Meausuring the accuracy of Naïve Bayes and Neural Networks on Gisette dataset

before and after dimensionality reduction

Grammenou Sotiria
School of Science and Technology
International Hellenic University Thermi, Greece
s.grammenou@ihu.edu.gr

Patrikios Georgios
School of Science and Technology
International Hellenic University Thermi, Greece
g.patrikios@ihu.edu.gr

Abstract—This study contains the findings regarding the project of Machine Learning course at International Hellenic University. The aim of this study is to measure the accuracy between Naïve Bayes and Neural Networks classifiers before and after dimensionality reduction on Gisette dataset. According to our findings, the accuracy of both classifiers is improved by using a significant lower dimensional representation of the original data.

Keywords—naïve bayes; neural networks; dimensionality reduction; principal component analysis; gisette; handwritten digit recognition

1. Introduction

Handwritten digit recognition (HDR) belongs to the family of Image recognition problems. Specifically, in HDR the goal is to separate highly confusable digits. At present, machine learning algorithms can achieve this goal with a relatively high accuracy almost like humans do.

In this study Naive Bayes and Neural Networks are trained on 'Gisette' dataset. Accuracy is used as an evaluation metric for both classifiers. Furthermore, scores are obtained before and after dimensionality reduction. The dataset describes a handwritten digit recognition problem. Specifically, is a two-class classification problem where the target is to separate '4' from '9'. This dataset is one of five datasets of the NIPS 2003 feature [1].

Moreover, two data files are given regarding 'Gisette' dataset for this study. A set of 6000 patterns and 5000 features constitutes the training set accompanied by its label set. In detail, the training set contains sparse continuous input variables. The digits have been size-normalized and centered in a fixed-size image of dimension 28x28. The label set is perfectly balanced containing 3000 targets for each of the two classes. In addition, the label set contains one binary feature in order to describe the two classes 4 and 9.

2. Description of models

Two classifiers are used in this coursework, Naïve Bayes (NB) and Neural Networks (NN). The first classifier, NB constitutes a probabilistic classifier inspired by the Bayes theorem under a simple assumption which is the attributes are conditionally independent. This method is specifically popular because of its simplicity and its generally good results. There are several types of Naïve Bayes classifiers regarding their distribution. The most commonly used are Gaussian NB, Multinomial NB and Bernoulli NB. All of them share the common principle that every pair of features being classified is independent of each other. Multinomial NB assumes that features have multinomial distribution which is a generalization of the binomial distribution. The assumption of the binomial distribution of features holds also for Bernoulli NB. Neither binomial nor multinomial distributions can contain negative values. For this reason, the last two types of Naive Bayes are not used since after Principal Component Analysis the transformed features contain negative values. Only Gaussian NB is used in this study since this classifier can handle negative values in feature's distribution [2].

Furthermore, the second classifier used in this study are Neural Networks (NNs). Generally, NNs can approximate complicated non-linear mappings and are commonly used for classifications and predictions. Moreover, they have long training time since a large number of parameters must be adjusted.

There are various types of Neural Networks, regarding the most popular is the multilayer perceptron (MLP), Convolutional neural network, Recursive neural network, Recurrent neural network and Long short-term memory [11]. This study uses a multilayer perceptron (MLP). It uses a nonlinear activation function in order to classify data that are not linearly separable. Every node in a layer connects to all nodes in the following layer making the network fully connected.

Moreover, the MLP used in this study is structured in three main layers, the input, the hidden and the output layer. The input layer contains the input neurons that send information to the hidden layer. The hidden layer consists of twenty neuros. Finally, the output layer which classifies the data. The connections in Neural Networks are associated with a set of weights and biases between each layer and the activation function in the neurons.

Regarding the learning phase, indirect optimization is used which is a process of searching for parameters that minimize or maximize the functions. In this study 'Adam' optimizer is used. Rectified Linear Unit (ReLU) is used as activation function. ReLU has a linear identity for all positive values and zero for all negative values, where it must be dealt with using approximations since they invalidate the probabilistic. ReLU's function simplicity leads it to be a relatively low-cost function to compute and rectifies vanishing gradient problem [4].

Principal component analysis (PCA) is a dimensionality reduction technique. The main idea of (PCA) is to reduce the dimensions of a dataset consisting of many variables that may be correlated with each other. In other words, PCA is used to eliminate useless repeated features. It originally stems from the field of linear algebra. The mathematics behind principle component depends on standard deviation, eigenvalues and eigenvectors [12].

