UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

UNSUPERVISED SEISMIC FACIES USING GAUSSIAN MIXTURE MODELS

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

ROBERT HARDISTY
Norman, Oklahoma
2017

UNSUPERVISED SEISMIC FACIES USING GAUSSIAN MIXTURE MODELS

A THESIS APPROVED FOR THE
CONOCOPHILLIPS SCHOOL OF GEOLOGY AND GEOPHYSICS

BY

_____
Dr. Kurt Marfurt, Chair

_____
Dr. Bradley Wallet

_____
Dr. Roger Slatt

To my family, friends, mentors

ACKNOWLEDGEMENTS

I would like to thank the sponsors of the Attribute-Assisted Seismic Processing and Interpretation Consortium for their financial support. I would also like to thank Schlumberger for providing Petrel licenses. I would also like to thank the New Zealand Petroleum and Minerals for providing public access to their seismic data.

People I would also like to thank are Tao Zhao and Lennon Infante for their previous efforts related to this research. I would like to thank Dr. Wallet for helping me understand machine learning, a complex topic that was brand new to me when I started this program. A special thanks to Dr. Slatt for letting me participate in his seminar and experience a completely different application of statistics. Finally, I would like to express my utmost gratitude to Dr. Marfurt for giving me a chance to be part of the AASPI team.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

As the use of seismic attributes becomes more widespread, multivariate seismic analysis has become more commonplace for seismic facies analysis. Unsupervised machine learning techniques provide methods of automatically finding patterns in data with minimal user interaction. When using unsupervised machine learning techniques such as K-means or Kohonen Self-Organizing Maps, the number of clusters can often be ambiguously defined and there is no measure of how confident the algorithm is in the classification of data vectors. The model-based probabilistic formulation of Gaussian mixture models (GMMs) allows for the number and shape of clusters to be determined in a more objective manner using a Bayesian framework that considers a model's likelihood and complexity. Furthermore, the development of alternative Expectation-Maximization algorithms has allowed GMMs to be more tailored to unsupervised seismic facies analysis. The Classification Expectation-Maximization algorithm classifies data vectors according to their posterior probabilities that provides a measurement of uncertainty and ambiguity (often called a soft classification). The Neighborhood Expectation-Maximization algorithm allows for spatial correlations to be considered to make classification volumes more realistic by enforcing spatial continuity. Co-rendering the classification with the uncertainty and ambiguity measurements produces an intuitive map of unsupervised seismic facies.

I apply a model-based classification approach using Gaussian mixture models to a turbidite system in Canterbury Basin, New Zealand to clarify results from an initial Self-Organizing Map and highlight areas of uncertainty and ambiguity. Special focus on a channel feature in the turbidite system using a Neighborhood Expectation-Maximization

algorithm shows it to be more realistic by considering spatial correlations within the data. I also use this model-based classification approach to highlight structural features on top of a Miocene submarine volcano in the Taranaki Basin, New Zealand.

CHAPTER 1

INTRODUCTION

With the prominence of 3D seismic data and new seismic attributes being introduced each year, multivariate analysis has become more commonplace for seismic facies analysis. Computer-based machine learning techniques provide means to automatically analyze enormous amounts of multivariate data, identifying patterns that would otherwise be overlooked by a human interpreter with a limited amount of time. Machine learning techniques are frequently divided into two categories: unsupervised and supervised learning. In the supervised case, the interpreter provides significance guidance to the algorithm concerning the hidden state of nature such as the underlying seismic facies. For example, an interpreter might instruct the computer to learn the difference between reservoir and non-reservoir facies in seismic data by providing the algorithm with a subset of the data that the interpreter knows represent the two cases. Ideally, the learning algorithm would then be able to find similar patterns within the larger dataset to differentiate the two. In unsupervised learning, patterns are automatically identified within the data with minimal interaction between the interpreter and the algorithm. In this scenario, patterns are automatically found by feeding the full dataset into the computer without any labeling of facies, allowing the data to "speak for themselves". In this thesis, I use an unsupervised learning approach using Self-Organizing Maps (SOM) (Kohonen, 1982) in conjunction with Gaussian mixture models to generate unsupervised seismic facies that can be assigned meaning through post-processing by the interpreter.

Self-Organizing Maps (SOM) is an unsupervised learning technique that has been previously applied to seismic facies analysis (*e.g.* Zhao et al., 2016; Roden et al., 2015;

Marroquin et al., 2009; Coleou et al., 2003; Strecker and Uden, 2003), with several commercial SOM analysis tools currently available. SOM aides in the understanding of high dimensional data by non-linearly projecting them onto a lower, usually 2-dimensional, latent space such that similar data points fall close to each other. Although natural clusters are formed on an SOM, the number of clusters can often be ambiguous and subjective. Gaussian mixture models offer a more quantitative assessment of SOM latent spaces because of its probabilistic formulation.

In general, mixture models are model-based statistical tools to approximate probability density functions. Gaussian mixture models (GMM) are good at modeling multimodal distributions by having a Gaussian distribution for each mode. Hathaway (1986) described GMM as a fuzzy-clustering technique. There are several different optimization criteria to solve different clustering problems. In this method, I define a GMM to be a non-orthogonal decomposition of the data that attempts to recover the true state of nature represented by the clusters. The classification approach using GMM to partition the data into different clusters is much like the K-means algorithm (MacQueen, 1967). In fact, the classification approach becomes identical to the K-means algorithm under certain conditions, such that GMM can be thought of as a probabilistic extension of K-means. Because GMMs are rooted in probability theory, GMMs with different shapes and numbers of clusters can be quantitatively compared to each other using likelihood estimates and other model selection criteria.

GMMs have had some success in seismic facies analysis (Han et al., 2010), but I am unaware of any previous material that combines GMM with SOM for seismic facies analysis. SOM is a beneficial first step because it reduces the dimensionality of the data,

and forms natural clusters of the data. Reducing the dimensionality of the data is an important process due to the curse of dimensionality (Bellman, 1957; Wallet, 2012). Using the SOM latent space as input to a GMM allows for the number of clusters to be objectively assessed, along with a classification of the data and measurements of uncertainty and ambiguity. Finally, GMMs offer a more quantitative approach to the interpretation of SOM latent spaces than simple visualization, and interactively picking polygons about a collection of neurons.

I begin my thesis with a review of the theory behind GMMs in Chapter 2. Next in Chapter 3, I apply the model-based classification approach using GMMs to a previously interpreted turbidite system in Canterbury Basin, New Zealand with special focus on a channel feature. In chapter 4, I once again apply the model-based classification approach to a Miocene submarine volcanic body in the Taranaki Basin, New Zealand. Finally, in Chapter 5, I conclude with analysis of clusters and their significance to interpretation. Descriptions of the different algorithms used to find mixture models are added as Appendices.

CHAPTER 2

Gaussian mixture model theory

A multivariate Gaussian distribution can be defined as follows:

$$\varphi(\mathbf{x}|\mu,\mathbf{C}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\mathbf{C}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})^{T}} \quad (2.1)$$

where d is the number of input attributes, $\mathbf{C}$ is a covariance matrix and $|\mathbf{C}|$ is the determinant of the covariance matrix, $\boldsymbol{\mu}$ is the mean, and the symbol, ^{T}, denotes the transpose of a matrix or vector. In general, a GMM approximates a probability density function by adding together weighted Gaussian distributions (Figure 2.1). A GMM can also be used for clustering and classification where a cluster of a GMM can be described by its weighted Gaussian distribution. A Gaussian mixture density with K clusters for a data vector $\mathbf{x}_n$, where $n = 1...N$, is given by

$$p(\mathbf{x}_n|\psi) = \sum_{k=1}^{K} \pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k) \quad (2.2)$$

where ψ denotes the parameters of the GMM, and π_k is the weight of the k^{th} cluster such that

$$\sum_{k=1}^{K} \pi_j = 1 \quad (2.3)$$

and $\pi_k \geq 0$. The posterior probability, $w_{n,k}$, of a data vector, $\mathbf{x}_n$, belonging to a certain cluster, k, is given by

$$w_{n,k} = \frac{\pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k,\mathbf{C}_k)}{\sum_{k=1}^{K} \pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k,\mathbf{C}_k)} = \frac{\pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k,\mathbf{C}_k)}{p(\mathbf{x}_n|\psi)}. \quad (2.4)$$

A data vector is then classified as belonging to the cluster with the greatest posterior probability, $\max_k(w_{n,k})$. Bensmail et al. (1996), propose to measure the uncertainty in the classification for a data vector, $\mathbf{x}_n$, using

$$A_n = 1 - \max_k(w_{n,k}), \tag{2.5}$$

and from now on will be referred to as ambiguity to distinguish itself from the GMMs uncertainty in describing the data vector. A data vector that lies in between two clusters is ambiguous but can still be described well by the model. To quantitatively assess the GMMs uncertainty in describing a data vector the Mahalanobis distance (Mahalanobis, 1936) can be used. The Mahalanobis distance from the data vector, $\mathbf{x}_n$, to the cluster that it belongs to is

$$U_n = \sqrt{(\mathbf{x}_n - \boldsymbol{\mu}_k)\mathbf{C}_k{}^{-1}(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}} \tag{2.6}$$

where $\boldsymbol{\mu}_k$ and $\mathbf{C}_k$ are the mean and covariance of the cluster that the data vector is classified to. Data vectors with a large Mahalanobis distance are farther from the cluster and are interpreted as more more uncertain and can be anomalous.

One method for objectively and quantitively generating a parsimonious model is to generate models with a range of differing clusters and covariance parameterizations, and then select the best model using a criterion such as the Bayesian Information Criterion (BIC) (Fraley and Raftery, 1998). The selected model can then be used to classify data vectors as being from a certain cluster to generate unsupervised seismic facies. Under this classification scheme, data vectors are assumed to belong to a single Gaussian distribution.

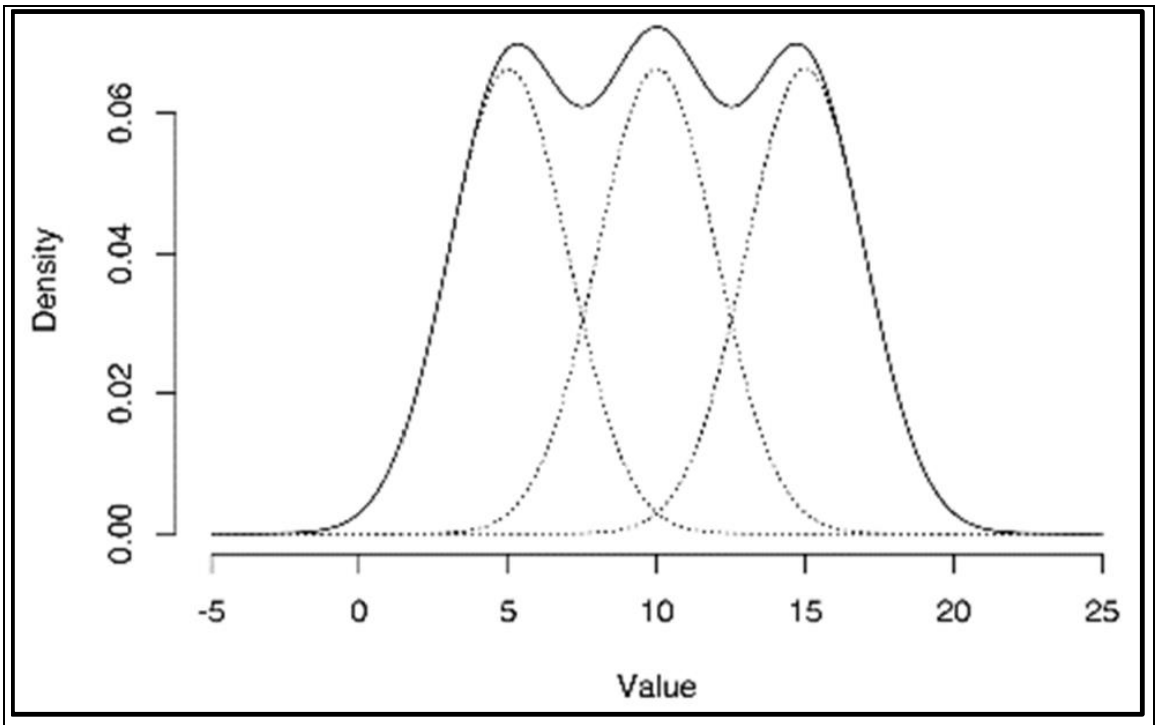

Figure 2.1. Example of a GMM with three mixture components. The overall density is estimated as the sum of the three Gaussian components. This density can be used to classify data vectors as well as provide a measure of uncertainty and ambiguity in the classification. (Modified from Wallet, Altimar, and Slatt 2014).

Initialization and Optimization

GMM initialization begins with a k-means clustering algorithm using the same number of clusters. The means, $\boldsymbol{\mu}_k$, and the covariance matrices, $\mathbf{C}_k$ of each cluster are initialized to be those computed using k-means (Macqueen, 1967), while the weights, π_k, are initialized to be the fraction of attribute vectors assigned to each k-means cluster. Dempster et al. (1977) learns the mixture parameters, $\{\pi_k, \boldsymbol{\mu}_k, \mathbf{C}_k,\}$, using an Expectation-Maximization (EM) algorithm that seeks to maximize the observed log-likelihood of ψ given by

$$L(\psi) = \sum_{n=1}^{N} \log\{ p(\mathbf{x}_n|\psi)\} = \sum_{k=1}^{K} \log\{ \sum_{n=1}^{N} \pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)\} \qquad (2.7)$$

If the objective is to classify data vectors, alternative EM algorithms such as Celeux and Govaert's (1992) Classification Expectation-Maximization (CEM) can be used where data vectors are assumed to be produced by a single Gaussian distribution. This classification approach defines a hard partition like the K-means algorithm and is interesting because the CEM becomes identical to K-means when the Gaussian mixture is parameterized as having equal proportions with clusters sharing a common covariance matrix of $\sigma^2\mathbf{I}$ where $\mathbf{I}$ is the identity matrix and $\sigma^2$is an unknown scalar that ends up not affecting the classification (Celeux and Govaert, 1992). Celeux and Govaert (1992) propose using a Stochastic Expectation-Maximization (SEM) algorithm in conjunction with the CEM when data sets are large. The SEM algorithm iterates for a user-defined number of times and the iteration with highest complete log-likelihood,

$$L_{\mathrm{c}}(\psi) = \sum_{k=1}^{K}\sum_{n=1}^{N} z_{nk}\log\{\pi_k\varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)\}, \tag{2.8}$$

is selected as the best model. The SEM algorithm simulates a GMM by guessing at the classification of data vectors. The randomness of the SEM algorithm helps to avoid the heavy dependence on initialization that the CEM algorithm relies on to find a maximum. The selected model from SEM can then optimized by the CEM algorithm until there is no change in the classification of data vectors (Figure 2.2).

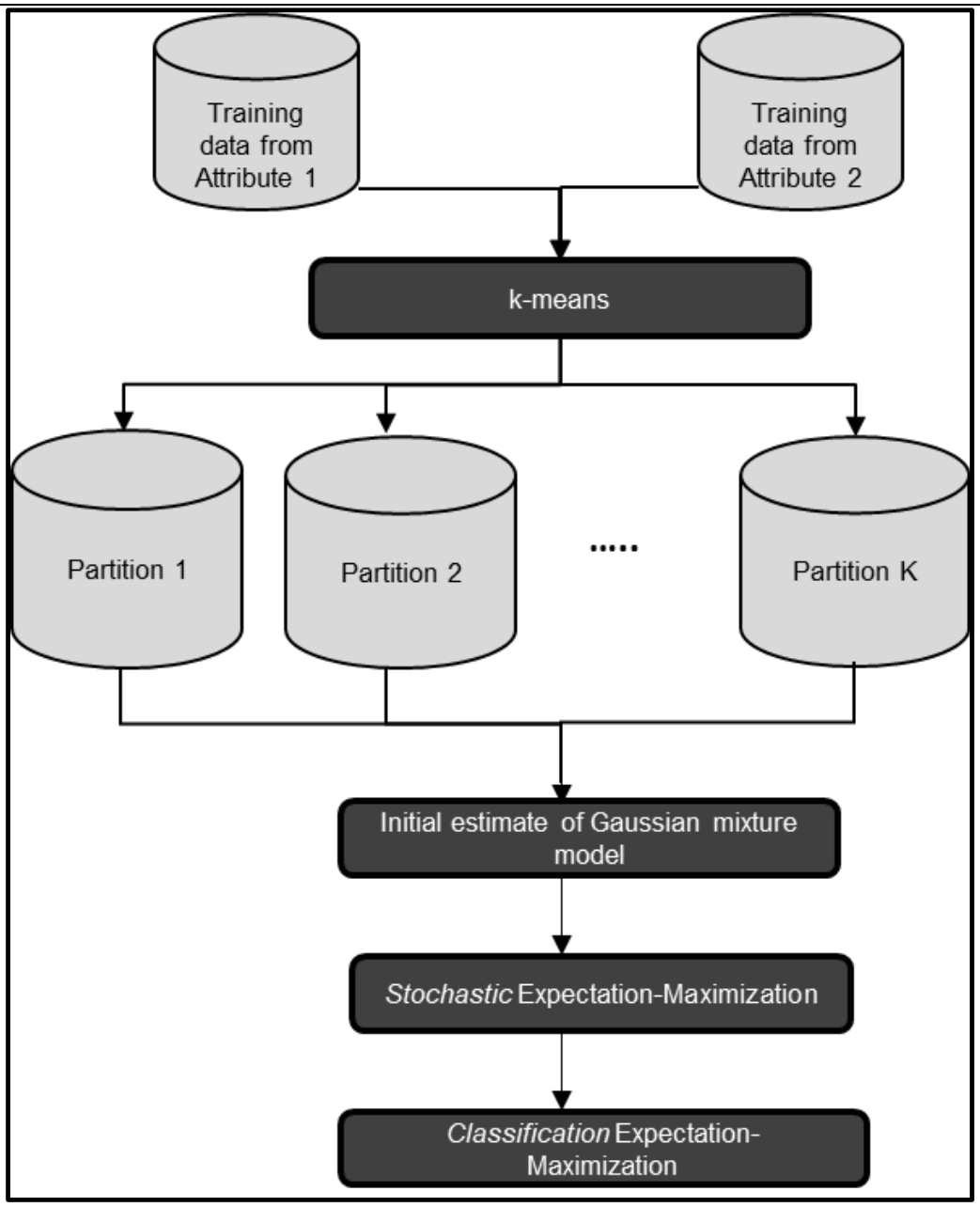

Figure 2.2. Flow chart for GMM generation. The training data is input to the k-means algorithm to create K partitions of the data that are used to provide the initial estimates of the GMM. The Stochastic Expectation-Maximization algorithm helps to avoid sub-optimal solutions by introducing a random component into the optimization process. The Classification Expectation-Maximization provides a final more deterministic solution.

Covariance matrix parameterizations

Following Celeux and Govaert (1993), the size, shape, and orientation of each covariance matrix associated with each cluster can be controlled through use of eigen-decomposition. The eigenvalue decomposition for a covariance matrix, $\mathbf{C}_k$, of the k^{th} cluster can be given by

$$\mathbf{C}_k = \alpha_k \mathbf{D}_k \mathbf{\Lambda}_k \mathbf{D}_k^{\mathrm{T}} \tag{2.9}$$

where $\alpha_k = |\mathbf{C}_k|^{1/d}$, $\mathbf{D}_k$ is the matrix of eigenvectors, and $\mathbf{\Lambda}_k$ is a diagonal matrix containing eigenvalues in decreasing order and normalized so their sum is unity. Using this decomposition α_k controls the volume of the covariance, $\mathbf{D}_k$ controls the orientation, and $\mathbf{\Lambda}_k$ controls the shape. By controlling the size, shape and orientation of the covariance matrix, the number of parameters needed to estimate a GMM can be reduced and more parsimonious models can be generated. For example, if the clusters were to have the same shape but different volumes, then the covariance parameterization would be written as $\mathbf{C}_k = \alpha_k \mathbf{D\Lambda D}^{\mathrm{T}}$. By forcing the clusters to share the parameters, $\mathbf{D\Lambda D}^{\mathrm{T}}$, the total number of parameters necessary to describe a mixture model will be reduced making the model more parsimonious. Alternative means of varying these parameters results in eight different workflows. Furthermore, Celeux and Govaert (1993) also consider parameterizations of the form

$$\mathbf{C}_k = \alpha_k \mathbf{I} \tag{2.10}$$

and

$$\mathbf{C}_k = \alpha_k \mathbf{B}_k \tag{2.11}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{B}_k$ is a diagonal matrix. This results in an additional two parameterizations for equation 2.10 and an additional four parameters for equation 2.11, resulting in a total of 14 parameterizations. Some parameterizations cannot be solved in closed form. The nine closed-form solutions from Celeux and Govaert (1993) are considered for this thesis (Table 2.1).

A trivial optimization of either log-likelihood function can be done by taking the limit as the determinant of the covariance matrix approaches zero resulting in an infinite likelihood value. Unfortunately, EM-based algorithms can converge to these singular matrices, especially if the data are redundant as is the case for seismic data (Marroquín et al., 2009). To deal with this challenge, a minimum width for all Gaussian distributions is used to prevent numerical instabilities associated with singular covariance matrices.

Table 2.1. Covariance parameterizations. Each covariance parameterization has a different number of parameters. The parameterizations are ordered from most parsimonious at the top to the most general in the bottom. The BIC has a penalty function that increases with the number of parameters.

Module symbol	**Covariance parameterization**	**Description**
EII	$\alpha\mathbf{I}$	Spherical. Equal volume.
VII	$\alpha_k\mathbf{I}$	Spherical. Varying volume.
EEI	$\alpha\mathbf{B}$	Diagonal. Equal volume. Equal shape.
EVI	$\alpha\mathbf{B}_k$	Diagonal. Equal volume. Varying shape.
VVI	$\alpha_k\mathbf{B}_k$	Diagonal. Varying volume. Varying shape.
EEE	$\alpha\mathbf{D}\mathbf{\Lambda}\mathbf{D}^{\mathrm{T}}$	Ellipsoidal. Equal volume. Equal shape. Equal orientation.
EEV	$\alpha\mathbf{D}_k\mathbf{\Lambda}\mathbf{D}_k^{\mathrm{T}}$	Ellipsoidal. Equal volume. Equal shape. Varying orientation.
EVV	$\alpha\mathbf{D}_k\mathbf{\Lambda}_k\mathbf{D}_k^{\mathrm{T}}$	Ellipsoidal. Equal volume. Varying shape. Varying orientation.
VVV	$\alpha_k\mathbf{D}_k\mathbf{\Lambda}_k\mathbf{D}_k^{\mathrm{T}}$	Ellipsoidal. Varying volume. Varying shape. Varying orientation.

Model selection

Likelihood is an effective way to compare models of the same complexity, but is not sufficient to compare models of differing complexity (i.e. differing number of clusters and covariance parameterization). Consider adding an additional cluster to any GMM and setting that cluster's weight to zero. The likelihood with and without the additional component would be the same, but the additional component doesn't add any useful information, such that the model without this additional component would be preferred (Wallet et al., 2014). In fact, any data set can be perfectly modeled with a GMM where the number of terms equal the number of observations and all variances being zero.

The model-based approach allows for model selection to be done in a more principled and formal way than heuristic approaches such as used in k-means (Figueiredo and Jain, 2002). Finding the true number of clusters in a data set may be an ill-posed problem in general, but there have been many different criteria that have been used to select the best model from amongst a suite of candidate models, including the Akaike (1973) Information Criterion (AIC) and the Minimum Message Length (Oliver, Baxter, and Wallace, 1996) technique. In this thesis, I use Schwarz's (1978) Bayesian Information Criterion (BIC) to choose the best model. The BIC is defined as

$$BIC = \log\big(L(\psi)\big) - \frac{1}{2} m \log(N), \tag{2.12}$$

where $L(\psi)$ is the observed log-likelihood of the mode, N is the number of training vectors, and m is the number of estimated parameters. The higher the BIC value, the stronger the evidence is for the model having the correct number of clusters and covariance parameterization.

Often the first decisive local maximum is selected as the best model, but for the case where a maximum doesn't exist a model is selected based on how much the BIC increases with successive clusters (Dasgupta and Raftery, 1995), effectively picking the shoulder of the curve. Fraley and Raftery (1998) find that although the regularity conditions for the BIC approximation don't hold for mixture models, the BIC method works well in practice. The BIC penalizes extraneous parameters and large sample sizes to choose the best model in terms of parsimony and likelihood.

