Predicting miles per gallon for Automobile Using Principal Component Analysis & Multiclass Classification

Concordia Institute for Information Systems Engineering, Concordia University

Pralaksh Mishra 40192765
https://github.com/pralaksh/INSE6220/

Abstract

Automobile industry is one of the fastest growing industries in the whole world. With increase in the automobile production, a massive amount of automobile parametric data is generated. This data helps us to analyse the city-wide fuel consumption in miles per gallon in terms of two multivalued and three continuous attributes. Various statistical methods and data mining techniques can be used to understand the data patterns and make observations. Principal component analysis (PCA) is one of the immensely famous dimensionality reduction techniques that is used to transform the correlated data into uncorrelated data. In this paper, we will discuss the principal component analysis (PCA) alongside implementation of machine learning models using the same dataset to observe the miles per gallon data for the vehicles during the 1970s era.

Keywords—Principal Component Analysis (PCA), Machine Learning Models, mpg (miles per gallon)

I. Introduction

The Automobile sector has gone through various changes in the last few decades. Flashback to 1970s, the cars used to be heavy and less fuel efficient. Old vehicles used more fuel compared to the new generation vehicles as the old generation automobiles were big and heavy. The automobile mpg dataset is a collection of 90 automobile records from the year 1970 to 1982. It consists of information such as the number of cylinders present in the car, the year in which the car was manufactured, the total weight of the car, the net fuel displacement and the acceleration. The data helps us to predict the city-wide fuel consumption in miles per gallon in terms of two multivalued and three continuous attributes. The attributes cylinders and class (year of manufacturing) are the two multivalued attributes while the other attributes are continuous. The weight and mpg are very closely related. If the weight of the automobile is more then it impacts the fuel efficiency of the car which makes the car to consume more fuel per gallon. The other ways to look at the main attribute of the cars is to focus on number of cylinders and engine size of the car. If the number of cylinders and the engine size is more then it means that the fuel displacement is more, this results in more and more consumption of fuel. As the year passes, the technology also improves and that directly impacts the engine technology used in cars. Therefore, the year of manufacturing of the car also plays an important role in determining whether the car consumes more fuel or less. The research problem tried to answer the question on whether it would be possible to make predictions on car mileage per gallon using other underlying factors, which were all in the selected dataset. This study involves numerical variables therefore, it's an observational study and will further be complemented by a case analysis with the mileage being the dependent variable. The independent variable are the remaining columns in the dataset. The main concentration is identifying the contributing factors that actively influence the fuel consumption. [1][2][3]

II. Dataset Description

In this report, the data is extracted from the UCI Machine Learning library which has more than 700 datasets. This dataset contains six columns. The column description is given as follows: -

1) **Mpg**
   It is abbreviated for miles per gallon. The purpose of this paper is to predict the miles per gallons for the automobiles. Miles per gallon is the measure of distance that the automobile can travel per gallon of fuel.
2) **Cylinders**
   Cylinder is one of the most essential parts of automobile. It's a chamber where the fuel is

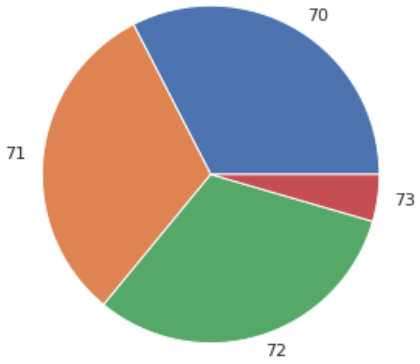
injected and combusted and eventually the power is generated.

3) **Displacement**
   Engine displacement is the measure of the cylinder volume which is swept by all the pistons.
4) **Weight**
   Weight signifies the total weight of the car which includes the engine and rest of the body.
5) **Acceleration**
   Acceleration is the rate at which the car can increase its pace. It is seen in the terms of the time that it takes to reach a particular speed.
6) **Class**
   The class column consists of the years in which the automobile under this dataset were manufactured.

The below is the dataset with first 30 rows: -

mpg	cylinders	displacement	weight	acceleration	class
18	8	307	3504	12	70
15	8	350	3693	11.5	70
18	8	318	3436	11	70
16	8	304	3433	12	70
17	8	302	3449	10.5	70
15	8	429	4341	10	70
14	8	454	4354	9	70
14	8	440	4312	8.5	70
14	8	455	4425	10	70
15	8	390	3850	8.5	70
15	8	383	3563	10	70
14	8	340	3609	8	70
15	8	400	3761	9.5	70
14	8	455	3086	10	70
24	4	113	2372	15	70
22	6	198	2833	15.5	70
18	6	199	2774	15.5	70
21	6	200	2587	16	70
27	4	97	2130	14.5	70
26	4	97	1835	20.5	70
25	4	110	2672	17.5	70
24	4	107	2430	14.5	70
25	4	104	2375	17.5	70
26	4	121	2234	12.5	70
21	6	199	2648	15	70
10	8	360	4615	14	70
10	8	307	4376	15	70
11	8	318	4382	13.5	70
9	8	304	4732	18.5	70

Below is the pie chart for the class column which shows the year in which the automobiles were manufactured.



