Classification, Selection and Feature Engineering Report

COMP20008

Fu-Sheng(Jeremy) Huang
1046432

Classification

Comparing Classification Algorithms

Approach

Before applying classification algorithms, pre-processing must be done to the data. First, exclude the irreverent columns, such as 'Time' and 'Year' from the extracted data, and inner merge them together takes the **intersection of two data, by 'Country Code'**. Then split the combined data into data (features) and classlabel (class/target) and impute the missing values in data with the feature's median value using SimpleImputer from sklearn.impute. Finally, split the data and classlabel into training data and test data and apply different classification algorithms that are **Decision tree and K-NN.**

Results and Evaluation

The executed program resulted prediction accuracy of 70.9% for decision tree, 72.7% for k-NN with k=3 and 76.4% for k-NN with k=7. Overall, k-NN, both k=3 and k=7 performed better on this dataset than using the decision tree (with maximum depth of 3). Often, **k-NN gives a higher accuracy over decision trees for low dimensional data**. However, the k-NN is often time consuming and cannot be applied to categorical data.

Feature Engineering and Selection

Approach

Same approach with task 2A was used up to the point of data imputation. For the purpose of the feature selection classlable is encoded using LabelEncoder from sklearn.preprocessing. Firstly, to create interaction pairs, use PolynomialFeatures from sklearn.preprocessing and with the parameter interaction_only=True. For clustering label, KMeans from sklearn.cluster was applied. Here, **number of clusters was set to 3.**

For the feature selection, SelectKBest from sklearn.feature_selection was used to select the 4 highest scoring features. The scoring function mutual_info_classif which computes the mutual information, and the dependency of a feature was used to obtain this.

For PCA, PCA from sklearn.decomposition is used with n_components=4.

Finally apply k-NN for three different methods.

Deciding on k for Kmeans

Here the elbow method was applied to visually decides the number of clusters for Kmeans (See fig 1 for the graph). Distortion indicates the change in SSE. To find the optimum number of clusters, the point where change in distortion is negligible was selected. In this case k=4 was selected.

Feature Selection Method

The mutual information classifier was used as scoring function. As mutual_info_classif is a non-parametric method, it does not require features to be normally distributed which allows it to detect non-linear correlations. As the sample size is large enough, mutual_info_classif is a suitable choice as scoring function in this feature selection. Attached figure (See fig 2 for the graph). is a bar graph of mutual information feature importance for each feature. Most importantly, imputation and feature generation or selection should not be influenced by the test data

Results and Evaluation

The program yields **based on random_state of 130** following accuracies for k-NN with k=4: 80.0% for feature engineering, 67.3% for PCA, and 78.3% for just using first for features. The results indicate that feature engineering has the highest accuracy. This is the case as for feature engineering out of a possible 211 features, the 4 best features that provides the most information were chosen out of 211 features. Whereas, PCA had to reduce the dimension from 20 features to 4 features which could be overly reducing the dimension resulting in less information that can be extracted from the resulting features. Similarly, selecting first four features does not give enough information especially when compared with feature engineering.

Improving Classification Accuracy

To further improve the accuracy, greedy feature selection method, such as recursive feature elimination, can be applied here to achieve a higher accuracy. However, more powerful machine is required as it will need more computation power and cost. We can also change random state value to increase the accuracy or increase the clusters number.

Reliability of Classification model

Considering the accuracy for the three methods is around 75% or less, it is not sensible to only use the classification models to predict the expectancy at birth as it could lead to a misleading prediction. Especially in PCA the accuracy is lower too much than the other which is not a reliable classification model for further prediction.

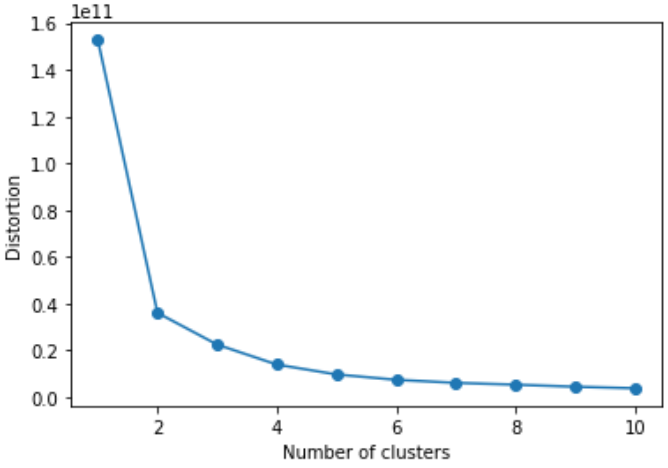

Figure 1: Cluster VS Distortion

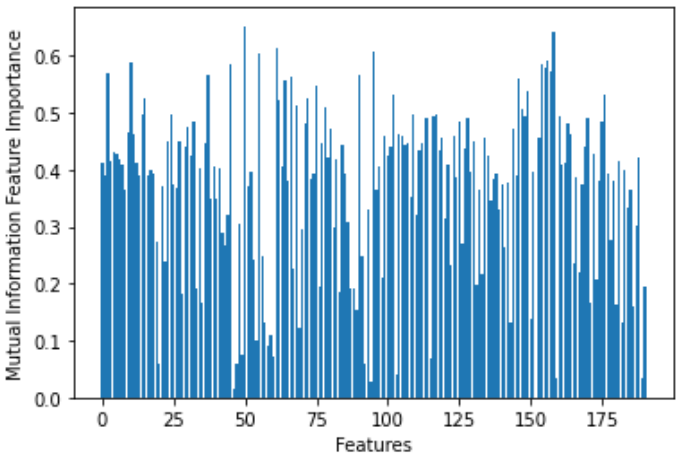

Figure 2: feature importance