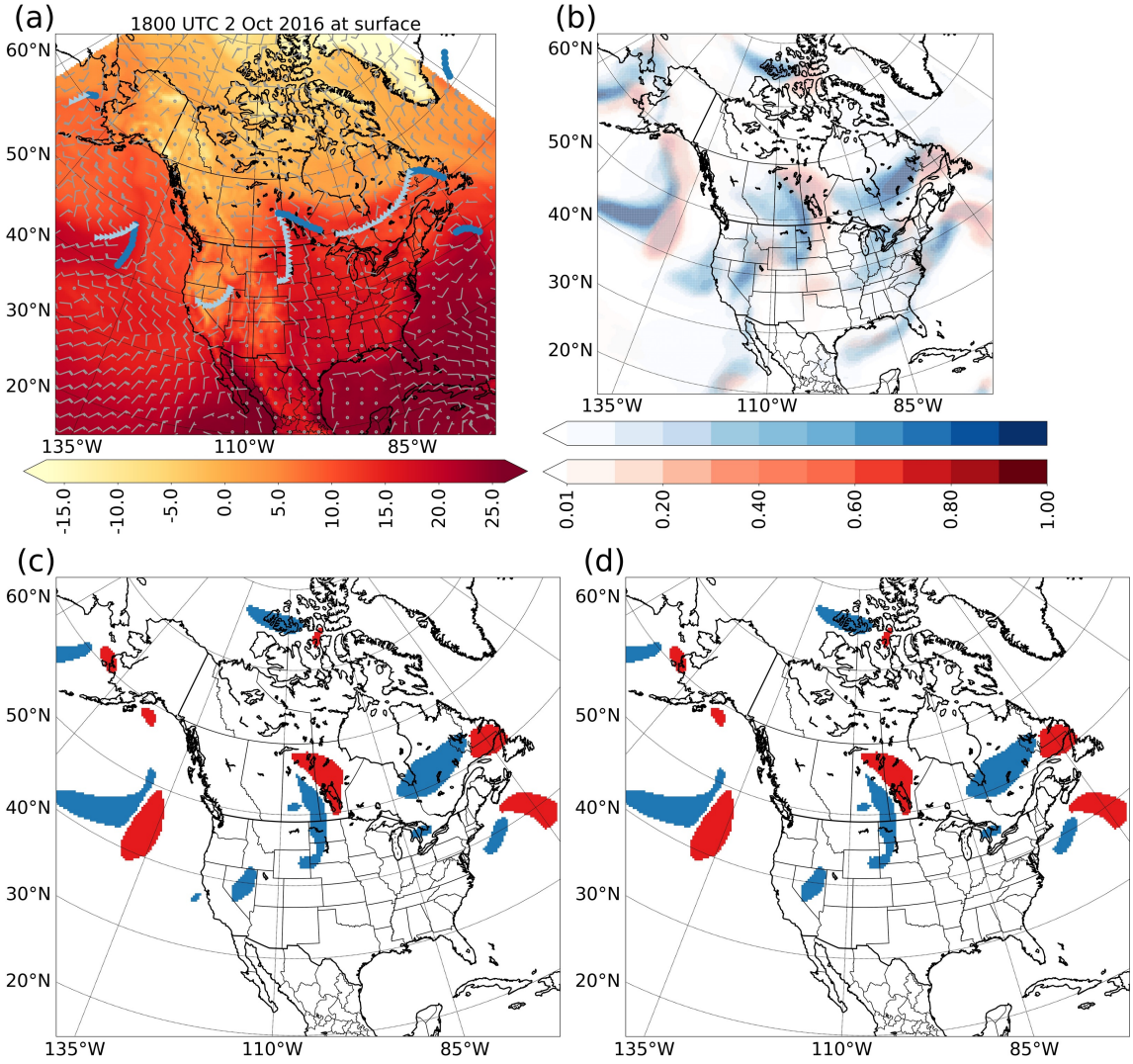


Figure 5.6: Object-conversion procedure for fronts. [a] Surface fields and WPC fronts at 1800 UTC 2 Oct 2016. Formatting is explained in Figure 3.10. [b] Gridded warm-front (red) and cold-front (blue) probabilities. [c] frontal zones, based on thresholds $p_W^* = p_C^* =$ 0.65 in Equation 5.4. [d] Same but after eliminating spurious zones.

Table 5.4: Experimental hyperparameters for object conversion. p_W^* and p_C^* are defined in Equation 5.4.

Hyperparameter	Values
p_W^*	0.200, 0.225, …, 0.800
p_C^*	0.200, 0.225, …, 0.800

After object conversion, four scores are computed for each set of p_W^* and p_C^*: POD, FAR, frequency bias, and CSI. These scores are also used for tornado prediction (Section 5.1), but they have different definitions in this setting. The reason is that correct nulls exist for tornado prediction (storms that are correctly predicted as non-tornadic in the next hour) but, after object conversion, do not exist for front detection. There are only warm-frontal and cold-frontal objects, no non-frontal objects. This is similar to the setting in which the NWS evaluates tornado warnings: each case is a segment of a tornado track, and there is no such thing as a non-tornado track, so there are no correct nulls. From Brooks (2004), the NWS defines the four scores as

$$\begin{aligned} \text{POD} &= \frac{a_A}{a_A + c}, \\ \text{FAR} &= \frac{b}{a_P + b}, \\ \text{Bias} &= \frac{\text{POD}}{1 - \text{FAR}}, \\ \text{and } \text{CSI}^{-1} &= \text{POD}^{-1} + (1 - \text{FAR})^{-1} - 1. \end{aligned} \tag{5.5}$$

The variables a_A, a_P, b, and c are defined below. "Actual" means according to the WPC data; "predicted" means according to the model after object conversion; and two grid points are "matched" when they are within a certain distance of each other, called the neighbourhood distance. For a neighbourhood distance of d_N, only grid points not within d_N of a masked grid point (Figure 3.12c) are used for evaluation. For a neighbourhood distance of 100 km, for example, this leaves 37 721 grid points.

- a_A = number of actual-oriented true positives, or actual frontal grid points matched with a predicted grid point of the same type (warm or cold)
- a_P = number of prediction-oriented true positives, or predicted frontal grid points matched with an actual grid point of the same type
- b = number of false positives, or predicted frontal grid points **not** matched with an actual grid point of the same type
- c = number of false negatives, or actual frontal grid points **not** matched with a predicted grid point of the same type

Despite the different definitions, each score has the same range and optimal values as listed in Section 5.1. Pairs of probability thresholds are ranked by CSI, since the other scores can be easily played, as discussed in Section 5.1. Results of this experiment are presented in Section 6.2.2.

Chapter 6

Performance Evaluation for CNNs

This chapter presents results of the hyperparameter experiments described in Chapter 5. Also, this chapter thoroughly evaluates the best model from each hyperparameter experiment on testing data. Before presenting results, I caution against two possible misinterpretations.

First, for each CNN experiment (finding the best MYRORSS model, GridRad model, and front-detection model), I have trained only one CNN for each configuration (set of hyperparameters). Since the training procedure is stochastic (Keras initializes model weights randomly, and the training examples in each batch are drawn randomly), there is sampling variability in the results. In other words, if the CNN experiments were run again, results would not be exactly the same. Ideally, I would have trained many CNNs with each configuration, allowing me to create confidence intervals and assess when one configuration is *significantly* better than another. However, each CNN experiment takes $\sim$20 000 core-hours of training time, which makes this approach infeasible. Thus, when discussing results on the validation data, I will focus on trends that occur across many hyperparameter configurations (*e.g.*, "data augmentation leads to better/worse performance"), rather than comparing individual configurations.

Second, when showing performance of the selected models on testing data, I will include confidence intervals. Note that these confidence intervals are created by bootstrapping data examples in the testing set, so they reflect sampling variability over the testing data. They do *not* reflect sampling variability over the space of possible models (CNNs that might have been obtained with the same hyperparameter configuration).

6.1 Tornado Prediction (best MYRORSS and GridRad models)

Figure 6.1 shows AUC and CSI for the 100 GridRad models on validation data (see Figure B.1 for POD, FAR, and frequency bias). The models clearly perform best with two dense layers and data augmentation turned on. The other two hyperparameters, L_2 weight and dropout rate, have a much smaller effect on performance. This suggests that data augmentation is the most effective regularization method for GridRad data. Although data augmentation increases the effective sample size much less than it increases the nominal sample size (because the 17 perturbed examples created from each original example are highly correlated), it clearly increases the effective sample size enough to improve predictions on independent data.

Figure 6.2 shows AUC and CSI for the 100 MYRORSS models on validation data (see Figure B.2 for POD, FAR, and frequency bias). Note that the MYRORSS models have much lower CSI than the GridRad models, due to lower event frequency (percentage of examples labeled "yes" or tornadic in the next hour). Specifically, event frequency is 2.96% for the GridRad validation data and 0.089% for the MYRORSS validation data. This is because MYRORSS data are created for every day of the year over the full CONUS, while GridRad data are centered around tornado outbreaks (Figure 3.1). For rare events it is difficult to make many positive forecasts without a large percentage of them being incorrect, which is required for a high POD and low FAR, both of which are required for a high CSI. As for GridRad, the MYRORSS models generally perform best with two dense layers and data augmentation turned on, but here the difference is less dramatic.

Figure 6.3 shows the performance of the selected MYRORSS and GridRad models on testing data, which are independent of the data used for training and hyperparameter-tuning (validation). Each point in the ROC curve or performance diagram (Roebber 2009) corresponds to one probability threshold. As the threshold increases from 0.0 to 1.0, POD and POFD decrease from 1.0 to 0.0, while FAR decreases from the event frequency to 0.0. The

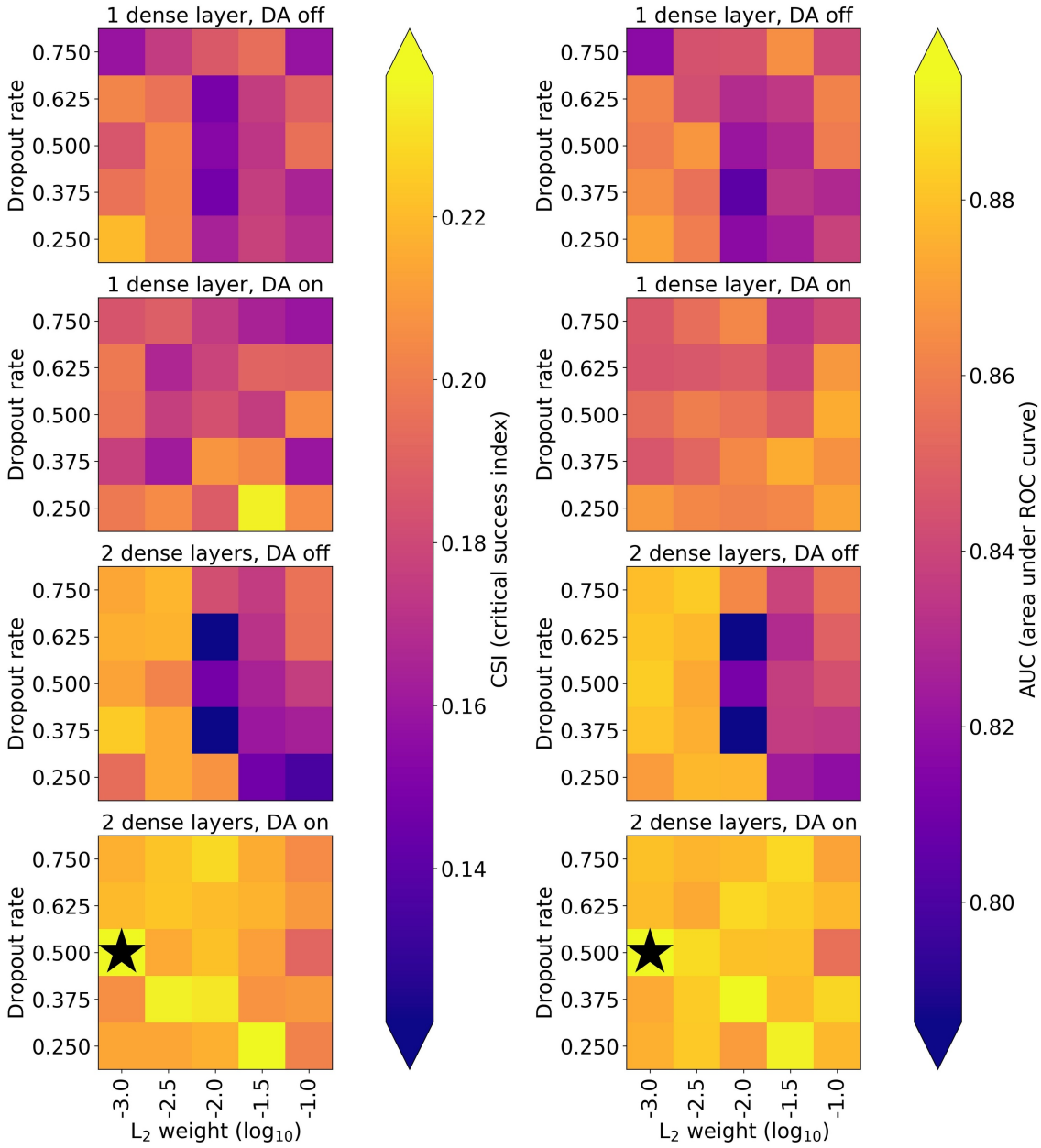


Figure 6.1: AUC and CSI for GridRad models on validation data. "DA" stands for data augmentation. The selected model (that with the highest AUC) is marked with a star.

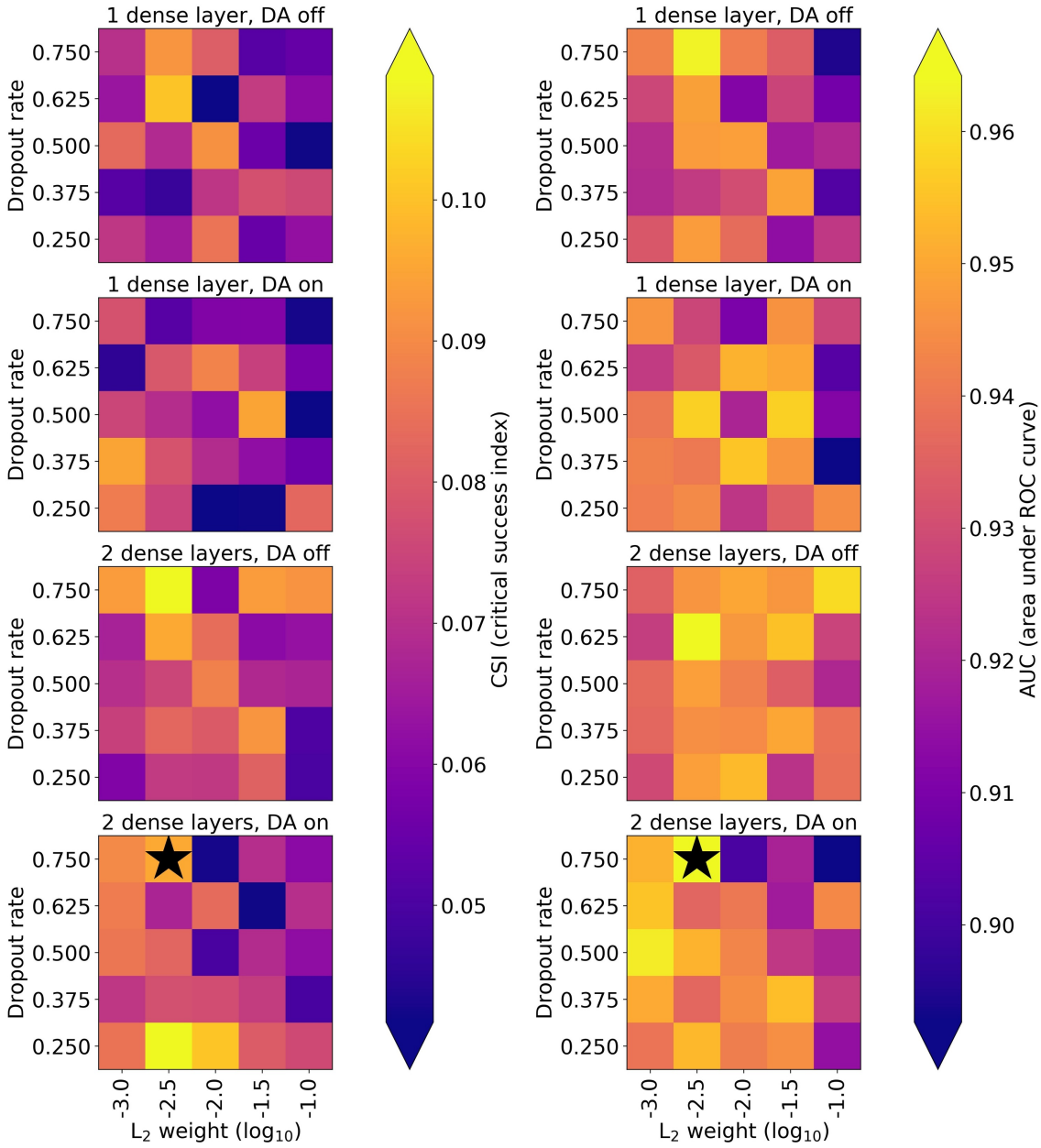


Figure 6.2: AUC and CSI for MYRORSS models on validation data. Formatting is explained in Figure 6.1.

shading around each curve represents the 95% confidence interval, determined by bootstrapping the testing examples 1000 times. The area under each curve (AUC or AUPD) is also reported as a 95% confidence interval, determined in the same way. The GridRad testing data contain 137 270 examples with an event frequency of 3.43%, while the MYRORSS testing data contain 4 608 403 examples with an event frequency of 0.22%. AUC > 0.9 for both models, which is generally considered the threshold for "excellent" performance (*e.g.*, Luna-Herrera et al. 2003; Muller et al. 2005; Mehdi et al. 2011). Because the ROC curve is insensitive to event frequency, this threshold can be used across prediction tasks. No such threshold exists for the performance diagram, which is highly sensitive to event frequency, as shown by the difference between MYRORSS and GridRad models. The stars in Figure 6.3 show the chosen probability threshold (that which maximizes validation CSI) for each model. This threshold nearly maximizes CSI on the testing data as well (shown in the performance diagram), but it leads to an unacceptably low POD (0.27 for MYRORSS and 0.48 for GridRad) for a costly event such as tornadoes. Considering that false negatives have a much greater cost than false positives, if one could assign numerical values to these costs, one could choose the probability threshold that minimizes cost. This would result in a lower threshold, causing the star to move towards the top-right of the ROC curve and top-left of the performance diagram – yielding a lower CSI and higher POD, POFD, FAR, and frequency bias.

Figure 6.4 is the same as Figure 6.3 but for strong tornadoes only (EF-2+), which suffer from less underreporting bias, as discussed at the end of Section 3.1.1. The testing sets used in Figure 6.4 are created by simply removing EF-0 and EF-1 tornadoes from those used in Figure 6.3. This decreases the GridRad event frequency from 3.43% to 1.36% and the MYRORSS event frequency from 0.22% to 0.06%. Assuming that model skill does not vary with tornado strength, one would expect this change to yield a similar ROC curve and worse performance diagram. However, the ROC curve improves significantly and the performance diagram remains roughly the same. This suggests that model skill improves

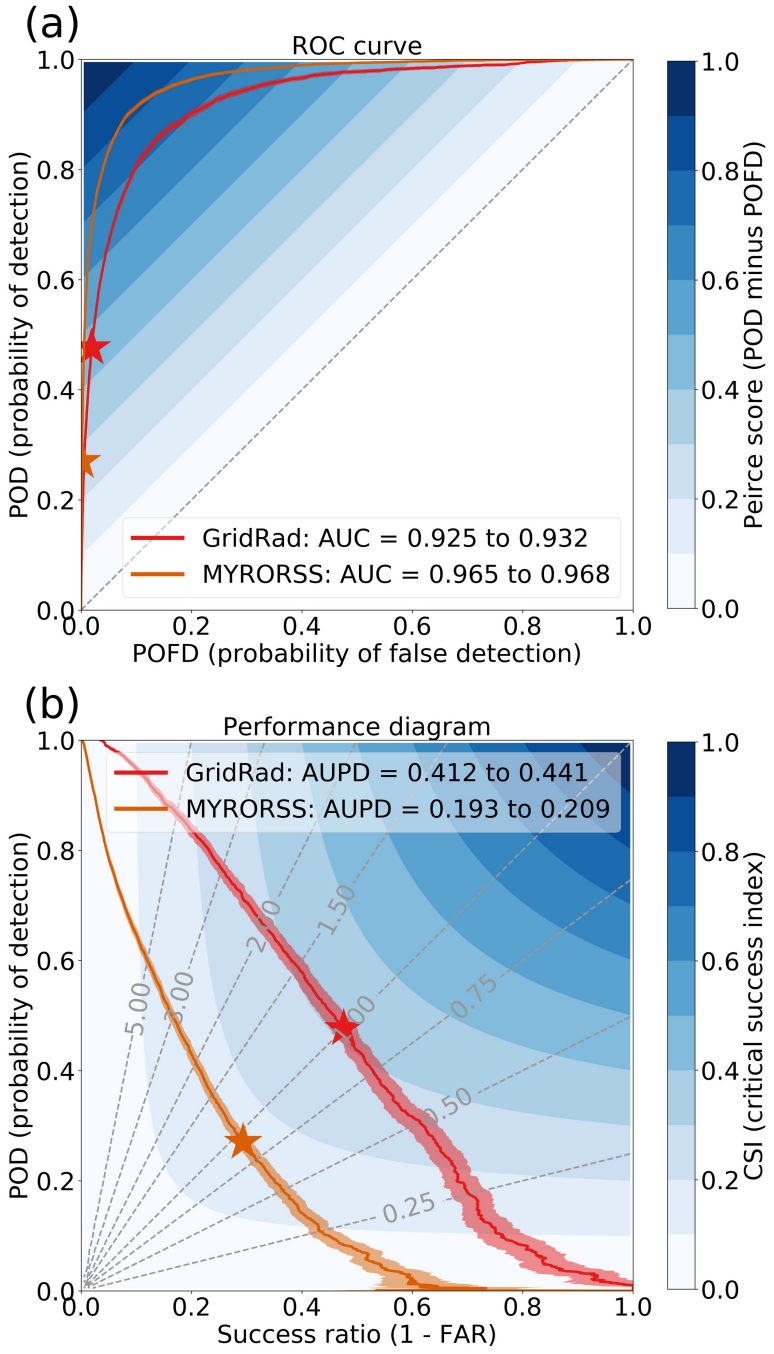


Figure 6.3: Performance of MYRORSS and GridRad models on testing data. Dark lines show the mean, and light shading shows the 95% confidence interval, determined by bootstrapping 1000 times. AUC is area under the ROC curve, and AUPD is area under the performance diagram, both given as 95% confidence intervals. The star represents the probability threshold that maximizes CSI on the validation data.

with tornado strength, which is likely caused by two factors. First, strong tornadoes are labeled more accurately, since they suffer from less underreporting bias; second, strong tornadoes generally have clearer signatures, especially in the radar image.

Figure 6.5 is the same as Figure 6.3, except that it shows performance only for matched examples in the testing data. For a MYRORSS and GridRad example (storm object) to be matched, they must occur at the same time and their centers must be within 10 km. There are 116 629 matched examples in the testing data[1], with an event frequency of 3.19%. As in the full testing set, the GridRad model has a better performance diagram than the MYRORSS model (*c.f.*, Figures 6.3b and 6.5b). However, the MYRORSS model has a substantially worse ROC curve for the matched testing set than the full testing set (*c.f.*, Figures 6.3a and 6.5a), leading to no significant difference between the two models on matched data. The main difference between the matched and full MYRORSS testing sets is that the full set contains many more non-tornadic storms, yielding a lower event frequency. Although the ROC curve is not sensitive to event frequency *per se*, most of the additional non-tornadic storms come from days without GridRad data, which are mostly days with little tornado activity in the CONUS. I speculate that non-tornadic storms on these days are easier to predict than those on outbreak days, leading to the better ROC curve for the full testing set.

As mentioned in the introduction, a ML model called ProbSevere is currently used annually in the Hazardous Weather Testbed, where it has received positive feedback from forecasters. ProbSevere forecasts the probability of any severe weather on a storm-by-storm basis. As shown in Figure 6 of Cintineo et al. (2018), ProbSevere achieves a CSI of 0.27 and POD of 0.55 with 4.94% event frequency. Figure 6.5 suggests that both the MYRORSS and GridRad models could achieve the same CSI with a lower event frequency

[1]In other words, 116 629 of 137 270 GridRad examples (85.0%) were matched to a MYRORSS example, while 116 629 of 4 608 403 MYRORSS examples (2.5%) were matched to a GridRad example. It is not surprising that so few MYRORSS examples were matched, because MYRORSS data are created for every day of the year while GridRad data are created only for selected days.

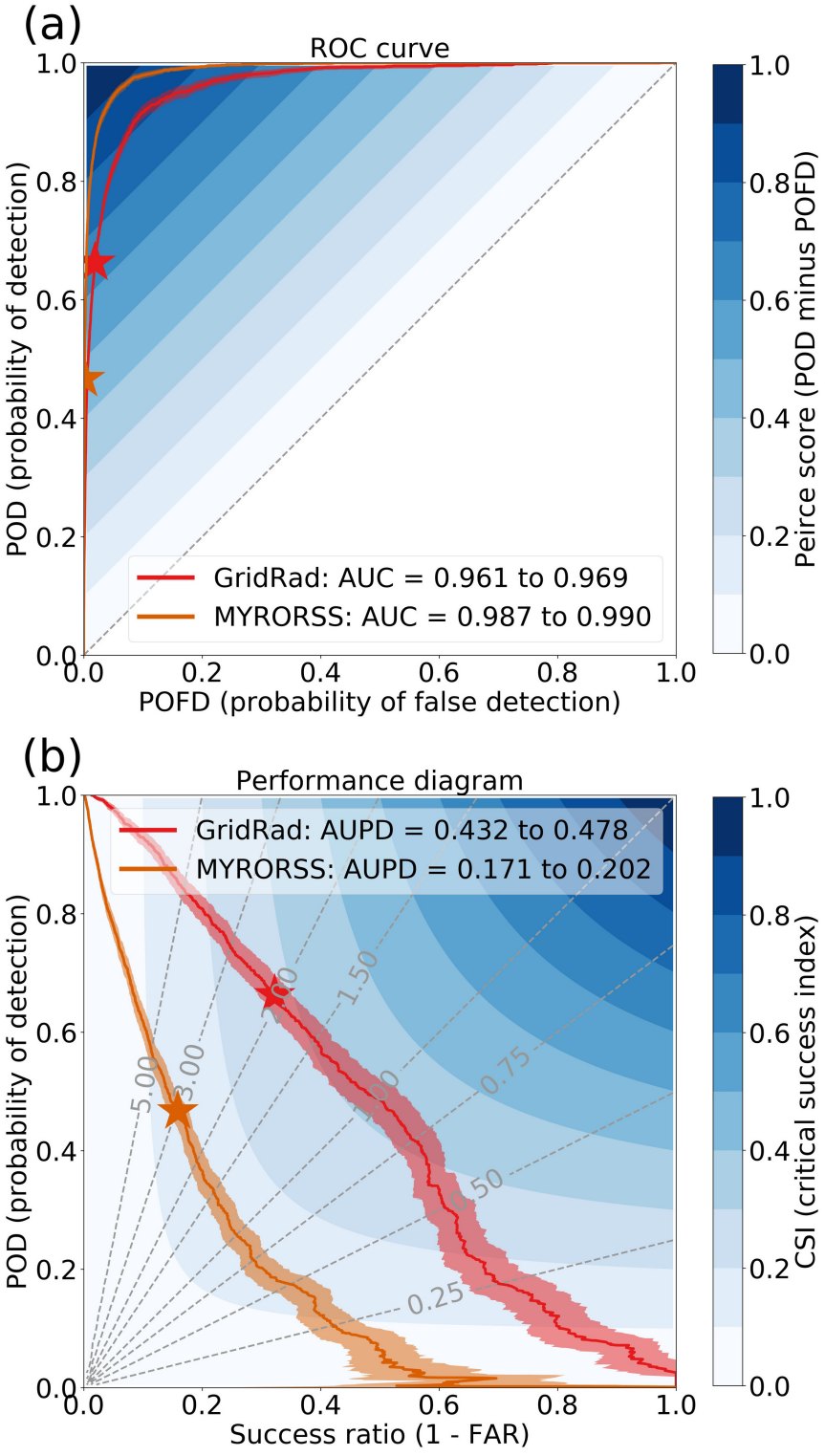


Figure 6.4: Same as Figure 6.3 but for strong tornadoes only.

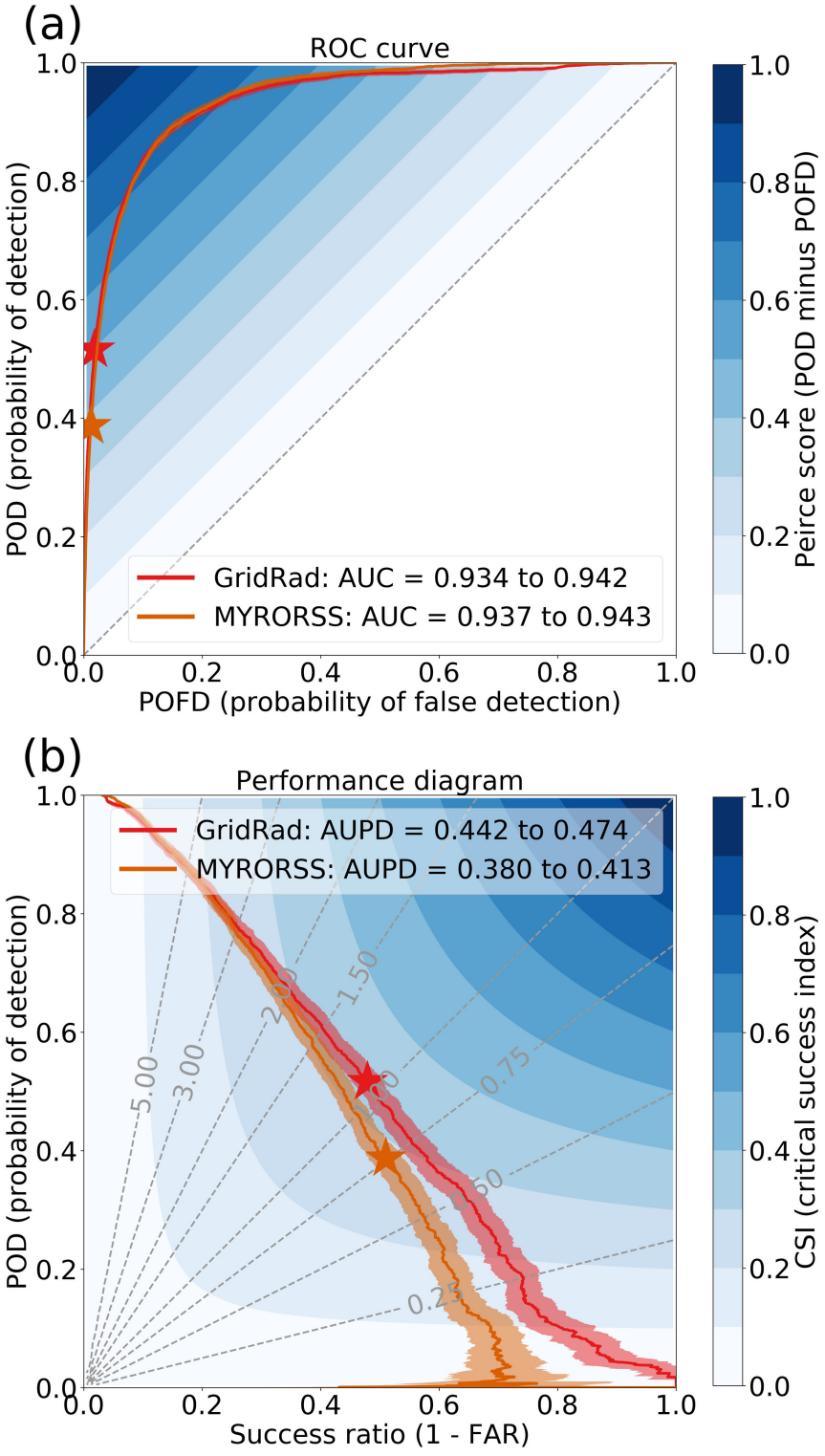


Figure 6.5: Same as Figure 6.3 but only for matched examples in testing data.

(3.19%) and higher POD ($\sim$0.7). Although the comparison is not completely fair (ProbSevere predicts all severe weather and uses MRMS, rather than MYRORSS, radar data), I believe the evidence is sufficient to suggest that my models would be useful in an operational setting.

Figures 6.6-6.7 break down the testing performance of the two models by time. As the number of tornadic examples increases, POD generally increases while FAR generally decreases, causing CSI to increase. In terms of AUC and CSI, both models perform best from April-July and during the afternoon and evening (the hours of 18-05 UTC, which end at 055959 UTC), when most tornadoes occur. In terms of AUC only, performance is excellent (> 0.9) for most hours and months. A notable exception is the GridRad model during winter (December, January, and February). These months have the fewest examples (2644, 975, and 3029 respectively), and nearly the fewest tornadic examples (50, 18, and 82 respectively), of all months in the GridRad testing data. This is also true for the training data, which means that relationships learned during training were controlled mostly by the other (non-winter) months, at the cost of performance in winter. The MYRORSS model also performs worst in winter. Lastly, both models have peaks in CSI for November and the hour of 10 UTC (100000-105959 UTC). The peak in November can be explained by a peak in tornadic examples, but the one at 10 UTC cannot. This peak is generally not significant (*i.e.*, the confidence interval for 10 UTC generally overlaps with confidence intervals for the adjacent hours), so it may be due merely to sampling error.

Figures 6.8-6.9 break down the testing performance of the two models by location, into 100-km grid cells. Figures 6.10-6.11 are analogous but use Gaussian smoothing with a 100-km *e*-folding radius. Grid cells with no tornadic examples are not shown, because this causes the scores to degenerate. Specifically, POD becomes $\frac{0}{0}$; FAR becomes $\frac{0}{0}$ or 1; CSI becomes $\frac{0}{0}$ or 0; and AUC becomes $\frac{0}{0}$. Most examples and most tornadic examples occur in the southeast quadrant of the CONUS, with a secondary maximum in tornadic examples in the southern Plains. In most areas, the GridRad model has an AUC > 0.8 (considered

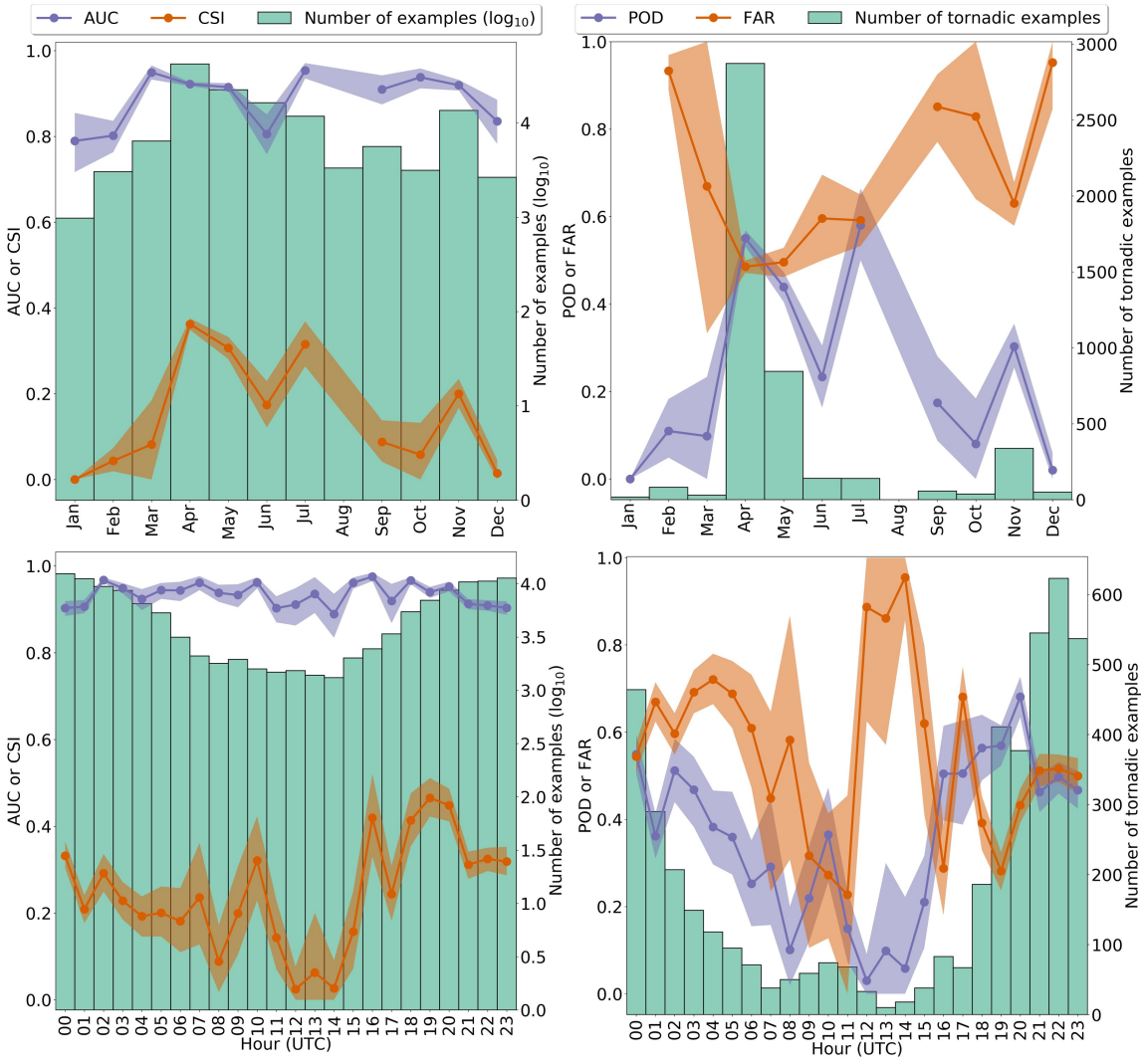


Figure 6.6: Monthly and hourly performance of the GridRad model on testing data. Dark lines show the mean, and light shading shows the 95% confidence interval, determined by bootstrapping 1000 times.

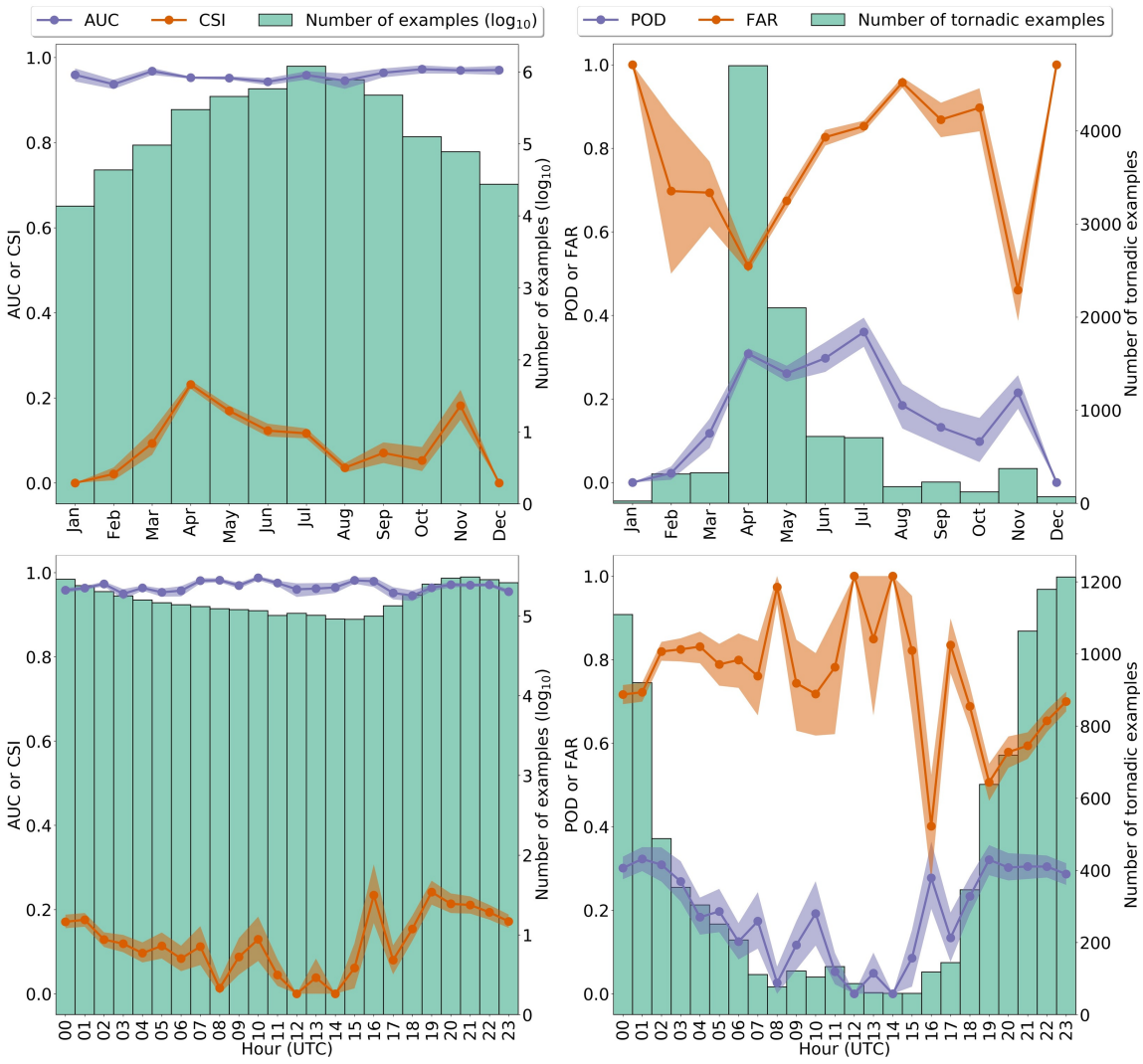


Figure 6.7: Monthly and hourly performance of the MYRORSS model on testing data. Formatting is explained in Figure 6.6.

"good" in the same papers that define 0.9 as "excellent") and the MYRORSS model has an AUC > 0.9. Areas with lower AUC generally have very few tornadic examples, thus very few examples with which to compute POD (one of the inputs to AUC), thus a large sampling error. For examples of this effect, see the three grid cells on the northern border of Kentucky in Figure 6.8. Many areas with few tornadoes in the GridRad data experience many tornadoes both climatologically[2] and in the MYRORSS data (Figure 6.9). This suggests that expanding the GridRad dataset to include tornadic storms in these areas – such as eastern Colorado, the northern Plains, and the Ohio River valley – would greatly improve performance there.

Figures 6.12-6.15 show extreme cases: the 100 best hits, worst false alarms, worst misses, and best correct nulls in the testing data (Table 6.1). These sets are constructed separately for the MYRORSS and GridRad models, with one constraint: a set cannot contain multiple examples from the same storm.[3] If the best hits or worst false alarms contain multiple time steps from one storm, only that with the highest probability is kept. Similarly, if the worst misses or best correct nulls contain multiple time steps from one storm, only that with the lowest probability is kept. Thus, each set contains examples from 100 different storm cells, which increases the diversity within each set by decreasing temporal autocorrelation. The composites shown in Figures 6.12-6.15 are created via probability-matched means (PMM; Ebert 2001).

Figure 6.12 shows the radar fields for extreme GridRad cases. For the sake of brevity, only three of the twelve heights are shown: 2, 6, and 10 km AGL. These will henceforth be called low-level, mid-level, and upper-level reflectivity. The "best hits" composite contains high reflectivity throughout the column, strong low-level convergence, and strong upper-level divergence, suggesting a strong updraft; high vorticity and spectrum width throughout

[2] `https://www.spc.noaa.gov/wcm/climo/alltorn.png`

[3] Two storm objects are considered part of the same storm if they are connected by any number of splits or mergers. For instance, all storm objects shown in "step 3" of Figure 3.7 are considered part of the same storm.

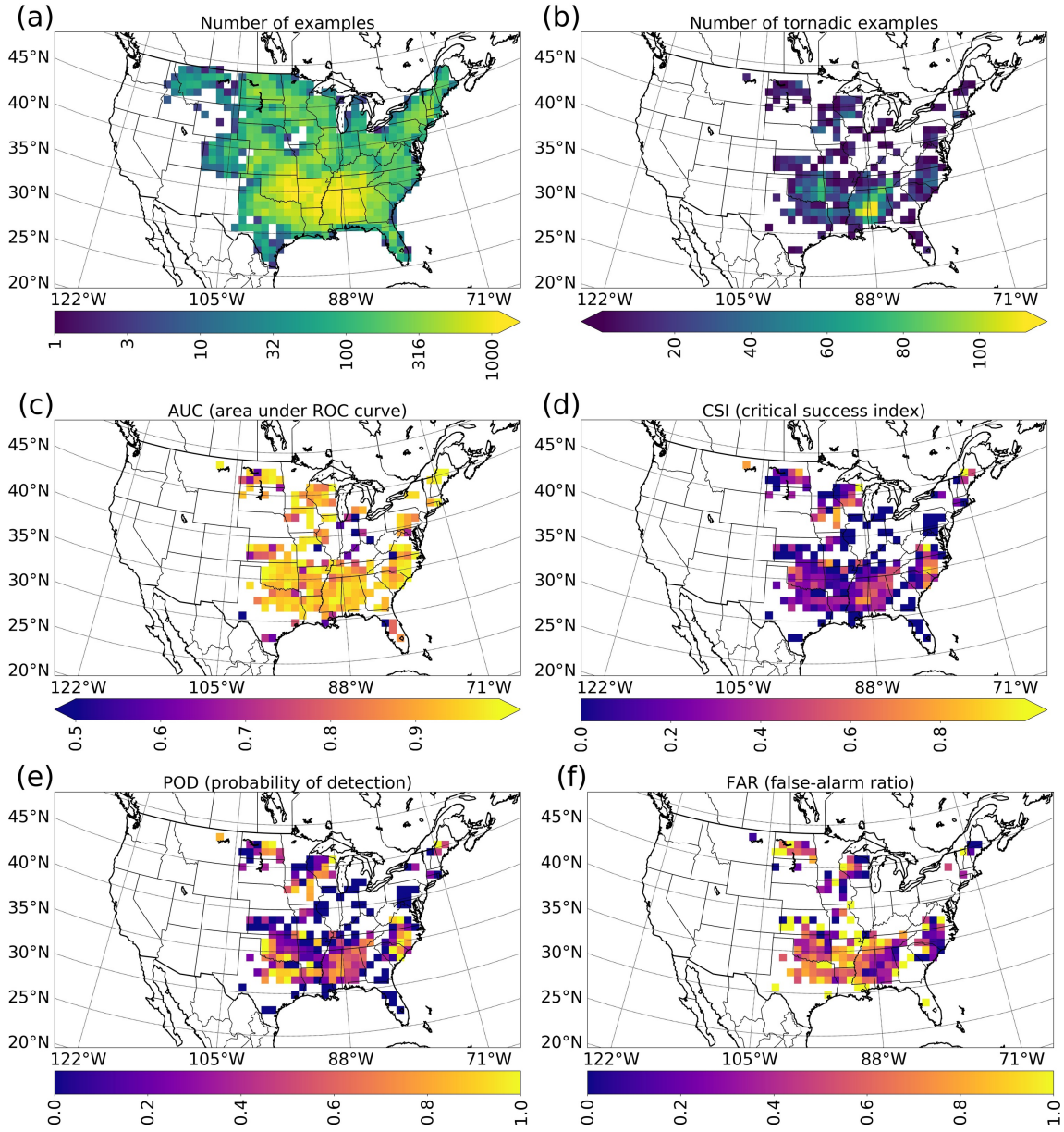


Figure 6.8: Regional performance of the GridRad model on testing data. Each grid cell is 100×100 km.

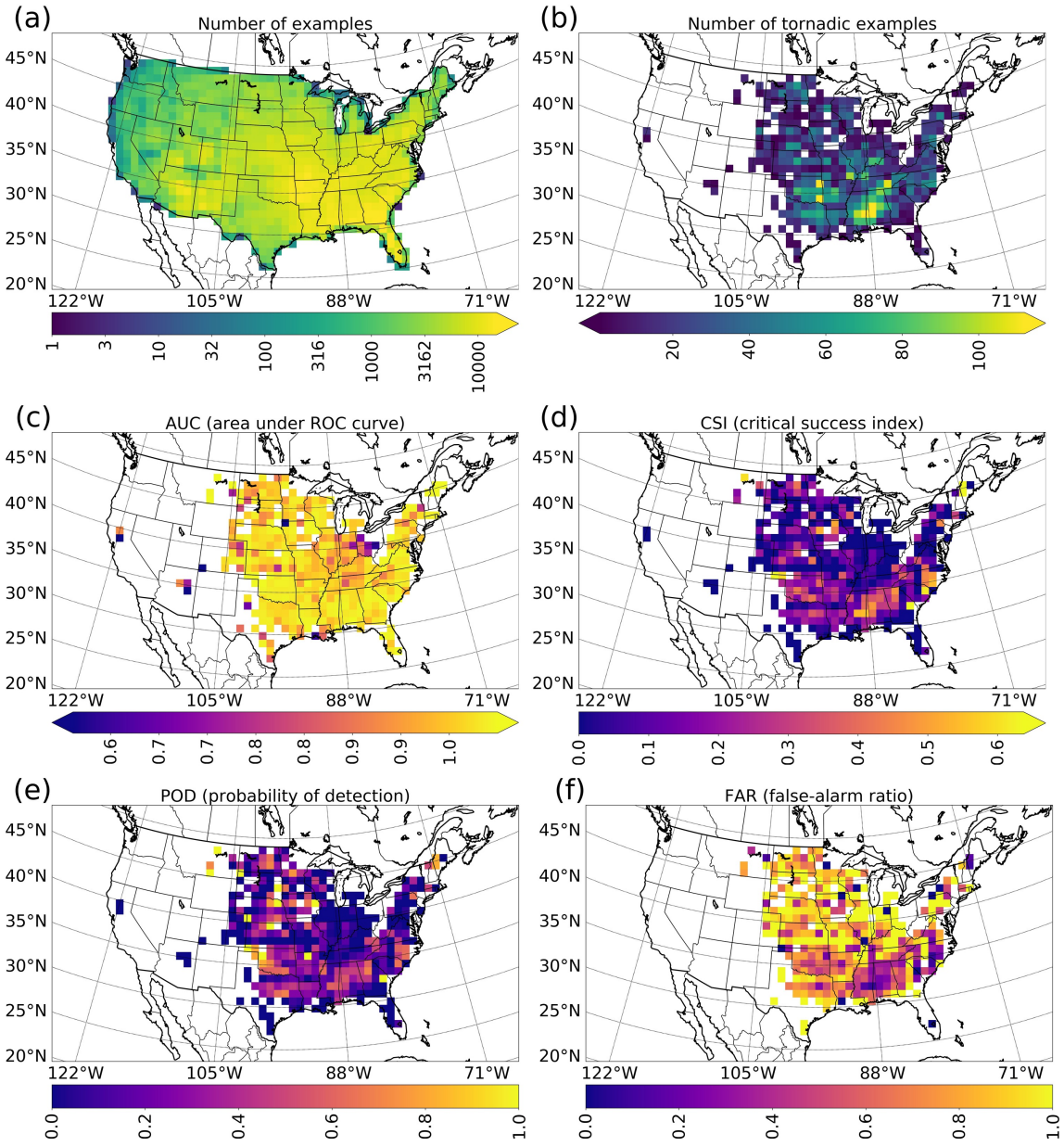


Figure 6.9: Regional performance of the MYRORSS model on testing data. Each grid cell is 100 $\times$ 100 km.

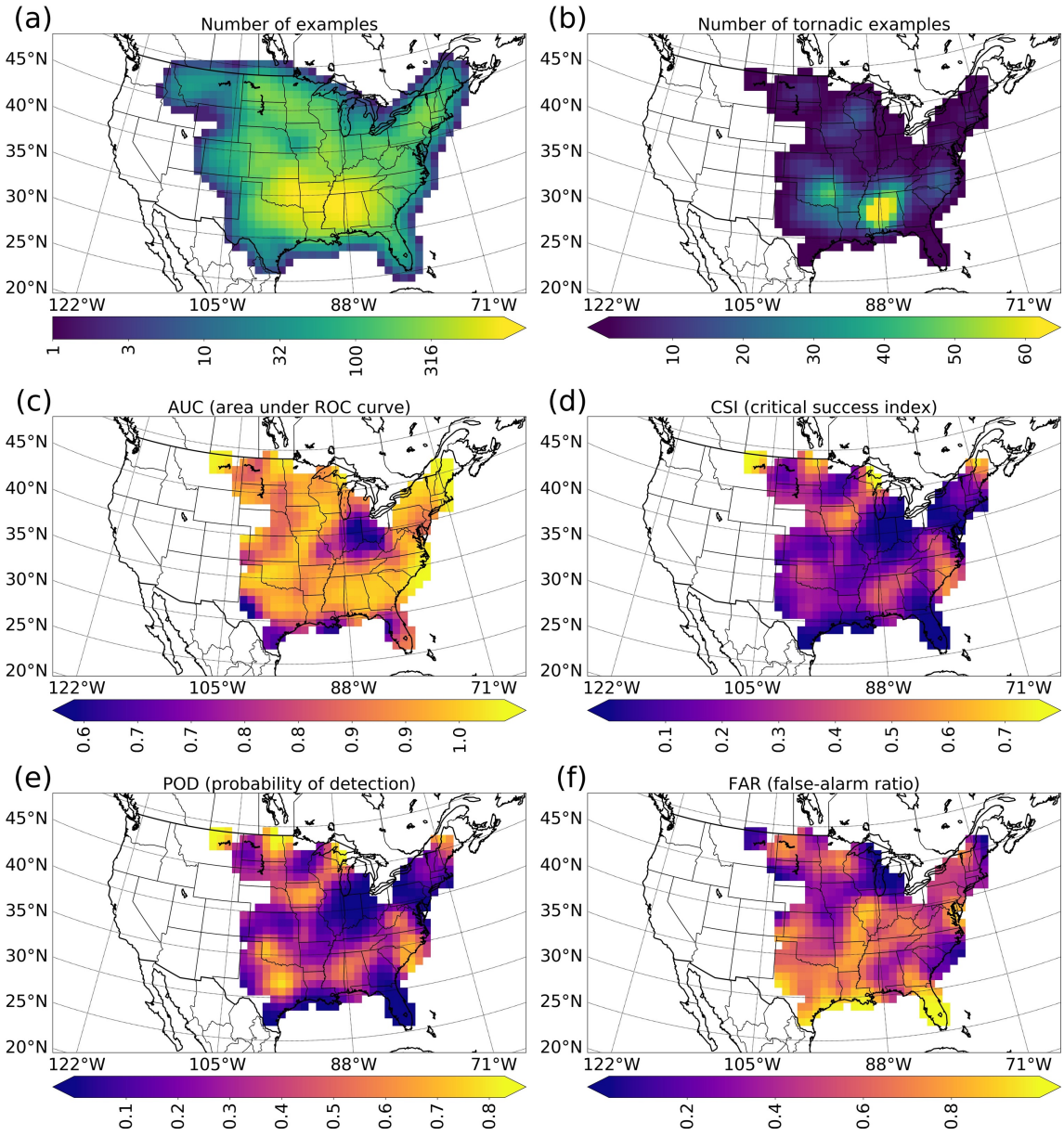


Figure 6.10: Same as Figure 6.8 but after Gaussian smoothing with a 100-km *e*-folding radius.

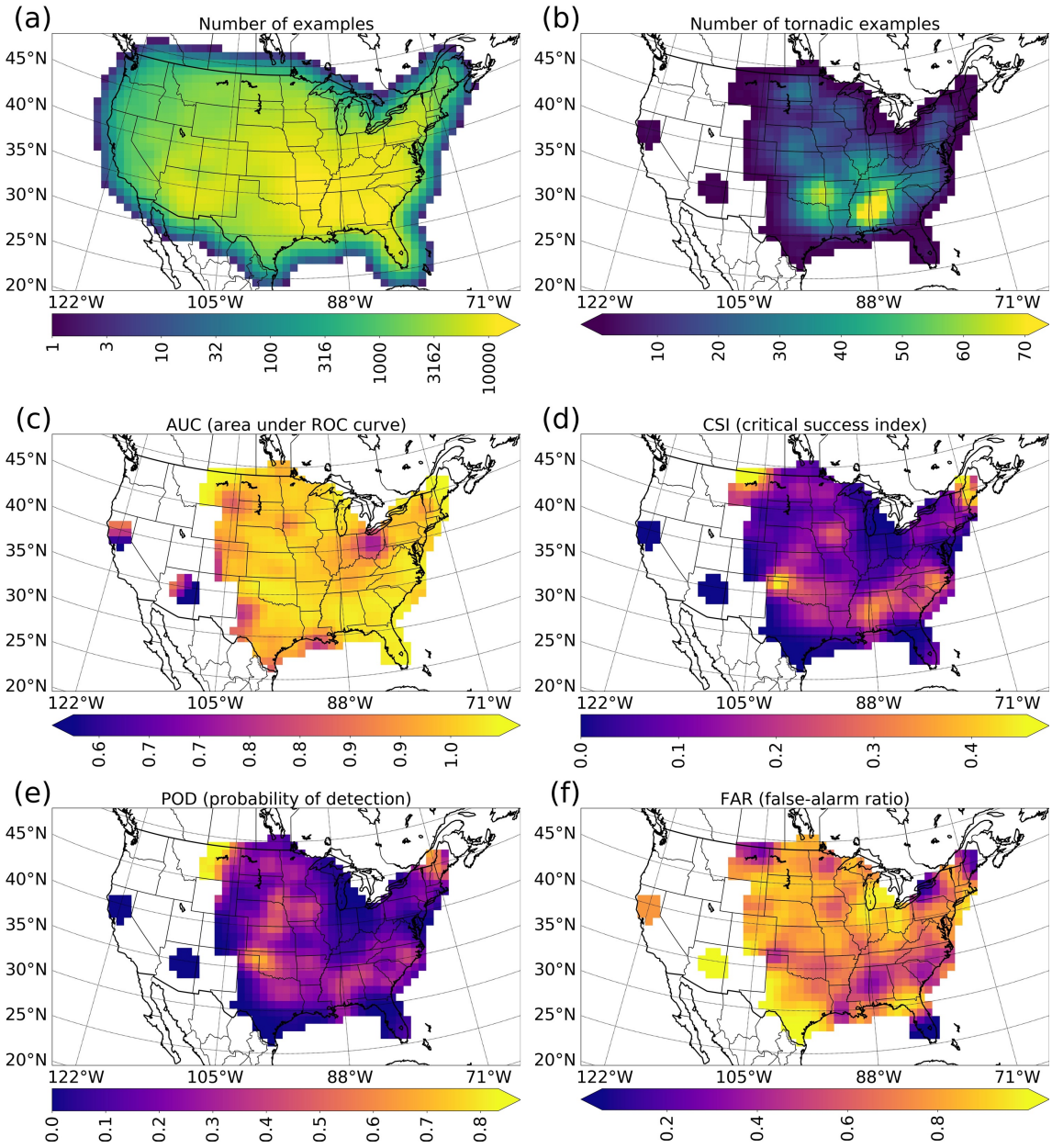


Figure 6.11: Same as Figure 6.9 but after Gaussian smoothing with a 100-km *e*-folding radius.