III. Principal Component Analysis (PCA)

Principal Component Analysis is a reduction technique which can be used to reduce a large set of variables to a smaller set without losing the relevant information in the data. This method is based on eigen decomposition for positive semi-definite matrices and singular value decomposition for rectangular matrices. The distinct principal components are min (n-1, p) since there are n observations of p variables. The first principal component has the greatest possible variance (that is, it accounts for as much heterogeneity in the data as possible), and each subsequent component has the highest variance possible while remaining orthogonal to the preceding components. [4] First principal component decides the direction of most variability in the data. If the variability caught is high in the first component, then it implies more information is caught by component. No other component can have higher variability than the first principal component. The second principal component is a linear combination of original predictors like the first component which catches the rest of variance in the dataset and is uncorrelated with the first principal component outcome. As a result, the correlation between the first and second component should be zero. The direction of two components is orthogonal, if they are uncorrelated [5][6]

IV. PCA Algorithm

These are the following steps for PCA Algorithm:

Step-1
Compute the centred data matrix Y= HX by subtracting off-column means.

Step-2
Compute the r*r covariance matrix S of the centred data matrix as follows:

$$S = \frac{1}{n-1} Y' Y$$

Step-3

Compute the eigenvectors and eigenvalues of S using eigen-decomposition

$$S = A \wedge A' = \sum_{J=1}^{P} \lambda_J a_j a_j'$$

where: A is a p x p orthogonal matrix (i.e. A*'* $A = I$) with columns $a_j = \left(a_{j1}, a_{j2}, \ldots\ldots, a_{jp} \right)$ that are S's eigenvectors.

$\wedge = diag\left(\lambda_1, \lambda_2, \ldots, \lambda_p \right)$ is a p × p diagonal matrix with the eigenvalues of S ordered in decreasing order as its components.

Step-4

Calculate the n × p transformed data matrix $Z = YA$

$$Z = \left(z'_1, z'_2, \ldots, z'_i, \ldots\ldots, z'_p \right) = \begin{bmatrix} z_{11} & z_{12} & \cdot\cdot & \cdot\cdot & z_{1p} \\ z_{21} & z_{22} & \cdot\cdot & \cdot\cdot & z_{2p} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ z_{n1} & z_{n2} & \cdot\cdot & \cdot\cdot & z_{np} \end{bmatrix}$$

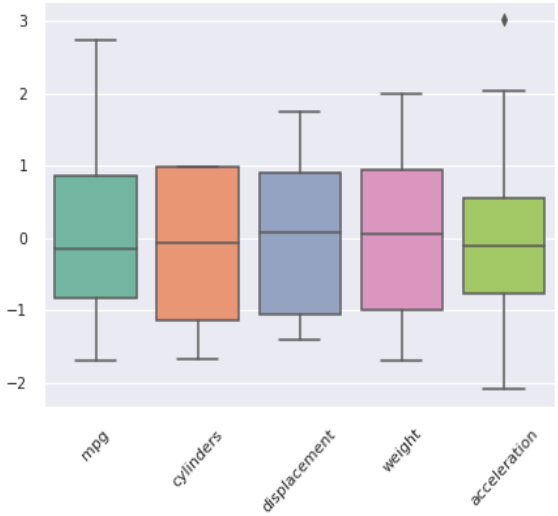
The variances of the PC column's eigenvalues are the eigenvalues. As a result, the first PC component results for as much variability in the data as possible, while each subsequent component score accounts for as much variability as possible.

V. PCA Implementation

With support of the MATLAB libraries, we brought PCA in python programming language. Using python and PCA, we analysed our data and were able to obtain different graphs for the results: -

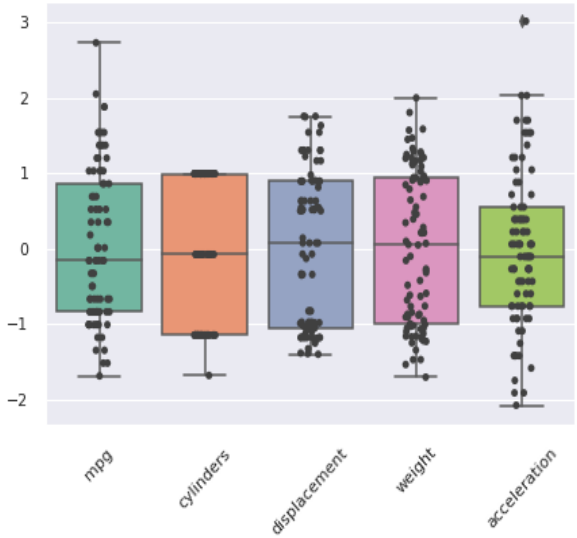
1) Box Plot

Box Plot creates a visual schematic representation of the distribution of the quantitative data. It divides the data into quartiles and whiskers with possibly showing the outliers.