In summary, the model-based classification methodology is as follows:

1. Define a minimum and maximum number of clusters [K_{min} , K_{max}] and a set of covariance parameterizations,
2. Condition the data using a Z-score transform *(optional),*
3. Decimate the inlines, crosslines, and vertical samples of the seismic attributes to create a training set of data vectors according to memory constraints,
4. Repeat steps 4-8 for each number of clusters and covariance parameterization for the training set of data vectors,
5. Generate an initial partition of the data using k-means
6. Initialize the stochastic expectation maximization (SEM) algorithm using the centroids from k-means as the mean, the within-cluster scattering matrix for the covariance, and weights proportional to the partitioning done by the k-means algorithm. Iterate SEM for a set number of times and find the best model according to the complete log-likelihood in equation 2.8,

7. Initialize the classification expectation maximization (CEM) algorithm using the results from SEM. Iterate CEM until the partition does not change.
8. Calculate the Bayesian Information Criterion (BIC).
9. Plot the BIC function for all models. The first decisive maximum provides strong evidence for the model.
10. Classify all the data vectors according to the selected model and calculate the uncertainty and ambiguity for each data vector to produce two seismic attribute volumes.

Synthetic example

Figure 2.3 shows a synthetic model used to illustrate the model based classification application to 3D seismic attributes where the two synthetic seismic attribute volumes represent three laterally continuous layers. The three layers were given attribute values that are drawn from the different 2D Gaussian distributions defined by Table 2.2. The synthetic data extend vertically 101 samples, with the first layer ranging between samples 1-10, the second between samples 11-50, and the third between samples 51-101 (the extent of inlines and crosslines are irrelevant due to the synthetic layers being laterally continuous).

Setting unknown number of clusters is set to range between a minimum of 1 and a maximum number of 10. All 14 covariance parameterizations are evaluated. The data are preconditioned using a Z-score transform (Figure 2.4). To create the training set of data vectors, the two synthetic attributes are decimated only in their inlines and crosslines because the synthetic attributes only vary vertically. A total of 21,816 training data

vectors are used in learning the mixture models. For each covariance matrix parameterization and for the number of candidate clusters, an initial partition is generated using the k-means algorithm, followed by optimization of the mixture model using the EM variants, SEM and CEM. For each mixture model, BIC values are shown in Figure 2.5, and the mixture model with three clusters and a covariance parametrization of $\lambda\mathbf{B}$ is selected as the best model because it has the highest BIC value. Applying the selected mixture model to the all the data vectors to produces a classification seismic volume.

The ellipsoids in the cross-plot of the Z-scores show the mean and covariance of each cluster in the selected GMM, while the mixture model weights of each cluster correspond to each layer's relative volume (Table 2.2). The classification correctly identifies the three layers and each layer is given a distinct color (Figure 2.6). Although the synthetic data were generated using the identity matrix, **I**, for the covariance matrix, the Z-score transform changes the variance along each attribute meaning the diagonals of the covariance matrix are no longer unity. This change requires the use of different variables along the diagonal to estimate the covariance matrix and explains why the covariance parameterization of $\lambda\mathbf{B}$ is correct.

Table 2.2. Synthetic attribute values and their distribution. The synthetic attribute values were sampled from a Gaussian distribution. The two attributes create a 2D attribute space from which three clusters can be identified.

	Synthetic Attribute 1	**Synthetic Attribute 2**	**Vertical Samples**	**By volume**
First layer	φ(10,1)	φ(10,1)	1-10	9.9%
Second layer	φ(20,1)	φ(10,1)	11-50	39.6%
Third layer	φ(10,1)	φ(30,1)	51-101	50.5%

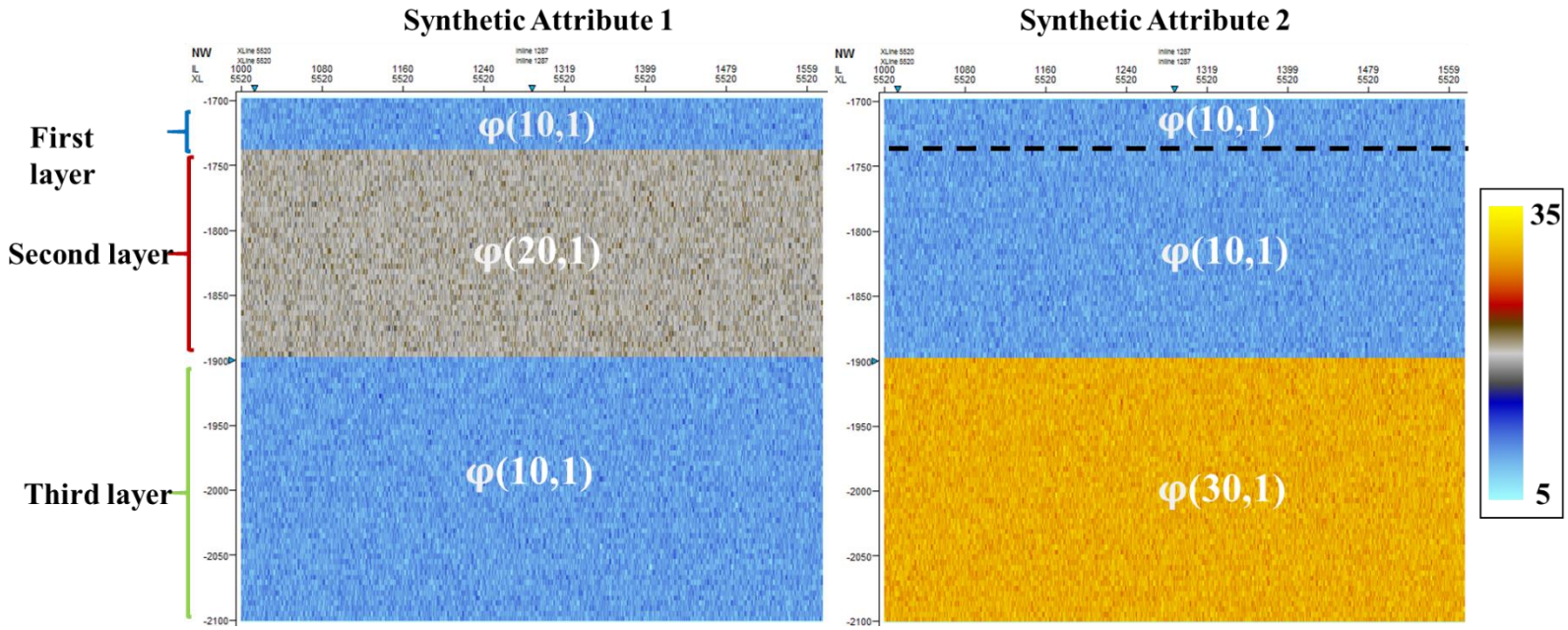

Figure 2.3. Cross-section of synthetic attributes. The layers are laterally continuous and their vertical extent can be seen on the left. The values of each layer were sampled from a Gaussian distribution described by the white text, $\varphi(\mu, \mathbf{C})$.

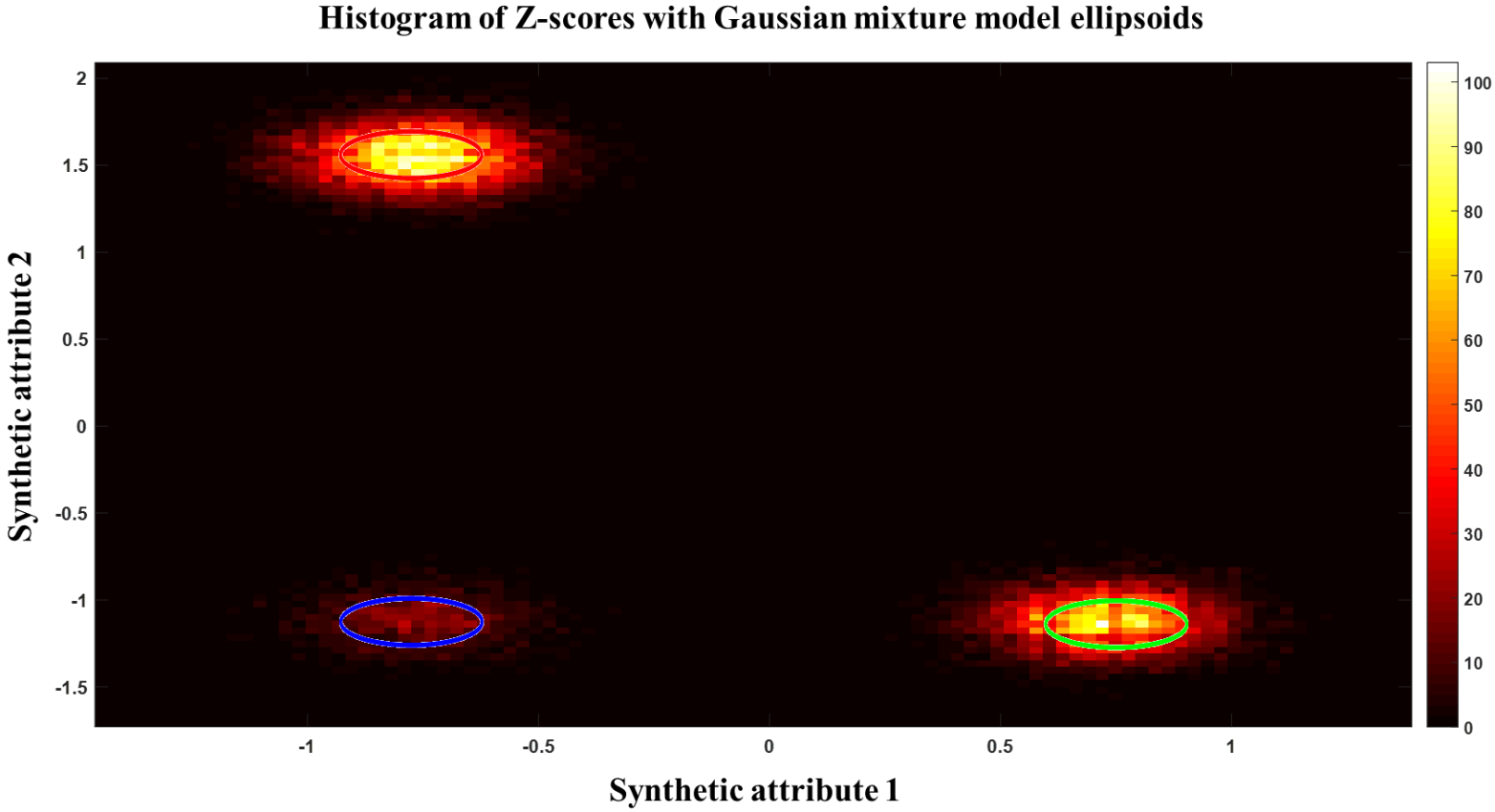

Figure 2.4. Histogram of Z-scores with Gaussian ellipsoids. The color of the ellipsoids matches the colors of the classification profile. The green cluster (bottom right) contains 50.5% of the model weight, the red (top left) 39.6%, and the blue (bottom right) 9.9%. Histogram is made using 100x100 bins.

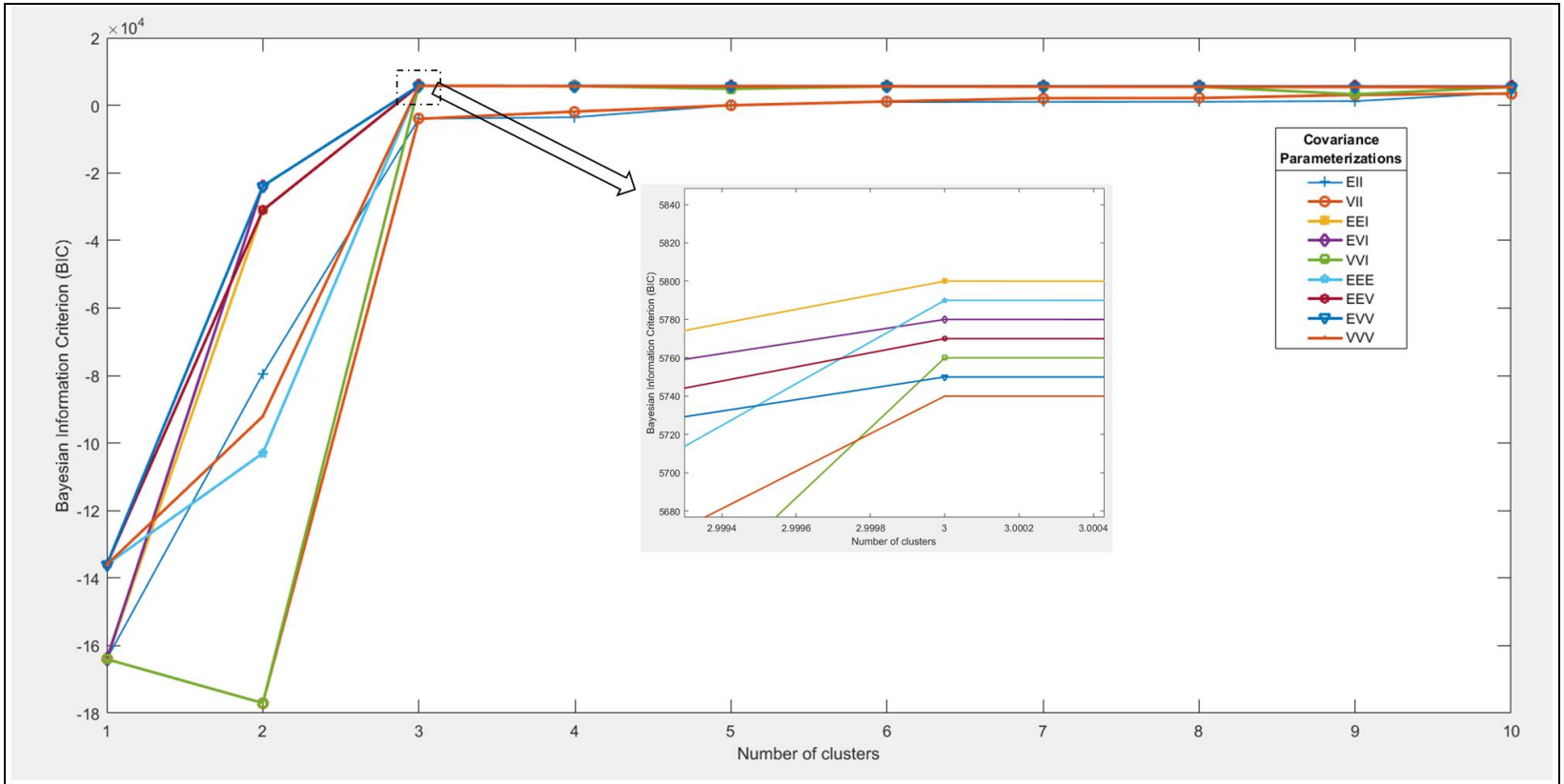

Figure 2.5. BIC for synthetic. For each covariance parameterization and number of clusters, the Bayesian Information Criterion (BIC) is calculated to evaluate each model in terms of likelihood and parsimony. The model with three clusters and a covariance parameterization of $\lambda\mathbf{B}$ has the highest BIC value and corresponds to the correct underlying distribution by which the data is sampled from.

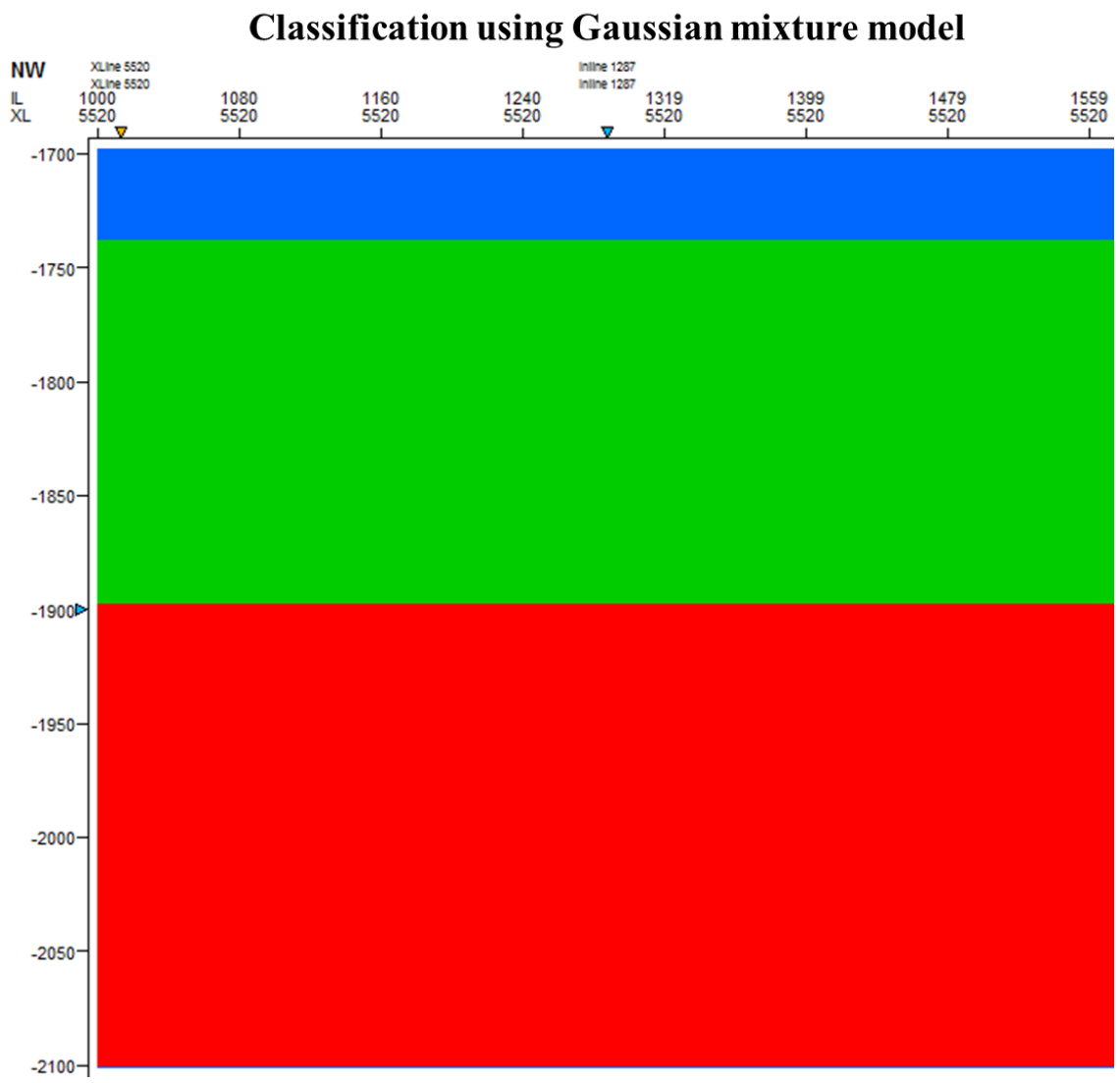

Figure 2.6. Classification of synthetic attributes. The classification shows three different layers which correspond to the underlying Gaussian distributions that the data was sampled from.

CHAPTER 3

CANTURBURY BASIN, NEW ZEALAND- WAKA3D

3.1 *Geologic setting and previous latent space modeling*

The seismic survey is located on the Canterbury Basin, offshore New Zealand (Figure 1). The Canterbury Basin contains more than 6000ft of Cretaceous to Tertiary fill (Cozens, 2011). Sediments in the basin were deposited in a single transgressive-regressive cycle driven by tectonics (Zhao and Marfurt, 2014). The seismic survey is in the transition zone of the continental rise and continental slope and contains many paleocanyons and turbidite deposits (Zhao and Marfurt, 2014).

A Miocene turbidite system was interpreted using a phantom horizon slice tied to a picked continuous reflector below the turbidite system by Zhao et al. (2016) (Figure 3.2, 3.3, and 3.4). Using a Self-Organizing Map (SOM) (Kohonen, 1982) Zhao et al. (2016) projected the information of four different attributes, peak spectral frequency, peak spectral magnitude, coherent energy, and curvedness (Figure 3.5), onto a 2D manifold. Peak spectral frequency and peak spectral magnitude were used to represent variation of the seismic response, coherent energy to the amplitude response, and curvedness to represent the magnitude of structural deformation (Zhao et al., 2016).

Following Zhao et al.'s (2016) interpretation, the red arrows in Figure 3.6 indicate possible slope fans, blue arrows high amplitude deposits, white arrows stacked channels, and black arrows sinuous channels that were difficult to see on the horizon slice through seismic amplitude shown in Figure 3.4.

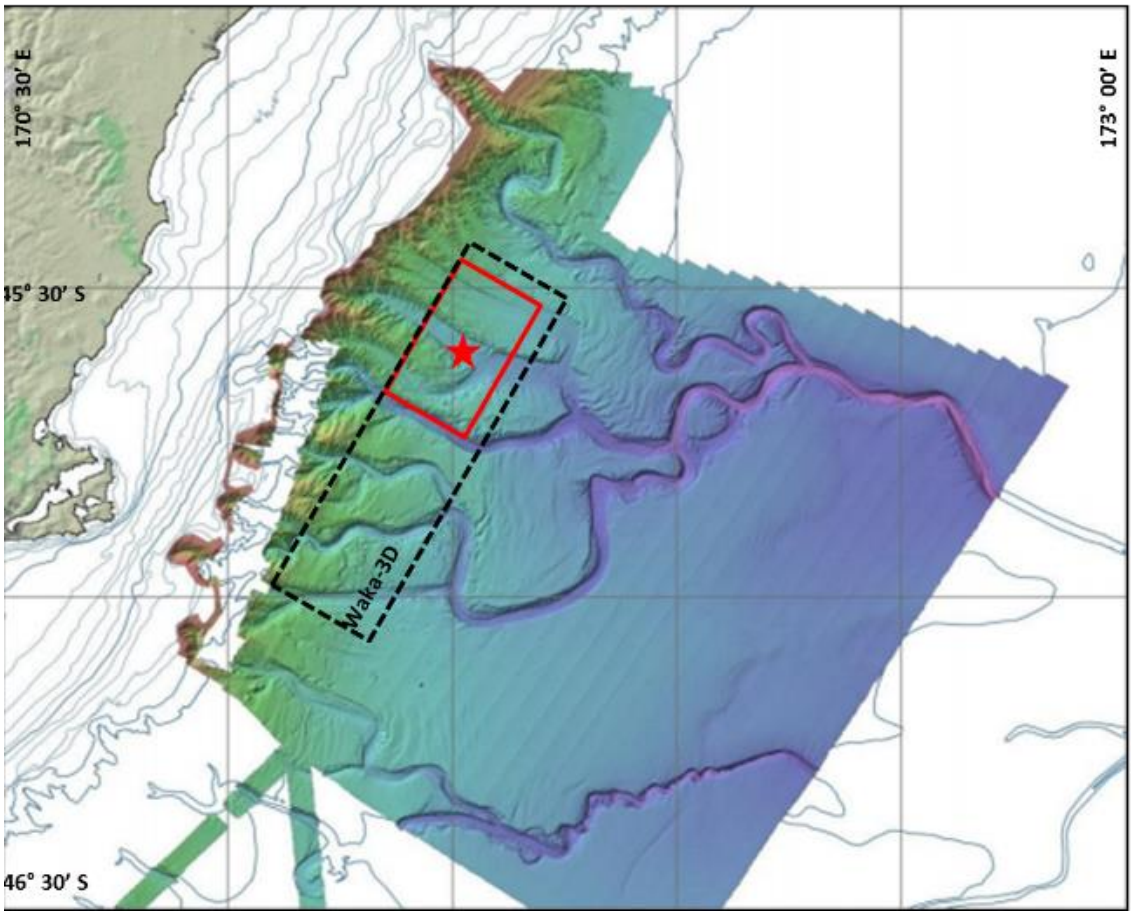

Figure 3.1. Aerial view of Waka 3D survey and study area. Modified from Zhao et al. (2016).

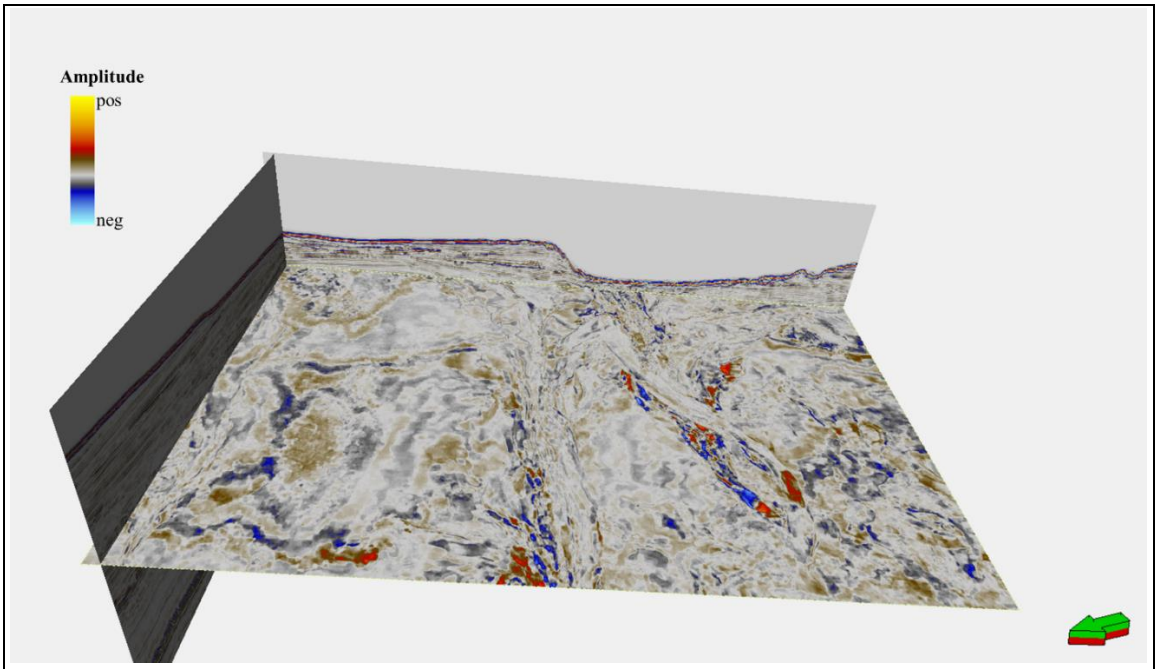

Figure 3.2. Amplitude slice at *t*=1.88s. Looking downdip.