Table 6.1: Definitions of extreme cases. "Probability" is the next-hour tornado probability forecast by the CNN. "Mean GridRad probability" is the mean forecast probability from the GridRad model over the 100 cases, and likewise for MYRORSS.

Set	**Definition**	**Mean GridRad probability**	**Mean MYRORSS probability**
Best hits	Tornadic examples with **highest** probabilities	99.2%	99.6%
Worst false alarms	**Non**-tornadic examples with **highest** probabilities	98.8%	99.6%
Worst misses	Tornadic examples with **lowest** probabilities	8.6%	11.9%
Best correct nulls	**Non**-tornadic examples with **lowest** probabilities	0.004%	0.04%

the column, suggesting a strong mesocyclone; and a hook echo in the low-level reflectivity field, which suggests the presence of a rear-flank downdraft (RFD) and is a signature often associated with tornadoes (Rhyzkov et al. 2005). Overall, the storm appears to be a supercell, the type responsible for most tornadoes and especially EF-2+ tornadoes (Table 2 of Smith et al. 2012). Also, the storm appears to be discrete (isolated from other storms), at least on its right and rear flanks. Although weak tornadoes are often produced by supercells in clusters, strong tornadoes are generally produced by discrete supercells (Table 2 of Smith et al. 2012).

The worst false alarms look very similar to the best hits, with one of the few notable differences being the absence of a low-level hook echo. I considered the possibility that these storms are not really false alarms – *i.e.*, produced unreported tornadoes – which is a known issue with the NWS storm reports, as discussed in Section 3.1.1. However, this explanation is implausible, because (a) most of these storms occur in the evening and near towns; (b) 76 of the 100 storms are associated with an NWS tornado warning, and the NWS generally seeks out reports to verify warnings that they have issued. Rather, I believe that the similarity between best hits and worst false alarms is due mainly to two factors. First, tornadoes are usually on the order of 100 metres across, so GridRad and MYRORSS have insufficient resolution to represent tornadoes and other relevant circulations. Second, "tornado" and "non-tornado" are discrete labels applied to a continuous spectrum of phenomena. Some funnel clouds very nearly reach the surface, while some tornadoes touch down for only a few seconds and produce minimal damage, but they are still labeled "tornado" and "non-tornado" respectively, which causes them to be treated as completely disparate phenomena. As indicated by the fact that 76% of the storms are associated with NWS warnings, this dichotomous labeling seems to cause problems for both humans and machines.

The worst misses are shallow and elongated storms with smaller values of reflectivity, spectrum width, vorticity, low-level convergence, and upper-level divergence. By inspection of the 100 individual storms, I have found that most are either (a) weak storms that subsequently intensify rapidly, allowing them to produce tornadoes in the next hour, or (b) embedded in a quasi-linear convective system (QLCS). The second failure mode is more common than the first and is well known to meteorologists (see discussion in Section 2.1.1). Lastly, the best correct nulls are generally weak and disorganized. By inspection of the 100 individual storms, I have found that most are non-dominant cells in a mesoscale convective system (MCS). Also, 19 of the 100 storms occur in the outer rain bands of Hurricane Irene.

Figure 6.13 shows the radar fields for extreme MYRORSS cases. Overall, each composite is very similar to its counterpart in the GridRad data, except that the best hits for MYRORSS contain a much fainter hook echo. Conclusions based on inspecting individual storms also hold for the MYRORSS data. First, false alarms generally occur in the evening and near towns, and 47 of the 100 are associated with NWS tornado warnings, which suggests that most of these storms are truly non-tornadic. Second, the worst misses are mostly QLCS cells, with some early-stage supercells. Third, the best correct nulls are mostly non-dominant MCS cells.

Figure 6.14 shows the soundings for extreme GridRad cases. For all composites other than the best correct nulls, the soundings are much more similar than the radar images. This suggests that the radar images are more important predictors, a claim investigated further via the interpretation methods in Chapter 7. All three soundings have a slight surface-based temperature inversion, above which the profile is conditionally unstable up to the high troposphere; a dry layer around 700 mb; and a strongly veering wind profile, with easterly surface winds veering sharply to southerly in the lowest 100 mb, then to westerly in the high troposphere. This corresponds to high wind shear, which is crucially favourable for tornadoes (Markowski and Richardson 2009, 2014; Anderson-Frey et al. 2016). The main

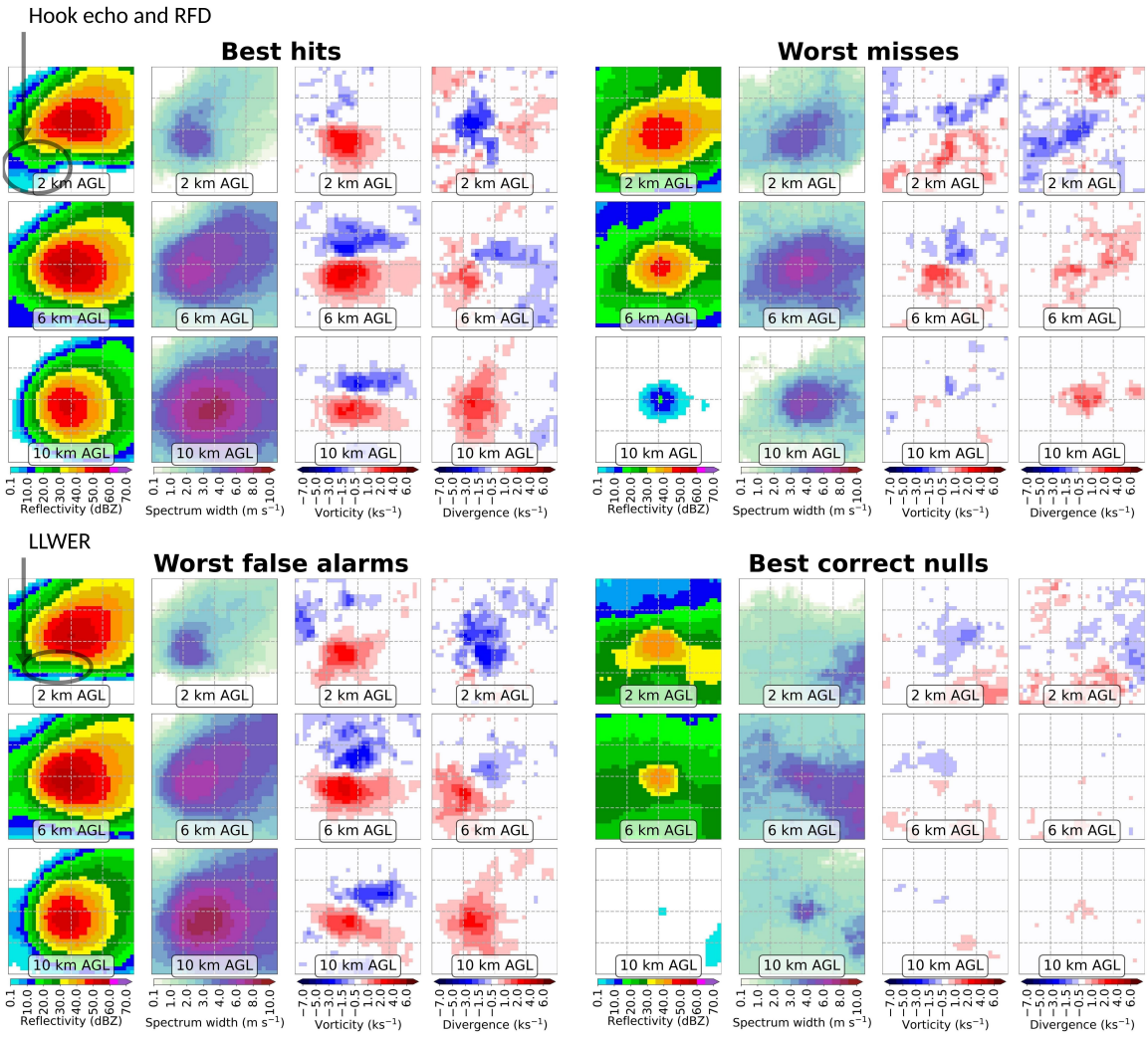


Figure 6.12: Radar fields for extreme GridRad cases. Storm motion points to the right. Each image is a PMM composite over 100 examples. The hook echo and rear-flank downdraft (RFD) are labeled on the best hits, similar to schematic in Figure 13 of Markowski (2002). The LLWER (low-level weak-echo region) is labeled on the worst false alarms (it is present for the best hits as well, but there is no room for another label).

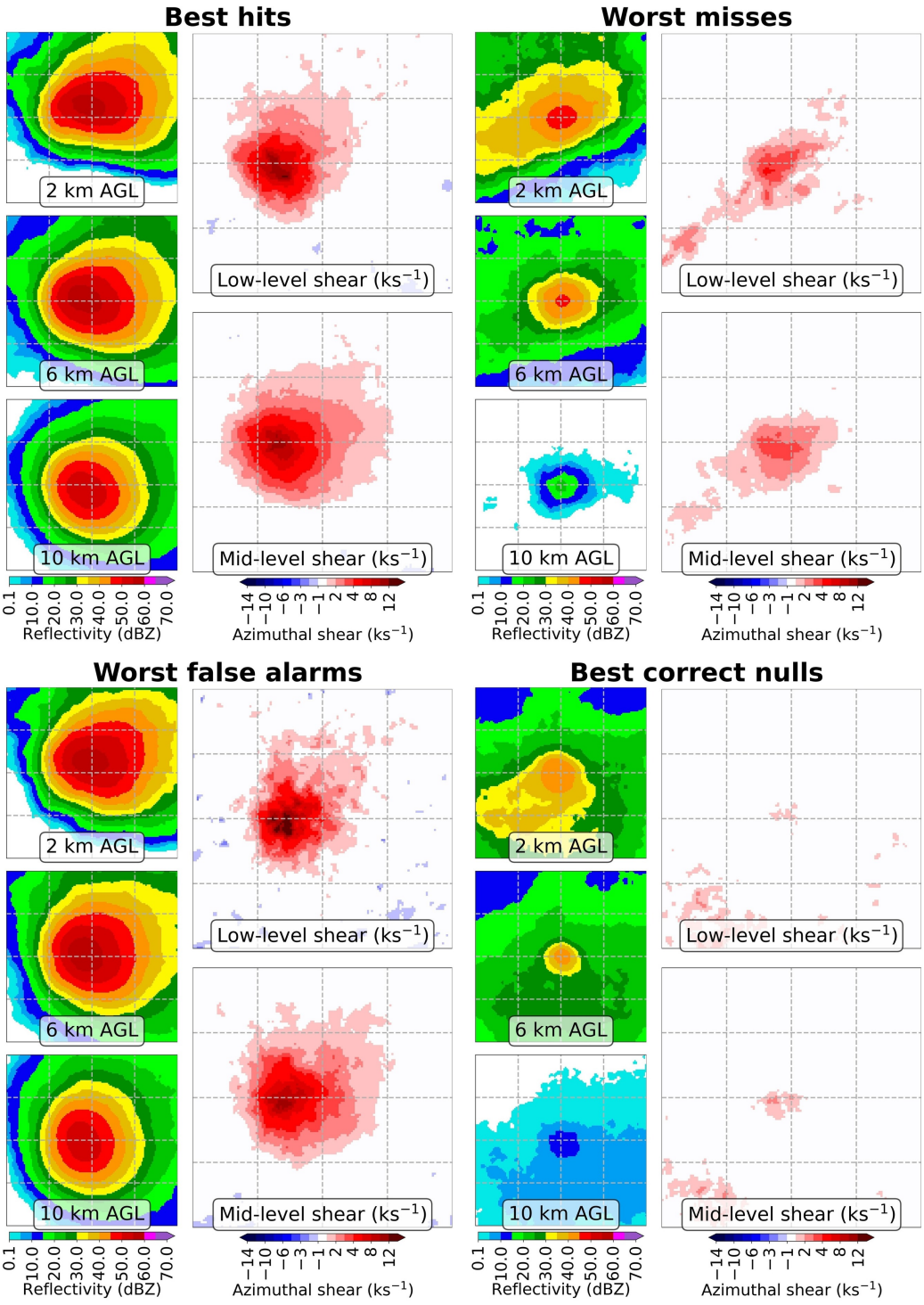


Figure 6.13: Radar fields for extreme MYRORSS cases. Storm motion points to the right. Each image is a PMM composite over 100 examples.

difference among the three soundings is that the worst misses have lower near-surface temperature and humidity, yielding less instability, and weaker winds aloft, yielding less wind shear. Compared to the other three composites, the best correct nulls have a stronger temperature inversion and much less low-level wind shear. By inspection of the 100 individual storms, many occur near fronts, causing the wind shift between 700 and 800 mb in the composite sounding.

Figure 6.15 shows the soundings for extreme MYRORSS cases. Each composite except the best correct nulls is very similar to its counterpart in the GridRad data (*c.f.*, Figures 6.14 and 6.15). For the best correct nulls, the MYRORSS sounding has a much stronger temperature inversion, and much less near-surface moisture, than the GridRad sounding. The difference occurs because the MYRORSS dataset contains many more non-tornadic storms than GridRad, allowing for more extreme non-tornadic cases.

6.2 Front Detection

This section evaluates the performance of the front-detection model, both before and after conversion of probabilities to frontal zones. Although previous studies have used CNNs for front detection (Liu et al. 2016; Kunkel et al. 2018), I do not compare my performance with theirs. Liu et al. (2016) reported only one evaluation score (accuracy) on a nearly balanced dataset (5600 frontal patches and 6500 non-frontal patches), which does not represent the spatial rarity of the event. Meanwhile, the evaluation methods of Kunkel et al. (2018), which is a conference presentation, are unclear. An experiment shown in previous work (Figure 13 of L19), using a CNN similar to those developed herein, suggested that the CNN strongly outperforms numerical frontal analysis.

6.2.1 Best CNN

Figure 6.16 shows the multiclass AUC of all 288 models on the validation data. The models generally perform best with predictors at two vertical levels, which supports the hypothesis

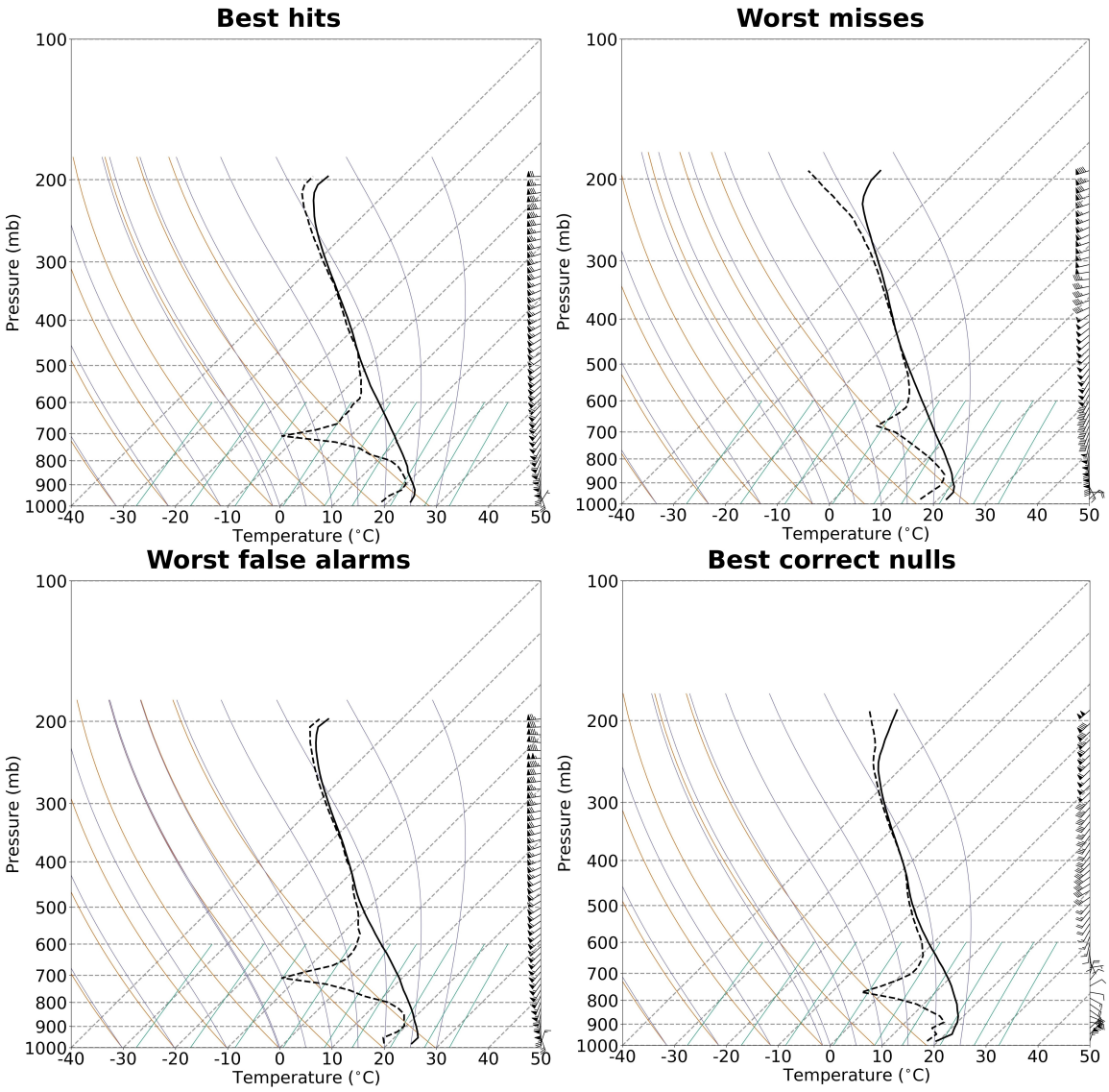


Figure 6.14: Soundings for extreme GridRad cases. Each sounding is a PMM composite over 100 examples.

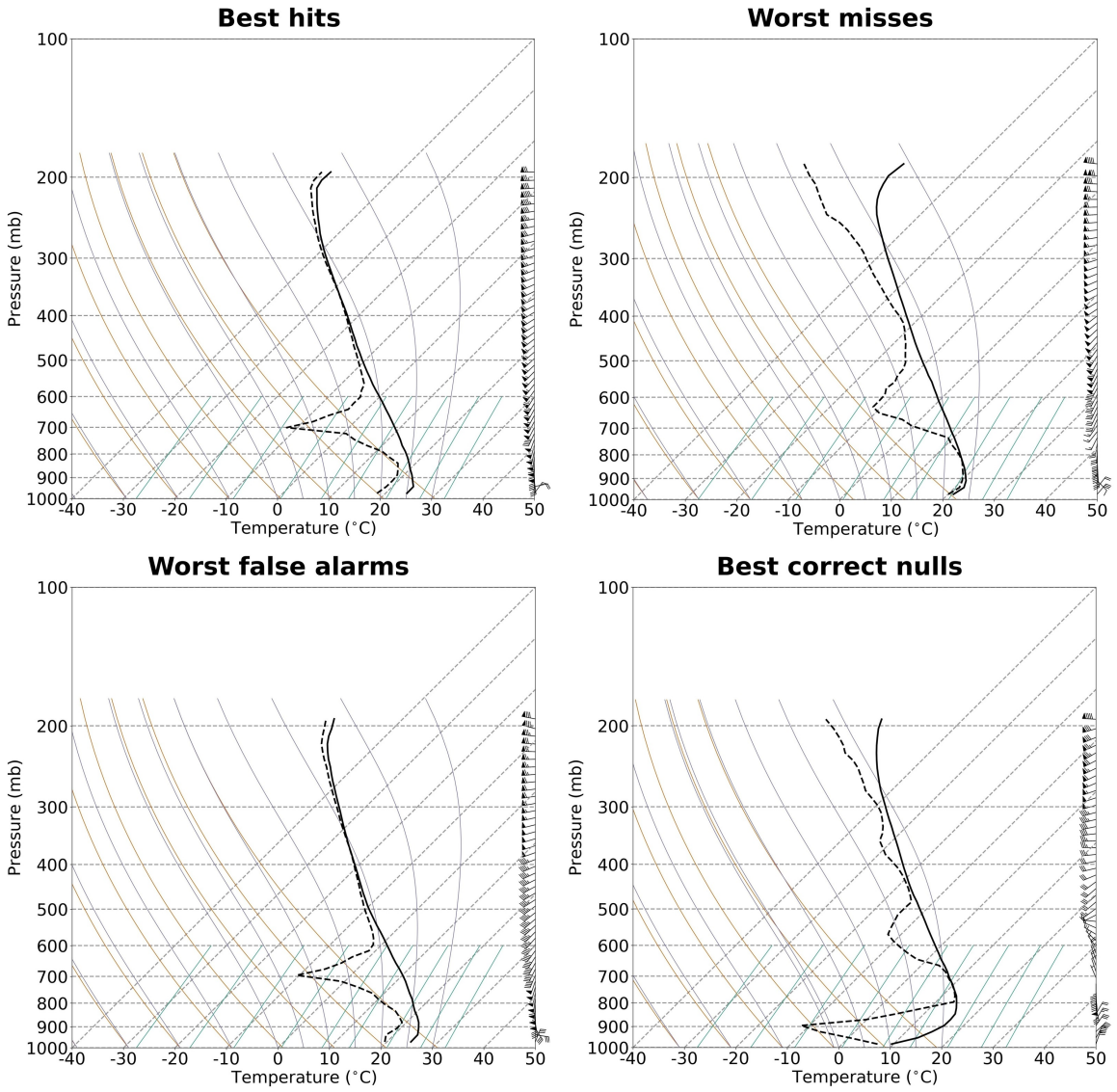


Figure 6.15: Soundings for extreme MYRORSS cases. Each sounding is a PMM composite over 100 examples.

from Section 5.2.1. There is no apparent trend in AUC with respect to the other hyperparameters: number of convolutional blocks, number of layers per block, or set of predictor variables. The last result may occur because the set always contains enough information (wind and at least one thermal variable) to identify most fronts. The best model (black star in Figure 6.16) has three convolutional blocks, one layer per block, and is trained on all variables except geopotential height at the surface and 900 mb. Ideally, I would have selected this model for the climatology, but I encountered problems in downloading data for the full climatological period (1979-2018) at levels other than the surface and 850 mb. Thus, I have selected the best model trained on surface and 850-mb data (black circle in Figure 6.16). The difference in AUC between the two models (0.829 versus 0.828) is not significant at the 95% confidence level, determined by bootstrapping 1000 times. The ROC curve for the selected model, also based on validation data, is shown in Figure 6.17.

Figures 6.18-6.21 show extreme cases for the selected model: the 1000 best hits for warm fronts, worst misses for warm fronts, best hits for cold fronts, worst misses for cold fronts, and best correct nulls. These sets are defined in Table 6.2 and based on the testing data. Figure 6.18 shows the spatial distribution of extreme cases (each dot corresponds to the center grid cell of one patch, which is the grid cell labeled by the CNN), and Figures 6.19-6.21 show composites created by probability-matched means (PMM). The worst false positives (*i.e.*, non-fronts mistaken for fronts) are not shown, because they look very similar to the best hits. Specifically, the worst false alarms for warm fronts look like the best hits for warm fronts, and likewise for cold fronts. This suggests that WPC meteorologists often miss fronts, which is consistent with my visual analysis (*e.g.*, in Figure 5.6, the WPC appears to have missed a cold front south of Nova Scotia and two fronts in the Canadian Arctic archipelago). It also suggests that to some extent the CNN can overcome this deficiency of the input data.

The best hits for warm fronts (Figure 6.19a) feature a strong cyclone, with the front wrapping around the east-northeast side of the cyclone. The front is visible in all four

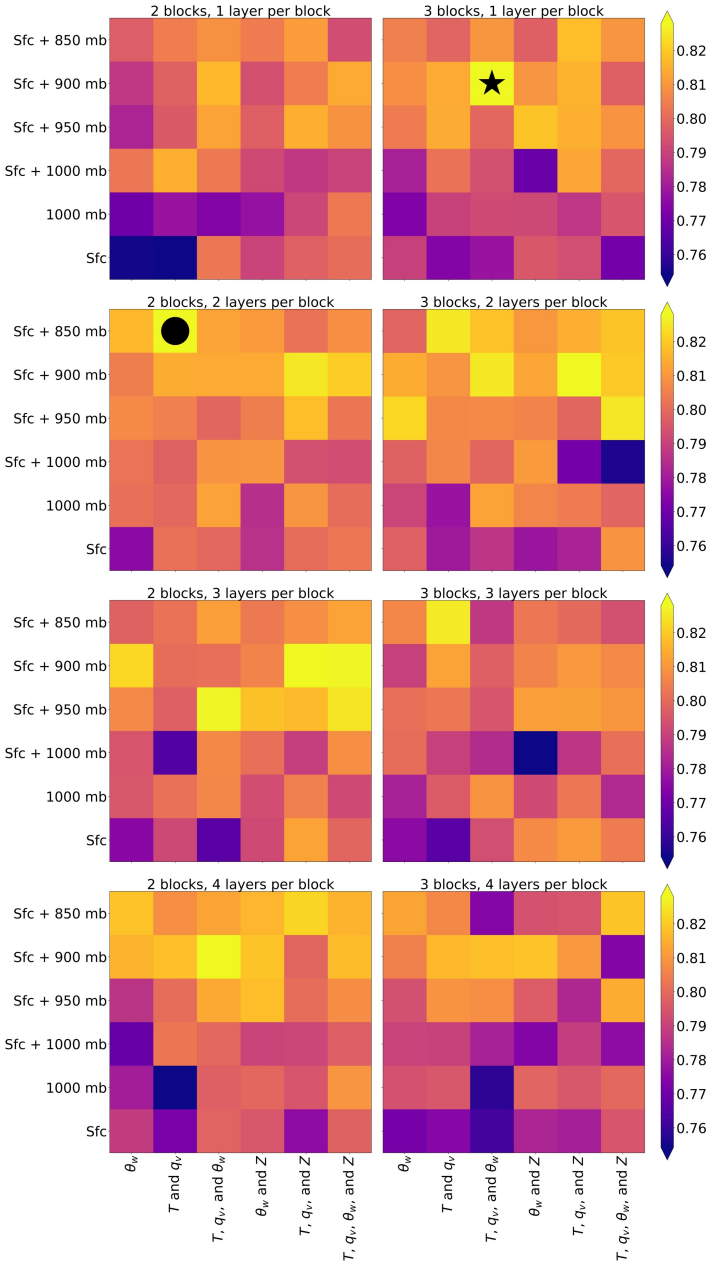


Figure 6.16: AUC for front-detection models on validation data. The *x*-axis shows predictor variables (in addition to *u*- and *v*-wind, which are always included), and the *y*-axis shows vertical levels. The best model (with highest validation AUC) is indicated by a black star; the selected model is indicated by a black circle.

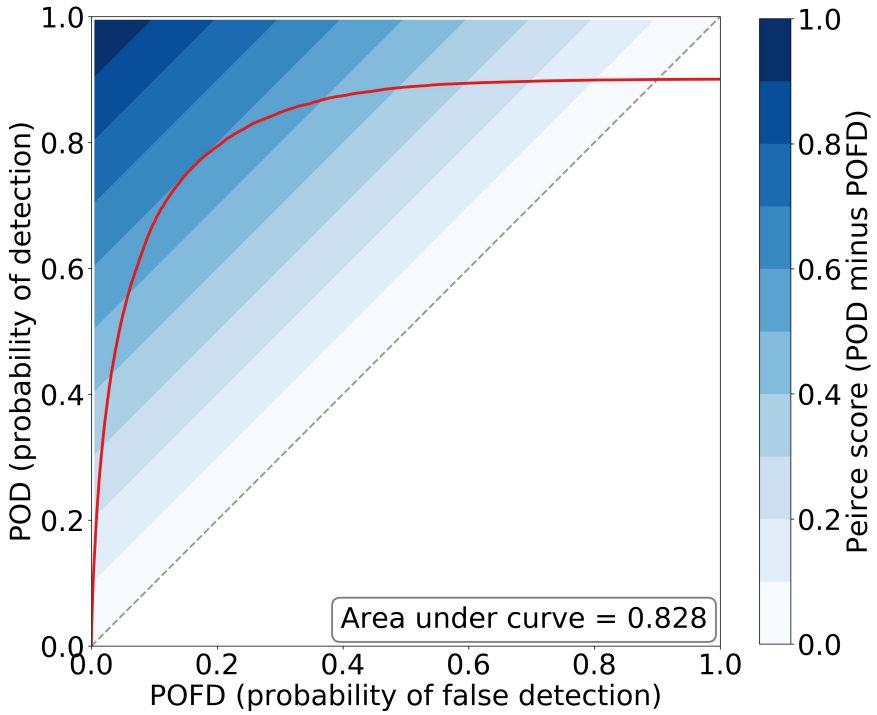


Figure 6.17: ROC curve on validation data for selected front-detection model. POD and POFD in this setting are defined in Equation 5.3.

Table 6.2: Definitions of extreme cases. p_W, p_C, and p_N are CNN-estimated probabilities of warm front, cold front, and no front, respectively.

Set	**Definition**	**Mean Probability**
Best hits for warm fronts	Warm-front patches with highest p_W	Mean p_W = 99.7%
Worst misses for warm fronts	Warm-front patches with highest p_N	Mean p_N = 99.0%
Best hits for cold fronts	Cold-front patches with highest p_C	Mean p_C = 99.7%
Worst misses for cold fronts	Cold-front patches with highest p_N	Mean p_N = 99.6%
Best correct nulls	Non-front patches with highest p_N	Mean p_N = 100.0%

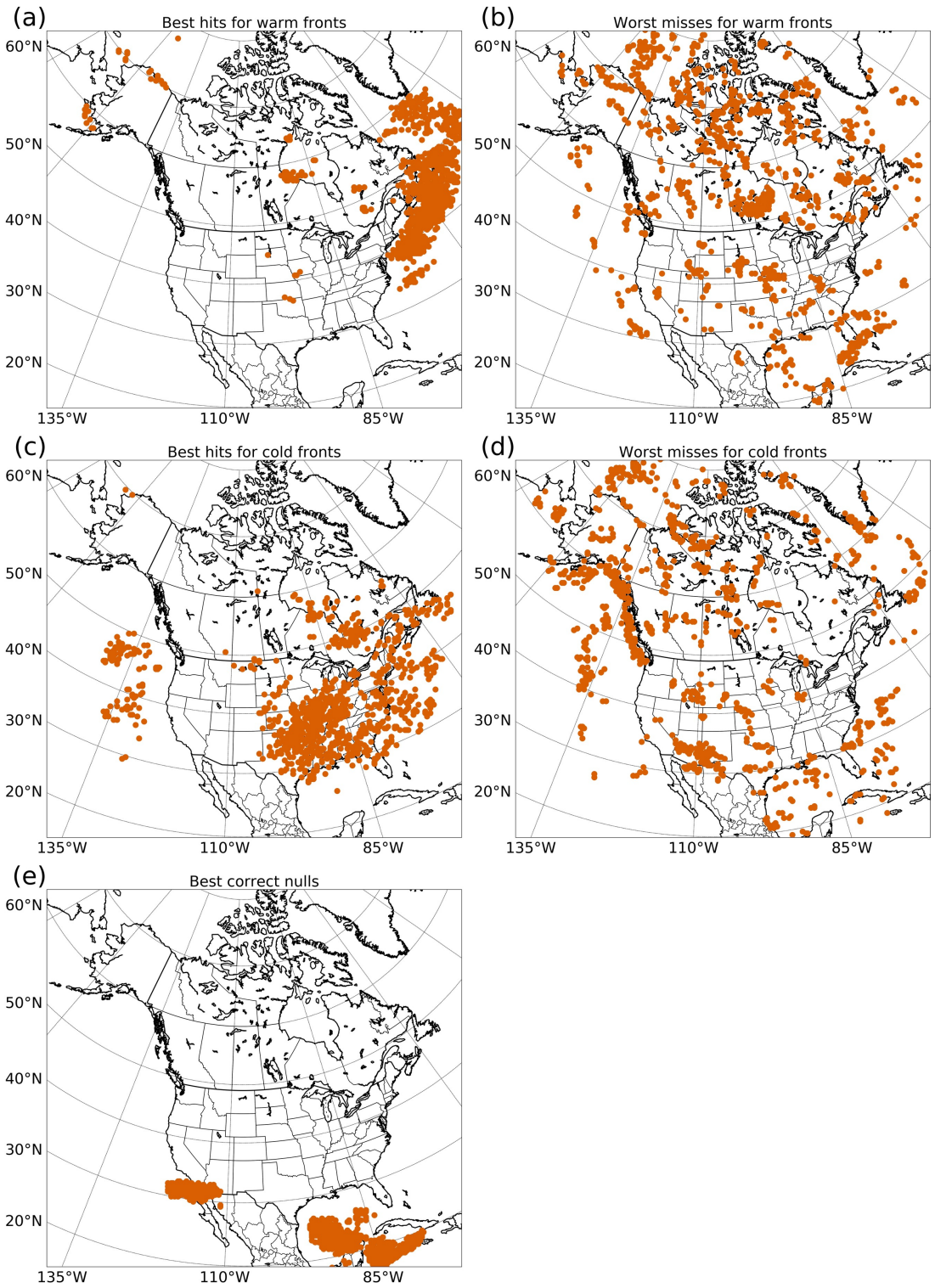


Figure 6.18: Spatial distribution of extreme cases for selected model. Each dot corresponds to the center grid cell of one patch, which is the grid cell labeled by the CNN. Each map contains 1000 dots.

thermal fields (surface and 850-mb temperature and humidity), and the wind shifts abruptly from southerly behind the front to easterly ahead of the front. The southerly wind behind the front is nearly antiparallel to the thermal gradients, causing strong warm and moist advection. Most of these fronts occur in the western Atlantic (Figure 6.18a), where strong warm fronts often propagate after developing over the warm Gulf Stream.

The worst misses for warm fronts (Figure 6.19b) feature weak anticyclonic flow at the surface, a deformation zone at 850 mb, and a general southwest-pointing thermal gradient. Although a strong temperature gradient crosses the center of the domain, there is very little warm advection here, due to the weak wind field. Compared to the best hits (Figure 6.18a), the worst misses (Figure 6.18b) are much less spatially concentrated. However, many cases occur near a land-sea boundary, especially in the Canadian Arctic. By inspecting the individual cases, I have identified three reasons that the CNN misses these fronts. First, many cases occur in complex topography, where the synoptic pattern is difficult to disentangle from smaller-scale patterns created by topography, which makes synoptic fronts more difficult to define. Second, there is often little to no thermal advection near the center grid cell, especially at the surface. Some of the cases have stronger advection at 850 mb, because the warm front is overriding stable cold air at the surface. Third, most fronts – especially strong fronts – are associated with cyclones rather than anticyclones and deformation zones, so cyclonic examples likely dominated the training procedure.

The best hits for cold fronts (Figure 6.20a) feature cyclonic flow throughout most of the domain, with the front oriented from northeast to southwest. Flow is stronger and more curved in the north, which suggests that the parent cyclone is north of the domain. If so, the cold front extends a long distance from its parent cyclone, since the domain is 1056 $\times$ 1056 km. Most of these fronts occur in the eastern CONUS and western Atlantic (Figure 6.18c), where mature cyclones with long cold fronts are common.

The worst misses for cold fronts (Figure 6.20b) feature an anticyclone with the apparent front wrapping around its southwestern edge, oriented from northwest to southeast. The

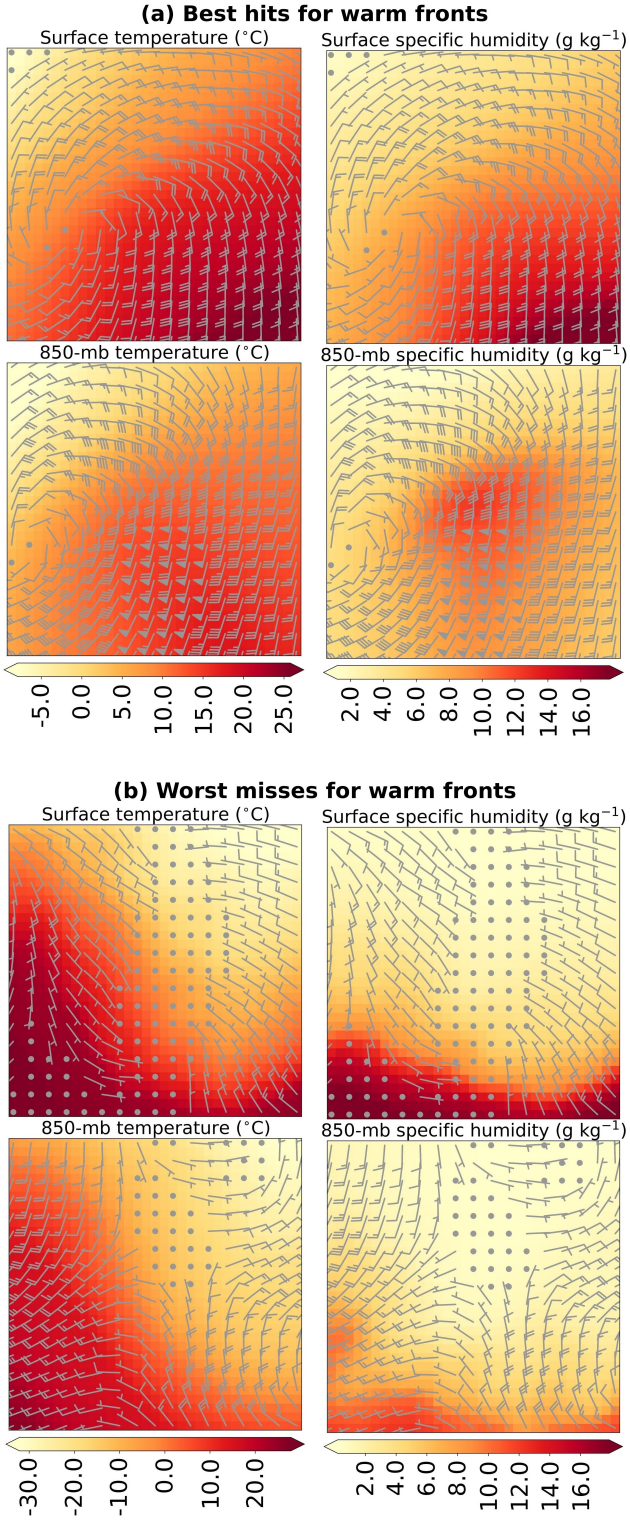


Figure 6.19: Extreme warm-frontal cases for selected model. Each map is a PMM composite over 1000 cases, and each case contains an actual (WPC) warm front passing through the center grid cell.

pattern looks similar to Arctic-outflow events in western Canada (West and Howard 2019), where Arctic air penetrates through the Rocky Mountains and often through the coastal mountains as well, reaching the Pacific. Based on individual inspection, this process is responsible for many cases along the west of British Columbia (Figure 6.18d). A similar process is also responsible for some cases in the southwestern CONUS (Figure 6.18d), where cold air penetrates through the mountains after reaching the southern Plains. The CNN misses cold fronts in Figure 6.20b for the same reasons that it misses warm fronts in Figure 6.19b, with the exception of warm fronts overriding cold surface air.

The best correct nulls (Figure 6.21) feature anticyclonic flow through most of the domain and nearly uniform surface temperatures, except for a cold tongue in the west and warm tongue in the east. These tongues are caused by land-sea boundaries, specifically near Baja California and the Yucatán peninsula (Figure 6.18e). The nearly homogeneous temperatures in between are associated with subtropical high pressure, which is a semi-permanent feature of the eastern Pacific and transient feature of the Caribbean and Gulf of Mexico. The CNN easily labels these examples as non-frontal, because despite strong thermal gradients appearing in the domain, they are far removed from the center grid cell.

6.2.2 Best Object-conversion Method

Figure 6.22 shows the CSI (Equations 5.5) on validation data for all pairs of probability thresholds. Although CSI appears to be maximized with higher thresholds than those attempted, higher thresholds lead to narrow, discontinuous frontal zones and very few warm fronts.[4] However, CSI punishes misses and false alarms equally, without accounting for the fact that the WPC appears to miss many fronts, so many "false alarms" produced by the CNN are not truly false alarms. To account for this, I have plotted weighted CSI (Figure 6.23), where FAR is multiplied by 0.5 in the last of Equations 5.5, so that false alarms

[4]The CNN generally produces smaller warm-front probabilities, because warm fronts are more diffuse than cold fronts and have smaller thermal gradients.

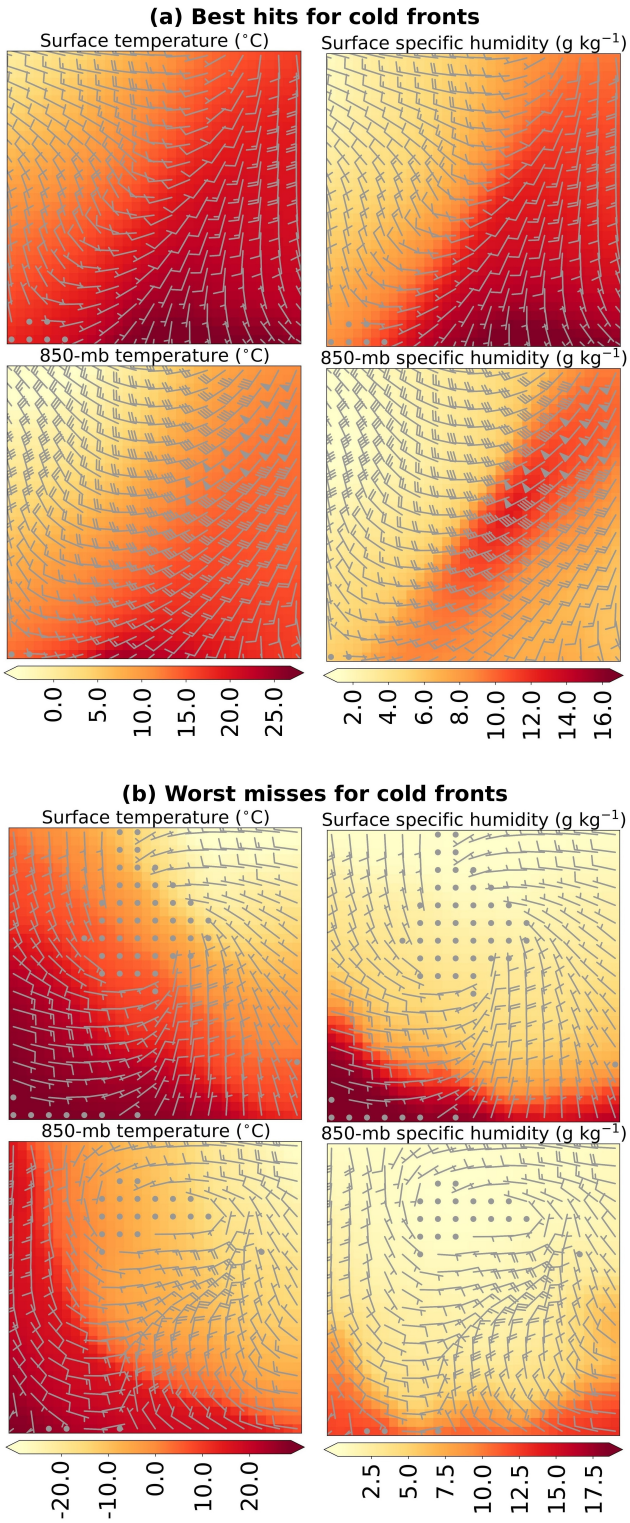


Figure 6.20: Extreme cold-frontal cases for selected model. Each map is a PMM composite over 1000 cases, and each case contains an actual (WPC) cold front passing through the center grid cell.

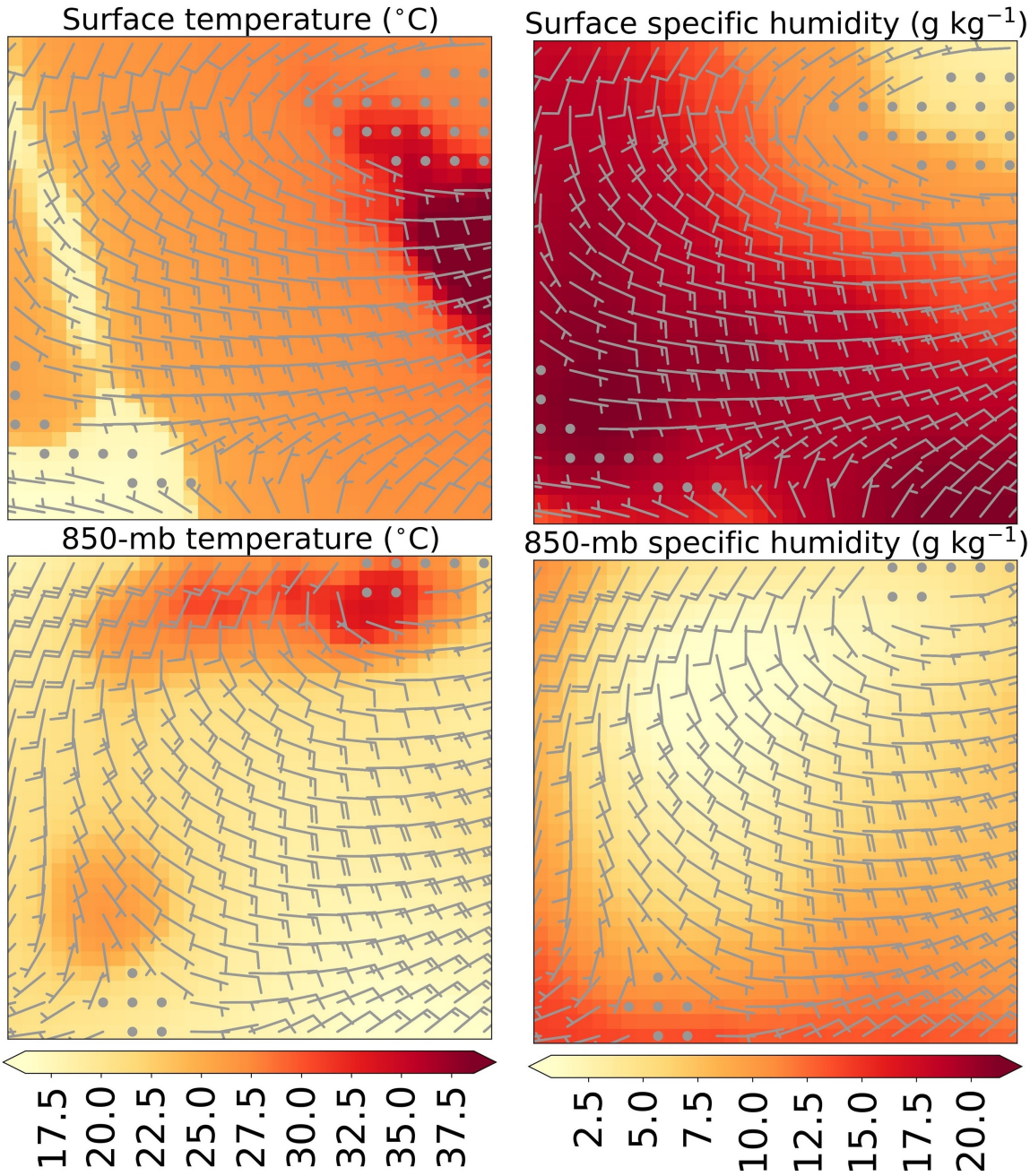


Figure 6.21: Best correct nulls for selected model. Each map is a PMM composite over 1000 cases, and each case contains no actual (WPC) front passing through the center grid cell.

are punished half as severely as misses. Weighted CSI is maximized with lower probability thresholds – for all neighbourhood distances above 50 km, with $p_C^* \approx 0.5$ and $p_W^* \approx 0.75$. However, these thresholds still cause too few warm fronts, while causing cold-frontal zones to be unrealistically wide ($\sim$1000 km). Thus, I have subjectively chosen thresholds of $p_W^* = p_C^* = 0.65$ (black circles in Figures 6.22 and 6.23). Like the calculations for CSI and weighted CSI, this subjective choice is based only on validation data. Figure 6.24 shows results for these thresholds on the testing set, based on all 2920 time steps in the year 2017. Scores improve as neighbourhood distance increases, because more WPC fronts are matched with CNN fronts and vice-versa.

Figure 6.25, also based on all 2920 time steps in the testing data, breaks down performance by location, using a neighbourhood distance of 100 km. The patterns are similar for other neighbourhood distances, but with worse scores for 50 km and better scores for 150-200 km. Note that the number of predicted fronts generally dwarfs the number of actual fronts (*c.f.*, Figures 6.25a and 6.25b). This is to be expected, since predicted fronts are 2-D zones while actual fronts are 1-D lines. The hole north of Baffin Island exists because the WPC rarely draws fronts here in general (Figure 3.12c), so this area was explicitly masked out for model evaluation. The hole in southern Alaska exists because there are no predicted (Figure 6.25a) or actual (Figure 6.25b) fronts here in the testing data. In terms of both CSI and weighted CSI, the model performs best in the Pacific and Atlantic oceans and the hill country of Texas. Enhanced performance in the oceans is likely due to fronts having simpler structure here, while enhanced performance in Texas may be due to small sample size (Figure 6.25a-b). The model performs worst in the mountain west, especially British Columbia through Alaska, and the high Arctic, especially the Canadian Arctic archipelago and west coast of Greenland. Most of this area has low POD and high FAR, which means that most WPC fronts are not matched with a CNN front and vice-versa. This lack of agreement makes sense, as complex terrain makes the location and orientation of fronts more difficult to define.

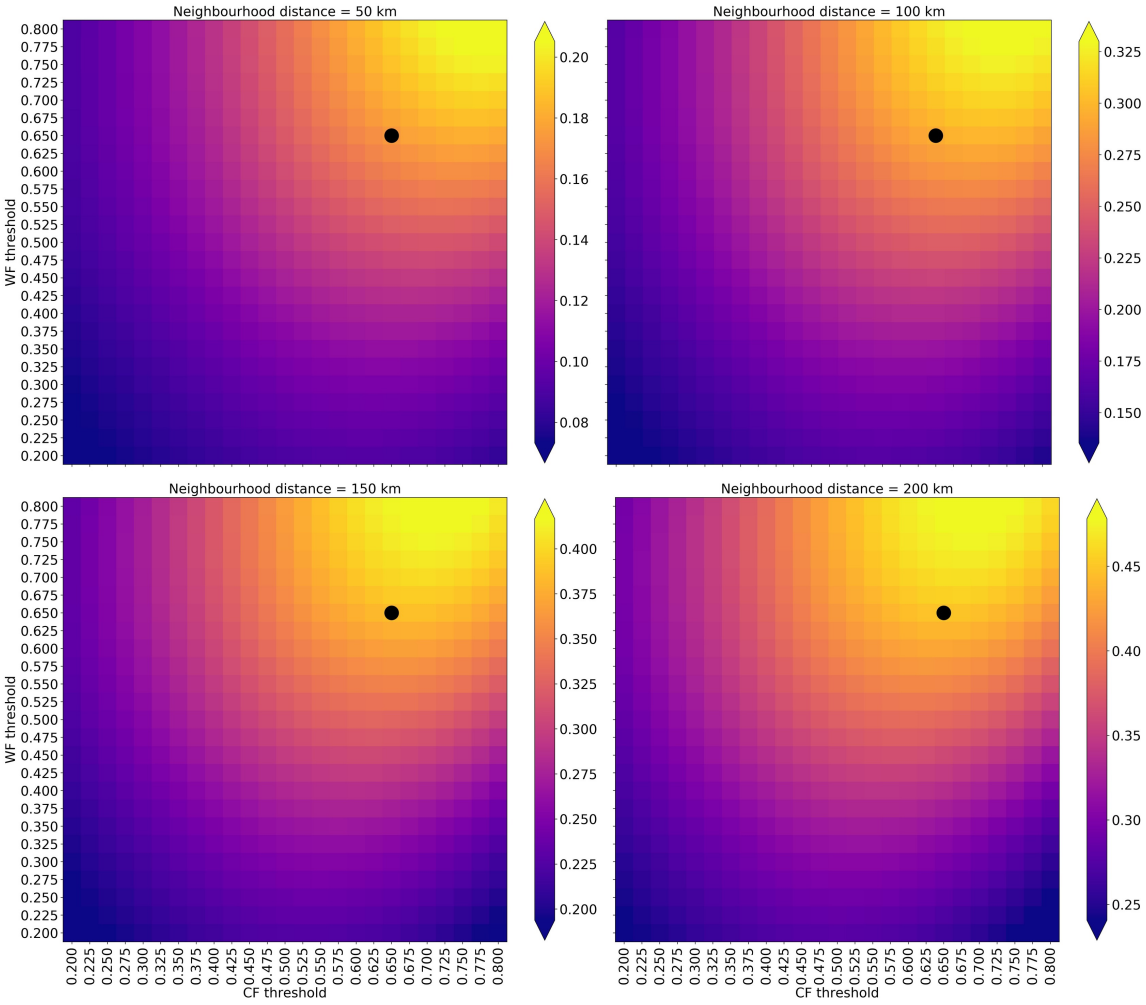


Figure 6.22: CSI for various probability thresholds on validation data. "WF threshold" and "CF threshold" are p_W^* and p_C^*, respectively, in Equation 5.4. The black circle shows subjectively selected thresholds.

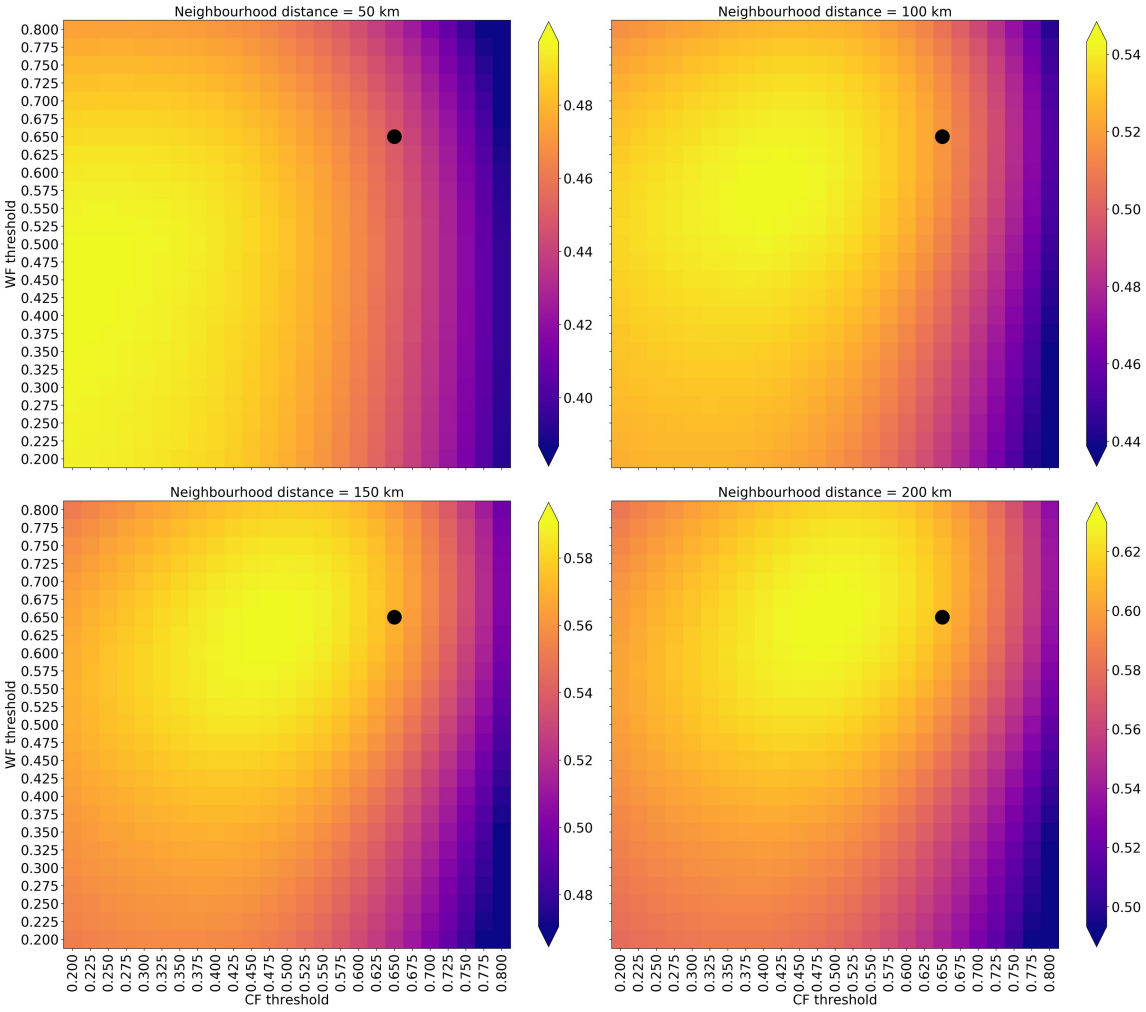


Figure 6.23: Weighted CSI for various probability thresholds on validation data. Formatted as in Figure 6.22.