2) Swarm Plot

A swarm plot here is drawn with box plot to show all observation along with some representation of the underlying distribution. Non-default axis limits must be set before drawing the plot.



3) Covariance Matrix

In this step we identify the correlation between variables. The association of various attributes is shown in the covariance matrix. It helps us to understand how our variables change with respect to mean. If we have the positive sign, that means the variables increase or decrease together otherwise they inversely correlated to each other.

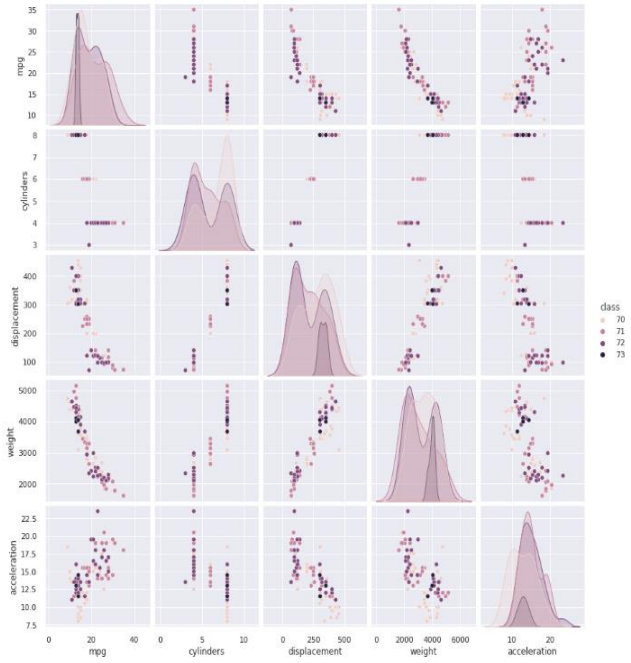
The first Eigenvector of the correlation matrix is $\left(a_{11}, a_{21}, \ldots . a_{p1} \right)$ and coefficients of the first principal component.

The second Eigenvector of the correlation matrix is $\left(a_{12}, a_{22}, \ldots . . a_{p2} \right)$ and coefficients of the 2nd principal component.

The coefficients of the pth principal factor and the pth Eigenvector of the correlation matrix are $\left(a_{1p}, a_{2p}, \ldots . . a_{pp} \right)$.

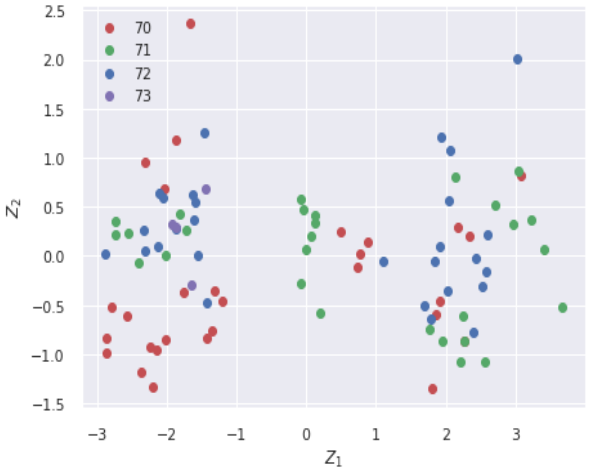
4) Pair Plot

A pair plot helps us to see both single variable distribution and associations between two variables. A pair plot is a matrix of scatterplots that lets you understand the pairwise relationship between different variables in a dataset. [7]



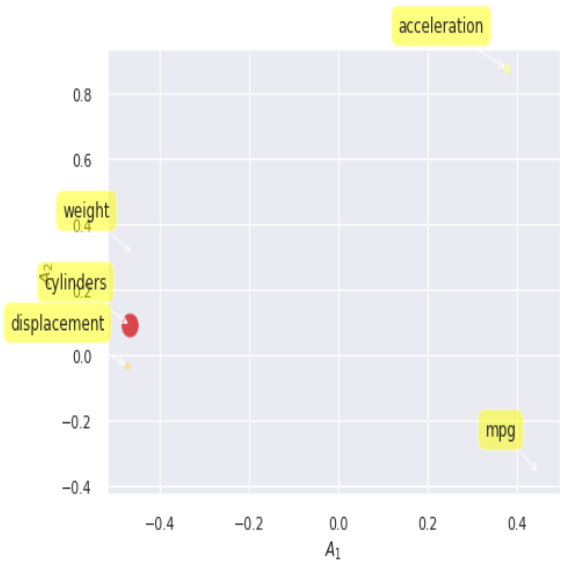
5) Scatter Plot

A scatter plot or scatter chart uses dots to represent values for two separate variables and the position of each dot on the horizontal and vertical axis indicates values for an individual data point. The scatter plot shows the different years that indicate the year of manufacturing of the cars.