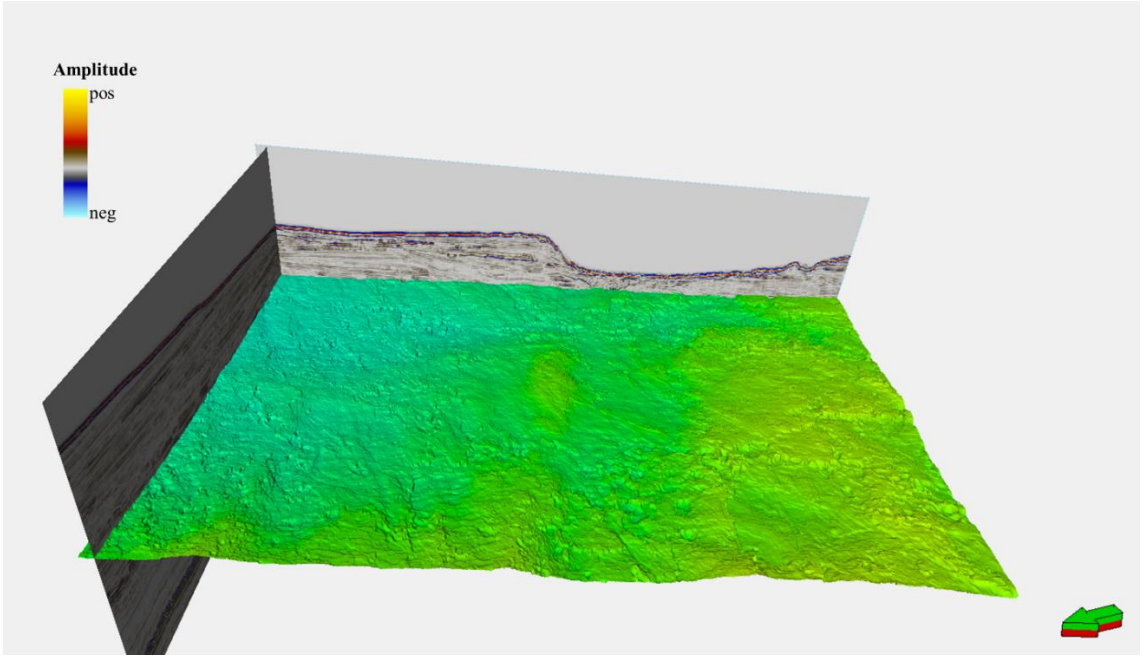

Figure 3.3. The picked horizon. Looking downdip.

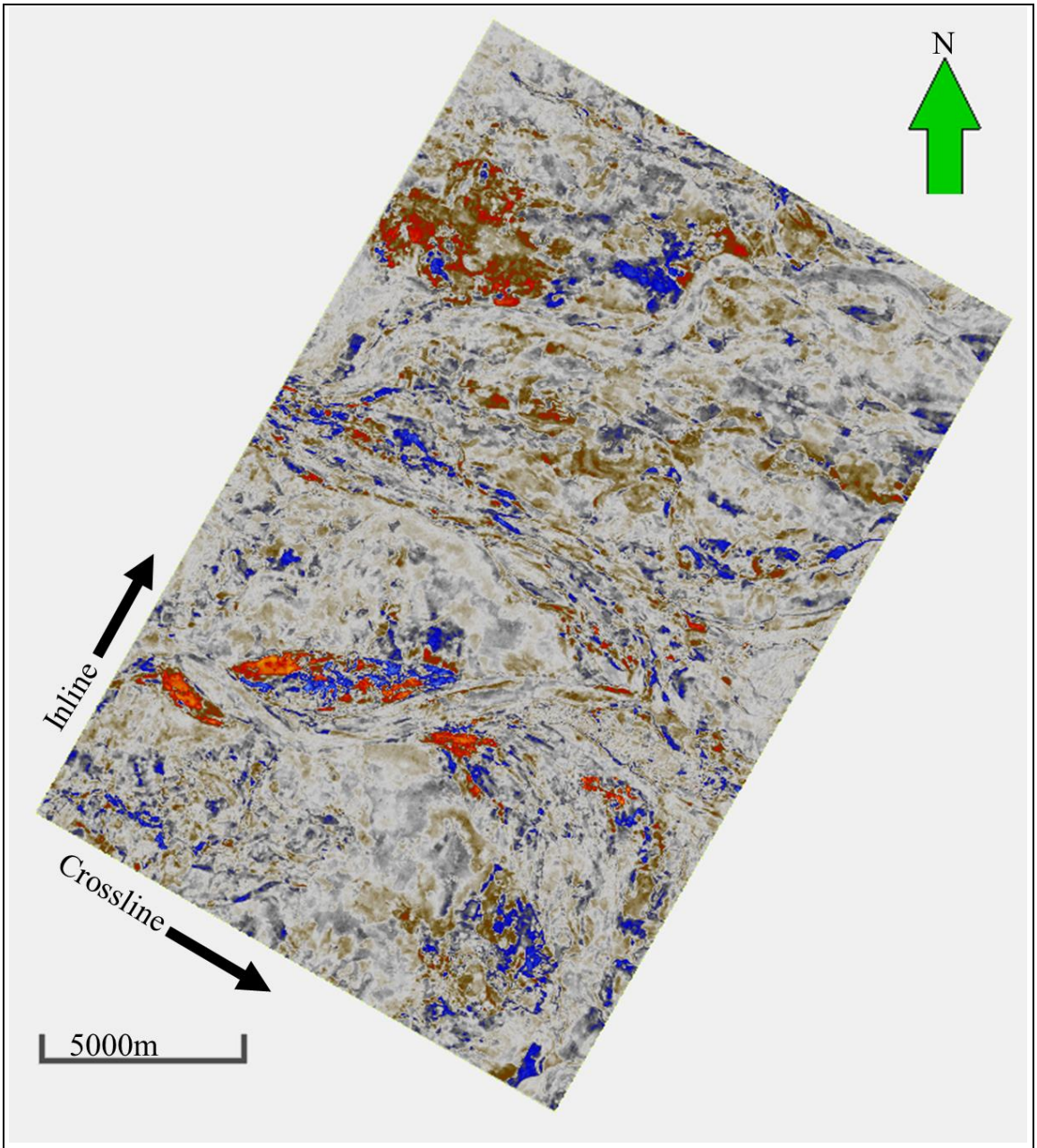

Figure 3.4. Horizon slice through the seismic amplitude volume.

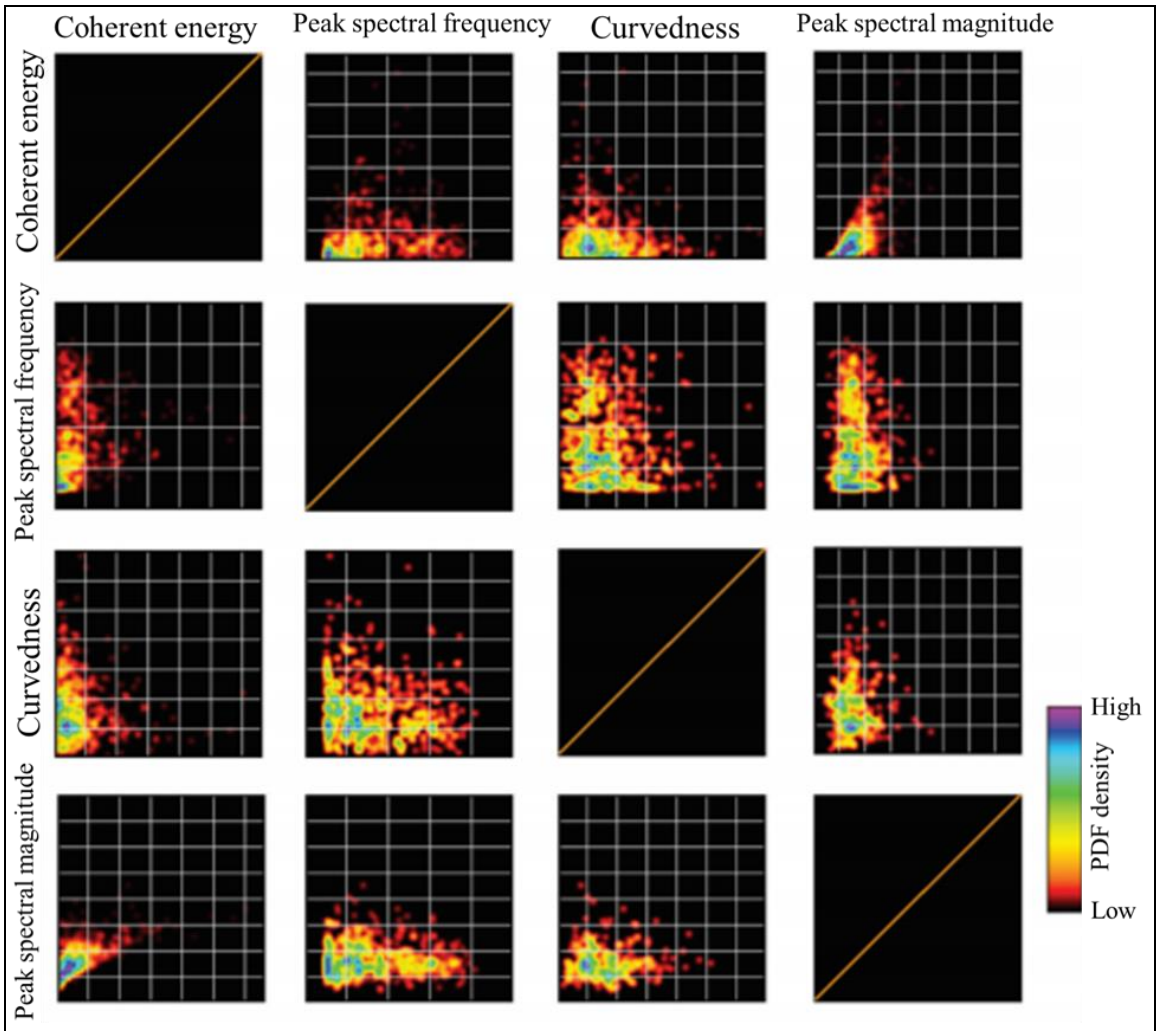

Figure 3.5. Crossplot of input attributes for SOM. The attributes used to construct the 2D SOM latent space. The cloud shapes indicate that correlation amongst the four attributes is low, indicating that they are mathematically independent and favorable attributes to use for machine learning. (Modified from Zhao et al., 2016)

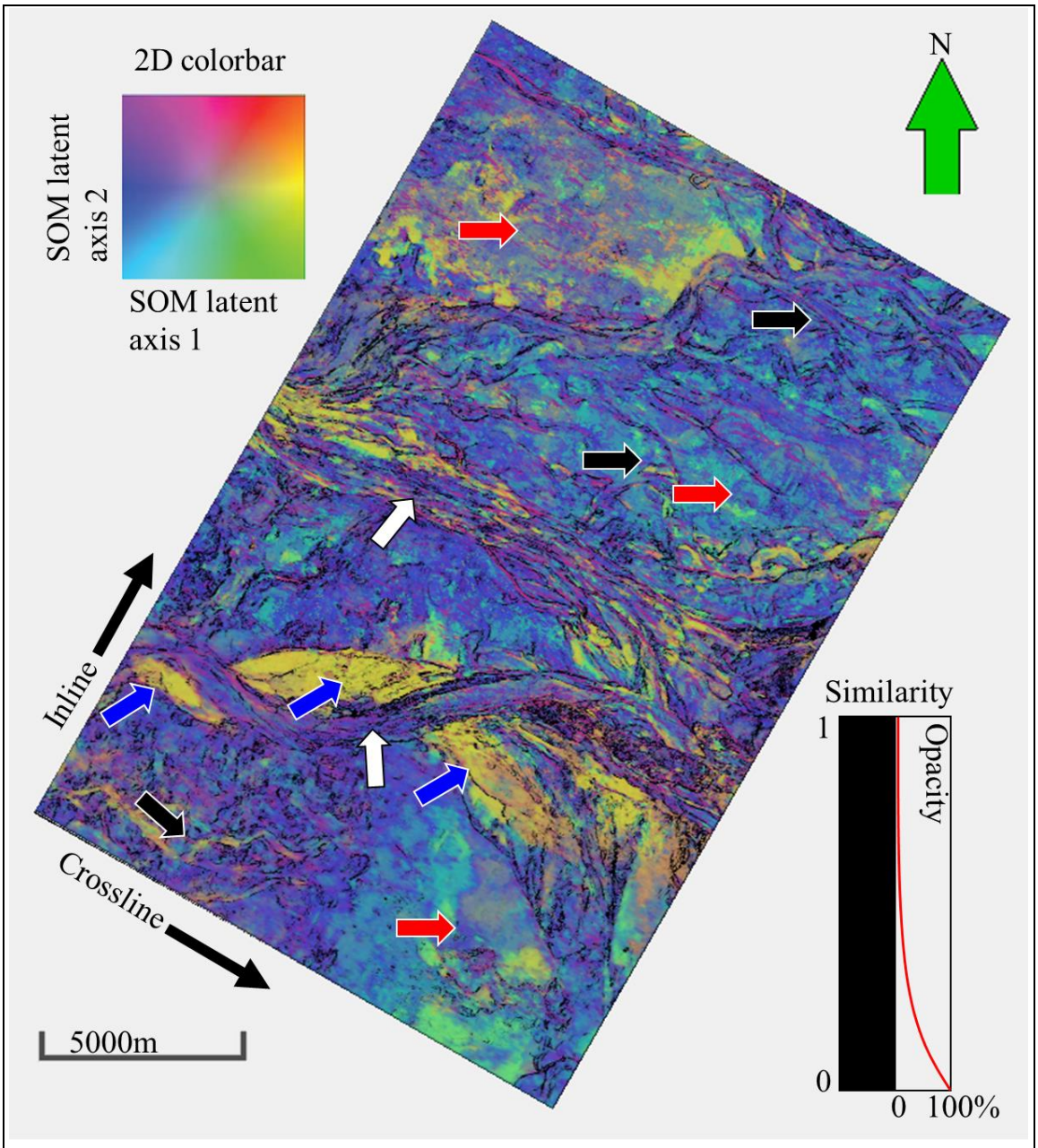

Figure 3.6. Previous work using Self-Organizing Maps. Red arrows indicate possible slope fans, blue arrows high amplitude deposits, white arrows stacked channels, and black arrows weak sinuous channels (Zhao et al. 2016).

3.2 *Methods*

I will use the two SOM latent axes produced by Zhao et al. (2016) as input to a GMM to further analyze the turbidite system with the goal of generating better defined (though still unsupervised) seismic facies. The two latent axes first undergo a Z-transform to condition the data making the population mean zero and the population covariance the identity matrix. A training data set is then extracted by sampling every 5^{th} time sample, and every 10^{th} inline and crossline from each SOM latent axis creating 52718 2-dimensional data vectors. The training data set is used to generate multiple GMM; the number of clusters range from 1-30, and all nine covariance parameterizations discussed in Chapter 2 are considered. The Bayesian Information Criterion is then calculated for each model so they can be quantitatively compared (Figure 3.7). For a visual comparison, GMMs with clusters 1-10 and an unconstrained covariance matrix are shown on the horizon (Figures 3.8-3.15). Generally, the BIC has a definitive maximum point with the addition of successive clusters decreasing the BIC and the model with the maximum BIC value is considered to be the best model. In this case however, the BIC seems to increase as the number of components increase. The increase may be due to cluster geometries in the latent space being poorly represented by a Gaussian distribution, thus creating the need for multiple Gaussian distributions to fit the cluster. Another explanation could be that given the large sample size of the training data, there may be a large number of small clusters that are statistically significant but not interpretationally significant. In consideration of this ambiguity, and because the BIC not increasing significantly by using additional components, I construct a mixture model with seven clusters and a covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$.

The means of the mixture model are well separated and the Gaussian ellipsoids exhibit minimal overlap (Figure 3.8). This separation is to be expected of the model-based classification approach as opposed to the mixture approach where data vectors don't necessarily belong to a single Gaussian distribution (Celeux and Govaert, 1993). The selected mixture model is applied to the entirety of both SOM latent axes to calculate the posterior probability of each data vector belonging to each cluster. The data vectors are classified according to the highest posterior probability and an estimate of the confidence in the classification is calculated resulting in a seismic attribute volume containing the class assignments, and two additional seismic attribute volume containing the uncertainty and ambiguity of the classification.

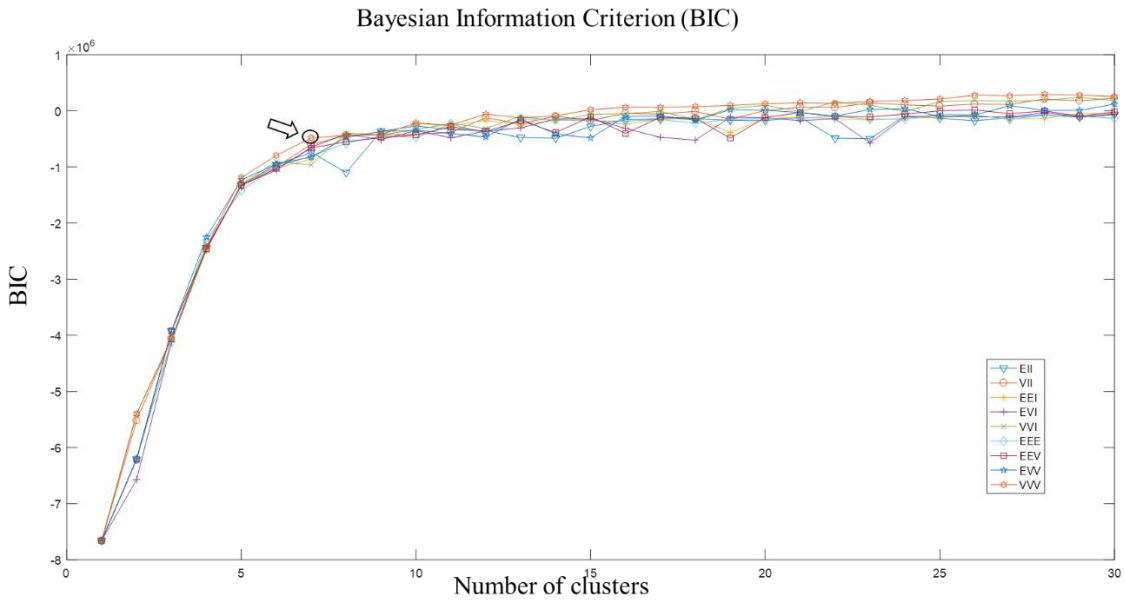

Figure 3.7. Bayesian Information Criterion. The module symbols, EII, VII, EEI, EVI, etc…, represent the different parametrizations of the covariance matrix. Since the BIC does not decrease with the number of clusters, the model on the shoulder of the curve is picked.

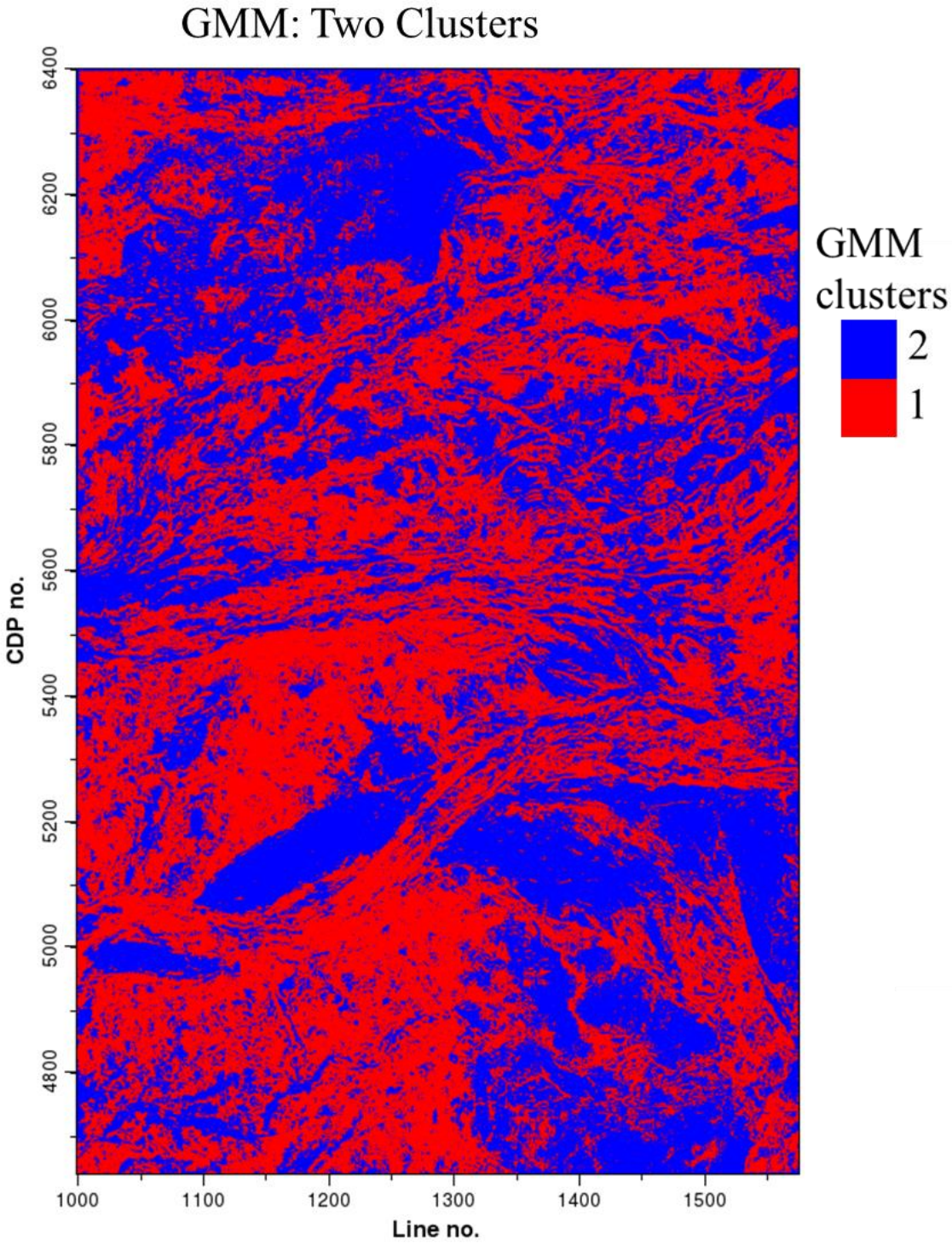

Figure 3.8. Classification of turbidite system using two GMM clusters. Cluster colors are arbitrary.

