The Setwise Stream Classification Algorithm

Charles Lett Jr.; Dr. Shamsudidn
Langston University, Oklahoma State University

Introduction

The main goal of this project was to learn about machine learning and to implement Dr. Aggarwal's Setwise Stream Classification algorithm, using a dataset collected from a real-world healthcare study to evaluate the proposed algorithm and to see what improvements could be made in the future. This alogrithm is a new approach to classifying data using machine learning with datasets composed of multiple subsets of data, allowing for a more accurate classification. This algorithm also runs in real time, accepting a data stream and dynamically creating a classification model. A lot of data, such as that collected from sensors, require a robust classification method that is able to work with complex datasets.

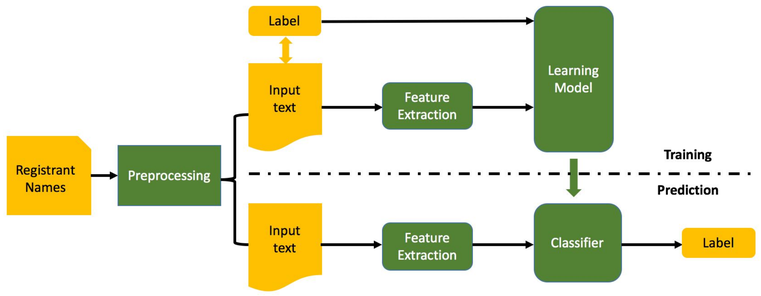


Figure 1. Sample Classification Diagram *[4]*.

Related Work

This research project is based mainly on the implementation of the Dr. Aggarwal's algorithm for classifying a streamed set of data, which is detailed in his paper: *The Setwise Stream Classification Problem [1]*, however, concepts are used from others, such as T. M. Cover and P. E. Hart's *Nearest Neighbor Pattern Classification [2]* as well.

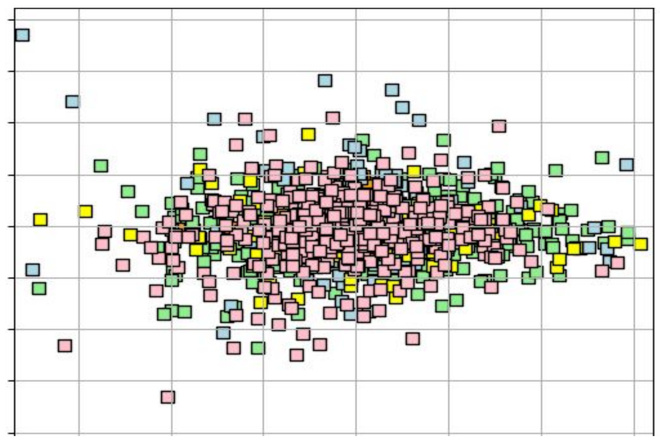
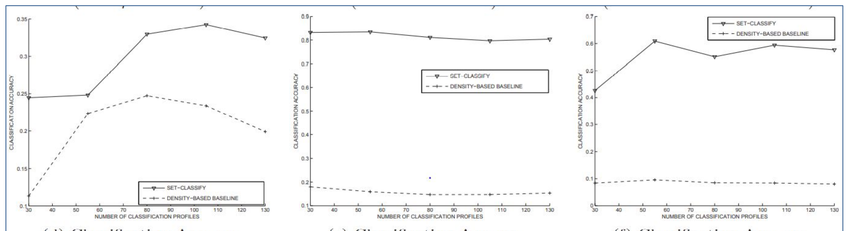


Figure 2. Setwise Classification Scenario generated from the UCI_HAR dataset.



(d) Classification Acc. vs. Number of Clusters (*Hub64* Data Set) (e) Classification Acc. vs. Number of Clusters (*E-Cover* Data Set) (f) Classification Acc. vs. Number of Clusters (*Gauss50Mix10D1000* Data)

Figure 3. Accuracy vs Profile Count, Aggarwal [1]

