Logistic Regresion and Kernel PCA

ABSTRACT

PURPOSE: Understand how well a single artificial neuron performs in learning and predicting whether a US college institution is public or private, and performing clustering on Fisher's Iris data for species I. setosa.

METHODS: Implementing the Perceptron Rule for low-dimensional data and comparing the accuracies of separating hyperplanes with Logistic Regression. Utilizing kernel PCA method for clustering and evaluating two different classifications of one data set.

RESULTS: Logistic Regression has a higher prediction accuracy and AUC value compared to Perceptron Rule. Kernel PCA with a Gaussian kernel function successfully classifies the dataset.

CONCLUSION: Artificial neuron performs well in classify binary labels, though its accuracy is a little less compare to Logistic Regression, on U.S college dataset. Using kernel PCA with a Gaussian kernel function is an effective approach to find clusters that accurately match the data labels.

Word count: 129

INTRODUCTION

The objective is to evaluate two different classification methods, between Perceptron Rule and Logistic Regression, and implement Kernel PCA for clustering purposes.

Perceptron Rule is a basic algorithm used to find weight vectors, a version of an artificial neuron developed by Frank Rosenblatt.[1] The algorithm is based on the idea that the hyperplane will go on constant changes while there are still misclassifications up until a certain iteration. In application, the update happens when there is a false negative or a false positive produced by the current weight vector during prediction. The Residual error terms are used to determine the direction of misclassification made by the current weight vector. The residual error is written as $e_i = y_i - q_i$. The error vector stores the differences when the true label subtracts the quantization of the scores. The error terms belong to one of the three values {-1, 0, 1}, where -1 correlates to a False Positive, 0 correlates to a correct prediction, and +1 correlates to a False Negative. When facing a case of False Positive, the weight vector at observation I become farther from data vector x of observation i. When facing a case of False Negative, the weight vector at observation i become closer to data vector x of observation i. This study will attempt to implement the Perceptron Rule in Batch Learning, which implements the basics into a structure in linear algebra. With m observations from the full data matrix, an augmented design matrix can be created, adding an extra column of ones. The observation labels are also gathered into a label vector y, the same idea applies to a quantization vector q. This approach allows all the observations to be calculated simultaneously at i iteration, as $\widehat{w}_k \leftarrow \widehat{w}_{k-1} + \hat{X}^T(\vec{y} - \vec{q}\,))$ [1].

Logistic Regression is an approach to map linear summation to probability, considering a nonlinear activation function that is continuous and differentiable. [2] The logistic activation function for neural networks is sometimes called a sigmoid activation function. When performing binary classification of observations, the logistic function also accumulates residual errors into a single scalar value that can be optimized.

Kernel PCA is the use of the Gram matrix for a scatter matrix when performing PCA. The first step to implement kernel PCA is to compute the Gram matrix, using a predefined kernel function, from the observations of the X data matrix. Then, we can compute the centered MxM Gram matrix for the data. Finally, we can compute the kernel PCA, using the spectral decomposition of the Gram matrix. This method sorts the eigenvalues and eigenvectors in descending order and projects the Gram matrix onto a specified dimension.

The 2 datasets used in this study are a MATLAB built-in dataset on Iris flowers and a US college statistics dataset from the 1995 issue of US News and World Report. The Iris flower dataset is a multivariate dataset increate by biologist Ronal Fisher in his 1936 paper.[3] The U.S. News and World Report's College Data is from the StatLin library at Carnegie Mellon University that went through processing to replace "yes/no" encoding with numerical codes.

This study attempts to answer the scientific question of the effectiveness when using a single artificial neuron perform in learning and predicting classifying whether a US college is public or private, and performing clustering on Fisher's Iris data for species I. setosa. The results are then plotted and portrayed using ROC curve to measure the efficiency.