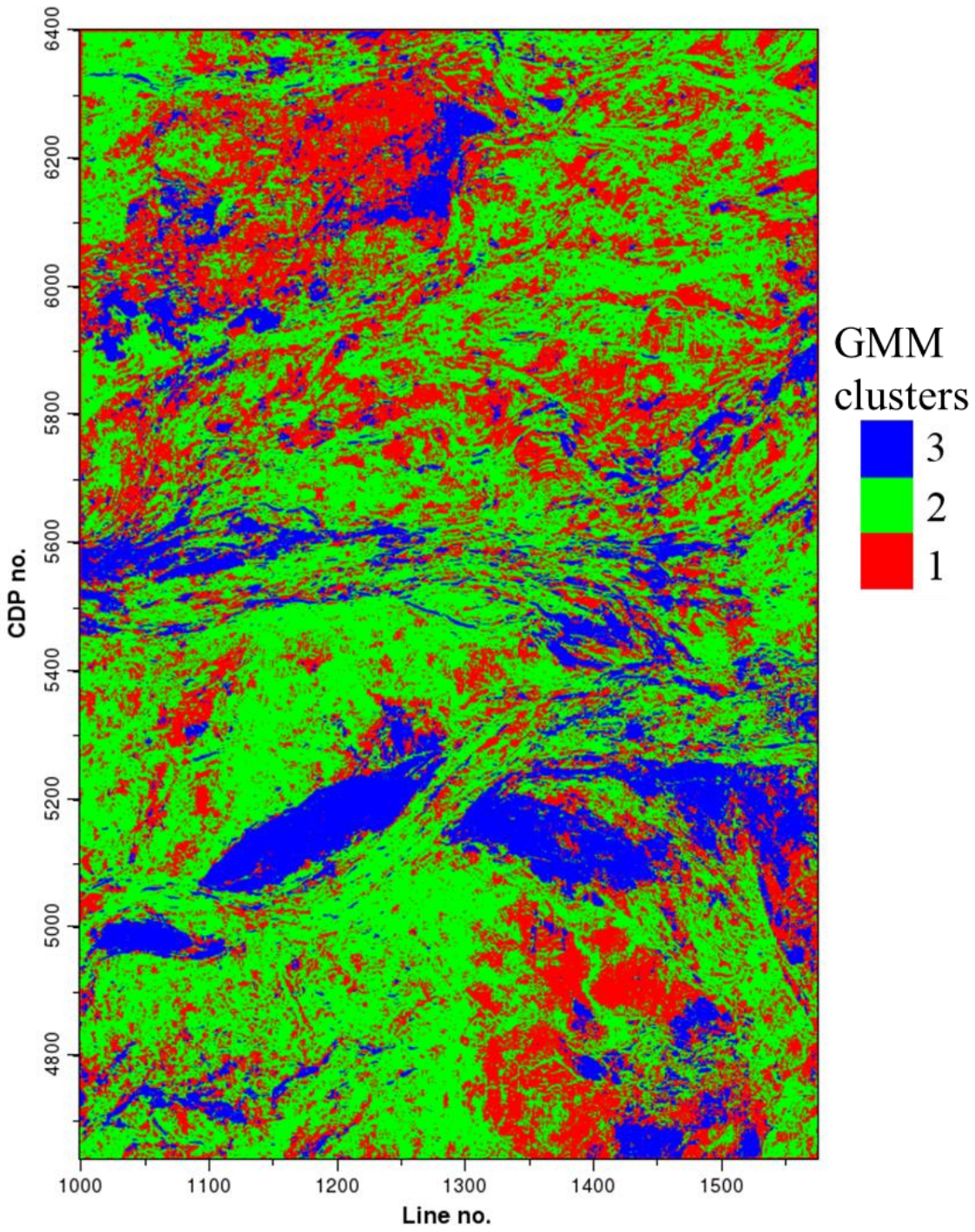

Figure 3.9. Classification of turbidite system using three GMM clusters. Cluster colors are arbitrary.

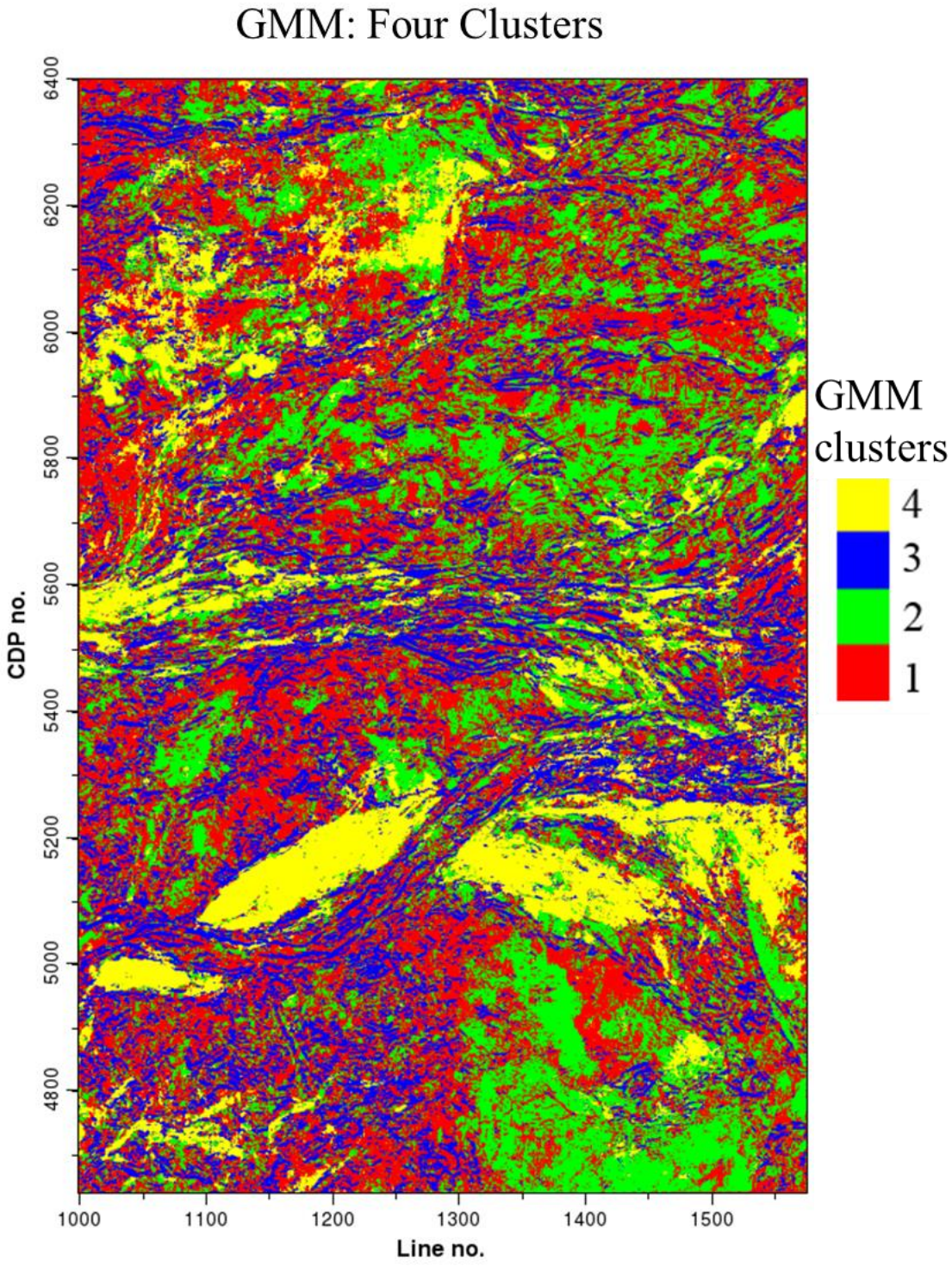

Figure 3.10. Classification of turbidite system using four GMM clusters. Cluster colors are arbitrary.

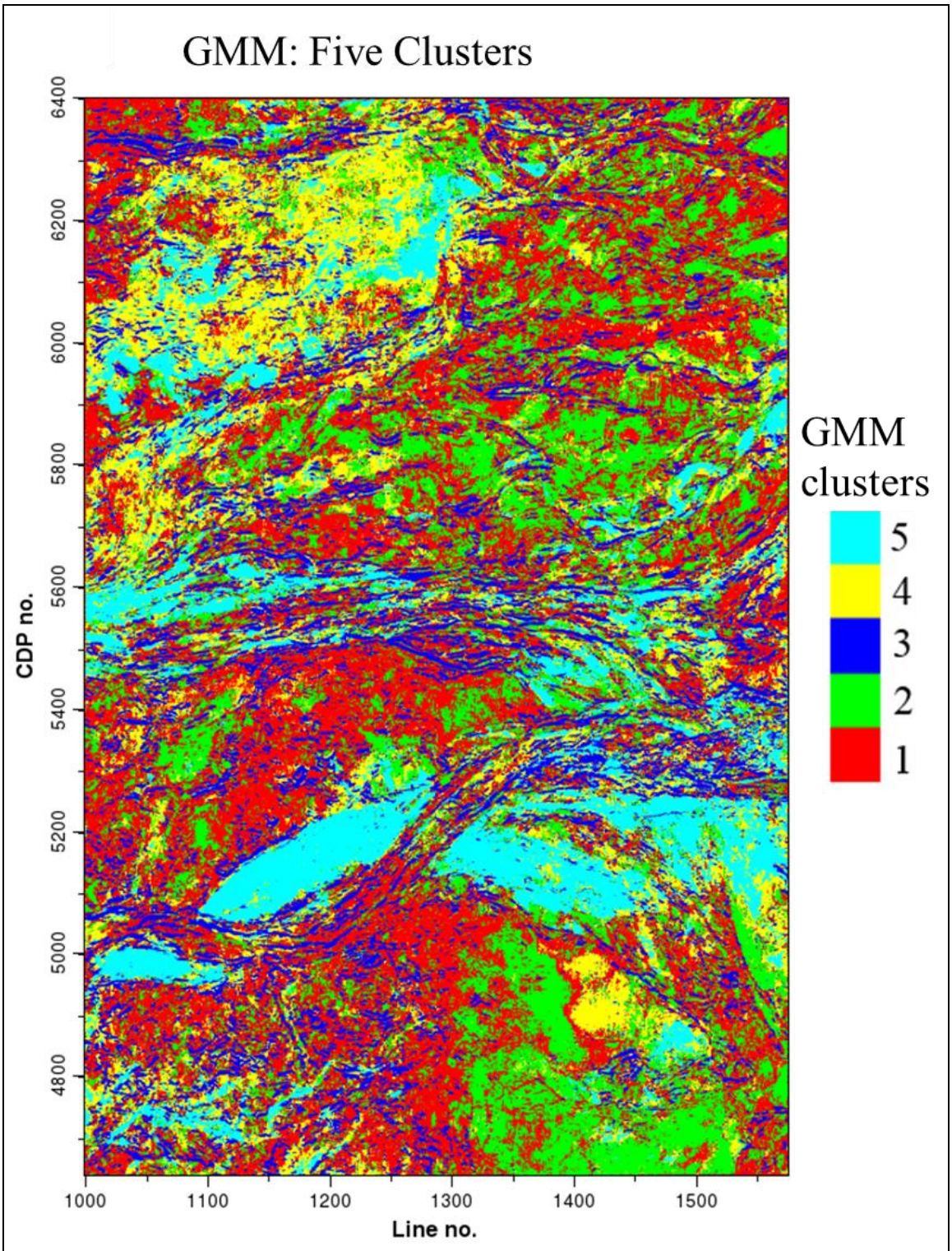

Figure 3.11. Classification of turbidite system using five GMM clusters. Cluster colors are arbitrary.

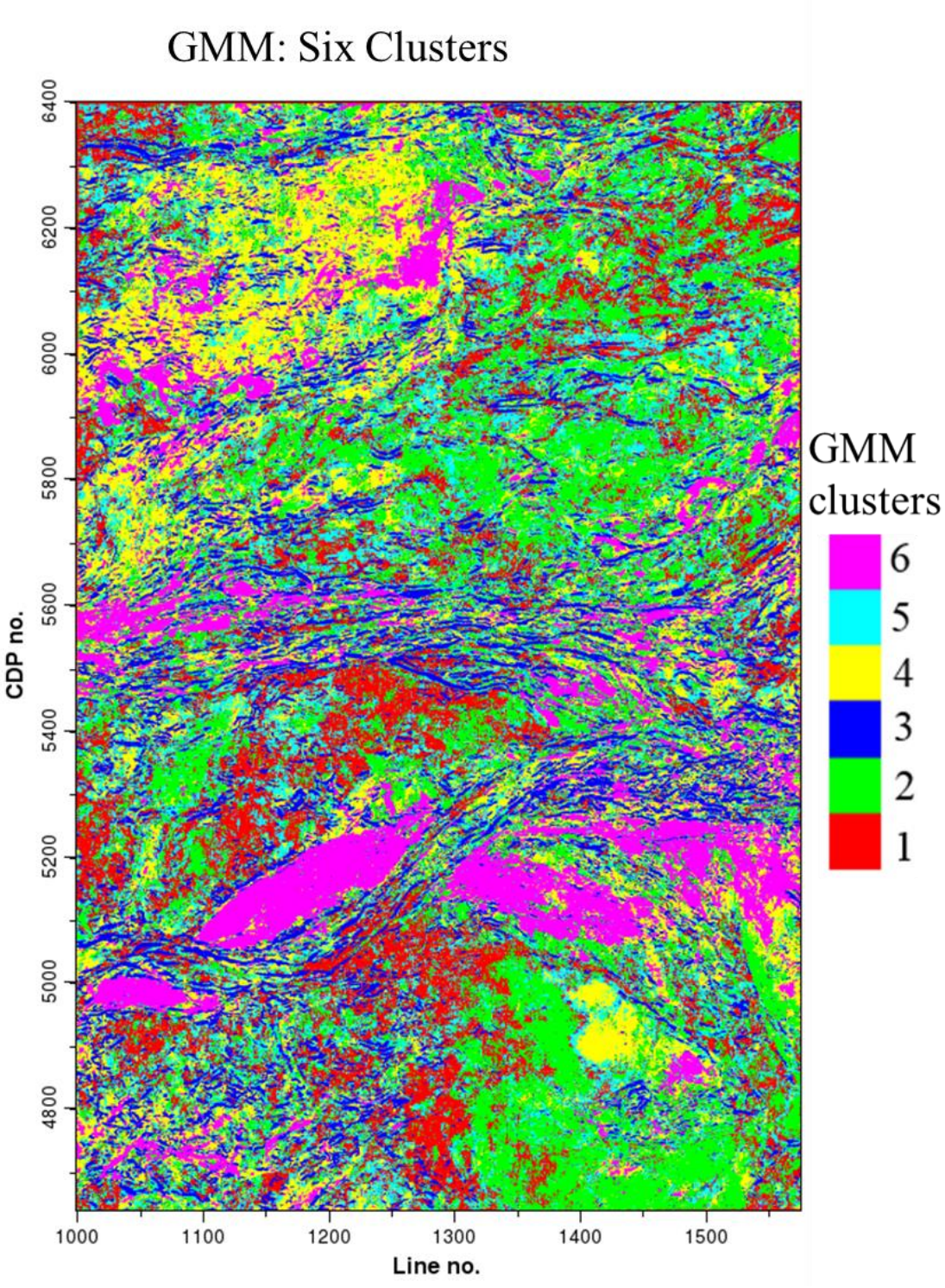

Figure 3.12. Classification of turbidite system using six GMM clusters. Cluster colors are arbitrary.

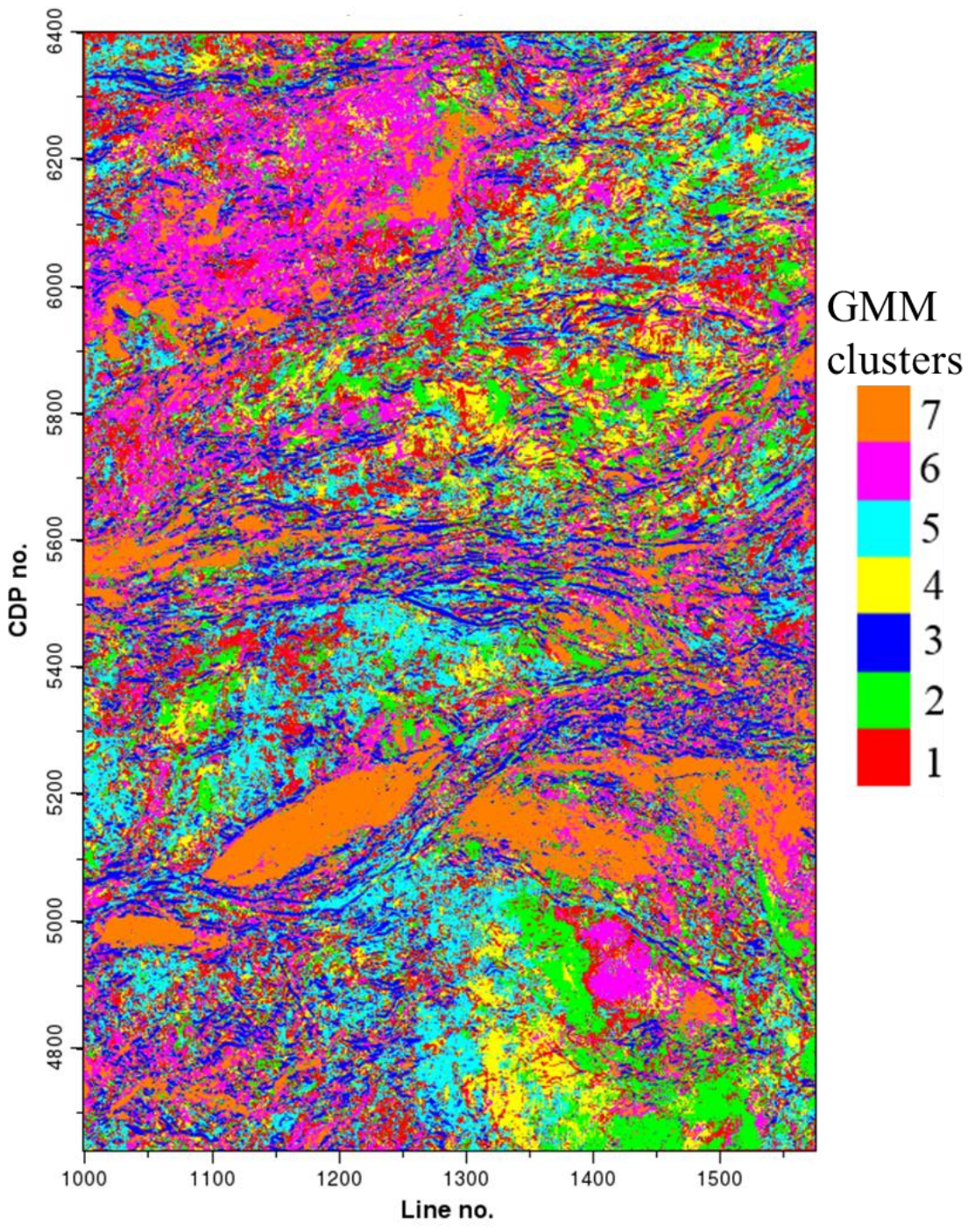

Figure 3.13. Classification of turbidite system using seven GMM clusters. Cluster colors are arbitrary.

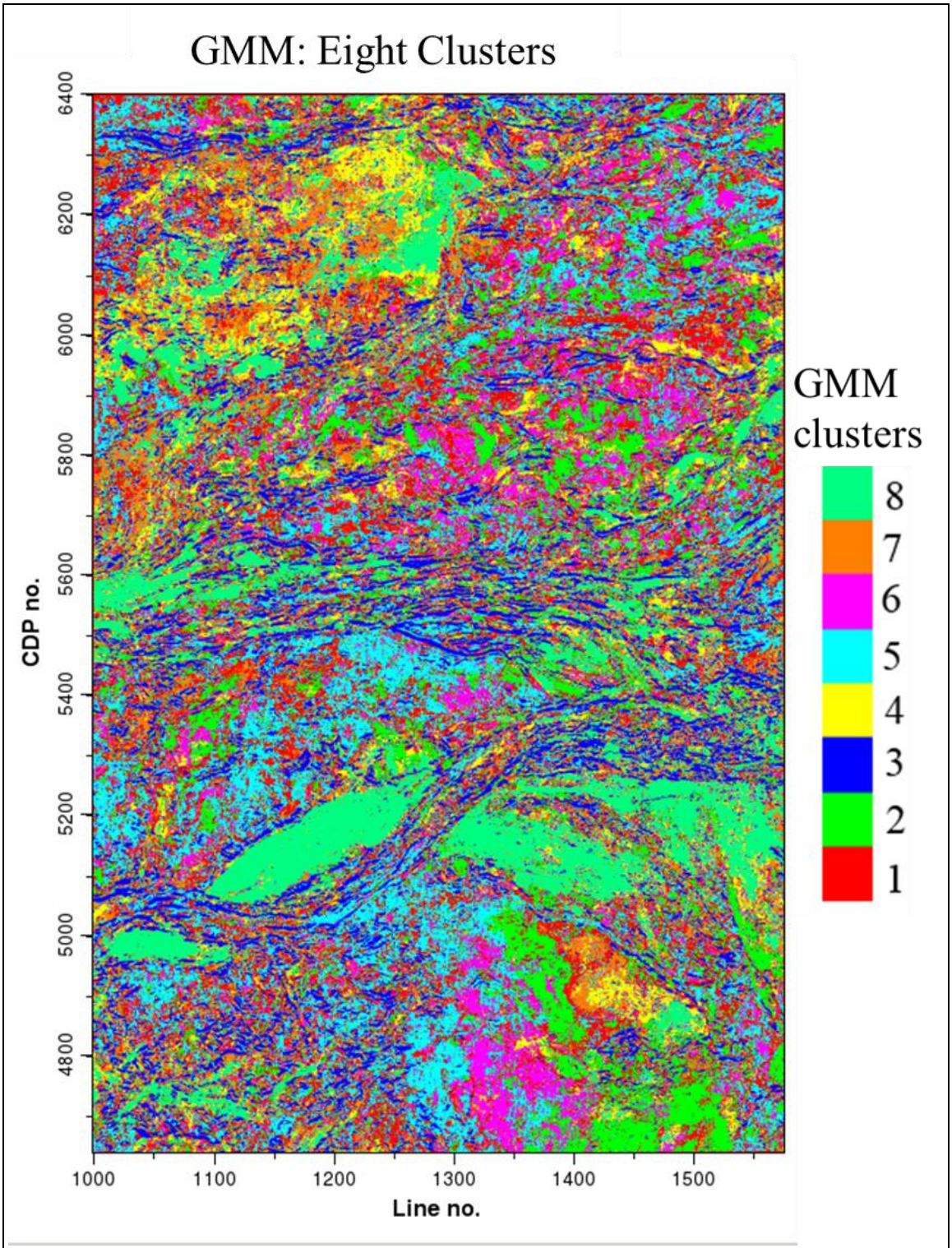

Figure 3.14. Classification of turbidite system using eight GMM clusters. Cluster colors are arbitrary.

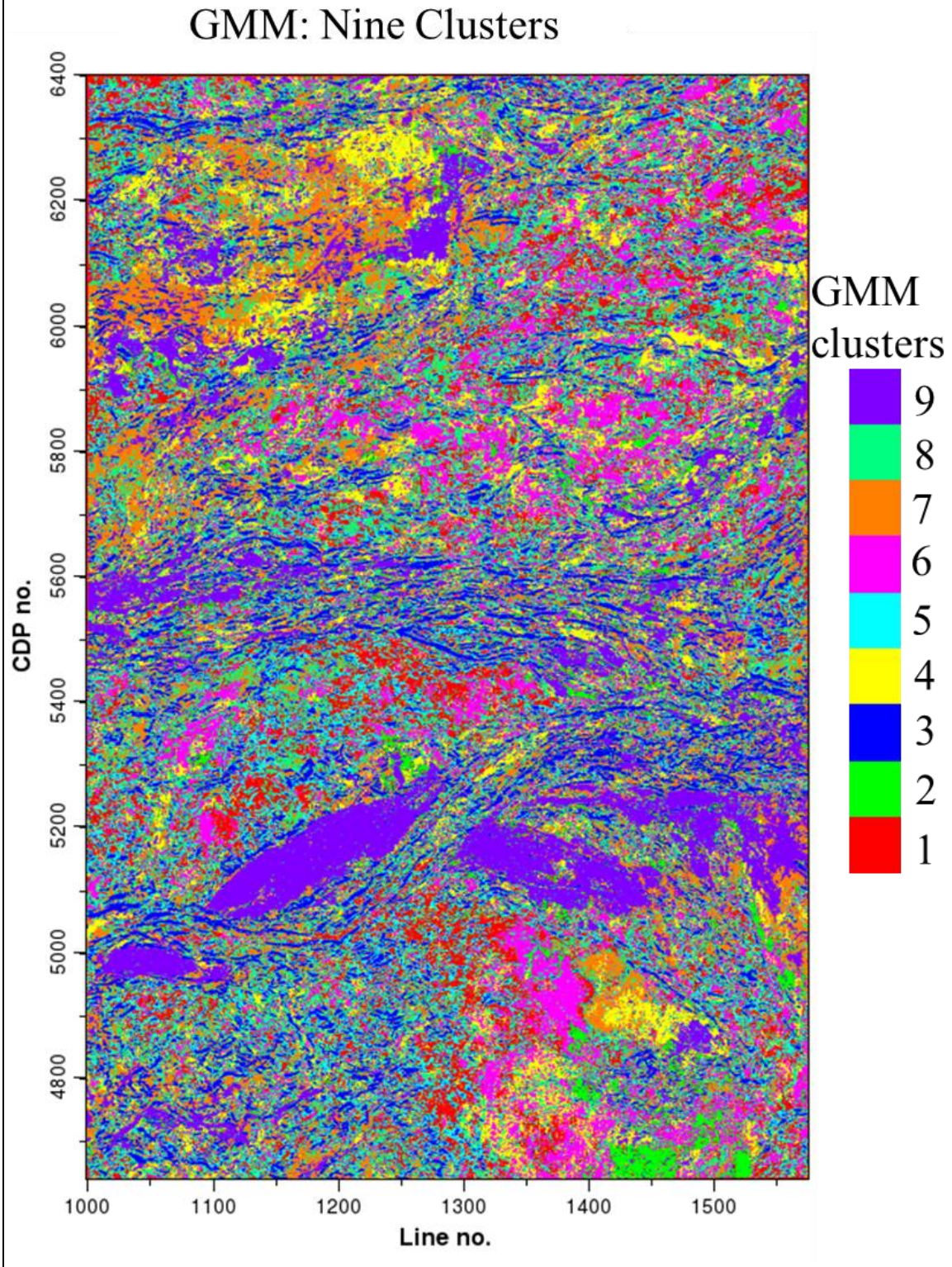

Figure 3.15. Classification of turbidite system using nine GMM clusters. Cluster colors are arbitrary.