2.1 Approach

1. First step is to see the accuracy score for the two classifiers. For this reason, 'scikit-learn.org' library is used. The accuracy is obtained by measuring the mean of accuracies obtained using a 20 folds cross-validation.

2.In the second step, a principal component analysis takes place. The explained variance for each principal component is plotted in order to determine the number of components which preserve the variance of the original data. Obviously, 5000 components hold 100% of the data variance.

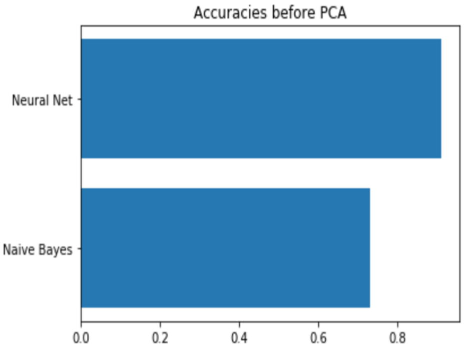
3. After the selection of the number of principal components, the two classifiers are retrained using the new datasets obtained after PCA. Again, using a 20 folds cross-validation, the accuracy is measured.

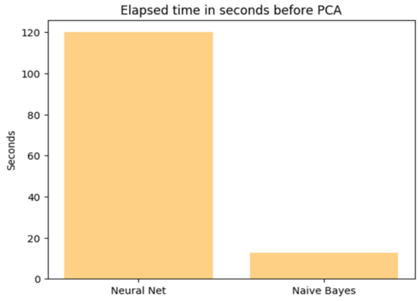
4. Finally, an in-depth analysis of the number of the principal components takes place since it is observed that the accuracy rises as the number of principal components decreases.

3. Comparative Experiments and Results

1. As mentioned above, the accuracies of the two classifiers are obtained using 20 folds cross-validation.

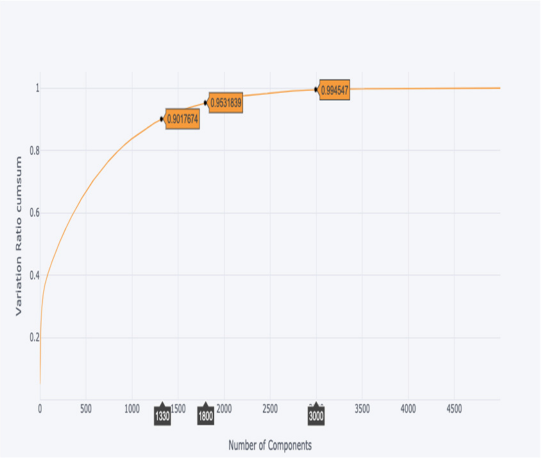
The scores before PCA are presented in the following bar chart. Specifically, Neural Networks score 91.19% while Naive Bayes classifier scores 73.24% in terms of accuracy.





As it is shown above, Neural Networks are performing higher in terms of accuracy. On the other hand, for Naive Bayes classifiers the time elapsed is 12.6 seconds while for the Neural Networks is 120 seconds.

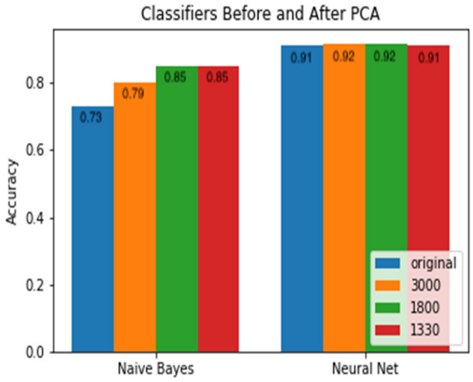
2. In this step, the explained variance for each attribute will be calculated in order to determine the number of principal components.



The plot denotes that for 3000 principal components 99.5% of the total variance of the original data is preserved. Additionally, 95% is preserved for 1800 principal components and 90% for 1330. Considering the above, three different datasets are created using PCA for 3000, 1800 and 1330 components.

3. After creating the new datasets, the classifiers are trained on the new data in order to measure their accuracy. The following visuals show the accuracies measured for the two classifiers before and after dimensionality reduction.