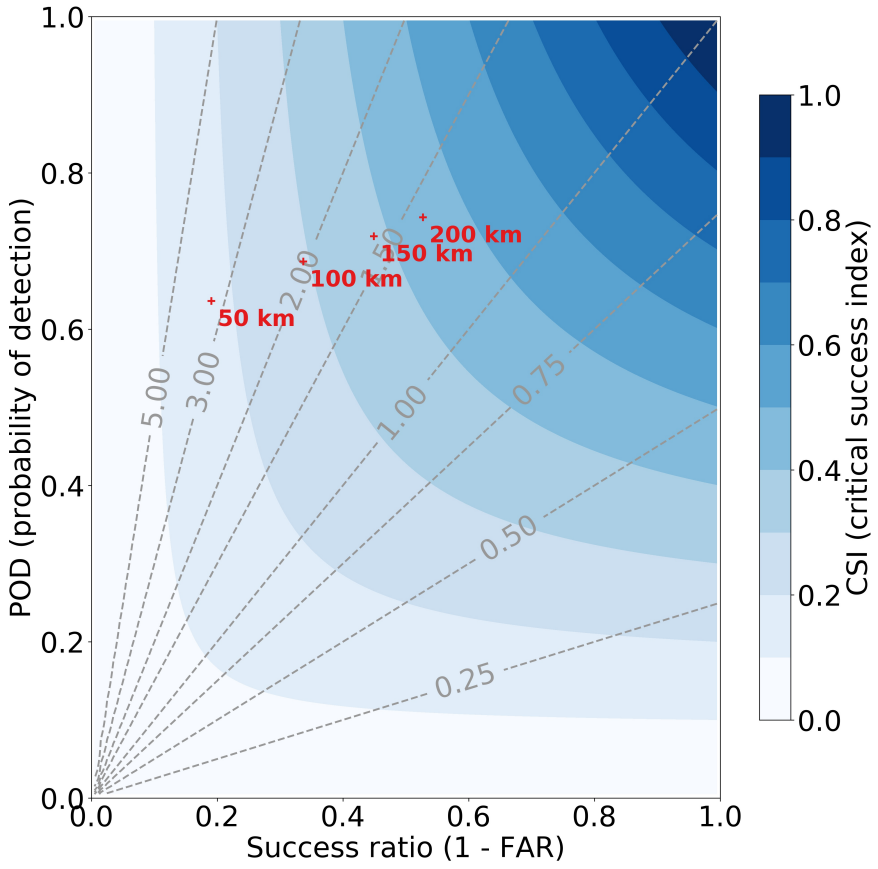


Figure 6.24: Performance diagram for front detection on testing data. Results are produced by the selected CNN (Section 6.2.1) with probability thresholds of 0.65. Quantities in the performance diagram are defined in Equations 5.5. Error bars (which are very small) show the 95% confidence interval, based on bootstrapping 1000 times.

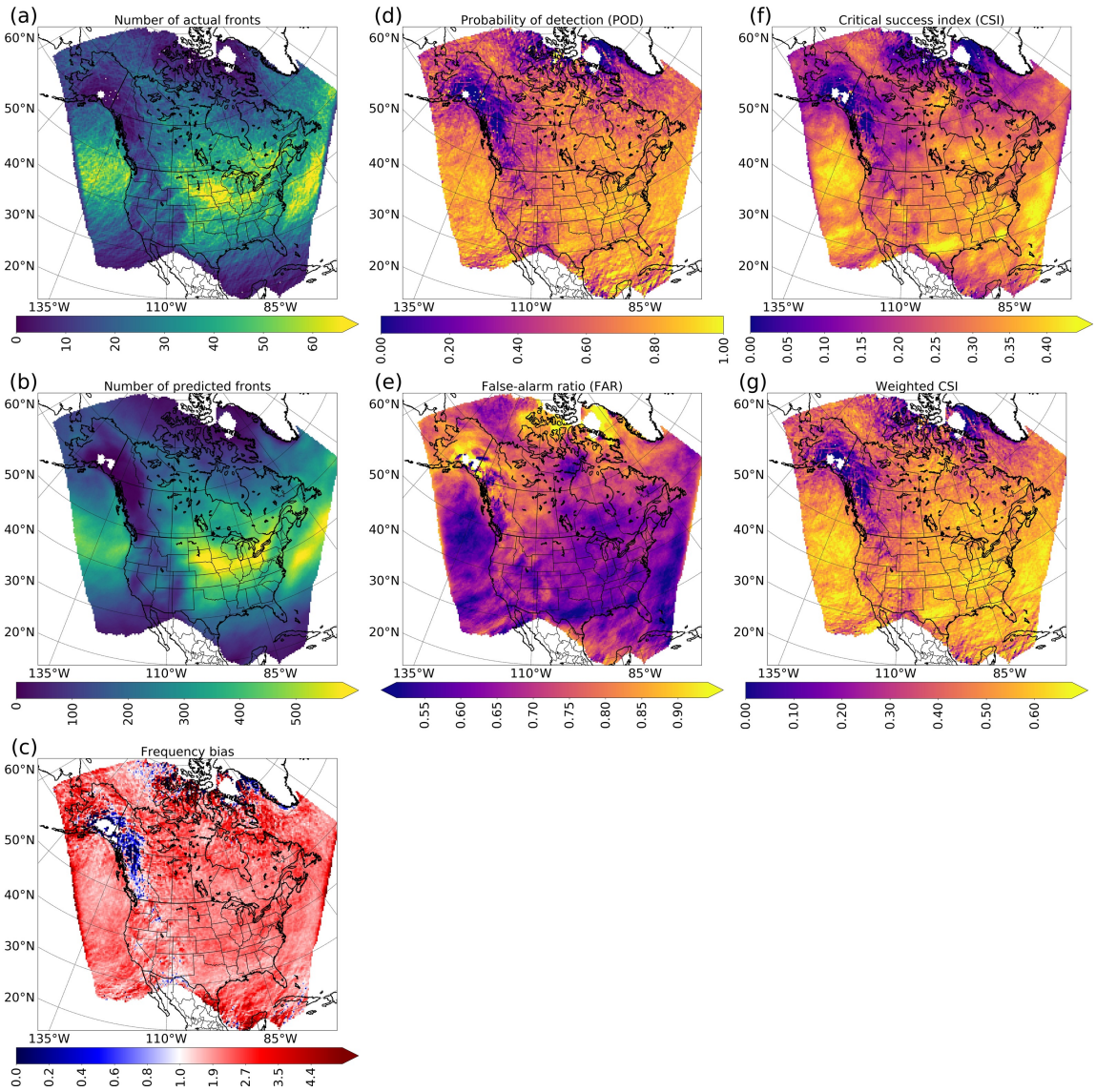


Figure 6.25: Regional performance of front detection on testing data. As in Figure 6.24, results are produced by the selected CNN with probability thresholds of 0.65. Quantities shown are defined in Equations 5.5.

Chapter 7

Interpretation of CNNs

This chapter discusses four interpretation methods used to understand what the models have learned. All four methods are applied to the testing data only, which ensures that they are not influenced by overfitting.[1] Section 7.1 describes the interpretation methods themselves, while Sections 7.2 and 7.3 discuss results for the tornado-prediction and front-detection models, respectively. Key results are summarized below.

I have made two contributions to the interpretation methods themselves. One is applying sanity checks to the saliency maps (Section 7.1.2), which ensure that the saliency maps truly depend on relationships learned by the models, as opposed to "noise." To my knowledge, no test for the fidelity of ML-interpretation results has yet been used in the physical sciences. Also, whereas the original sanity checks (Adebayo et al. 2018) depend on visual analysis, I have added a test to determine statistical significance. My second contribution is applying physical constraints to backward optimization (Section 7.1.4), which yields more realistic synthetic storms. This is a first step in expanding the adoption of physics-guided ML in the geosciences, as suggested by two recent position papers (Reichstein et al. 2019; Gil et al. 2019).

My interpretation results also contribute to the atmospheric-science literature. For tornado prediction, I have found three key results. First, the permutation test suggests that vorticity (azimuthal shear in MYRORSS) is the most important predictor, which helps justify recent efforts to improve estimates of this variable (Mahalik et al. 2019). Second, although the proximity sounding is generally not important, backward optimization finds

[1] For the selected tornado-prediction and front-detection models in this study, I have found that overfitting is minimal. In other words, model performance is similar for the training, validation, and testing data, with the exception of differences due to event frequency. However, using testing data for interpretation is an additional safeguard.

that low-level instability and wind shear are important predictors. The importance of low-level wind shear has also been highlighted in recent literature. Third, saliency maps for the MYRORSS model suggest that tornado probability decreases with low-level reflectivity in the RFD. This is consistent with recent work, suggesting that RFDs with too much negative buoyancy can lead to tornadogenesis failure. Saliency maps for the GridRad model emphasize the discreteness of the storm rather than the RFD, as discussed at the beginning of Section 7.2. Meanwhile, I have found two key results for front detection. First, class-activation maps (CAM) suggest that vertical continuity (between features at the surface and 850 mb) is more important for warm fronts than for cold fronts. The reason could be that warm fronts generally have weaker thermal gradients, so predictors at one vertical level are less often sufficient for warm-front detection. The second key result is produced by CAMs for the "worst misses," where meteorologists have identified a front at the center grid cell but the CNN has not. In many of these cases, there is weak thermal advection near the center grid cell and the CAM highlights a part of the domain with stronger advection, which appears more front-like. Subjectively, this suggests that the CNN can "correct" faulty human labels.

7.1 Methods

This section describes the interpretation methods themselves.

7.1.1 The Permutation Test

The permutation test quantifies the overall importance of each predictor variable, averaged over grid points and testing examples. The "importance" of predictor x_j is determined by how much model performance declines when x_j is permuted – *i.e.*, when spatial maps of x_j are randomly shuffled over testing examples. If performance declines significantly, x_j is an important predictor. If performance does not decline significantly, x_j is unimportant either in general or on its own (*i.e.*, x_j is correlated with other predictors). There are four

versions of the permutation test: single-pass forward (Breiman 2001), multi-pass forward (Lakshmanan et al. 2015b), single-pass backward, and multi-pass backward. Animations of the forward single-pass and multi-pass methods are shown in Supplementary Figures ES5 and ES6 of McGovern et al. (2019).

In the single-pass forward algorithm (Table C.1), only one predictor is permuted at the same time. The k^{th}-most important predictor, according to the test, is that for which permuting causes the k^{th}-largest decline in performance. In the multi-pass forward algorithm (Table C.2), the single-pass forward algorithm is run for N steps, where N is the total number of predictors. After each step the best intact (yet-to-be-permuted) predictor is permuted forever, so during the k^{th} step, k predictors are permuted at the same time. At the end of the multi-pass forward algorithm, all predictors have been permuted, causing the model to perform no better than random. The k^{th} predictor to be permuted forever is considered the k^{th}-most important. In the single-pass backward algorithm (Table C.3), all predictors are permuted at the beginning; then only one predictor is depermuted ("cleaned") at the same time. The k^{th}-most important predictor is that for which cleaning causes the k^{th}-greatest improvement in performance. In the multi-pass backward algorithm (Table C.4), all predictors are permuted at the beginning; then the single-pass backward algorithm is run for N steps. After each step the best still-permuted predictor is cleaned forever, so during the k^{th} step, k predictors are intact at the same time while the rest are permuted. At the end of the multi-pass backward algorithm, all predictors have been cleaned, giving back the original dataset.

The four versions of the test handle correlated predictors differently. For example, suppose that $x_n = 2x_m$, where x_m and x_n are two predictors. Thus, x_m and x_n are completely redundant (Pearson correlation of 1.0) and should have nearly identical effects on the trained model. Also, suppose that x_m and x_n are highly important and not correlated with any other predictors. In the single-pass forward test, neither x_m nor x_n will be considered important, because permuting one and not the other will have a minimal effect on

model performance. In the single-pass backward test, both x_m and x_n will be considered important, because cleaning either one will substantially improve model performance, even while leaving the other permuted. In the multi-pass forward test, once x_m has been permanently permuted, the most important remaining predictor should be x_n, since the dataset no longer contains information redundant with x_n. However, x_m will not be permanently permuted until late in the algorithm (since x_n contains the same information), so as in the single-pass forward test, neither predictor will be considered important.[2] In the multi-pass backward test, once x_m has been permanently cleaned, x_n will be considered unimportant, since the dataset contains information redundant with x_n. x_m will be permanently cleaned early in the algorithm, so x_m will be considered highly important and x_n will be considered unimportant.[3]

7.1.2 Saliency Maps

Saliency (Simonyan et al. 2014) is defined by

$$s = \frac{\partial a}{\partial x}\bigg|_{x=x_0}, \tag{7.1}$$

where a is the activation of a neuron in the model; x is a scalar predictor (one variable at one grid point); and x_0 is the value of x in a testing example. Thus, saliency is an approximation to $\frac{\partial a}{\partial x}$, linearized around the true value of x in a testing example. This can be computed for all scalar predictors x, resulting in a map. In this study a is always the activation of an output neuron, which is a class probability (tornado, no front, warm front, or cold front). For example, saliency maps shown in Section 7.2 quantify the sensitivity of tornado probability to each input, answering the question "How could the storm be perturbed to increase/decrease tornado probability?"

[2]The last two sentences assume that x_m is the first of the two predictors to be permanently permuted. If not, replace "x_m" with "x_n" and vice-versa.

[3]The last two sentences assume that x_m is the first of the two predictors to be permanently cleaned. If not, replace "x_m" with "x_n" and vice-versa.

A key advantage of saliency maps is that they share the dimensions of the predictors, allowing them to be viewed as images and overlain with the predictors. A key disadvantage is that, while saliency highlights the most important predictors for changing the model's prediction, it says nothing about which predictors are most important in generating the actual prediction. This disadvantage can be alleviated by methods such as layerwise relevance propagation (Samek et al. 2017; Montavon et al. 2018) and class-activation maps. Another disadvantage is that saliency is a linear approximation, so it quantifies the model's sensitivity when the predictors are perturbed locally but not globally. Global sensitivity can be shown in partial-dependence plots (PDP; Friedman 2001), but it would be very cumbersome to plot this graph for every scalar predictor (for each model developed herein, the number of scalar predictors is of order 10^4-10^5). Thus, my way of alleviating this disadvantage is with backward optimization, whose connection with saliency maps will be explained in Section 7.1.4.

Adebayo et al. (2018) proposed three sanity checks to ensure that saliency maps truly depend on relationships learned by the model, rather than artifacts caused by the model architecture or grid dimensions, such as the Buell patterns (Buell 1979) that often arise in principal-component analysis. The first sanity check is the edge-detector test, which compares actual saliency maps (produced by the trained model) with feature maps produced by an untrained edge-detector (a convolutional filter with fixed weights, like those used in traditional image-processing applications). The second is the model-parameter-randomization test, which compares saliency maps from the trained model before and after randomizing the weights in some layers. The third is the data-randomization test, which compares actual saliency maps with those produced by an equivalent model trained on random labels (where labels are shuffled among training examples). The model trained on random labels should overfit, learning relationships that can predict the random training labels but do not generalize out of training. Thus, when applied to non-training data, the model should produce nonsensical saliency maps, bearing no resemblance to true physical relationships. I

have not implemented the data-randomization test, because for both tornado prediction and front detection, I was unable to train a model that overfit randomly labeled data.[4] Although the resulting models produced nonsensical saliency maps for the testing data as expected, since they had not learned to make skillful predictions, this test added no value beyond the edge-detector test.

Each sanity check produces a set of dummy saliency maps (one for each testing example), which I compare to the actual saliency maps. To assess statistical significance, I apply a two-tailed Monte Carlo test (sometimes called bootstrapped randomization; Dwass 1957) to the composite difference, as described below. The method of probability-matched means (PMM) is used because it preserves spatial structure better than taking the arithmetic mean at each grid point.

1. For each predictor variable in each example, rescale actual saliency values to percentiles, using all grid points. Do the same for dummy saliency values. Without this rescaling, there are often large differences in magnitude between the actual and dummy saliency maps, causing nearly all differences to be considered statistically significant.

2. Create the sampling distribution, containing 20 000 randomized differences for each scalar predictor. Specifically, do the following 20 000 times:

   (a) Randomly shuffle examples between the actual and dummy sets (A and D respectively), yielding A' and D'. Both A' and D' contain a mix of actual and dummy saliency maps.

[4] Even with all forms of regularization (L_2, dropout, and data augmentation) turned off, I could not train a model that outperforms random guessing on the training data. I suspect that this is because I use large training batches (over 1000 examples each) with little temporal autocorrelation (because the examples are drawn from different times), as discussed in Sections 5.1 and 5.2.

(b) Take the PMM for both A' and D'. Record the randomized composite difference (PMM over set A' minus PMM over set D') for each scalar predictor (each variable at each grid point).

3. For each scalar predictor:

   (a) Find the true composite difference (PMM over set A minus PMM over set D). This is the test statistic.

   (b) Find the percentile of the test statistic within the sampling distribution.

   (c) If the percentile is < 2.5 or > 97.5, the test statistic is significant at the 95% confidence level.

When conducting multiple hypothesis tests (here, one for each scalar predictor), one must control the false-discovery rate (FDR). For example, if each test were conducted independently with a 95% confidence level, one would expect 5% of differences to be "significant" even with purely random data. However, in step 2a the examples are shuffled once, rather than shuffling independently for each scalar predictor. Thus, the shuffling procedure preserves correlations among the scalar predictors, which obviates the need to explicitly control the FDR.

7.1.3 Class-activation Maps (CAM)

Class activation (Zhou et al. 2016) is the amount of evidence for a particular class. The original CAM method (Zhou et al. 2016) works only for a specific type of CNN architecture, so I use a generalized version called gradient-weighted CAM (Grad-CAM; Selvaraju et al. 2017). The original CAM method requires the CNN to end with a global-average-pooling layer, whereas Grad-CAM allows for a wider variety of architectures, including those ending with dense layers.

Under Grad-CAM, class activation is defined separately for each data example, at each convolutional layer $\mathscr{L}$, at each grid point (i, j) in the feature maps output by $\mathscr{L}$. Specifically,

$$E_{ij,k} = \max(\sum_{c=1}^{C} \alpha_k^c A_{ij}^c, 0)$$
$$\text{and } \alpha_k^c = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{\partial \tilde{p}_k}{\partial A_{ij}^c}. \tag{7.2}$$

A_{ij}^c is the value for channel c at grid point (i, j) of the feature map output by layer $\mathscr{L}$; $\tilde{p}_k$ is the psuedo-probability[5] of class k; M and N are the number of spatial rows and columns, respectively, in the feature map output by layer $\mathscr{L}$; C is the number of channels in the feature map output by layer $\mathscr{L}$; and $E_{ij,k}$ is the class activation (or evidence) for class k at grid point (i, j). Although Equations 7.2 assume that the feature map (and thus the input map) has two spatial dimensions, Grad-CAM can easily be generalized to any number of spatial dimensions.[6]

By applying Equations 7.2 to every grid point for a given convolutional layer and data example (iterating over i and j), one can create a class-activation *map* (CAM). In the ML literature, these maps are commonly plotted on top of the input data (predictors). However, the spatial dimensions of the feature map output by layer $\mathscr{L}$ may not match the spatial dimensions of the input, due to pooling layers in between. Thus, before plotting the CAM, it is common practice to upsample the CAM to the dimensions of the input data. For all CAMs shown in this study, I use cubic interpolation to do this upsampling.

Conceptually, are four key differences between CAMs and saliency maps. First, CAMs yield one value per grid point, whereas saliency maps yield one value per scalar predictor. Second, CAMs yield only non-negative values (in this context there is no such thing as

[5]Before applying the activation function to the output layer, which in this study is either sigmoid (for tornado prediction) or softmax (for front detection).

[6]For example, in three dimensions, let the depth coordinate be d and the number of depths be D. Then $E_{ij,k}$ is replaced with $E_{ijd,k}$; A_{ij}^c is replaced with A_{ijd}^c; and the right-hand side of the second equation becomes $\frac{1}{MND} \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{d=1}^{D} \frac{\partial \tilde{p}_k}{\partial A_{ijd}^c}$, a spatial average over three dimensions rather than two.

negative evidence), whereas saliency can be positive or negative. Third, CAMs highlight the most important values for generating the model's actual prediction, whereas saliency maps highlight the most important values for changing the prediction. Fourth, CAMs are specific to a single convolutional layer, while saliency maps are not. In general, CAMs produced by deeper layers are smoother, with large values confined to a smaller region. This is because deeper layers learn higher-level abstractions, which allows them to focus more selectively on important parts of the image.

7.1.4 Backward Optimization (BWO)

BWO (Erhan et al. 2009) creates a synthetic input that extremizes (minimizes or maximizes) the activation of a particular neuron in the model. BWO is sometimes called activation maximization (Erhan et al. 2009), feature optimization (Olah et al. 2017), or optimal input (Toms et al. 2019). The BWO procedure is basically training in reverse. During training, gradient descent (Section 6.5 of Goodfellow et al. 2016) is used to adjust weights in a way that minimizes the loss function (Equation 4.3). During BWO, gradient descent is used to adjust predictor values in a way that extremizes the neuron activation. In this study I focus on output neurons, for which the activation is a class probability. For example, if the goal is to maximize tornado probability, BWO creates a prototypical tornadic storm; if the goal is to minimize tornado probability, BWO creates a prototypical non-tornadic storm.

The BWO procedure involves many iterations. At each iteration the synthetic example is updated via the rule

$$\mathbf{X} \leftarrow \mathbf{X} - \alpha \frac{\partial J}{\partial \mathbf{X}}, \tag{7.3}$$

where $\mathbf{X}$ is the tensor of predictor values; J is the loss function; $\frac{\partial J}{\partial \mathbf{X}}$ is a gradient tensor with the same dimensions as $\mathbf{X}$; and α is the learning rate, usually a positive number $\ll 1$. Both α and the number of iterations are hyperparameters.

In the simplest framework, $J = (p - p^*)^2$, where p is the CNN-generated class probability and p^* is the desired probability (0.0 or 1.0). Thus, Equation 7.3 can be written as

$$\mathbf{X} \leftarrow \mathbf{X} - 2\alpha(p - p^*)\frac{\partial p}{\partial \mathbf{X}}, \tag{7.4}$$

where $\frac{\partial p}{\partial \mathbf{X}}$ is the saliency map. Thus, BWO consists of serially adding a small fraction of the saliency map to the predictor map. Although saliency is a linear approximation, a new saliency map is created at each iteration of BWO, linearized around the new synthetic storm. Thus, BWO overcomes the linear limitation of saliency maps.

Gradient descent adjusts but does not initialize values, so it requires a starting point or "baseline." Some options are all-zeros, random noise, or a real testing example. The advantage of all-zeros and random noise is that the baseline does not resemble a real example, so the synthetic example ultimately produced is more novel (different from the baseline). The disadvantage is that, because the baseline is very unrealistic, the synthetic example is often very unrealistic as well. This problem is alleviated by using a real example as the baseline. Sturmfels et al. (2020) discuss similar effects of the baseline on an interpretation method called "integrated gradients." However, in my experience the synthetic examples are still quite unrealistic, so I add constraints to the loss function used in Equations 7.3-7.4.

The loss function with constraints is defined by

$$\begin{aligned} J = (p - p^*)^2 + \lambda_2 \|\mathbf{X} - \mathbf{X_0}\|^2 + \lambda_{\text{minmax}} \sum_j \|\max(\mathbf{X_j^{\text{min}}} - \mathbf{X_j}, \mathbf{0})\|^2 \\ + \lambda_{\text{minmax}} \sum_j \|\max(\mathbf{X_j} - \mathbf{X_j^{\text{max}}}, \mathbf{0})\|^2 \end{aligned}, \tag{7.5}$$

where the first term is the unconstrained loss function; the second term is the L_2 penalty on the difference between the original example $\mathbf{X_0}$ and synthetic example $\mathbf{X}$; the third term is the penalty for violating minimum-constraints; and the fourth term is the penalty for violating maximum-constraints. $\mathbf{X_j}$ is a tensor with values of the j^{th} variable only; $\mathbf{X_j^{\text{min}}}$ is a tensor of the same shape, where every value is the minimum allowed for the j^{th} variable; $\mathbf{X_j^{\text{max}}}$ is a tensor of the same shape, where every value is the maximum allowed for the j^{th}

variable; and $\mathbf{0}$ is the tensor of the same shape, where every value is zero. For instance, if the j^{th} variable is fractional relative humidity, $\mathbf{X}_\mathbf{j}^{\text{min}} = \mathbf{0}$ and $\mathbf{X}_\mathbf{j}^{\text{max}} = \mathbf{1}$. λ_2 is the strength of the L_2 penalty, and λ_{minmax} is the strength of the min-max penalty, both hyperparameters.

For the tornado models, I use $\lambda_2 = 1$ and $\lambda_{\text{minmax}} = 10$, with the following min-max constraints. For the front-detection models, I use $\lambda_2 = 1$ and $\lambda_{\text{minmax}} = 0$, as I have found min-max constraints to be unnecessary.

- Minimum reflectivity in radar image = 0 dBZ
- Minimum spectrum width in radar image = 0 m s^{-1}
- Minimum θ_v in sounding = 0 K
- Minimum relative humidity in sounding = 0%
- Minimum specific humidity in sounding = 0 g kg^{-1}
- Maximum relative humidity in sounding = 100%
- Maximum specific humidity in sounding = 1000 g kg^{-1}

It is possible that not squaring the terms in Equation 7.5 (*e.g.*, using $\|\mathbf{X} - \mathbf{X_0}\|$ rather than $\|\mathbf{X} - \mathbf{X_0}\|^2$) would look to smoother and more realistic synthetic data. This possibility is reiterated in the discussion of future work in Chapter 9.

7.2 Tornado Prediction

This section discusses interpretation results for both the MYRORSS and GridRad models. The key difference, revealed by saliency maps, is that the GridRad model is more sensitive to discreteness of the storm while the MYRORSS model is more sensitive to storm structure near the rear-flank downdraft. However, one could argue that this difference is caused by artifacts related to CNN architecture, grid shape, or storm samples (*e.g.*, the best hits for MYRORSS and best hits for GridRad are generally different storms). These objections are

addressed in Appendix D, where interpretation methods are run on matched MYRORSS and GridRad storms, using CNNs with identical architecture. Results in Appendix D are generally similar to those presented in this section, including the "key difference" mentioned above.

7.2.1 GridRad Model

Figure 7.1 shows results of the permutation test, which measures the overall importance of each predictor variable. In each panel, the most important predictor is at the top while the least important is at the bottom. According to all four versions of the test, the most important predictor is reflectivity; the second-most important is vorticity; and these rankings are significant at the 95% confidence level, indicated by the bold font. According to two versions of the test, the third-most important predictor is divergence in the radar image; according to the other two versions, the third-most important predictor is v-wind in the proximity sounding. The reason for this discrepancy is not obvious, since Pearson correlations among these predictors are quite small (Table 7.1), suggesting that they have little interdependence. After the top three predictors, differences between successive predictors are generally not significant, indicated by the lack of bold font. In general, radar variables (representing the current storm) are much more important than sounding variables (representing the near-storm environment). This is not surprising, given that the model predicts tornadoes for pre-existing storms over a short lead time (one hour). If the lead time were longer or predictions were made for future storms (accounting for convective initiation), the relative importances of the radar image and sounding would likely be closer.

Figure 7.2 shows saliency maps for extreme cases. Saliency maps are computed individually for each storm, then composited via probability-matched means (PMM). For the sake of brevity, saliency is shown for only three radar variables at three heights. Saliency maps for divergence (the other radar variable) generally suggest that tornado probability increases with low-level convergence, and upper-level divergence, in the reflectivity core.

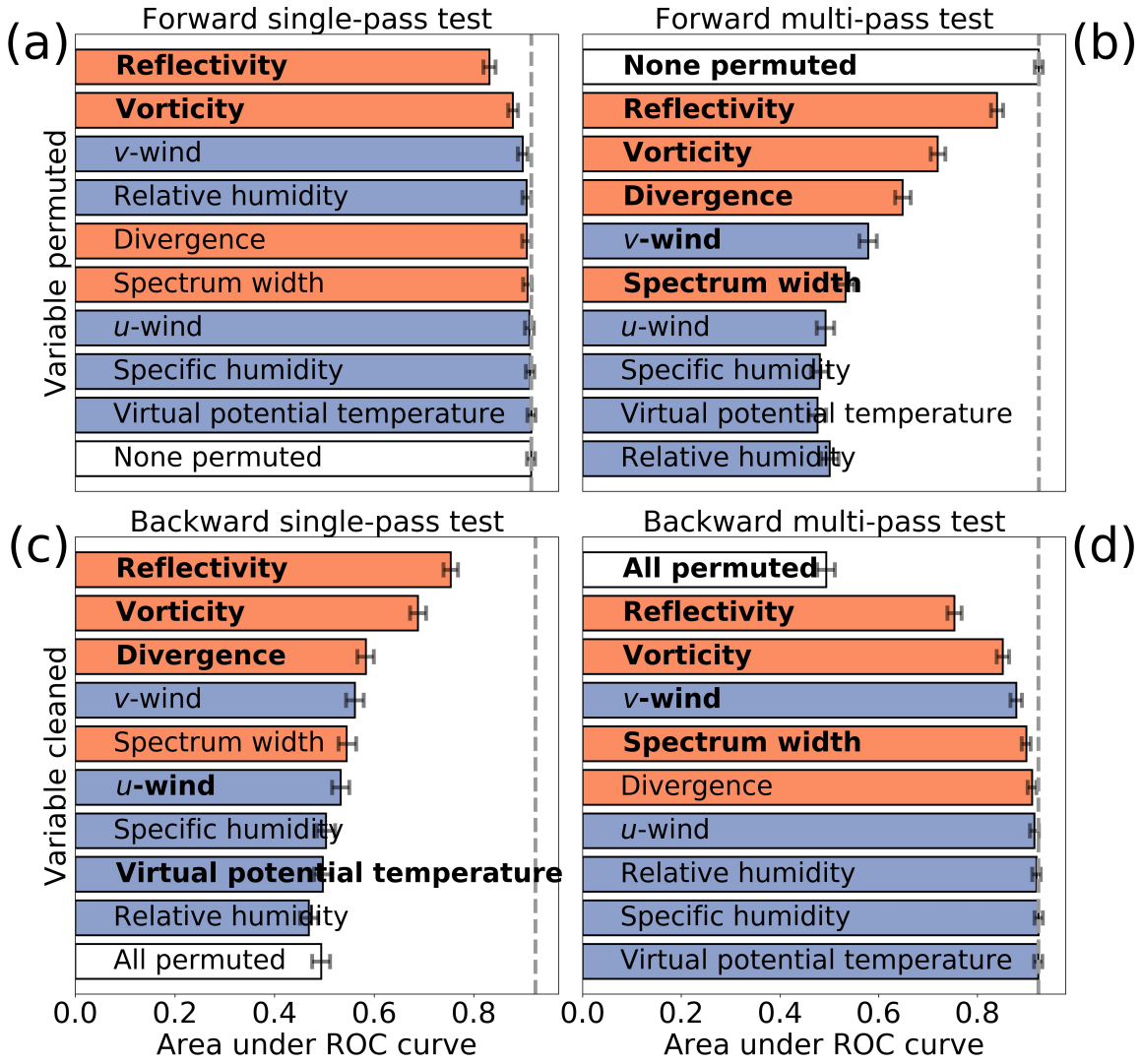


Figure 7.1: Results of permutation test for GridRad model, based on 30 000 randomly selected testing examples. The 30 000 examples are the same for all four versions of the test. Radar variables are in orange, and sounding variables are in purple. Grey error bars show the 95% confidence interval, determined by bootstrapping area under the ROC curve (AUC) 1000 times. If variable x_j is in bold font, this means x_j is significantly more important than the variable below (at the 95% confidence level, also determined by bootstrapping 1000 times).

Table 7.1: Pearson correlations among top GridRad variables. These correlations are computed for the same 30 000 examples used in the permutation test. To compute the correlation between v-wind (which is a sounding variable) and a radar variable, the spatial mean is first taken for each example, since the sounding and radar variables are on different grids.

	Reflectivity	**Vorticity**	**Divergence**	***v*-wind**
Reflectivity	1.000	0.077	-0.124	0.109
Vorticity		1.000	-0.017	0.045
Divergence			1.000	0.009
***v*-wind**				1.000

Meanwhile, saliency maps for the proximity sounding generally suggest that tornado probability increases with low-level humidity and low-level instability (high θ_v near the surface with lower θ_v just above). For the best hits (Figure 7.2a), saliency is maximized on the right-rear flank of the storm, where a potential tornado would be expected. Tornado probability increases with reflectivity, vorticity, and spectrum width at all heights. Saliency for reflectivity and spectrum width is highest at 10 km AGL, while saliency for vorticity is highest at 2 km AGL. Thus, the model is particularly sensitive to low-level rotation and storm depth. For the worst false alarms (Figure 7.2b), the saliency map is similar to the best hits, except that values are slightly larger. This is because model probabilities are slightly higher for the best hits (99.2% versus 98.8%), meaning that probabilities for the best hits have less room to increase.

For the worst misses (Figure 7.2c), saliency is generally positive in the reflectivity core and negative in the surrounding area. This suggests that tornado probability would increase if the storms were made stronger and more discrete. Also, the shapes of positive saliency maxima for the worst misses (dark solid contours in Figure 7.2c) are similar to maxima in the corresponding radar fields for the best hits (heat maps in Figure 7.2a). For example,

the saliency maximum for 10-km reflectivity in Figure 7.2c matches the 10-km reflectivity core in Figure 7.2a; the saliency maximum for 2-km vorticity in Figure 7.2c matches the 2-km vorticity maximum in Figure 7.2a; etc. This suggests that the most effective way to increase tornado probability for the worst misses, which are mostly QLCS cells, is to make them look like the best hits, which are supercells. This is further evidence that the model has not learned useful relationships for QLCS tornadoes. Also, note that saliency for the worst misses is generally much higher than for the best hits and worst false alarms, because probabilities for the worst misses (average of 8.6%) have much more room to increase. For the best correct nulls (Figure 7.2d), the saliency map is similar to the worst misses, except that values are much lower. This is because saliency measures the model's sensitivity to small changes in the predictors (Equation 7.1) and, since the best correct nulls look very unlike tornadic storms, small changes can increase probability by only a small amount.

Figure 7.3 shows class-activation maps (CAM), which quantify the amount of tornado evidence at each grid point, for extreme cases. CAMs are computed individually for each storm, then composited via PMM. As discussed in Section 7.1.3, each convolutional layer produces a different CAM. Figure 7.3 shows CAMs for the third convolutional layer, which is the deepest (Figure 5.2). These maps are similar to CAMs produced by the first two layers, except that tornado evidence (non-zero values) is confined to a smaller portion of the 3-D space. For the best hits and worst false alarms (Figure 7.3a-b), tornado evidence is maximized on the right-rear flank and in the low-level weak-echo region. For the worst misses, tornado evidence is confined to the area that looks most like a supercell, with the highest values of all three radar variables. Tornado evidence decreases sharply outside this maximum, likely because the surrounding area reveals the storms' linear structure, which is generally not associated with tornadoes. Like the saliency map (Figure 7.2c), this suggests that the model has not learned useful relationships for QLCS tornadoes. For the best correct nulls (Figure 7.3d), non-zero tornado evidence coincides with the maximum in low-level vorticity. However, these class activations are very small (note the logarithmic

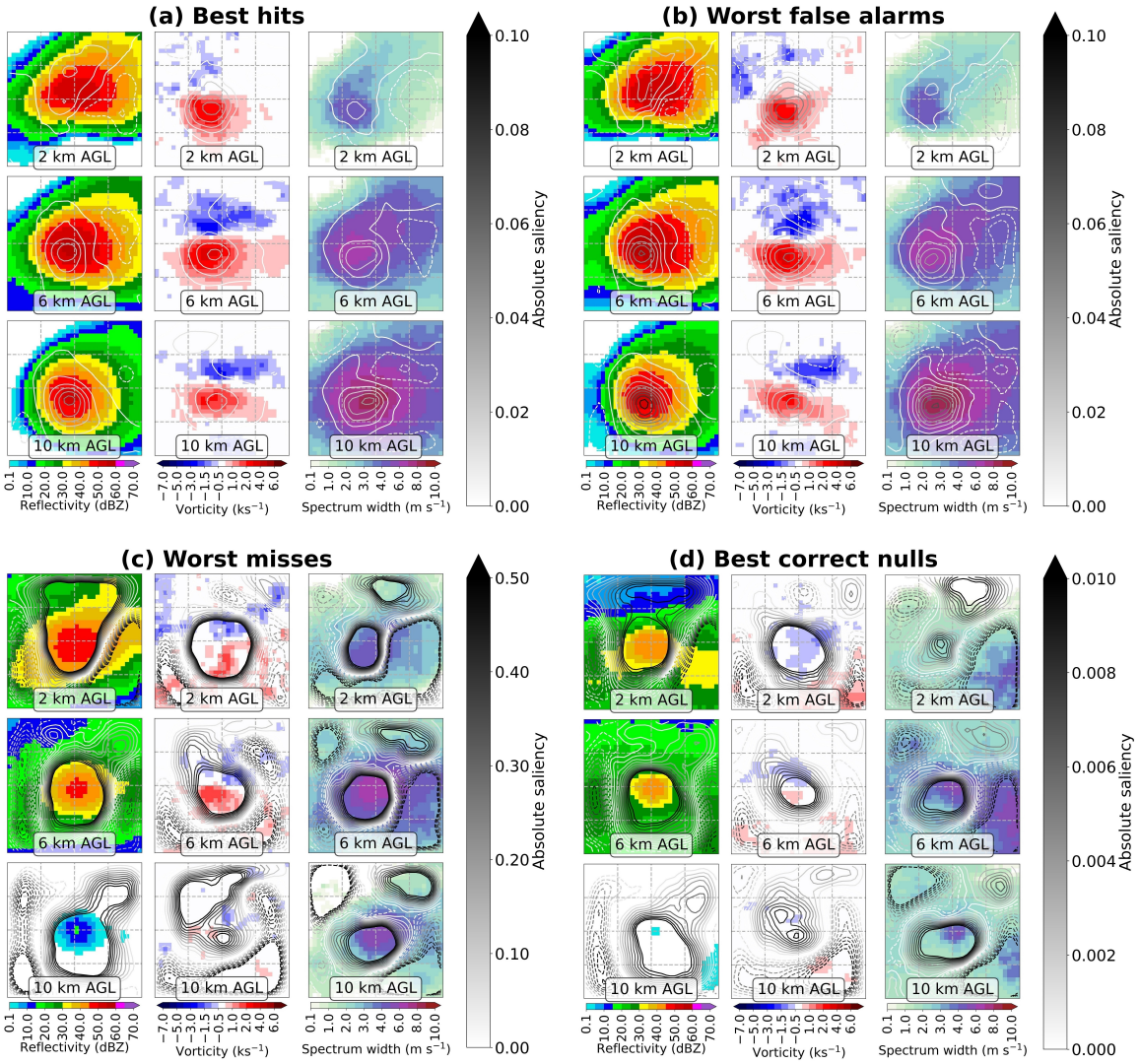


Figure 7.2: Saliency maps for extreme GridRad cases. Storm motion points to the right. Heat maps show the radar fields and are identical to those shown in Figure 6.12, while line contours show saliency. Solid contours indicate positive saliency (tornado probability increases with the underlying radar value), while dashed contours indicate negative saliency (tornado probability decreases with the underlying radar value). Each image (both the radar and saliency fields) is a PMM composite over 100 examples.

colour scale), likely because they are not collocated with reflectivity values above 35 dBZ, generally considered the threshold for supercells (Trapp et al. 2005b; Smith et al. 2012).

Figure 7.4 shows synthetic storms created by applying BWO to the best hits, with the goal of decreasing tornado probability to 0.0. BWO is applied to each storm individually, and the resulting synthetic storms are composited via PMM. On average for the 100 best hits, tornado probability thereby decreases from 99.2% to 6.9%. In general, BWO decreases all three radar variables in the core and mesocyclone (makes the storms weaker) and increases all three variables in the surrounding area (makes the storms less discrete). BWO has very little effect on the soundings, with two exceptions. First, it decreases low-level moisture, which also decreases CAPE for surface-based convection. Second, it rotates the wind clockwise at the bottom two levels, decreasing low-level wind shear, which is especially important for tornadogenesis (Markowski and Richardson 2009, 2014). Thus, BWO reveals that even though the sounding is generally much less important than the radar image (Figure 7.1), parts of the sounding rival the radar image in importance. However, the changes made by BWO do not seem to support a probability decrease to 6.9%. In other words, the synthetic storms still appear to have high tornado potential. This is because BWO seeks small changes that dramatically decrease tornado probability. Thus, BWO creates adversarial examples (Goodfellow et al. 2014), which "trick" the model into making an unrealistic prediction. Although constraints in the loss function (Equation 7.5) alleviate this problem, they do not solve it completely. Figure 7.5 shows the same results but with no constraints ($\lambda_2 = \lambda_{\text{minmax}} = 0$ in Equation 7.5), which results in large spatial discontinuities for both the dewpoint and temperature profiles in the sounding (panel d).

Figure 7.6 shows synthetic storms created by applying BWO to the worst misses, with the goal of increasing tornado probability to 1.0. On average for the 100 storms, tornado probability thereby increases from 8.6% to 95.3%. In general, changes are opposite to those made for the best hits, where the goal was to decrease probability. BWO increases all three radar variables in the core and decreases all three variables outside the core, especially in

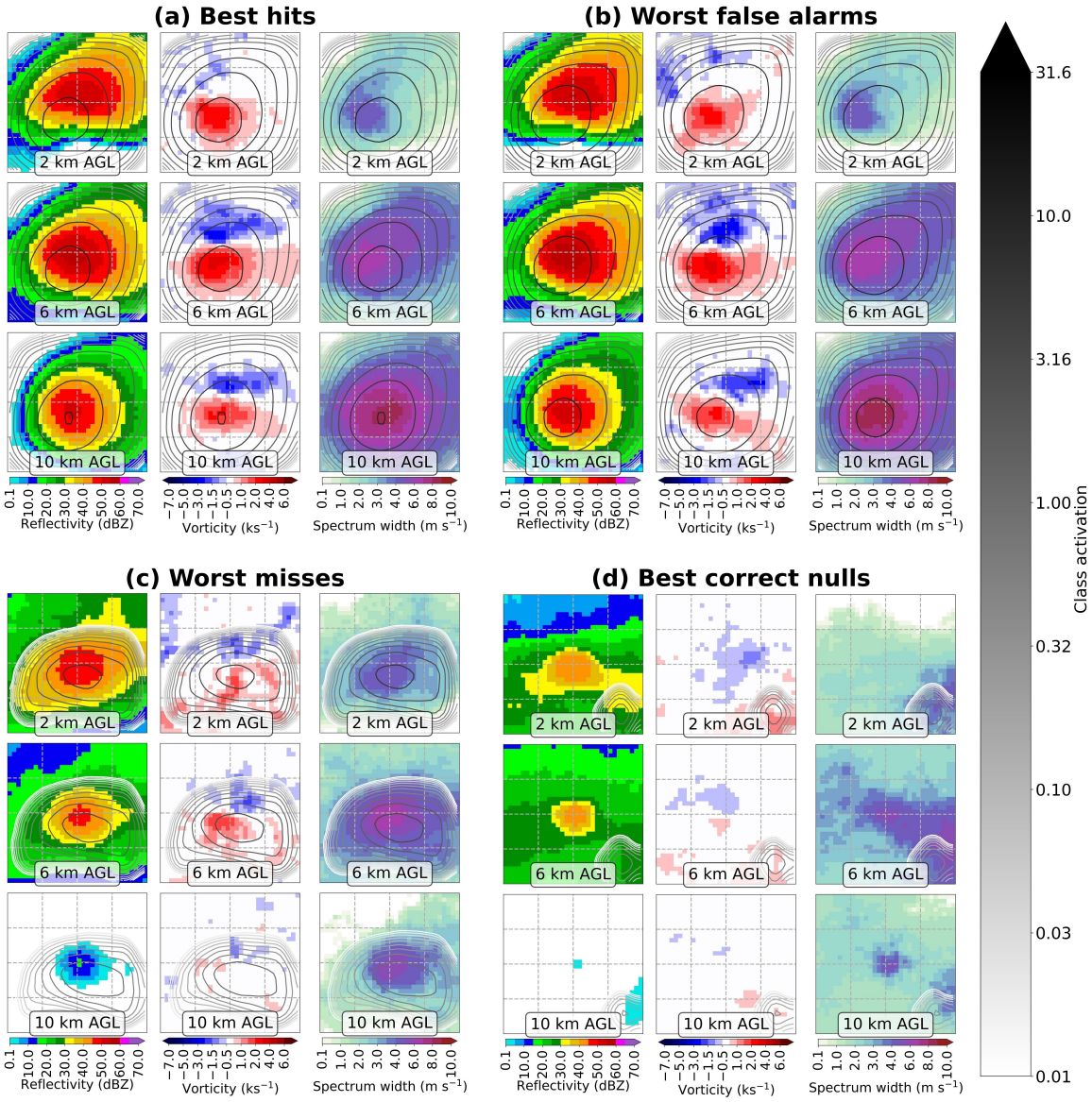


Figure 7.3: Class-activation maps for extreme GridRad cases. Storm motion points to the right. Heat maps show the radar fields and are identical to those shown in Figure 6.12, while line contours show class activation. Each image (both the radar and class-activation fields) is a PMM composite over 100 examples.

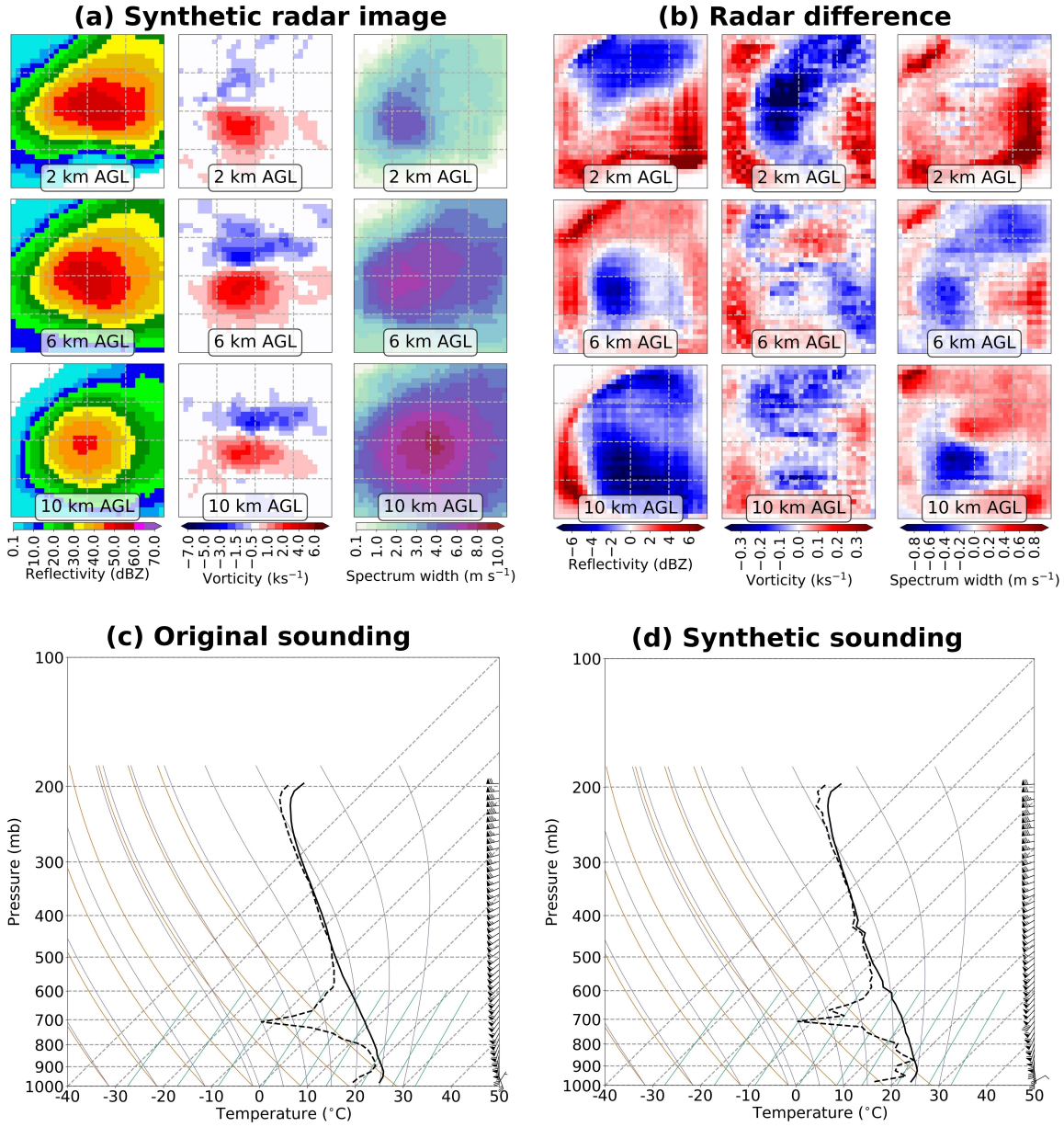


Figure 7.4: Backward optimization for best hits in GridRad model. Storm motion points to the right, and each image is a PMM composite over 100 examples. BWO is applied individually to each storm, with the goal of decreasing tornado probability. The original radar image (before optimization) is shown in Figure 6.12a. [a] Synthetic radar image, after optimization. [b] Difference (synthetic minus original radar image). [c] Original sounding, before optimization. [d] Synthetic sounding, after optimization.

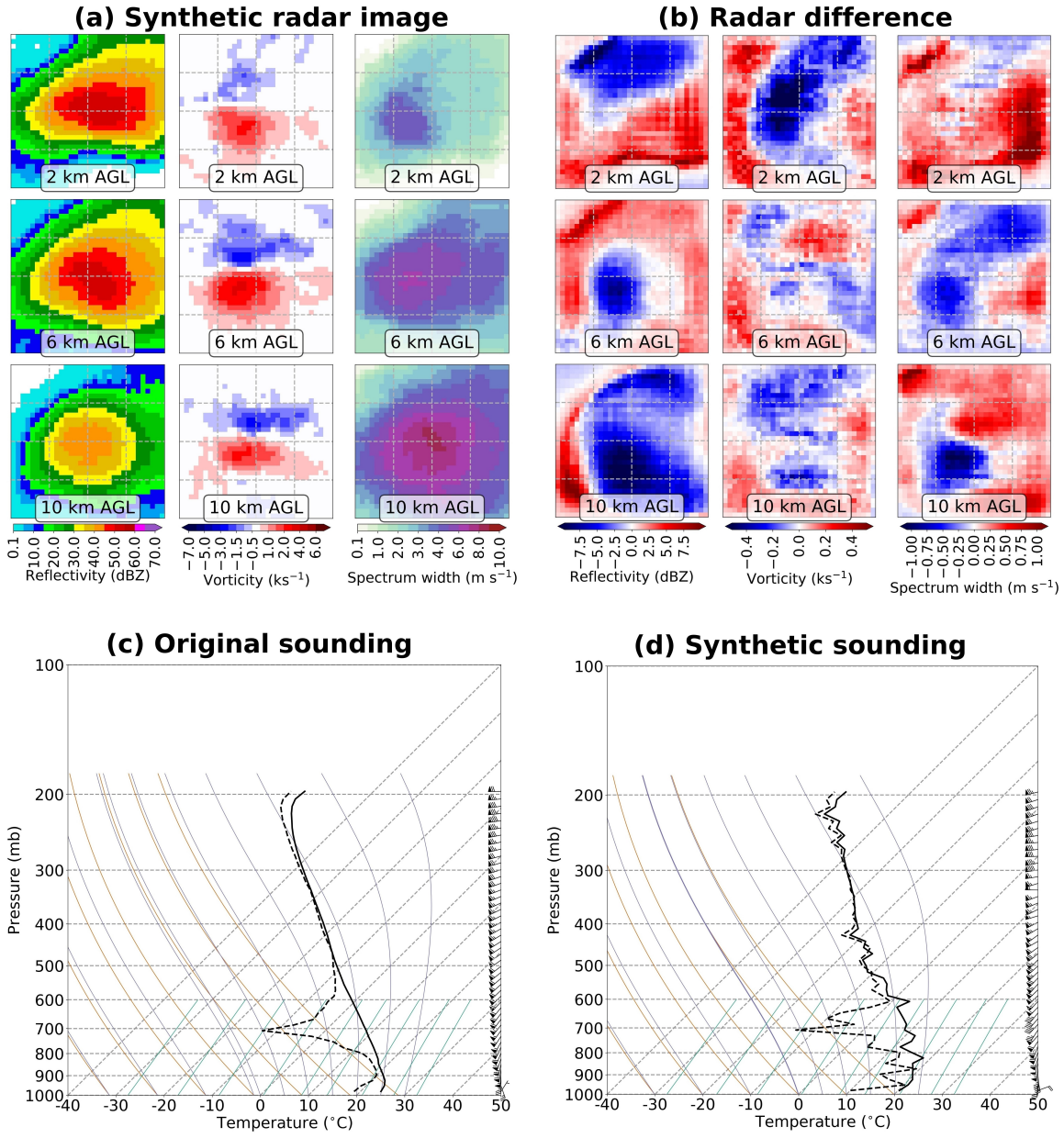


Figure 7.5: Unconstrained backward optimization for best hits in GridRad model. Same as Figure 7.4, except that backward optimization was done with no constraints.

the forward flank, making the storm appear somewhat more discrete and less like a QLCS cell. Again, BWO has very little effect on the sounding except near the surface, where it increases moisture and rotates the wind counterclockwise, the second change increasing low-level wind shear.

Figures 7.7 and B.3-B.4 show sanity checks for saliency maps on all extreme cases except the worst false alarms, for which results are very similar to the best hits. The model-parameter-randomization test (Section 7.1.2) is shown only for the third convolutional layer and second dense layer (Figure 5.2). Results for the first dense layer produce a similar number of significant differences, while results for the first two convolutional layers produce fewer significant differences, because randomizing weights in shallower layers has a smaller effect on the model's prediction and therefore on saliency.

Figure 7.7 shows results for the best hits. All three dummy saliency maps (Figure 7.7b-d) have substantially greater values than the actual saliency map (Figure 7.7a), but the stippling shows significant differences between rescaled saliency values, as discussed in Section 7.1.2. The model-parameter-randomization test (Figure 7.7c-d) leads to significant differences for 23% and 27% of scalar predictors, respectively, which suggests that actual saliency maps truly depend on the learned weights in both layers. Meanwhile, the edge-detector test (Figure 7.7b) leads to significant differences for 20% of scalar predictors, but differences at 6 and 10 km AGL are largely insignificant. This suggests that the shape of saliency maps (*i.e.*, rescaled values) at 6 and 10 km AGL can be mostly replicated by an untrained edge-detection filter. However, the actual (unscaled) saliency values cannot be replicated, as shown by the difference in colour scales between Figures 7.7a and 7.7b. Results are similar for the worst misses (Figure B.3) and best correct nulls (Figure B.4), except that significant differences for the edge-detector test are slightly less confined to 2 km AGL. In general, saliency maps for the GridRad model cannot be trivially replicated. This and the model's good performance suggest that interpretation results reflect useful physical relationships learned by the model.

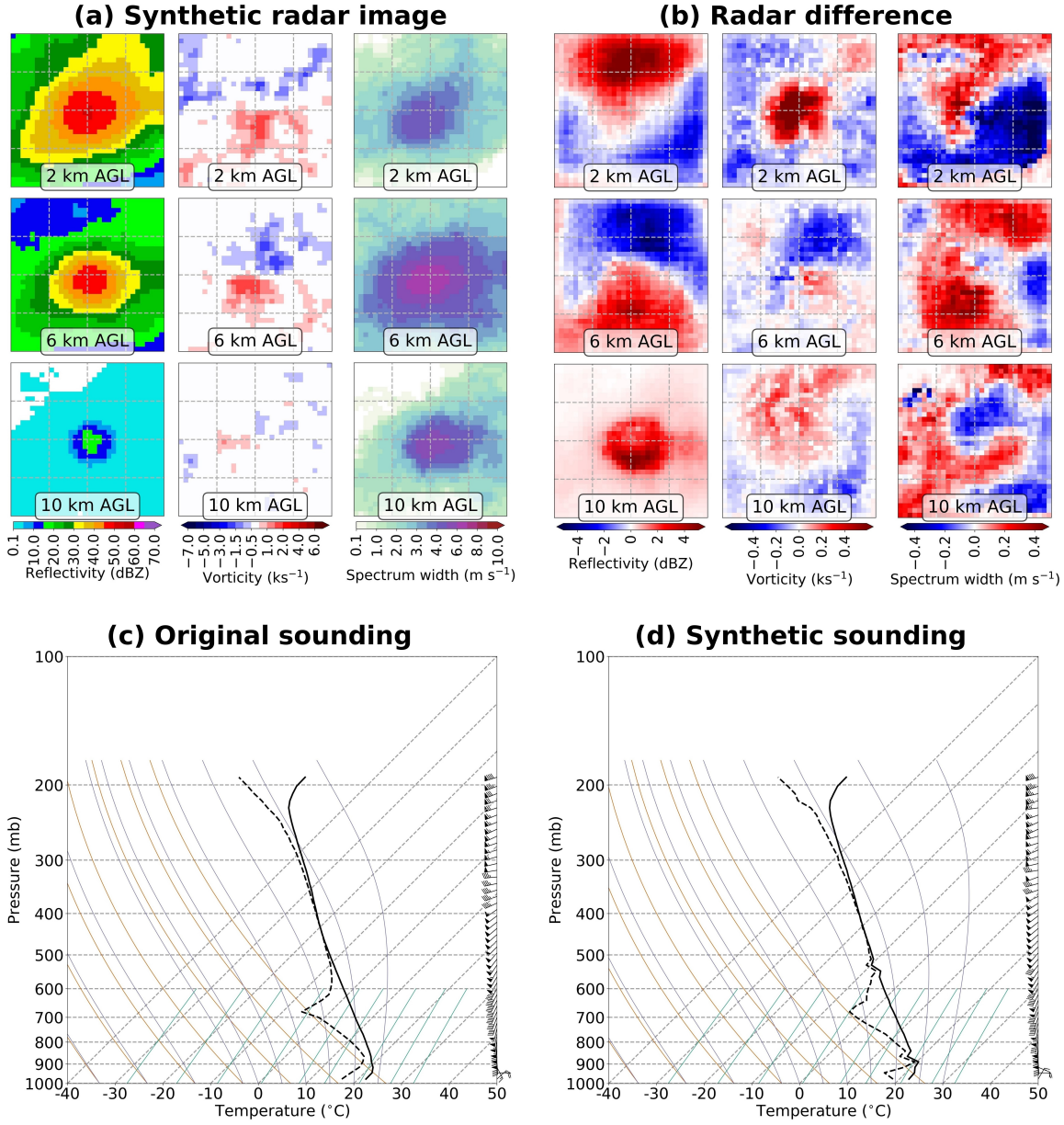


Figure 7.6: Backward optimization for worst misses in GridRad model. Storm motion points to the right, and each image is a PMM composite over 100 examples. BWO is applied individually to each storm, with the goal of increasing tornado probability. The original radar image (before optimization) is shown in Figure 6.12c. [a] Synthetic radar image, after optimization. [b] Difference (synthetic minus original radar image). [c] Original sounding, before optimization. [d] Synthetic sounding, after optimization.