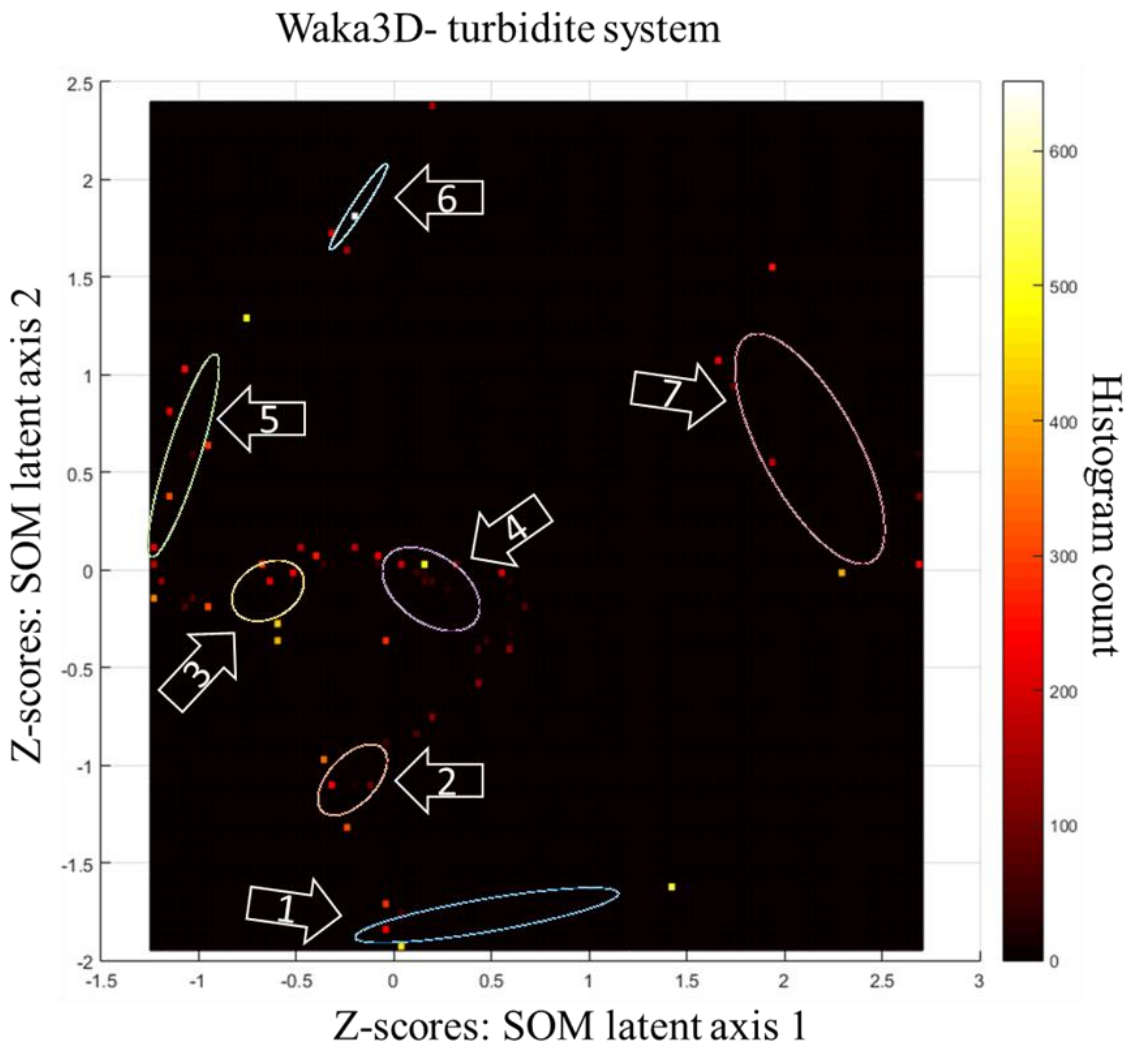

Figure 3.16. Histogram of training data with a new GMM.Gaussian ellipsoids represent a one standard deviation contour of the Gaussian distribution and don't represent the mixing weights of the mixture model. The covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$ allows for the Gaussian ellipsoids to vary in terms of size, shape, and orientation. The colors of the ellipsoids here are arbitrary and are not representative of any inter-cluster relationship.

Table 3.1. Mixture model parameters for the turbidite system. Selected model is built using seven clusters and a covariance parameterization of the form $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$.			
Cluster	π (%)	$\mu\begin{pmatrix}\text{SOM latent axis 1}\\ \text{SOM latent axis 2}\end{pmatrix}$	$\mathbf{C}$
1	11.5%	$\begin{matrix}0.489\\ -1.77\end{matrix}$	$\begin{matrix}0.458 & 0.075\\ 0.075 & 0.019\end{matrix}$
2	9.5%	$\begin{matrix}-0.201\\ -1.08\end{matrix}$	$\begin{matrix}0.031 & 0.017\\ 0.017 & 0.032\end{matrix}$
3	19.5%	$\begin{matrix}-0.636\\ -0.117\end{matrix}$	$\begin{matrix}0.034 & 0.008\\ 0.008 & 0.024\end{matrix}$
4	19.7%	$\begin{matrix}0.203\\ -0.108\end{matrix}$	$\begin{matrix}0.062 & -0.022\\ -0.022 & 0.045\end{matrix}$
5	18.8%	$\begin{matrix}-1.07\\ 0.572\end{matrix}$	$\begin{matrix}0.034 & 0.083\\ 0.083 & 0.268\end{matrix}$
6	9.14%	$\begin{matrix}-0.17\\ 1.84\end{matrix}$	$\begin{matrix}0.022 & 0.031\\ 0.031 & 0.048\end{matrix}$
7	12%	$\begin{matrix}2.15\\ 0.607\end{matrix}$	$\begin{matrix}0.148 & -0.158\\ -0.158 & 0.344\end{matrix}$

3.3 *Results and Interpretation*

In the GMM, the cluster numbers are arbitrary and do not reflect any sort of measurement between each other which can make the choice of color bar difficult. Since the classification is an integer number, a discrete color bar should be used such that a single color represents each cluster. Zhao et al. (2016) used a 2D color bar to represent two SOM latent axes. The GMM was generated using training vectors from the Z-scores of the two SOM latent axes. For the sake of comparison, each cluster is assigned a color close to where the clusters' means would be on the 2D color bar from Zhao et al. (2016) as if the color bar was rescaled to fit the range of Z-scores (Figure 3.17). Part of the

reason the SOM latent axes look smooth is due to the interpolation scheme used by visualization software package. In the GMM classification, interpolation is not used because there is no relationship between the clusters and the cluster number, meaning the classification map of the GMM will not be inherently smooth.

The continuous 2D SOM latent space is reduced to 7 different clusters each of which have a different color. The choice of color bar makes it difficult to differentiate some of the facies (Figure 3.18). Co-rendering the ambiguity with the classification itself mutes areas where the model isn't confident in the classification (Figure 3.19). The horizon slice is segmented into just seven different clusters.

Although each cluster may have a significant amount of the mixture model weight, the clusters may be lacking in geological or geophysical significance. Furthermore, since the construction of the mixture model did not contain any spatial information, the clusters are not expected to be spatially continuous. Specifically, the 2nd, 3rd, and 5th clusters are spatially discontinuous making it hard to correlate them with any architectural elements or geologic features. The 3rd cluster is widespread and can be found in almost every architectural element on the horizon. The 2nd cluster and 5th cluster seem to be more restricted in their distribution, but don't seem to have any obvious structure. The 1st, 4th, 6th, and 7th clusters contain at least some areas with a continuous distribution that may be part of a channel or sand lobe.

The 6th cluster has the smallest amount of weight in the mixture model, but shows clear lineaments along the sinuous channels. The 4th cluster has the largest amount of weight in the mixture model, and is relatively continuous and corresponds well with previously interpreted slope fan and lobe deposits previously mapped by Zhao et al.

(2016) in the northern and southern parts of the horizon. The 1st cluster contains facies around previously interpreted slope fan and lobe deposits in the southern corner and northeastern part of the horizon. The 7th cluster contains facies that correspond mostly to features interpreted by Zhao et al. (2016) as sandy channels.

Closely inspecting the ambiguity, the 3rd cluster seems to contain the most ambiguous facies (Figure 3.19). The 3rd cluster is in close proximity to the 4th and 5th clusters in the latent space which may be causing the GMM to be ambiguous when representing multiple data vectors that fall between them. Furthermore, the 2nd, 3rd, and 5th clusters are observed to be spatially discontinuous which may merit interpreting the three clusters as being a larger, single cluster.

The Mahalanobis distance is used a measure the GMM's uncertainty in describing a data vector (Figure 3.20). The Mahalanobis distance is calculated for each point with respect to the GMM component that the data vector got classified to. Areas with a large Mahalanobis distance can be considered anomalous because they aren't close to the GMM component that it belongs to. Co-rendering with the classification shows two anomalous areas of the 4th cluster that indicate possible slope fans. Highly ambiguous and uncertain areas are interpreted as meaningless anomalies.

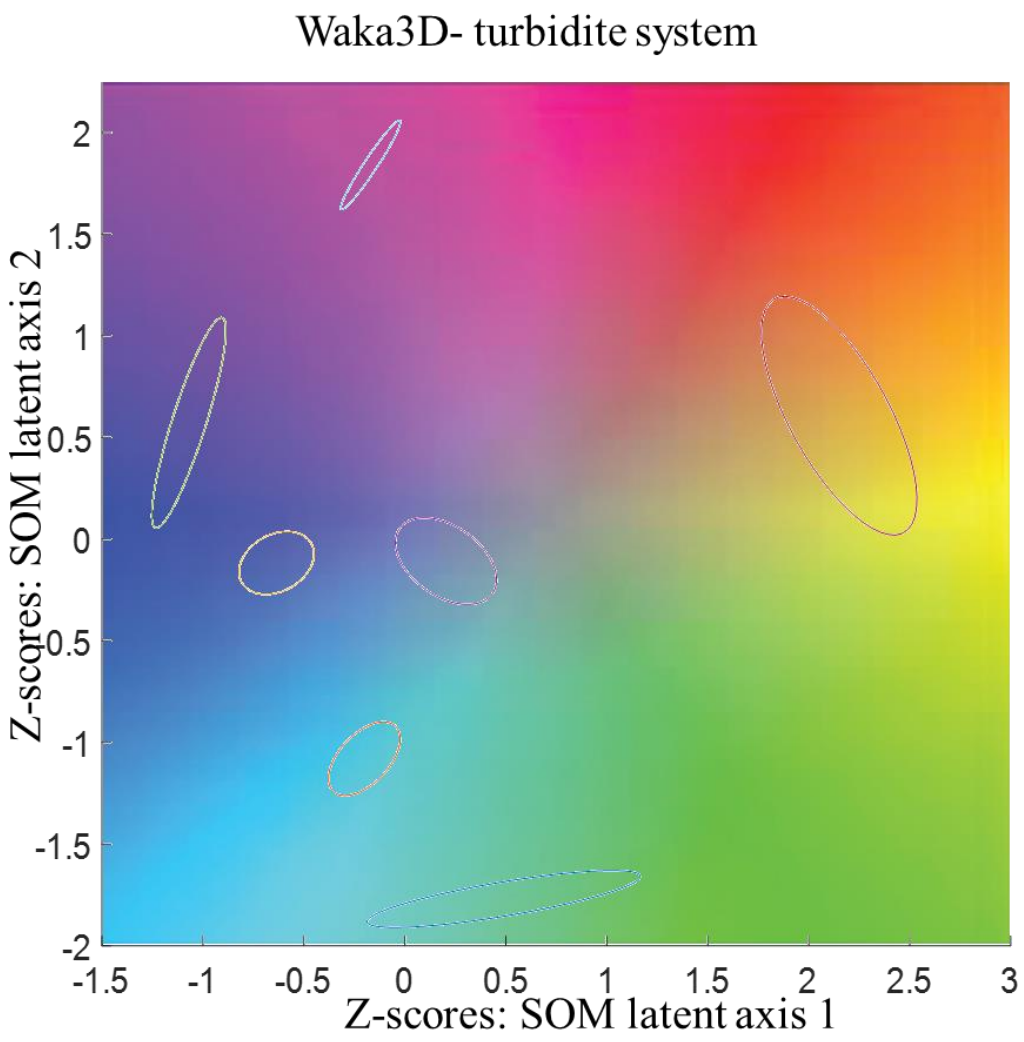

Figure 3.17. Gaussian ellipsoids with 2D SOM color bar in background. The means of each cluster are in the center of each Gaussian ellipsoid and each cluster is assigned a color where its mean lies on the 2D color bar.

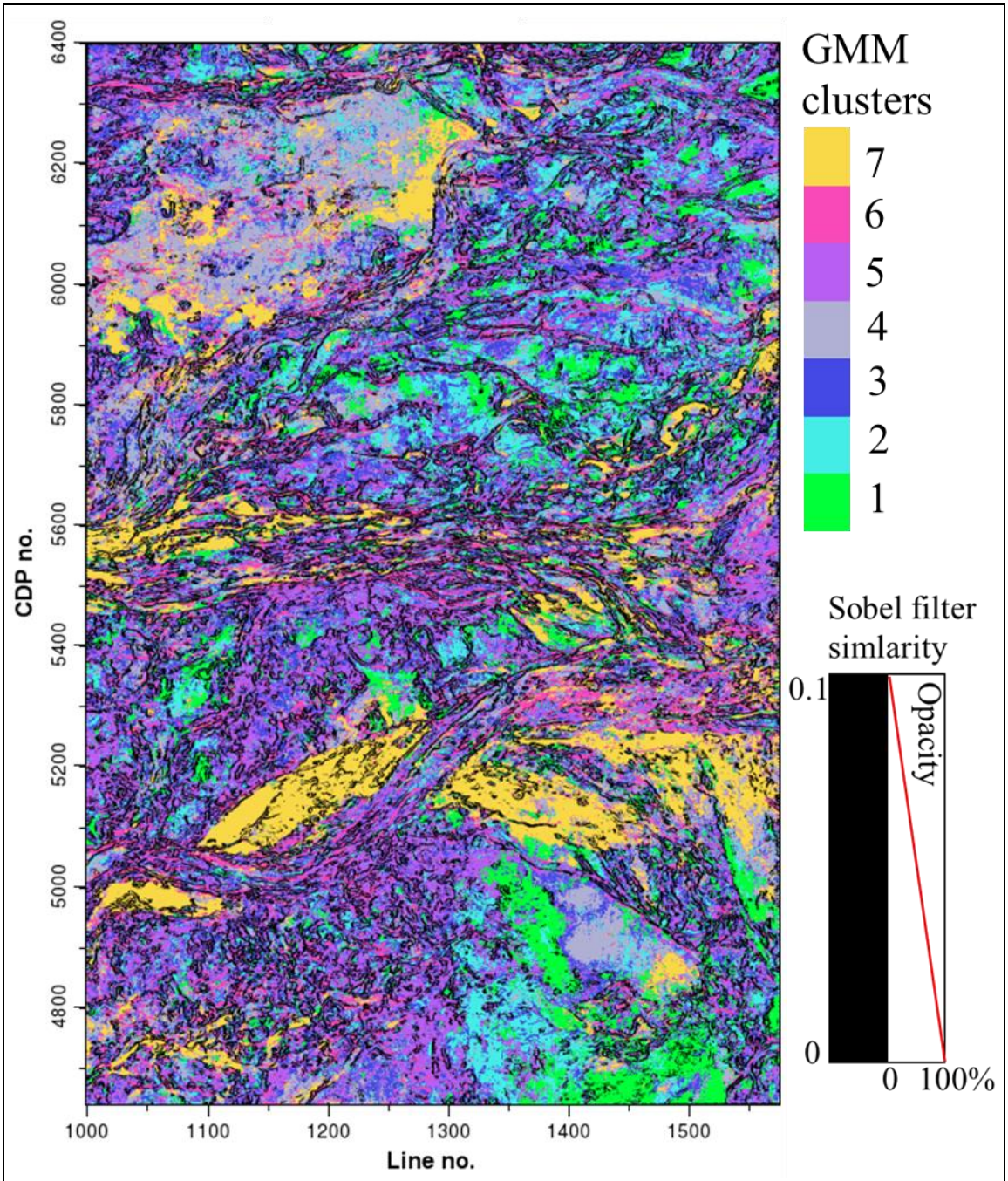

Figure 3.18. Classification of turbidite system using GMM. Each cluster is assigned a color similar to where the mean of each cluster would lie on the 2D color bar of the 2D SOM latent space from Figure 3.9. Similarity is overlaid in black to show boundaries.

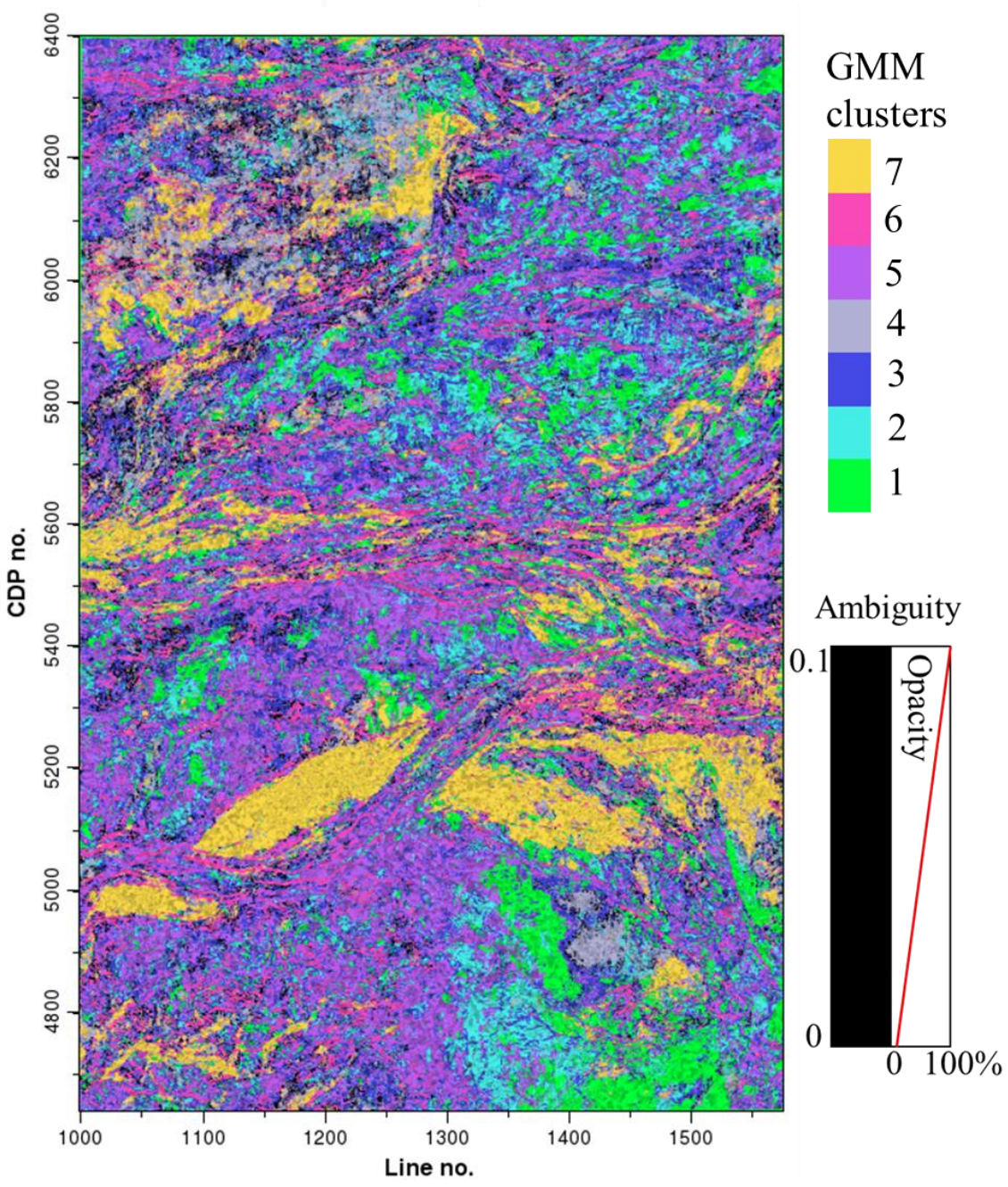

Figure 3.19. Ambiguity of turbidite system. The classification is overlaid by the ambiguity to hide areas where the mixture model is not so confident in the classification. The ambiguity in the classification along the horizon seems to mostly correspond with the 2nd cluster. The ambiguity has a range of [0,1-1/*K*] where *K* is the number of clusters; in this case [0,0.857].

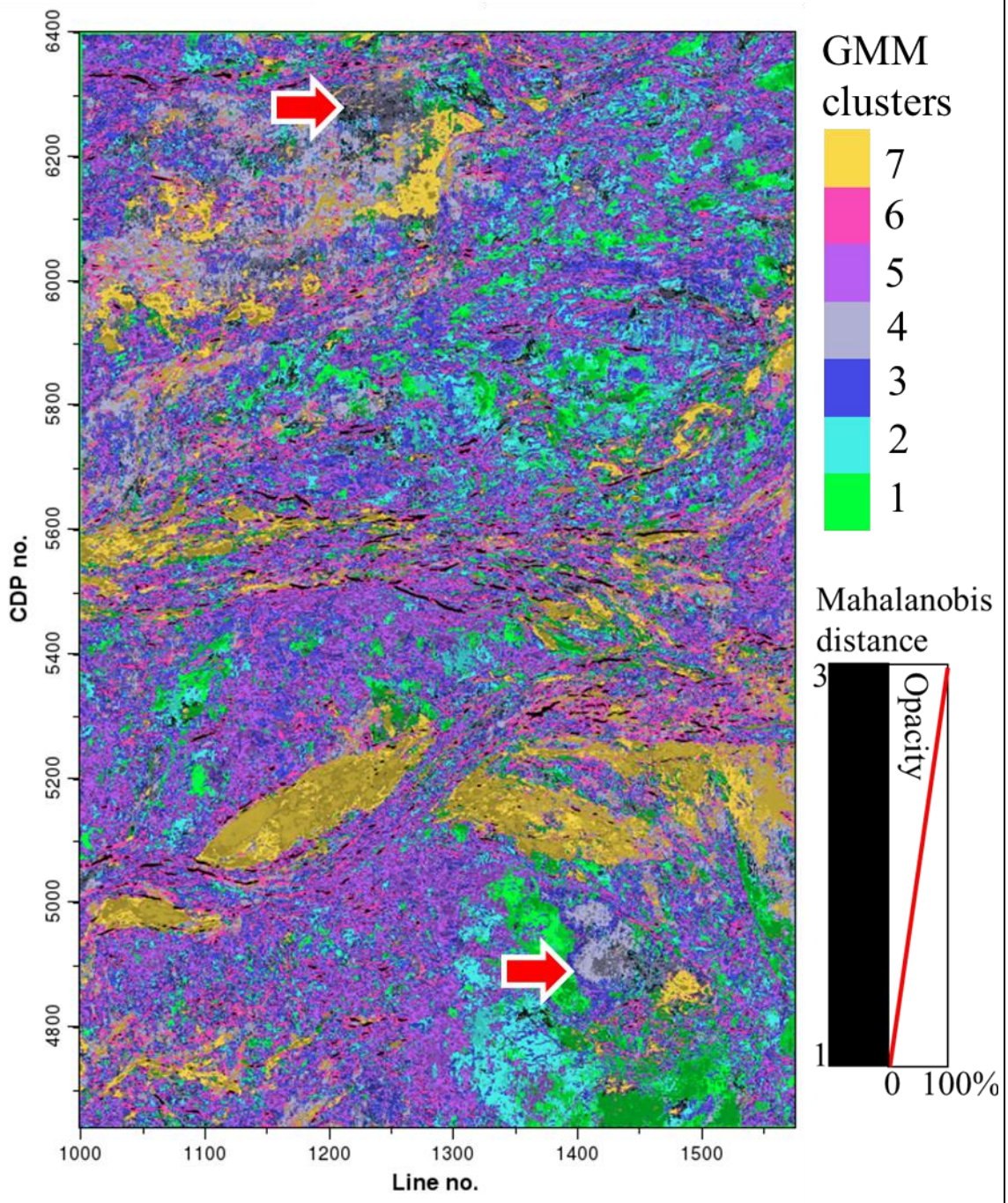

Figure 3.20. GMM classification with uncertainty. Classification co-rendered with the Mahalanobis distance as a measure of uncertainty.