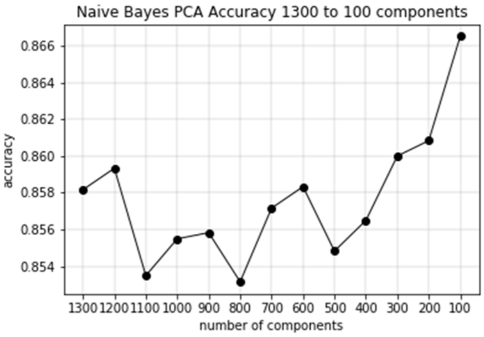
	Accuracies	
	Naive Bayes	Neural Net
Original data before PCA	0.733	0.912
3000 principal components	0.799	0.917
1800 principal components	0.85	0.916
1330 principal components	0.852	0.911

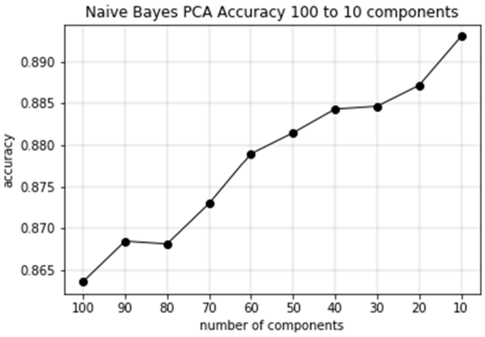


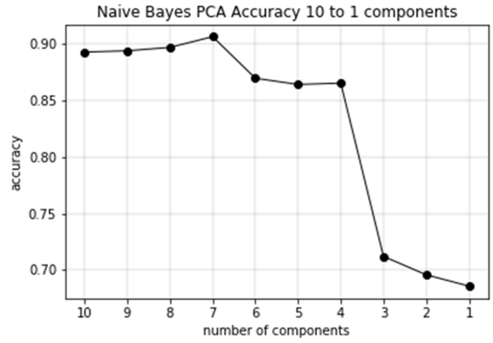
Overall, Neural Networks score a higher accuracy than Naïve Bayes classifier. In detail, NNs' highest accuracy is 91.7% using 3000 principal components while for Naïve Bayes is 85.2% using 1330 principal components. Furthermore, NNs' accuracy is 91.4% with 0.3% declination. This means that the classifier's accuracy it is not affected by the change in the number of principal components. On the other hand, the Naïve Bayes classifier's accuracy rises as the number of components decreases. For this reason, in the next step an extensive analysis takes place in order to determine the optimal number of principal components which maximizes the accuracy for "Gisette" dataset.

4. In the last step, an extensive analysis takes place aiming to determine the optimal number of principal components for each of the two classifiers (NB, NN). Specifically, for both classifiers the accuracy is measured for various ranges of principal components. The ranges are formed this way in order to reduce the computational cost. Firstly, the accuracy is measured for the range of 1300 to 100 components by decreasing each time by 100 components. Then, the accuracy is measured for the range of 100 to 10 by decreasing 10 principal components and finally for a range 10 to 1 by decreasing 1 component each time.

Below are presented the results for Naïve Bayes Classifier. Three plots are created according to the process mentioned above. Note that, y axis represents the accuracy while x axis the number of components.

