PROJECT REPORT

STAT 639 – DATA MINING AND ANALYSIS

SPRING 2023

Group 21

Divyansh Bokadia (UIN:733000614)

Rishabh Bassi (UIN:532008692)

Sankalp Chapalgaonkar (UIN:233005696)

SUPERVISED LEARNING

During the initial phase of data preprocessing, we conducted a thorough assessment to identify any missing values, and no such gaps were found. Subsequently, the data was standardized to ensure equitable weighting of all features during analysis. Following the scaling process, a meticulous examination was carried out to **detect highly correlated variables**, and redundant ones were prudently eliminated to mitigate the potential challenge of multicollinearity. Additionally, we explored various **feature selection algorithms** to identify the most important features in the data such as Principal Component Analysis (PCA), Lasso, Recursive Feature Elimination (RFE), and Boruta (we got best results for this one). Lasso was performed and optimal lambda was found out using Nested CV which was one standard deviation away from the mean. We chose not to use PCA finally as a feature selection technique because PCA modified the original dataset by transforming the features into new components, making the interpretation of results more difficult. In RFE, we tried multiple feature sizes and applied cross-validation (CV) to evaluate the performance of the different subsets of features to prevent overfitting to the data and that the selected features are generalizable to new data. The optimal feature subset ended up in the range of 160-180 features, which suggests that these features were the most informative for the analysis and that additional features may not have contributed much to the performance of the model.

The variability in feature selection introduced by the randomization in Recursive Feature Elimination (RFE) led to different subsets of features being selected in each cycle, making it difficult to control and reproduce the results. To address this issue, we employed **Boruta**, to identify relevant features while accounting for the variability introduced by randomization.

Through this approach, we were able to identify **a set of 23 features** that were relevant for our analysis.

In the subsequent step, we applied the **Nested Cross-Validation (CV)** technique on our dataset, which comprises an outer loop of CV that partitions the data into training and testing sets, and an inner loop of CV that is utilized to optimize the hyperparameters of the model. We perform this using two methods : Caret Package and Manual Nested CV and then compare their results for getting the best stable configuration. We opted for a 10-fold split to perform the aforementioned process, which was used to appraise the efficacy of multiple classifiers, including Logistic Regression, LDA, SVM, Random Forest, Boosting model GBM and XGBoost model. The optimal hyperparameters were selected using a grid search approach. After tuning the hyperparameters using nested CV, we selected XGBoost as the best classifier based on its performance metrics. This process led to an **estimated Accuracy of 82.5%**. **The optimal parameters were interaction-depth = 9, eta = 0.1, number of rounds = 2000, lambda = 0.5 and gamma = 0.5 and early stopping rounds = 100.**

Method	Accuracy (FD)	Accuracy (RD)
Logistic Regression	53.33%	-
LDA	50%	-
SVM	50%	-
KNN	66.67%	-
Random Forest	68.5%	79%
Boosting (GBM)	65%	71%
Boosting (XGBoost)	**77%**	**82.5%**

Figure 1: Final Accuracy with Full Dataset(FD) and Reduced Dataset(RD)

UNSUPERVISED LEARNING

For unsupervised clustering problem, we implemented five clustering algorithms, K-means, hierarchical clustering, DBSCAN, Gaussian mixture models (GMM), and OPTICS algorithm on the given dataset that had 784 dimensions. Given the intricate nature of the high-dimensional dataset, its visualization and analysis posed significant challenges. Therefore, as a preliminary step, we conducted dimensionality reduction using Principal Component Analysis (PCA). Following this, we leveraged three distinct methods, including Silhouette Method, CH Index, and NbClust Method, to determine the optimal number of clusters.

To achieve dimensionality reduction, we employed PCA, and our findings indicated that using 187, 250, and 382 dimensions explained 90%, 95%, and 99% of the variance, respectively. By utilizing PCA, we were able to effectively decrease the complexity of the dataset, thereby facilitating the clustering process. The silhouette method was used to determine K in K means clustering algorithm. Silhouette scores measure how well each data point is clustered by calculating the similarity within the same cluster and the dissimilarity to the nearest neighboring cluster and it recommended the number of clusters to be 3 or 6, as these two options had the highest score as seen in Figure 2. NbClust provides 30 indexes for determining the optimal number of clusters in a data set. Figure 3 shows that 11 indices recommend 3 as the optimum number of clusters.