METHODS

This study produces an evaluation by comparing two classifications of data for two different datasets. For the data for post-secondary educational institutions, the approach implements Perceptron Rule for low-dimensional data, as well as computing the accuracies of separating hyperplanes for classifiers. For the well-known Iris dataset, the method implements a kernel PCA method of clustering to classify the data.

The method for the first given task is evaluating the performance of Perceptron Rule and Logistic Regression on post-secondary institutions dataset. With the university dataset, the second column classifying whether the school is public or private is used as the label vector for the classification task. The data matrix is then augmented with a 1's vector for the Perceptron initialization step.

The function a5q1 calls to sepbinary function to perform the computation of Perceptron Rule using the 'learning rate' eta. For the number of maximum iterations predefined within the function (imax = 10000), the for loop is implementing the Perceptron Rule Batch Learning approach. The m observations are gathered into an augmented design matrix. The observation labels are gathered into a label vector. The quantization scores are gathered into a quantization vector q. When working with all the variables simultaneously through each iteration, the weights update by being shifted towards the misclassified scores. The learning rate is being implemented to ensure that the shift of this hyperplane happens smoothly, and the moves are not too drastic, eta is set at 0.001. An estimated weight vector and a scalar number of iterations used are returned from the function.

The a5q1 function then displays the ROC curves and the accuracies of the hyperplanes generated by Perceptron Rule and Logistic Regression. The accuracy can be calculated by dividing all correct guesses (TP + TN) by all predictions. To find the sum of all True Positive plus True Negative, we first compute the quantization vectors for both approaches. This will allow the calculation of the error vector, which stores the difference between the true labels to the prediction results. Within the vector error, the number of times the 0 value appears to equate to the number of times the predictions are correct. Thus, counting up all the 0's in the two error vectors will give us the total correct guesses for the two methods. Then, taking the two TP+TN values over the total number of predictions, we get the accuracy of each method respectively.

The method for the second given task is to use a kernel PCA method to reduce the data matrix to 2D for k-means clustering on the Iris dataset. Through this, we are computing a Gram matrix for a Gaussian kernel function to implement kernel PCA.

The function a5q2 calls to gramgauss, which computes the Gram matrix for the data in Xmat. Within gramgauss, modifications are applied onto the default Gram matrix using the Gaussian exponential exp(-1/sigm28norm(X_i -X_j)^2). This is done by looping through the predefined Kmat matrix at every row and column and change the entry respectively. Then this Gram matrix's spectral decomposition is used by Kernel PCA. With that, we attempt to sort the eigenvalues and eigenvectors in descending order. The first two vectors are then used to project the Gram matrix to a 2D projection.

This study works with the built-in Iris dataset in MATLAB, with 150 instances and 4 attributes, and the "collegenum.csv" dataset, with 777 observations on 18 variables.

To evaluate the results from the study, the AUC values along with hyperplane prediction accuracies are displayed in the console for Perceptron Rule and Logistic Regression. For the Perceptron Rule, the AUC value is 0.9495 and the hyperplane accuracy is 0.8764. For the Logistic Regression, the AUC values if 0.9600, and the hyperplane accuracy is 0.9112. The calculation results show that Logistic Regression has better predictions than Perceptron Rule.