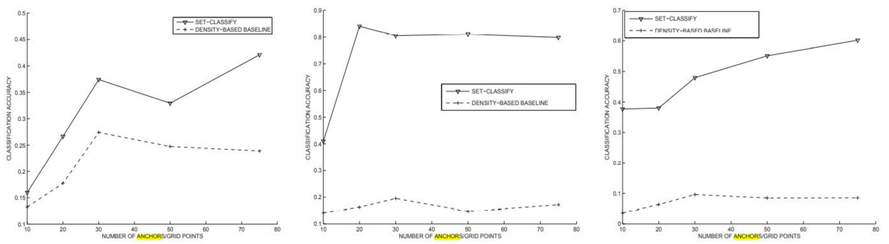
Technical Approach

(My implementation was created using Python on a Windows 10 system with a 4.2Ghz processor and 32GB of RAM)

- Algorithm was used on the *Human Activity Recognition Using Smartphones* Dataset (UCI_HAR) [3].
- First step was to partition the training portion of the dataset in order to create a sample of the data. The sample data was then used in a k-means algorithm to generate anchors
- Next, fingerprints, each representing an entity (or person in case of this dataset) and profiles (used to keep track of classes) are initialized.
- The rest of the training data is passed into the algorithm where the fingerprints and profiles are updated as each data point is processed.
- Finally, the test portion of the data is inputted into the algorithm, where each piece of data is added to a new fingerprint and updated with a predicted label using a nearest neighbors approach.

	(Personal) UCI_HAR	(Aggarwal) Hub64	(Aggarwal) E-Cover	(Aggarwal) Gauss50Mix10D1000
Accuracy	18.05%	> 40% (27%)	> 80% (20%)	~ 60% (~10%)
Anchor #	5	75	75	80
Profile #	16	80	80	50

Table 1. Accuracy, number of anchors and profiles (Note: % = set-classify anchor, (%) = density-based)



(a) Classification Acc. vs. Number of Anchors (or Grid Points) (*Hub64* Data Set) (b) Classification Acc. vs. Number of Anchors (or Grid Points) (*E-Cover* Data Set) (c) Classification Acc. vs. Number of Anchors (or Grid Points) (*Gauss50Mix10D1000* Data)

Figure 4. Accuracy vs. Anchor Count, *Aggarwal [1]*

Results

- I assumed that as the number of anchors and profiles increased, so would the accuracy, and when used on a small synthetic dataset that was the case for the most part, however, once I ran the UCI_HAR dataset through my algorithm I found that no matter how many anchors or profiles were chosen, the accuracy of predictions made remained within the margin of error alongside being low.
  - Afterwards I ran the UCI_HAR dataset through a 'random forest' classifier to verify that the problem wasn't the data.
- My hypothesis regarding accuracy was neither verified or dismissed due to my implementation not producing satisfactory results. On the other hand, Dr. Aggarwal's experiments show that classification accuracy tends to increase as the number of anchors increase, although after a certain number of anchors and profiles accuracy will stabilize and produce diminishing returns.

Conclusions

- The Setwise Stream Classification algorithm is a marginally accurate and efficient approach to classifying complex sets of data, where a dataset contains subsets of data and given its ability to run in real time, allows for more accurate, on-the-fly predictions based on data collected in the moment.
- Now as to my personal work, I found that my implementation is flawed when it comes to complex datasets and could definitely use work with the real time aspect, however, the results that were produced show that there is a lot of potential for this implementation.

References

[1] Charu C. Aggarwal. 2014. The setwise stream classification problem. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (*KDD '14*). Association for Computing Machinery, New York, NY, USA, 432–441. DOI:https://doi.org/10.1145/2623330.2623751
[2] T. M. Cover, and P. E. Hart. *Nearest Neighbor Pattern Classification*. IEEE Transactions on Information Theory, 13(1), pp. 21–27, 1967
[3] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. *A Public Domain Dataset for Human Activity Recognition Using Smartphones*. *21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013*. Bruges, Belgium 24-26 April 2013.
[4] Huayi Jing, *Registrant Classification using Machine Learning*, February, 19, 2021, Retrieved from:https://blog.nzrs.net.nz/registrant-classification/

Acknowledgements

Thank you in part to the National Science Foundation OK-LSAMP Program Grant No. HRD-1911370 OK-LSAMP for supporting this project and its author(s). Any opinions, findings, conclusions, and such are representative of the author(s) and do not necessarily reflect the views of the National Science Foundation. We would also like to thank Ethan Strickler alongside the other members of Oklahoma State University's REU program for their help and support.