STAT 639: Data Mining & Analysis

FINAL PROJECT

HERIN SAVLA (931002085), PRITHV SINGH PRADHAN (731008352), SMRITI SINGH(630009456)

SUPERVISED LEARNING TASK

Goal: The goal of the first task is to classify a dataset containing 500 predictors (p), 1000 observations (n) into two classes (0 and 1) based on the training data provided where n=400.

Approach: We started with checking for missing values, class imbalance, collinearity and zero variance across the entire dataset. We tried different feature selection techniques such as Lasso and Boruta to get rid of unwanted noise which would negatively impact any model built. We simulated all the models with 5-fold cross-validation for hyperparameter tuning and model validation. The metrics used for model evaluation were misclassification rate and AUC score.

kNN: We modeled kNN with Boruta reduced scaled dataset and PCA reduced dataset. The value of controlling parameter *k* was set as the size of the predictor space and the mean validation accuracy for each value of *k* was calculated. We obtained *k*=43 with PCA, *k*=6 without PCA. Without PCA, the model obtained the lowest error rate of **24.75%.**

SVM**:** After scaling the dataset, we used the tune.svm function to find optimum values for parameters *gamma*-set smoothness to the decision boundary and control variance of the model and *cost*- penalty to control error. The *svm* model was fitted with the *radial* kernel as it performs well with non-linearly separable data.

Random Forest**:** The parameters tuned for building random forest were *mtry*- number of variables randomly sampled as candidates at each split and *ntrees*- number of trees to grow. ***Logistic Regression**:* After eliminating the highly correlated variables and identifying significant variables with p-value less than 0.01, *glm* model was fitted to the reduced dataset and CV error was calculated. LASSO and Ridge Regression was also used to minimize the complexity of the model and avoid overfitting. Cross validation was used to obtain the optimal value of the regularization parameter, lambda and glm model was built using the optimal value of lambda.

LDA & QDA: We identified the significant variables with p-value less than 0.01and fit the *lda* model on the reduced dataset. We chose the top 50 variables with the highest correlation with the dependent variable and fit the *qda* model on the reduced dataset.

Decision Trees: Pruning of decision tree was performed based on the value of stopping criteria, *CP (complexity parameter)* identified to build an optimal model. The value of *CP* was identified based on the minimum cross validated error obtained through plotcp() function.

Summary:

Classification Model	**Misclassification rate (Cross Validated)**	**AUC Score**
kNN	With PCA- 37% Without PCA- 24.75%	0.6223 0.6376
SVM	24.75%	0.7444
Random Forest	26.75%	0.7288
Logistic	40.5% 44.75% (LASSO & Ridge)	0.5858 0.5
LDA & QDA	40.5% (LDA) 43.25% (QDA)	0.5858 0.5476
Decision Trees	32.5% (Pruned)	0.6567

Conclusion: **kNN** (without PCA) with Boruta reduced dataset resulted in the lowest error of 23.25% would be the preferred classification model for this dataset .

UNSUPERVISED LEARNING TASK

Goal: An un-labelled data set consisting of 1000 observations and 784 variables was provided and the objective was to obtain an optimal value of groups or clusters, K.

Approach: We have applied methods like *K-means Clustering*, *Hierarchical Clustering*, *DBSCAN Clustering* and *PAM (Partitioning around Medoids)*. First, we studied the means and variances and concluded that it was important to standardize the given data to avoid the clustering measure being highly influenced by the columns with a higher scale. Next, we used PCA to reduce the dimension of the dataset into a subset of representative variables. We chose the top 152 PCAs using the '*kaiser criterion*' having *eigenvalues* greater than 1. For each clustering method, we have applied "*Within Sum of Squares*", "*Silhouette*" and "*Gap Statistics*" to get optimal values of K.

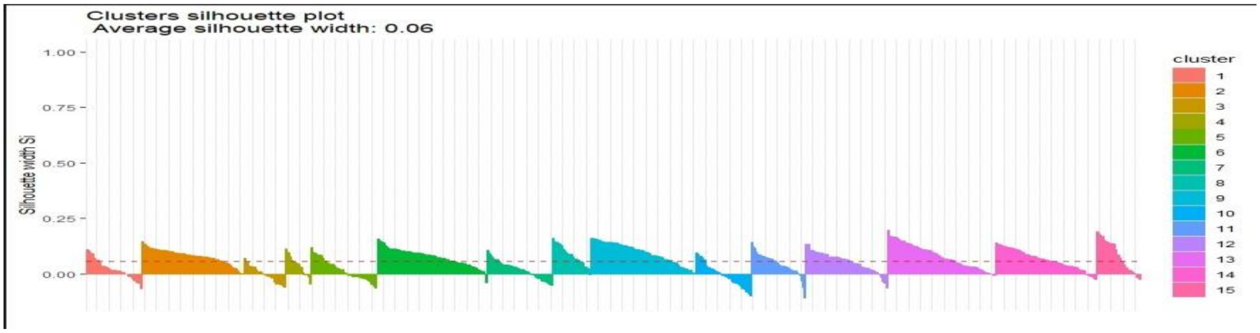
K-Means Clustering: Out of the optimal values that were suggested, we chose k=6 for K-means clustering based on the clustering output and between the sum of square percentages. We observed that the first two PCAs account for only about 17% of variances explained, and the performance was evaluated using '*Dunn index*' to be poor.

Hierarchical Clustering: Out of the optimal values that were suggested, we chose k=10 suggested by gap statistics along with *'Euclidean'* distance and *'complete'* method. We found that there were about 976 observations in cluster 1 only. The performance evaluated using '*Dunn index*' was not good.

Partitioning Around Medoids (PAM): PAM is very robust to outliers. Out of the optimal values suggested, we chose k=3 as it had the highest average silhouette width of 0.054, and the data points are evenly distributed among the clusters.

DBSCAN: We found the optimal value of *eps* to be 30 corresponding to the nearest neighbor as 2 and removed the noise from the data and refined it. We again searched for the best value of K on cleaned data. We chose K=15 as it has the best silhouette length and has the maximum between the sum of square percentage.

Conclusion: We choose **K=15** based on the DBSCAN method after removing the outliers and observe the data is evenly distributed across the 15 clusters. K- means coupled with DBSCAN is our preferred method of use. It is likely that plotting the clusters on a 2-dimensional graph won't give us a clear picture, but the other mathematical conclusions helped us to reach our determination.



Methods	Possible values of k	Chosen value of k
K means Clustering	2, 6, 11	6
Hierarchical Clustering	8, 10	10
PAM	2, 3, 16	3
DBSCAN	4, 8, 15	15

REFERENCES:

1. Support Vector Machine Example (rstudio-pubs-static.s3.amazonaws.com)
2. DBScan Clustering in R Programming - GeeksforGeeks