RESULTS

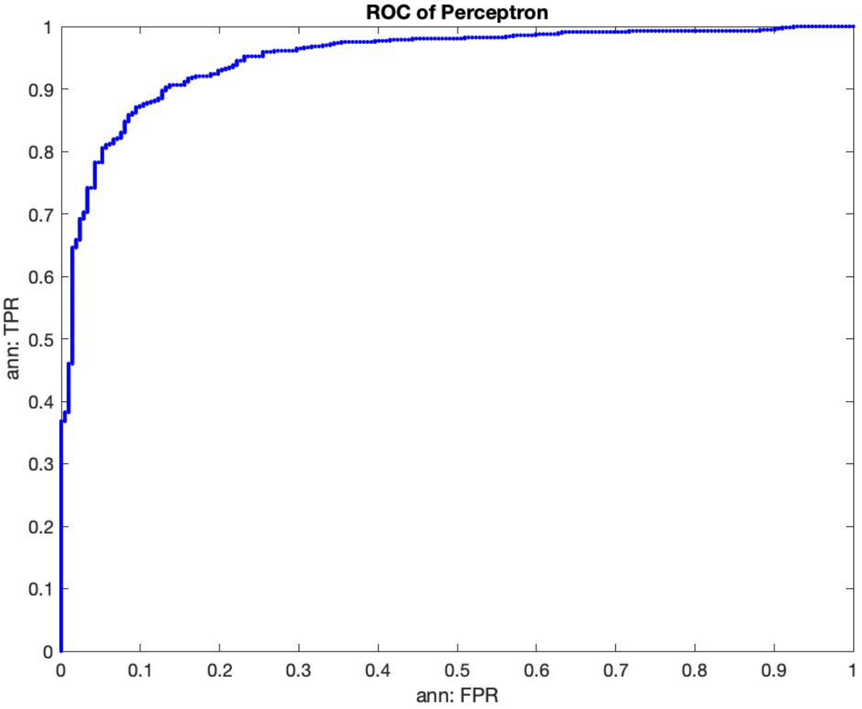

Figure 1: Plot of ROC with Perceptron Rule approach. The figure demonstrates the relationship between clinical sensitivity (True Positive Rate) and specificity.

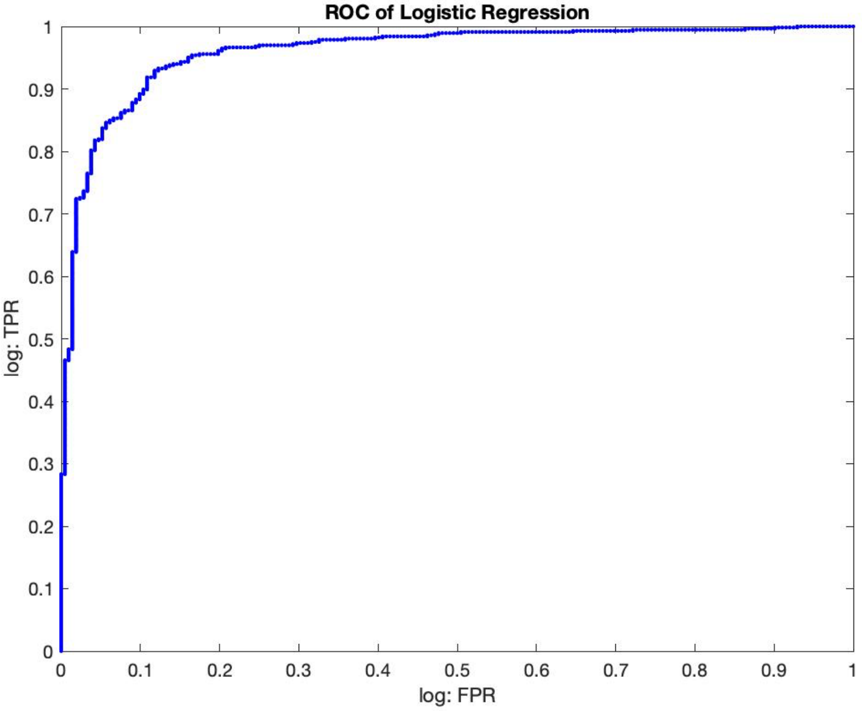

Figure 2: Plot of ROC with Logistic Regression approach. The figure demonstrates the relationship between clinical sensitivity (True Positive Rate) and specificity.