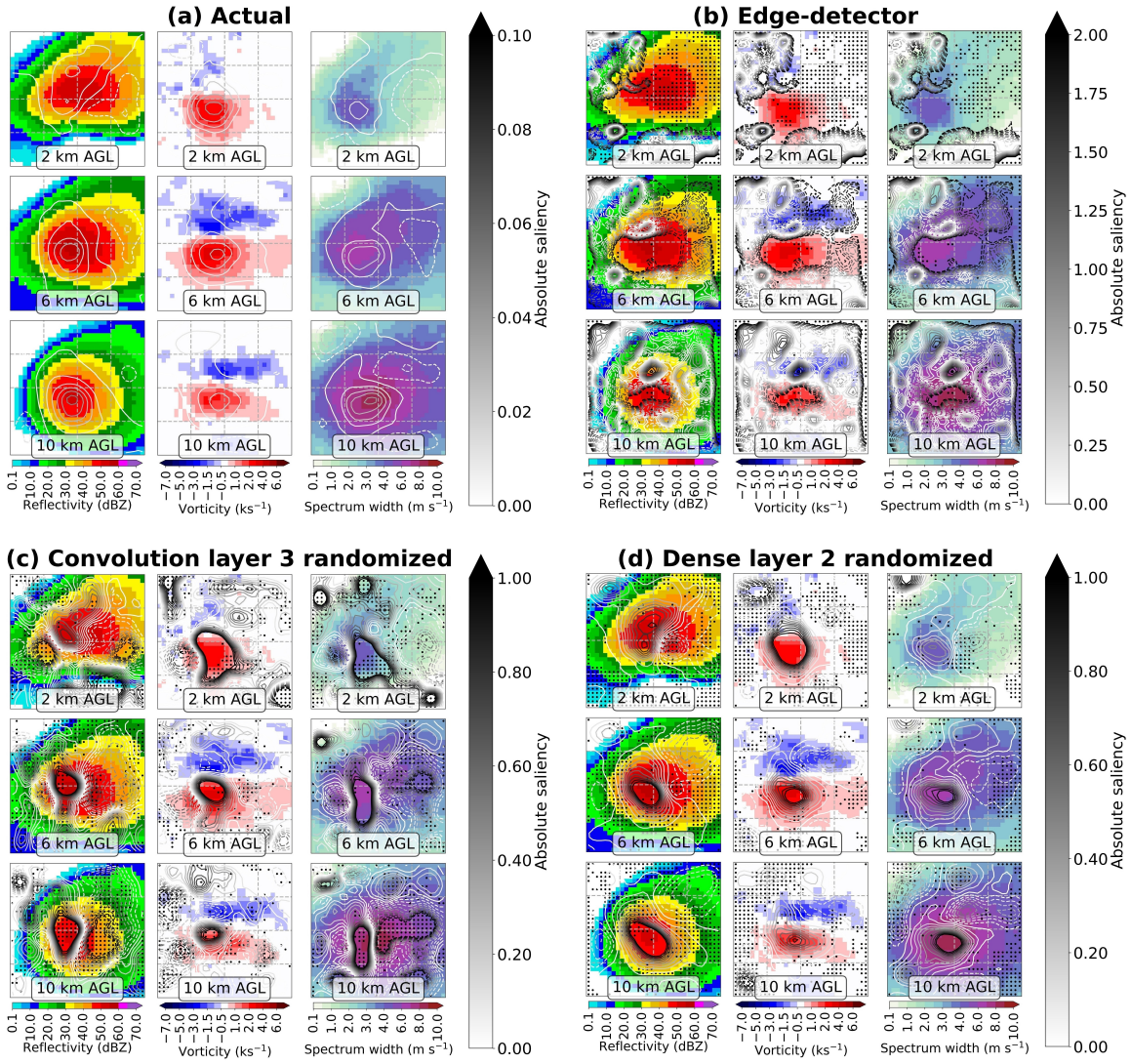


Figure 7.7: Sanity checks for saliency maps on best hits in GridRad model. Formatting is explained in Figure 7.2. [a] Actual saliency map (same as Figure 7.2a). [b] Saliency map produced by an untrained edge-detector. [c] Saliency map produced after randomizing weights in the third convolutional layer. [d] Saliency map produced after randomizing weights in the second dense layer. In panels b-d, stippling shows where the actual and dummy saliency maps are significantly different at the 95% confidence level.

7.2.2 MYRORSS Model

Figure 7.8 shows results of the permutation test, which measures the overall importance of each predictor variable. In this case the model performs 2-D convolution (Figure 5.3), so it treats reflectivity at each height as a separate variable ("channel" in ML jargon). This is why reflectivity is shuffled independently at each height for the MYRORSS model, while full 3-D maps are shuffled together for the GridRad model (*c.f.*, Figures 7.8 and 7.1). Since the MYRORSS permutation test is more computationally expensive, it uses 10 000 examples, whereas the GridRad permutation test uses 30 000.

For the MYRORSS model, all four versions of the test suggest that low-level azimuthal shear is the best predictor. For the two single-pass versions (Figures 7.8a and 7.8c), exact rankings after the top predictor are generally not significant, indicated by the lack of bold font. According to the backward multi-pass test (Figure 7.8d), the most important predictors after low-level shear are 4-km reflectivity, mid-level shear, 5-km reflectivity, 10-km reflectivity, and v-wind. Only the last of these is a sounding variable. The other versions of the test suggest that 4 and 10 km are the most important heights for reflectivity, which is broadly consistent with the backward multi-pass test. I speculate that 10-km reflectivity is important because the model is sensitive to storm depth (consistent with GridRad saliency maps; Section 7.2.1), while 4-km reflectivity is important because the model is sensitive to low-level storm structure. Although reflectivity is available at lower heights (down to 1 km), the permutation test suggests that 4-km reflectivity is a better predictor. This is likely because (a) some parts of the CONUS have poor radar coverage in the lowest few kilometres and (b) some storms are high-based, so their structures cannot be adequately parsed by measurements in the lowest few kilometres. All versions except the forward single-pass test (Figure 7.8a) suggest that radar variables are much more important than sounding variables, consistent with GridRad results (Figure 7.1). This is likely because the radar variables are highly correlated with each other (Figure 7.9), so other than low-level

azimuthal shear, which is a direct indicator of low-level rotation, permuting a single radar variable has little effect on model performance.

Figure 7.10 shows saliency maps for extreme cases. For the best hits (Figure 7.10a), saliency is maximized in the rear part of the reflectivity core. In this area tornado probability increases with low-level azimuthal shear, mid-level azimuthal shear, and reflectivity at all heights. Saliency for azimuthal shear is greater than for reflectivity; saliency for azimuthal shear is greatest at the low levels (0-2 km AGL); and saliency for reflectivity is greatest at the upper level (10 km AGL). Thus, like the GridRad model (Figure 7.2a), the MYRORSS model is particularly sensitive to low-level rotation and storm depth. There are two notable differences between the MYRORSS and GridRad saliency maps (*c.f.*, Figures 7.2 and 7.10). First, the MYRORSS model has negative saliency for low-level reflectivity behind the core, which appears to be in the rear-flank downdraft (RFD). Thus, tornado probability decreases with reflectivity in the RFD, possibly because higher reflectivity suggests more evaporative cooling, making the air too negatively buoyant to be ingested into the updraft (see discussion in Section 2.1.1). Second, the MYRORSS model has large negative saliency for low-level azimuthal shear just outside the mesocyclone, possibly because low-level convergence contracts the mesocyclone during tornadogenesis (Section 2.1.1). For the worst false alarms, the MYRORSS saliency map is very similar to the best hits (*c.f.*, Figures 7.10a and 7.10b) and therefore is not discussed further.

For the worst misses (Figure 7.10c), saliency for reflectivity is generally positive in the left-front quadrant of the core and negative in the right-rear quadrant, while saliency for azimuthal shear is generally positive inside the mesocyclone and negative outside. Positive saliency for reflectivity is greatest at 10 km AGL, while negative saliency for reflectivity is greatest at 2 km AGL, again suggesting that the model is particularly sensitive to storm depth and a low-level weak-echo region. As for the GridRad model (Figure 7.2c), changes suggested by the MYRORSS model would make the storms look more like supercells rather than QLCS cells. The main difference between the two models is that the GridRad model

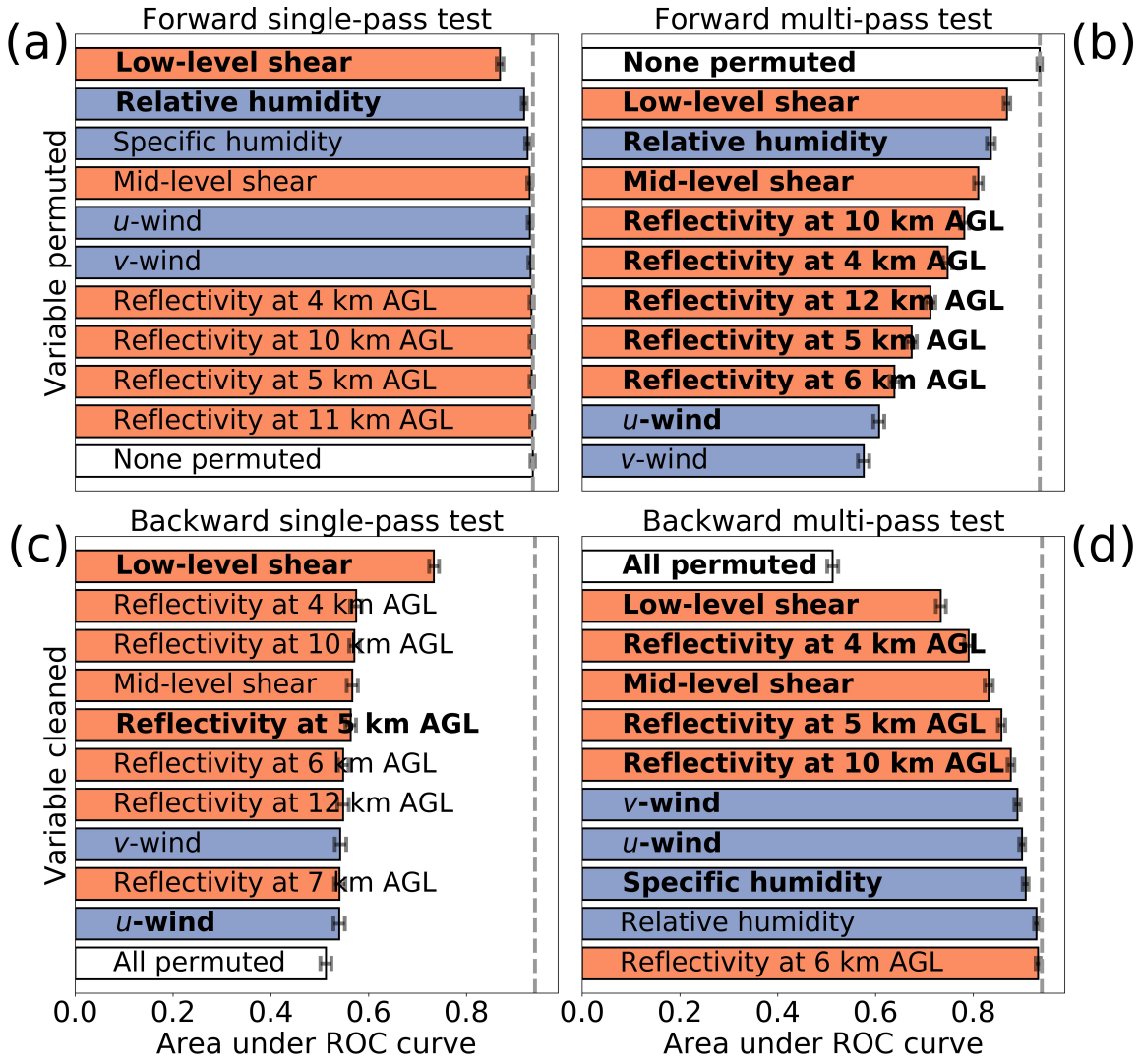


Figure 7.8: Results of permutation test for MYRORSS model, based on 10 000 randomly selected testing examples. The 10 000 examples are the same for all four versions of the test. Radar variables are in orange, and sounding variables are in purple. Grey error bars and bold font are explained in the caption of Figure 7.1.

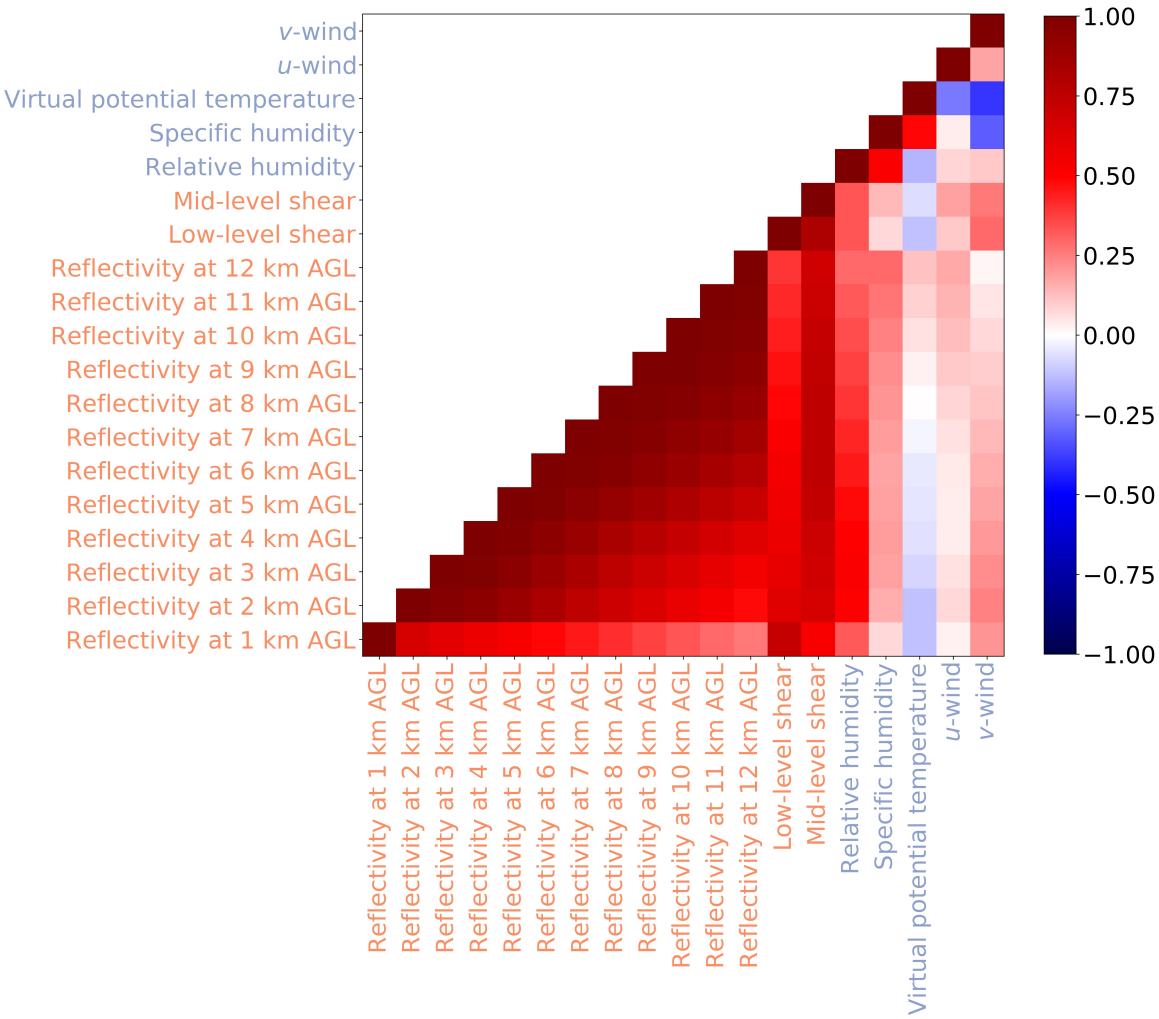


Figure 7.9: Pearson correlations in MYRORSS data, based on the same 10 000 examples used for the permutation test (Figure 7.8). Radar variables are in orange, and sounding variables are in purple. To compute the correlation between a sounding variable and radar variable, the spatial mean is first taken for each example, since the sounding and radar variables are on different grids.

has negative saliency around the edge of the image for all variables (favouring more discrete storms), while the MYRORSS model does not. Thus, the GridRad model is more sensitive to the discreteness of storms, while the MYRORSS model is more sensitive to the structure of the storms themselves. For the best correct nulls, the MYRORSS saliency map is similar to the worst misses (*c.f.*, Figures 7.10c and 7.10d), except that values are much lower. As for the GridRad model, this is because the best correct nulls look very unlike tornadic storms, so small perturbations can increase tornado probability by only a small amount.

Figure 7.11 shows class-activation maps (CAM), which quantify the amount of tornado evidence at each grid point, for extreme cases. These maps are based on the fifth convolutional layer, which is the deepest (Figure 5.3). Results are qualitatively similar to those for the GridRad model (Figure 7.3), unless otherwise noted. For the best hits and worst false alarms (Figure 7.11a-b), tornado evidence is maximized along the right flank and in the low-level weak-echo region. For the worst misses (Figure 7.11c), tornado evidence is maximized where large azimuthal shear coincides with large upper-level reflectivity. Tornado evidence decreases sharply outside this maximum, likely because the surrounding area reveals the storms' linear structure. For the best correct nulls (Figure 7.11d), non-zero tornado evidence roughly coincides with the maximum in azimuthal shear. As for the GridRad model (Figure 7.3d), these class activations are very small, likely because they are not accompanied by supercell-like reflectivity values of 35+ dBZ. Unlike the GridRad model, maximum tornado evidence is not exactly collocated with maximum azimuthal shear (vorticity in the GridRad model). There is an offset of several kilometres between the two maxima in Figure 7.11d (recall that the domain is 48 $\times$ 48 km). Analogous CAMs for the first four convolutional layers (Figure 7.12) help explain why. Deeper layers in the CNN have learned more strongly that the best correct nulls contain no tornado evidence, pushing areas of non-zero class activation to the outside of the domain. If there were a sixth convolutional layer, extrapolation of this trend suggests that the contours would disappear completely and the entire domain would contain exactly zero class activation.

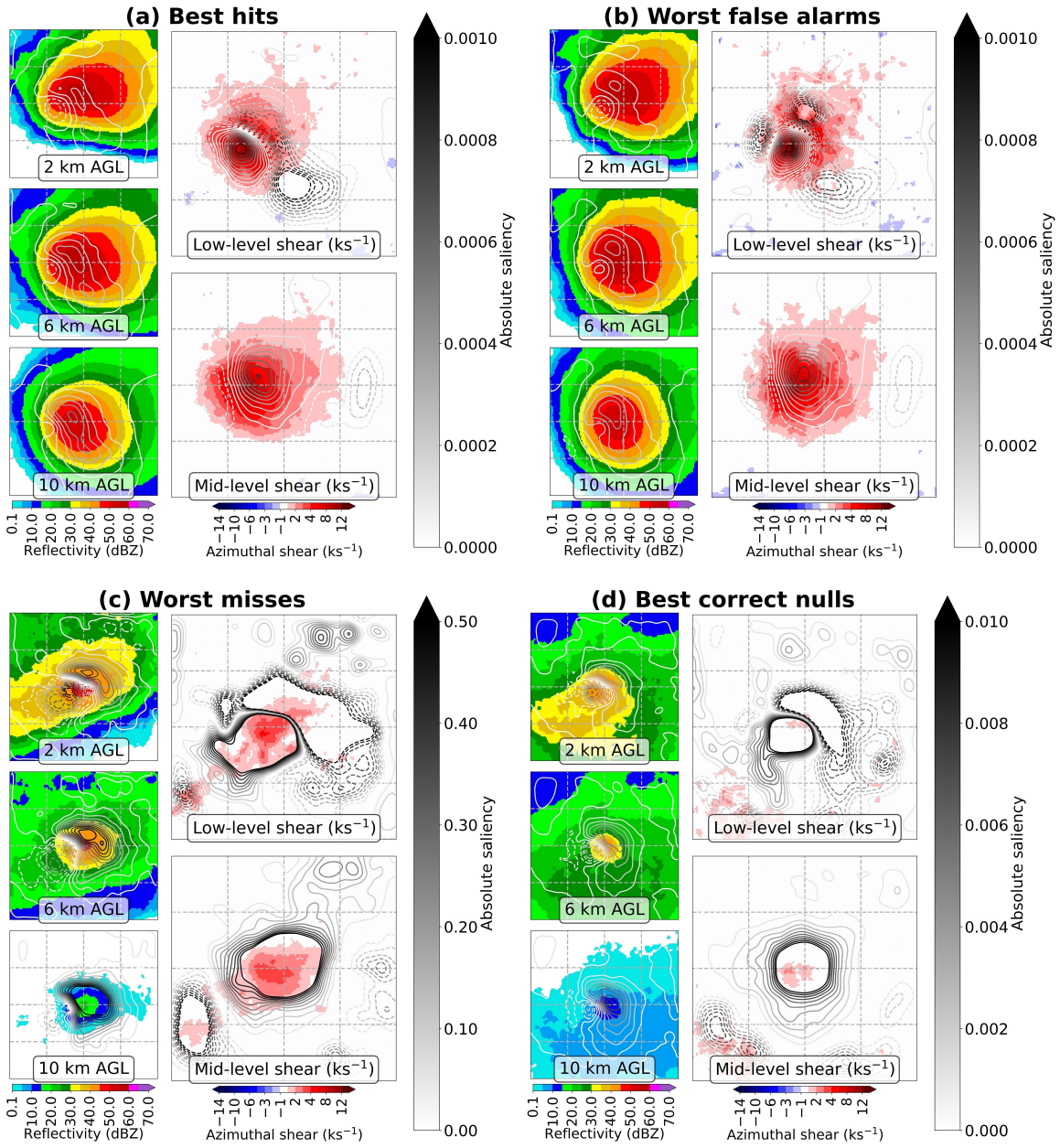


Figure 7.10: Saliency maps for extreme MYRORSS cases. Storm motion points to the right. Heat maps show the radar fields and are identical to those shown in Figure 6.13, while line contours show saliency. Solid contours indicate positive saliency (tornado probability increases with the underlying radar value), while dashed contours indicate negative saliency (tornado probability decreases with the underlying radar value). Each image (both the radar and saliency fields) is a PMM composite over 100 examples.

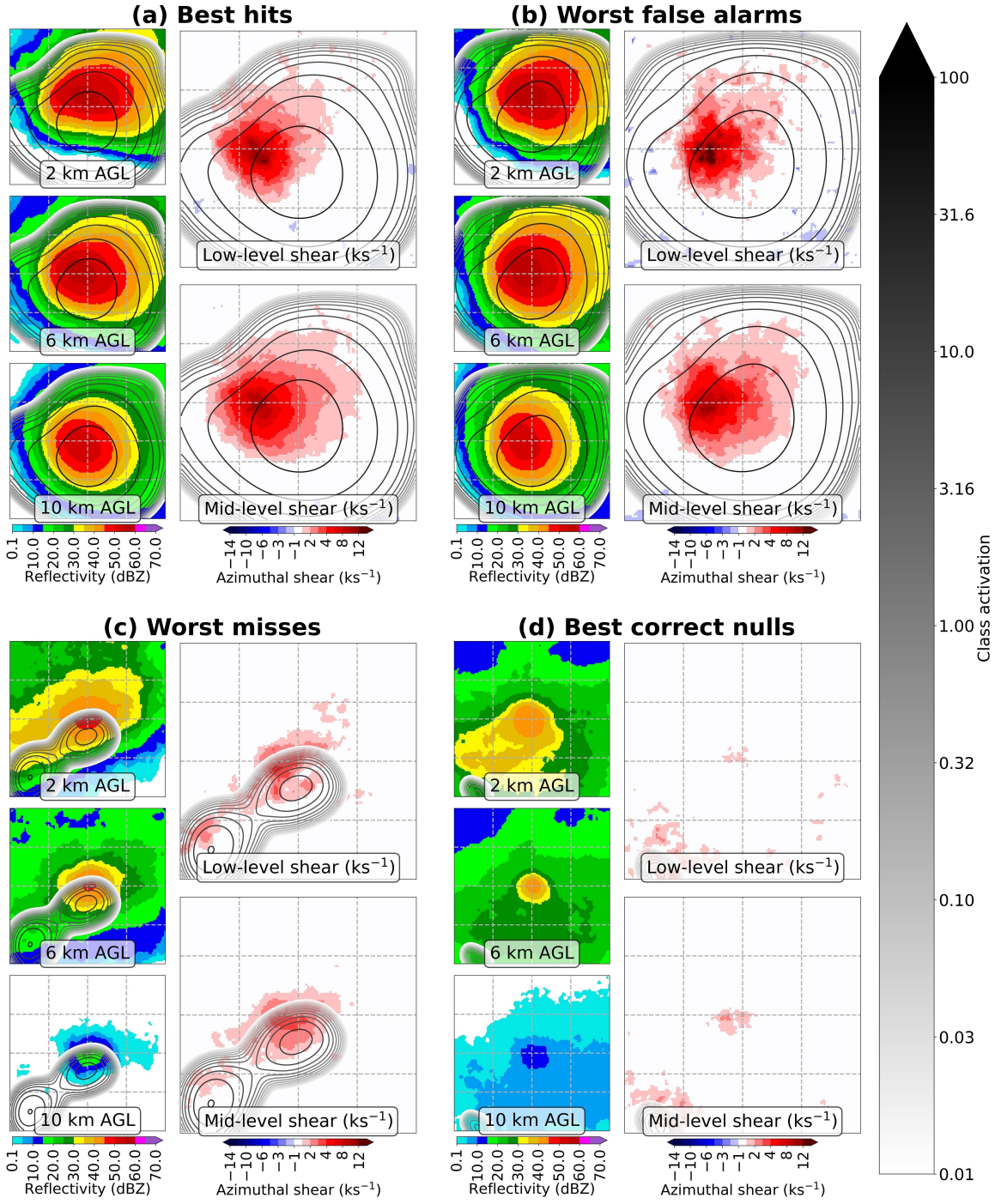


Figure 7.11: Class-activation maps for extreme MYRORSS cases. Storm motion points to the right. Heat maps show the radar fields and are identical to those shown in Figure 6.13, while line contours show class activation. Each image (both the radar and class-activation fields) is a PMM composite over 100 examples.

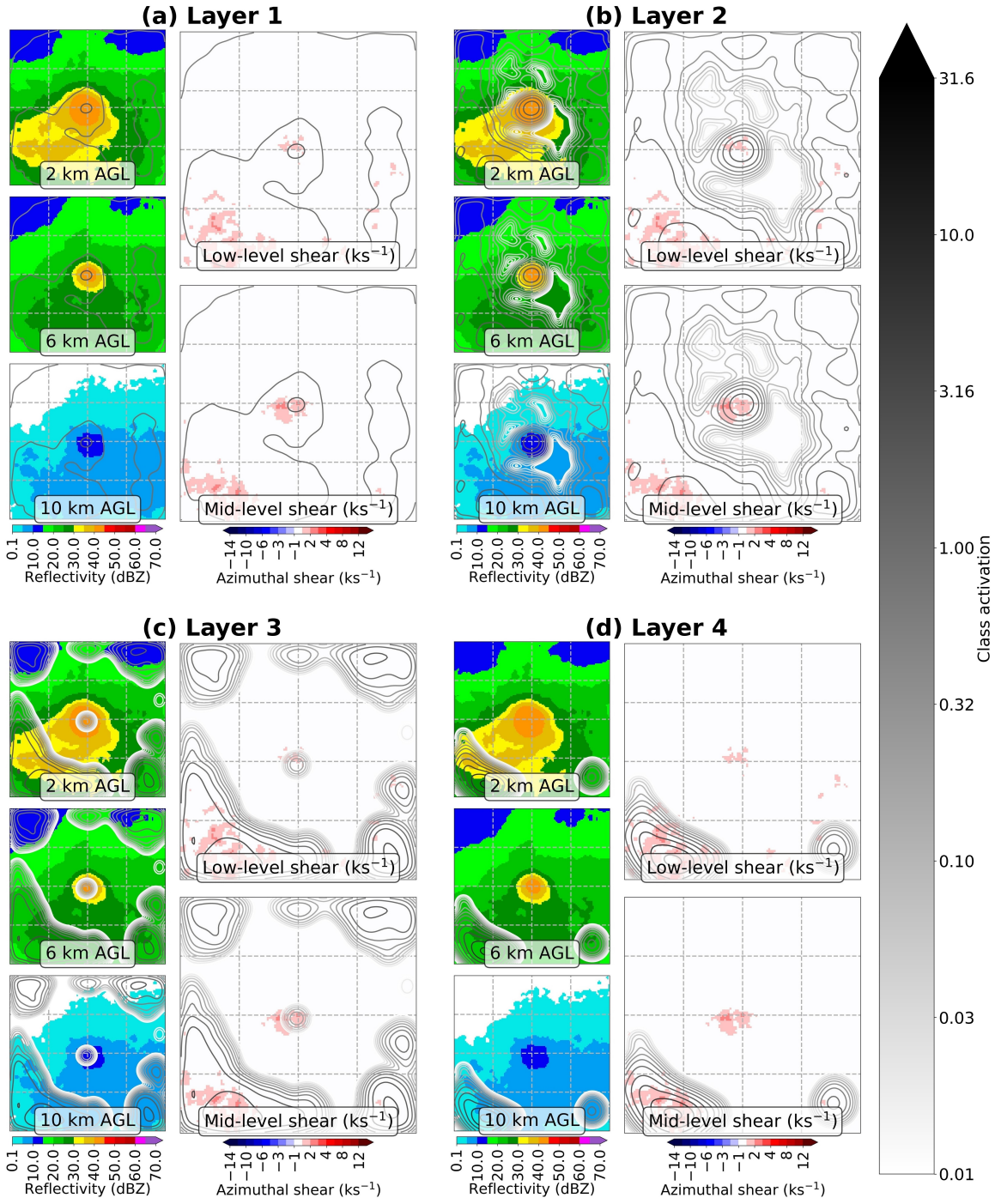


Figure 7.12: Class-activation maps for best correct nulls in MYRORSS model. Formatting is explained in Figure 7.11.

Figure 7.13 shows synthetic storms created by applying backward optimization (BWO) to the best hits, with the goal of decreasing tornado probability to 0.0. On average for the 100 storms, this decreases probability from 99.6% to 9.7%. In the reflectivity core and mesocyclone, BWO increases low-level reflectivity while decreasing reflectivity aloft, resulting in a weaker storm with a less distinct low-level weak-echo region. BWO also increases reflectivity around the edge of the storm, thus making it somewhat less discrete. BWO has a negligible effect on azimuthal shear, changing most values by no more than 4 $\times$ 10^{-4} ks^{-1} (4 $\times$ 10^{-7} s^{-1}). Meanwhile, in the proximity sounding, BWO creates a strong surface-based temperature inversion and decreases near-surface moisture, leading to strong capping and zero CAPE for surface-based convection. Unlike for the GridRad model (Figure 7.4), changes made by the MYRORSS model, particularly in the sounding, seem to justify the large decrease in tornado probability. However, the differences created by BWO are dominated by high-frequency noise or "checkerboard patterns," which are most prevalent in the difference images for azimuthal shear (Figure 7.13b) and the lower levels of the synthetic sounding (Figure 7.13d). Adding a smoothness constraint to the loss function (Equation 7.5) might alleviate this problem. However, the smoothness constraint would need to be applied carefully, because real-world data often contain sharp gradients, such as the dry layer at 700 mb in the original sounding (Figure 7.13c).

Figure 7.14 shows synthetic storms created by applying BWO to the worst misses, with the goal of increasing tornado probability to 1.0. On average for the 100 storms, this increases probability from 11.9% to 93.8%. In general, changes are similar to those made for the best hits but opposite in sign. BWO decreases low-level reflectivity in the core, increases reflectivity aloft in the core, and negligibly impacts azimuthal shear. In the proximity sounding, BWO increases temperature and moisture below 900 mb and increases wind speed from 700-800 mb, where there was originally a speed minimum. In changing the low-level temperature profile, BWO creates a strong inversion, which is not conducive to surface-based convection. However, for convection based at $\sim$950 mb, CAPE and shear

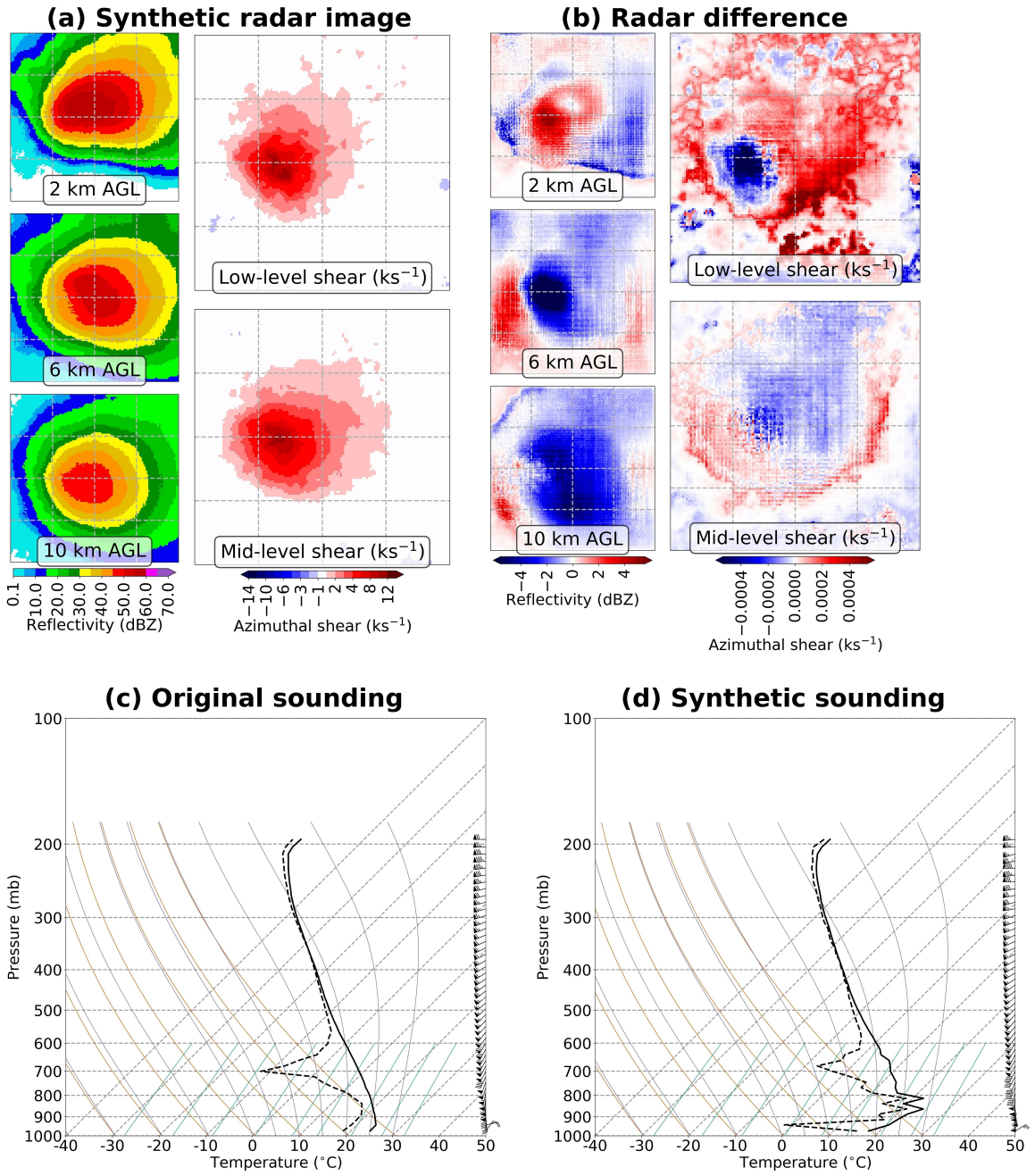


Figure 7.13: Backward optimization for best hits in MYRORSS model. Storm motion points to the right, and each image is a PMM composite over 100 examples. BWO is applied individually to each storm, with the goal of decreasing tornado probability. The original radar image (before optimization) is shown in Figure 6.13a. [a] Synthetic radar image, after optimization. [b] Difference (synthetic minus original radar image). [c] Original sounding, before optimization. [d] Synthetic sounding, after optimization.

would be extremely high, allowing for explosive storm development and possibly tornadogenesis.

Figures 7.15 and B.5-B.6 show sanity checks for saliency maps on all extreme cases except the worst false alarms, for which results are very similar to the best hits. The model-parameter-randomization test (Section 7.1.2) is shown for only the fifth convolutional layer and second dense layer (Figure 5.3). For the best hits, dummy saliency maps (Figure 7.15b-d) have much greater values than the actual saliency map (Figure 7.15a), although the stippling shows significant differences between rescaled values. The model-parameter-randomization test (Figure 7.15c-d) leads to significant differences for 27% and 37% of scalar predictors, which suggests that saliency maps truly depend on learned weights in each of the two layers. Meanwhile, the edge-detector test (Figure 7.15b) leads to significant differences for only 5% of scalar predictors. As for the GridRad model (Figure 7.7b), while this suggests that the shapes of saliency maps can be mostly replicated by an untrained edge-detector, the actual (unscaled) values cannot be replicated. Specifically, the edge-detector greatly overestimates saliency, as shown by differences between the colour scales Figures 7.15a and 7.15b. Results for the worst misses (Figure B.5) and best correct nulls (Figure B.6) are similar, with the edge-detector test yielding significant differences for $\sim$3% of scalar predictors and the model-parameter-randomization test yielding significant differences for $\sim$20%. As for the GridRad model, these results suggest that MYRORSS saliency maps cannot be trivially replicated.

7.3 Front Detection

This section shows interpretation results for the front-detection model (Figure 4.1). As in Sections 7.2.1 and 7.2.2, the permutation test, class-activation maps, and backward optimization are discussed in detail. However, saliency maps are discussed only briefly at the end, because (a) saliency maps for a vector field such as wind are difficult to understand

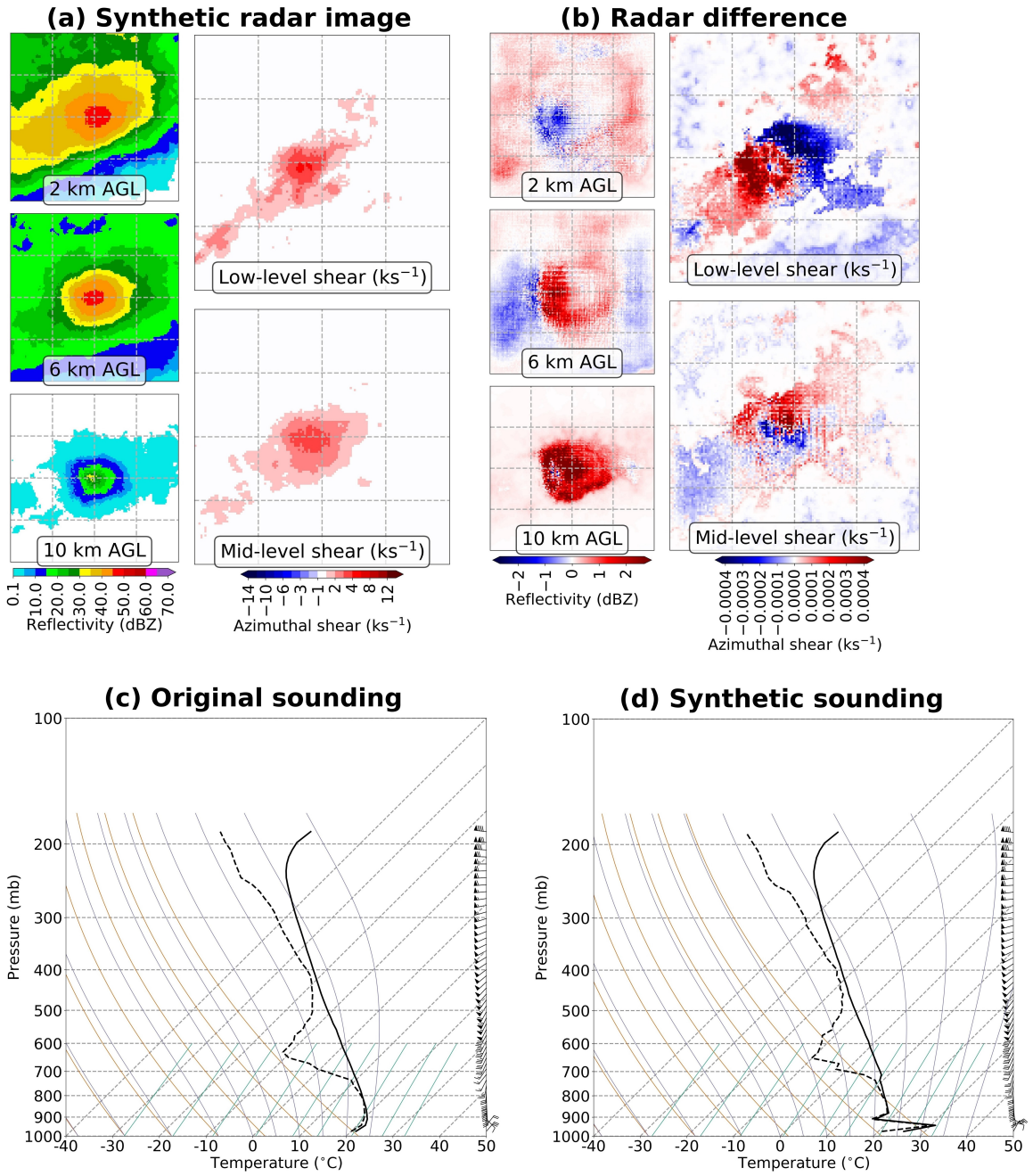


Figure 7.14: Backward optimization for worst misses in MYRORSS model. Storm motion points to the right, and each image is a PMM composite over 100 examples. BWO is applied individually to each storm, with the goal of increasing tornado probability. The original radar image (before optimization) is shown in Figure 6.12c. [a] Synthetic radar image, after optimization. [b] Difference (synthetic minus original radar image). [c] Original sounding, before optimization. [d] Synthetic sounding, after optimization.

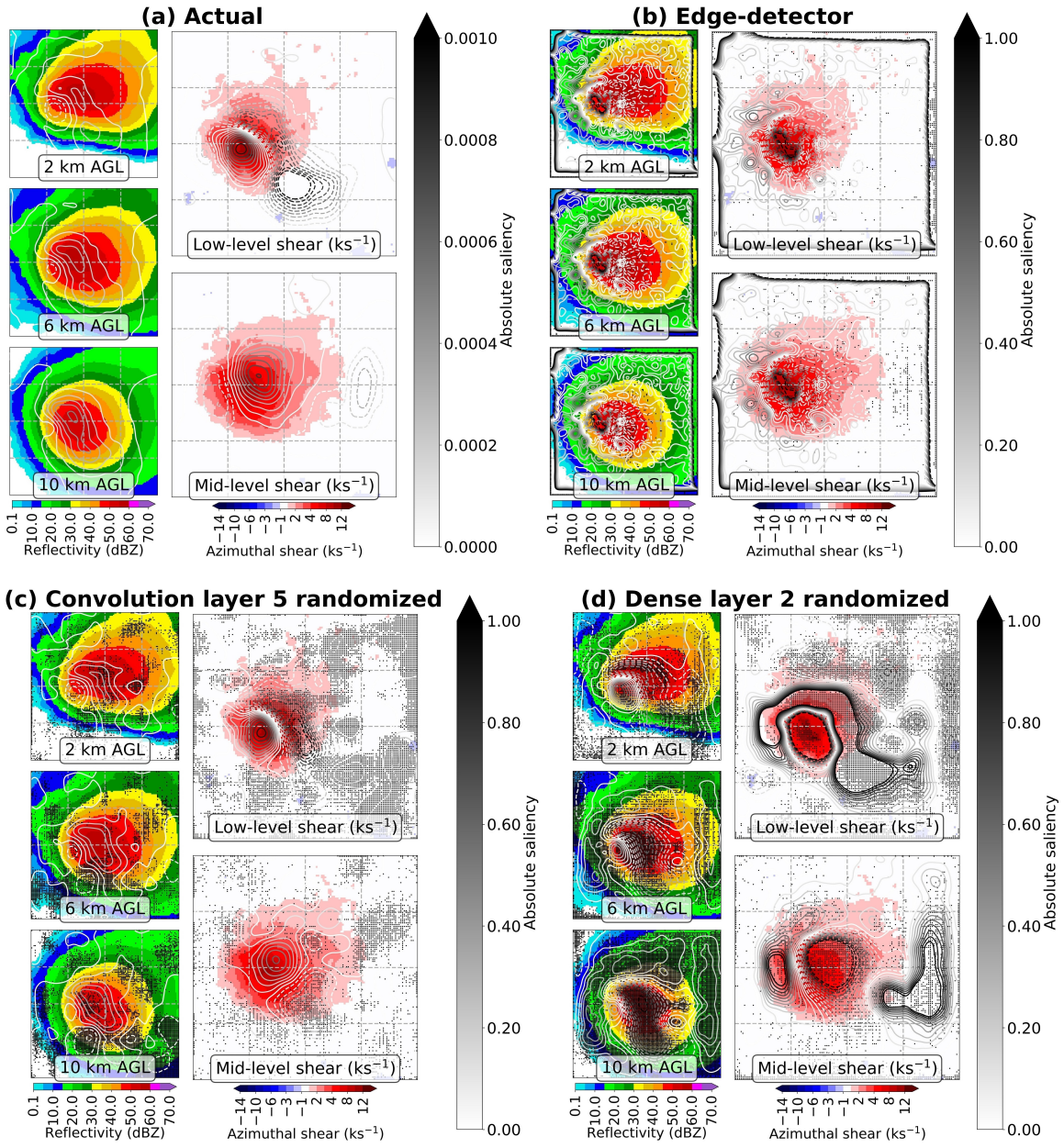


Figure 7.15: Sanity checks for saliency maps on best hits in MYRORSS model. Formatting is explained in Figure 7.10. [a] Actual saliency map (same as Figure 7.10a). [b] Saliency map produced by an untrained edge-detector. [c] Saliency map produced after randomizing weights in the fifth convolutional layer. [d] Saliency map produced after randomizing weights in the second dense layer. In panels b-d, stippling shows where the actual and dummy saliency maps are significantly different at the 95% confidence level.

and (b) backward optimization, which is a non-linear generalization of saliency maps, tells a similar story more cleanly.

Figure 7.16 shows results of the permutation test. As for the tornado models, the cost function is negative area under the ROC curve (AUC). However, the front-detection model uses multiclass AUC, while the tornado models use binary AUC. Multiclass AUC requires much more computing time, so the permutation test for the front-detection model does not include error bars, which require bootstrapping the AUC many times.

All four versions of the test agree that the top predictor is surface v-wind, followed by surface u-wind and 850-mb temperature, though not necessarily in that order. The forward multi-pass test (Figure 7.16b) suggests that surface u-wind is the second-best predictor, while the other versions suggest that 850-mb temperature is second-best. The reason for this discrepancy is not apparent, since the top three predictors have little interdependence according to the Pearson correlation (Figure 7.17). In general, specific humidity is the least important predictor. This is partly because specific humidity is strongly correlated with temperature (Figure 7.17), which is the better thermal predictor, thus diminishing the apparent importance of specific humidity. However, in the backward multi-pass test (Figure 7.16d), even though surface and 850-mb humidity are the least important predictors, model performance still improves notably when the two predictors are cleaned. This contrasts with results for the tornado models (Figures 7.1d and 7.8d), where performance improves very little as the last few predictors are cleaned. Although Figure 7.16 has no error bars, I believe that these results are significant, since permutation tests for the tornado models have small error bars (Figures 7.1d and 7.8d) and this test is based on more examples (50 000 versus 30 000 or 10 000).

Figures 7.18-7.20 show class-activation maps (CAM) for the extreme cases defined in Table 6.2. All CAMs are produced by the fourth convolutional layer, which is the last (Figure 4.1). Figure 7.18 shows CAMs for extreme warm fronts, quantifying the amount of

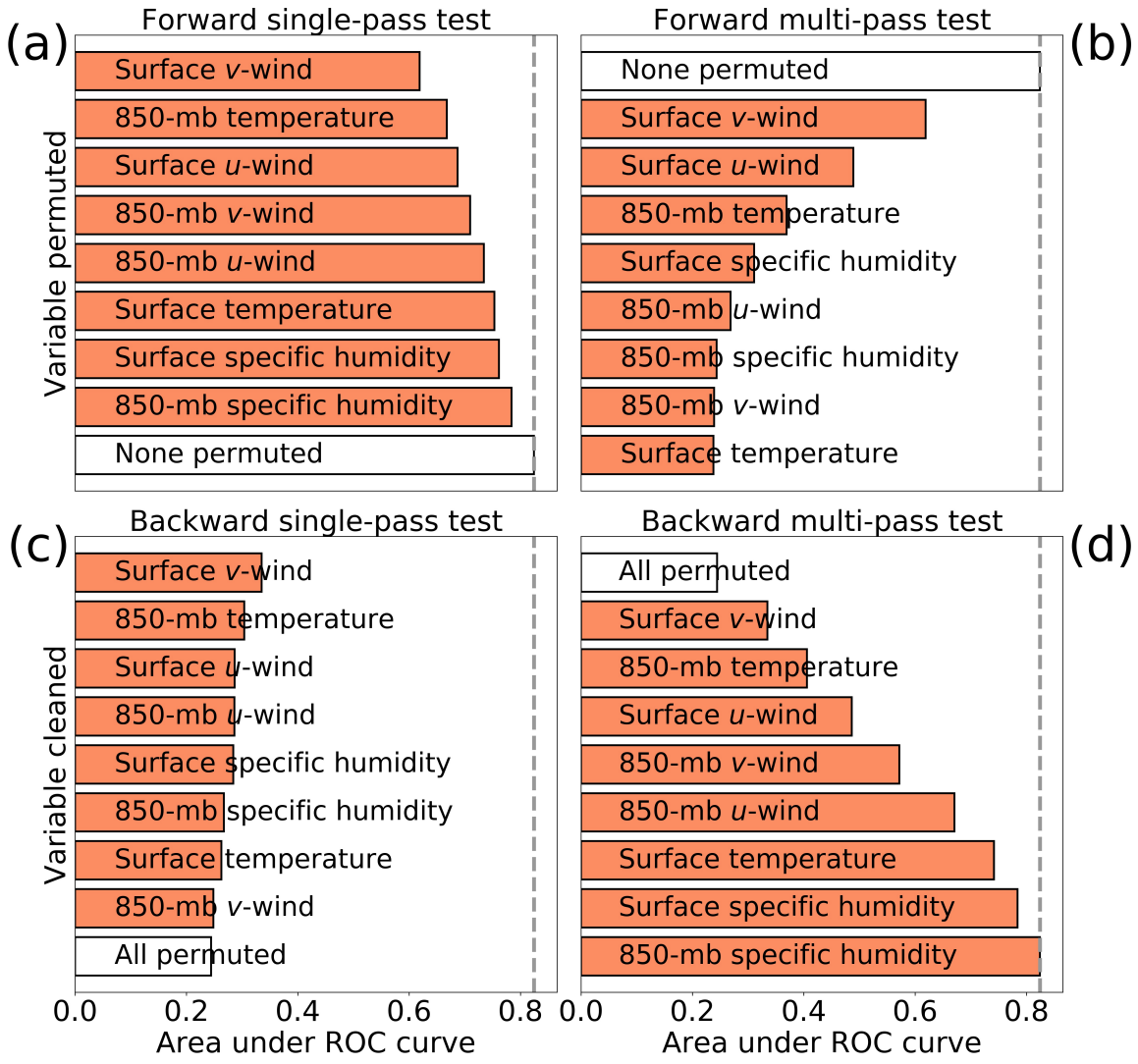


Figure 7.16: Results of permutation test for front-detection model, based on 50 000 randomly selected testing patches. The 50 000 patches are the same for all four versions of the test.

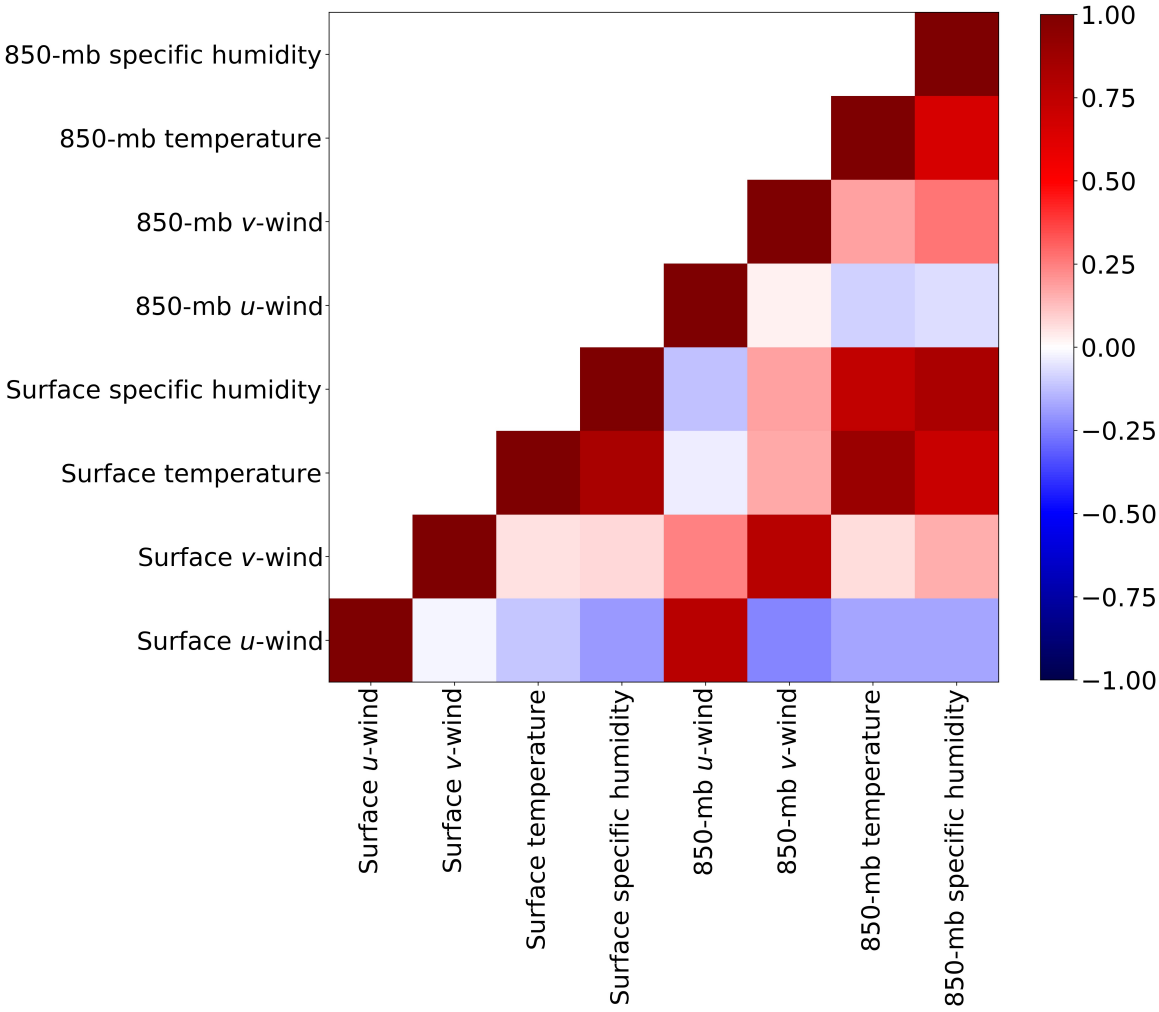


Figure 7.17: Pearson correlations among predictors for front detection, based on the same 50 000 examples used for the permutation test (Figure 7.16).

warm-front evidence at each grid point. For the best hits (Figure 7.18a), evidence is maximized with the clearest frontal features at 850 mb: a strong temperature gradient, strong humidity gradient, and abrupt wind shift from southerly in the warm sector to easterly ahead of the warm front. Meanwhile, for the worst misses (Figure 7.18b), warm-front evidence is maximized in the northwest corner of the domain, where there is warm and moist advection at 850 mb. However, the model produces very low warm-front probabilities for these cases (Table 6.2), because there is little to no thermal advection near the center grid cell. Visual inspection suggests that in many cases the WPC front is misplaced (should not pass through the center grid cell), so the model is justified in producing low probabilities for these cases. This analysis, together with Figure 7.18b, shows that not only can the model correctly identify the absence of a warm front in the center, but it can also identify the part of the domain most likely to contain a warm front. This provides further evidence that the model has overcome errors in its training data.

Figure 7.19 shows CAMs for extreme cold fronts, quantifying the amount of cold-front evidence at each grid point. For the best hits (Figure 7.19a), the strongest evidence is collocated with the strongest gradients in surface wind, surface temperature, surface humidity, and 850-mb humidity. Moving away from the front, evidence decreases more sharply in the warm sector than in the cold sector, likely because the cold sector is dominated by cold-air advection from the northwest, which occurs frequently behind cold fronts and is therefore an indirect cold-front signature. Meanwhile, for the worst misses (Figure 7.19b), cold-front evidence is maximized in the southeast corner, where there is cyclonic wind along with cold and dry advection. As for warm fronts, visual inspection of the worst misses suggests that often the WPC front is misplaced and should not pass through the center grid cell. Thus, as in the worst misses for warm fronts (Figure 7.18b), the model correctly identifies the absence of a front in the center and identifies the part of the domain most likely to contain a front.

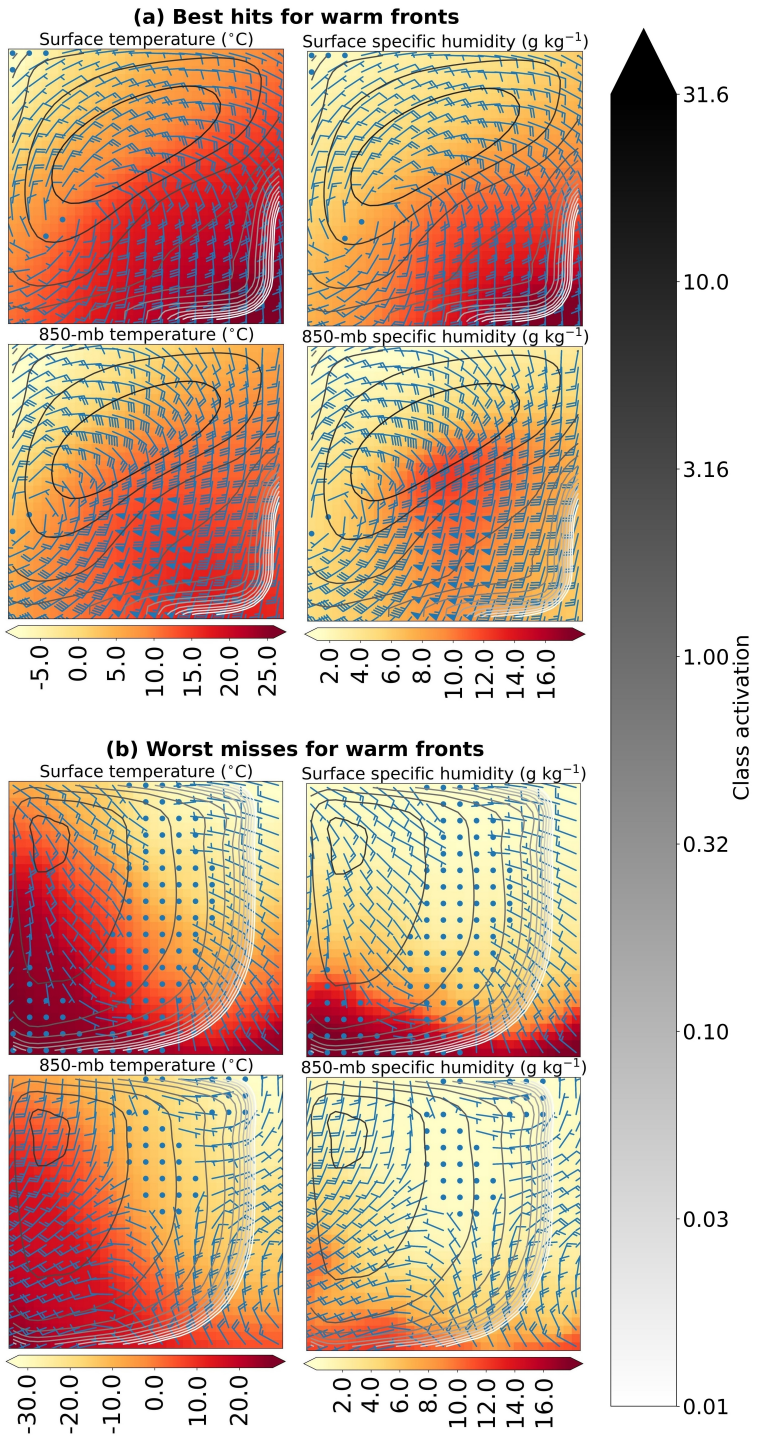


Figure 7.18: Class-activation maps for extreme warm fronts. Heat maps and wind barbs show predictors and are identical to those shown in Figure 6.19, while line contours show class activation. Each image (predictor and class-activation fields) is a PMM composite over 1000 examples.