Table 4.2. Interpretation of clusters	
Cluster	**Interpretation**
1	Continuous; associated with slope fan and lobe deposits
2	Discontinuous; some constraint in distribution
3	Discontinuous; widespread distribution
4	Continuous; possibly slope fan and lobe deposits
5	Discontinuous; some constraint in distribution
6	Clear lineaments along channels
7	Continuous; constrained to channels, slope fan and lobe deposits; sandy

3.4 *Spatial considerations*

Combining Self-Organizing maps with GMM facilitates a more quantitative approach to determining the number of clusters as opposed to a qualitative visual inspection. However, the classification isn't guaranteed to be spatially continuous because there is no spatial information in the input attributes nor in the mixture model. In the next section, a special variant of the expectation-maximization (EM) algorithm called the *neighborhood* EM algorithm is implemented to consider spatial correlations using a small subset of the horizon to highlight a channel feature (Figure 3.21). Model selection is not being considered with model parameters being set to four clusters (called components in the next section) with an unconstrained covariance matrix, $\mathbf{C}_k = \alpha_k \mathbf{D}_k \mathbf{\Lambda}_k \mathbf{D}_k^{\mathrm{T}}$.

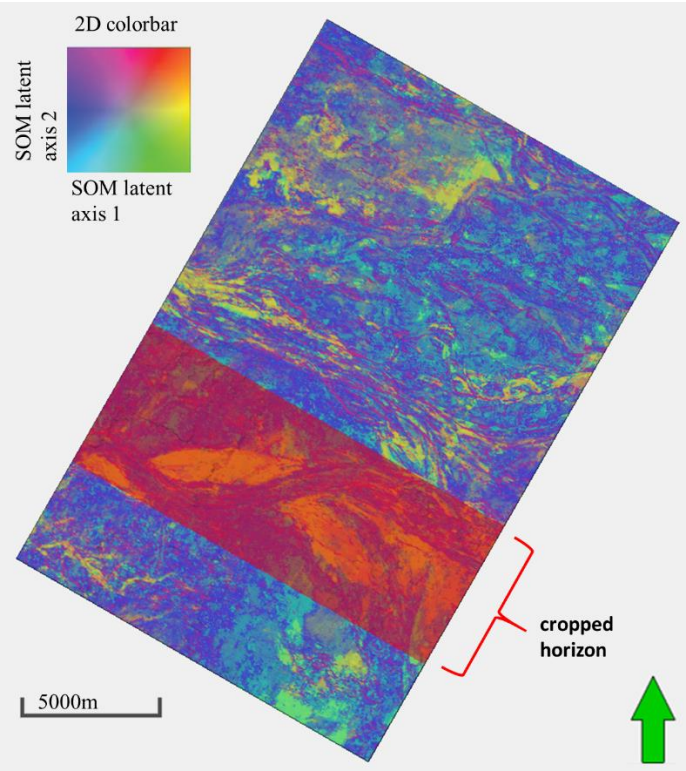

Figure 3.21. Cropped horizon to show channel feature. A subset of the SOM latent space axes is cropped to highlight a channel feature. The large memory requirements of the neighborhood expectation-maximization algorithm require a smaller subset of the data to be used.

3.5 Neighborhood expectation-maximization (NEM) algorithm and application

(The following is from an accepted SEG abstract by Hardisty and Wallet 2017).

Learning of a GMM using the EM algorithm is a purely statistical construct and doesn't consider spatial correlations. In general, facies are expected to be at least laterally continuous to some extent. To account for spatial correlations of the latent space the Neighborhood-EM (NEM) algorithm is implemented and compared to the results of the conventional EM. A more detailed discussion of the NEM algorithm can be found in Appendix A.

The area of interest has been interpreted as a possible channel feature by Zhao et al. (2016). The area of interest consists of 456 crosslines x 576 inlines x 23 time samples.

The SOM latent axis 1 and SOM latent axis 2 are used as inputs for two different GMM; one mixture model using the conventional EM algorithm and another using the NEM algorithm. A training set is constructed by uniformly sampling every 125th voxel (one voxel for every 5th inline, crossline, and time sample). For the NEM algorithm, the spatial weight parameter is set to 0.1 and the neighborhood window for which spatial correlations are considered is a 3-dimensional window of length 15 inlines x 15 crosslines x15 time samples.

Two cross sections are made, A-A' and B-B', to show the channel feature in three dimensions (Figure 3.22). Previously this was interpreted by Zhao et al. (2016) as a possible muddy channel cutting through a sandy channel. In both the EM and NEM case the sandy channel is dominated by the 4th component of the mixture model and is colored tan. Likewise, the muddy channel is dominated by the 2nd and 3rd components of the mixture model, and are colored red and green respectively. The NEM algorithm successfully segments the image into more spatially continuous facies. However, there are unnatural hard right angles in the classification due to the sub-sampling of the training set of data.

Cross section A-A' shows the high amplitude channel being delineated by the tan colored facies and being surrounded by the blue colored facies (Figure 3.23). The NEM algorithm improves the segmentation by removing the anomalous red facies above the high amplitude feature. In both EM and NEM the red and green facies are not within the high amplitude feature.

Cross section B-B' goes more or less along the flow direction of the tan colored channel (Figure 3.24). The combination of red and green colored facies provide good

channel segmentation. The NEM algorithm removes many of the red-colored facies in in the high amplitude areas and replaces them with tan-colored facies.

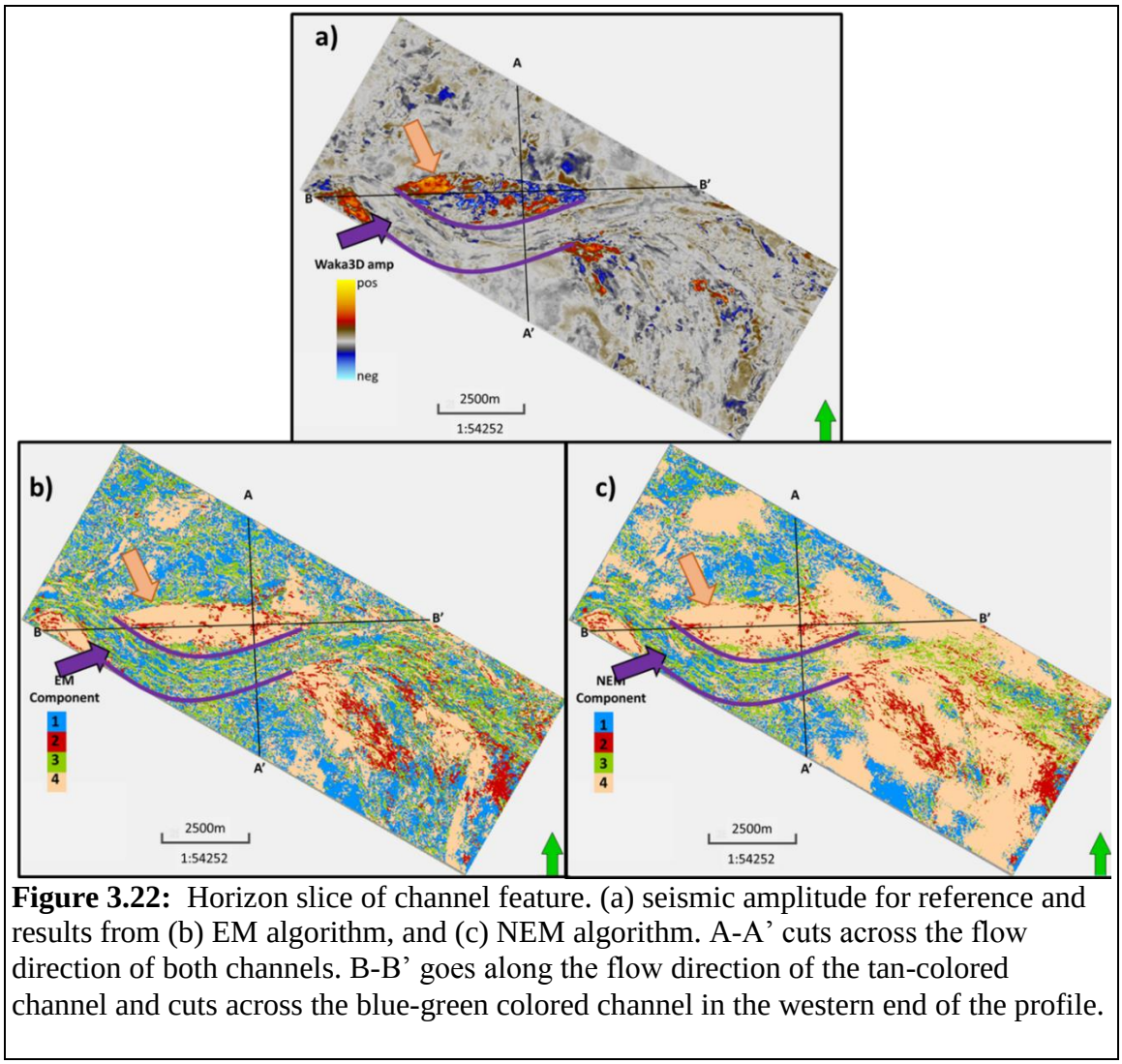

Figure 3.22: Horizon slice of channel feature. (a) seismic amplitude for reference and results from (b) EM algorithm, and (c) NEM algorithm. A-A' cuts across the flow direction of both channels. B-B' goes along the flow direction of the tan-colored channel and cuts across the blue-green colored channel in the western end of the profile.

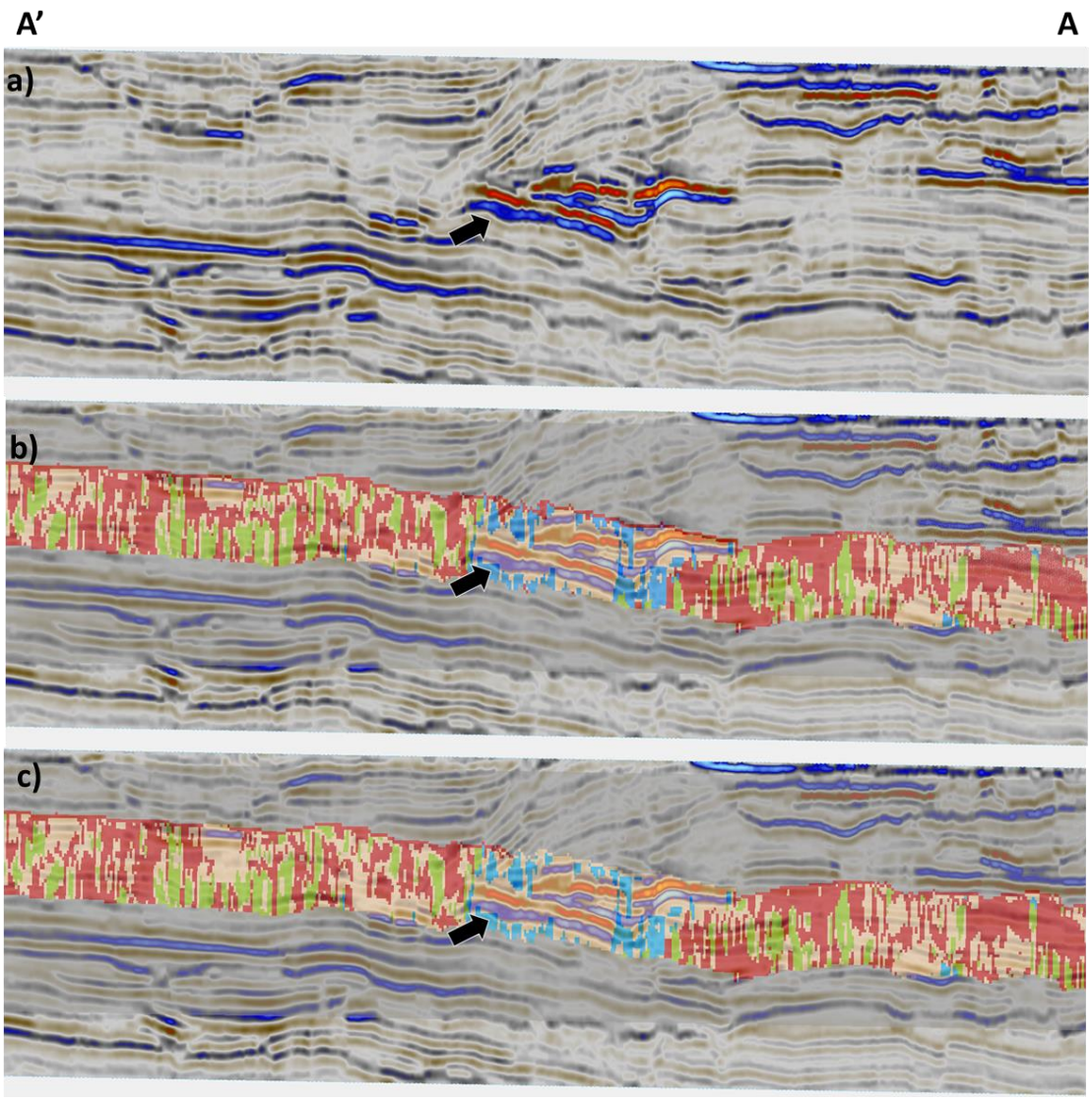

Figure 3.23: Profile A-A' of (a) seismic amplitude, (b) EM algorithm, and (c) NEM algorithm. Cuts perpindicular to the flow direction of tan colored channel. The black arrow indicates a high amplitude feature.

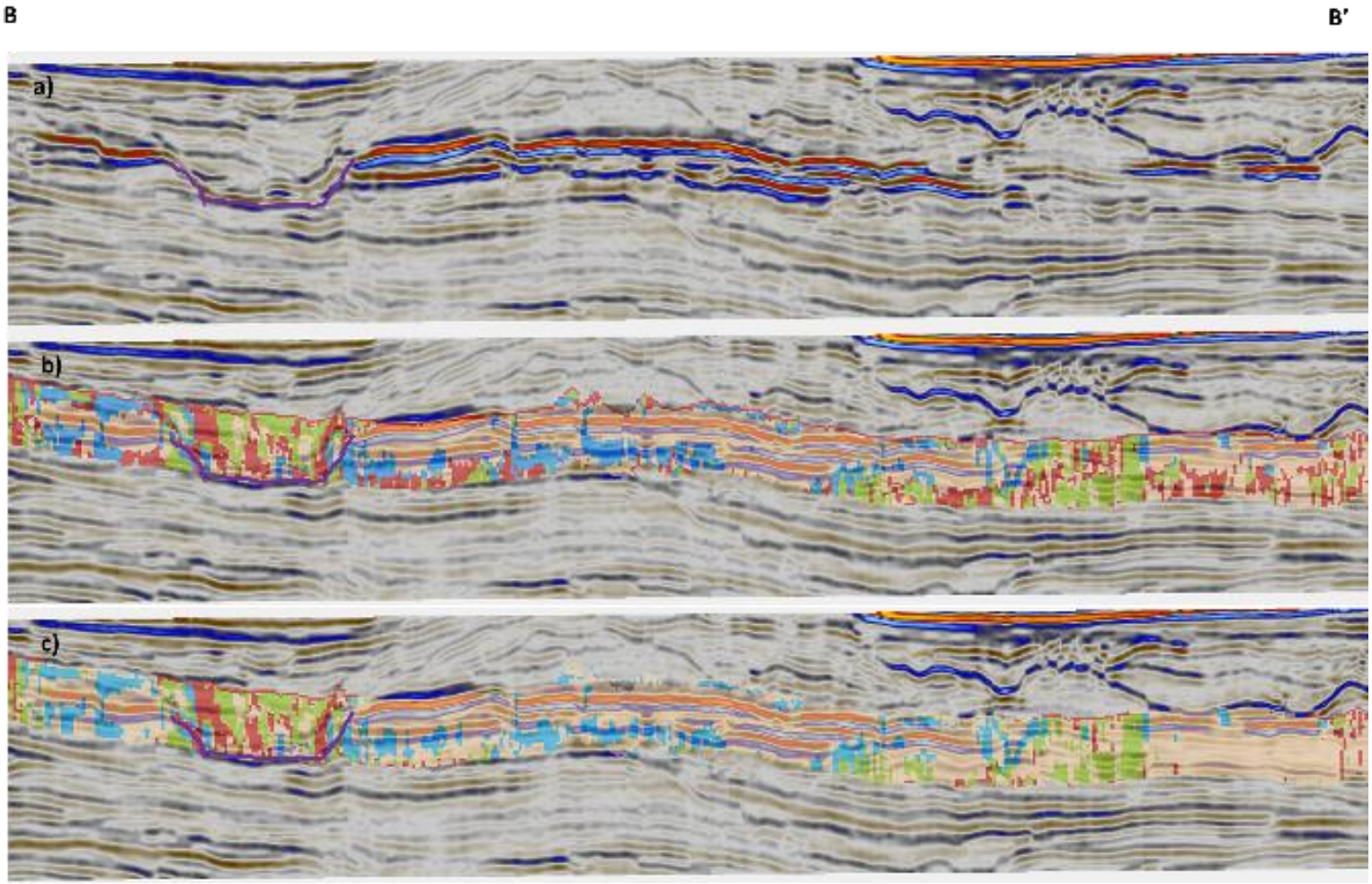

Figure 3.24: Profile B-B' of (a) seismic amplitude, (b) EM algorithm, and (c) NEM algorithm. The channel outlined in purple is composed of all the facies. In the NEM algorithm, C, constrains the red facies to mostly the channel fill unlike the EM algorithm.

CHAPTER 4

TARANAKI BASIN, NEW ZEALAND- KORA3D

4.1 *Geologic setting*

Within the Northern Graben of the Taranaki Basin there lies the Miocene-age submarine Kora Volcano that is the focus of this chapter. Radial faulting hypothesized to be caused by the cooling and collapse episodes of the volcanic body have been previously identified by Infante and Marfurt (2017).

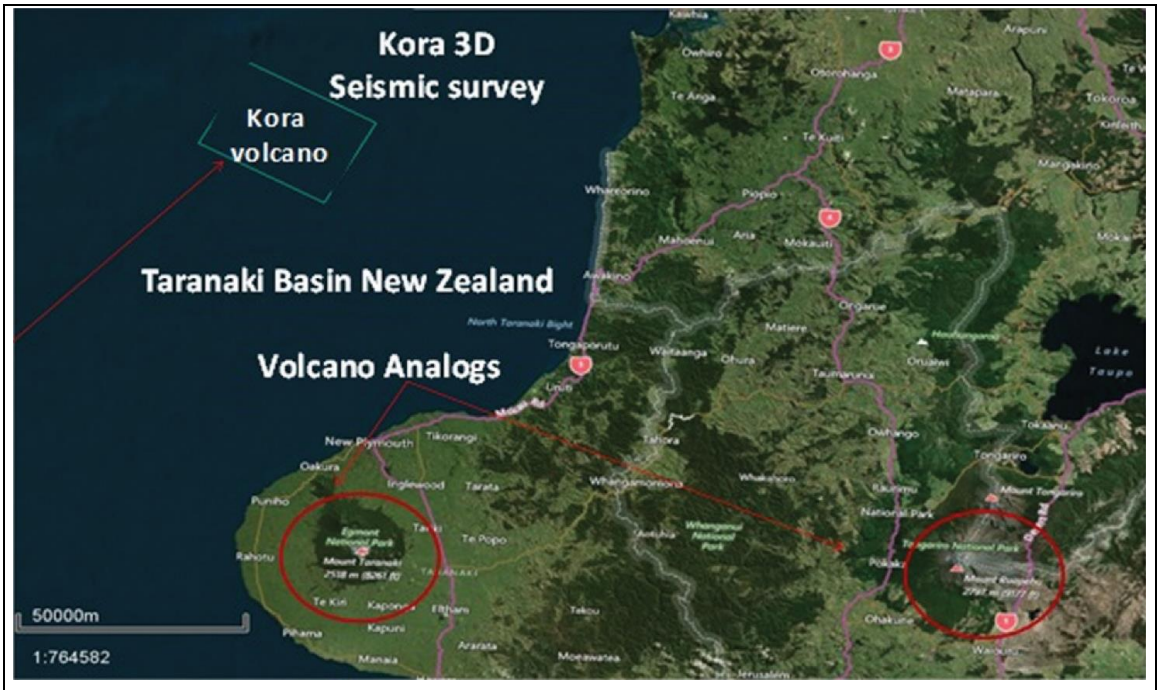

Figure 4.1. Aerial view of study area in Northern Graben of Taranaki Basin (Modified from Infante and Marfurt, 2017). Red circles signify modern volcano analogs for the Miocene volcano.

4.2 Methods

The Kora3d seismic data volume images the Kora Volcano and formed the basis of study published by Infante and Marfurt (2017). A horizon identifying the top of the Miocene volcanic is used to identify structural features (Figure 4.2). First, the inline dip and crossline dip measurements are calculated and filtered to be used for curvature attribute calculations. In addition to mapping the most-positive and most-negative curvature, I compute curvedness, reflector convergence magnitude, and reflector rotation about the normal (Figure 4.3).

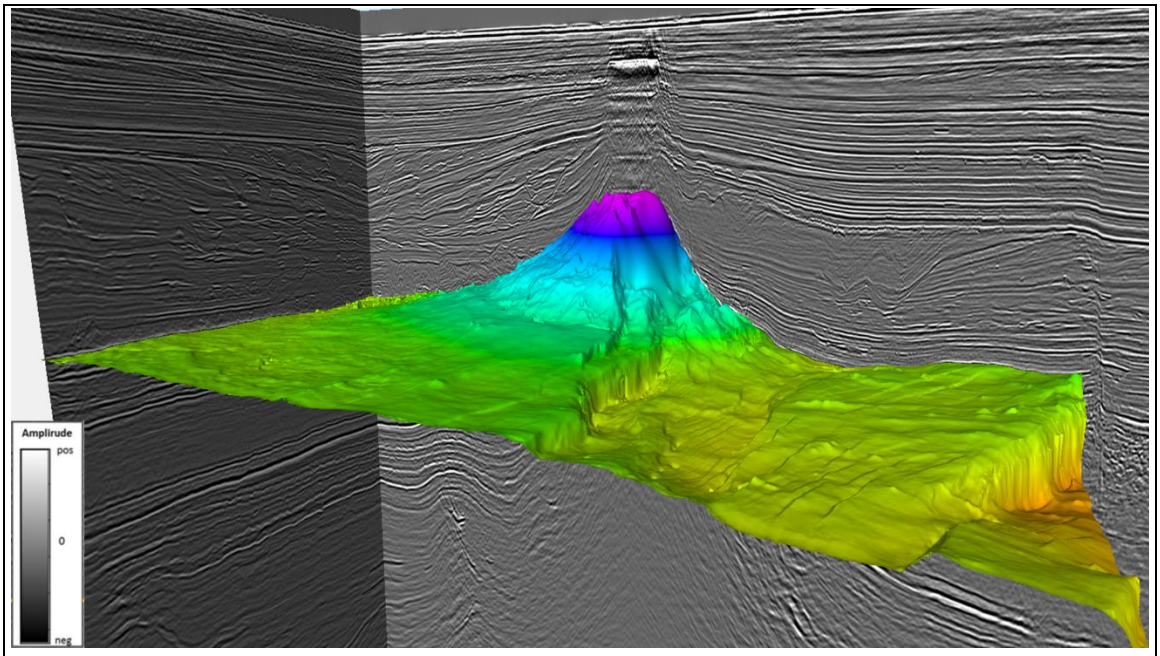

Figure 4.2. Top of the Miocene volcanic. Also contains onlapping wedge features on the outer edges deposited after the formation of the volcano.