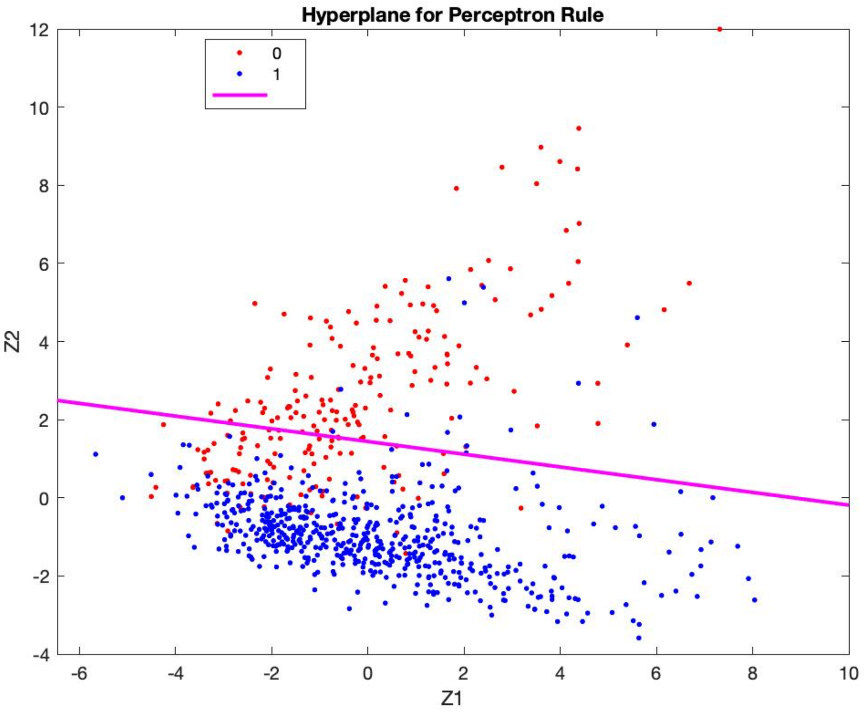

Figure 3: Plot of data and hyperplane generated by Perceptron Rule. The figure demonstrates linear separability of the two class of private and public U.S. post-secondary institutions.

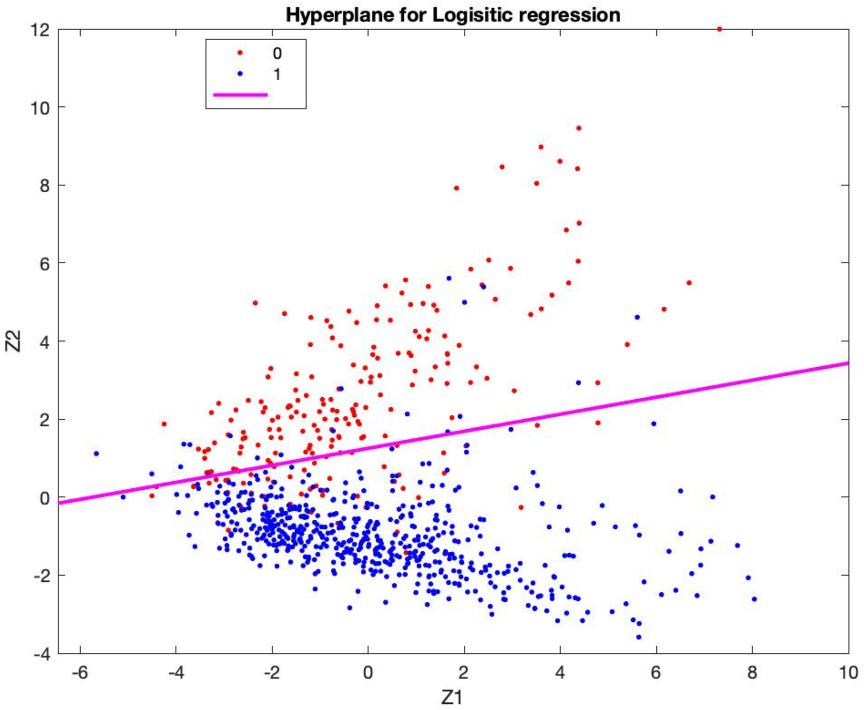

Figure 4: Plot of data and hyperplane generated by Logistic Regression. The figure demonstrates linear separability of the two class of private and public U.S. post-secondary institutions.

