SML Project Report

Jeremiah Malsawmkima Rokhum
Roll Number: 2021533
CSAI - IIITD
jeremiah21533@iiitd.ac.in

Manshaa Kapoor
Roll Number: 2021540
CSAI - IIITD
manshaa21540@iiitd.ac.in

I. Introduction

In this project, we are provided with a training data set (train.csv) that consists of 4098 columns. The first and last columns in the data set contain the ”ID” and target values(”Category”) respectively. Our task is to use this data to build a machine-learning model that can predict the target category for the given ”ID”.

Here, we will use various machine-learning techniques to train our model on the provided training data and generate predictions for the test data. The accuracy of our predictions will be the primary metric for evaluating the performance of our model

II. Algorithms used for building our Machine Learning Model

A. Dimensionality Reduction Algorithm

a) PCA: Principal components (PCA) are constructed in such a manner that the first principal component accounts for the largest possible variance in the data set. It is a popular technique for analyzing large data sets containing a high number of dimensions/features per observation, increasing the interpretability of data while preserving the maximum amount of information, and enabling the visualization of multidimensional data. [1]

b) LDA: Linear Discriminant Analysis (LDA) is a supervised learning algorithm used for classification tasks in machine learning. It is a technique used to find a linear combination of features that best separates the classes in a dataset. LDA works by projecting the data onto a lower-dimensional space that maximizes the separation between the classes. It does this by finding a set of linear discriminants that maximize the ratio of between-class variance to within-class variance. In other words, it finds the directions in the feature space that best separates the different classes of data. LDA assumes that the data has a Gaussian distribution and that the covariance matrices of the different classes are equal. It also assumes that the data is linearly separable, meaning that a linear decision boundary can accurately classify the different classes [2].

B. Outlier Detection Algorithm

a) Local Outlier Factor: The local outlier factor is based on a concept of a local density, where locality is given by k nearest neighbors, whose distance is used to estimate the density. By comparing the local density of an object to the local densities of its neighbors, one can identify regions of similar density and points that have a substantially lower density than their neighbors. These are considered to be outliers. [3].

b) K-Means: K-Means clustering is a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid), serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells. k-means clustering minimizes within-cluster variances (squared Euclidean distances), but not regular Euclidean distances, which would be the more difficult Weber problem: the mean optimizes squared errors, whereas only the geometric median minimizes Euclidean distances. For instance, better Euclidean solutions can be found using k-medians and k-medoids. We assign membership based on the proximity with cluster centroid and halt when maximum iterations are reached or centroids are stabilized. [4].

C. Classification Algorithm

a) Random Forest Classifier: Random Forests or Random decision forests is an ensemble learning method for classification, regression, and other tasks that operates by constructing a multitude of decision trees at training time. We use Bootstrap sampling to create multiple decision trees in which we take random subsets of data points from the training set to create N smaller data set and s decision tree is fitted on each subset. Given k number of records, we take out n number of records and m number of features, and decision trees are constructed for each entry. The output of all decision trees is used for the final classification decision. Further, Tree ensemble is carried out by taking the majority vote of the classes predicted by all output decision trees to improve the overall accuracy of the classifier. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned. Random decision forests correct for decision trees' habit of over-fitting to their training set. [5].

b) Logistic Regression: It is a statistical model that models the probability of an event taking place by having log-odds for the event be a linear combination of one or more independent variables.in binary logistic regression, there is a single binary dependent variable, coded by an indicator variable, where the two values are labeled ”0” and ”1”, while the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable. The corresponding probability of the value labeled ”1” can vary between 0 (certainly the value ”0”) and 1 (certainly the value ”1”), hence the labeling; Given a set of independent variables, the goal of logistic regression is to estimate the probability of an event occurring indicating the dependent variable being equal to 1. It uses a logistic function known as Sigmoid Function to transform a linear combination of the independent variables into a probability value between 0 to 1. [6].