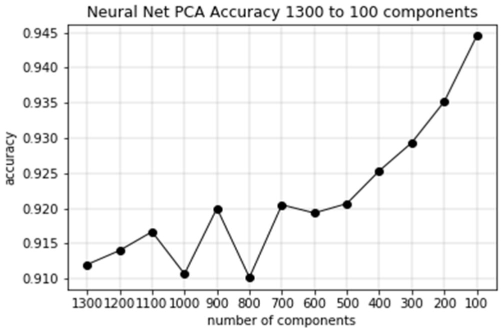

Naive Bayes PCA Accuracy 1300 to 100 components

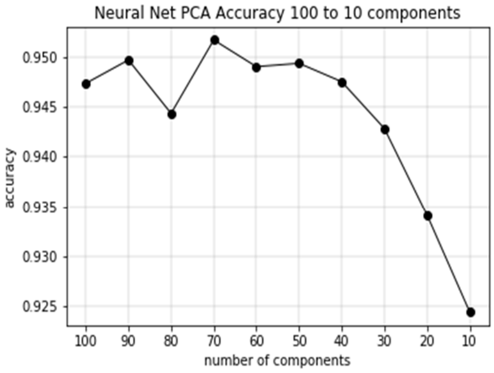

Naive Bayes PCA Accuracy 100 to 10 components

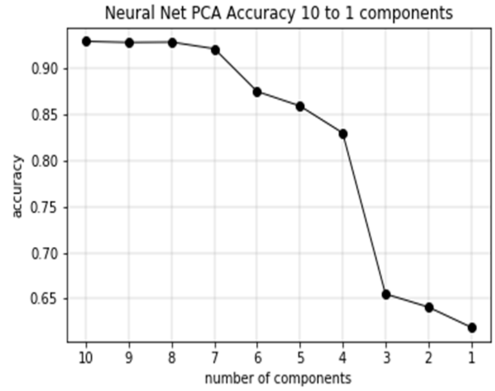

Naive Bayes PCA Accuracy 10 to 1 components

It is observed that while the number of principal components is decreasing the accuracy is rising until 7 components where maximum accuracy is reached 90.7%.

Regarding the Neural Networks classifier, the same steps are followed producing the plots below.


Neural Net PCA Accuracy 1300 to 100 components


Neural Net PCA Accuracy 100 to 10 components


Neural Net PCA Accuracy 10 to 1 components

For this classifier the optimal number of principal components is 70 where 95.2% accuracy is scored. While decreasing from the 70 components accuracy decreases as well.

The following table denotes the accuracies obtained for two classifiers before and after dimensionality reduction. Overall, Neural Networks outperform Naïve Bayes classifier. It is worth mentioning that for 7 principal components the difference between the two classifiers in accuracy is 0.013.

	Accuracies	
	Naive Bayes	Neural Net
Original data before PCA	0.733	0.912
3000 principal components	0.799	0.917
1800 principal components	0.85	0.916
1330 principal components	0.852	0.911
70 principal components	0.873	0.952
7 principal components	0.907	0.92

4. Conclusions

The aim of this study is to train two different classifiers, Naive Bayes and Neural Networks, on 'Gisette' dataset and measure their accuracy before and after dimensionality reduction.

Before dimensionality reduction, Neural Networks outperform Naive Bayes classifier in terms of accuracy. After performing principal component analysis, it is observed that the classifiers reach their highest accuracy by using different number of principal components. Specifically, Naive Bayes' optimal number of components is 7 while for Neural Networks is 70. Although, the accuracy of both classifiers is improved by using PCA, for Naive Bayes there is a significant improvement of 17%. Respectively, for Neural Networks there is a 4.4% improvement in accuracy. Principal component analysis reduces the high dimensionality on 'Gisette' dataset. In detail, from 5000 features it proven that the optimum number is 7 and 70 for Naive Bayes and Neural Networks classifier respectively. Other studies regarding 'Gisette' dataset have found similar results [5] [7]. PCA provides a low-dimensional representation of data which cannot only maximize performance metrics but also minimize computational cost.

5. References

[1] Isabelle Guyon (2003). Design of experiments for the NIPS 2003 variable selection benchmark. Retrieved from archive.ics.uci.edu

[2] Kostas Diamantaras (2019). The Naïve Bayes Classifier [Notes]. Retrieved from elearn.ihu.edu.gr

[3] Data Monsters (2017). 7 types of Artificial Neural Networks for Natural Language Processing. Retrieved from medium.com

[4] Jason Brownlee (2019). A Gentle Introduction to the Rectified Linear Unit (ReLU). Retrieved from machinelearningmastery.com

[5] Jainendra Upreti (2017). Principal Component Analysis - [Notes]. Retrieved from rstudio-pubs-static.s3.amazonaws.com

[6] A. Gupta (2016). Neural Networks in Data Processing. Retrieved from ijarcet.org

[7] Principal Component Analysis. Retrieved from www.dezyre.com

[8] Gavril Ognjanovski (2019). Everything you need to know about Neural Networks and Backpropagation Retrieved from towardsdatascience.com

[9] Kostas Diamantaras (2019). Cross-Validation Retrieved from elearn.ihu.edu.gr

[10] Sigmoid function (2020). Retrieved from en.wikipedia.org

[11] SuperDataScience Team (2018). Convolutional Neural Networks Retrieved from superdatascience.com

[12] H. F. Aamir Khan (2011). Principal

Component Analysis-Linear Discriminant

[13] S. Wold (1987). Principal Component Analysis