JOURNAL OF
EMERGING INVESTIGATORS

Article

Similarity graph-based semi-supervised methods for multiclass data classification

[1]Ashwin Balaji, [2]Ekaterina Merkurjev
[1]Novi High School, Novi, Michigan
[2]Michigan State University, Department of Mathematics, East Lansing, Michigan

SUMMARY

The purpose of the study was to determine whether graph-based machine learning techniques, which have increased prevalence in the last few years, can accurately classify data into one of many clusters, while requiring less labeled training data and parameter tuning as opposed to traditional machine learning algorithms. We hypothesized that traditional machine learning algorithms, such as support vector machines (SVM), neural networks, and random forests, would perform accurately with less labeled training data and parameter tuning compared to their graph-based counterparts. We tested three traditional algorithms, (SVM, neural networks, and random forests), and two graph-based algorithms, (K Nearest Neighbors (KNN) and a graph-based adaptation of the classical Merriman-Bence-Osher scheme for estimating mean curvature). We ran each algorithm across three datasets of varying dimensionality, or number of features – the data banknote dataset, letter recognition dataset, and breast cancer dataset contained 5, 26, and 30 features, respectively. Algorithms were analyzed using training data, taken as a subset of each overall dataset, and averaged across four iterations. Our results did not support the hypothesis as the traditional algorithms did not outperform the graph-based techniques on all datasets, regardless of dimensionality. We determined that the accuracy of graph-based and traditional classification algorithms depends directly upon the number of features of each dataset, the number of classes in each dataset, and the amount of labeled training data used.

INTRODUCTION

Data classification and segmentation, defined as the process of categorizing data into a pre-specified number of clusters, is a machine learning task that is vital to the creation of algorithms with predictive capabilities and has applications in virtually every field. This task is incredibly challenging because of the reliance on large, labeled training datasets. In fact, modern machine learning techniques, such as support vector machines (SVM) and neural networks, require large amounts of cleaned, processed, and labeled data to create accurate models (1). State-of-the-art deep learning techniques, such as convolutional neural networks, require tuning of millions of free parameters to produce optimal results (2). In recent years, graph-based semi-supervised approaches have been developed, requiring less parameter tuning and labeled data to perform accurately (3). As opposed to traditional machine learning algorithms, which may use mathematical approximations and three-dimensional visualizations, these graph-based models operate on the framework of graph-theory. They can be structured through graphs consisting of vertices (or node points) and their connecting edges. The vertices can represent different elements in datasets, whereas their connecting edges can represent different relationships between each element. These representative graphs make data and patterns easily visible to the human eye and subsequently allow high-dimensional patterns to be detected by computer algorithms (4).

This project provides a comparison of multiple modern machine learning techniques to their newer graph-based counterparts. We analyze three traditional classification methods, (SVM, neural networks, and random forests), and two graph-based methods, (K nearest neighbors (KNN) and a graph-based adaptation of the classical numerical Merriman-Bence-Osher (MBO) scheme). We assessed the models across three different datasets, two binary classification datasets and one multiclass classification dataset, ranging from 600 to 20,000 elements (5-7).

SVMs function by producing a decision barrier between separate classes. In the case of dual-class segmentation, the SVM produces a singular decision barrier to separate data into two separate classes, known as a hyperplane (5). Neural networks use a series of input layers with associated weights and biases to classify data into different categories. The network, modeled off the human brain and its system of neurons, takes in an input layer with a size corresponding to the number of input features. Likewise, the output layer takes in a size equivalent to the number of classes in the segmentation problem (8). Random forests function as a collection of multiple, separate decision trees. Decision trees take in multiple, randomized subsets of a set of data. A decision tree then takes in an attribute at its root node and conducts a series of if/else conditionals to determine to which class each element corresponds (9). The KNN is a simple algorithm that predicts the class of a point based off the majority of its pre-specified nearest neighbors, or points with least distance to the point in question when the data is graphically represented. Based off the number of nearest neighbors, the algorithm will assess the classes off the points that are closest to the point for which the class needs to be predicted and take the majority to predict a point's class (10). The first graph-based semi-supervised learning method uses an adaptation of the Merriman-Bence-Osher scheme for multiclass segmentation. The model builds on a vector-field application of the Ginzburg-Landau function – "A vector-valued quantity is assigned to every node on the graph, such that each of its components represents the fraction of the phase, or class, present in that particular node" (3). In conjunction with the Merriman-Bence-Osher scheme for analyzing motion of a certain hypersurface by mean curvature, the Ginzburg-Landau adaptation is used to create a graph-based model for binary class segmentation. This is then extended into multiclass segmentation.