III. Why choose Logistic Regression over Linear Regression

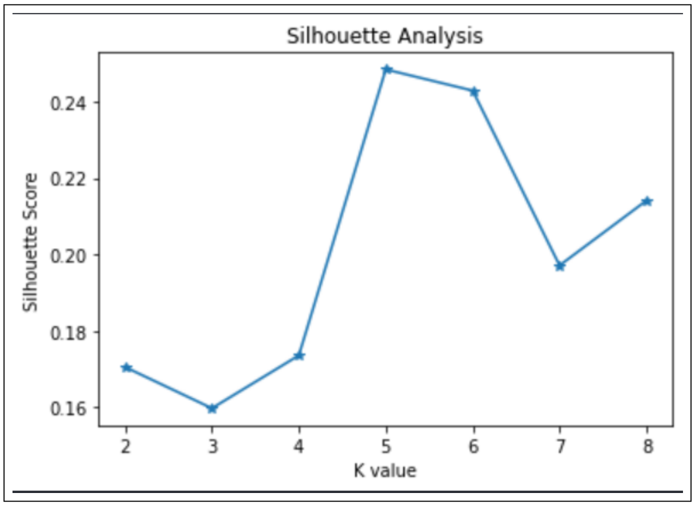
We choose Logistic Regression over Linear Regression because the problem set we were given had to estimate the relationship between a dependent variable and one or more independent variables. We had to make predictions about a categorical variable instead of a continuous one.

IV. Hyperparameter Tuning

A. K-Means Clustering and Local Outlier Factor

One of the key challenges while using K-means clustering and Local Outlier Factor is finding the optimal value of k as choosing an incorrect value of k will result in poor clustering of the data. There are several methods to determine the optimal value of k. We will use Silhouette Analysis for the same.

a) Silhouette Analysis: Silhouette Analysis is a method of interpretation and validation of consistency within clusters of data [7]. A silhouette score is a measure of how similar an object is to its own cluster. The silhouette coefficient is calculated using the equation - Where S(i) is the silhouette coefficient, a(i) is a measure of how dissimilar i is to its own cluster, and b(i) is the average distance from i to all the other clusters, where a(i) and b(i) can be calculated using - The value of S(i) ranges from [-1,1] where S(i)=1 indicates that the data is appropriately clustered, S(i)=-1 indicates that the samples have been assigned to the wrong cluster and S(i)=0 indicates that the datum is on the border of two natural clusters. To find the optimal value of k for K-Means Clustering, we pick a range of values for the parameter k within the range of [2,8] and train the model for each value of i in [2,8]. Further, we calculate the corresponding silhouette coefficient and plot them to observe the fluctuations and finally pick the k where the silhouette score is the highest. We can observe that at n=5, the silhouette score is the highest so k=5 is the optimal value of k.



B. PCA and LDA

By reducing the dimensionality of the data to a very small number of components, you may lose important information that is spread across the original features. This can result in a loss of discriminatory power, where the reduced-dimensional data may not effectively capture the underlying structure or patterns in the data In some cases, having too few components may result in overfitting, where the reduced-dimensional data may not generalize well to new, unseen data. This can happen when the reduced-dimensional data is not representative of the underlying data distribution, and the model may not perform well when applied to new data points.

Having too many components can result in overfitting, where the model captures noise or irrelevant variations in the data, rather than the underlying patterns or structure. This can lead to poor performance on new, unseen data, as the model may be too complex and not generalize well.

Using trial and error, we happened to find that the best n_components for LDA and PCA is 19 and 400 respectively.

C. Random Forest Classifier

In this classifier, we went for the default value of n_estimator as 100.

V. Some Common Mistakes

- We tried using Linear Regression instead of Logistic Regression and it provided us with a lot of hurdles when it came to accuracy scores.
- We also tried to use different values of n_components like below 100 and above 1000 for PCA and it didn’t give us the desired accuracy scores.
- We also tried to use different values of n_components for LDA too. But, we had to stick with 19.
- One of the hurdles that we still face now, is that the outliers are always coming out to be different which causes the accuracy to not be consistent.
- We tried using K-Nearest Neighbour as our classifier algorithm and it always decreased the accuracy.
- We tried using Decision Tree as our classifier algorithm and it always decreased the accuracy.

VI. References

References

[1] https://en.wikipedia.org/wiki/Principal-component-analysis
[2] https://en.wikipedia.org/wiki/Linear-discriminant-analysis
[3] https://en.wikipedia.org/wiki/Local-outlier-factor
[4] https://en.wikipedia.org/wiki/K-means-clustering
[5] https://en.wikipedia.org/wiki/Random-forest
[6] https://en.wikipedia.org/wiki/Logistic-regression
[7] https://en.wikipedia.org/wiki/Silhouette-(clustering)