Note that CAMs for warm-front patches are best aligned with frontal features at 850 mb (Figure 7.18), while CAMs for cold fronts are best aligned with frontal features at the surface (Figure 7.19). This suggests that vertical continuity is more important for warm fronts, probably because they tend to have weaker thermal gradients.

Figure 7.20 shows CAMs for the best correct nulls, quantifying the amount of warm- or cold-front evidence at each grid point. In the warm-front CAM (Figure 7.20a), although values are low, evidence is maximized on the western edge, where there is warm advection at both the surface and 850 mb. In the cold-front CAM (Figure 7.20b), evidence is maximized on the northern and eastern edges, but this area has little to no cold or dry advection. This is due to the same effect as the class-activation maximum in Figure 7.11d, which is curiously not collocated with the vorticity maximum. Deeper layers in the CNN learn more strongly that the best correct nulls contain no evidence for the positive class, pushing areas of non-zero class activation towards the edge of the image (Figure 7.12).

Figure 7.21 shows synthetic data created by applying backward optimization (BWO) to the best hits for cold fronts, with the goal of increasing no-front probability to 1.0. Average no-front probability for the 1000 cases thereby increases from 0.15% to 94.1%. BWO decreases thermal quantities in the center of the domain, especially surface humidity and 850-mb temperature; increases thermal quantities in a ring around the center; and makes the surface wind more anticyclonic in a ring around the center. This last change has two effects. First, it decreases the wind speed, thus decreasing thermal advection. Second, most fronts in the training data are associated with a cyclone (as discussed in Section 6.2.1), so making the wind more anticyclonic makes the data look less similar to fronts seen previously by the model. BWO changes the 850-mb temperature field more than the surface temperature field, which is consistent with the permutation test (Figure 7.16) suggesting that temperature is more important at 850 mb. Also, BWO changes the surface wind field more than the 850-mb wind field, consistent with the permutation test suggesting that surface wind is more important.

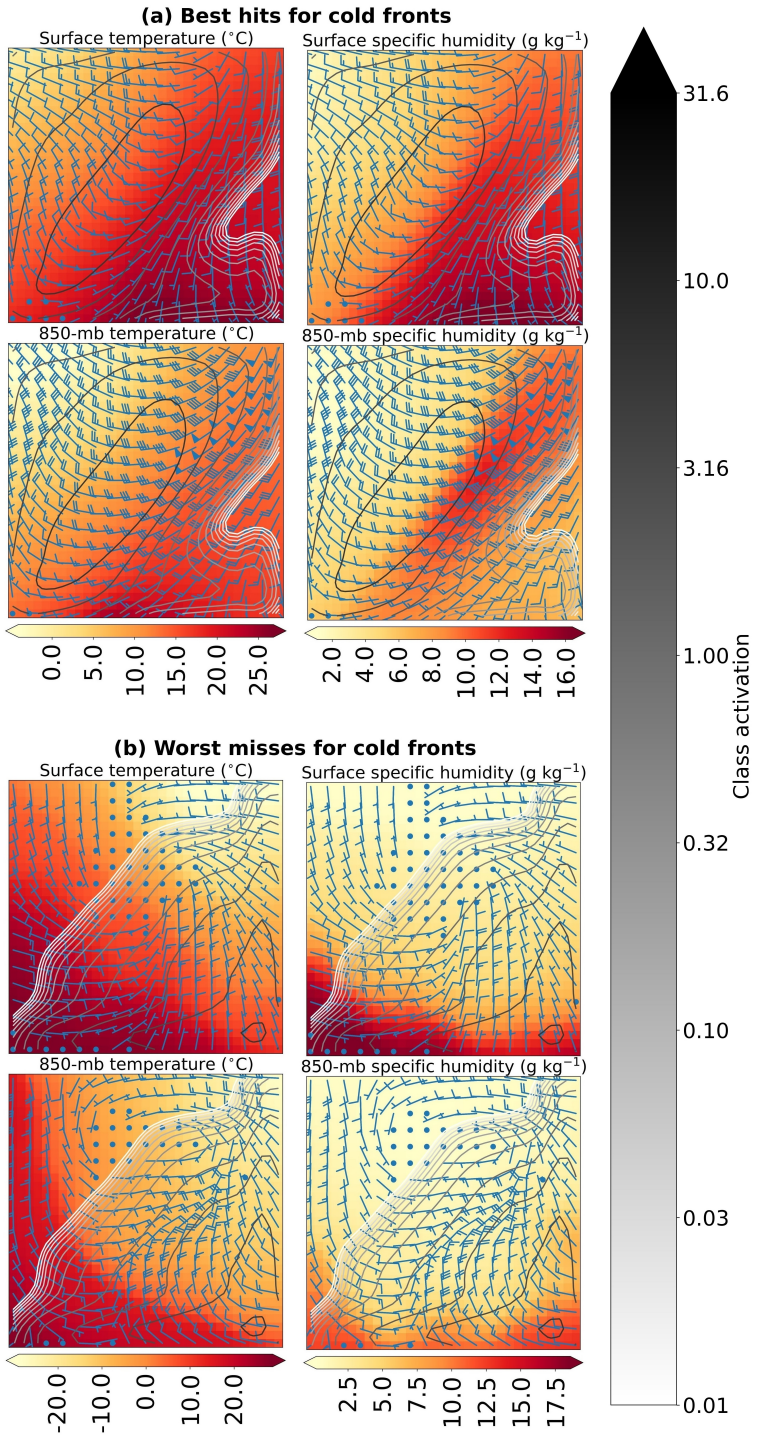


Figure 7.19: Class-activation maps for extreme cold fronts. Heat maps and wind barbs show predictors and are identical to those shown in Figure 6.20. Other formatting is explained in Figure 7.18.

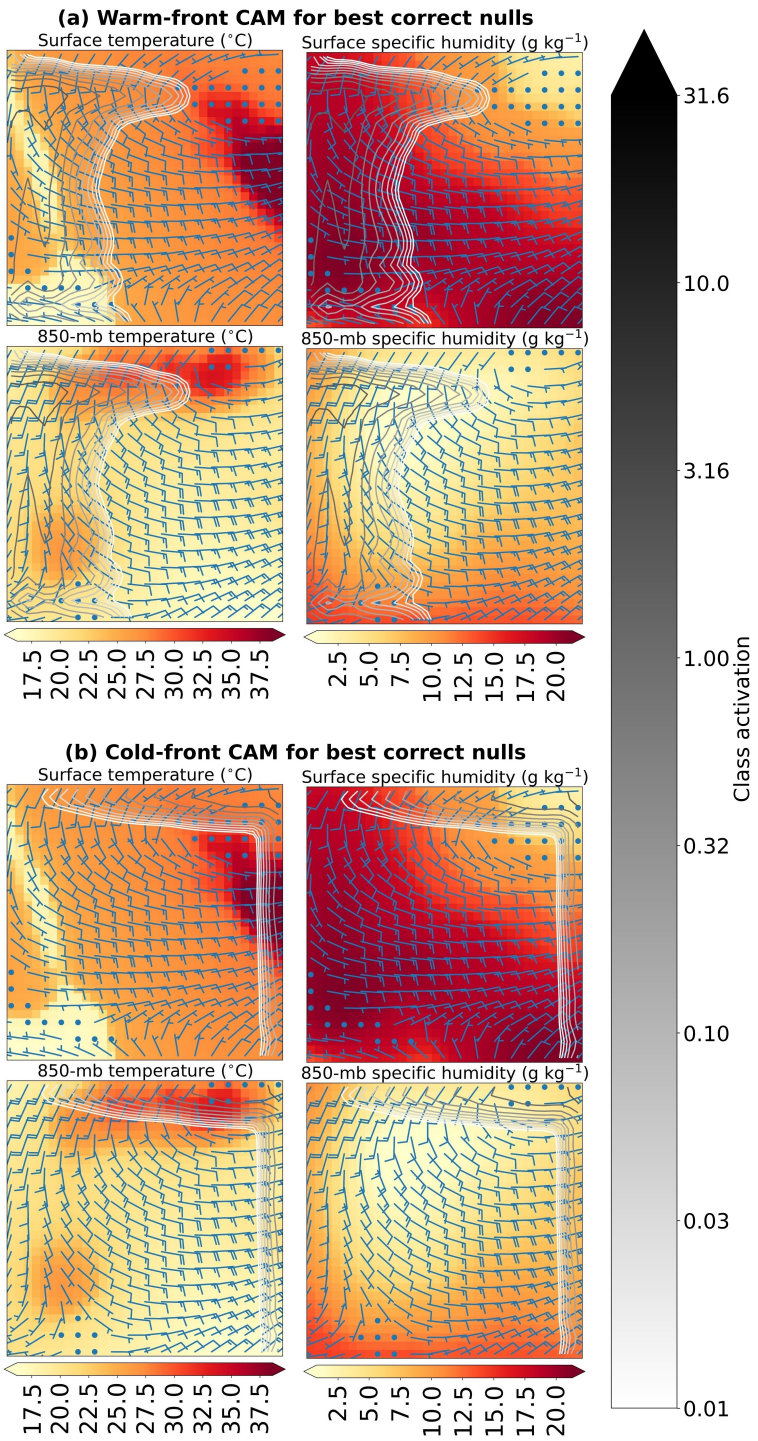


Figure 7.20: Class-activation maps for best correct nulls. Heat maps and wind barbs show predictors and are identical to those shown in Figure 6.21. Other formatting is explained in Figure 7.18.

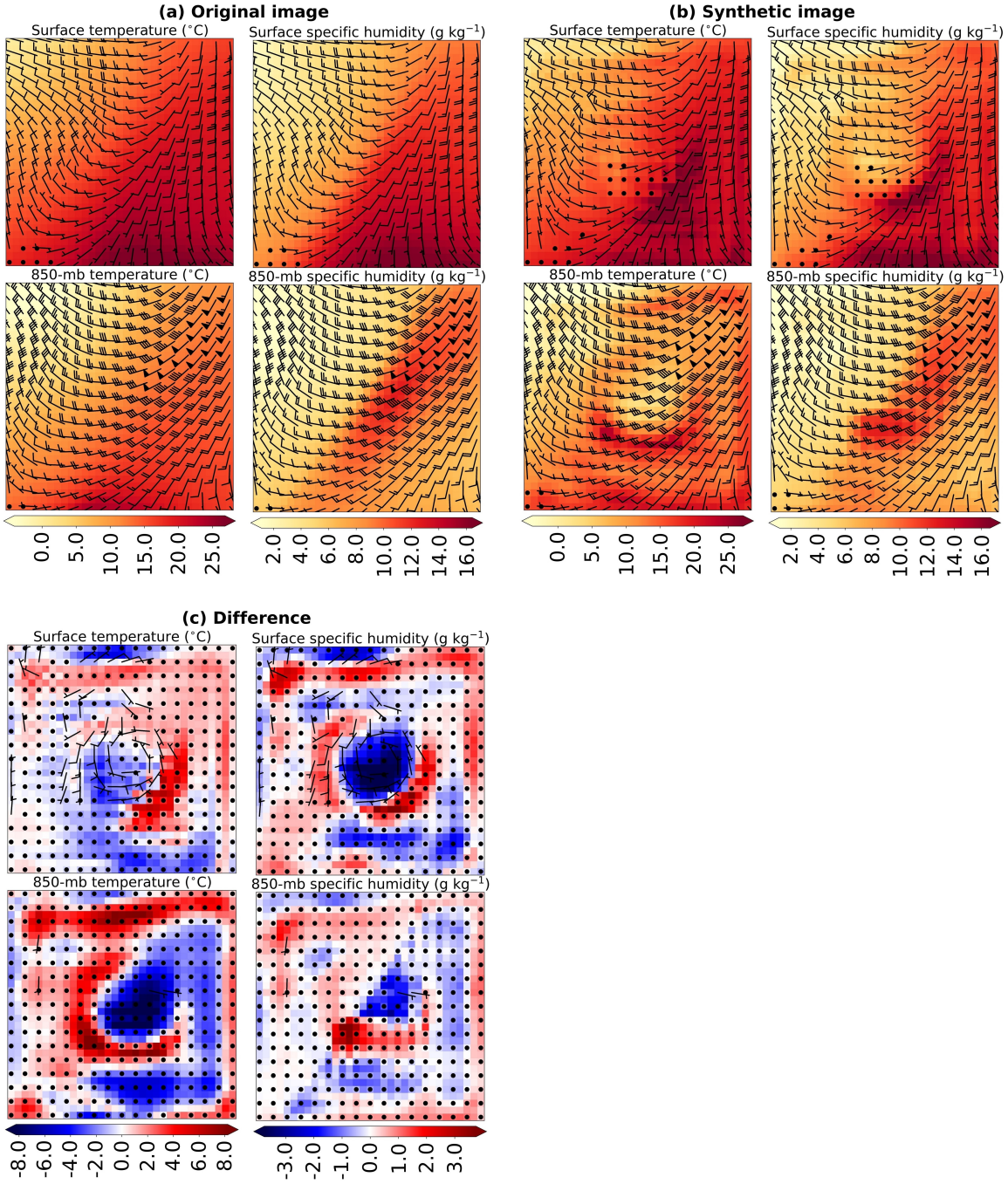


Figure 7.21: Backward optimization on best hits for cold fronts. Each image is a PMM composite over 1000 examples. BWO is applied individually to each example, with the goal of increasing no-front probability. [a] Original image, before optimization. [b] Synthetic image, after optimization. [c] Difference (synthetic minus original image).

Figure 7.22 shows synthetic data created by applying BWO to the worst misses for cold fronts, with the goal of increasing cold-front probability to 1.0. Average cold-front probability for the 1000 cases thereby increases from 0.38% to 93.5%. BWO increases thermal quantities in the center of the domain, decreases thermal quantities just north of center, and creates northerly wind in this area, generating cold and dry advection from the north. BWO also decreases thermal quantities just south of center and increases southerly wind in this area, generating cold and dry advection from the south. Thus, the tendency of BWO is to create two converging cold fronts, which is not physically realistic. Despite this, the synthetic image (Figure 7.22b) has only one apparent front, because changes in the southern area are not large enough to reverse the southerly wind here. However, the length and intensity of the synthetic cold front (very short with very strong thermal gradients) are unrealistic. Like some patterns found in synthetic thunderstorms (Section 7.2), this suggests that more constraints are needed to create fully realistic weather data.

Figure 7.23 shows sanity checks for one set of saliency maps produced by the model.[7] These are cold-front saliency maps on the 1000 best hits for cold fronts. As for the tornado models, dummy saliency maps (Figure 7.23b-d) greatly overestimate raw saliency values, but the stippling shows significant differences between rescaled values. The edge-detector test (Figure 7.23b) leads to significant differences for 80% of scalar predictors, while the model-parameter-randomization test (Figure 7.23c-d) leads to 87% and 90% significant differences for the two layers. Thus, all sanity checks suggest that the actual saliency maps (Figure 7.23a) truly reflect relationships learned by the CNN. Percentage of significant differences is similar for the other sets of extreme cases (Table 6.2).

[7]The wind fields are omitted, because plotting saliency for wind requires plotting the u- and v-components separately, which is non-intuitive and doubles the number of panels.

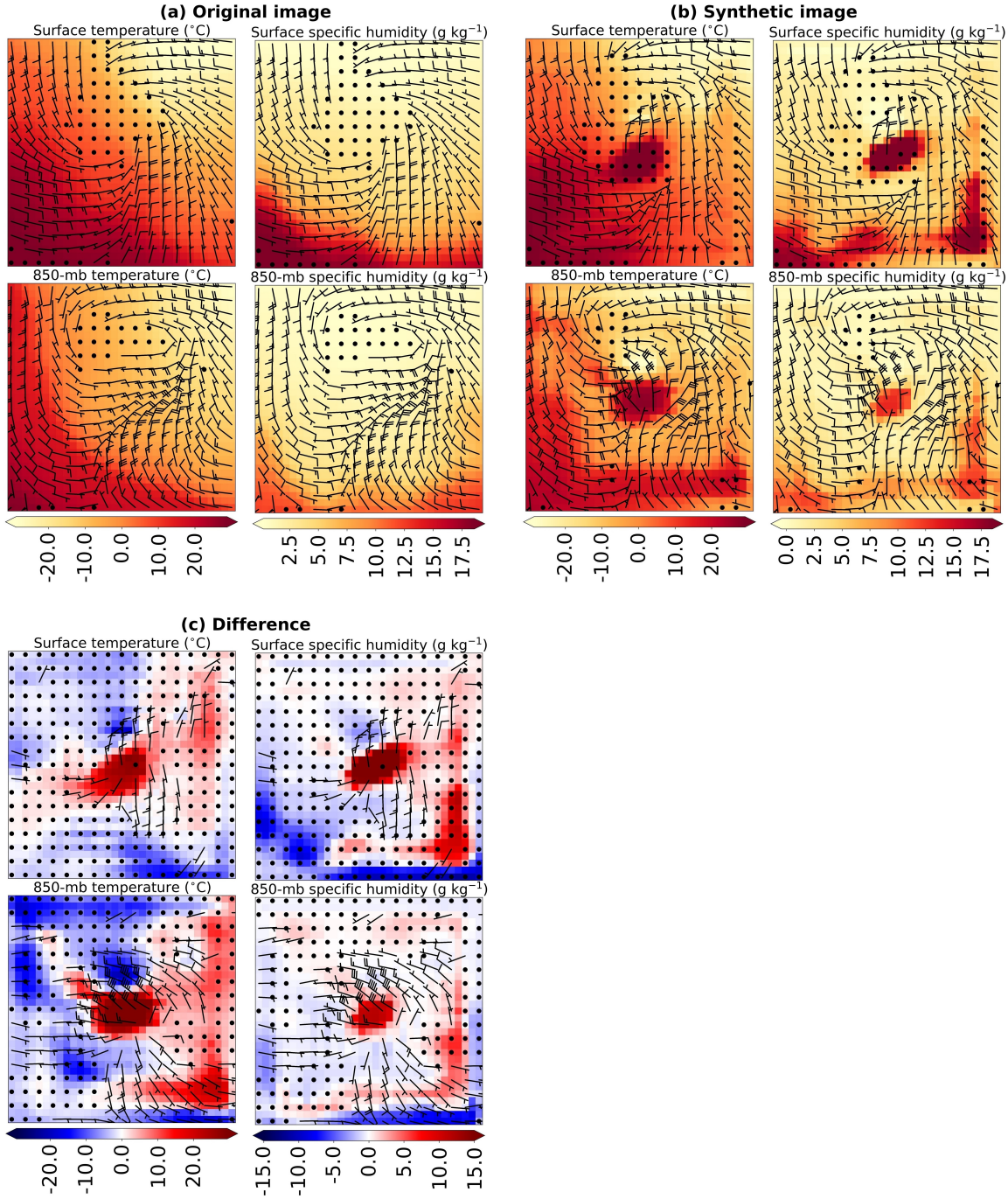


Figure 7.22: Backward optimization on worst misses for cold fronts. BWO is applied individually to each example, with the goal of increasing cold-front probability. Formatting is explained in Figure 7.21.

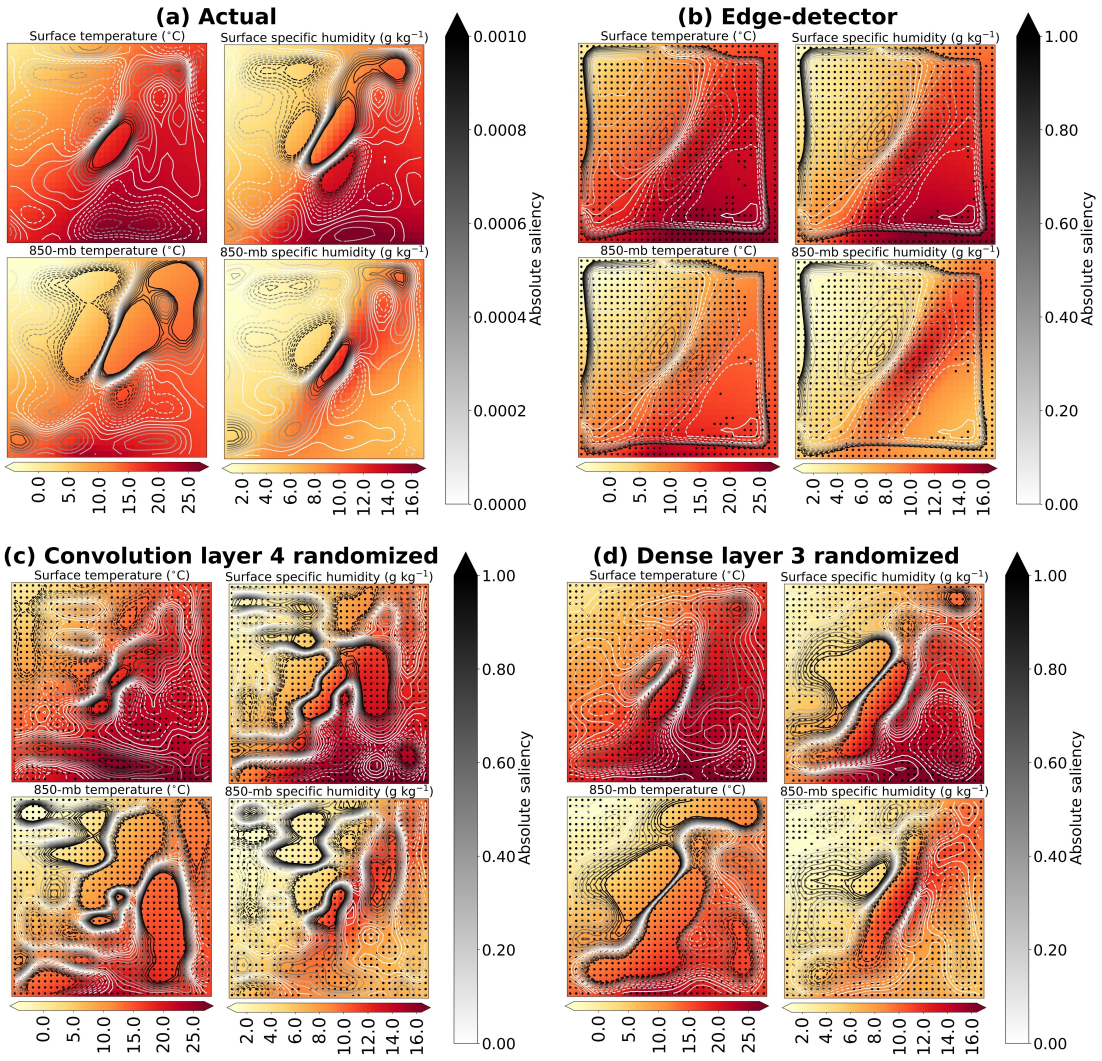


Figure 7.23: Sanity checks for saliency maps on best hits for cold fronts. Heat maps show predictors and are identical to those shown in Figure 6.20, while line contours show saliency. Solid contours indicate positive saliency (cold-front probability increases with the underlying value), while dashed contours indicate negative saliency (cold-front probability decreases with the underlying value). Each image (both predictor and class-activation fields) is a PMM composite over 1000 examples. [a] Actual saliency map. [b] Saliency map produced by an untrained edge-detector. [c] Saliency map produced after randomizing weights in the fourth convolutional layer. [d] Saliency map produced after randomizing weights in the third dense layer. In panels b-d, stippling shows where the actual and dummy saliency maps are significantly different at the 95% confidence level.

Chapter 8

Climatology of Warm and Cold fronts

Section 8.1 describes methods used to create the climatology; Sections 8.2-8.4 discuss results (overall averages, ENSO-relative variability, and long-term trends); and Section 8.5 discusses sensitivity of the climatology to hyperparameters.

8.1 Methods

I apply the selected CNN from the hyperparameter experiment (Figure 4.1), with the selected probability thresholds from the object-conversion experiment (65%; see Figure 5.6), to the full grid from 1979-2018. During model development, some grid cells were masked out (Figure 3.12) because the WPC does not label fronts in these areas, so ground truth is unavailable. However, when creating the climatology, the model has already been trained and ground truth is no longer needed, so no grid cells are masked out. Although the ERA5 contains hourly time steps, I apply the CNN to three-hourly time steps, because this process is very computationally expensive (see discussion at beginning of Section 5.2.1). For the selected CNN, labeling the full grid takes $\sim$20 minutes per time step on one computing core. The climatological period is 350 640 hours long, leading to 116 880 core-hours of computing time for hourly time steps. Even when parallelized over 90 cores (the most I can reliably obtain for an extended time), the resulting clock time is 54 days. Paring the dataset down to three-hourly time steps reduces the clock time to 18 days.

I investigate two properties in the climatology: frequency and length of both warm and cold fronts. Warm- (cold-)front frequency is defined as the fraction of time steps with a warm (cold) front at the given location, and length is defined as the major-axis length of the frontal zone. Length is not computed for fronts that touch the edge of the grid, because in these cases there is no way to ascertain how far the front extends beyond the

edge. I conduct each analysis independently for the four seasons: winter (DJF), spring (MAM), summer (JJA), and fall (SON). I explore both ENSO-relative variability and long-term trends in front frequency.

For variability, I define ENSO state as the standardized anomaly of the Niño 3.4 index, computed with detrended sea-surface temperature (SST) anomalies relative to a 1981-2010 baseline (Climate Prediction Center 2020). Each comparison – between a composite of ENSO-neutral months and a composite of non-neutral months – is done for one season, so that ENSO signals are not confounded with seasonal signals. The non-neutral composite contains months with El Niño, La Niña, strong El Niño, or strong La Niña conditions, all defined in Table 8.1. Statistical significance is determined by a two-tailed Monte Carlo test (defined in Section 7.1.2) with 20 000 iterations. As discussed in Section 7.1.2, because entire spatial maps are shuffled between the two composites (rather than shuffling independently at each grid point), the shuffling procedure accounts for spatial correlation, so I do not use an additional method to control the false-discovery rate (FDR).

Table 8.1: Definition and sample size for each ENSO phase. z is the standardized anomaly of the Niño 3.4 index.

Composite	**Sample size (number of months)**			
	Winter (DJF)	**Spring (MAM)**	**Summer (JJA)**	**Fall (SON)**
Neutral ($\lvert z\rvert < 0.5$)	38	71	71	50
El Niño ($z \geq 0.5$)	35	23	24	34
La Niña ($z \leq -0.5$)	47	26	25	36
Strong El Niño ($z \geq 1.0$)	22	11	9	20
Strong La Niña ($z \leq -1.0$)	18	6	8	21

To compute long-term trends and assess their significance, I use the two-tailed Mann-Kendall test (Mann 1945; Kendall 1975) as implemented in the pyMannKendall package (Hussain and Mamhud 2019). The outputs at each grid cell are the slope of the Theil-Sen line, interpreted as the linear trend over time, and the p-value. In this case the test is conducted independently at each grid point and therefore does not account for spatial correlation, so the FDR must be explicitly controlled. I use Equation 3 from Wilks (2016), which outputs a p-value threshold (p^*) that ensures an upper bound on the FDR. I use a maximum FDR of 0.10 as in Schemm et al. (2018). Only grid points with $p \leq p^*$ are considered significant.

8.2 Overall Averages

8.2.1 Cold-front Frequency

Figure 8.1 shows the frequency of cold fronts in all four seasons. In winter (Figure 8.1a), cold fronts over the Pacific are mostly confined to a band from 25-50 °N, with the highest frequencies from 30-40 °N. This swath curves northward as it approaches the west coast of North America, following the typical cyclone track (EH06; Hodges et al. 2011, henceforth H11). Most of these fronts dissipate when they encounter high terrain along the west coast or subtropical air in southern California. Landfalling cold fronts on the west coast are most common in the Pacific Northwest, and fronts reaching the interior (beyond the coastal states) are most common in the desert southwest, where the terrain is less steep. Over the continent, cold fronts are most common from the lee of the Rockies to the Appalachians, an area with enhanced cyclone activity (EH06, H11). The southern extent of regular cold-front occurrence is the Yucatán peninsula of Mexico. Over the Atlantic, cold fronts generally occur east of the Gulf Stream, along another known cyclone track (EH06, H11). There are also regional maxima near the northwest corner of South Carolina, associated with cold-air damming in the lee of the Appalachians (Rackley and Knox 2016), and in the northeast Gulf of Mexico.

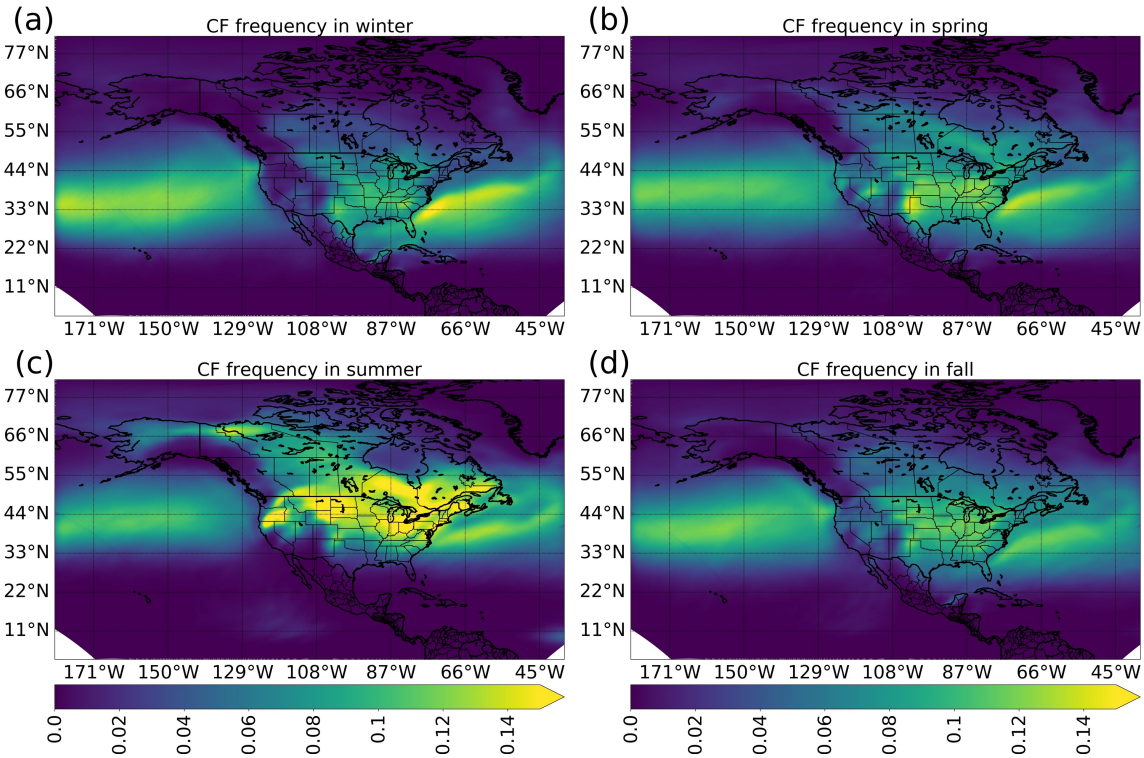


Figure 8.1: Frequency of cold fronts in each season from 1979-2018. Frequency is the fraction of 3-hour time steps with a cold front.

In the summer (Figure 8.1c), cold fronts generally do not occur south of 35 °N, while maxima in the Pacific and Atlantic are shifted ∼10° north of their winter counterparts. EH06 and H11 show similar results for the distribution of cyclones. In much of Canada and the northern Plains, cold-front frequency peaks in the summer. These are all effects of the global annual heating cycle, discussed in Section 2.2.2. Regionally, the summer maximum in the Pacific Northwest (Figure 8.1c) coincides with lower surface elevations, and the narrow band from Washington State to southwestern Alberta suggests that a common path for cold air from the Canadian prairies is through Crowsnest Pass and the Columbia Basin. A similar pattern is present, although with lower frequencies, in the spring and fall.

In the spring and fall (Figure 8.1b,d), cold fronts have a similar distribution to winter, except that maxima in the Pacific and Atlantic are less prominent and shifted ∼5° northward, the latter effect due to the global annual heating cycle. This shift also affects the continent, leading to fewer fronts in the Gulf of Mexico and more east of the Rockies. The spring also features prominent regional maxima: in the lee of the Appalachians, due to cold-air damming; in southern Nevada, at the southern edge of the Great Basin; and in a swath from Great Slave Lake (the large lake in the southern Northwest Territories) to James Bay. This swath may be associated with the northern part of the Alberta Clipper cyclone track (Thomas and Martin 2007).

8.2.2 Warm-front Frequency

Figure 8.2 shows the frequency of warm fronts in all seasons. Overall, there are three main differences between warm and cold fronts. First, warm fronts often coincide with land-sea boundaries, while cold fronts do not. By inspecting individual cases, I have confirmed that such warm fronts are identified only when there is warm advection across the boundary (*i.e.*, the CNN does not misclassify stationary fronts as warm fronts), but the advection is often weak. Meanwhile, weak cold advection across a land-sea boundary is generally not identified as a cold front. This likely occurs because the CNN has learned that cold fronts

are generally stronger, so larger gradient/advection quantities are required for cold fronts to meet the 65% probability threshold. Second, large-scale maxima in warm-front frequency generally occur $\sim 10^{\circ}$ north of their cold-frontal counterparts (*c.f.*, Figures 8.1 and 8.2), consistent with mean frontal positions relative to their parent cyclones. Third, warm fronts are common in the tropical eastern Pacific, while cold fronts are nearly absent. B11b found similar results (their Figure 3), and B11a found that these fronts are characterized almost entirely by humidity gradients (their Figure 4). Figure 8.3 shows two examples, with dry air likely originating from subsidence under subtropical high pressure. According to Figure 8.2, these warm fronts are most common in the summer and nearly absent in the winter, which suggests that humidity gradients here reach the warm-front-detection threshold only when there is abundant moist air from the tropics to contrast with dry air. However, B11b found that these fronts are most common in the winter (their Figure 3).

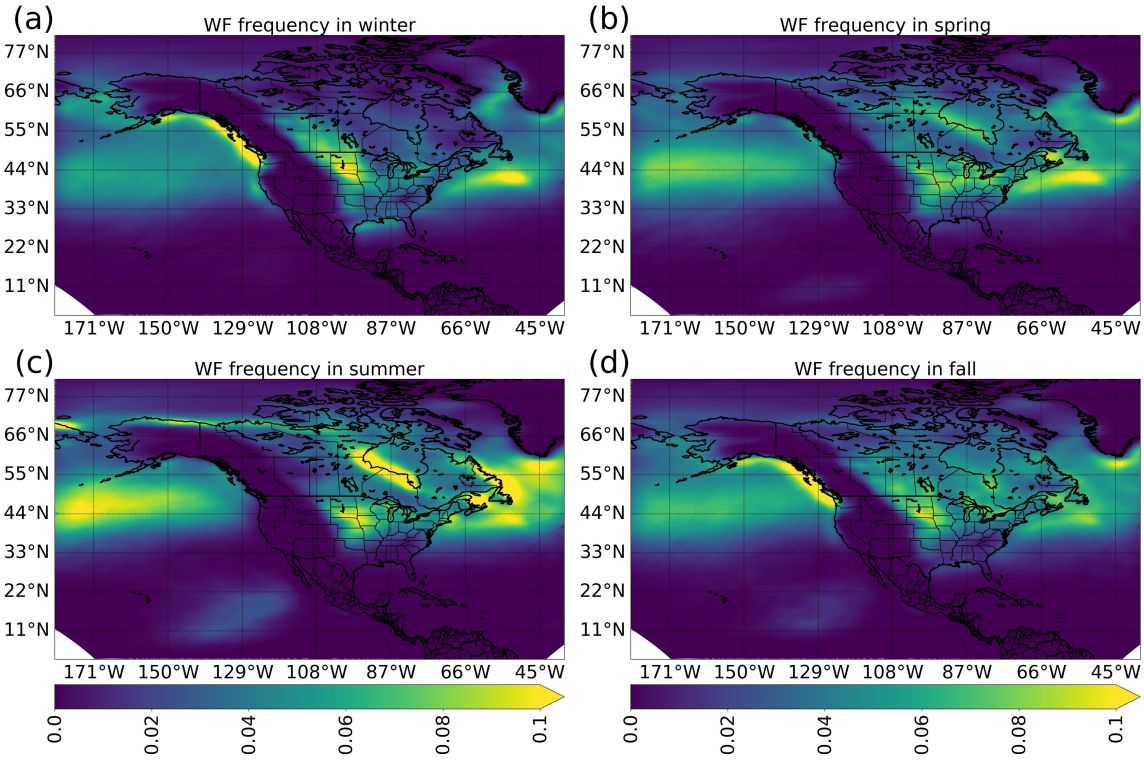


Figure 8.2: Same as Figure 8.1 but for warm fronts.

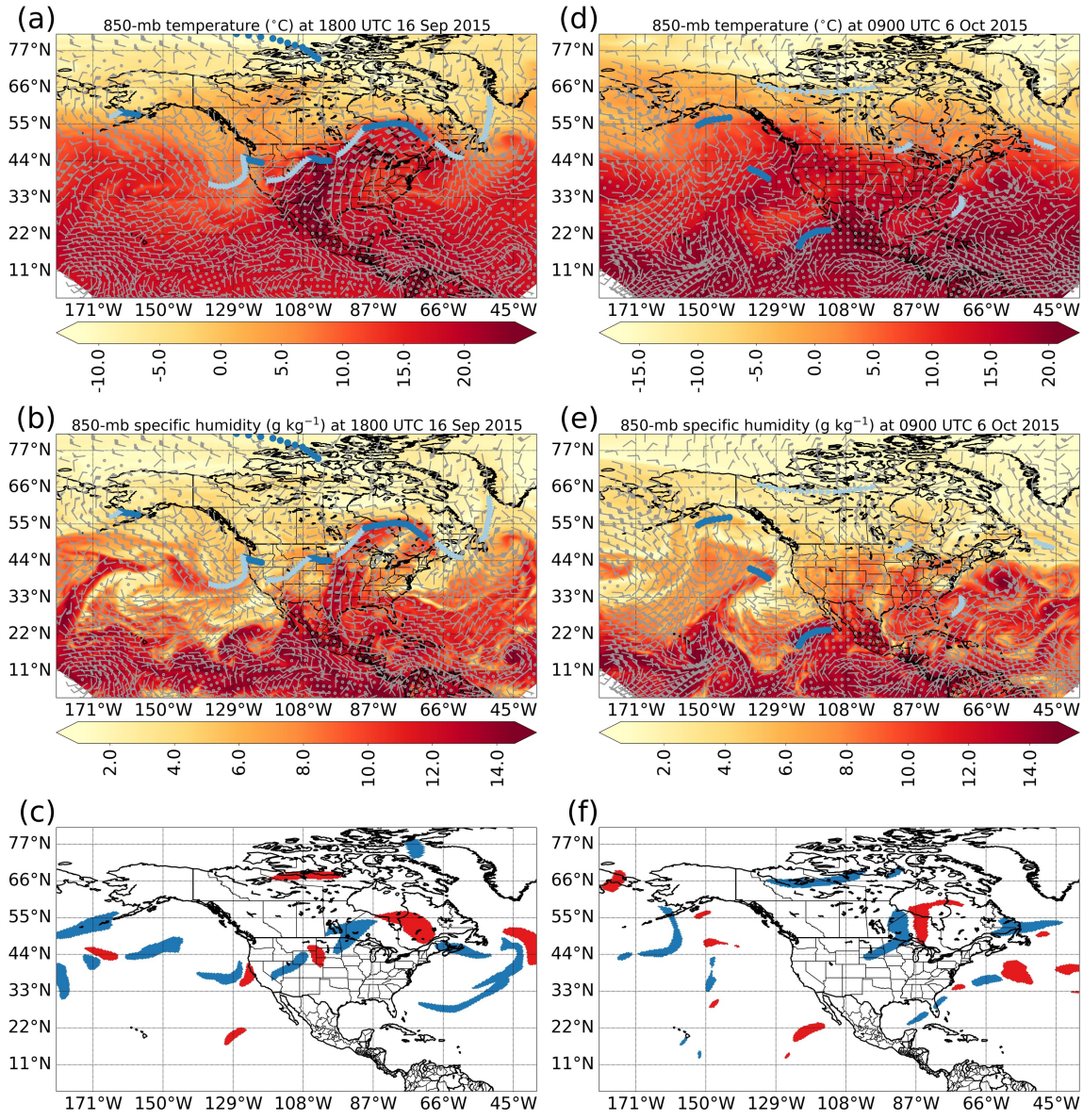


Figure 8.3: Examples of warm fronts in eastern tropical Pacific. [a-b] Predictors at 850 mb at 1800 UTC 16 Sep 2015. Formatted as in Figure 3.10, except that the heat map is no longer θ_w. [c] CNN predictions at 1800 UTC 16 Sep 2015. [d-f] Same as left column but for 0900 UTC 6 Oct 2015.

In winter, there are two additional differences between the warm- and cold-front distributions (*c.f.*, Figures 8.1a and 8.2a). First, warm fronts have a maximum along the west coast from central California to Alaska, while cold fronts have a minimum in this area. Fronts in the eastern Pacific generally travel eastward (from ocean towards land), and in winter the ocean is generally warmer than the adjacent land, enhancing warm advection but diminishing cold advection. Thus, cold fronts in the eastern Pacific dissipate farther offshore than do warm fronts. Second, warm fronts have a maximum along the Gulf coast, which may be caused by frequent southerly flow from the Gulf of Mexico over the relatively cool land.

Fall warm fronts (Figure 8.2d) have a similar distribution to winter, with the following exceptions. First, large-scale maxima are shifted slightly northward in the fall, most evident in the Pacific and Atlantic cyclone tracks. Second, the fall has regional maxima over James Bay and the Gulf of Saint Lawrence (between Newfoundland and Nova Scotia). These are associated with warm advection from water bodies, which are near their annual maximum temperatures, over the relatively cool land. Third, the fall has a maximum in the tropical eastern Pacific, due to fronts dominated by humidity gradients (Figure 8.3).

Spring warm fronts (Figure 8.2b) have a similar distribution to fall, with the following exceptions. First, spring has only a weak maximum along the west coast, because the eastern Pacific is generally not much warmer than the adjacent land, especially in late spring. Second, the spring has maxima just south of the Great Lakes and southwest of James Bay, whereas the fall has maxima just north of the Great Lakes and over James Bay. The fall maxima are due to warm advection from water bodies over relatively cool land, while the spring maxima are due to warm advection from land over relatively cool water. Third, the spring has a maximum along the entire southwest shore of Hudson Bay, associated with warm advection over landfast sea ice. Fourth, the spring has a maximum from the Plains to the southern Great Lakes, consistent with Morgan et al. (1975).

Summer warm fronts (Figure 8.2c) are distributed similarly to spring, with notable exceptions as follows. First, large-scale maxima are shifted poleward, $\sim 5^\circ$ north of their spring counterparts and $\sim 10^\circ$ north of their winter counterparts. Second, the maximum along the southwest shore of Hudson Bay is much stronger, aided by a stronger land-sea temperature gradient (the land warms a lot from spring to summer, while the water does not). Third, there is a long maximum over the Arctic coast of North America and eastern Siberia, also associated with offshore warm advection.

8.2.3 Front Length

Figures 8.4-8.5 show the mean length of cold and warm fronts, respectively, in all four seasons. Cold fronts are generally twice as long as warm fronts. During the winter, there are three large-scale maxima in cold-front length (Figure 8.4a): in the Pacific, along the east coast of the United States, and just east of the Gulf Stream. The highest values (mean length of $\sim$3000 km) occur just east of the Gulf Stream, where long cold fronts trailing from mature cyclones often separate cold continental air from warm tropical air. The other seasons (Figure 8.4b-d) have similar patterns, except that cold fronts are shorter, due partly to the global annual heating cycle preventing cold fronts from reaching as far south. For winter warm-front length (Figure 8.5a), there are also three large-scale maxima: the Pacific, the lee of the Rockies, and just east of the Gulf Stream (likely associated with the same mature cyclones that cause the maximum in cold-front length). In the last two areas, warm-front length peaks in the winter; in the Pacific, warm-front length peaks in the spring to fall (Figure 8.5b-d), when northward transport of warm air is stronger, causing frequency to peak as well (Figure 8.2).

8.3 ENSO-relative Variability

This subsection discusses the influence of ENSO on warm- and cold-front frequencies throughout the domain, with ENSO phases defined in Table 8.1. The focus is primarily

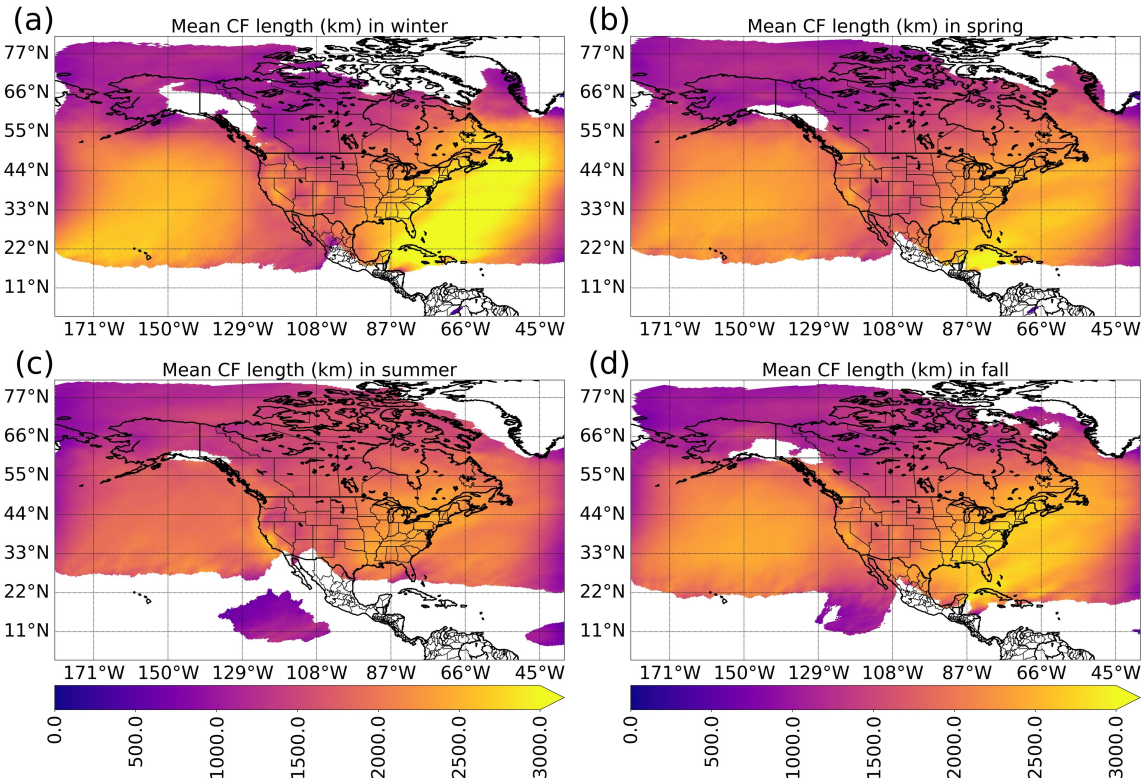


Figure 8.4: Average cold-front length (km) in each season from 1979-2018. Length is computed for each front at each time step, and the value at grid cell g is the average for all cold fronts touching g. For each season, grid cells with < 100 cold fronts are not plotted, in order to reduce clutter.

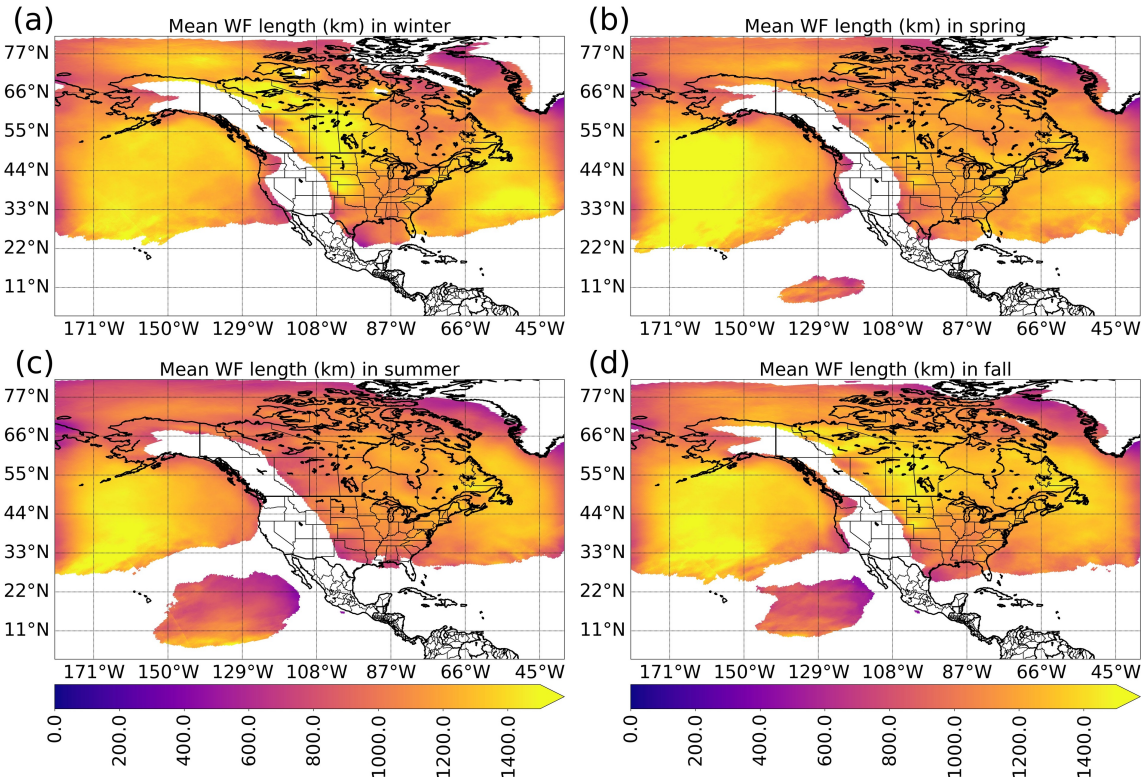


Figure 8.5: Same as Figure 8.4 but for warm fronts.

on winter, since ENSO teleconnections are strongest in winter (Horel and Wallace 1981; Trenberth et al. 1998) and the two previous studies linking fronts to ENSO (RS15 and Hardy and Henderson 2003, henceforth HH03) also focus on winter. Fronts will sometimes be linked to cyclones, even though this work does not explicitly identify cyclones. This will be done only for areas known to be active cyclone tracks, where (a) a large swath of increased warm-front frequency is collocated with, or just north of, a large swath of increased cold-front frequency; (b) previous work suggests that the same ENSO phase increases cyclone frequency in the same area.[1] However, these links are made with the caveat that not all fronts are related to cyclones, so the changes in front frequency may be due to other phenomena.

8.3.1 Winter

Figure 8.6 shows ENSO influences in the winter. Changes plotted are absolute, not relative. For example, if the baseline frequency (during ENSO-neutral months) is 0.06 and the change is +0.01, this signifies an increase from 0.06 to 0.07, which is a relative increase of 17% ($\frac{0.01}{0.06}$). El Niño (Figure 8.6a-b) has significant influences in the Pacific, CONUS, Gulf of Mexico, parts of the Atlantic, and parts of sub-Arctic Canada. In the Pacific, both warm and cold fronts have a swath of increased frequency to the south and decreased frequency to the north, representing a southward shift in front activity. This is due to a southward shift in cyclone activity, shown in Figure 8 of EH06, driven by southward displacement of the subtropical Pacific jet (Trenberth et al. 1998).

Farther east, El Niño increases both warm- and cold-front frequencies over the Gulf of Mexico, while decreasing frequencies over the CONUS. This is due to an eastward extension of the subtropical Pacific jet into the Atlantic (Figure 13 of Schemm et al. 2018, henceforth S18), causing more cyclones in the Gulf of Mexico and fewer in the CONUS. This result is consistent with Figure 7b-c of S18, which suggests that east-Pacific El Niño

[1]The same logic applies if all instances of "increase" in this sentence are replaced by "decrease."

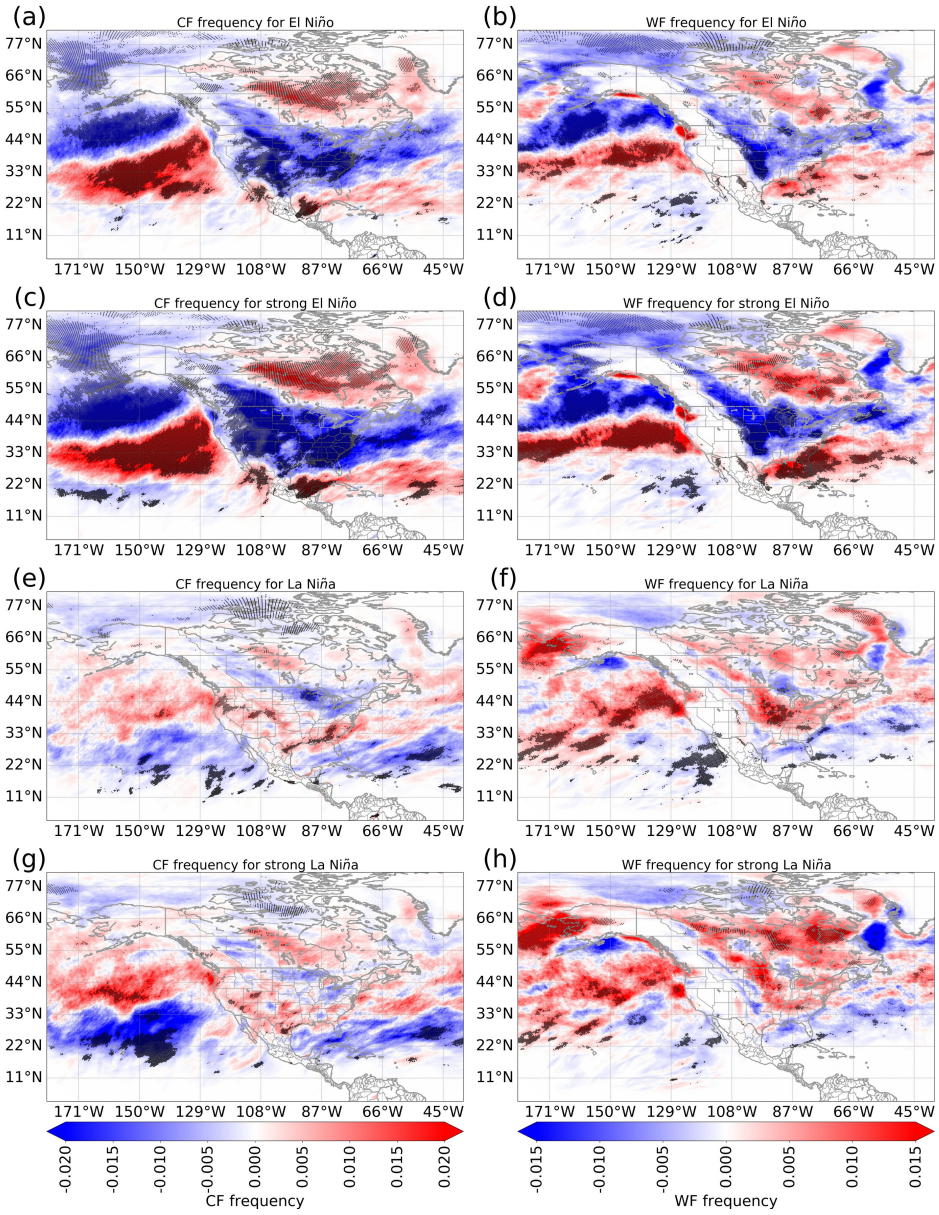


Figure 8.6: Front frequency by ENSO phase during winter. Each panel shows the composite difference (frequency during the given phase minus frequency during ENSO-neutral). Stippling shows where this difference is significant at the 95% confidence level, determined by a two-tailed Monte Carlo test with 20 000 iterations. [a] Composite difference in cold-front frequency during El Niño. [b] Same but for warm fronts. [c-d] Same as a-b but for strong El Niño. [e-f] Same as a-b but for La Niña. [g-h] Same as a-b but for strong La Niña.

increases cyclogenesis in the Gulf of Mexico, while central-Pacific El Niño decreases cyclogenesis in the lee of the Rockies. My results could be due to additive contributions from these two modes, but this is speculative, as I do not investigate inter-ENSO variability. The pattern over the CONUS and Gulf of Mexico extends eastward into the Atlantic (Figure 8.6a-b), suggesting a southward shift here as in the Pacific. Also, the warm-front increase extends up the east coast as far north as Virginia, while the cold-front increase reaches southern Florida. Considering mean frontal positions relative to their parent cyclones, this pattern mildly suggests enhanced cyclone activity along the southern part of the east coast. This would be consistent with S18 and EH06 (their Figure 8).

Elsewhere, El Niño yields few significant differences, except a swath of increased warm- and cold-front frequencies from Great Slave Lake to northern Québec. This may be caused by an increase in cyclone frequency due to a collocated anomalous polar jet stream during El Niño (left panel of Figure 4 in Cook and Schaefer 2008).

Figure 8.6c-d shows analogous results for strong El Niño, created by increasing the threshold from 0.5 to 1.0 (Table 8.1). Almost everywhere in the domain, increasing the threshold increases both the amplitude and significance of the signal. Notably, the southward shift in the Atlantic becomes more broadly significant, although most grid points are still not significant.