For each method, we averaged the accuracies from four different subsets of each to determine the relationship between training size and accuracy for each model. We hypothesized that traditional machine learning classification models would require less labeled data and parameter tuning to perform accurately across all three datasets. However, on average, the graph-based datasets consistently performed more accurately on the low and medium dimensionality dataset and had a steeper learning curve on the medium dimensionality dataset, meaning that it comparatively performed better with more training data. This research can provide insight into how the new field of graph-based techniques can create more accurate semi-supervised classification models.

RESULTS

By increasing the training size fed into each model, we can assess the model's overall performance, as well as the amount of training data needed to consistently produce a generally accurate model. For each dataset, training sets of increasing size were taken as subsets of the overall dataset. For each training set size, the performance of the MBO adaptation, neural network, and SVM algorithms were averaged across four random testing subsets of the data. Due to the minimal variation in accuracy across iterations, the KNN algorithm was averaged across only three random subsets of the data per training set, with each iteration using a different number of nearest neighbors. For the two binary classification datasets, training accuracies were assessed in multiples of 50. In the multiclass classification dataset, training accuracies were assessed in multiples of 1000.

Results on cancer dataset

The cancer dataset was a binary dataset with a high dimensionality. The dataset contained 30 features. On the cancer dataset, the traditional SVM and graph based KNN algorithms were able to consistently produce accuracy in the 90th percentile. The graph based MBO adaptation outperformed the traditional neural network, with an average accuracy and peak accuracy of approximately 80.9% and 94.2%, respectively, as opposed to an average and peak accuracy of approximately 71.8 and 85.9 %, respectively. The traditional random forest produced an average accuracy of 88.2 % and a peak accuracy of 91.2 % on 500 elements **(Figure 1, Table 1)**.

Results on data banknote dataset

The data banknote dataset was a binary dataset with low dimensionality. The dataset contained 5 features. On the data banknote set, the random forest model averaged an accuracy of 82.075 % and produced an accuracy of 92.3 % at a training size of 1200. The MBO, SVM, KNN, and neural network models all averaged an accuracy of around 99 %, yet the MBO Adaptation consistently outperformed the other two algorithms, producing a near perfect accuracy with a fraction of the labeled data (**Figure 2, Table 2)**.

Results on letter recognition dataset

The letter recognition dataset was a multiclass dataset with a medium dimensionality. The dataset contained 26 features. On the letter recognition dataset, all four models seemed to perform smoothly, with the MBO model performing

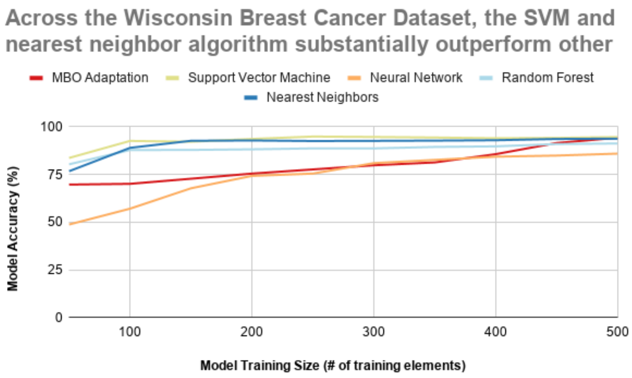

Figure 1. The Wisconsin Breast Cancer Dataset. The SVM and nearest neighbor algorithms substantially outperformed other models.

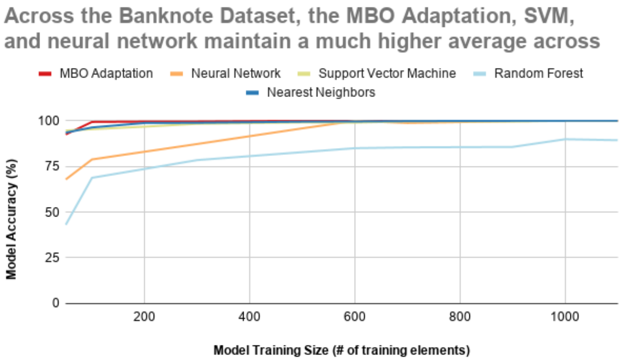

Figure 2. The Banknote Dataset. The MBO Adaptation, SVM, and neural network maintain a much higher average across accuracy for training sizes under 600 elements.

slightly better than the others. The MBO model achieved a peak accuracy of 97.1 % at a training size of 19,000, while the SVM, neural network, random forest, and KNN models all achieved accuracies of 90.4, 96.4, 96.7, and 94.9 %, respectively **(Figure 3, Table 3)**.

