Introduction to Machine Learning and Applications Term Project Report

Alokparna Bandyopadhyay
Computer Science Department
Wayne State University
Fall 2018
bandyopadhyay.alokparna@wayne.edu

I. INTRODUCTION

A. *Problem Statement:*

Nearly 38% of adults in the United States are obese, with rates rising stably and significantly over the past decade. Obesity places adults at risk for developing a plethora of serious medical comorbidities including cardiovascular disease, cancer and premature death. Identifying the salient risks for obesity and variance among subpopulations is imperative to optimize prevention efforts and treatment. Risk factor analysis is a common methodology to identify, rank and understand the underlying obesity risk factors and to inform prevention and treatment of preventable physical and mental health conditions more broadly. Risk factor analysis examines the complicated relation between output and input variables, where they are the outcome and features, respectively.

B. *Dataset:*

Behavioral Risk Factor Surveillance System (BRFSS) dataset is phone-based survey data collected from all the states/districts in the U.S. and filed by Centers for Disease Control and Prevention (CDC).

C. *Observations:*

The dataset has 358 features/columns and 450016 observations/rows. By analyzing the dataset, it was noticed that outcome value/class level is not mentioned in any of the columns. So, it is basically an unsupervised dataset, majority of the data being categorical (qualitative). It was also noticed that out of 358 columns, all the columns have one or more null values.

II. METHODS

A. *Dataset Preprocessing:*

The dataset is downloaded from as .xpt file from https://www.cdc.gov/brfss/annual_data/annual_2017.html and then converted into .csv file, which was then loaded in Python using pandas package for dataframe. The dataset shape is (450016, 358). Initially the dataset is treated as unsupervised data as there was no Class/Target value assigned to any of the columns and outcome was uncertain.

Since it was noticed that out of 358 columns, many of the columns have one or more null values, when a dropna(axis='columns') function is applied to the dataset to remove the columns with null values, 296 columns are dropped, and the original dataset shape gets reduced to (450016, 62).

Another attempt is taken to delete only those columns which have 50% null values. After removing the columns with more than 50% of null values, it was noticed that 175 columns are dropped, and the original dataset shape is reduced to (450016, 183). This process seemed better as it still retains more than half of the dataset features. This new dataset with 183 columns is used for further processing and unsupervised learning.

The missing values in the remaining columns are then imputed. Since most columns have categorical data, the imputation is done using ‘mode’, which replaces the null values of a column with the most frequent value in that column.

Some less important categorical features like "_STATE", "FMONTH", "IDATE", "IMONTH", "IDAY", "IYEAR", "DISPCODE", "SEQNO" and "_PSU" are dropped from the dataset as they tend to add noise to the data.

Later in the experiment, the dataset is made supervised to train the data with supervised learning models. The BMI level is treated as target and used to add a new column ‘Class’ to the dataset. A BMI value of 25 is taken as cut-off, where BMI $\geq$ 25 means the person is overweight (Class = 1), otherwise the person is normal (Class = -1). The actual BMI related columns are hence dropped from the database to reduce noise.

B. *Machine Learning Models*

a) *Unsupervised Learning*

For unsupervised learning, principal component analysis followed by k-means clustering is performed.

K-means clustering is a type of unsupervised learning, which is used for unlabeled data. The algorithm works iteratively to assign each data point to one of k groups based on the similarity of the features. To find the optimal number of clusters in the data, the user needs to run the K-means clustering algorithm for a range of k values and compare the results. Finding the optimal value of k is very important for K-means algorithm to perform its best and there are various

techniques for determining the exact value of k. One method to find the optimal value of k is the Elbow Method (using the within-cluster sum of squares), which is also used in this project.

A very high-level description of PCA is that it serves as a dimensionality reduction method on the features of the original dataset by projecting these features onto a lower dimensional space. Since even after pre-processing the BRFSS dataset has 183 columns/features, principal component analysis is done on the dataset to reduce the number of features to 2 and hence reduce the processing speed and time. After PCA, the new data is extracted using a clustering algorithm. In this project, K-means clustering is executed on the new data after PCA with an optimal value of k. The clusters are then visualized using matplotlib.

b) Supervised Learning

For supervised learning, Decision Tree Classifier using both Gini Index and Entropy are used in this project. Decision Trees are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified. An attribute with lower Gini index should be preferred. Entropy is the measure of uncertainty of a random variable, it characterizes the impurity of an arbitrary collection of examples. The higher the entropy the more the information content.

Next, the default AdaBoost classifier with Decision Tree Classifier as its base estimator is used to model the dataset. Ada-boost or Adaptive Boosting is one of ensemble boosting classifier which combines multiple classifiers to increase the accuracy of classifiers. It is an iterative ensemble method which builds a strong classifier by combining multiple poorly performing classifiers I order to get a highly accurate strong classifier.

Finally, Multi-Layer Perceptions of Neural Network are used to train the dataset. Neural Networks are a machine learning framework that attempts to mimic the learning pattern of natural biological neural networks. To create a neural network, layers of perceptrons are added together, creating a multi-layer perceptron model of a neural network.

Confusion matrix, accuracy score and classification report of all the above mentioned supervised classifiers are extracted and the results are compared.