During La Niña, the cold-front response (Figure 8.6e) is generally opposite of that for El Niño. La Niña pushes the subtropical Pacific jet northward, which precludes its extension into the Atlantic (Figure 13 of S18). These effects push the Pacific and Gulf of Mexico cyclone tracks northward (Figure 8 of EH06), leading to a northward shift in front activity. La Niña also causes upper-level ridging over the west coast and troughing over the Rockies (S18), which enhances cyclogenesis in the lee of the Rockies (Figure 7a of S18). These cyclones generally propagate eastward, leading to increased front frequency in much of the eastern CONUS. However, these responses are not broadly significant. Increasing the threshold to strong La Niña (Figure 8.6g) enhances the significance of the northward

shift in the Pacific, but most grid points here are still not significant. Also, increasing the La Niña threshold destroys significance in many parts of the CONUS – notably the southeast, Minnesota, and Wisconsin. This contrasts with El Niño, for which increasing the threshold generally enhances significance. Three factors are likely to blame. First, La Niña teleconnections are weaker (*e.g.*, Lyon and Barnston 2005; Butler et al. 2014). Second, SST anomalies during La Niña are smaller, and restricting the range of a predictor generally decreases its correlation with the predictand (Lyon and Barnston 2005). Third, increasing the La Niña threshold decreases sample size more (from 47 to 18 months) than increasing the El Niño threshold (from 35 to 22 months), as shown in Table 8.1.

The warm-front response to La Niña (Figure 8.6f) is similar to the cold-front response, with two notable exceptions. First, warm-front frequency increases strongly in the Bering Sea, while cold-front frequency is nearly unchanged. The warm-front increase is significant and enhanced by increasing the threshold to strong La Niña (Figure 8.6h). La Niña shifts the semi-permanent Aleutian low westward (Figures 5a-b of Niebauer 1998), shifting its eastern flank from the Aleutian Islands to the Bering Sea, which may allow for more southerly advection of mild Pacific air over the Bering Sea. This would also be consistent with decreased warm-front frequencies over the Aleutian Islands, although the latter are not significant. Second, warm-front frequency increases northwest of Hawaii. Based on visual inspection (examples in Figure 8.7), many of these are back-door fronts associated with an anticyclone. These fronts are characterized by much stronger humidity gradients than temperature gradients, due to large-scale subsidence in the subtropics. Such warm fronts may occur more often during winter La Niña due to stronger subtropical high pressure (Figure 10a of Reason et al. 2000).

Overall, my results for winter are consistent with RS15 (their Figure 6b), who found that increasing the Niño 3.4 index causes a significant southward shift in front frequency at most longitudes. The main differences are as follows. First, their southward shift in the Pacific occurs over a different range of latitudes, with increases from 25-50 °N and decreases north

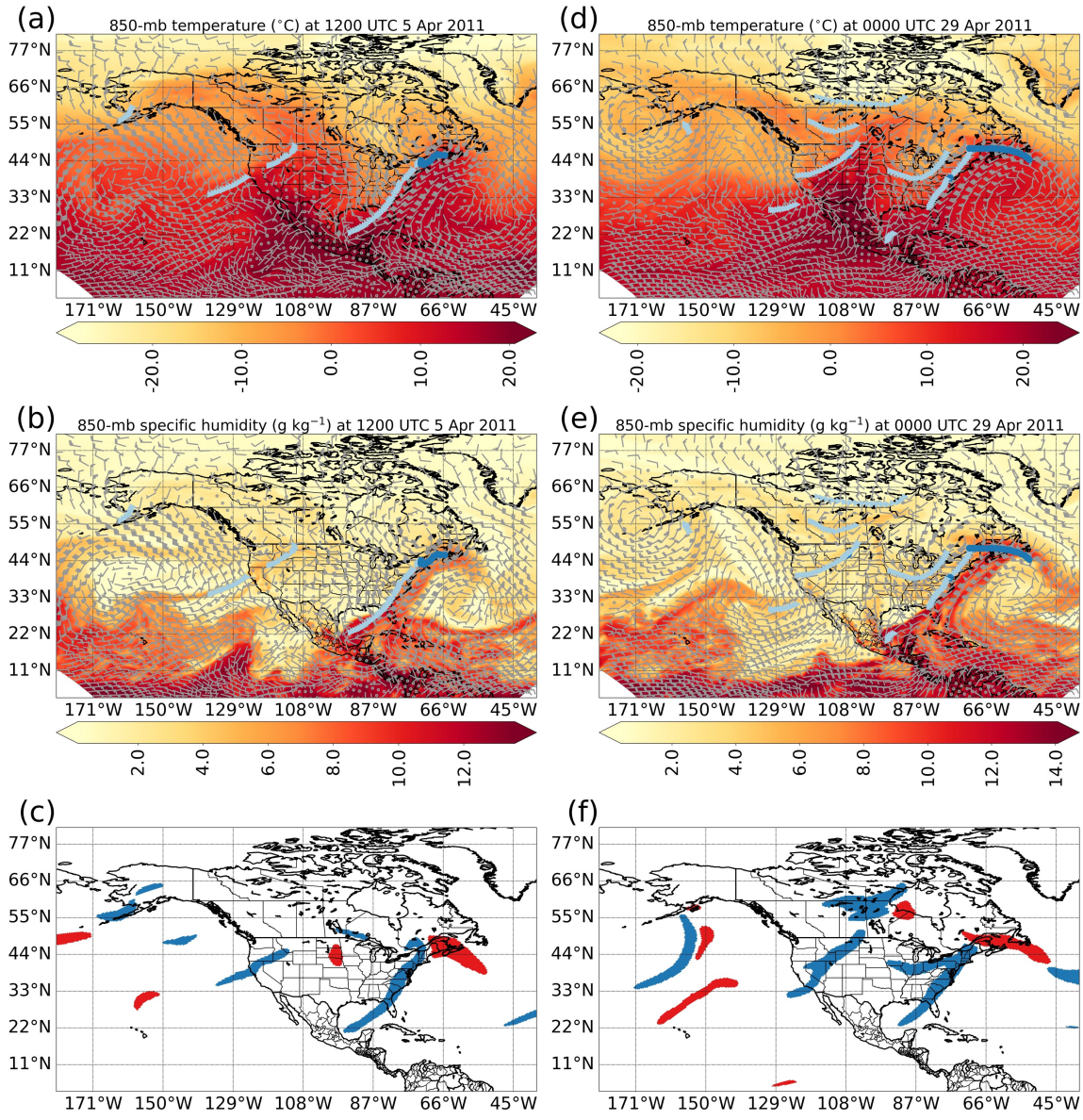


Figure 8.7: Examples of warm fronts near Hawaii. [a-b] Predictors at 850 mb at 1200 UTC 5 Apr 2011. Formatted as in Figure 3.10, except that the heat map is no longer θ_w. [c] CNN predictions at 1200 UTC 5 Apr 2011. [d-f] Same as left column but for 0000 UTC 29 Apr 2011.

of 60 °N. Second, their results in the Atlantic are more broadly significant, especially in the increasing swath to the south. Third, their increases over the eastern Canadian Arctic and Greenland are generally significant, while mine are generally insignificant. However, two factors hinder direct comparison with RS15. First, I analyze warm and cold fronts separately, while they make no distinction. Second, they use the wind-shift method for front detection (Simmonds et al. 2012), which rarely detects warm fronts, as noted in the conclusion of RS15. Thus, their analysis is dominated by cold fronts, although it is unclear to what extent.

Lastly, HH03 investigated ENSO influences on cold-front frequency in the southeastern CONUS and found no significant response. This lack of significance is consistent with my results for La Niña (Figure 8.6e,g) but not El Niño (Figure 8.6a,c), where I have found a significant decrease in most of the area. Several factors are likely to blame for this disagreement. HH03 used a manual front-detection method; they used a 5° grid rather than a 32-km grid, making their grid cells over 200 times larger; they used the Southern Oscillation Index (SOI), rather than Niño 3.4, to define ENSO phase; and their analysis period was 1961-1990, as opposed to 1979-2018.

8.3.2 Spring

The cold-front response to spring El Niño (Figure 8.8a) is similar to the winter response (Figure 8.6a), with two notable exceptions. First, the spring response occurs farther south, with almost no significant grid points north of 50 °N. This suggests that ENSO teleconnections work over a smaller meridional distance in the spring, consistent with weaker SST anomalies (Trenberth et al. 1998) and weaker westerly winds connecting the mid-latitudes to the heating source in the equatorial troposphere (Horel and Wallace 1981). Also, deep convection in the mid-latitudes is much more frequent and intense during spring, and the upscale feedbacks of deep convection might mask ENSO teleconnections. Second, the spring response is weaker and less significant, although it still suggests a southward shift

in front activity at most longitudes. Also, increasing the El Niño threshold enhances the cold-front response almost everywhere in the domain (Figure 8.8c). Increases in the southern CONUS, although generally not significant in this work, are consistent with Allen et al. (2015, left column of their Figure 3), showing a collocated increase in northerly wind and decrease in CAPE.

Warm fronts have a similar response (Figure 8.8b), but there are fewer significant grid points overall. Also, the tropical eastern Pacific features a marginally significant northward shift in warm fronts, which is enhanced by strong El Niño (Figures 8.8b,d). Recall that these fronts are due almost entirely to humidity gradients (Figure 8.3). El Niño strengthens the Hadley cell (Seager et al. 2003), which features dry descending air in the subtropics and equatorward flow of this dry air in the low-level trade winds. A stronger Hadley cell might allow for drier air in the descending branch (faster descent leads to less entrainment), thus allowing the humidity gradient between dry subtropical air and moist tropical air to exceed the warm-front-detection threshold farther north. However, this is speculative and requires future investigation.

The cold-front response to La Niña is generally not significant and not enhanced by increasing the threshold to strong La Niña (Figure 8.8e,g). In many areas strong La Niña even flips the sign of the response, such as west of Vancouver Island and in the northern Gulf of Mexico. This suggests that the response is highly non-linear, which is not surprising, given that (a) ENSO teleconnections are weaker in spring than in winter; (b) La Niña teleconnections are weaker than El Niño teleconnections. The largest significant area is a cold-front increase near the Pacific Northwest, which is collocated with a marginally significant warm-front increase (Figure 8.8f,h). This suggests an increase in cyclone frequency, which may be due to an eastward extension of the cyclone track during La Niña, noted by EH06 (their Figures 8 and 9c-d). However, their results are for the JFM season, which has only one month of overlap with spring as defined herein (MAM), so the comparison is indirect.

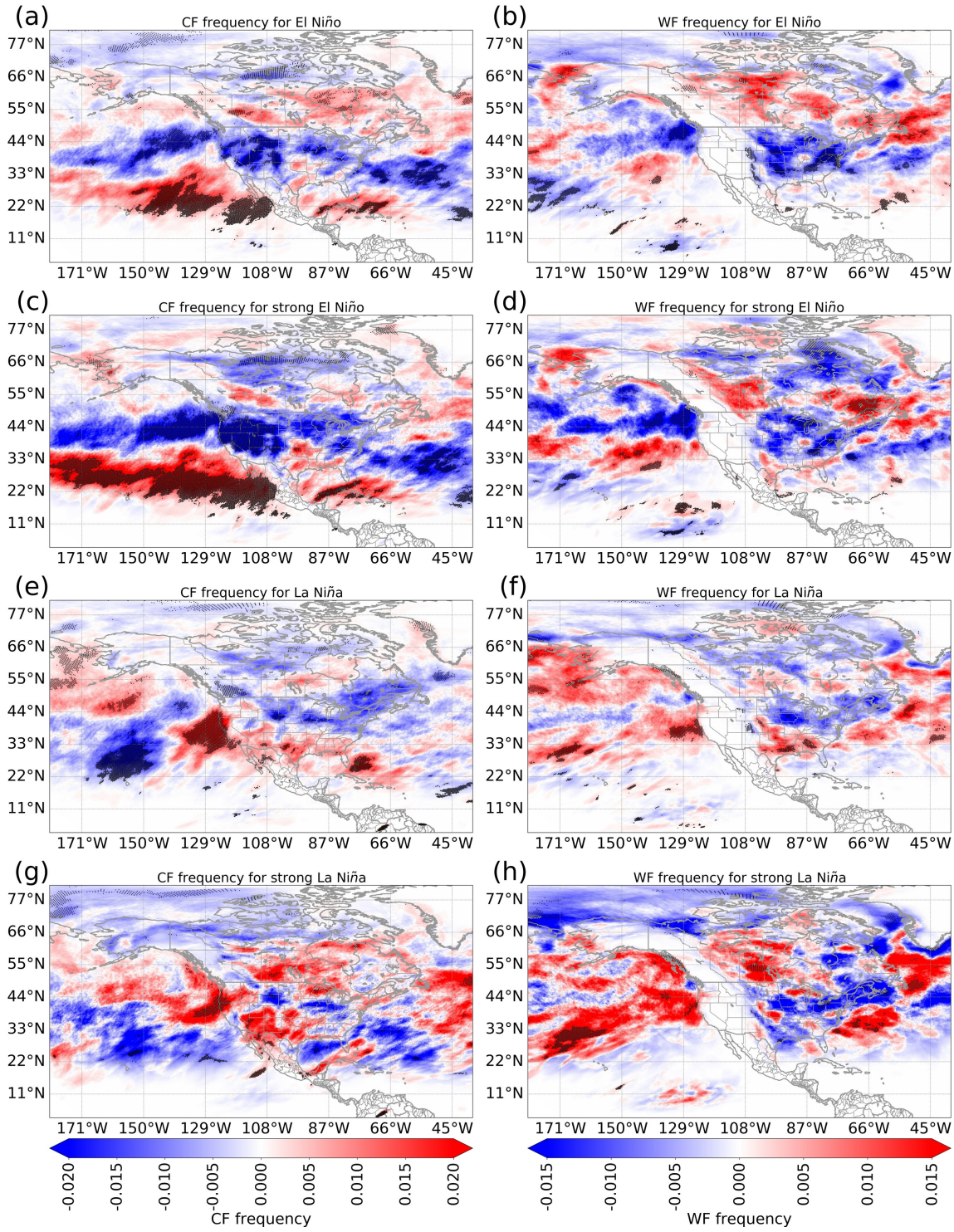


Figure 8.8: Front frequency by ENSO phase during spring. Formatting is explained in Figure 8.6.

The warm-front response to La Niña (Figure 8.8f,h) also suffers from lack of significance and non-linearity (*e.g.*, in the southeastern CONUS, where strong La Niña turns moderate increases into moderate decreases). Notable exceptions are as follows. First, there is a marginally significant decrease from the northern Plains to the Canadian maritimes. Some of this area features decreased cold-front frequency as well, which suggests a decrease in cyclone frequency, consistent with RS15 (their Figure 6c). However, since their results are for winter and my cold-front decreases in the area are statistically insignificant, this interpretation is highly speculative. Second, there are marginally significant warm-front increases east of the mid-Atlantic coast and Newfoundland. The increase near Newfoundland could be due to a collocated increase in cyclone frequency during La Niña (Figure 8 of EH06), but again, their results are for the JFM season.

8.3.3 Summer and Fall

The summer response to El Niño (Figure 8.9a-d) is marginally significant in three areas. First, there is a southward shift in both warm and cold fronts over the Pacific cyclone track, consistent with southward jet displacement. Second, there is a northward displacement of warm fronts over the tropical eastern Pacific, as in the spring. Third, there is a southward shift in both warm and cold fronts over the Atlantic cyclone track, the cause of which is unclear. Winter El Niño causes an eastward extension of the subtropical Pacific jet into the Atlantic, but during summer this does not occur, because the jet is farther north (separated from the Atlantic by more land) and weaker. Meanwhile, the summer response to La Niña (Figure 8.9e-h) is marginally significant in two areas. The first is a northward shift in both front types from $\sim$30-60 °N, consistent with northward jet displacement. The second is an increase in both front types from $\sim$10-15 °N in the eastern Pacific. This is not quite opposite of the El Niño pattern, because (a) the La Niña pattern does not include a spatially coherent decrease to the north, so it does not suggest a southward shift; (b) the La Niña pattern involves cold fronts as well, as dry advection in the summer is strong enough to be

detected by the CNN. This pattern, like the El Niño pattern in the same area, requires future investigation.

The fall response to both ENSO phases (Figure 8.10) is broadly similar to the summer response, featuring meridional shifts associated with the subtropical Pacific jet. The main differences are (a) the fall response occurs farther south, consistent with fronts in general occurring farther south; (b) the warm-front response in the tropical eastern Pacific is quite different, featuring a southward shift during La Niña and spatially incoherent pattern during El Niño.

8.4 Long-term Trends

Figure 8.11 shows the trend in warm- and cold-front frequencies for each season. Changes plotted are absolute, not relative. For example, if the frequency at the beginning of the period is 0.06 and the change is +0.01, this signifies an increase from 0.06 to 0.07. Results are generally not significant, because the method used to control false-discovery rate (see end of Section 8.1) is overly conservative, as noted in the paper that introduced it (Wilks 2016). The resulting *p*-value thresholds, listed in the caption of Figure 8.11, are exactly zero for three panels and never exceed 0.015.

The winter trend (Figure 8.11a-b) suggests a marginally significant poleward shift in front activity over the mid-latitude cyclone track, suggesting a poleward shift in cyclone activity. This is consistent with many studies on global warming (Wu et al. 2011; Davis and Rosenlof 2012; Lucas et al. 2014; Schmidt and Grise 2017; Tamarin and Kaspi 2017). A similar northward shift is generally observed in the other seasons (Figure 8.11c-h) but with no statistical significance. Outside of the mid-latitude cyclone track, there are only two significant trends. The first is a warm-front decrease along the north coast of Labrador in the winter (Figure 8.11b), which could be caused by sea-ice decline (Figures 2 and 4 of Parkinson 2014; Figure 2 of Stroeve et al. 2014). Replacing sea ice with open water would increase air temperature over the ocean, thus decreasing the thermal gradient during periods

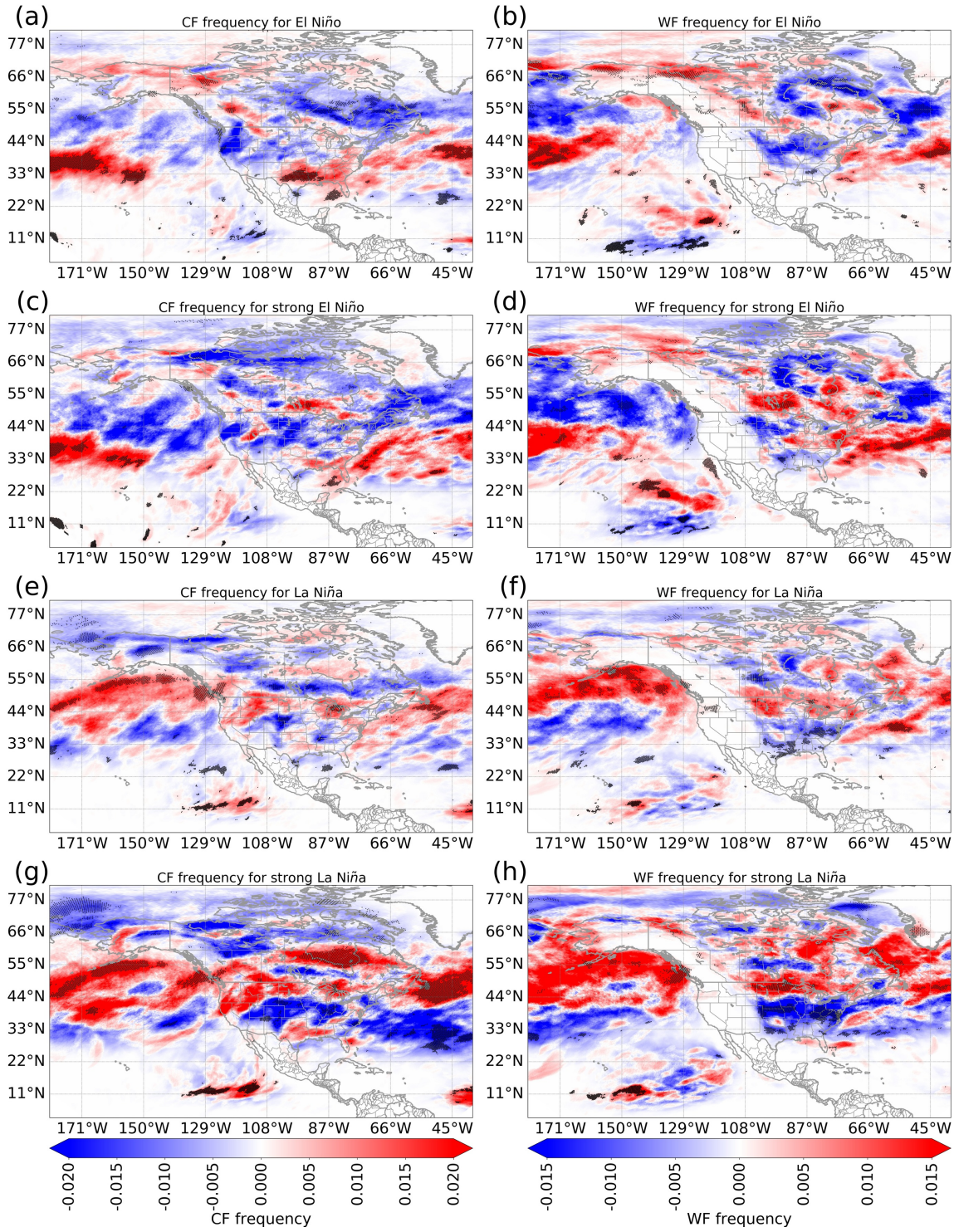


Figure 8.9: Front frequency by ENSO phase during summer. Formatting is explained in Figure 8.6.

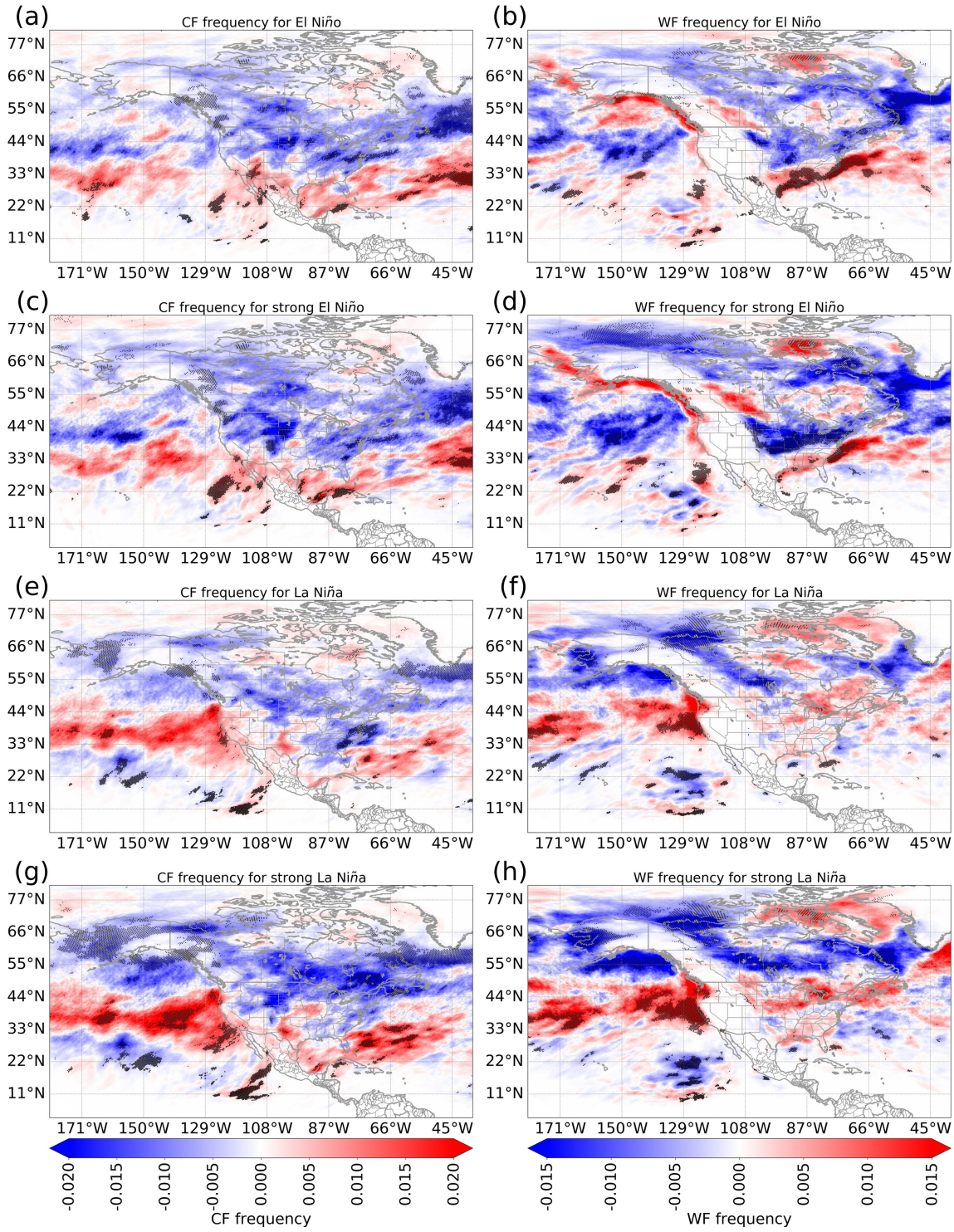


Figure 8.10: Front frequency by ENSO phase during fall. Formatting is explained in Figure 8.6.

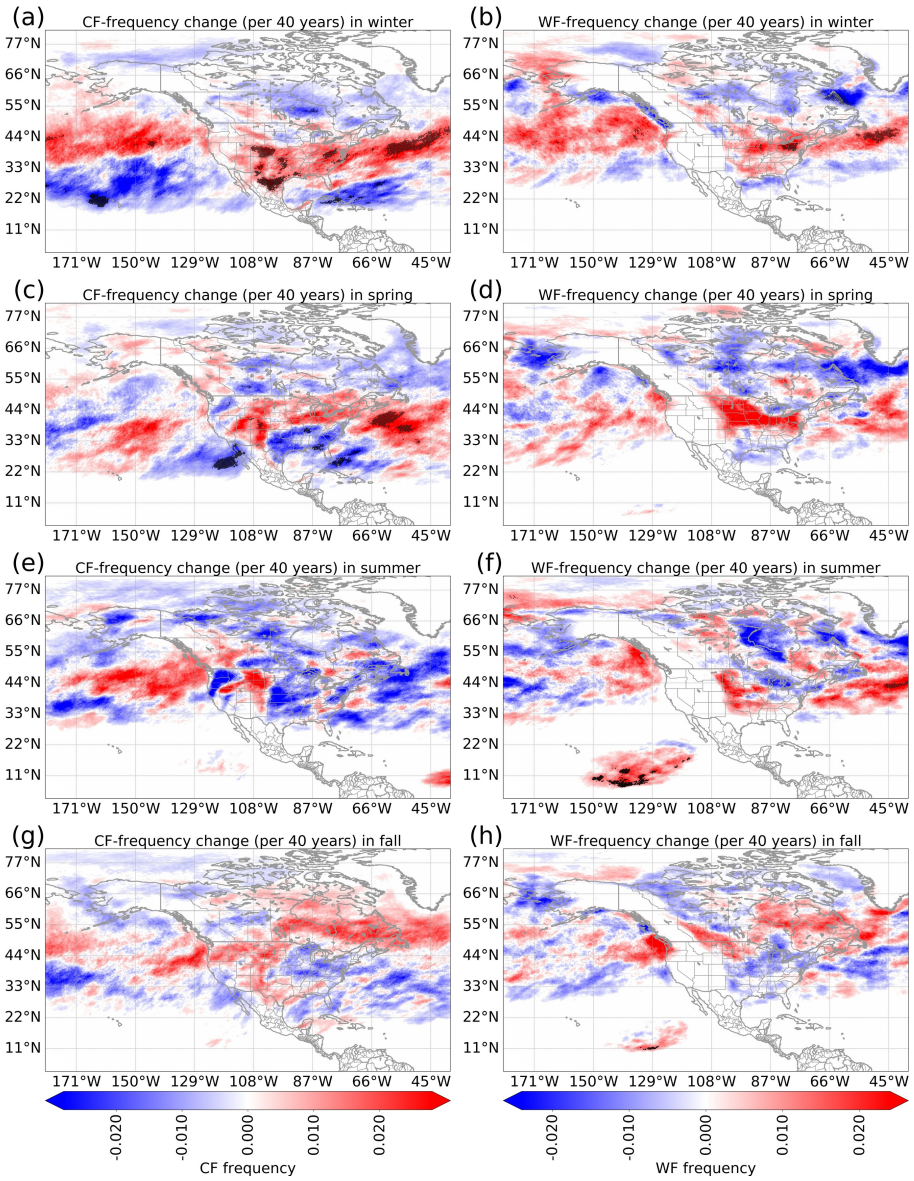


Figure 8.11: Linear trend in front frequency during each season. Each panel shows the absolute change per 40 years. Stippling shows where this difference is significant with a 10% false-discovery rate (FDR), determined by a two-tailed Mann-Kendall test. [a] Winter cold-front frequency. Here the FDR equation yields a p-value threshold (p^*) of 0.015. [b] Winter warm-front frequency (p^* = 0.009). [c-d] Same as a-b but for spring, with p^* = 0.012 for cold fronts and 0 for warm fronts. [e-f] Same but for summer, with p^* = 0 for cold fronts and 0.009 for warm fronts. [g-h] Same but for fall, with p^* = 0 for cold fronts and 0.001 for warm fronts.

of offshore warm advection. The same pattern (although without statistical significance) is present throughout the year except in fall, the one completely ice-free season in the area. The second significant trend is a warm-front increase in the tropical eastern Pacific during summer (Figure 8.11f), when warm fronts here are most common (Figure 8.2c). A similar increase is found by B11a (their Figure 2). As noted in B11a, increased front frequency in this area would be consistent with poleward expansion of the Hadley cell under global warming (Davis and Rosenlof 2012; Lucas et al. 2014; Schmidt and Grise 2017), allowing tropical moisture to advance farther north and meet dry subtropical air. However, future investigation of this relationship is needed, because while B11a note that El Niño should have a similar effect, my results for El Niño are quite different (*c.f.*, Figures 8.11f and 8.9b,d).

Two other studies investigate long-term trends in front frequency over the northern hemisphere, the first of which is B11a. Outside of the tropical eastern Pacific, they found a significant northward shift in the Pacific cyclone track and significant decrease over the Atlantic cyclone track (their Figure 2). This conflicts with my results in the Atlantic (Figure 8.11), which suggest a shift in preferred front locations rather than an overall decrease. However, direct comparison with B11a is hindered by two factors. First, they used a different front-detection method, based only on 850-mb θ_w; second, they analyzed trends for 1989-2009, as opposed to 1979-2018.

The second previous study is RS15, who analyzed trends in both the winter and summer. As mentioned in Section 8.3, their method is known to rarely detect warm fronts, so this discussion will assume that their results include cold fronts only. In the winter (their Figure 1a) they found a northward shift in cold fronts over the Pacific, an increase over the southern CONUS, a decrease over much of the Canadian Arctic and sub-Arctic, and an increase over the western Atlantic from 30-40 °N. My results are broadly consistent (Figure 8.11a), with two exceptions. First, RS15 have more significant grid points, which may be caused by different methods for controlling the false-discovery rate. Second, they do not

have a decreasing swath south of the increasing swath in the Atlantic, which may be due to edge effects, as their domain terminates around 25 °N. Meanwhile, in the summer RS15 found mostly decreases in cold-front frequency (their Figure 1b). This conflicts with my results (Figure 8.11e), which do not suggest an overall decrease. However, in areas where RS15 found a significant decrease, my results suggest a decrease as well. These include the Pacific around 35 °N and 175 °W, the Atlantic southeast of the Canadian maritimes, and a large part of northern Canada. Decreases in northern Canada are consistent with Arctic amplification of global warming (Serreze and Barry 2011), reducing thermal gradients between the Arctic and mid-latitudes.

8.5 Sensitivity Tests

This section briefly examines the sensitivity of the climatology to two hyperparameters of the object-conversion method (Section 5.2.2): minimum frontal-zone length and minimum CNN probability. Sensitivity is not investigated for hyperparameters of the CNN itself, because this would require applying several CNNs to the 40-year period, which is too computationally expensive, as discussed at the beginning of Section 8.1.

Figure 8.12 shows the response of front frequency to strong winter El Niño, based on different minimum lengths: 200, 400, and 600 km, where 200 km is the selected value used throughout this chapter. The minimum length makes little difference, as the vast majority of CNN-detected fronts are longer than 600 km. Figure 8.13 shows analogous results for different probability thresholds: 0.65 (the selected value), 0.50, and 0.80. In all cases the warm- and cold-front thresholds are the same ($p_W^* = p_C^*$ in Equation 5.4). Results are broadly similar among the three thresholds, except that higher thresholds eliminate more fronts outside of the mid-latitude cyclone track. This is likely because the strongest fronts tend to be associated with mid-latitude cyclones.

In summary, the results shown are almost completely insensitive to minimum front length, while results outside of the mid-latitude cyclone track are somewhat sensitive to

minimum probability. The same conclusions hold for analyses other than strong winter El Niño, which for the sake of brevity are not shown. Results might be more sensitive to hyperparameters of the CNN itself, especially the predictor variables. For example, if the predictors did not include humidity, I suspect that the CNN would no longer detect fronts in the eastern tropical Pacific, as in the experiment of B11a (their Figure 4).

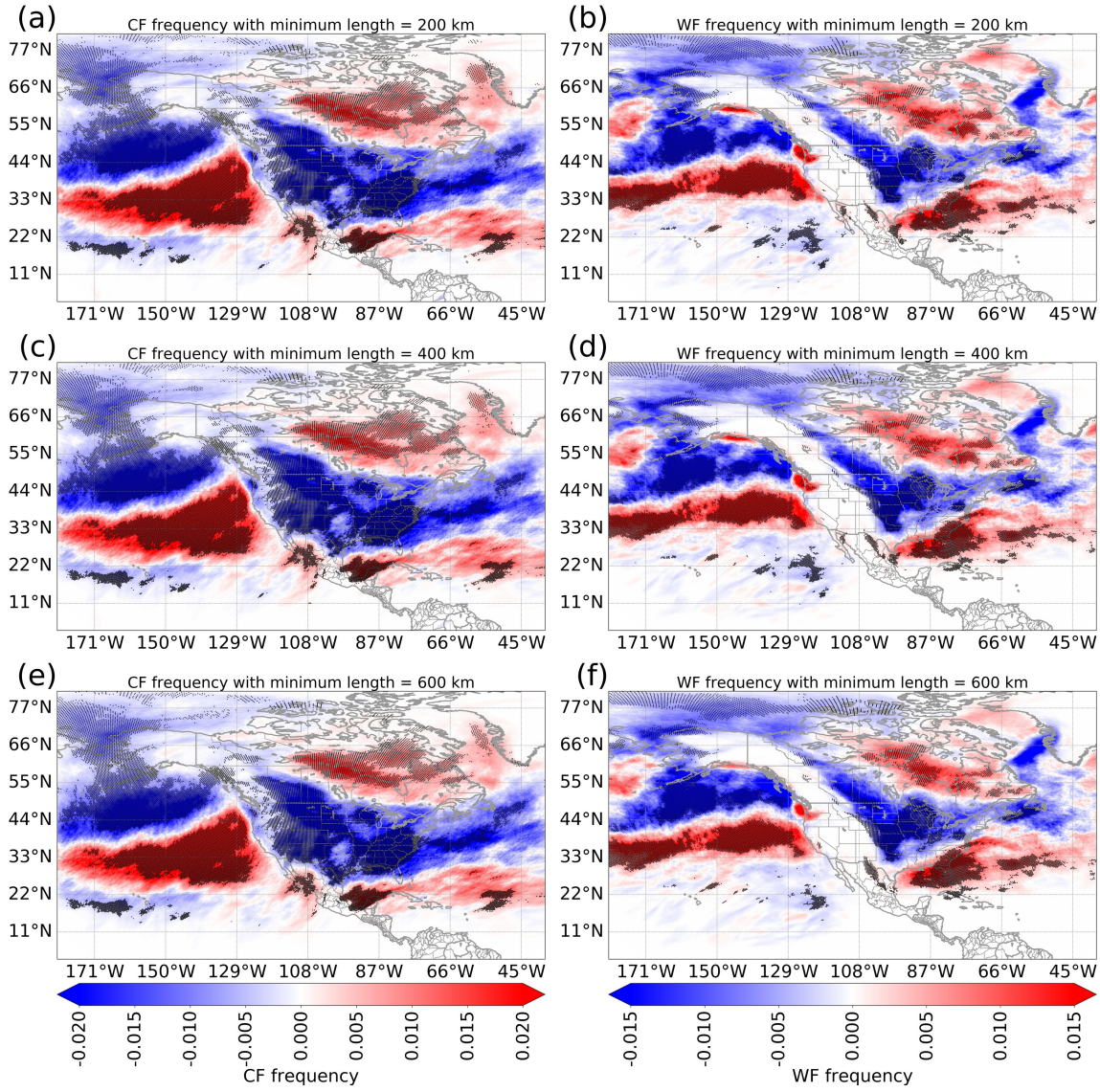


Figure 8.12: Front frequency during strong winter El Niño for different minimum lengths. Each panel shows the composite difference (frequency during strong El Niño minus frequency during ENSO-neutral). Stippling is explained in Figure 8.6. [a] Composite difference in cold-front frequency, based on minimum length of 200 km. [b] Same but for warm fronts. [c-d] Same as a-b but for minimum length of 400 km. [e-f] Same as a-b but for minimum length of 600 km.

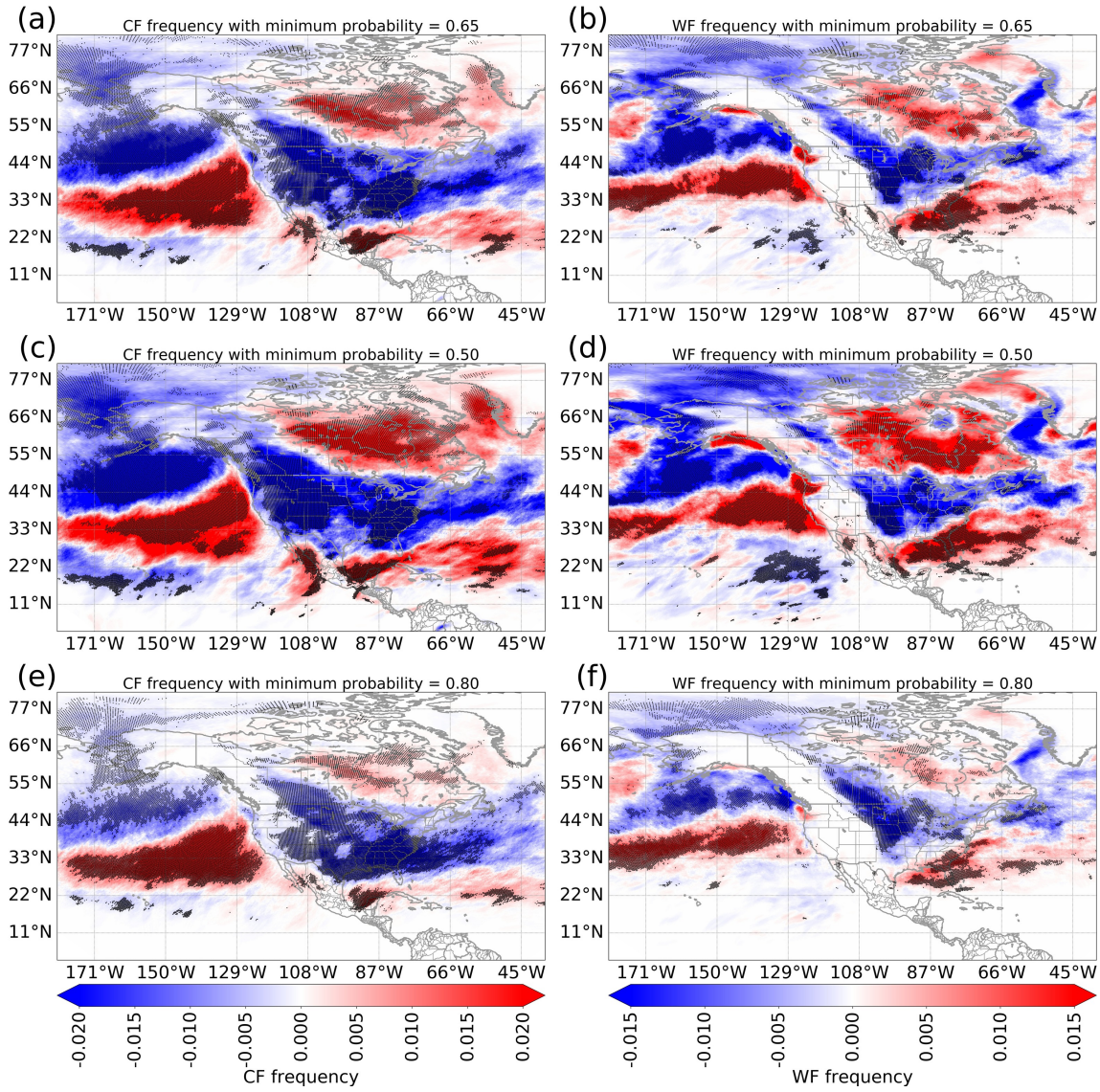


Figure 8.13: Front frequency during strong winter El Niño for different minimum CNN probabilities. Formatting is explained in Figure 8.12. [a] Composite difference in cold-front frequency, based on minimum probability of 0.65. [b] Same but for warm fronts. [c-d] Same as a-b but for minimum probability of 0.50. [e-f] Same as a-b but for minimum probability of 0.80.

Chapter 9

Conclusions

I developed and tested convolutional neural networks (CNN), a type of deep-learning method, for the tasks of next-hour tornado prediction and front detection. I conducted hyperparameter experiments to find the best CNN for each task, and I evaluated the selected models on independent testing data. I used various interpretation methods to understand physical relationships learned by the models, as well as sanity checks to ensure that these relationships are significant. Finally, I created a 40-year climatology of CNN-detected warm fronts, analyzed their variability with respect to the El Niño – Southern Oscillation (ENSO), and analyzed their long-term trends. The rest of this chapter discusses results and suggestions for future work.

For tornado prediction, I trained CNNs with radar data from either MYRORSS or GridRad, along with proximity soundings from the Rapid Refresh model. Both the MYRORSS and GridRad models performed best when trained with data augmentation, the practice of applying small perturbations to the predictor fields while assuming that the label (here, tornadic or non-tornadic) remains the same. When evaluated on independent testing data (from the year 2011, used for neither training nor hyperparameter-tuning), the MYRORSS model achieved an AUC of 0.97 and critical success index (CSI) of 0.17, while the GridRad model achieved an AUC of 0.93 and CSI of 0.31. These differences are caused by the presence of many more trivial nulls (easy-to-predict non-tornadic storms) in the MYRORSS dataset. Trivial nulls increase the AUC but decrease the CSI, which is sensitive to event frequency. Specifically, testing data for MYRORSS and GridRad have event frequencies of 0.22% and 3.43%, respectively. When evaluated on matched storms in the testing data (yielding a 3.19% event frequency for both models), the models have statistically identical AUCs and nearly identical CSIs (both $\gtrsim 0.30$). Also, results for strong

tornadoes (EF-2+) are slightly better than for all tornadoes, which suggests that model skill increases with tornado strength. Comparison to ProbSevere, a ML model currently used for operational severe-weather prediction, suggests that my models would be competitive in an operational setting. Work is currently planned to operationalize the models, discussed later in this chapter. For both the MYRORSS and GridRad models, the best hits are tornadic supercells; the worst false alarms are non-tornadic supercells; the worst misses are mostly cells in quasi-linear convective systems (QLCS), while some are early-stage supercells that subsequently undergo rapid intensification and tornadogenesis; and the best correct nulls are mostly non-dominant cells in mesoscale convective systems (MCS). The worst false alarms are mostly associated with NWS tornado warnings, which suggests that they are a failure mode for humans as well.

I applied four interpretation methods to both the MYRORSS and GridRad models. First, the permutation test suggested that vorticity and reflectivity are the most important GridRad variables; low-level azimuthal shear (0.5 $\times$ maximum vorticity from 0-2 km above ground) is the most important MYRORSS variable; and for both models the radar image is much more important than the proximity sounding. Second, saliency maps for both models highlighted the importance of a deep reflectivity core and strong mesocyclone. Also, saliency maps for the GridRad model suggested that tornadoes are more likely for discrete storms, while those for the MYRORSS model suggested that tornado probability decreases with low-level reflectivity in the rear-flank downdraft (RFD). This is consistent with previous work, which suggested that RFDs with too much negative buoyancy can lead to tornadogenesis failure. Third, class-activation maps (CAM) for supercells (the best hits and worst false alarms) highlighted the importance of the right-rear flank, while class activation for non-supercells (the worst misses and best correct nulls) generally followed the highest vorticity. Fourth, backward optimization suggested that the most important parts of the sounding are low-level instability and wind shear; the latter has also been highlighted in recent literature. Finally, sanity checks showed that saliency maps for both

models cannot be trivially replicated. This and the models' good performance suggest that interpretation results reflect useful physical relationships learned by the models.

For front detection, the labels were fronts drawn by meteorologists at the Weather Prediction Center (WPC); the predictors were ERA5 fields on a 33×33 patch; and the task was to determine front type at the center grid cell (no front, warm front, or cold front). The models generally performed best when trained with data from two vertical levels instead of one, especially the surface plus 850 or 900 mb. The selected model was trained with u-wind, v-wind, temperature, and specific humidity at the surface and 850 mb. I used an object-conversion method to turn the CNN output from gridded probabilities into deterministic frontal zones. When evaluated on independent testing data with a 100-km neighbourhood distance, the frontal zones achieved a probability of detection (POD) of 0.69, false-alarm ratio (FAR) of 0.66, CSI of 0.29, and frequency bias of 2.03. Although the FAR and frequency bias are high, visual analysis suggests that the WPC often misses fronts, so I consider these values acceptable. The best hits (WPC fronts with high CNN probabilities) are generally associated with extratropical cyclones; the worst misses are generally associated with weak flow in anticyclones and sometimes deformation zones, leading to weak thermal advection; and the best correct nulls occur under subtropical high pressure with nearly uniform surface temperatures except at land-sea boundaries.

I also applied interpretation methods to the front-detection model. First, the permutation test suggested that the most important predictors are surface wind (both u and v) and 850-mb temperature. Second, CAMs for warm-front patches were best aligned with frontal features at 850 mb, while CAMs for cold-front patches were best aligned with frontal features at the surface. This suggests that vertical continuity is more important for warm fronts, probably because they tend to have weaker thermal gradients. Also, for the "worst misses" (where the WPC identifies a front at the center grid cell but the CNN does not), class activation is maximized away from the center grid cell, in the area that subjectively appears most frontal. This indicates that the CNN has not merely mimicked, and has potentially

improved upon, its training data. Third, backward optimization reduces cold-front probability by reducing thermal advection and cyclonic wind near the center grid cell, while it increases cold-front probability by creating two converging cold fronts. The second result is physically unrealistic and points to the need for better physical constraints in backward optimization (which may include using unsquared terms in the loss function, as discussed at the end of Section 7.1.4). Finally, as for the tornado models, sanity checks showed that saliency maps for front detection are significantly different from noise.

I used CNN-detected fronts to create a climatology from 1979 to 2018. According to averages over the 40-year period, fronts are most common in the mid-latitude cyclone track; warm fronts generally occur north of cold fronts, consistent with mean frontal positions relative to their parent cyclones; and fronts in the mid-latitude cyclone track migrate $\sim 10^{\circ}$ poleward from winter to summer, due to the global annual heating cycle. Although these large-scale results are not new, they provide a useful sanity check that lends credence to the remaining results. The 40-year averages also include many regional effects, such as cold-air damming in the lee of the Appalachians; warm and cold fronts near land-sea boundaries, due to differences in heat capacity and the annual temperature cycle; and warm fronts northwest of Hawaii and in the tropical eastern Pacific, characterized mainly by gradients in moisture rather than temperature.

I also investigated ENSO-relative variability and long-term trends in front frequency. ENSO effects are most pronounced in the winter and spring, when the associated sea-surface temperature anomalies are largest and teleconnections in general are strongest. Also, El Niño has greater effects than La Niña, consistent with previous work on general teleconnections. El Niño (La Niña) leads to a southward (northward) displacement of the mid-latitude cyclone track and associated fronts, due to a southward (northward) displacement of the subtropical Pacific jet. During winter (and to a lesser extent, the spring and fall), El Niño also causes an eastward extension of the subtropical jet into the Atlantic, enhancing cyclone and front activity in the southern CONUS. These results are broadly

consistent with RS15 but not HH03, who concluded that ENSO does not significantly impact winter fronts in the southern CONUS. ENSO signals in the summer and fall are mostly insignificant, but they are consistent in suggesting a southward (northward) migration of the cyclone track during El Niño (La Niña). Some regional patterns require future investigation. For example, in the tropical eastern Pacific, El Niño (La Niña) generally leads to a northward (southward) shift in warm fronts, but the pattern is not seasonally consistent. As for long-term trends, the mid-latitude cyclone track appears to have moved northward over the last 40 years, consistent with previous studies. However, my trend results are mostly insignificant, due partly to using an overly conservative method to control the false-discovery rate.

Future work on tornado prediction should address three issues. The first is prediction of QLCS tornadoes, which are a common failure mode for humans, ML, and other automated methods. This might require radar data with higher spatial or temporal resolution, especially in the vertical dimension, to resolve the shallow mesovortices that usually precede QLCS tornadoes. The second issue, which applies to all ML applications in atmospheric science, is to ensure that the model respects physical laws. It is unclear how this would be done for a model that predicts only one scalar (*e.g.*, tornado probability), but it would be a key step in building trustworthy ML. For example, Beucler et al. (2019a,b) have built custom "constraint layers" into a neural network that make it satisfy various conservation laws. Third, the tornado models developed herein (or similar models) should be operationalized, so that they can be evaluated in real-time and potentially aid in forecaster decision-making. The models could be integrated into the Probabilistic Hazard Information tool (PHI; Karstens et al. 2018), which already displays storm-by-storm output from several ML models and creates warning polygons to determine the spatial extent of the threat. The predictions could be accompanied by interpretation maps, such as a saliency or class-activation map for each storm. Cintineo et al. (2018) have discovered that even simple attempts to explain ML predictions, like showing the raw predictor values (their

Figure 1), increases forecaster trust. Also, interpretation maps could help the forecasters to identify failure modes (*e.g.*, cases where the maps are physically implausible), where the ML needs to be overridden by human expertise. A proposal to operationalize the tornado models has recently been recommended for funding. Another proposal, currently under review, involves operationalization and "trust studies," where both predictions and interpretation maps from the CNN would be judged by humans. To operationalize the models developed herein, they would need to be applied to MRMS, which is the real-time and less quality-controlled version of MYRORSS. The two datasets may be similar enough that the MYRORSS model could be applied to MRMS without retraining. Otherwise, acquiring enough MRMS data for retraining could be a significant hurdle, as discussed in Section 3.1.1.

Future work on front detection should address the following issues. The first is computational sustainability: as discussed at the beginning of Section 5.2.1, it takes $\sim$20 minutes to apply the CNN to a full grid. The solution may involve switching from patch classification to semantic segmentation, or making patch classification more efficient by saving intermediate calculations and reusing them for nearby grid cells. The second issue is investigating variability with respect to climate oscillations other than ENSO, such as the Arctic Oscillation and North Atlantic Oscillation. Third, one could use the WPC bulletins to investigate the climatology of occluded and stationary fronts, which to my knowledge has never been done. Fourth, one could investigate the climatological link between fronts and high-impact weather phenomena, such as thunderstorms and precipitation. Previous attribution studies (Fawbush and Miller 1954; Miller 1959; Garner 2013; Catto and Pfahl 2013; Catto et al. 2015) have used NFA or manual analysis and have not differentiated between front types. I believe that my 40 years of CNN-detected fronts could be used to learn more about these links. Fifth, I believe that combining ML with a 3-D definition of fronts would allow more to be learned about fronts and their climatology, especially upper-level fronts.

Sixth, a proposal recently recommended for funding will operationalize the front-detection model developed herein for use by forecasters.

My contributions are summarized as follows. I have demonstrated that novel ML methods can be used to model complex phenomena with uncertain predictors and labels, unlike the curated datasets (*e.g.*, photographs of cats and dogs) typically used to develop the methods. I have created tornado-prediction models that perform competitively with a model currently used in operations, and I have identified several relationships – such as those involving the RFD, low-level wind shear, and QLCS tornadoes as a failure mode – that appear in numerical-modeling and observational studies. Discovering new fundamental knowledge about tornadoes would likely require a large amount of high-resolution radar (*e.g.*, mobile radar) and environmental data, as the problem has been extensively studied. Also, I have created a front-detection model that appears to overcome several deficiencies of the human labels used to train it. I have used the model to create a climatology, which I have compared to the few pre-existing climatologies, identifying several discrepancies that should be targeted for future investigation. Thus, for two very different weather phenomena, I have demonstrated that deep learning can advance both prediction and understanding.

Bibliography

Adebayo, J., J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, 2018: Sanity checks for saliency maps. *Conference on Neural Information Processing Systems*, Montréal, Canada, Neural Information Processing Systems Foundation, URL http://papers.nips.cc/paper/8160-sanity-checks-for-saliency-maps.

Adrianto, I., T. Trafalis, and V. Lakshmanan, 2009: Support vector machines for spatiotemporal tornado prediction. *International Journal of General Systems*, **38 (7)**, 759–776, URL https://doi.org/10.1080/03081070601068629.

Agee, E., and E. Jones, 2009: Proposed conceptual taxonomy for proper identification and classification of tornado events. *Weather and Forecasting*, **24 (2)**, 609–617, URL https://doi.org/10.1175/2008WAF2222163.1.

Allen, J., M. Tippett, and A. Sobel, 2015: Influence of the El Niño/Southern Oscillation on tornado and hail frequency in the United States. *Nature Geoscience*, **8 (4)**, 278, URL https://doi.org/10.1038/ngeo2385.

AMS, 2012a: Front – AMS Glossary. URL http://glossary.ametsoc.org/wiki/Front, accessed: 2020-02-26.

AMS, 2012b: Tornado outbreak – AMS Glossary. URL http://glossary.ametsoc.org/wiki/Tornado_outbreak, accessed: 2020-04-16.

AMS, 2013: Tornado – AMS Glossary. URL http://glossary.ametsoc.org/wiki/Tornado, accessed: 2020-03-02.

Anderson, C., C. Wikle, Q. Zhou, and J. Royle, 2007: Population influences on tornado reports in the United States. *Weather and Forecasting*, **22 (3)**, 571–579, URL https://doi.org/10.1175/WAF997.1.

Anderson-Frey, A., Y. Richardson, A. Dean, R. Thompson, and B. Smith, 2016: Investigation of near-storm environments for tornado events and warnings. *Weather and Forecasting*, **31 (6)**, 1771–1790, URL https://doi.org/10.1175/WAF-D-16-0046.1.

Atkins, N., and M. St. Laurent, 2009: Bow echo mesovortices. Part II: Their genesis. *Monthly Weather Review*, **137 (5)**, 1514–1532, URL https://doi.org/10.1175/2008MWR2650.1.

Benjamin, S., and Coauthors, 2004: An hourly assimilation-forecast cycle: The RUC. *Monthly Weather Review*, **132 (2)**, 495–518, URL https://doi.org/10.1175/1520-0493(2004)132%3C0495:AHACTR%3E2.0.CO;2.

Benjamin, S., and Coauthors, 2016: A North American hourly assimilation and model forecast cycle: The Rapid Refresh. *Monthly Weather Review*, **144 (4)**, 1669–1694, URL https://doi.org/10.1175/MWR-D-15-0242.1.

Berry, G., C. Jakob, and M. Reeder, 2011a: Recent global trends in atmospheric fronts. *Geophysical Research Letters*, **38 (21)**, URL https://doi.org/10.1029/2011GL049481.

Berry, G., M. Reeder, and C. Jakob, 2011b: A global climatology of atmospheric fronts. *Geophysical Research Letters*, **38 (4)**, URL https://doi.org/10.1029/2010GL046451.

Beucler, T., M. Pritchard, S. Rasp, P. Gentine, J. Ott, and P. Baldi, 2019a: Enforcing analytic constraints in neural-networks emulating physical systems. *arXiv pre-prints*, **1909 (00912)**, URL https://arxiv.org/abs/1909.00912.

Beucler, T., S. Rasp, M. Pritchard, and P. Gentine, 2019b: Achieving conservation of energy in neural network emulators for climate modeling. *arXiv pre-prints*, **1906 (06622)**, URL https://arxiv.org/abs/1906.06622.

Bjerknes, J., and H. Solberg, 1922: Life cycle of cyclones and the polar front theory of atmospheric circulation. *Geofysiske Publikationer*, **3 (1)**, 1–18.

Bluestein, H., 1993: *Synoptic-Dynamic Meteorology in Midlatitudes: Observations and Theory of Weather Systems*, Vol. 2. Oxford University Press.

Bosart, L., V. Pagnotti, and B. Lettau, 1973: Climatological aspects of eastern United States back-door cold frontal passages. *Monthly Weather Review*, **101 (8)**, 627–635, URL https://doi.org/10.1175/1520-0493(1973)101%3C0627:CAOEUS%3E2.3.CO;2.

Bowman, K., and C. Homeyer, 2017: GridRad - Three-Dimensional Gridded NEXRAD WSR-88D Radar Data. Research Data Archive at the National Center for Atmospheric Research, Computational and Information Systems Laboratory, http://rda.ucar.edu/datasets/ds841.0.

Brandes, E., 1984: Relationships between radar-derived thermodynamic variables and tornadogenesis. *Monthly Weather Review*, **112 (5)**, 1033–1052, URL https://doi.org/10.1175/1520-0493(1984)112%3C1033:RBRDTV%3E2.0.CO;2.

Breiman, L., 2001: Random forests. *Machine Learning*, **45 (1)**, 5–32, URL https://link.springer.com/article/10.1023%2FA%3A1010933404324.

Brooks, H., 2004: Tornado-warning performance in the past and future: A perspective from signal detection theory. *Bulletin of the American Meteorological Society*, **85 (6)**, 837–843, URL https://doi.org/10.1175/BAMS-85-6-837.

Brooks, H., and J. Correia, 2018: Long-term performance metrics for National Weather Service tornado warnings. *Weather and Forecasting*, **33 (6)**, 1501–1511, URL https://doi.org/10.1175/WAF-D-18-0120.1.

Brotzge, J., S. Nelson, R. Thompson, and B. Smith, 2013: Tornado probability of detection and lead time as a function of convective mode and environmental parameters. *Weather and Forecasting*, **28 (5)**, 1261–1276, URL https://doi.org/10.1175/WAF-D-12-00119.1.