Curvedness is the square root of the sum of squares of the two principle curvatures and is used here as a single measure of deformation. Reflector convergence magnitude and reflector rotation about the normal are two different measurements of the more generic reflector rotation vector, which is the curl of the normal vector to the dip

measurements. Reflector convergence azimuth is not being used because its cyclical nature isn't recognized well by machine learning techniques. In this data volume, reflector convergence magnitude measures vertical changes in dip where onlap, offlap, and erosional truncation would occur, while reflector rotation about the normal measures lateral changes in dip, where different rates of deposition across a fault or

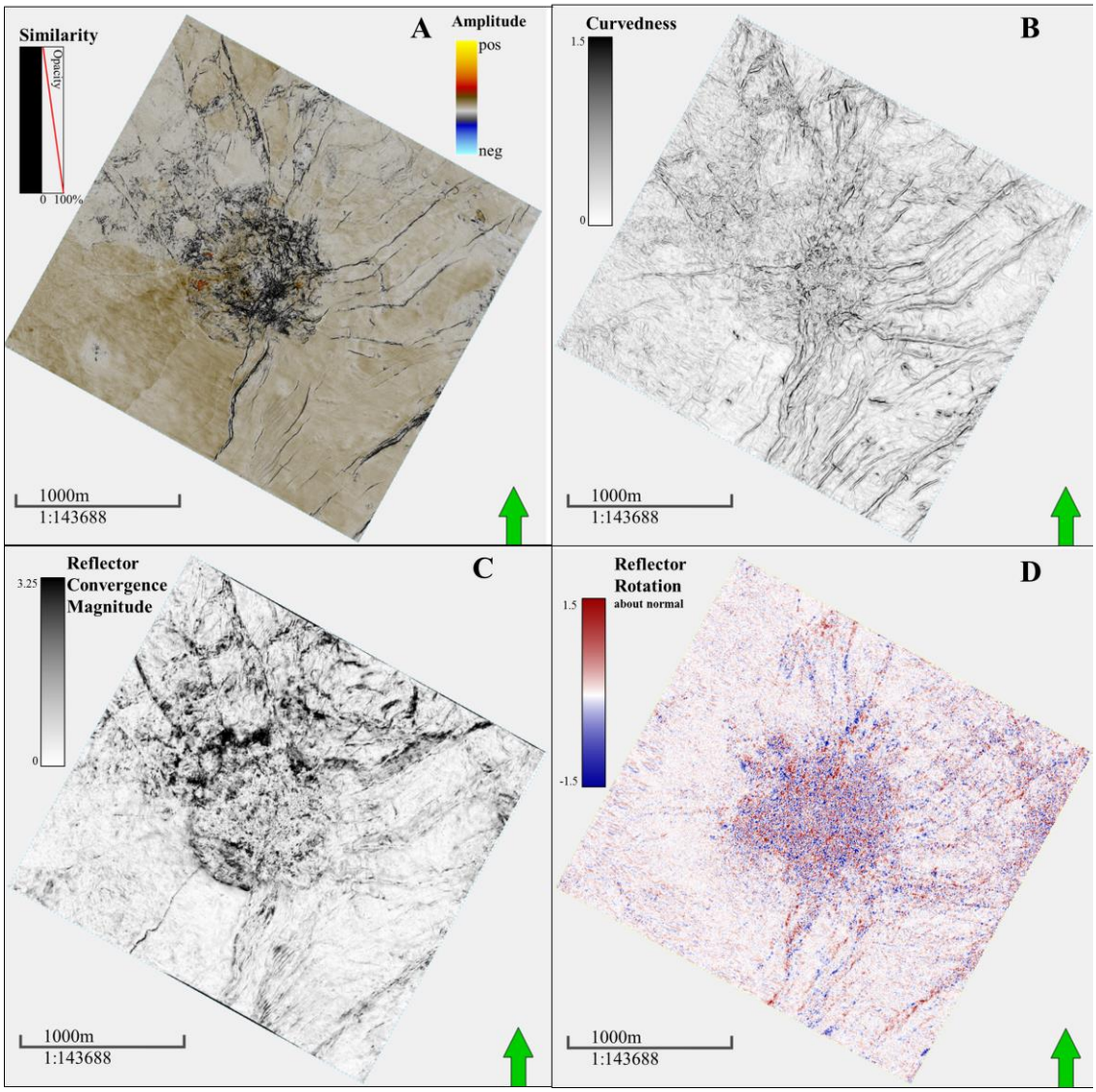

Figure 4.3. Top of the Miocene volcanic horizon slices. **A)** Seismic amplitude with similarity overlay **B)** Curvedness attribute **C)** Reflector convergence magnitude **D)** Reflector rotation about the normal direction.

rotation of reflectors about a fault would occur. Combining these three curvature attributes should show structural features with similar reflector orientation and deformation.

The next step is to generate a latent space using Self-Organizing Maps (SOM) (Kohonen, 1982). The three curvature attributes, curvedness, reflector convergence magnitude, and reflector rotation about the normal, define a 3D attribute space when cross-plotted against each other. The SOM is being used to project the 3D attribute space onto a deformed 2D SOM latent space. The deformation of the SOM latent space will cause similar attribute values to be close together while preserving the topology of the 3D attribute space. The 2D SOM latent space is output as a pair of attribute volumes that can be displayed on a 2D color bar (Figure 4.4).

GMMs are constructed from the 2D SOM latent space and the Bayesian Information Criterion (BIC) is calculated as a measure to compare all the different mixture models (Figure 4.5). The model with nine clusters and covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$ is selected because it is a local maximum and the BIC doesn't increase significantly after that point (Figure 4.6 and Table 4.1). The covariance of many of these clusters have a small eigenvalue and a much larger eigenvalue which can make the ellipsoid look more like a line (ex. Cluster 8 in the top left and Cluster 9 in the top right). Applying the selected mixture model to the data outputs a classification, ambiguity, and uncertainty attribute volumes.

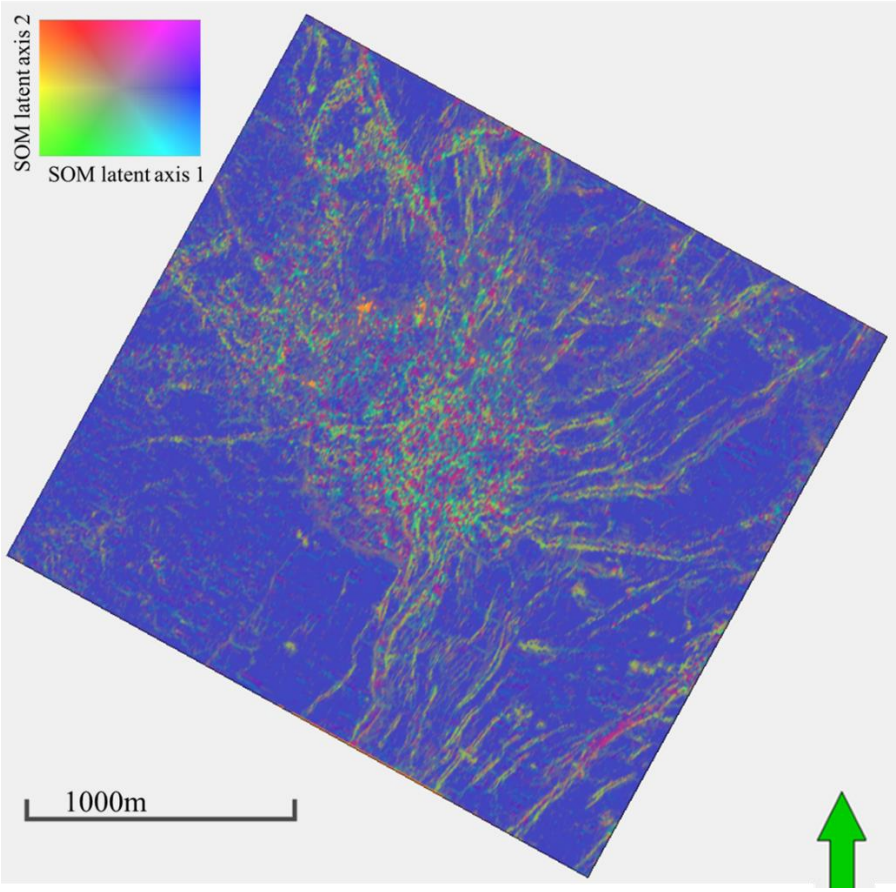

Figure 4.4. The SOM latent space. The SOM latent space is constructed use curvedness, reflector convergence magnitude, and reflector rotation about the normal.

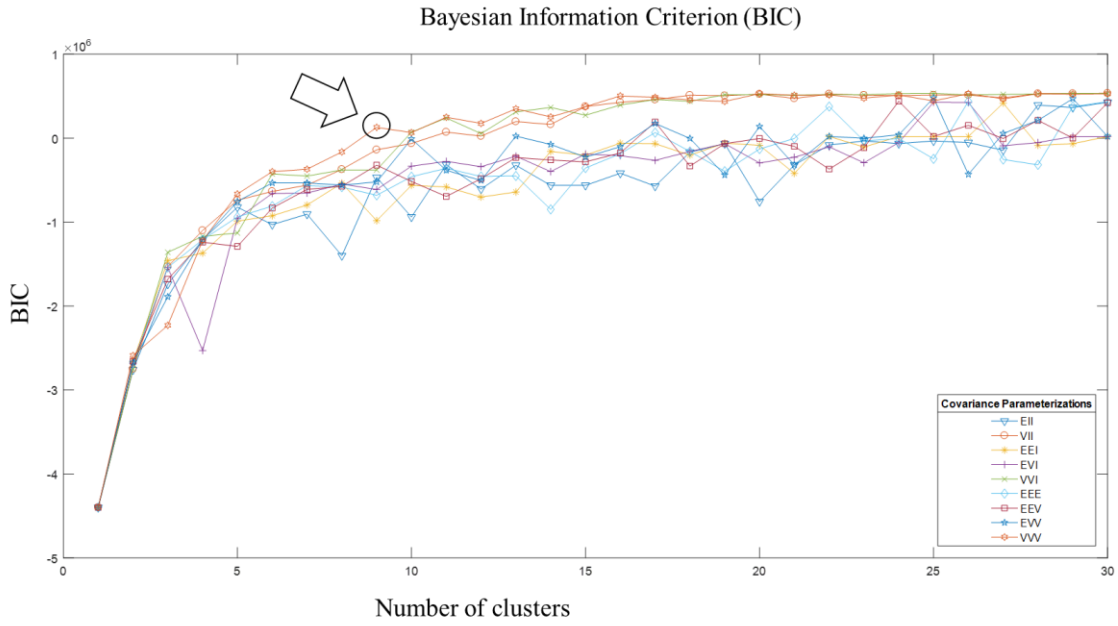

Figure 4.5. Bayesian Information Criterion. Arrow pointing to the selected model of 9 clusters and a covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$. Since the BIC does not decrease with the number of clusters, the model on the shoulder of the curve is picked. The sudden drops in BIC are likely due to clusters having near zero weight and being penalized without any increase to likelihood.

Table 4.1. Mixture model parameters for volcanic body. The selected model has 9 clusters and covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$.			
Cluster	π (%)	$\mu\begin{pmatrix}\text{SOM latent axis 1}\\ \text{SOM latent axis 2}\end{pmatrix}$	$\mathbf{C}$
1	1.96%	$\begin{matrix}0.430\\ -3.64\end{matrix}$	$\begin{matrix}1E-4 & 0\\ 0 & 1E-4\end{matrix}$
2	5.22%	$\begin{matrix}-0.469\\ -1.47\end{matrix}$	$\begin{matrix}1E-4 & 0\\ 0 & 1E-4\end{matrix}$
3	22.6%	$\begin{matrix}-0.761\\ -0.126\end{matrix}$	$\begin{matrix}1E-4 & 0\\ 0 & 1E-4\end{matrix}$
4	23.8%	$\begin{matrix}-0.216\\ -0.628\end{matrix}$	$\begin{matrix}0.193 & -0.196\\ -0.196 & 0.192\end{matrix}$
5	20.4%	$\begin{matrix}-0.430\\ -0.337\end{matrix}$	$\begin{matrix}3.82E-4 & 1.52E-3\\ 1.52E-3 & 8.33E-3\end{matrix}$
6	9.58%	$\begin{matrix}-0.176\\ -0.762\end{matrix}$	$\begin{matrix}1E-4 & 0\\ 0 & 1E-4\end{matrix}$
7	9.63%	$\begin{matrix}1.67\\ -0.413\end{matrix}$	$\begin{matrix}0.295 & -0.305\\ -0.305 & 0.637\end{matrix}$
8	5.49%	$\begin{matrix}-0.149\\ 2.3\end{matrix}$	$\begin{matrix}0.359 & 0.510\\ 0.510 & 0.726\end{matrix}$
9	1.35%	$\begin{matrix}4.68\\ 2.87\end{matrix}$	$\begin{matrix}1.84 & -1.24\\ -1.24 & 0.838\end{matrix}$

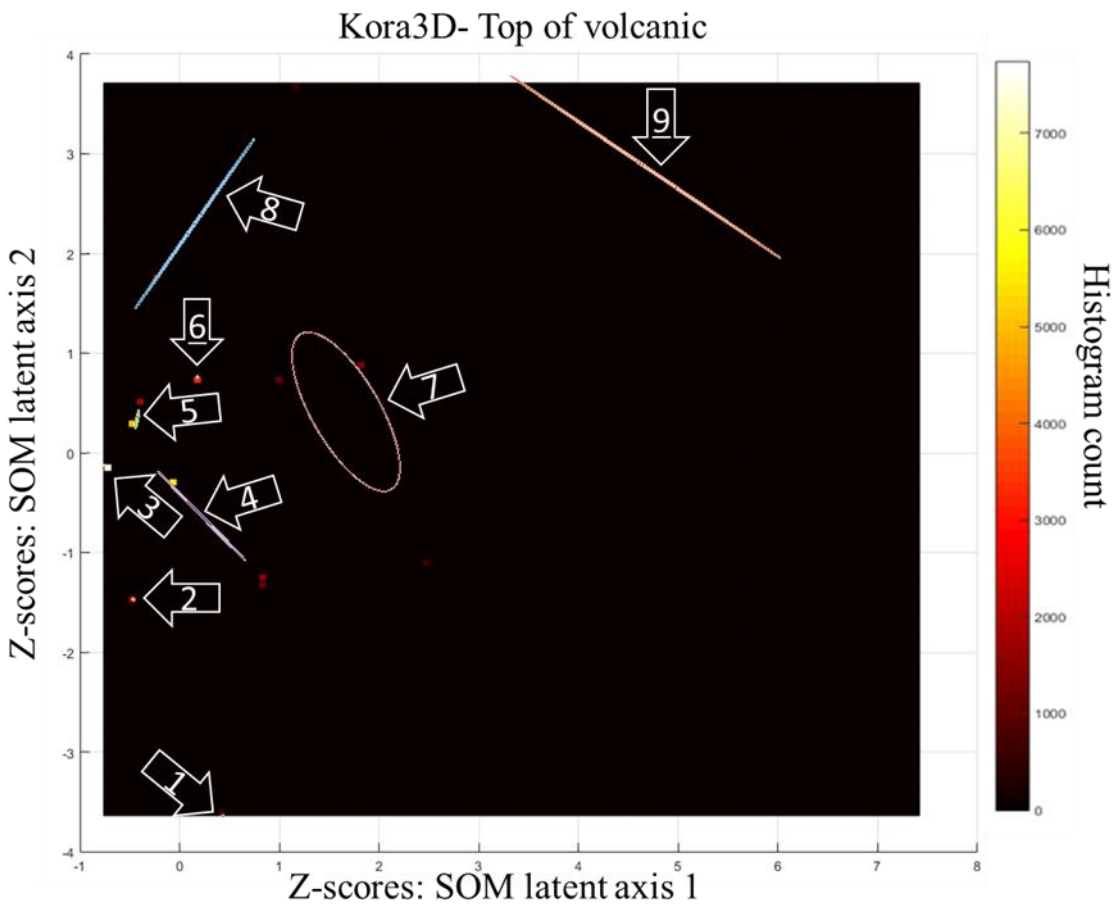

Figure 4.6. Histogram of training data with Gaussian ellipsoids. The Gaussian ellipsoids represent a one standard deviation contour of the Gaussian distribution and don't represent the mixing weights of the mixture model. The covariance parameterization of $\lambda_j \mathbf{D}_j \mathbf{A}_j \mathbf{D}_j^T$ allows for the Gaussian ellipsoids to vary in terms of size, shape, and orientation. The colors of the ellipsoids here are arbitrary and are not representative of any inter-cluster relationship.

4.3 Results and Interpretation

Each cluster is assigned a color close to where the clusters' means would be on the 2D SOM color bar as if the color bar was rescaled to fit the range of Z-scores (Figure 4.7). The continuous 2D color bar used for the SOM latent space is essentially reduced to nine different discrete colors with a measurement of how ambiguous the mixture model is in the color (Figure 4.8). The most dominant feature on the horizon slice is perhaps the

circular chaotic collection of clusters in the middle of the horizon corresponding to the location of the submarine volcano indicating that the reflectors within the volcano don't have a single trend.

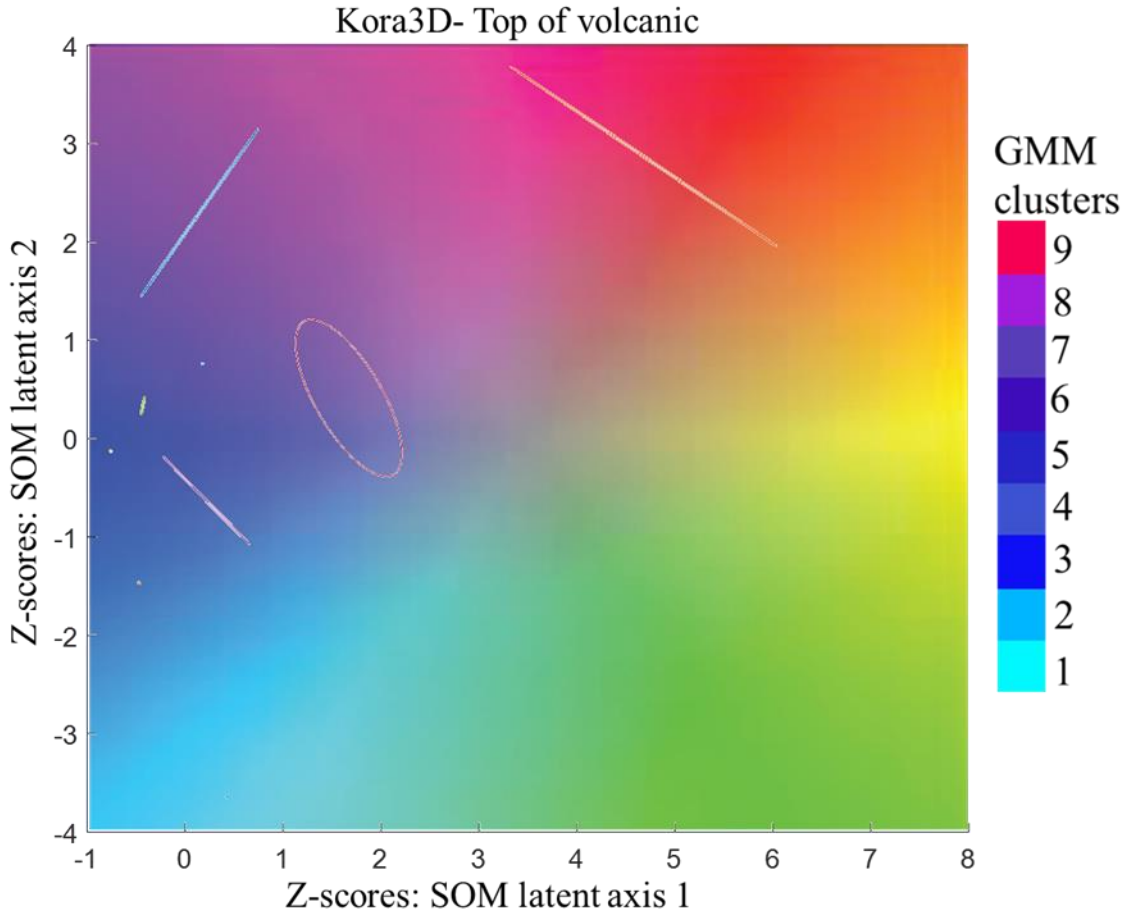

Figure 4.7. Gaussian ellipsoids with scaled 2D SOM color bar in background. The means of each cluster are in the center of each Gaussian ellipsoid, and each cluster is assigned a color close to where the mean lies.

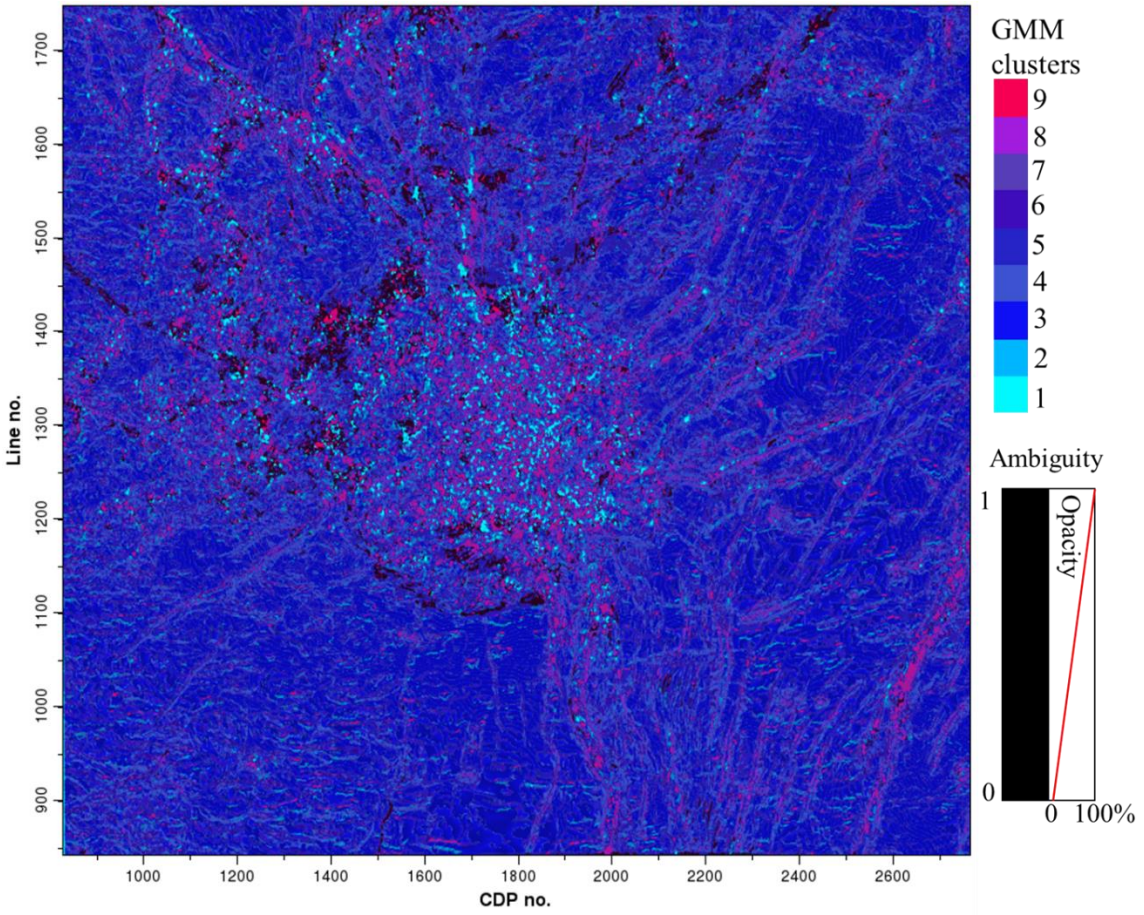

Figure 4.8. GMM classification with ambiguity. The chaotic collection of clusters in the middle correspond to the Miocene submarine volcano and indicate that the reflectors don't have a single trend in that area.