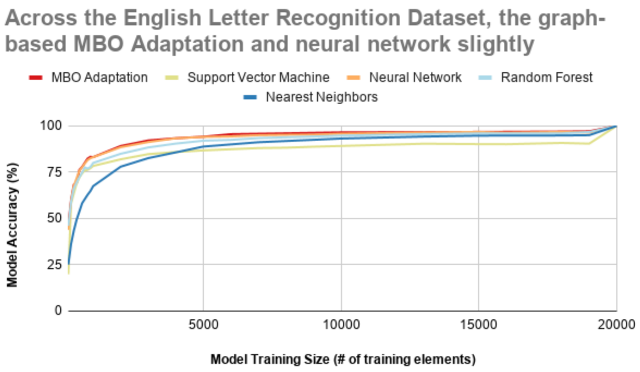

Figure 3. The English Letter Recognition Dataset. The graph-based MBO Adaptation and neural network slightly outperform all other models.

DISCUSSION

To create accurate machine learning models for the task of data classification, large amounts of labeled training data must be collected, often through extensive and laborious research conducted by trained professionals. We analyzed graph-based machine learning classification algorithms and compared them against the more commonly used and traditional machine learning algorithms. We hypothesized that the traditional machine learning methods, such as SVMs, would require less labeled training data to perform accurately across all datasets. Repeated iterations and multiclass adaptations to these traditional algorithms may have increased their reliability as compared to relatively new graph-based classification algorithms (1). We found that on datasets with more features, random forest trees and graph-based techniques required less labeled data to perform accurately as opposed to traditional techniques. On binary classification datasets, the SVM algorithm required less labeled data as compared to other algorithms to consistently perform accurately. On the low dimensionality banknote dataset and medium dimensionality letter recognition dataset, the MBO outperformed the other models. In the case of the banknote dataset, the MBO adaptation was able to reach an accuracy of over 99% at a training size of 100 elements, as opposed to 600 elements on the second-best performing model, the SVM algorithm. In the case of the letter recognition dataset, the MBO adaptation was able to consistently provide a 1-2% increase compared to the second-best performing model, the neural network model. We noted that graph-based algorithms tended to work better on multi-class data, whereas both graph-based and traditional models performed

Table 1. Statistical analysis of model performance across the Wisconsin Breast Cancer Dataset.

	SVM	Neural Network	Random Forest	MBO Adaptation	KNN
Mean	92.82%	73.25%	88.24%	80.94%	90.88%
Median	94.02%	75.44%	88.62%	81.28%	92.65%
Standard Deviation	3.35%	13.09%	3.03%	9.13%	5.16%
Range	11.20%	37.10%	10.91%	24.52%	17.03%

Table 2. Statistical analysis of model performance across the Data Banknote Dataset.

	SVM	Neural Network	Random Forest	MBO Adaptation	KNN
Mean	98.67%	92.02%	78.70%	99.16%	98.75%
Median	99.33%	98.76%	84.97%	99.94%	99.61%
Standard Deviation	1.71%	11.68%	14.82%	2.22%	2.04%
Range	5.35%	32.14%	46.82%	7.49%	6.50%

Table 3. Statistical analysis of model performance across the English Letter Recognition Dataset.

	SVM	Neural Network	Random Forest	MBO Adaptation	KNN
Mean	79.50%	86.51%	84.20%	84.52%	74.33%
Median	85.32%	93.68%	91%	89.14%	82.58%
Standard Deviation	16.05%	13.82%	13.76%	14.13%	23.04%
Range	80.09%	56.12%	53.92%	52.88%	74.87%

well on binary datasets, depending on dimensionality (graph-based models performed better on the banknote dataset, with lower dimensionality). These observations allow us to vary model choice based on the dimensionality of datasets and the number of classes. Despite overfitting and underfitting techniques, as well as hyperparameter tuning, suboptimal model parameters may have skewed the results of some algorithms disproportionately. Within the binary Wisconsin Breast Cancer Dataset, the data contained an approximately 60/40 split between benign and malignant cells as opposed to the ideal 50/50 split. This may have skewed each algorithm to classify malignant instances more poorly while also skewing the outcome towards benign classifications. Additionally, the complexity of the SVM, Neural Networks, and Random Forest models, in contrast to the KNN model, may have skewed the analysis of the accuracy of graph-based methods, causing the KNN to comparatively perform poorly on the high dimensionality dataset. We plan to extend this research by including more multi-class datasets and by testing on different graph-based algorithms to confirm our findings.

MATERIALS AND METHODS

Analysis of cancer dataset

The initial evaluations of the five algorithms were conducted across Wisconsin Breast Cancer Dataset (7, 11) consisting of 569 elements, each with 30 features associated with statistical attributes of patients' cells. Each element was classified into one of two classes, malignant or benign. The shape and structure of a cell nucleus is important for determining how a cell is working. If a cell has a change in its nuclear shape or texture, this can indicate a switch from a normal to a cancerous cell. These changes are usually detected by human pathologists, and identification of normal and abnormal nuclear shapes and structures can heavily aid early cancer screening. On the breast cancer dataset, the SVM used a linear kernel. The MBO adaptation utilized a C value of 51, time step of 0.002, eigenvalues of 211, sigma value of three, and number of nearest neighbors as 6. The neural network took in an input layer of 30, with a hidden layer of 15 neurons, a hidden layer of 10 neurons with a 'ReLu' activation function and an output layer of one. The Neural Network was run across 15 epochs. Using SKLearn, the Random Forest operated across 4 iterations of varying numbers of estimators. The KNN was averaged across three iterations of three, five, and seven nearest neighbors.