Buell, C., 1979: On the physical interpretation of empirical orthogonal functions. *Conference on Probability and Statistics*, Banff, Canada, American Meteorological Society.

Butler, A., L. Polvani, and C. Deser, 2014: Separating the stratospheric and tropospheric pathways of El Niño – Southern Oscillation teleconnections. *Environmental Research Letters*, **9 (2)**, 024 014, URL http://dx.doi.org/10.1088/1748-9326/9/2/024014.

Caruso, J., and J. Davies, 2005: Tornadoes in nonmesocyclone environments with preexisting vertical vorticity along convergence boundaries. *Journal of Operational Meteorology*, **6 (4)**, 1–36, URL http://nwafiles.nwas.org/ej/pdf/2005-EJ4.pdf.

Catto, J., E. Madonna, H. Joos, I. Rudeva, and I. Simmonds, 2015: Global relationship between fronts and warm conveyor belts and the impact on extreme precipitation. *Journal of Climate*, **28 (21)**, 8411–8429, URL https://doi.org/10.1175/JCLI-D-15-0171.1.

Catto, J., and S. Pfahl, 2013: The importance of fronts for extreme precipitation. *Journal of Geophysical Research: Atmospheres*, **118 (19)**, 10 791–10 801, URL https://doi.org/10.1002/jgrd.50852.

Chollet, F., 2018: *Deep Learning with Python*. Manning, Shelter Island, New York.

Chollet, F., and Coauthors, 2020: Keras. GitHub, URL https://github.com/fchollet/keras.

Cintineo, J., M. Pavolonis, J. Sieglaff, and D. Lindsey, 2014: An empirical model for assessing the severe weather potential of developing convection. *Weather and Forecasting*, **29 (3)**, 639–653, URL https://doi.org/10.1175/WAF-D-13-00113.1.

Cintineo, J., and Coauthors, 2018: The NOAA/CIMSS ProbSevere Model: Incorporation of total lightning and validation. *Weather and Forecasting*, **33 (1)**, 331–345, URL https://doi.org/10.1175/WAF-D-17-0099.1.

Clark, A., and Coauthors, 2012: An overview of the 2010 Hazardous Weather Testbed Experimental Forecast Program Spring Experiment. *Bulletin of the American Meteorological Society*, **93 (1)**, 55–74, URL https://doi.org/10.1175/BAMS-D-11-00040.1.

Climate Prediction Center, 2020: Monthly ERSSTv5 (1981-2010 base period) Niño 3.4 (5° North - 5° South)(170-120° West). Accessed: 2020-02-12, https://www.cpc.ncep.noaa.gov/data/indices/ersst5.nino.mth.81-10.ascii.

Cook, A., and J. Schaefer, 2008: The relation of El Niño – Southern Oscillation (ENSO) to winter tornado outbreaks. *Monthly Weather Review*, **136 (8)**, 3121–3137, URL https://doi.org/10.1175/2007MWR2171.1.

Crum, T., and R. Alberty, 1993: The WSR-88D and the WSR-88D operational support facility. *Bulletin of the American Meteorological Society*, **74 (9)**, 1669–1687, URL https://doi.org/10.1175/1520-0477(1993)074%3C1669:TWATWO%3E2.0.CO;2.

Davies, J., 2006: Tornadoes with cold core 500-mb lows. *Weather and Forecasting*, **21 (6)**, 1051–1062, URL https://doi.org/10.1175/WAF967.1.

Davies-Jones, R., 1984: Streamwise vorticity: The origin of updraft rotation in supercell storms. *Journal of the Atmospheric Sciences*, **41 (20)**, 2991–3006, URL https://doi.org/10.1175/1520-0469(1984)041%3C2991:SVTOOU%3E2.0.CO;2.

Davies-Jones, R., 2017: Roles of streamwise and transverse partial-vorticity components in steady inviscid isentropic supercell-like flows. *Journal of the Atmospheric Sciences*, **74 (9)**, 3021–3041, URL https://doi.org/10.1175/JAS-D-16-0332.1.

Davies-Jones, R., and H. Brooks, 1993: Mesocyclogenesis From a Theoretical Perspective. *The Tornado: Its Structure, Dynamics, Prediction, and Hazards*, C. Church, D. Burgess, C. Doswell, and R. Davies-Jones, Eds., American Geophysical Union, Washington, District of Columbia, 105–114, URL https://doi.org/10.1029/GM079p0105.

Davies-Jones, R., R. Trapp, and H. Bluestein, 2001: Tornadoes and Tornadic Storms. *Severe Convective Storms*, C. Doswell, Ed., American Meteorological Society, Boston, Massachusetts, 167–222, URL https://link.springer.com/chapter/10.1007/978-1-935704-06-5_5.

Davis, S., and K. Rosenlof, 2012: A multidiagnostic intercomparison of tropical-width time series using reanalyses and satellite observations. *Journal of Climate*, **25 (4)**, 1061–1078, URL https://doi.org/10.1175/JCLI-D-11-00127.1.

Dieleman, S., K. Willett, and J. Dambre, 2015: Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Nature*, **450 (2)**, 1441–1459, URL https://doi.org/10.1093/mnras/stv632.

Doswell, C., A. Moller, and H. Brooks, 1999: Storm spotting and public awareness since the first tornado forecasts of 1948. *Weather and Forecasting*, **14 (4)**, 544–557, URL https://doi.org/10.1175/1520-0434(1999)014%3C0544:SSAPAS%3E2.0.CO;2.

Dubey, A., and V. Jain, 2019: Comparative study of convolution neural network's ReLU and leaky-ReLU activation functions. *Applications of Computing, Automation and Wireless Systems in Electrical Engineering*, 873–880, URL https://doi.org/10.1007/978-981-13-6772-4_76.

Dwass, M., 1957: Modified randomization tests for nonparametric hypotheses. *The Annals of Mathematical Statistics*, **28 (1)**, 181–187, URL https://www.jstor.org/stable/2237031.

Ebert, E., 2001: Ability of a poor man's ensemble to predict the probability and distribution of precipitation. *Monthly Weather Review*, **129 (10)**, 2461–2480, URL https://doi.org/10.1175/1520-0493(2001)129%3C2461:AOAPMS%3E2.0.CO;2.

Eichler, T., and W. Higgins, 2006: Climatology and ENSO-related variability of North American extratropical cyclone activity. *Journal of Climate*, **19 (10)**, 2076–2093, URL https://doi.org/10.1175/JCLI3725.1.

Elsner, J., L. Michaels, K. Scheitlin, and I. Elsner, 2013: The decreasing population bias in tornado reports across the central Plains. *Weather, Climate, and Society*, **5 (3)**, 221–232, URL https://doi.org/10.1175/WCAS-D-12-00040.1.

Erhan, D., Y. Bengio, A. Courville, and P. Vincent, 2009: Visualizing higher-layer features of a deep network. Tech. rep., https://www.researchgate.net/profile/Aaron_Courville/publication/265022827_Visualizing_Higher-Layer_Features_of_a_Deep_Network/links/53ff82b00cf24c81027da530.pdf.

Fawbush, E., and R. Miller, 1954: The types of airmasses in which North American tornadoes form. *Bulletin of the American Meteorological Society*, **35 (4)**, 154–165, URL https://doi.org/10.1175/1520-0477-35.4.154.

Friedman, J., 2001: Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, **29 (5)**, 1189–1232, URL https://www.jstor.org/stable/2699986.

Fukushima, K., and S. Miyake, 1982: Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition*, **15 (6)**, 455–469, URL http://www.cs.cmu.edu/afs/cs/user/bhiksha/WWW/courses/deeplearning/Fall.2016/pdfs/Fukushima_Miyake.pdf.

Gagne, D., S. Haupt, and D. Nychka, 2019: Interpretable deep learning for spatial analysis of severe hailstorms. *Monthly Weather Review*, **147 (8)**, 2827–2845, URL https://doi.org/10.1175/MWR-D-18-0316.1.

Gagne, D., A. McGovern, J. Basara, and R. Brown, 2012: Tornadic supercell environments analyzed using surface and reanalysis data: A spatiotemporal relational data-mining approach. *Journal of Applied Meteorology and Climatology*, **51 (12)**, 2203–2217, URL https://doi.org/10.1175/JAMC-D-11-060.1.

Gallo, B., A. Clark, and S. Dembek, 2016: Forecasting tornadoes using convection-permitting ensembles. *Weather and Forecasting*, **31 (1)**, 273–295, URL https://doi.org/10.1175/WAF-D-15-0134.1.

Gallo, B., and Coauthors, 2017: Breaking new ground in severe weather prediction: The 2015 NOAA/Hazardous Weather Testbed Spring Forecasting Experiment. *Weather and Forecasting*, **32 (4)**, 1541–1568, URL https://doi.org/10.1175/WAF-D-16-0178.1.

Garner, J., 2013: A study of synoptic-scale tornado regimes. *Electronic Journal of Severe Storms Meteorology*, **8 (3)**, 1–25, URL https://ejssm.org/ojs/index.php/ejssm/article/viewArticle/119.

Gil, Y., and Coauthors, 2019: Intelligent systems for geosciences: An essential research agenda. *Communications of the Association for Computing Machinery*, **62 (1)**, 76–84, URL https://dl.acm.org/doi/10.1145/3192335.

Goodfellow, I., Y. Bengio, and A. Courville, 2016: *Deep Learning*. MIT Press, URL https://www.deeplearningbook.org.

Goodfellow, I., J. Shlens, and C. Szegedy, 2014: Explaining and harnessing adversarial examples. *arXiv pre-prints*, **1412 (6572)**, URL https://arxiv.org/abs/1412.6572.

Gossart, A., S. Helsen, J. Lenaerts, S. Vanden Broucke, N.P.M. van Lipzig, and N. Souverijns, 2019: An evaluation of surface climatology in state-of-the-art reanalyses over the Antarctic Ice Sheet. *Journal of Climate*, **32 (20)**, 6899–6915, URL https://doi.org/10.1175/JCLI-D-19-0030.1.

Hardy, J., and K. Henderson, 2003: Cold front variability in the southern United States and the influence of atmospheric teleconnection patterns. *Physical Geography*, **24 (2)**, 120–137, URL https://doi.org/10.2747/0272-3646.24.2.120.

Harrison, D., and C. Karstens, 2017: A climatology of operational storm-based warnings: A geospatial analysis. *Weather and Forecasting*, **32 (1)**, 47–60, URL https://doi.org/10.1175/WAF-D-15-0146.1.

Harvard University, 2020: Hadley Cells. URL https://www.seas.harvard.edu/climate/eli/research/equable/hadley.html, accessed: 2020-03-02.

Hersbach, H., 2016: The ERA5 atmospheric reanalysis. *AGU Fall Meeting*, San Francisco, California, American Geophysical Union, URL https://ui.adsabs.harvard.edu/abs/2016AGUFMNG33D..01H/abstract.

Hersbach, H., and D. Dee, 2016: ERA5 reanalysis is in production. *ECMWF Newsletter*, **147 (7)**, 5–6, URL https://www.ecmwf.int/en/newsletter/147/news/era5-reanalysis-production.

Hewson, E., 1936: The application of wet-bulb potential temperature to air mass analysis. *Quarterly Journal of the Royal Meteorological Society*, **62 (266)**, 387–420, URL https://doi.org/10.1002/qj.49706226604.

Hewson, T., 1998: Objective fronts. *Meteorological Applications*, **5 (1)**, 37–65, URL https://doi.org/10.1017/S1350482798000553.

Hinton, G., N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, 2012: Improving neural networks by preventing co-adaptation of feature detectors. *arXiv pre-prints*, **1207 (0580)**, URL https://arxiv.org/abs/1207.0580.

Hobbs, P., J. Locatelli, and J. Martin, 1996: A new conceptual model for cyclones generated in the lee of the Rocky Mountains. *Bulletin of the American Meteorological Society*, **77 (6)**, 1169–1178, URL https://doi.org/10.1175/1520-0477(1996)077%3C1169:ANCMFC%3E2.0.CO;2.

Hodges, K., R. Lee, and L. Bengtsson, 2011: A comparison of extratropical cyclones in recent reanalyses ERA-Interim, NASA MERRA, NCEP CFSR, and JRA-25. *Journal of Climate*, **24 (18)**, 4888–4906, URL https://doi.org/10.1175/2011JCLI4097.1.

Homeyer, C., and K. Bowman, 2017: Algorithm Description Document for Version 3.1 of the Three-Dimensional Gridded NEXRAD WSR-88D Radar (GridRad) Dataset. Tech. rep., University of Oklahoma. URL http://gridrad.org/pdf/GridRad-v3.1-Algorithm-Description.pdf.

Homeyer, C., J. McAuliffe, and K. Bedka, 2017: On the development of above-anvil cirrus plumes in extratropical convection. *Journal of the Atmospheric Sciences*, **74 (5)**, 1617–1633.

Horel, J., and J. Wallace, 1981: Planetary-scale atmospheric phenomena associated with the Southern Oscillation. *Monthly Weather Review*, **109 (4)**, 813–829, URL https://doi.org/10.1175/1520-0493(1981)109%3C0813:PSAPAW%3E2.0.CO;2.

Hunter, J., 2007: Matplotlib: A 2D graphics environment. *Computing in Science and Engineering*, **9 (3)**, 90–95, URL https://zenodo.org/record/3264781#.XXWyWHVKgUw.

Hussain, M., and I. Mamhud, 2019: pyMannKendall: A Python package for non-parametric Mann-Kendall family of trend tests. *Journal of Open Source Software*, **4 (39)**, 1556, URL https://doi.org/10.21105/joss.01556.

Insurance Information Institute, 2019: Facts + Statistics: Tornadoes and Thunderstorms. URL https://www.iii.org/fact-statistic/facts-statistics-tornadoes-and-thunderstorms.

Ioffe, S., and C. Szegedy, 2015: Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, Lille, France, International Machine Learning Society, URL https://arxiv.org/abs/1502.03167.

Jergensen, G., A. McGovern, R. Lagerquist, and T. Smith, 2020: Classifying convective storms using machine learning. *Weather and Forecasting*, **35 (4)**, 537–559, URL https://doi.org/10.1175/WAF-D-19-0170.1.

Kain, J., and Coauthors, 2008: Some practical considerations regarding horizontal resolution in the first generation of operational convection-allowing NWP. *Weather and Forecasting*, **23 (5)**, 931–952, URL https://doi.org/10.1175/WAF2007106.1.

Karstens, C., and Coauthors, 2018: Development of a human-machine mix for forecasting severe convective events. *Weather and Forecasting*, **33 (3)**, 715–737, URL https://doi.org/10.1175/WAF-D-17-0188.1.

Kendall, M., 1975: *Rank Correlation Measures*. Charles Griffin, London, United Kingdom.

Kingma, D., and J. Ba, 2017: Adam: A method for stochastic optimization. *arXiv preprints*, **1412 (6980v8)**, URL https://arxiv.org/abs/1412.6980v8.

Kitzmiller, D., W. McGovern, and R. Saffle, 1995: The WSR-88D severe weather potential algorithm. *Weather and Forecasting*, **10 (1)**, 141–159, URL https://doi.org/10.1175/1520-0434(1995)010%3C0141:TWSWPA%3E2.0.CO;2.

Krizhevsky, A., I. Sutskever, and G. Hinton, 2012: Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 1097–1105, URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networ.

Kunkel, K., J. Biard, and E. Racah, 2018: Automated detection of fronts using a deep learning algorithm. *Conference on Artificial and Computational Intelligence and its Applications to the Environmental Sciences*, Austin, Texas, American Meteorological Society, URL https://ams.confex.com/ams/98Annual/meetingapp.cgi/Paper/333480.

Kurth, T., and Coauthors, 2018: Exascale deep learning for climate analytics. *International Conference for High Performance Computing, Networking, Storage, and Analysis*, Dallas, Texas, Institute of Electrical and Electronics Engineers (IEEE), URL https://doi.org/10.1109/SC.2018.00054.

Lagerquist, R., J. Allen, and A. McGovern, 2020a: Climatology and variability of warm and cold fronts over North America from 1979-2018. *Journal of Climate*, **conditionally accepted**.

Lagerquist, R., A. McGovern, and D. Gagne, 2019: Deep learning for spatially explicit prediction of synoptic-scale fronts. *Weather and Forecasting*, **34 (4)**, 1137–1160, URL https://doi.org/10.1175/WAF-D-18-0183.1.

Lagerquist, R., A. McGovern, C. Homeyer, D. Gagne, and T. Smith, 2020b: Deep learning on three-dimensional multiscale data for next-hour tornado prediction. *Monthly Weather Review*, **conditionally accepted**.

Lakshmanan, V., I. Adrianto, T. Smith, and G. Stumpf, 2005: A spatiotemporal approach to tornado prediction. *IEEE International Joint Conference on Neural Networks*, **3**, 1642–1647, URL https://doi.org/10.1109/IJCNN.2005.1556125.

Lakshmanan, V., B. Herzog, and D. Kingfield, 2015a: A method for extracting postevent storm tracks. *Journal of Applied Meteorology and Climatology*, **54 (2)**, 451–462, URL https://doi.org/10.1175/JAMC-D-14-0132.1.

Lakshmanan, V., C. Karstens, J. Krause, K. Elmore, A. Ryzhkov, and S. Berkseth, 2015b: Which polarimetric variables are important for weather/no-weather discrimination? *Journal of Atmospheric and Oceanic Technology*, **32 (6)**, 1209–1223, URL https://doi.org/10.1175/JTECH-D-13-00205.1.

Lakshmanan, V., and T. Smith, 2010: An objective method of evaluating and devising storm-tracking algorithms. *Weather and Forecasting*, **25 (2)**, 701–709, URL https://doi.org/10.1175/2009WAF2222330.1.

Lakshmanan, V., T. Smith, G. Stumpf, and K. Hondl, 2007: The Warning Decision Support System – Integrated Information. *Weather and Forecasting*, **22 (3)**, 596–612, URL https://doi.org/10.1175/WAF1009.1.

Lin, J., and T. Qian, 2019: Switch between El Niño and La Niña is caused by subsurface ocean waves likely driven by lunar tidal forcing. *Scientific Reports*, **9 (1)**, 1–10, URL https://doi.org/10.1038/s41598-019-49678-w.

Liu, Y., and Coauthors, 2016: Application of deep convolutional neural networks for detecting extreme weather in climate datasets. *arXiv pre-prints*, **1605 (01156)**, URL https://arxiv.org/abs/1605.01156.

Low, T., and D. Hudak, 1997: Development of air mass climatology analysis for the determination of characteristic marine atmospheres. Part I: North Atlantic. *Theoretical and Applied Climatology*, **57 (3-4)**, 135–153, URL https://doi.org/10.1007/BF00863609.

Lucas, C., B. Timbal, and H. Nguyen, 2014: The expanding tropics: A critical assessment of the observational and modeling studies. *Wiley Interdisciplinary Reviews: Climate Change*, **5 (1)**, 89–112, URL https://doi.org/10.1002/wcc.251.

Luna-Herrera, J., G. Martinez-Cabrera, R. Parra-Maldonado, J. Enciso-Moreno, J. Torres-Lopez, F. Quesada-Pascual, R. Delgadillo-Polanco, and S. Franzblau, 2003: Use of receiver operating characteristic curves to assess the performance of a microdilution assay for determination of drug susceptibility of clinical isolates of Mycobacterium tuberculosis. *European Journal of Clinical Microbiology and Infectious Diseases*, **22 (1)**, 21–27, URL https://doi.org/10.1007/s10096-002-0855-5.

Lyon, B., and A. Barnston, 2005: ENSO and the spatial extent of interannual precipitation extremes in tropical land areas. *Journal of Climate*, **18 (23)**, 5095–5109, URL https://doi.org/10.1175/JCLI3598.1.

Maas, A., A. Hannun, and A. Ng, 2013: Rectifier nonlinearities improve neural network acoustic models. *International Conference on Machine Learning*, Atlanta, Georgia, International Machine Learning Society, URL http://robotics.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf.

Mahale, V., J. Brotzge, and H. Bluestein, 2012: An analysis of vortices embedded within a quasi-linear convective system using X-band polarimetric radar. *Weather and Forecasting*, **27 (6)**, 1520–1537, URL https://doi.org/10.1175/WAF-D-11-00135.1.

Mahalik, M., B. Smith, K. Elmore, D. Kingfield, K. Ortega, and T. Smith, 2019: Estimates of gradients in radar moments using a linear least squares derivative technique. *Weather and Forecasting*, **34 (2)**, 415–434, URL https://doi.org/10.1175/WAF-D-18-0095.1.

Mann, H., 1945: Nonparametric tests against trend. *Econometrica: Journal of the Econometric Society*, **13 (3)**, 245–259, URL http://dx.doi.org/10.2307/1907187.

Markowski, P., 2002: Hook echoes and rear-flank downdrafts: A review. *Monthly Weather Review*, **130 (4)**, 852–876, URL https://doi.org/10.1175/1520-0493(2002)130%3C0852:HEARFD%3E2.0.CO;2.

Markowski, P., and Y. Richardson, 2009: Tornadogenesis: Our current understanding, forecasting considerations, and questions to guide future research. *Atmospheric Research*, **93 (1-3)**, 3–10, URL https://doi.org/10.1016/j.atmosres.2008.09.015.

Markowski, P., and Y. Richardson, 2014: What we know and don't know about tornado formation. *Physics Today*, **67 (9)**, 26–31, URL http://dx.doi.org/10.1063/PT.3.2514.

Markowski, P., J. Straka, and E. Rasmussen, 2002: Direct surface thermodynamic observations within the rear-flank downdrafts of nontornadic and tornadic supercells. *Monthly Weather Review*, **130 (7)**, 1692–1721, URL https://doi.org/10.1175/1520-0493(2002)130%3C1692:DSTOWT%3E2.0.CO;2.

Marzban, C., and G. Stumpf, 1996: A neural network for tornado prediction based on Doppler radar-derived attributes. *Journal of Applied Meteorology*, **35 (5)**, 617–626, URL https://doi.org/10.1175/1520-0450(1996)035%3C0617:ANNFTP%3E2.0.CO;2.

Mattson, M., 2014: Superior pattern processing is the essence of the evolved human brain. *Frontiers in Neuroscience*, **8 (265)**, 1–17, URL https://doi.org/10.3389/fnins.2014.00265.

McCandless, T., D. Gagne, B. Kosovic, S. Haupt, and T. Brummet, 2018: Machine learning parameterizations of surface layer fluxes. *AGU Fall Meeting*, Washington, DC, American Geophysical Union, URL https://ui.adsabs.harvard.edu/abs/2018AGUFMEP51E1867M/abstract.

McGovern, A., R. Lagerquist, and D. Gagne, 2020: Using machine learning and model interpretation and visualization techniques to gain physical insights in atmospheric science. *Proceedings of the International Conference on Learning Representations*, **accepted**.

McGovern, A., R. Lagerquist, D. Gagne, G. Jergensen, K. Elmore, C. Homeyer, and T. Smith, 2019: Making the black box more transparent: Understanding the physical implications of machine learning. *Bulletin of the American Meteorological Society*, **100 (11)**, 2175–2199, URL https://doi.org/10.1175/BAMS-D-18-0195.1.

Mehdi, T., N. Bashardoost, and M. Ahmadi, 2011: Kernel smoothing for ROC curve and estimation for thyroid stimulating hormone. *International Journal of Public Health Research*, **Special Issue**, 239–242, URL https://core.ac.uk/download/pdf/11492539.pdf.

Mercer, A., C. Shafer, C. Doswell, L. Leslie, and M. Richman, 2009: Objective classification of tornadic and nontornadic severe weather outbreaks. *Monthly Weather Review*, **137 (12)**, 4355–4368, URL https://doi.org/10.1175/2009MWR2897.1.

Mercer, A., C. Shafer, C. Doswell, L. Leslie, and M. Richman, 2012: Synoptic composites of tornadic and nontornadic outbreaks. *Monthly Weather Review*, **140 (8)**, 2590–2608, URL https://doi.org/10.1175/MWR-D-12-00029.1.

Metz, C., 1978: Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, **8 (4)**, 283–298, URL http://www.umich.edu/~ners580/ners-bioe_481/lectures/pdfs/1978-10-semNucMed_Metz-basicROC.pdf.

Miller, R., 1959: Tornado-producing synoptic patterns. *Bulletin of the American Meteorological Society*, **40 (9)**, 465–472, URL https://doi.org/10.1175/1520-0477-40.9.465.

Montavon, G., W. Samek, and K. Müller, 2018: Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, **73**, 1–15, URL https://doi.org/10.1016/j.dsp.2017.10.011.

Morgan, G., D. Brunkow, and R. Beebe, 1975: Climatology of Surface Fronts. Tech. rep., Illinois State Water Survey. URL https://www.ideals.illinois.edu/bitstream/handle/2142/94429/ISWSC-122.pdf?sequence=1.

Morris, W., and P. Smith, 2001: Cyclolysis: A diagnosis of two extratropical cyclones. *Monthly Weather Review*, **129 (11)**, 2714–2729, URL https://doi.org/10.1175/1520-0493(2001)129%3C2714:CADOTE%3E2.0.CO;2.

Muller, M., G. Tomlinson, T. Marrie, P. Tang, A. McGeer, D. Low, A. Detsky, and W. Gold, 2005: Can routine laboratory tests discriminate between severe acute respiratory syndrome and other causes of community-acquired pneumonia? *Clinical Infectious Diseases*, **40 (8)**, 1079–1086, URL https://doi.org/10.1086/428577.

Nair, V., and G. Hinton, 2010: Rectified linear units improve restricted Boltzmann machines. *International Conference on Machine Learning*, Haifa, Israel, International Machine Learning Society, URL https://www.cs.toronto.edu/~hinton/absps/reluICML.pdf.

Niebauer, H., 1998: Variability in Bering Sea ice cover as affected by a regime shift in the North Pacific in the period 1947–1996. *Journal of Geophysical Research: Oceans*, **103 (C12)**, 27 717–27 737, URL https://doi.org/10.1029/98JC02499.

NWS, 2007: Reading the Coded Surface Bulletin. Accessed: 2019-12-09, http://www.wpc.ncep.noaa.gov/html/read_coded_bull.shtml.

NWS, 2016: *Storm Data* preparation. National Weather Service Instruction 10-1605. Tech. rep., http://www.nws.noaa.gov/directives.

Olah, C., A. Mordvintsev, and L. Schubert, 2017: Feature visualization. *Distill*, doi:10.23915/distill.00007, URL https://distill.pub/2017/feature-visualization.

Orlanski, I., 1975: A rational subdivision of scales for atmospheric processes. *Bulletin of the American Meteorological Society*, **56 (5)**, 527–530, URL https://doi.org/10.1175/1520-0477-56.5.527.

Ortega, K., T. Smith, J. Zhang, C. Langston, Y. Qi, S. Stevens, and J. Tate, 2012: The Multi-year Reanalysis of Remotely Sensed Storms (MYRORSS) project. *Conference on Severe Local Storms*, Nashville, Tennessee, American Meteorological Society, URL https://ams.confex.com/ams/26SLS/webprogram/Handout/Paper211413/p4_74_ortegaetal_myrorss.pdf.

Parkinson, C., 2014: Spatially mapped reductions in the length of the Arctic sea ice season. *Geophysical Research Letters*, **41 (12)**, 4316–4322, URL https://doi.org/10.1002/2014GL060434.

Racah, E., C. Beckham, T. Maharaj, S. Kahou, Prabhat, and C. Pal, 2017: ExtremeWeather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events. *Advances in Neural Information Processing Systems*, Long Beach, California, Neural Information Processing Systems, URL http://papers.nips.cc/paper/6932-extremeweather-a-large-scale-climate-dataset-for-semi-supervised-de.

Rackley, J., and J. Knox, 2016: A climatology of southern Appalachian cold-air damming. *Weather and Forecasting*, **31 (2)**, 419–432, URL https://doi.org/10.1175/WAF-D-15-0049.1.

Reason, C., R. Allan, J. Lindesay, and T. Ansell, 2000: ENSO and climatic signals across the Indian Ocean basin in the global context: Part I, Interannual composite patterns. *International Journal of Climatology*, **20 (11)**, 1285–1327, URL https://doi.org/10.1002/1097-0088(200009)20:11%3C1285::AID-JOC536%3E3.0.CO;2-R.

Reichstein, M., G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, and Prabhat, 2019: Deep learning and process understanding for data-driven Earth system science. *Nature*, **566**, 195–204, URL https://doi.org/10.1038/s41586-019-0912-1.

Rhyzkov, A., T. Schuur, D. Burgess, and D. Zrnić, 2005: Polarimetric tornado detection. *Journal of Applied Meteorology*, **44 (5)**, 557–570, URL https://doi.org/10.1175/JAM2235.1.

Roberts, B., M. Xue, and D. Dawson, 2020: The effect of surface drag strength on mesocyclone intensification and tornadogenesis in idealized supercell simulations. *Journal of the Atmospheric Sciences*, **early online release**, URL https://doi.org/10.1175/JAS-D-19-0109.1.

Roebber, P., 2009: Visualizing multiple measures of forecast quality. *Weather and Forecasting*, **24 (2)**, 601–608, URL https://doi.org/10.1175/2008WAF2222159.1.

Rotunno, R., and J. Klemp, 1985: On the rotation and propagation of simulated supercell thunderstorms. *Journal of the Atmospheric Sciences*, **42 (3)**, 271–292, URL https://doi.org/10.1175/1520-0469(1985)042%3C0271:OTRAPO%3E2.0.CO;2.

Rudeva, I., and I. Simmonds, 2015: Variability and trends of global atmospheric frontal activity and links with large-scale modes of variability. *Journal of Climate*, **28 (8)**, 3311–3330, URL https://doi.org/10.1175/JCLI-D-14-00458.1.

Samek, W., T. Wiegand, and K. Müller, 2017: Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv pre-prints*, **1708 (08296)**, URL https://arxiv.org/abs/1708.08296.

Schemm, S., G. Rivière, L. Ciasto, and C. Li, 2018: Extratropical cyclogenesis changes in connection with tropospheric ENSO teleconnections to the north Atlantic: Role of stationary and transient waves. *Journal of the Atmospheric Sciences*, **75 (11)**, 3943–3964, URL https://doi.org/10.1175/JAS-D-17-0340.1.

Schemm, S., I. Rudeva, and I. Simmonds, 2015: Extratropical fronts in the lower troposphere – global perspectives obtained from two automated methods. *Quarterly Journal of the Royal Meteorological Society*, **141 (690)**, 1686–1698, URL https://doi.org/10.1002/qj.2471.

Schenkman, A., and M. Xue, 2016: Bow-echo mesovortices: A review. *Atmospheric Research*, **170**, 1–13, URL https://doi.org/10.1016/j.atmosres.2015.11.003.

Schmidt, D., and K. Grise, 2017: The response of local precipitation and sea level pressure to Hadley cell expansion. *Geophysical Research Letters*, **44 (20)**, 10 573–10 582, URL https://doi.org/10.1002/2017GL075380.

Schneider, R., and A. Dean, 2008: A comprehensive 5-year severe storm environment climatology for the continental United States. *Conference on Severe Local Storms*, Nashville, Tennessee, American Meteorological Society, URL https://www.researchgate.net/profile/Russell_Schneider/publication/253461395_A_COMPREHENSIVE_5YEAR_SEVERE_STORM_ENVIRONMENT_CLIMATOLOGY_FOR_THE_CONTINENTAL_UNITED_STATES/links/02e7e534419dadd476000000.pdf.

Schultz, D., D. Keyser, and L. Bosart, 1998: The effect of large-scale flow on low-level frontal structure and evolution in midlatitude cyclones. *Monthly Weather Review*, **126 (7)**, 1767–1791, URL https://doi.org/10.1175/1520-0493(1998)126%3C1767:TEOLSF%3E2.0.CO;2.

Schultz, D., and G. Vaughan, 2011: Occluded fronts and the occlusion process: A fresh look at conventional wisdom. *Bulletin of the American Meteorological Society*, **92 (4)**, 443–466, URL https://doi.org/10.1175/2010BAMS3057.1.

Seager, R., N. Narnik, Y. Kushiner, W. Robinson, and J. Miller, 2003: Mechanisms of hemispherically symmetric climate variability. *Journal of Climate*, **16 (18)**, 2960–2978, URL https://doi.org/10.1175/1520-0442(2003)016%3C2960:MOHSCV%3E2.0.CO;2.

Selvaraju, R., M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, 2017: Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Conference on Computer Vision*, Venice, Italy, IEEE, URL http://openaccess.thecvf.com/content_iccv_2017/html/Selvaraju_Grad-CAM_Visual_Explanations_ICCV_2017_paper.html.

Serreze, M., and R. Barry, 2011: Processes and impacts of Arctic amplification: A research synthesis. *Global and Planetary Change*, **77 (1-2)**, 85–96, URL https://doi.org/10.1016/j.gloplacha.2011.03.004.

Shafer, C., C. Doswell, L. Leslie, and M. Richman, 2010a: On the use of areal coverage of parameters favorable for severe weather to discriminate major outbreaks. *Electronic Journal of Severe Storms Meteorology*, **5 (7)**, 1–27, URL https://ejssm.org/ojs/index.php/ejssm/article/viewArticle/72.

Shafer, C., A. Mercer, C. Doswell, M. Richman, and L. Leslie, 2009: Evaluation of WRF forecasts of tornadic and nontornadic outbreaks when initialized with synoptic-scale input. *Monthly Weather Review*, **137 (4)**, 1250–1271, URL https://doi.org/10.1175/2008MWR2597.1.

Shafer, C., A. Mercer, L. Leslie, M. Richman, and C. Doswell, 2010b: Evaluation of WRF model simulations of tornadic and nontornadic outbreaks occurring in the spring and fall. *Monthly Weather Review*, **138 (11)**, 4098–4119, URL https://doi.org/10.1175/2010MWR3269.1.

Shafer, C., A. Mercer, M. Richman, L. Leslie, and C. Doswell, 2012: An assessment of areal coverage of severe weather parameters for severe weather outbreak diagnosis. *Weather and Forecasting*, **27 (4)**, 809–831, URL https://doi.org/10.1175/WAF-D-11-00142.1.

Shapiro, L., and G. Stockman, 2000: *Computer Vision*.

Shapiro, M., and D. Keyser, 1990: Fronts, Jet Streams, and the Tropopause. *Extratropical Cyclones*, C. Newton, and E. Holopainen, Eds., American Meteorological Society, URL https://doi.org/10.1007/978-1-944970-33-8_10.

Silver, D., and Coauthors, 2016: Mastering the game of Go with deep neural networks and tree search. *Nature*, **529 (7587)**, 484–489, URL https://doi.org/10.1038/nature16961.

Simmonds, I., K. Keay, and J. Bye, 2012: Identification and climatology of Southern Hemisphere mobile fronts in a modern reanalysis. *Journal of Climate*, **25 (6)**, 1945–1962, URL https://doi.org/10.1175/JCLI-D-11-00100.1.

Simonyan, K., A. Vedaldi, and A. Zisserman, 2014: Deep inside convolutional networks: Visualizing image classification models and saliency maps. *arXiv pre-prints*, **1312**, URL https://arxiv.org/abs/1312.6034.

Skinner, P., and Coauthors, 2018: Object-based verification of a prototype Warn-on-Forecast system. *Weather and Forecasting*, **33 (5)**, 1225–1250, URL https://doi.org/10.1175/WAF-D-18-0020.1.

Smith, B., R. Thompson, J. Grams, C. Broyles, and H. Brooks, 2012: Convective modes for significant severe thunderstorms in the contiguous United States. Part I: Storm classification and climatology. *Weather and Forecasting*, **27 (5)**, 1114–1135, URL https://doi.org/10.1175/WAF-D-11-00115.1.

Smith, T., and Coauthors, 2016: Multi-radar Multi-sensor (MRMS) severe weather and aviation products: Initial operating capabilities. *Bulletin of the American Meteorological Society*, **97 (9)**, 1617–1630, URL https://doi.org/10.1175/BAMS-D-14-00173.1.

Snook, N., M. Xue, and Y. Jung, 2019: Tornado-resolving ensemble and probabilistic predictions of the 20 May 2013 Newcastle-Moore EF5 tornado. *Monthly Weather Review*, **147 (4)**, 1215–1235, URL https://doi.org/10.1175/MWR-D-18-0236.1.

Sobash, R., J. Kain, D. Bright, A. Dean, M. Coniglio, and S. Weiss, 2011: Probabilistic forecast guidance for severe thunderstorms based on the identification of extreme phenomena in convection-allowing model forecasts. *Weather and Forecasting*, **26 (5)**, 714–728, URL https://doi.org/10.1175/WAF-D-10-05046.1.

Sobash, R., G. Romine, C. Schwartz, D. Gagne, and M. Weisman, 2016a: Explicit forecasts of low-level rotation from convection-allowing models for next-day tornado prediction. *Weather and Forecasting*, **31 (5)**, 1591–1614, URL https://doi.org/10.1175/WAF-D-16-0073.1.

Sobash, R., C. Schwartz, G. Romine, K. Fossell, and M. Weisman, 2016b: Severe weather prediction using storm surrogates from an ensemble forecasting system. *Weather and Forecasting*, **31 (1)**, 255–271, URL https://doi.org/10.1175/WAF-D-15-0138.1.

SPC, 2019: Storm Prediction Center – Mesoscale Analysis Pages. Accessed: 2019-12-08, http://www.spc.noaa.gov/exper/mesoanalysis/.

Starzec, M., C. Hometer, and G. Mullendore, 2017: Storm Labeling in Three Dimensions (SL3D): A volumetric radar echo and dual-polarization updraft classification algorithm. *Monthly Weather Review*, **145 (3)**, 1127–1145, URL https://doi.org/10.1175/MWR-D-16-0089.1.

Stensrud, D., and Coauthors, 2009: Convective-scale Warn-on-Forecast system: A vision for 2020. *Bulletin of the American Meteorological Society*, **90 (10)**, 1487–1500, URL https://doi.org/10.1175/2009BAMS2795.1.

Stensrud, D., and Coauthors, 2013: Progress and challenges with Warn-on-Forecast. *Atmospheric Research*, **123**, 2–16, URL https://doi.org/10.1016/j.atmosres.2012.04.004.

Stroeve, J., T. Markus, L. Boisvert, J. Miller, and A. Barrett, 2014: Changes in Arctic melt season and implications for sea ice loss. *Geophysical Research Letters*, **41 (4)**, 1216–1225, URL https://doi.org/10.1002/2013GL058951.

Sturmfels, P., S. Lundberg, and S. Lee, 2020: Visualizing the impact of feature attribution baselines. *Distill*, URL https://doi.org/10.23915/distill.00022.

Tamarin, T., and Y. Kaspi, 2017: The poleward shift of storm tracks under global warming: A Lagrangian perspective. *Geophysical Research Letters*, **44 (20)**, 10 666–10 674, URL https://doi.org/10.1002/2017GL073633.

Tarek, M., F. Brissette, and R. Arsenault, 2019: Evaluation of the ERA5 reanalysis as a potential reference dataset for hydrological modeling over North-America. *Hydrological and Earth System Sciences: Discussions*, 1–35, URL https://doi.org/10.5194/hess-2019-316.

Taylor, K., R. Stouffer, and G. Meehl, 2012: An overview of CMIP5 and the experiment design. *Bulletin of the American Meteorological Society*, **93 (4)**, 485–498, URL https://doi.org/10.1175/BAMS-D-11-00094.1.

Thomas, B., and J. Martin, 2007: A synoptic climatology and composite analysis of the Alberta Clipper. *Weather and Forecasting*, **22 (2)**, 315–333, URL https://doi.org/10.1175/WAF982.1.

Toms, B., E. Barnes, and I. Ebert-Uphoff, 2019: Physically interpretable neural networks for the geosciences: Applications to Earth system variability. *arXiv pre-prints*, **1912 (01752)**, URL https://arxiv.org/abs/1912.01752.

Trafalis, T., I. Adrianto, M. Richman, and S. Lakshmivarahan, 2013: Machine-learning classifiers for imbalanced tornado data. *Computational Management Science*, **11 (4)**, 403–418, URL https://doi.org/10.1007/s10287-013-0174-6.

Trapp, R., 1999: Observations of nontornadic low-level mesocyclones and attendant tornadogenesis failure during VORTEX. *Monthly Weather Review*, **127 (7)**, 1693–1705, URL https://doi.org/10.1175/1520-0493(1999)127%3C1693:OONLLM%3E2.0.CO;2.

Trapp, R., G. Stumpf, and K. Manross, 2005a: A reassessment of the percentage of tornadic mesocyclones. *Weather and Forecasting*, **20 (4)**, 680–687, URL https://doi.org/10.1175/WAF864.1.

Trapp, R., S. Tessendorf, E. Godfrey, and H. Brooks, 2005b: Tornadoes from squall lines and bow echoes. Part 1: Climatological distribution. *Weather and Forecasting*, **20 (1)**, 23–34, URL https://doi.org/10.1175/WAF-835.1.

Trapp, R., and M. Weisman, 2003: Low-level mesovortices within squall lines and bow echoes. Part II: Their genesis and implications. *Monthly Weather Review*, **131 (11)**, 2804–2823, URL https://doi.org/10.1175/1520-0493(2003)131%3C2804:LMWSLA%3E2.0.CO;2.

Trenberth, K., G. Branstator, D. Karoly, A. Kumar, N.-C. Lau, and C. Ropelewski, 1998: Progress during TOGA in understanding and modeling global teleconnections associated with tropical sea surface temperatures. *Journal of Geophysical Research: Oceans*, **103 (C7)**, 14 291–14 324, URL https://doi.org/10.1029/97JC01444.

Wallace, J., and P. Hobbs, 2006: *Atmospheric Science: An Introductory Survey*, Vol. 2. Elsevier.

Wang, L., K. Scott, L. Xu, and D. Clausi, 2016: Sea ice concentration estimation during melt from dual-pol SAR scenes using deep convolutional neural networks: A case study. *IEEE Transactions on Geoscience and Remote Sensing*, **54 (8)**, 4524–4533, URL https://doi.org/10.1109/TGRS.2016.2543660.

West, G., and R. Howard, 2019: Arctic Outflow. University of British Columbia, URL https://www.eoas.ubc.ca/courses/atsc113/snow/met_concepts/06-met_concepts/06l-arctic-outflow/.

Wheatley, D., K. Knopfmeier, T. Jones, and G. Creager, 2015: Storm-scale data assimilation and ensemble forecasting with the NSSL Experimental Warn-on-Forecast System. Part I: Radar data experiments. *Weather and Forecasting*, **30 (6)**, 1795–1817, URL https://doi.org/10.1175/WAF-D-15-0043.1.

Wilks, D., 2016: "The stippling shows statistically significant grid points": How research results are routinely overstated and overinterpreted, and what to do about it. *Bulletin of the American Meteorological Society*, **97 (12)**, 2263–2273, URL https://doi.org/10.1175/BAMS-D-15-00267.1.

Wilson, K., P. Heinselman, C. Kuster, D. Kingfield, and Z. Kang, 2017: Forecaster performance and workload: Does radar update time matter? *Weather and Forecasting*, **32 (1)**, 253–274, URL https://doi.org/10.1175/WAF-D-16-0157.1.

Wimmers, A., C. Velden, and J. Cossuth, 2019: Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery. *Monthly Weather Review*, **147 (6)**, 2261–2282, URL https://doi.org/10.1175/MWR-D-18-0391.1.

World Meteorological Organization, 2014: El Niño / Southern Oscillation. Tech. rep. URL https://library.wmo.int/doc_num.php?explnum_id=7888.

Wu, Y., M. Ting, R. Seager, H.-P. Huang, and M. Cane, 2011: Changes in storm tracks and energy transports in a warmer climate simulated by the GFDL CM2.1 model. *Climate Dynamics*, **37 (1-2)**, 53–72, URL https://doi.org/10.1007/s00382-010-0776-4.

Xu, X., M. Xue, and Y. Wang, 2015: The genesis of mesovortices within a real-data simulation of a bow echo system. *Journal of the Atmospheric Sciences*, **72 (5)**, 1963–1986, URL https://doi.org/10.1175/JAS-D-14-0209.1.

Yussouf, N., D. Dowell, L. Wicker, K. Knopfmeier, and D. Wheatley, 2015: Storm-scale data assimilation and ensemble forecasts for the 27 April 2011 severe weather outbreak in Alabama. *Monthly Weather Review*, **143 (8)**, 3044–3066, URL https://doi.org/10.1175/MWR-D-14-00268.1.

Zhou, B., A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, 2016: Learning deep features for discriminative localization. *Conference on Computer Vision and Pattern Recognition*, Las Vegas, Nevada, IEEE, URL https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Zhou_Learning_Deep_Features_CVPR_2016_paper.html.

Appendices

Appendix A

Estimation of Storm Velocity

This chapter describes the algorithm used to estimate storm velocity, which is important for storm-tracking (Section 3.1.4) and tornado attribution (Section 3.1.5). To estimate the velocity of storm S at time t_0, I use a backward difference. For a k^{th}-order backward difference, the procedure is as follows, letting t_{-k} be k time steps before t_0.

1. Find k^{th}-order predecessors of storm S at time t_0. These are typically storms at time t_{-k}, but if a track begins after t_{-k}, predecessors may occur after t_{-k}. Storm C in Figure A.1 is an example, occurring only two time steps before storm E.

2. If no predecessors were found, proceed to step 3. Otherwise, compute one velocity estimate for each predecessor, using the following equation:

$$\vec{v} = \left(\frac{\Delta x}{\Delta t}, \frac{\Delta y}{\Delta t} \right). \tag{A.1}$$

   Δt is the elapsed time (seconds); Δx is the eastward displacement of the storm center over this time (metres); Δy is northward displacement (metres); and $\vec{v}$ is the resulting estimate (m s^{-1}).

   Then average the estimates from Equation A.1 over all predecessors, to get one estimate for storm S at time t.

3. If step 2 yielded a velocity estimate, stop. Otherwise, use the following neighbourhood-based estimate:

$$\vec{v}_S = \frac{\sum_{S'} \exp(-\frac{d_{S'}}{\tau}) \vec{v}_{S'}}{\sum_{S'} \exp(-\frac{d_{S'}}{\tau})}, \tag{A.2}$$

   where $\vec{v}_{S'}$ is the velocity of storm S'; $d_{S'}$ is the distance between centers of storms S and S'; τ = 100 km is the e-folding distance of the neighbourhood filter; and $\sum_{S'}$ sums

over all storms at time t with a backward-difference-based velocity estimate. If no storm at time t has a backward-difference-based velocity estimate, set $\vec{v}_S$ to the zero vector (assume that the storm is stationary). This fallback case occurs very rarely.

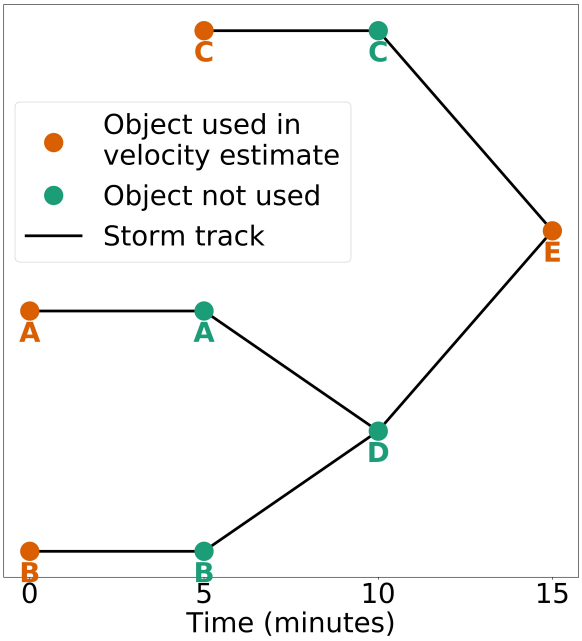


Figure A.1: Predecessors for estimation of storm velocity. For a third-order estimate of the velocity of storm E at t = 15 minutes, the predecessors used are storm A at 0 minutes, storm B at 0 minutes, and storm C at 5 minutes.

Appendix B

Additional Data

This chapter contains additional data that are omitted in the main text for the sake of brevity.

Table B.1: List of GridRad days and domains. Each entry contains the convective day, followed by latitudes in °N, then longitudes in °W.

8 Jan 2011, 25.5-32.0, 93.5-102.0	9 Jan 2011, 25.5-32.0, 93.5-102.0
25 Jan 2011, 25.0-31.0, 80.0-84.0	24 Feb 2011, 31.0-38.5, 82.0-95.0
25 Feb 2011, 34.0-41.0, 117.0-124.5	4 Mar 2011, 28.5-34.5, 86.5-96.0
5 Mar 2011, 28.5-34.5, 86.5-96.0	18 Mar 2011, 35.5-42.5, 119.5-124.5
19 Mar 2011, 32.0-35.5, 78.5-103.5	29 Mar 2011, 28.5-36.5, 84.5-94.5
30 Mar 2011, 25.0-30.5, 79.0-86.5	31 Mar 2011, 25.0-30.5, 79.0-86.5
9 Apr 2011, 40.5-45.5, 90.0-98.5	10 Apr 2011, 42.0-47.0, 86.5-93.5
14 Apr 2011, 30.0-41.5, 74.5-99.5	15 Apr 2011, 30.0-41.5, 74.5-99.5
16 Apr 2011, 30.0-41.5, 74.5-99.5	21 Apr 2011, 28.5-35.5, 96.5-104.5
24 Apr 2011, 30.5-37.0, 92.5-101.0	26 Apr 2011, 30.0-37.0, 81.0-97.0
27 Apr 2011, 30.0-37.0, 81.0-97.0	1 May 2011, 31.5-36.5, 89.5-96.5
22 May 2011, 35.5-46.5, 87.0-95.5	24 May 2011, 32.0-39.0, 92.5-101.5
26 May 2011, 29.5-43.0, 74.0-91.5	28 May 2011, 43.0-49.0, 87.5-96.5
29 May 2011, 40.0-45.5, 81.5-94.0	31 May 2011, 39.5-46.5, 67.0-80.5
1 Jun 2011, 39.5-46.5, 67.0-80.5	8 Jun 2011, 43.0-48.5, 67.0-71.5
10 Jun 2011, 37.0-42.0, 80.5-97.0	12 Jun 2011, 38.0-47.5, 74.5-108.0
22 Jun 2011, 31.5-39.0, 83.0-90.5	27 Jun 2011, 36.5-42.0, 79.5-85.5

Table B.2: List of GridRad days and domains (continuation of Table B.1).

29 Jun 2011, 43.0-49.0, 104.0-117.0	6 Jul 2011, 36.5-42.0, 101.0-108.5
8 Jul 2011, 45.0-49.5, 96.0-104.5	16 Jul 2011, 44.0-49.0, 93.5-104.0
17 Jul 2011, 44.0-49.0, 93.5-104.0	26 Jul 2011, 40.0-47.0, 69.5-104.0
2 Aug 2011, 25.0-30.5, 80.0-84.5	7 Aug 2011, 41.0-46.5, 82.0-87.0
21 Aug 2011, 41.0-45.0, 70.5-80.0	26 Aug 2011, 32.0-41.5, 72.5-80.5
27 Aug 2011, 32.0-41.5, 72.5-80.5	28 Aug 2011, 38.0-45.0, 71.0-77.0
17 Sep 2011, 31.5-37.5, 94.0-102.5	25 Sep 2011, 32.5-40.0, 84.5-94.0
27 Sep 2011, 38.5-44.0, 83.0-88.0	7 Oct 2011, 36.5-41.0, 97.0-103.0
8 Oct 2011, 36.5-41.0, 97.0-103.0	18 Oct 2011, 24.5-30.5, 78.5-84.0
7 Nov 2011, 32.0-38.0, 92.5-103.0	8 Nov 2011, 28.5-36.5, 92.0-98.0
15 Nov 2011, 28.5-37.5, 75.5-92.0	16 Nov 2011, 28.5-37.5, 75.5-92.0
21 Nov 2011, 31.5-36.0, 92.0-98.0	21 Dec 2011, 30.0-35.5, 82.5-89.0
22 Dec 2011, 29.0-35.0, 83.0-93.5	9 Apr 2012, 33.5-37.5, 95.0-101.0
13 Apr 2012, 34.5-37.0, 95.0-100.5	14 Apr 2012, 36.0-41.5, 95.5-101.0
9 Jan 2013, 29.0-33.0, 89.5-95.0	10 Jan 2013, 29.0-33.0, 89.5-95.0
12 Jan 2013, 34.5-39.0, 85.0-91.5	18 Feb 2013, 28.0-35.0, 91.0-97.5
19 Feb 2013, 36.0-42.0, 118.0-124.5	5 Mar 2013, 32.0-35.0, 81.0-88.0
17 Mar 2013, 31.0-37.0, 81.0-92.0	18 Mar 2013, 31.0-37.0, 81.0-92.0
24 Mar 2013, 35.0-40.0, 80.0-90.5	29 Mar 2013, 31.5-43.0, 92.0-104.0
30 Mar 2013, 31.5-43.0, 92.0-104.0	7 Apr 2013, 37.0-40.5, 91.5-102.0
11 Apr 2013, 29.0-36.5, 81.5-92.0	23 Apr 2013, 29.0-33.5, 89.0-95.5
24 Apr 2013, 29.0-33.5, 89.0-95.5	26 Apr 2013, 34.0-37.5, 93.0-102.0
28 Apr 2013, 29.0-35.0, 79.5-87.5	7 May 2013, 42.0-45.0, 111.0-116.0
8 May 2013, 32.0-41.0, 93.5-106.0	16 May 2013, 31.0-35.0, 91.0-98.0
19 May 2013, 34.0-43.0, 89.0-102.0	20 May 2013, 31.5-40.0, 93.0-99.0
23 May 2013, 30.5-35.0, 98.0-103.0	26 May 2013, 40.0-44.0, 90.0-97.0

Table B.3: List of GridRad days and domains (continuation of Table B.1).

27 May 2013, 37.0-43.5, 93.5-104.5	28 May 2013, 32.0-43.0, 93.0-104.0
31 May 2013, 34.5-40.5, 87.0-99.0	4 Jun 2013, 29.0-35.0, 78.5-86.0
6 Jun 2013, 32.0-37.0, 75.0-83.0	12 Jun 2013, 38.0-45.0, 80.0-96.0
19 Jun 2013, 30.5-36.0, 100.0-105.5	21 Jun 2013, 40.0-46.0, 94.0-107.0
26 Jun 2013, 35.0-40.5, 82.0-91.0	12 Jul 2013, 44.5-47.5, 93.0-99.0
15 Jul 2013, 46.5-49.5, 101.5-107.0	25 Jul 2013, 36.0-40.0, 98.0-104.0
27 Jul 2013, 38.5-44.5, 74.0-81.0	3 Aug 2013, 34.5-42.0, 100.0-106.0
5 Aug 2013, 36.5-41.0, 96.0-103.0	6 Aug 2013, 42.5-48.5, 86.5-102.0
12 Aug 2013, 38.0-45.0, 100.5-109.0	26 Aug 2013, 44.0-48.0, 87.0-97.5
1 Sep 2013, 40.5-45.0, 87.0-90.0	11 Sep 2013, 46.0-50.0, 89.5-97.0
15 Sep 2013, 24.5-30.0, 79.5-83.5	16 Sep 2013, 24.5-30.0, 79.5-83.5
29 Sep 2013, 44.0-49.0, 119.5-125.5	30 Sep 2013, 44.0-49.0, 119.5-125.5
4 Oct 2013, 40.0-45.5, 91.0-100.5	7 Oct 2013, 37.0-47.0, 69.5-78.5
12 Oct 2013, 28.0-34.5, 93.5-100.0	17 Nov 2013, 34.5-46.0, 80.5-94.0
25 Nov 2013, 29.0-36.5, 75.0-88.5	26 Nov 2013, 29.0-36.5, 75.0-88.5
14 Dec 2013, 28.0-31.5, 80.0-88.5	21 Dec 2013, 29.5-39.0, 82.0-97.0
27 Apr 2014, 34.0-42.0, 91.5-99.0	10 May 2014, 36.0-43.0, 90.0-99.0
11 May 2014, 36.0-44.5, 92.0-102.0	21 May 2014, 37.5-41.0, 101.0-106.0
16 Jun 2014, 41.0-44.0, 89.0-100.0	17 Jun 2014, 41.5-48.0, 94.5-106.0
18 Jun 2014, 43.5-46.5, 98.0-100.0	12 Oct 2014, 29.5-40.5, 84.5-95.5
13 Oct 2014, 29.5-40.5, 84.5-95.5	6 May 2015, 32.5-41.5, 95.5-100.0
19 May 2015, 29.0-37.0, 94.0-103.0	24 May 2015, 36.0-41.0, 97.0-105.0
25 May 2015, 25.0-41.0, 89.0-105.0	27 May 2015, 29.5-41.5, 96.0-104.0
4 Jun 2015, 34.0-43.0, 93.0-108.0	23 Dec 2015, 33.5-42.0, 84.0-92.5
15 Apr 2016, 34.5-40.5, 99.0-104.0	9 May 2016, 33.0-41.5, 94.0-100.0
24 May 2016, 35.5-41.0, 97.0-104.0	25 May 2016, 35.5-40.0, 95.0-99.5

Table B.4: List of GridRad days and domains (continuation of Table B.1).

