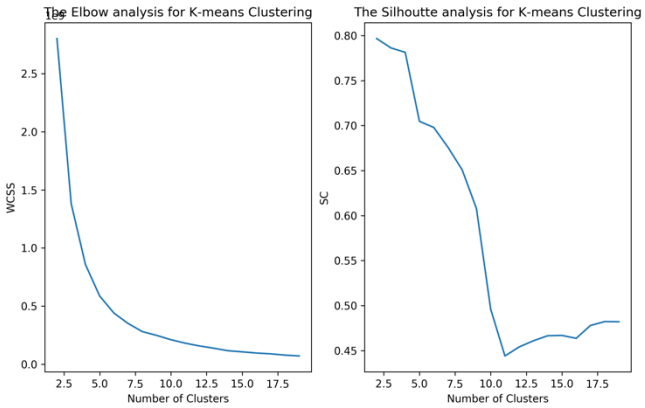
Jeanelle Abanto
1133815
JABANTO

For the first task, minor preprocessing was done on the data. Firstly, '..' entries in world data were replaced with *np.nan* for later imputation. The world data was then merged with the life data on country name and country code, which means that countries listed in world but not in life data were eliminated. After which, the features and class labels were selected from the merged data before splitting the data into 66% (2/3) training and 34% (1/3) test sets. After the split, missing values on the training and test sets were imputed using the median fitted on the training set. Then finally, the data was scaled and normalized before performing the classification algorithms with all other parameters of the decision tree and k-NN algorithms set to default. As a result, both the Decision Tree and 5-NN algorithms gave the same performance results with a prediction accuracy of 76.190%. However, the performance of the Decision Tree varies due to having not set the random state parameter, with a decreased accuracy score of 73.016% which is lower than the accuracy of 10-NN at 74.603%. Thus, for this experiment, in general the 5-NN algorithm gives the highest prediction accuracy.

For the second task, the same preprocessing and imputation methods were implemented on the dataset. In order to create a new feature using the class labels from K-means clustering on world data, K-means clustering with clusters ranging between 2 to 20 was applied on world data. This was done to find out the number of clusters using Elbow and Silhouette score analyses.



A point in the elbow with a relatively high Silhouette score was chosen to be the number of clusters. Based on the graphs, a cluster size of 5 was chosen for the K-means clustering, which was then fitted to the training set to predict the class labels of the test set.

The other part of feature engineering is the generation of interaction term pairs. This was done using the method *PolynomialFeatures()* with degree=2. Since this function will generate unnecessary polynomial features, the interaction term pairs were filtered out by selecting features with a maximum power of 1 and a sum of powers equal to 2. This means that for a generated polynomial $[a, b, a^2, ab, b^2]$, only ab will be selected wherein the power of both a and b is 1, and the sum of their powers is 2. Similar to K-means clustering, the interaction term pairs were

generated after the split of training set and test set. After generating the interaction term pairs and the class labels from K-means clustering, the generated features were concatenated to the original features on the training and test sets. The values are then normalized and scaled for the succeeding computations.

For the first implementation of 5-NN, features with correlation greater than 0.9 were eliminated leaving 58 features to choose from. Then, a combination of *SelectFromModel()* using *LogisticRegression* as an estimator, and *SelectKBest()* with k=4 were implemented. Using the first method, which selects features based on probabilities, the number of features was further reduced to 22. Methods ANOVA F, Chi-squared and Mutual Information were used on *SelectKBest()* to check which would give the best accuracy for the 5-NN algorithm, keeping the result of MI in general. Since the Chi-squared will cause an error if there are negative values, the training set was scaled using *MinMaxScaler()* during feature selection; the *StandardScaler()* was still used for the implementation of 5-NN. From the three methods, MI yielded the highest prediction accuracy of 84.127% for 5-NN classification using features [3, 10, 14, 19].

Among the three methods, feature engineering produced the highest prediction accuracy of 84.127% followed by the naïve feature selection and PCA with 74.603% and 73.016% accuracies respectively. On this experiment, the feature engineering yielded an improved prediction accuracy for the 5-NN algorithm, whereas PCA and naïve feature selection produced a lower accuracy similar to the accuracies of 10-NN and Decision Tree. Although the generated interaction term pair features may contain more information to achieve higher prediction accuracy, features having the same factor are more than likely to be highly correlated. Thus, after dropping highly correlated features, the information that can be obtained from the generated features may be no better than selecting four features from the original 20 features.

Other techniques that can be implemented to improve classification accuracy could be removing outliers, if any. Also, doing some research and understanding the data to gain domain knowledge can help improve the methods to choose, add, or remove features. Visualizing the features is also recommended to actually see the relationships between features, e.g., a heatmap of the correlation of features. Moreover, VAT analysis could be performed to possibly cross validate the number of clusters in the data for K-means clustering. K-fold cross validation or n-fold cross validation can also be implemented to guarantee accuracy is not just a result of a ‘lucky’ or ‘unlucky’ split on the data.

Since there are function parameters which were left to their default values, like the random state of mutual information for selecting 4 features, the result may vary leading to a reduced reliability. Since K-fold cross validation was not performed, the accuracy gathered on this experiment may just be a result of a ‘lucky’ or ‘unlucky’ split on the data with the fact that there are only more than a hundred data divided into training and test sets thus, the result may not be as reliable due to possible biased sampling. However, the result of this investigation is a good starting point towards a more reliable and accurate prediction model, e.g., based on this, focus could be given to improving feature engineering and exploring more appropriate prediction models, and applying k-fold cross validation to the dataset for a more reliable prediction model.