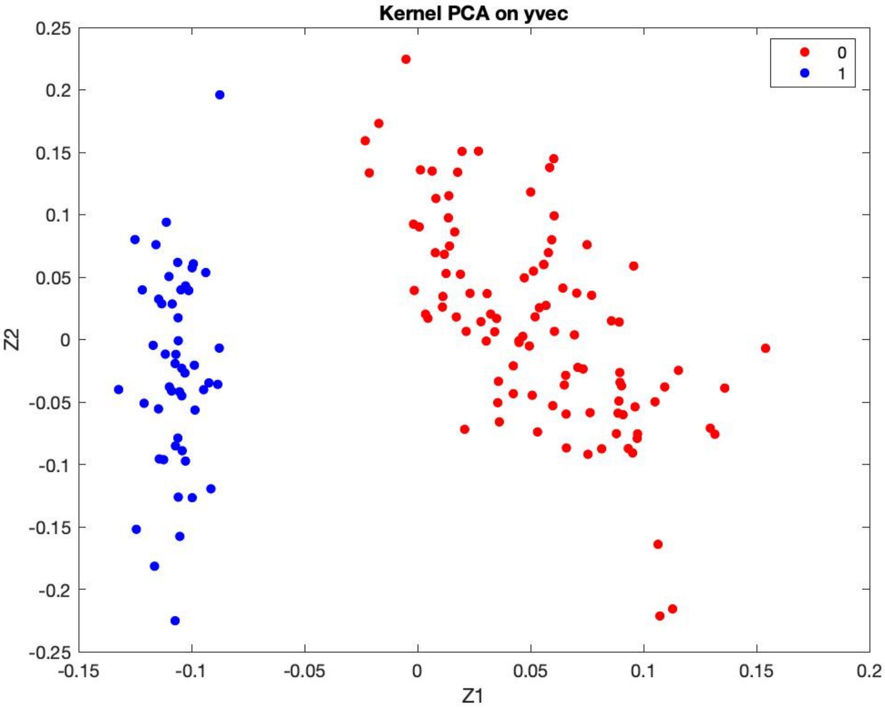

Figure 5: Plot of Iris data in two separate figures, encoding 0 as "red" and 1 as "blue". Fisher's Iris data set when processed usign kernel PCA.