III. Experiment and Results

A. Unsupervised Learning

The initial dataset is unsupervised. K-means clustering is done on the unsupervised data to create clusters after finding an optimal value of 'k' using the Elbow Method. The Elbow method is a method of interpretation and validation of consistency within cluster analysis designed to help finding the appropriate number of clusters in a dataset. Figure 1 shows the Elbow graph for the BRFSS dataset. A clear elbow at k=4 can be observed, indicating that 4 is the best number of clusters for the K-means algorithm to perform it best.

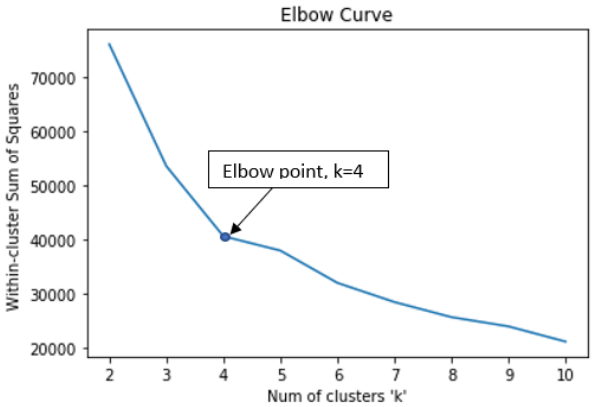

Fig 1. Elbow curve for finding the optimal cluster size

Furthermore, the dimension of the database is reduced by Principal Component Analysis, followed by K-means to visualize the clusters in their principal component space, as illustrated in Figure 2.

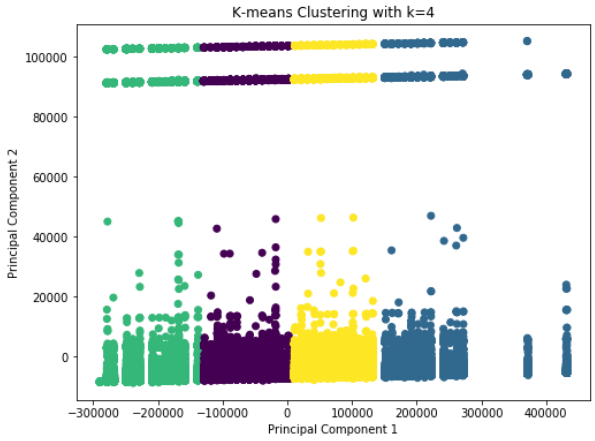

Fig 2. Clusters formed after PCA and K-means clustering

B. Supervised Learning

The database is then made supervised using BMI as a target to train the data with supervised learning models. The target variable is added to the last column of the dataset as 'Class' and given the value of 1 if BMI $\geqslant$ 25 (overweight), else it is given the value of -1 if BMI < 25 (normal).

The various supervised learners used in this project are Decision Tree Classifier using both Gini Index and Entropy, AdaBoost Classifier with Decision Tree Classifier as the base estimator and Multi-Layer Perceptron Classifier. Confusion matrix, accuracy score and classification report of all the above mentioned

supervised classifiers are extracted and tabularized in Table I for comparison.

TABLE I. CLASSIFICATION REPORT AND ACCURACY SCORE OF SUPERVISED CLASSIFIERS

Classifier	Precision	Recall	F-Score	Accuracy
DecisionTree using Gini Index	0.93	0.93	0.93	92.87698
DecisionTree using Entropy	0.91	0.90	0.90	89.87378
AdaBoost using Decision Tree	0.85	0.85	0.85	85.22510
MLP Neural Network	0.92	0.92	0.92	92.10590

From Table I it can be observed that all the supervised learners used in this project work quite well, with Decision Tree Classifier using Gini Index being the most accurate with an approximate accuracy of 93%, followed by the MLP Neural Network with an approximate accuracy of 92%.

It is also observed that the default AdaBoost classifier with DecisionTreeClassifier as its base estimator has the least accuracy of 85% (approx.). AdaBoost being an Ensemble learner is expected to increase performance of a weak learner like Decision Tree. But it is observed here that this assumption is not always true. Here the both the Decision Tree Classifiers with Gini Index and Entropy work better than the AdaBoost classifier. One main disadvantage of AdaBoost is that it is sensitive to noise data. It is highly affected by outliers because it tries to fit each point perfectly. This maybe the reason why its accuracy decreases when used in a huge noise-prone dataset like BRFSS data.

IV. CONCLUSION

In this project the huge dataset of BRFSS data is pre-processed using dataset cleaning techniques and various supervised and unsupervised learning methods of machine learning are executed on the processed dataset. Firstly, the dataset is treated as unsupervised and a principal component analysis of the dataset is done followed by cluster analysis using K-means clustering. The same dataset is then made supervised using BMI record and further classification is done on the data. The results of the four supervised learners are noted and compared to understand how each of them works in terms of accuracy and classification report.

BIBLOGRAPHY

[1] https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

[2] https://www.datascience.com/blog/k-means-clustering

[3] https://www.kaggle.com/arthurtok/principal-component-analysis-with-kmeans-visuals

[4] https://stackabuse.com/decision-trees-in-python-with-scikit-learn/

[5] https://www.datacamp.com/community/tutorials/adaboost-classifier-python

[6] https://www.kdnuggets.com/2016/10/beginners-guide-neural-networks-python-scikit-learn.html