27 Mar 2017, 27.5-39.0, 92.0-105.5	28 Mar 2017, 27.5-39.0, 92.0-105.5
2 Apr 2017, 29.0-37.0, 78.5-90.5	3 Apr 2017, 29.0-37.0, 78.5-90.5
4 Apr 2017, 30.0-41.0, 78.0-90.0	5 Apr 2017, 30.0-41.0, 78.0-90.0
16 May 2017, 30.5-48.0, 87.5-105.0	18 May 2017, 31.0-40.5, 89.5-104.5
28 Jun 2017, 36.5-47.5, 87.0-104.0	29 Jun 2017, 35.5-45.0, 90.0-105.0
18 Jul 2017, 40.5-48.5, 87.5-104.0	19 Jul 2017, 40.5-48.5, 87.5-104.0
18 Mar 2018, 24.0-50.0, 66.0-125.0	19 Mar 2018, 24.0-50.0, 66.0-125.0
2 Apr 2018, 24.0-50.0, 66.0-125.0	3 Apr 2018, 24.0-50.0, 66.0-125.0

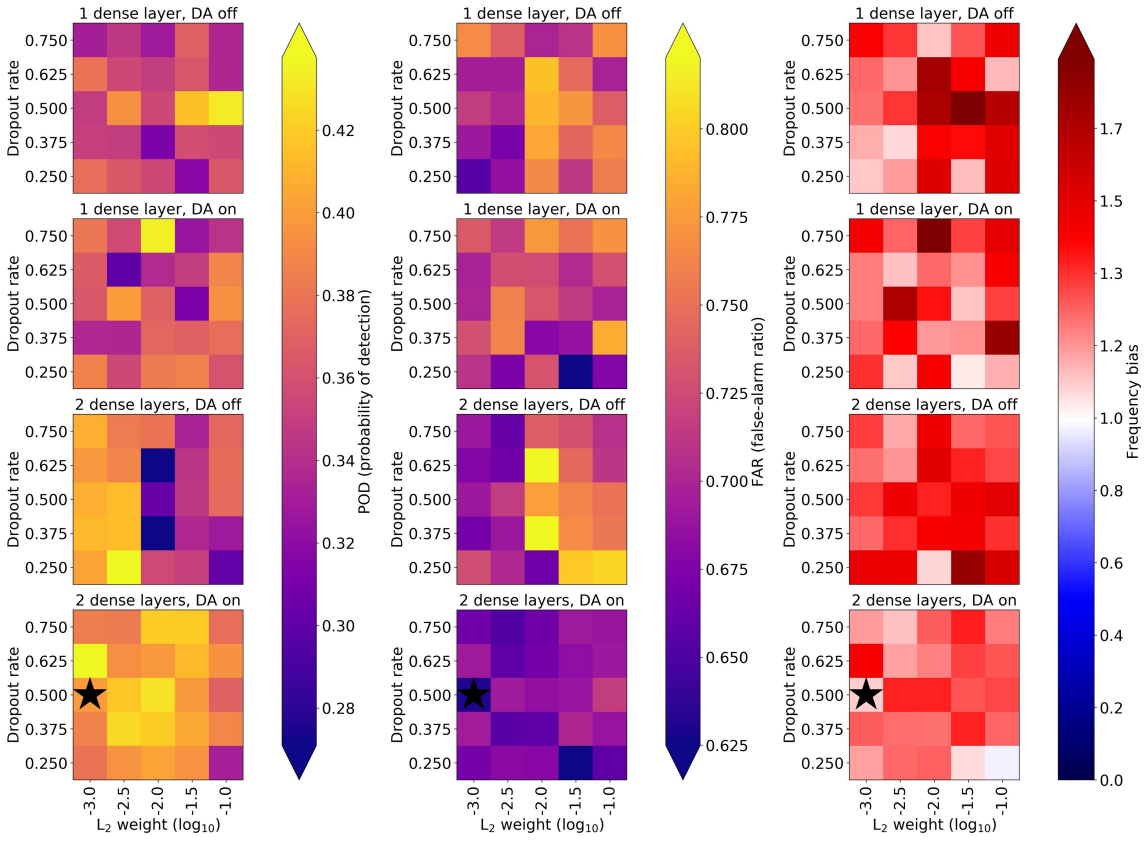


Figure B.1: POD, FAR, and frequency bias for GridRad models on validation data. Formatting is explained in Figure 6.1.

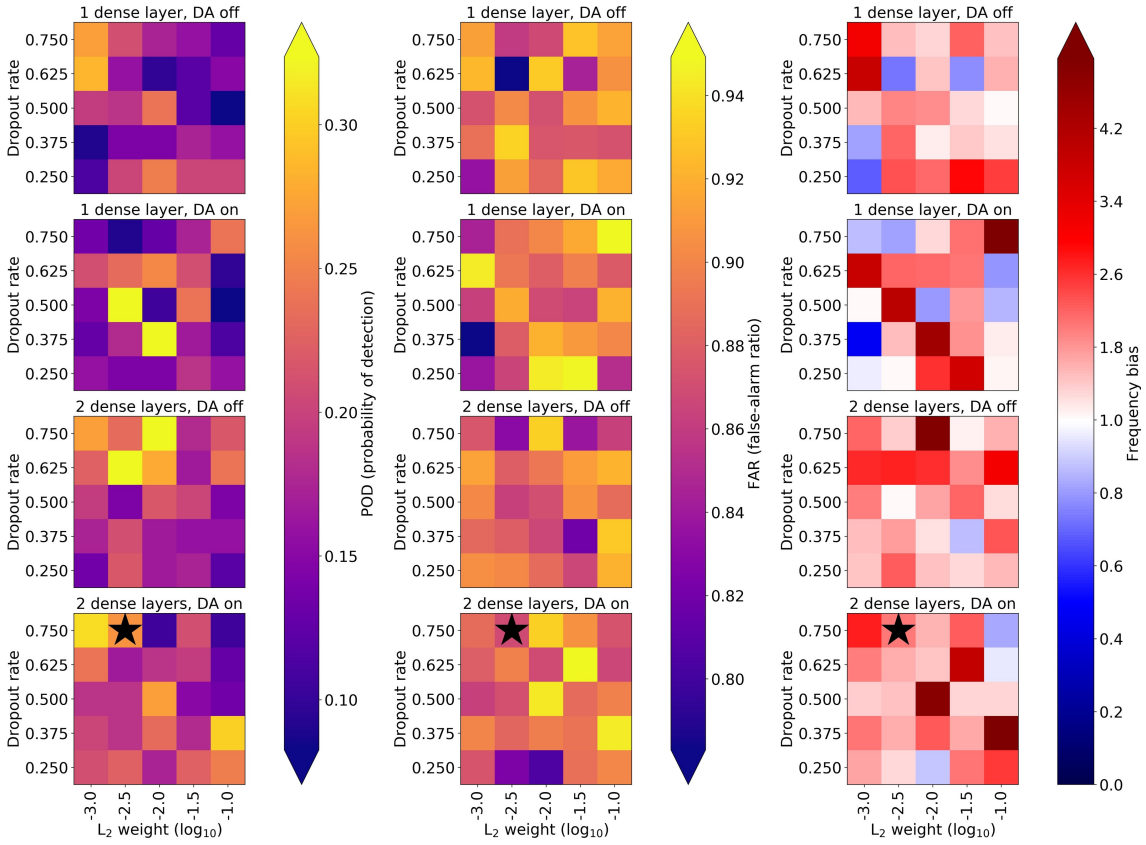


Figure B.2: POD, FAR, and frequency bias for MYRORSS models on validation data. Formatting is explained in Figure 6.1.

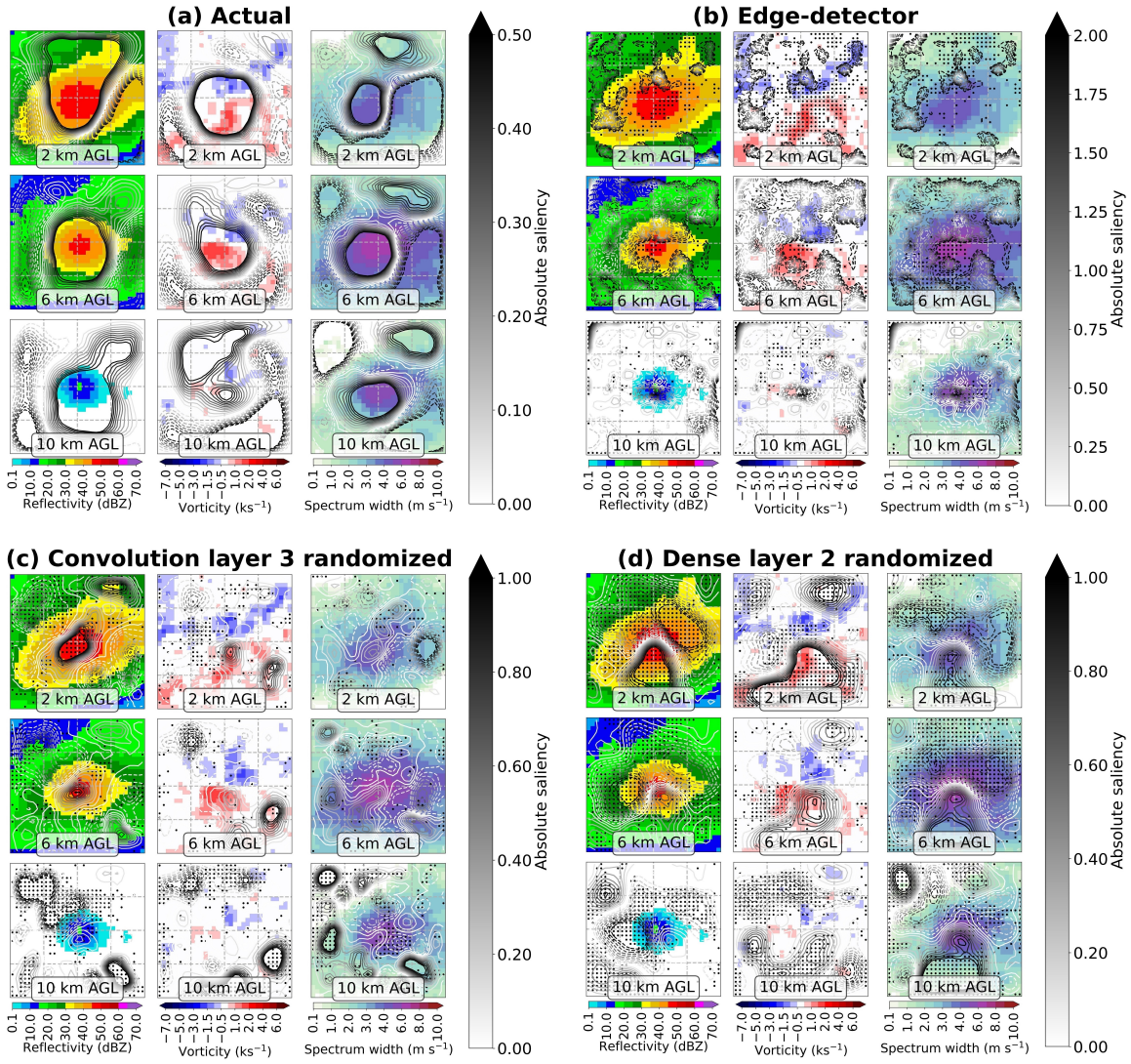


Figure B.3: Same as Figure 7.7 for but the worst misses.

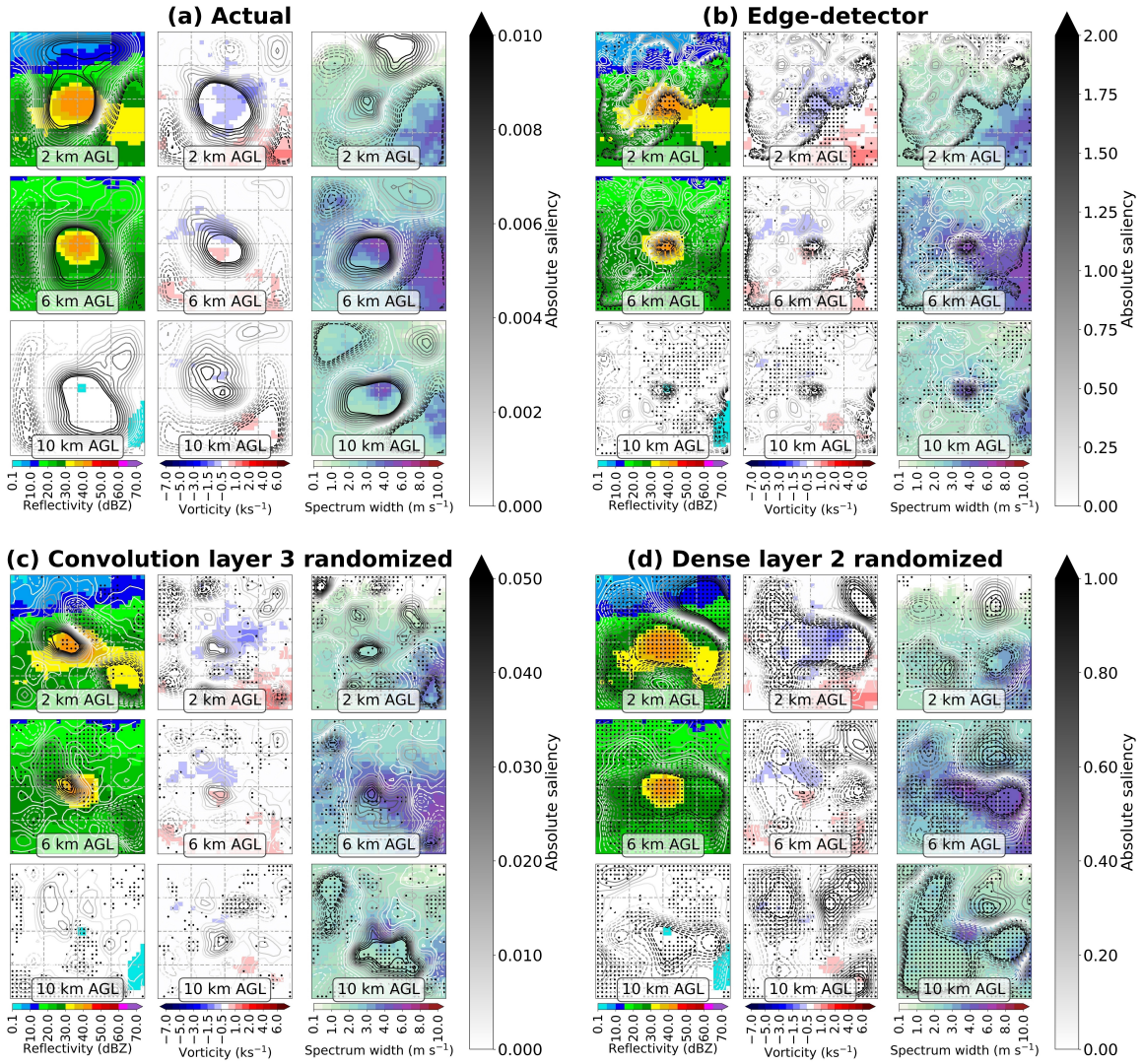


Figure B.4: Same as Figure 7.7 for but the best correct nulls.

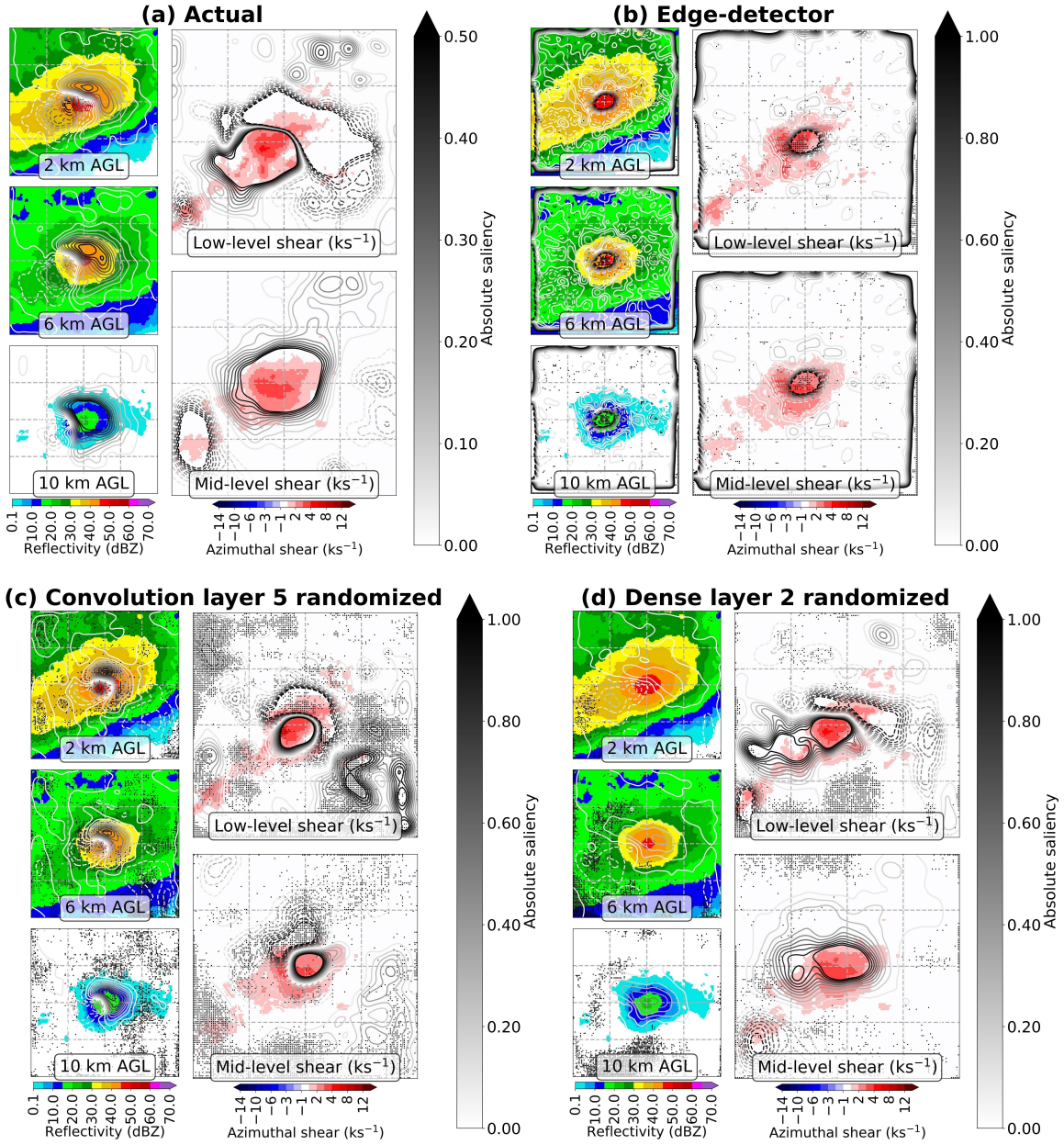


Figure B.5: Same as Figure 7.15 for but the worst misses.

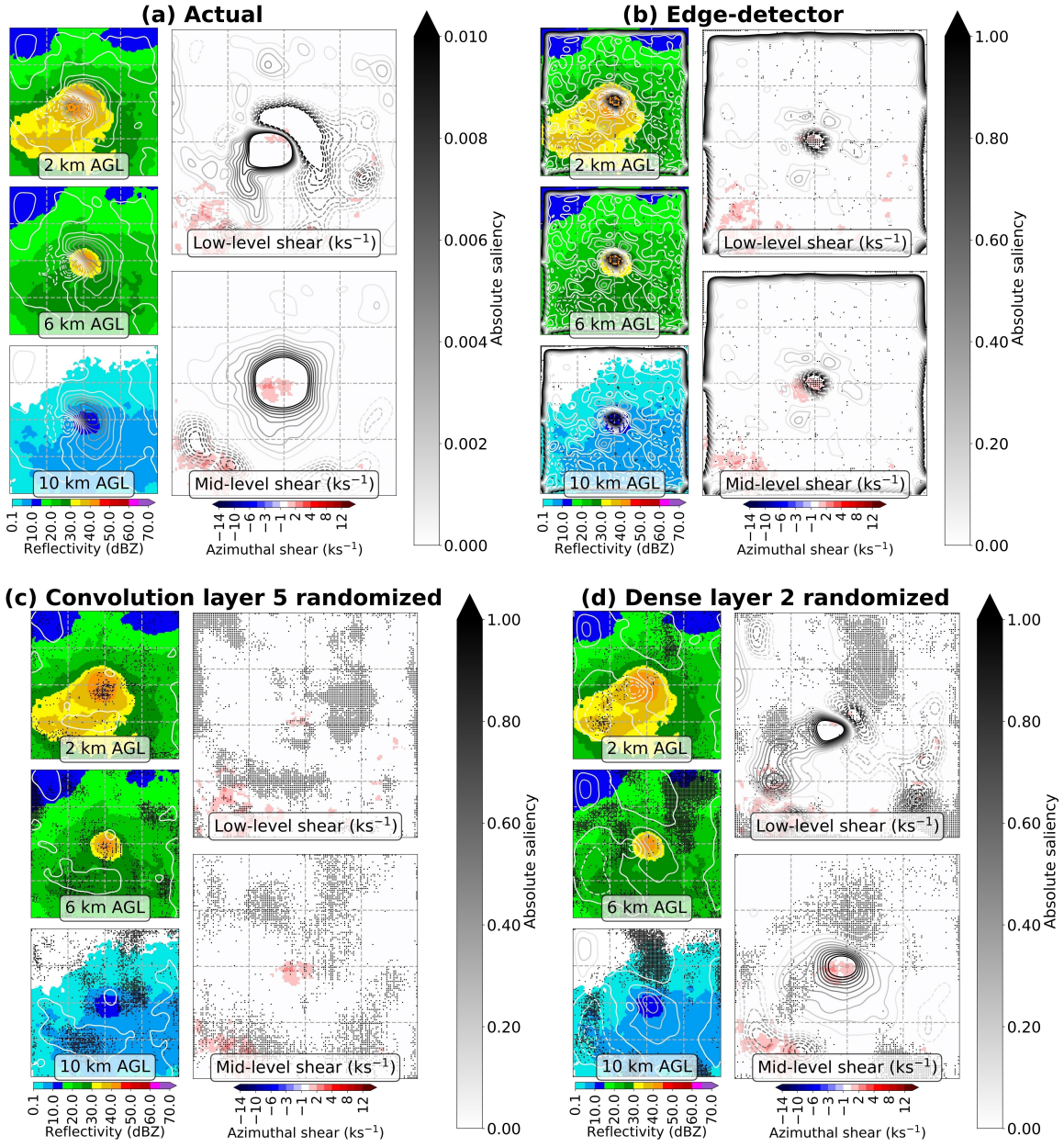


Figure B.6: Same as Figure 7.15 for but the best correct nulls.

Appendix C

Algorithms for Permutation Test

This chapter presents formal algorithms for the four versions of the permutation test: single-pass forward (Table C.1), multi-pass forward (Table C.2), single-pass backward (Table C.3), and multi-pass backward (Table C.4).

Input : Trained model (F), clean testing set ($\mathbf{X}$), number of predictors (N)

Output: Loss on clean testing set ($\mathscr{L}_0$)

Output: Loss after permuting each predictor ($\mathscr{L}_j$ for $j \in [1, N]$)

$\mathscr{L}_0 \leftarrow$ loss of F on clean testing set $\mathbf{X}$

foreach $j \in [1, N]$ **do**

  $\mathbf{X}'_{\mathbf{j}} \leftarrow$ copy $\mathbf{X}$ to new variable

  Permute spatial maps of j^{th} predictor over examples in $\mathbf{X}'_{\mathbf{j}}$

  $\mathscr{L}_j \leftarrow$ loss of F on dirty testing set $\mathbf{X}'_{\mathbf{j}}$

end

Table C.1: Algorithm for single-pass forward permutation test. Each predictor is permuted once in the testing data, and the resulting loss is recorded. Only one predictor at a time is permuted. The predictor whose permutation causes the k^{th}-highest loss, is considered the k^{th}-most important.

Input : Trained model (F), clean testing set ($\mathbf{X}$), number of predictors (N)

Output: Loss on clean testing set ($\mathscr{L}_0$)

Output: Loss after permanently permuting each predictor ($\mathscr{L}_j$ for $j \in [1, N]$)

$\mathscr{L}_0 \leftarrow$ loss of F on clean testing set $\mathbf{X}$

C (set of clean predictors) $\leftarrow \{1, 2, \ldots, N\}$

while C is not empty **do**

  $L_{\max}$ (maximum loss at this step) $\leftarrow -\infty$

  j^* (index of best predictor at this step) $\leftarrow$ None

  foreach $j \in C$ **do**

    $\mathbf{X}'_{\mathbf{j}} \leftarrow$ copy $\mathbf{X}$ to new variable

    Permute spatial maps of j^{th} predictor over examples in $\mathbf{X}'_{\mathbf{j}}$

    $L \leftarrow$ loss of F on dirty testing set $\mathbf{X}'_{\mathbf{j}}$

    if $L > L_{max}$ **then**

      $L_{\max} \leftarrow L$; $j^* \leftarrow j$

  end

  $\mathscr{L}_{j^*} \leftarrow L_{\max}$

  $C \leftarrow C - \{j^*\}$ (remove j^* from C)

  $\mathbf{X} \leftarrow \mathbf{X}'_{\mathbf{j}^*}$ (permute $j^{*\text{th}}$ predictor permanently)

end

Table C.2: Algorithm for multi-pass forward permutation test. One or more predictors at a time may be permuted. At the beginning of the algorithm, the testing set is clean (no predictors are permuted); at the end, the testing set is completely random (all predictors are permuted). The k^{th} predictor to be permanently permuted is considered the k^{th}-most important.

Input : Trained model (F), random testing set ($\mathbf{X}'$), number of predictors (N)

Output: Loss on random testing set ($\mathscr{L}_0$)

Output: Loss after cleaning each predictor ($\mathscr{L}_j$ for $j \in [1, N]$)

$\mathscr{L}_0 \leftarrow$ loss of F on random testing set $\mathbf{X}'$

foreach $j \in [1, N]$ **do**

  $\mathbf{X}''_{\mathbf{j}} \leftarrow$ copy $\mathbf{X}'$ to new variable

  Clean j^{th} predictor in $\mathbf{X}''_{\mathbf{j}}$ (assign spatial maps to correct examples)

  $\mathscr{L}_j \leftarrow$ loss of F on partly cleaned testing set $\mathbf{X}''_{\mathbf{j}}$

end

Table C.3: Algorithm for single-pass backward permutation test. At the beginning the testing set is completely randomized (all predictors are permuted). Each predictor is cleaned (depermuted) once, and the resulting loss is recorded. Only one predictor at a time is cleaned. The predictor whose cleaning causes the k^{th}-lowest loss, is considered the k^{th}-most important.

Input : Trained model (F), random testing set ($\mathbf{X}'$), number of predictors (N)

Output: Loss on random testing set ($\mathscr{L}_0$)

Output: Loss after permanently cleaning each predictor ($\mathscr{L}_j$ for $j \in [1, N]$)

$\mathscr{L}_0 \leftarrow$ loss of F on random testing set $\mathbf{X}$

D (set of dirty predictors) $\leftarrow \{1, 2, \ldots, N\}$

while D is not empty **do**

  L_{min} (minimum loss at this step) $\leftarrow \infty$

  j^* (index of best predictor at this step) $\leftarrow$ None

  foreach $j \in D$ **do**

    $\mathbf{X}''_{\mathbf{j}} \leftarrow$ copy $\mathbf{X}'$ to new variable

    Clean j^{th} predictor in $\mathbf{X}''_{\mathbf{j}}$ (assign spatial maps to correct examples)

    $L \leftarrow$ loss of F on partly cleaned testing set $\mathbf{X}''_{\mathbf{j}}$

    if $L < L_{min}$ **then**

      $L_{\text{min}} \leftarrow L$; $j^* \leftarrow j$

  end

  $\mathscr{L}_{j^*} \leftarrow L_{\text{min}}$

  $D \leftarrow D - \{j^*\}$ (remove j^* from D)

  $\mathbf{X}' \leftarrow \mathbf{X}''_{\mathbf{j}^*}$ (clean $j^{*\text{th}}$ predictor permanently)

end

Table C.4: Algorithm for multi-pass backward permutation test. One or more predictors at a time may be cleaned (depermuted). At the beginning of the algorithm, the testing set is completely randomized (all predictors are permuted); at the end, the testing set is clean (no predictors are permuted). The k^{th} predictor to be permanently cleaned is considered the k^{th}-most important.

Appendix D

Direct Comparison of MYRORSS and GridRad Interpretation Results

As discussed in Section 7.2, one could argue that differences between MYRORSS and GridRad interpretation results are caused by artifacts related to CNN architecture or grid shape, rather than fundamental differences between the two datasets. These objections are addressed here with a more direct comparison. In Section 7.2 the key difference between the two models, revealed by saliency maps, is that the GridRad model is more sensitive to discreteness of the storm while the MYRORSS model is more sensitive to storm structure near the rear-flank downdraft. The same difference is revealed by the direct comparison in this chapter. In general, results in this chapter are similar to those found in Section 7.2.

D.1 Methods

GridRad data are converted to MYRORSS format, then used to train a CNN with the same architecture as the original MYRORSS model (Figure 5.3). This CNN will be called the upscaled-GridRad model.[1] The procedure for converting GridRad data to MYRORSS format, applied separately to each example, is described below.

1. At each horizontal grid point, compute low-level azimuthal shear (S_{LL}) by taking the maximum of 2 $\times$ vorticity from 0-2 km AGL.[2]

2. At each horizontal grid point, compute mid-level azimuthal shear (S_{ML}) by taking the maximum of 2 $\times$ vorticity from 3-6 km AGL.

[1] Because converting GridRad data to MYRORSS format involves a decrease in grid spacing and thus an increase in nominal resolution.

[2] Physically, $S = \frac{1}{2}\zeta$, where S is azimuthal shear and ζ is vorticity. Thus, to convert from vorticity units to azimuthal-shear units, one must multiply by 2. This is similar to the conversion from metres to kilometres: even though 1 m = 0.001 km, to convert a length from kilometres to metres, one must multiply by 1000.

3. Upsample reflectivity, S_{LL}, and S_{ML} from 1.5-km to 0.375-km horizontal spacing, using bicubic interpolation.

4. Stack the 12 horizontal reflectivity maps with the two shear maps, yielding the 128 $\times$ 128 $\times$ 14 tensor shown in Figure 5.3.

To find the best hyperparameters for the upscaled-GridRad model, I ran the same experiment as for the MYRORSS model, described in Section 5.1. The best upscaled-GridRad model has a L_2 weight of $10^{-1.5}$, dropout rate of 0.625, two dense layers, and is trained with data augmentation. Figure D.1 shows the performance of this model on testing data.

D.2 Results

Figure D.2 shows results of the permutation test for the upscaled-GridRad model. These results are directly comparable with those for the MYRORSS model (Figure 7.8), because the two tests use matched storm objects. Specifically, at each time step, each GridRad storm is matched to the nearest MYRORSS storm, as long as MYRORSS centroids are within 10 km (the same method used to match storm objects for Figure 6.5). There are two notable differences between results for the two models. First, all four versions of the permutation test suggest that low-level azimuthal shear is the best MYRORSS predictor (Figure 7.8), while mid-level shear is the best upscaled-GridRad predictor (Figure D.2). However, the upscaled-GridRad model ranks low- and mid-level shear more closely to each other (first/second or first/third) than does the MYRORSS model (first/third or first/fourth). This suggests that in the upscaled-GridRad data, useful information for azimuthal shear is more distributed across the two layers. Since tornadoes occur in the low levels, *a priori* reasoning suggests that low-level shear should be the best predictor. This is true for MYRORSS but not GridRad, I suspect, because MYRORSS has much higher native resolution for azimuthal shear (0.005° versus 0.0208° grid spacing). This allows the MYRORSS model to

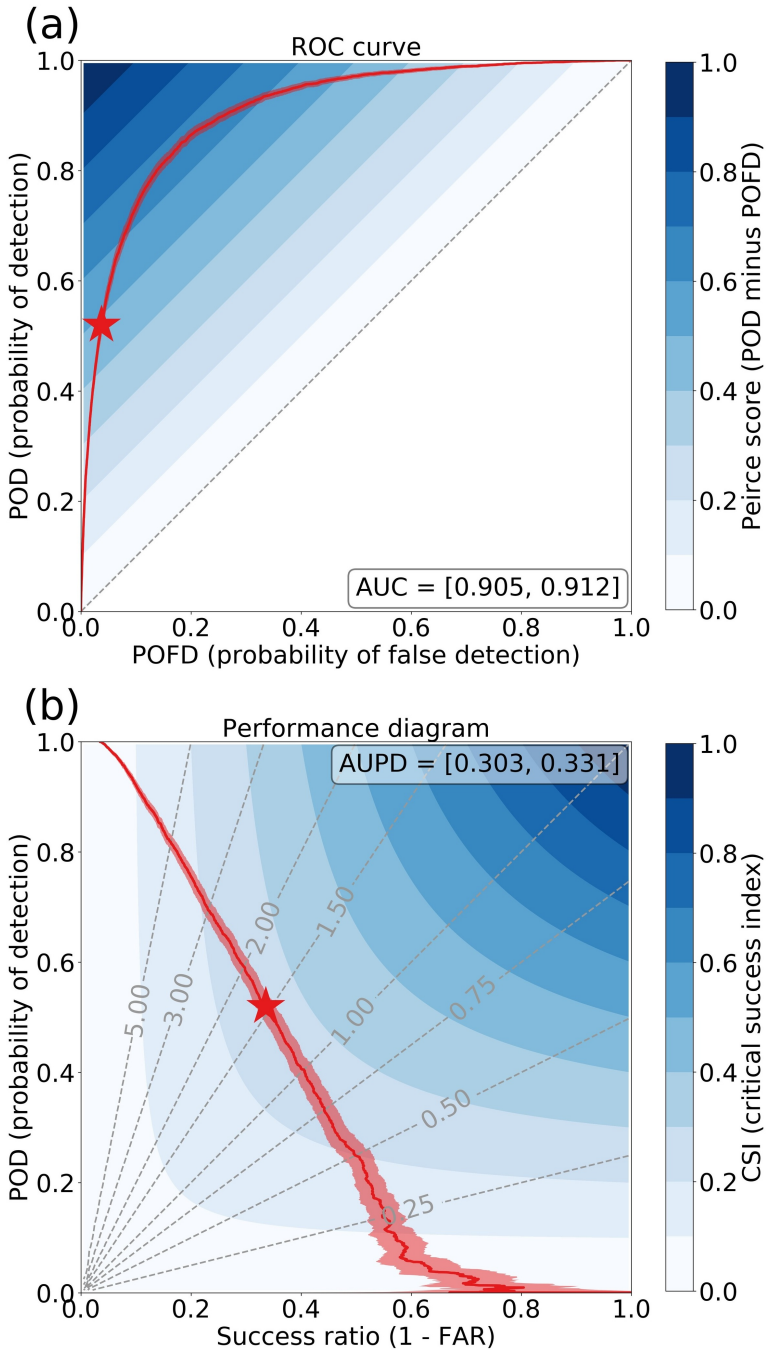


Figure D.1: Performance of chosen upscaled-GridRad model on testing data. Formatting is explained in Figure 6.3.

better identify tornadic and pre-tornadic signatures at the low levels, while the upscaled-GridRad model must rely on both low- and mid-level shear. The second notable difference is that all versions of the permutation test rank *u*- and *v*-wind as the most important sounding variables for the upscaled-GridRad model (Figure D.2), while humidity is sometimes more important for the MYRORSS model (Figure 7.8).

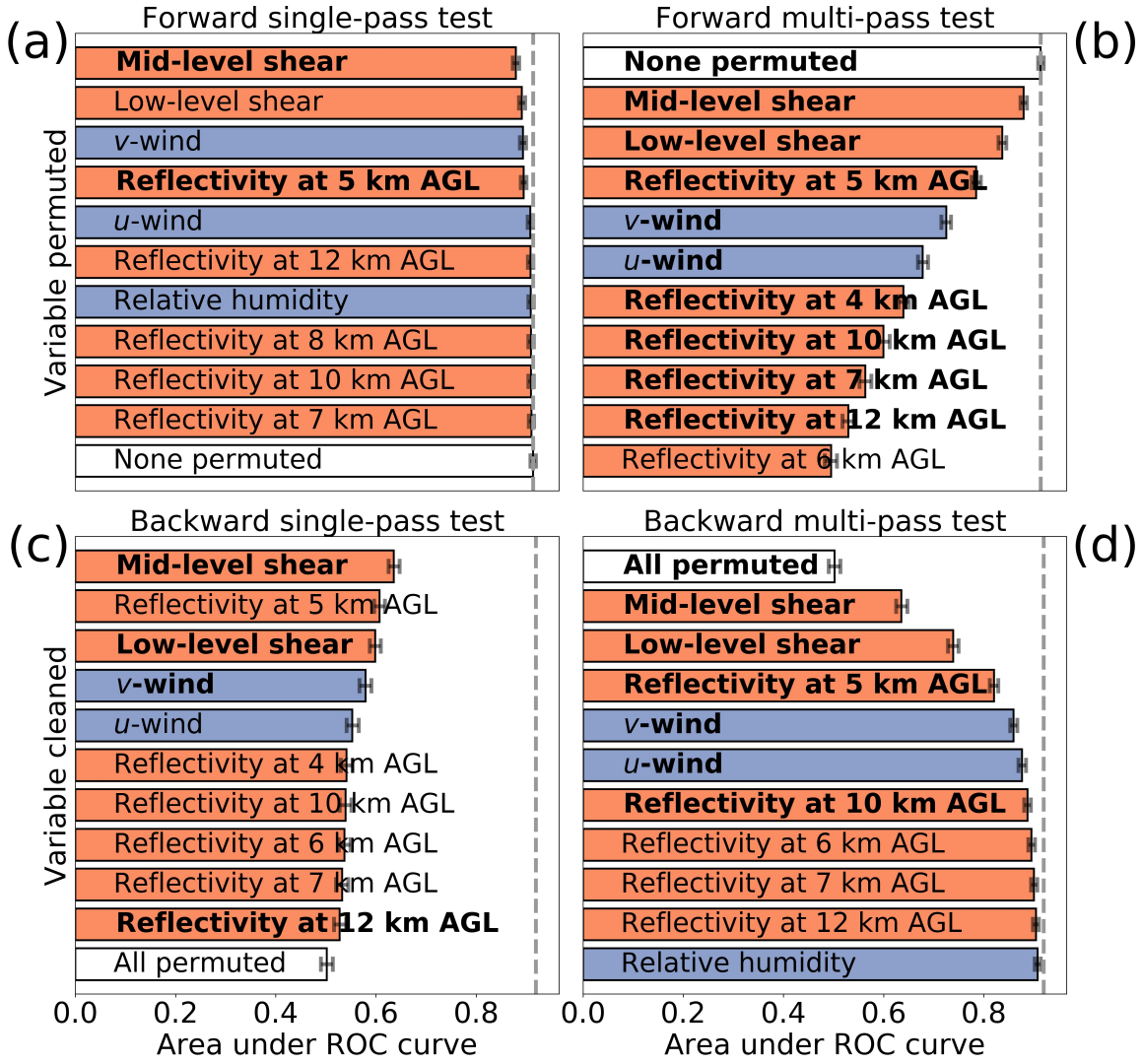


Figure D.2: Results of permutation test for upscaled-GridRad model. Formatting is explained in Figure 7.8, and this test is based on the same storms used in Figure 7.8.

Figures D.3 and D.4 show saliency maps for matched extreme cases. Matching is done with the 10-km criterion discussed above, and matched extreme cases are defined the same

way as unmatched extreme cases (Table 6.1), except that probabilities from both models are considered (Table D.1). Thus, the best hits are next-hour-tornadic storms with high probabilities from both models; the worst misses are next-hour-tornadic storms with low probabilities from both models; etc.

Table D.1: Matched extreme cases for tornado prediction. The last three columns show the mean probability forecast for each model, as well as the mean of both.

Set	**Mean Forecast Probability**		
	MYRORSS model	**Upscaled-GridRad model**	**Both**
Best hits	98.6%	98.4%	98.5%
Worst false alarms	98.1%	98.1%	98.1%
Worst misses	55.9%	17.8%	36.9%
Best correct nulls	0.3%	0.05%	0.18%

MYRORSS saliency maps for matched extreme cases (Figure D.3) are nearly identical to the original ones (Figure 7.10). Meanwhile, there are two notable differences between MYRORSS and upscaled-GridRad saliency maps. First, especially for the best hits and worst false alarms, the MYRORSS model has negative saliency for reflectivity in the RFD (Figure D.3a-b), while the upscaled-Gridrad model does not (Figure D.4a-b). Second, the upscaled-GridRad model has negative saliency for reflectivity at all heights around the edge of the storm (Figure D.4), while the MYRORSS model has negative saliency for reflectivity almost exclusively in the RFD (Figure D.3).

Overall, these differences suggest that the MYRORSS model focuses more on features within the storm, while the upscaled-GridRad model focuses more on the surrounding area and favours discrete storms. The same conclusion was made in Section 7.2 but is strengthened by this more direct comparison. I speculate that the MYRORSS model focuses more on features within the storm because the MYRORSS data have a higher native resolution

(Table 3.1), allowing the MYRORSS model to parse features within the storm more effectively.

Figures D.5 and D.6 show class-activation maps (CAM) for matched extreme cases. MYRORSS CAMs here are nearly identical to the original ones (Figure 7.11). Meanwhile, there are two notable differences between CAMs for the MYRORSS and upscaled-GridRad models. First, for the worst misses, tornado evidence in the MYRORSS model (Figure D.5c) covers a much larger area than in the GridRad model (Figure D.6c). This is likely because the MYRORSS model generally predicts much higher probabilities for these storms (Table D.1). Second, for the best hits and worst false alarms, the GridRad model finds more tornado evidence in the hook echo itself (Figure D.6a-b) – which tends to be collocated with the RFD (Markowski 2002) – while the MYRORSS model finds more evidence on the inflow side of the hook echo (Figure D.5a-b). This result does not appear in the original class-activation maps (*c.f.*, Figures 7.3a-b and 7.11a-b) and requires future investigation.

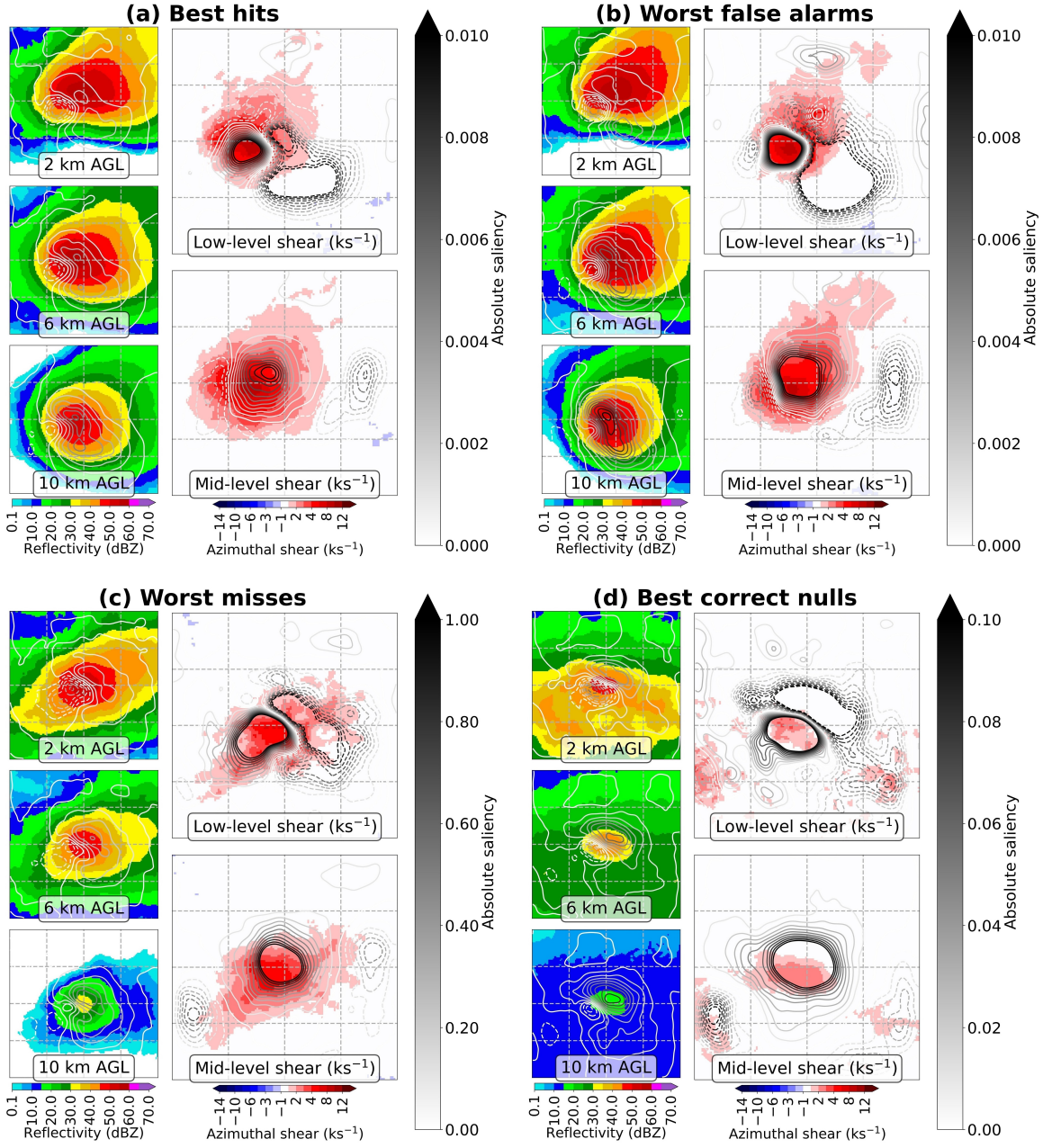


Figure D.3: Saliency maps created by MYRORSS model for matched extreme cases. Storm motion points to the right. Other formatting is explained in Figure 7.10.

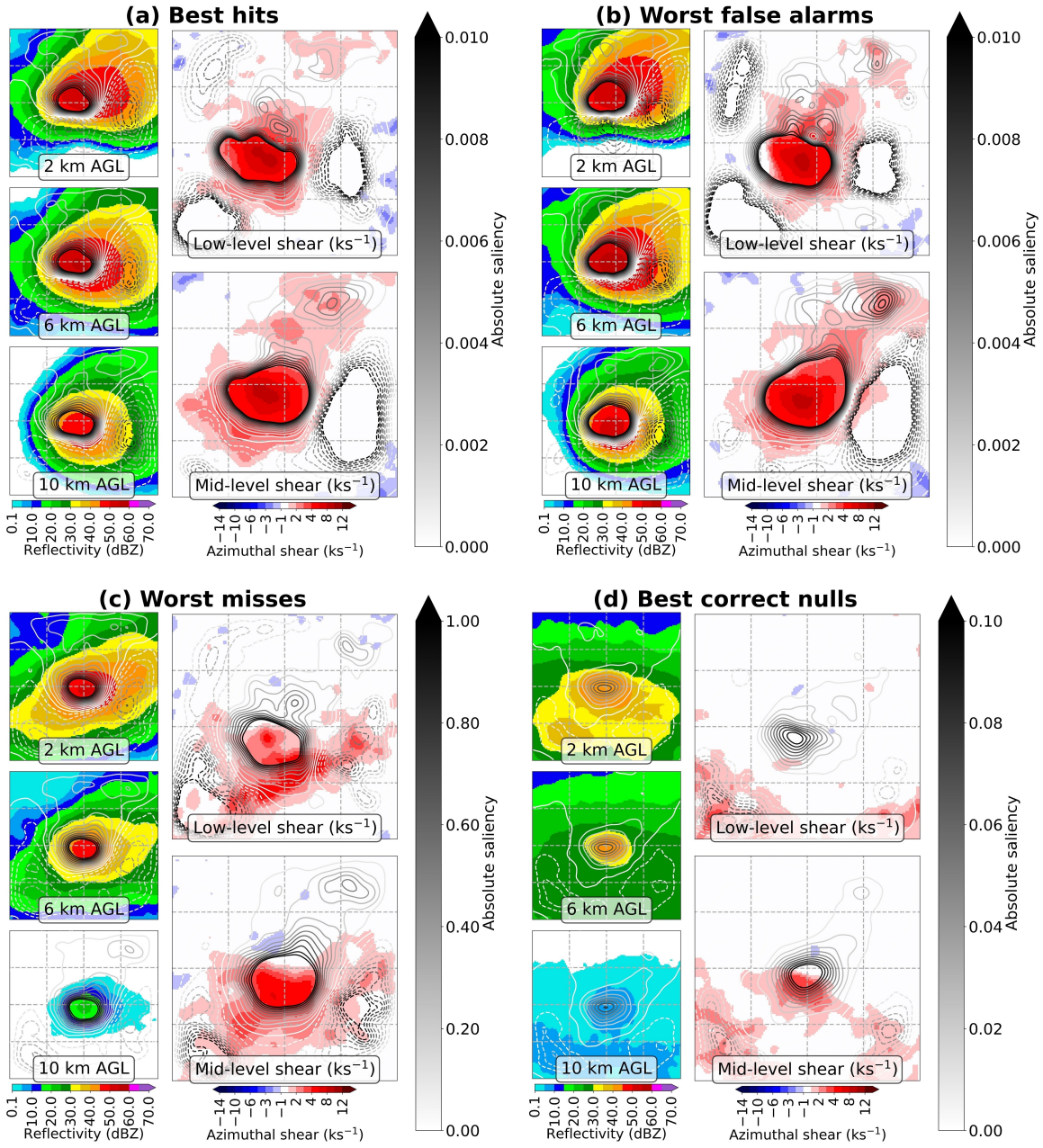


Figure D.4: Saliency maps created by upscaled-GridRad model for matched extreme cases. Storm motion points to the right, and these maps are based on the same storms used in Figure D.3.

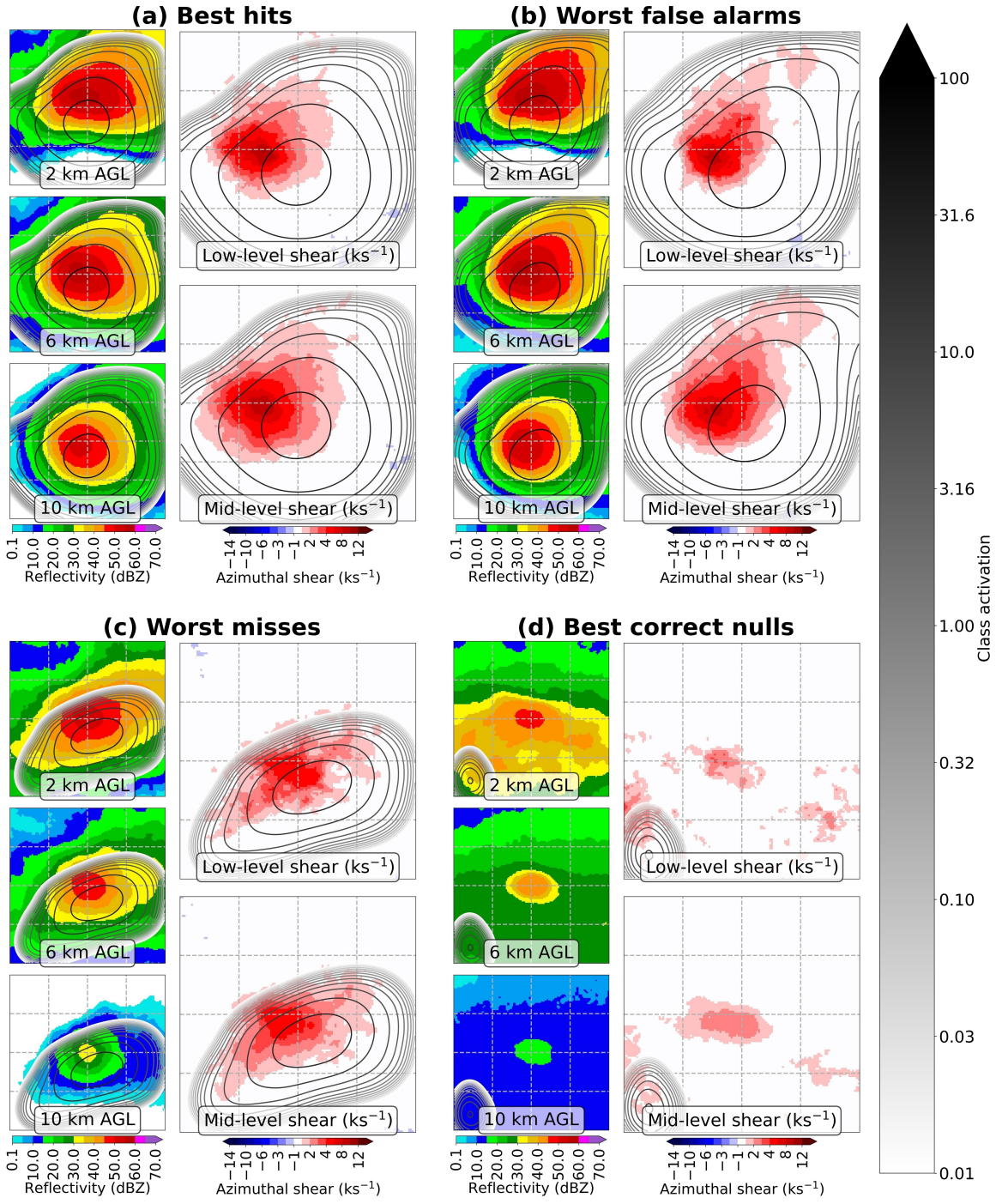


Figure D.5: Class-activation maps created by MYRORSS model for matched extreme cases. Storm motion points to the right. Other formatting is explained in Figure 7.11.

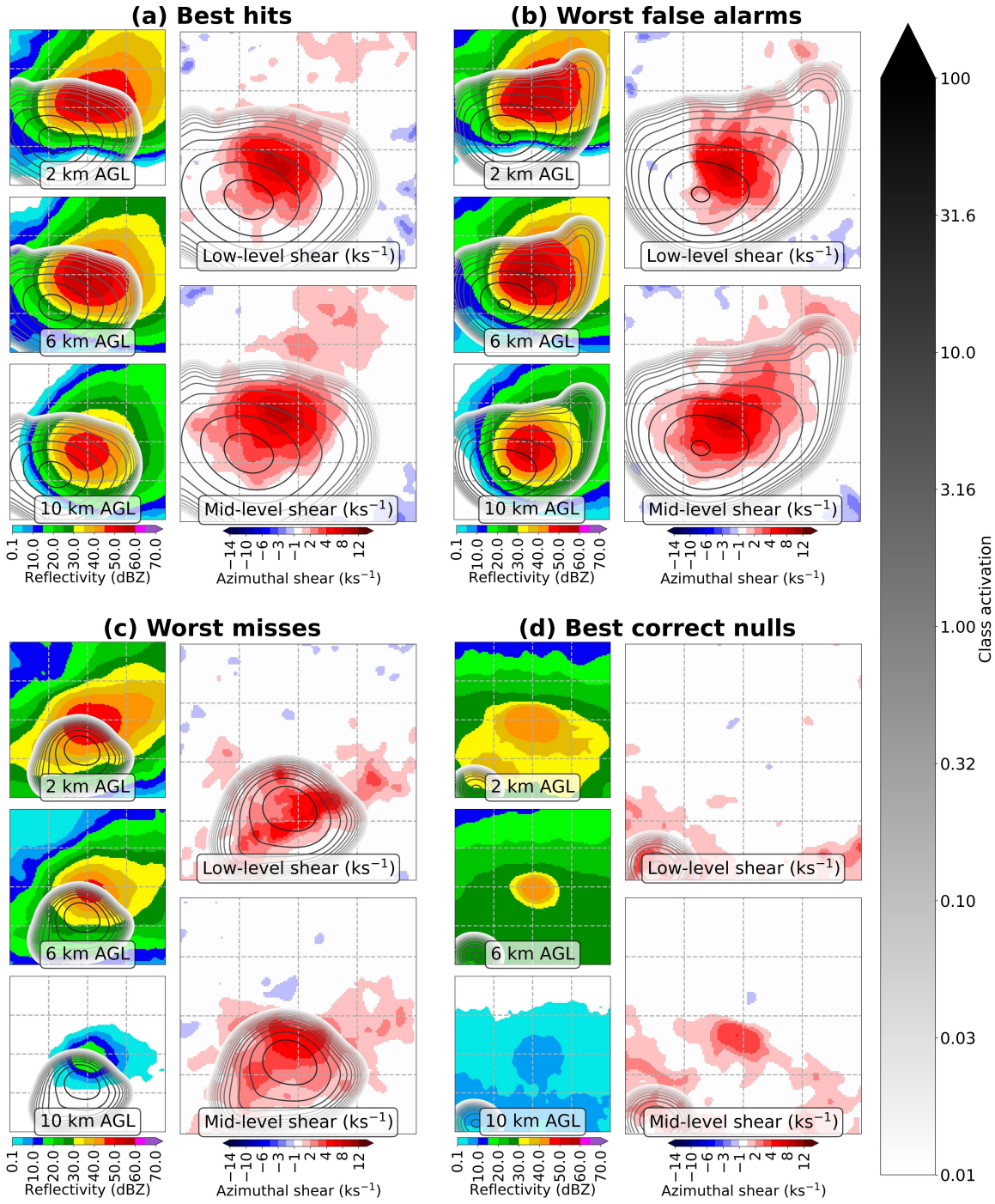


Figure D.6: Class-activation maps created by upscaled-GridRad model for matched extreme cases. Storm motion points to the right, and these maps are based on the same storms used in Figure D.5.

Appendix E

Supercomputer Configurations for Training CNNs

This study includes three types of CNNs: the MYRORSS models (*e.g.*, Figure 5.3), GridRad models (*e.g.*, Figure 5.2), and front-detection models (*e.g.*, Figure 4.1). The MYRORSS models perform 1-D and 2-D convolution; the GridRad models perform 1-D and 3-D convolution; and the front-detection models perform 2-D convolution. I have found that graphical processing units (GPU) yield a significant acceleration (reduction of 80-90% in wall-clock time) for training models with 3-D convolution but only a marginal acceleration (reduction of 33-50%) for training models without 3-D convolution. Thus, I use GPUs to train the GridRad models but only central processing units (CPU) to train the MYRORSS and front-detection models. I train the GridRad models on the Casper supercomputer provided by the National Center for Atmospheric Research (NCAR). To allow for the greatest acceleration, I use a version of Keras that has been specially compiled by the system-administrators for Casper GPUs. Each Casper node with GPUs contains eight GPUs, but Slurm (the workload-manager), which automatically allocates jobs to CPUs, does not do the same for GPUs. Thus, I use Python code in the training script[1] to allocate jobs to GPUs on the same node, where one job is the training of one CNN. I use the following header in the Slurm file, which ensures that I have exclusive access to one node with GPUs. Without exclusive access, another user's jobs could run on the same GPUs at the same time (because Slurm does not "see" GPUs), causing the GPUs to run out of memory.

#SBATCH --ntasks=8

#SBATCH --nodes=1

[1] `https://github.com/thunderhoser/GewitterGefahr/blob/master/gewittergefahr/scripts/train_many_cnns_3d_gridrad.py`

```
#SBATCH --mem=1000G
#SBATCH --partition="dav"
#SBATCH --exclusive
#SBATCH --gres=gpu:v100:8
#SBATCH --time=24:00:00
```

I train the MYRORSS and front-detection models with seven CPUs each, on either Casper or Schooner, the latter provided by the University of Oklahoma. To ensure that the seven CPUs work in parallel, I include the following lines at the top of the training script, which is in Python.

```
import os
import keras.backend as K

os.environ['CUDA_VISIBLE_DEVICES'] = '-1'

K.set_session(K.tf.Session(config=K.tf.ConfigProto(
      intra_op_parallelism_threads=7,
      inter_op_parallelism_threads=7,
      allow_soft_placement=False
)))
```

The `os.environ` command and `allow_soft_placement=False` ensure that, even if the node contains GPUs, the GPUs are not used. Requesting exclusive access to an entire node (which, as mentioned above, is necessary to prevent interference from other users' jobs) results in longer waiting times in the queue, which offset the marginal acceleration provided by GPUs for models without 3-D convolution. To request seven CPUs, I use the

following Slurm header (each CPU has 8 GB of random-access memory, so requesting 56 GB is equivalent to requesting seven CPUs).

```
#SBATCH --ntasks=1
#SBATCH --nodes=1
#SBATCH --mem=56G
#SBATCH --time=48:00:00
```