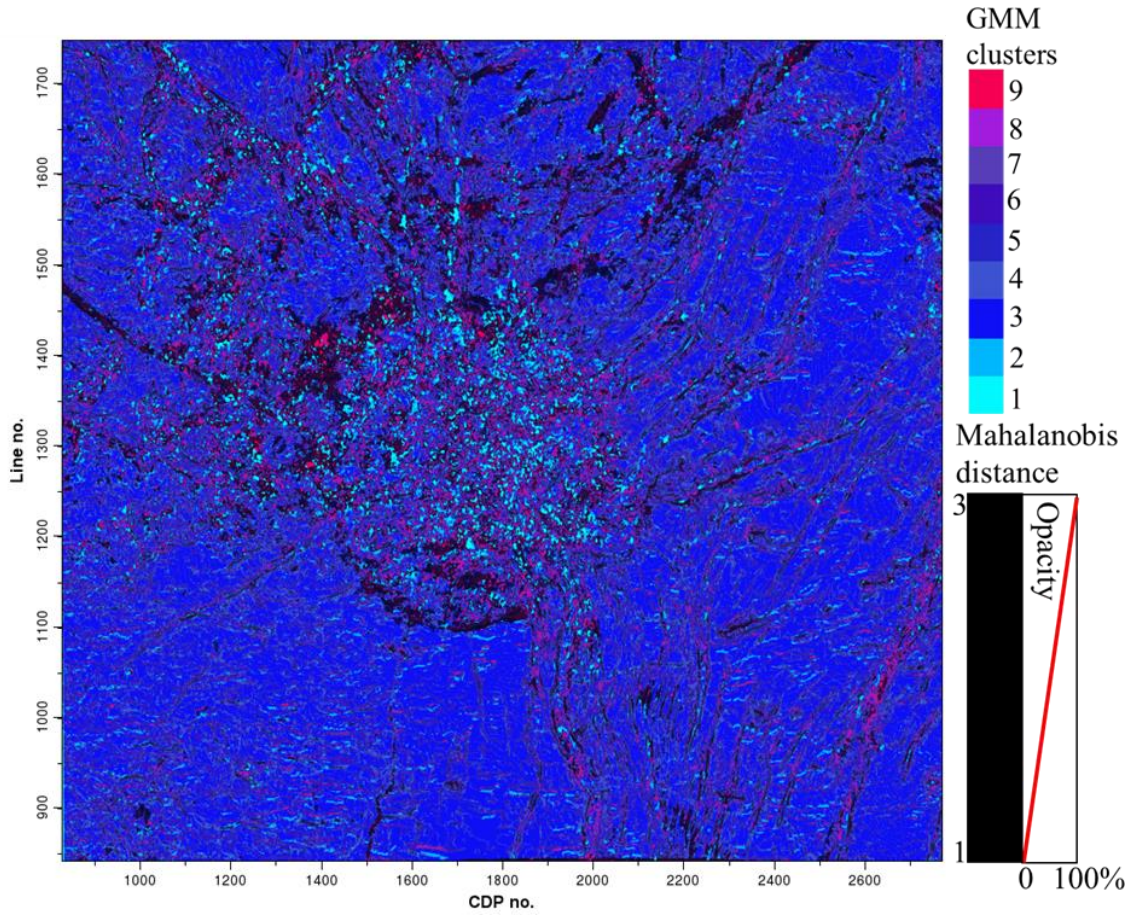

Figure 4.9. GMM classification with uncertainty The chaotic collection of clusters in the middle correspond to the Miocene submarine volcano and indicate that the reflectors don't have a single trend in that area.

Considering the spatial distribution of clusters can aide the interpretation. The 4th and 5th clusters don't have areas of high spatial continuity indicating that these clusters may not be representing anything geological. The 4th cluster is widespread, contains the second highest weight of the mixture model, and contains the most ambiguity in general. The 5th cluster seems to be concentrated outside of the volcano. However, other than those generalizations these clusters lack any clear structure or geological meaning.

The 3rd cluster has spatially continuous areas and is widespread which might be expected considering it takes up the most weight of the mixture model. The 3rd cluster

tends to avoid areas with lineaments. Moreover, the 3rd cluster is highly ambiguous when it does exist near areas with lineaments. The 1st cluster interpreted as having little deformation where reflectors don't have much rotation relative to the horizon.

The 1st, 2nd,6th, 7th, 8th, and 9th clusters contain lineaments related to a radial faulting pattern. The 1st, 2nd, and 9th clusters seem to delineate this pattern best in the southern part of the horizon (Figure 4.10). The classification is shown to have high confidence in areas where fault patterns may go unnoticed when using just the Sobel filter similarity attribute. In the western part of the horizon, radial faulting trending east-west is best delineated by the 1st and 8th cluster with the 1st cluster representing the thrown-down side and the 8th cluster representing the thrown-up side (Figure 4.11).

Table 4.2. Interpretation of clusters	
Cluster	**Interpretation**
1	Radial faulting; thrown-up block on western part
2	Clear lineaments associated with faulting
3	Little to no deformation; no change in reflectors relative to horizon.
4	Discontinuous. Not meaningful
5	Discontinuous. Not meaningful
6	Clear lineaments associated with faulting
7	Clear lineaments associated with faulting
8	Radial faulting; thrown-down block on western part
9	Clear lineaments associated with faulting

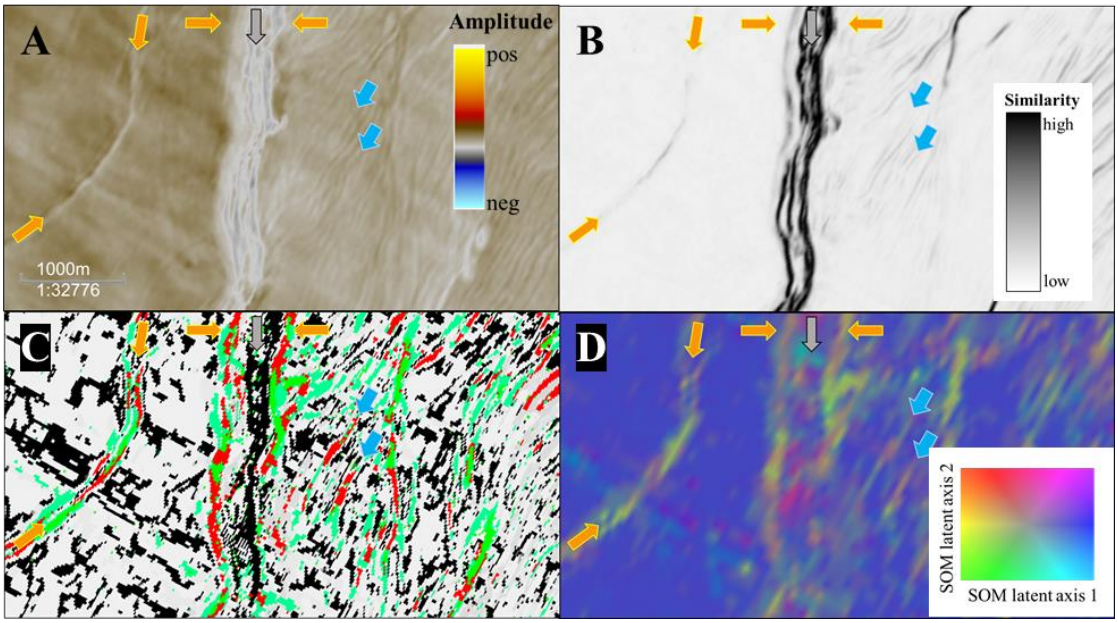

Figure 4.10. Southern part of horizon. **A)** Seismic amplitude, **B)** Sobel filter similarity, **C)** mixture model clusters **D)** SOM latent space. Orange arrows indicating radial faulting, gray arrow indication high ambiguity within a radial fault, and blue arrows showing features appear that in the SOM latent space and mixture model classification, but might go unnoticed in the seismic amplitude or Sobel filter similarity.

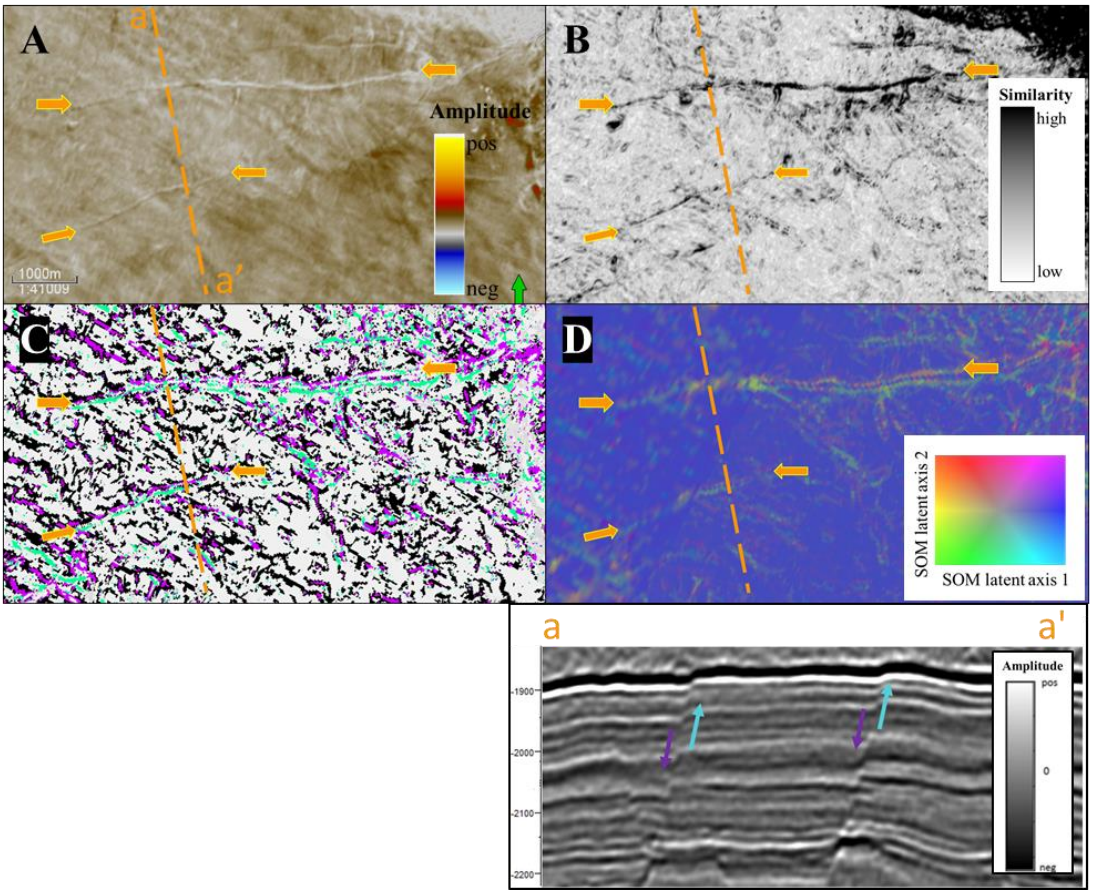

Figure 4.11. West trending radial faulting. **A)** Seismic amplitude, **B)** Sobel filter similarity, **C)** mixture model clusters **D)** SOM latent space. On the horizon slice, the edge of the thrown-up block is characterized by the 1st cluster, and the edge of the thrown-down block is characterized by the 8th cluster.

CHAPTER 5

CONCLUSIONS

When using self-organizing maps, the number of clusters is determined by a subjective interpretation of the number of distinct colors the interpreter sees. The probabilistic formulation of GMM allows for the number and shape of clusters to be determined from a more quantitative viewpoint via a Bayesian framework by considering a model's likelihood and complexity. Although the number of clusters can be evaluated, geophysical features may be represented by more than one cluster, especially when evaluating these clusters in terms of architectural elements. The representation of geophysical features by more than one cluster is often due to the difference between mathematical clusters and interpretational features. In addition, there is no reason for a natural cluster to be expressed by a Gaussian distribution. Due to the flexibility of GMM, a single natural cluster may be approximated by more than one Gaussian distribution, requiring subsequent "clumping" to aid the interpretation. The cases where the Bayesian Information Criterion doesn't decrease with the number clusters may be indicative of natural classes not being described well by a single Gaussian distribution.

The classification of data vectors using posterior probabilities results in measurements of uncertainty and ambiguity. Co-rendering the classification with the uncertainty and ambiguity measurements can produce an intuitive map of unsupervised seismic facies. Although uncertainty can identify anomalies, ambiguity should also be taken into account to determine how meaningful the anomaly is.

Geophysics has often been described as the search for anomalies and GMMs can help identify and evaluate such features. Optimizing a GMM using a combination of the SEM an CEM algorithms seems to work better than the EM algorithm from a computational standpoint. For interpretation, the CEM algorithm and classification approach makes interpretation much simpler and straightforward because data vectors are assumed to belong to a single cluster. The covariance parametrizations don't seem to affect the BIC as much as the addition of a new cluster and computation time can be reduced by only considering the unconstrained covariance matrix. However, singularities in the unconstrained covariance matrix can be difficult to deal with and a simpler parametrization might be considered. Ambiguity and uncertainty measurements highlight the strengths of GMMs as an interpretation tool. Plotting ambiguity along saturation, uncertainty along lightness, and the classification along hue on a hue-lightness-saturation color plot may improve the intuitiveness of the unsupervised seismic facies.

REFERENCES

Akaike, H., 1974, A New Look at Statistical Model Identification, **19**, 716-723.

Ambroise, C., M. Dang, and G. Govaert, 1997, Clustering of spatial data by the EM algorithm: geoENV I – Geostatistics for Environmental Applications, **9**, 493-504

Bensmail, H., G. Celeux, A. E. Raftery, and C. P. Robert, 1997, Inference in model-based cluster analysis: Statistical Computing, **7**, 1-10.

Celeux, G. and G. Govaert, 1992, A Classification EM algorithm for clustering and two stochastic versions: Computation Statistics & Data Analysis, **14**, 315-332.

Celeux, G. and G. Govaert, 1993, Gaussian Parsimonious clustering models: Pattern Recognition, **28**, 781-793.

Cozens, N., 2011, A Study of Unconventional Gas Accumulation in Dannevirke Series (Paleogene) Rocks, Canterbury Basin, New Zealand, thesis: Victoria University of Wellington, New Zealand.

Coleou, T., M. Poupon, and A. Kostia, 2003, Unsupervised seismic facies classification: A review and comparison of techniques and implementation: The Leading Edge, **22**, 942–953.

Dempster, A.P., N. M. Laird, and D. B. Rubin, 1977, Maximum likelihood from incomplete data via the EM algorithm: Journal of Royal Statistical Society, **39**, 1-38.

Edwards, A. W. F., 1974, The history of likelihood: International Statistical Review, **42**, 9-15.

Figueredo, M. A.T., and Anil K. Jain, 2002, Unsupervised learning of finite mixture models: IEEE Transactions on Pattern Analysis and Machine Intelligence: **24**, 381-395

Hathaway, R. J., 1986, Another interpretation of the EM algorithm for mixture distributions: Statistics & Probability Letters, **4**, 53-56.

Han, M., Y. Zhao, G. Li, and A. C. Reynolds, 2010, Application of EM algorithms for seismic facies classification: Computational Geosciences, **15**, 421-429.

Infante-Paez, L. and K. J. Marfurt, 2017, Seismic expression and geomorphology of igneous bodies: A Taranaki Basin, New Zealand, case study: Interpretation, **5**, SK121-140.

Kohonen, T., 1982, Self-organized formation of topologically correct feature maps: Biological Cybernetics, 43, 59-69.

MacQueen, J., 1967, Some methods for classification and analysis of multivariate observations: in L. M. Le Cam, and J. Neyman, eds., Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, 281–297.

Mahalanobis, P. C., 1936, On the generalized distance in statistics: In Proceedings National Institute of Science, India, **2**, 49-55.

Marroquín, I. D., J. -J. Brault, and B. S. Hart, 2009, A visual data-mining methodology to conduct seismic facies analysis, part 1: Testing and comparison with other unsupervised clustering methods: Geophysics, **74**, P1-P11.

Oliver, J., R. Baxter, C. Wallace, 1996, Unsupervised learning using MML: Proc. 13th International Conference on Machine Learning, 364-372.

Roden, R., T. Smith, and D. Sacrey, 2015, Geologic pattern recognition from seismic attributes: Principle component analysis and self-organizing maps: Interpretation, **3,** SAE59-SAE83.

Schwarz, G., 1978, Estimating the dimension of a model: The Annals of Statistics, 6, 461–464.

Strecker, S. and R. Uden, 2002, Data mining of 3D poststack seismic attribute volumes using Kohonen self-organizing maps: The Leading Edge, **21**, 1032-1037.

Wallet, Bradley C., Roderick P. Altimar, and Roger M. Slatt, 2014, Unsupervised classification of λρ and μρ attributes derived from well log data in the Barnett Shale: 84th Annual International Meeting of the SEG, Expanded Abstracts, 1594-1598.

APPENDIX A: Definitions

Ambiguity- A measure of uncertainty in the classification (*e.g.* a data vector that lies in between two GMM components would be ambiguous). Quantitatively defined in equation 2.5. See Bensmail et al. (1996).

Classification- The assignment of data vectors to a certain cluster. Usually done by assigning a data vector to the cluster that has the highest posterior probability (equation 2.4) (the NEM algorithm is an exception to this).

Cluster- A subset of data vectors that are similar to each other in some sense. See Hathaway (1986). In this thesis, the data vectors assigned to a GMM component are evaluated as a cluster.

Expectation-Maximization (EM)- A versatile algorithm used to find optimal estimates of the parameters of a GMM by finding a local maximum to the observed log-likelihood function (equation 2.7). See Appendix B and Dempster (1977).

Likelihood- A relative measure to different parameter values. Sufficient to compare GMMs of similar complexity. See Edwards (1974).

Mahalanobis distance- Measures how many standard deviations away a data vector is from a component of a GMM. See Mahalanobis (1936).

Neighborhood Expectation-Maximization (NEM)- Special variant of the EM algorithm which considers spatial correlations of the data. See Appendix C.

Uncertainty- A measure of how well a GMM describes a data vector; not to be confused with ambiguity. The Mahalanobis distance is a measure of uncertainty.

Z-score- A data vector where the population mean is subtracted and then divided by the population standard deviation. Used to condition the data initially.

APPENDIX B: Expectation-Maximization (EM)

The conventional expectation-maximization (EM) algorithm is an iterative method that finds a maximum likelihood or maximum a posteriori estimate. For a GMM, the EM algorithm finds the optimal values for the mixture model parameters, $\{\pi_k, \mathbf{\mu}_k, \mathbf{C}_k,\}$. The EM algorithm alternates between an *E-step* and an *M-step* to find a local maximum to the observed log-likelihood function. The *E-step* and *M-step* are repeated until the observed log-likelihood doesn't increase significantly or some other convergence criteria has been met. If we let $\varphi(\mathbf{x}|\mu, \mathbf{C})$ be a Gaussian distribution and $p(\mathbf{x}_n|\psi)$ be a Gaussian mixture density as defined in Chapter 2, the *E-step* and *M-step* can be described as follows:

- *E-step*: Accumulate the *N* by *K* matrix of posterior probabilities, **w**,

$$w_{n,k} = \frac{\pi_k \varphi(\mathbf{x}_n|\mathbf{\mu}_k,\mathbf{C}_k)}{\sum_{k=1}^{K} \pi_k \varphi(\mathbf{x}_n|\mathbf{\mu}_k,\mathbf{C}_k)} = \frac{\pi_k \varphi(\mathbf{x}_n|\mathbf{\mu}_k,\mathbf{C}_k)}{p(\mathbf{x}_n|\psi)} \tag{B.1}$$

- *M-step:* For each cluster, *k*, update the mixture parameters, $\{\pi_k, \mathbf{\mu}_k, \mathbf{C}_k,\}$, using the posterior probabilities in the *E-step.* Denoting the update parameters to be $\{\pi_k^+, \mathbf{\mu}_k^+, \mathbf{C}_k^+\}$, the update equations for an unconstrained covariance matrix become:

$$\pi_k^+ = \frac{\sum_{n=1}^{N} \pi_k \varphi(\mathbf{x}_n|\mathbf{\mu}_k,\mathbf{C}_k)}{N} \ , \tag{B.2}$$

$$\mathbf{\mu}_k^+ = \frac{\sum_{n=1}^{N} w_{n,k}\mathbf{x}_n}{\sum_{n=1}^{N} w_{n,k}} \text{ , and} \tag{B.3}$$

$$\mathbf{C}_k^+ = \frac{\sum_{n=1}^{N} w_{kn}(\mathbf{x}_n - \mathbf{\mu}_k^+)(\mathbf{x}_n - \mathbf{\mu}_k^+)^{\mathrm{T}}}{\sum_{n=1}^{N} w_{n,k}} . \tag{B.4}$$

APPENDIX C: Stochastic Expectation-Maximization (SEM) and Classification Expectation-Maximization (CEM)

The Classification Expectation-Maximization (CEM) and Stochastic Expectation-Maximization (SEM) algorithms are like the conventional EM algorithm but with the addition of a C-step for CEM and an S-step for SEM. The C- and S-step of each algorithm defines a partition of the input data, with the mixture model parameters being updated for each partition. The CEM algorithm defines the partition using K-dimensional binary indicator vectors, $\mathbf{z}_n = \{z_{n1}, z_{n2}, \ldots z_{nK}\}$ for $n = 1 \ldots N$, where z_{nk} has a value of 1 if and only if the data vector $\mathbf{x}_n$ belongs to cluster k (Biernacki et al., 2000). The CEM algorithm seeks to maximize the complete log-likelihood of ψ given by

$$L_{\mathrm{c}}(\psi) = \sum_{k=1}^{K} \sum_{n=1}^{N} z_{nk} \log\{\pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)\} \,. \tag{C.1}$$

Key components of the CEM algorithm include the:

- *E-step*: Accumulate the *N* by *K* matrix of posterior probabilities, **w**,

  $$w_{n,k} = \frac{\pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)}{\sum_{k=1}^{K} \pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)} = \frac{\pi_k \varphi(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{C}_k)}{p(\mathbf{x}_n|\psi)}$$

  (C.2)

- *C-step*: Create *K*-partitions by assigning each $\mathbf{x}_n$ to the cluster that provides the highest posterior probability according to the matrix **w** computed in the *E-step*. Formally this is done by using a *K*-dimensional latent variable, $\mathbf{z}_n$, indicating the cluster to which each $\mathbf{x}_n$ belongs. Let P_k denote the resulting partition for the k^{th} cluster and M_k be the number of data vectors in the partition. In other words, $\mathbf{x}_n \epsilon P_k$ if $z_{nk} = 1$, and M_k is the number of elements in P_k.

- *M-step*: For each cluster, *k*, update the mixture parameters, $\{\pi_k, \boldsymbol{\mu}_k, \mathbf{C}_k,\}$, with the respective partition defined in the previous step. Denoting the update parameters to be $\{\pi_k^+, \boldsymbol{\mu}_k^+, \mathbf{C}_k^+\}$, the update equations for an unconstrained covariance matrix become:

$$\pi_k^+ = \frac{M_k}{N} \ , \tag{C.3}$$

$$\boldsymbol{\mu}_k^+ = \frac{1}{M_k}\sum_{\mathbf{x}_n \epsilon P_k} \mathbf{x}_n \ \text{, and} \tag{C.4}$$

$$\mathbf{C}_k^+ = \frac{1}{M_k}\sum_{\mathbf{x}_n \epsilon P_k} (\mathbf{x}_n - \boldsymbol{\mu}_k^+)(\mathbf{x}_n - \boldsymbol{\mu}_k^+)^{\mathrm{T}} \ . \tag{C.5}$$

The SEM algorithm is identical, except that the *C*-step is replaced by the *S*-step:

- *S-step*: Create *K*-partitions by randomly assigning each $\mathbf{x}_n$ to a cluster using the posterior probabilities in the matrix generated from the *E-step*

The EM-based algorithms find a local maximum to a log-likelihood function, and may provide suboptimal solutions. The SEM algorithm helps to avoid these sub-optimal solutions by reducing the dependence on initialization. However, since the S-step is random, the partition it creates is also random such that point-wise convergence cannot be guaranteed. For this reason, the number of SEM iterations are set to be a large fixed number, and the iteration with the highest complete log-likelihood, $L_{\mathrm{c}}(\psi)$, is selected as the best model. To make the final result more deterministic, the resulting SEM model is used to initialize the CEM algorithm which then gives a final partition and GMM.

APPENDIX D: Neighborhood Expectation-Maximization (NEM)

Learning of a GMM using the EM algorithm is a purely statistical construct and doesn't consider spatial correlations. In general, facies are expected to be at least laterally continuous to some extent. To account for spatial correlations of the latent space the Neighborhood expectation-maximization (NEM) algorithm can be implemented. The conventional EM algorithm can be viewed as a variant of coordinate descent on a certain objective function,

$$\mathrm{D}(\boldsymbol{w}, \psi) = \sum_{k=1}^{K} \sum_{n=1}^{N} w_{k,n} [\log\{ w_{k,n}\} - \log\{\pi_k \varphi(\mathbf{x}| \boldsymbol{\mu}_k, \mathbf{C}_k)\}] \qquad \text{(D.1)}$$

where $w_{k,n}$ are the elements of the responsibility matrix, $\boldsymbol{w}$ (Hathaway, 1986). Ambroise et al. (1996) introduced a regularization term to take into account the spatial information of the data,

$$\mathrm{G}(\mathbf{w}) = \frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{p=1}^{N} w_{ik} \cdot w_{pk} \cdot v_{ip} \qquad \text{(D.2)}$$

where v_{ip} are the elements of a N x N "neighborhood matrix", **v**. The new objective function then becomes

$$\mathrm{U}(\mathbf{w}, \psi) = \mathrm{D}(\mathbf{w}, \psi) + \beta \cdot \mathrm{G}(\mathbf{w}) \qquad \text{(D.3)}$$

where $\beta \geq 0$ and determines the weight of the spatial term, G(**w**). The "neighborhood matrix", **v**, for this application has been chosen to be

$$v_{ip} = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_p \text{ neighbors} \\ 0 & \text{else} \end{cases}, \qquad \text{(D.4)}$$

and $\mathbf{x}_i$ and $\mathbf{x}_p$ are neighbors if they both lie within a user-defined window. The benefit of the NEM algorithm is that the responsibilities of neighboring voxels are considered when deciding which mixture component a voxel belongs to.