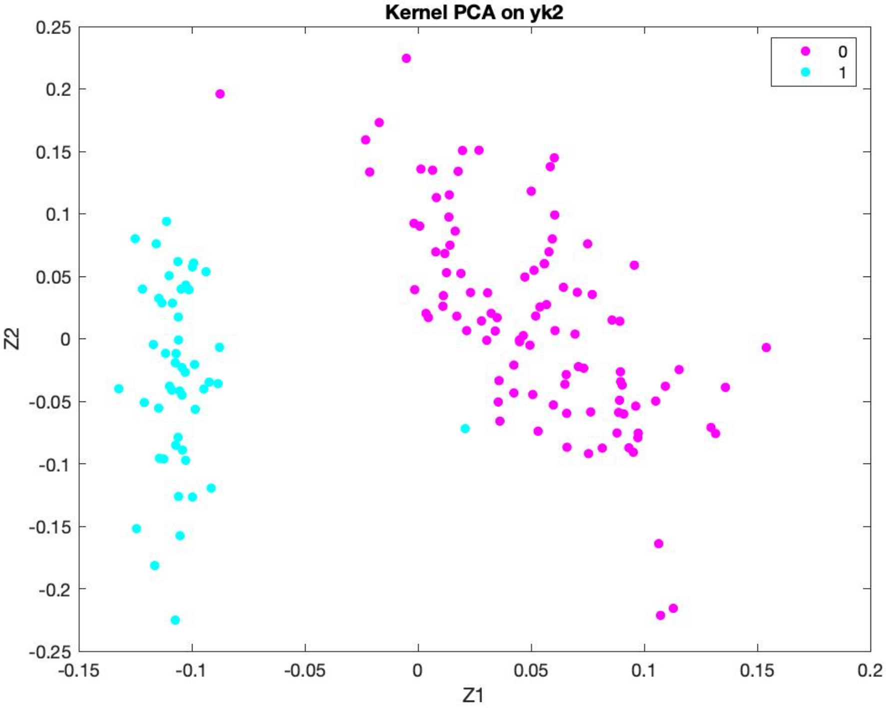

Figure 6: Plot of Iris data in two separate figures, encoding 0 as “magenta” and 1 as “cyan”. Fisher’s Iris data set when processed usign kernel PCA.

DISCUSSION

For our first task, the result proves that artificial neuron is robust in classification task while the performance of Logistic is better than Perceptron Rule, there is not a substantial gap between the accuracies or AUC score. Upon inspecting the first two AUC curves, Figure 1 and Figure 2, of Perceptron Rule and Logistic Regression, both demonstrate high compatibility with AUC scores 0.9495 and 0.9600 respectively. Furthermore, inspecting Figure 3 and Figure 4 with 2D PCA reduced data and hyperplanes, we can see the linear separation between the two classes. With this representation, we can see how the Logistic Regression hyperplane outperforms Perceptron Rule. Red data points can concentrate more on the upper separation created by the hyperplane. While with Perceptron Rule, there is a considerable number of red data points misclassified in the bottom separations. This observation is further explained with the hyperplane accuracies, 0.8764 for Perceptron Rule and 0.9112 for Logistic Regression. Based on the first four figures, one simple artificial neuron still manages to demonstrate a great capacity to predict binary classification. Thus, with more neurons and layers, the accuracy can greatly improve for the neural network approach.

For the second task, the kernel PCA with Gaussian kernel seems to be an appropriate choice for finding clusters that match the data labels. Observing Figure 5 and Figure 6, The data scored in 2D with results of k-means clustering shows that all data vectors are correctly clustered.

In real-world applications, there are extensive applications in areas where the system needs to learn and adapt to incoming data in sync. Neural networks focus on extract meaning from complex and imprecise datasets to detect certain patterns and trends for the prediction task. The greatest result of training a neural network is its ability to recreate training and learn on its own for future predictions. These are the building blocks for many complex applications in information processing. Besides binary classification tasks as we see in this study, neural networks possess astonishing attributes that make them the solution for many complex prediction problems. A few outstanding advanced attributes in neural networks are Self Organization, Prognosis, or Fault Tolerance.[5] For Self-Organization, the ability to cluster and classify a variety of data allows artificial neural networks to tackle complicated visual problems in medical image analysis or even real-time operation. With Prognosis, neural networks can detect patterns in unseemingly places, such as weather, earthquakes, even fire movements. For Fault Tolerance, neural networks are equipped to fill in the blanks even when significant parts of the process are missing.

REFERENCES

[1] Ellis Randy E. Class 28: PCA Classification – Perceptron Rule. [unpublished lecture note]. CISC 271: Linear Data Analysis, Queen’s University; lecture given March 2021

[2] Ellis Randy E. Class 30: PCA Classification – Logistic Regression. [unpublished lecture note]. CISC 271: Linear Data Analysis, Queen’s University; lecture given April 2021

[3] Ellis Randy E. Class 32: Nonlinear Separation – Kernel PCA. [unpublished lecture note]. CISC 271: Linear Data Analysis, Queen's University; lecture given April 2021

[4] UCI Machine Learning Repository: Iris Data Set. [cited 2021Apr16]. Available from: https://archive.ics.uci.edu/ml/datasets/iris

[5] Real-Life Applications of Neural Networks [Internet]. Smartsheet. [cited 2021Apr16]. Available from: https://www.smartsheet.com/neural-network-applications