Analysis of banknote dataset

The second dataset tested was a data banknote authentication dataset (5) consisting of 1,372 elements, each taking on attributes from a set of forged and genuine banknote-like species, for a total of four features. These features were then mapped to one of two classes - genuine or forged. The binary classification dataset, the banknote dataset, included statistical attributes associated with over 1,300 (400x400 pixel) images of genuine and counterfeit banknote images. Counterfeit and genuine banknotes differ in the quality of fibers throughout the notes, and accurate computer algorithms can aid ATMs and bank executives in the banknote verification process. Similar to the previous dataset, the SVM used a linear kernel. The Neural Network algorithm took an input layer of four neurons and a direct output layer of one neuron, running across 25 epochs. The MBO Adaptation used a C value of 250, time step of 0.003, number of eigenvalues of 200, sigma of 1.25, and a nearest neighbors value of six. The Random Forest Algorithm ran across four different iterations on a varying number of nearest estimators, and the KNN was averaged across three iterations of three, five, and seven nearest neighbors.

Analysis of letter recognition dataset

The third dataset tested was the letter recognition dataset (6), consisting of 20,000 elements. The dataset took in 16 features relating to a set of handwritten letters and assigned them to one of the 26 letters in the English alphabet. By detecting characters and strings of characters, letter detection software can decrease labor costs by aiding the usability of interactive user interfaces and translation software. The SVM took in a linear kernel. The Neural Network algorithm took in an input layer of 16 with a 'ReLu' activation, a hidden layer of eight neurons with 'ReLu' activation, and an output layer of one neuron. The Random Forest algorithm ran on an average of four iterations of varying nearest neighbors values, 45, 50, 55, and 60. The MBO adaptation utilized parameters corresponding to a time step of approximately 0.0003, an eigenvalue count of 250, and a sigma value of 1.25. The KNN model averaged accuracies across three iterations of three, five, and seven nearest neighbors. All three of these datasets were taken from the UC Irvine ML repository. Except for the MBO adaptation, we coded all methods using python 3.7.10, Keras, Sci-kit learn, and Libsvm packages (12-15).

Received: January 13, 2021
Accepted: June 7, 2021
Published: September 11, 2021

REFERENCES

1. Liu, Yufeng, and Ming Yuan. "Reinforced Multicategory Support Vector Machines." *Journal of Computational and Graphical Statistics* 20, no. 4 (2011): 901–19. https://doi.org/10.1198/jcgs.2010.09206.
2. Metz, Cade. "Google Researchers Are Learning How Machines Learn." The New York Times. March 6, 2018. https://www.nytimes.com/2018/03/06/technology/google-artificial-intelligence.html.
3. Garcia-Cardona, Cristina, Ekaterina Merkurjev, Andrea L. Bertozzi, Arjuna Flenner, and Allon G. Percus. "Fast Multiclass Segmentation Using Diffuse Interface Methods on Graphs," 2013. https://doi.org/10.21236/ada580102.

4. Valput, Damir. "Machine Learning with Graphs: the next Big Thing?" Datascience.aero, January 13, 2021. datascience.aero/machine-learning-graphs/.
5. Volker Lohweg and Helene Doerksen. *UCI machine learning repository*, 2013.
6. David J Slate. *UCI machine learning repository*, 1991.
7. Street Nick W. Mangasarian Olvi L. Wolberg, William H. *UCI machine learning repository*, 1995.
8. Hardesty , Larry. "Explained: Neural Networks." *MIT News*. Massachusetts Institute of Technology, 2017. news.mit.edu/2017/explained-neural-networks-deep-learning-0414.
9. "Sklearn.ensemble.RandomForestClassifier." *scikit*. Accessed May 14, 2021. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.
10. "1.6. Nearest Neighbors." *scikit*. Accessed May 14, 2021. https://scikit-learn.org/stable/modules/neighbors.html.
11. W.N. Street, W.H. Wolberg and O.L. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on Electronic Imaging: *Science and Technology*, volume 1905, pages 861-870, San Jose, CA, 1993.
12. Fchollet. "Fchollet/Keras-Resources." GitHub. github.com/fchollet/keras-resources.
13. G. van Rossum. Python tutorial. Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, May 1995.
14. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blon-del, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. *Scikit-learn*: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
15. Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.