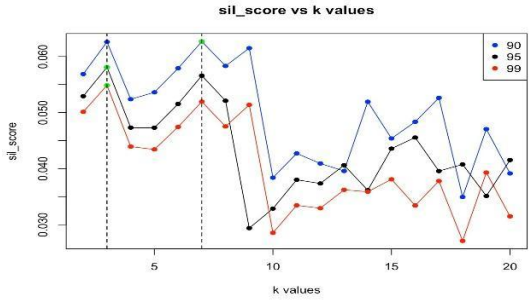

Figure 2: Silhouette score vs K

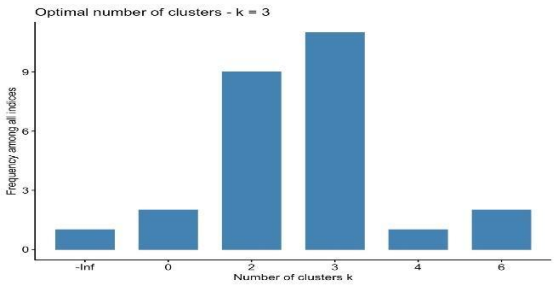

Figure 3: NbClust Result

On the other hand, the CH index is calculated using the between-cluster sum of squares and total within-cluster sum of squares. $CH(K) = \frac{BSS/K-1}{TWSS/n-K}$ We calculated the CH-index for different values of K and different PC spaces. The results were plotted in Figure 4, which shows the CH-index versus K for different PC dimension sets for hierarchical, complete linkage clustering.

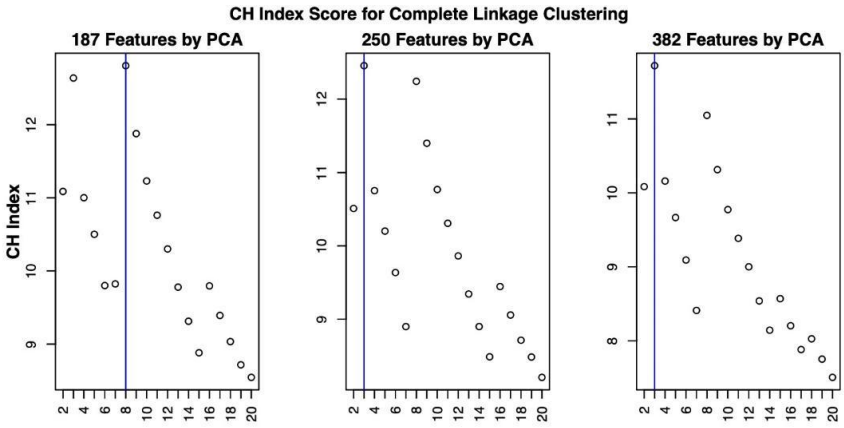

Figure 4: CH index for Complete Linkage Clustering versus k

We can see that the highest CH-index value was obtained for K=8 in the 187-dimension PC space, K=3 in the 250-dimension space as well as 382-dimension space. Based on the results from the silhouette method, CH index, and NbClust method, we found that the **Optimal number of clusters for the given dataset was 3**. Figure 5 shows the clusters formed using KMeans for K=3 and visualized in 2-D space where the x-axis and y-axis corresponds to PCA-1 and PCA-2 which accounts for 0.3% of the dataset each.

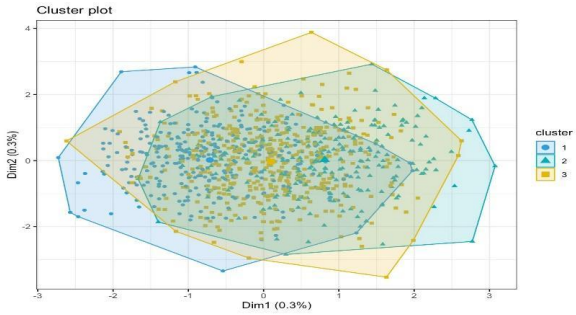

Figure 5: K-Means Clustering Output