6) Principal Component Analysis

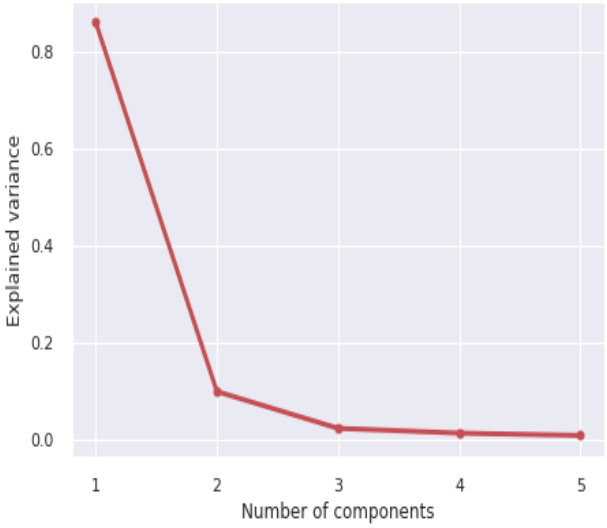
Principal Component Analysis is one of the key methods to determine the contribution of each attribute to the principal components. For this purpose, PC coefficients are plotted against each other.



7) Scree Plot

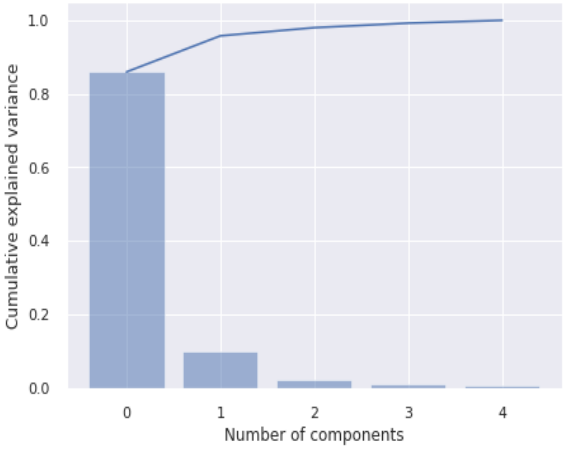
A scree plot is a graphical tool used in the selection of the number of relevant components or factors to be considered in a principal components analysis or a factor analysis. [8]

Based on the analysis, which is presented in Scree plot, we can infer that the first two components account for more than 90% of the variance. Therefore, after data reduction we finalise d=2.



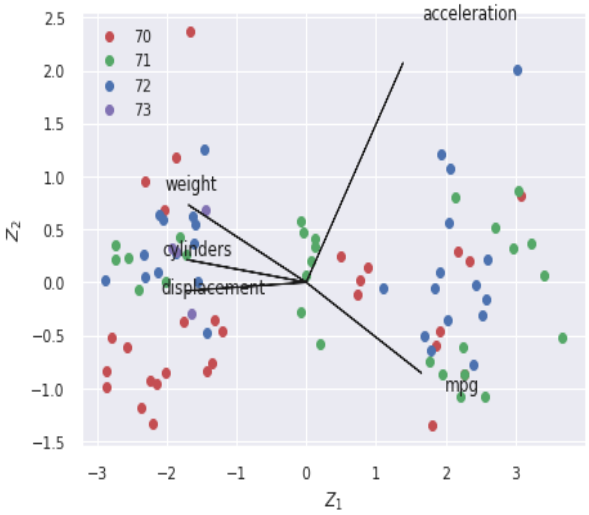
8) Pareto Chart

It can be seen from the Pareto charts that the low-dimensional space can be reduced to two dimensions because the first two components result for more than 90% of the variance.



9) Biplot

In Biplot, the PC coefficients for each vector and the principal component scores for each observation can be seen. The observations can be represented by points and the sectors are represented by vectors. The position and length of each vector shows the contribution to the two main components. As we can infer that most of the data points are distributed around the zero lines.



VI. PCA Results

The main goal of PCA is to reduce the dataset's dimension. The n x p dataset is reduced to a smaller size using the eigenvector matrix (A). The eigenvector matrix obtained after applying PCA is: -

$$\begin{bmatrix} 0.45052008 & -0.36162259 & -0.67171876 & -0.46161435 & -0.0443492 \\ -0.4673529 & 0.09036072 & -0.57981359 & 0.36949244 & -0.54836825 \\ -0.4726486 & -0.03404902 & -0.36688515 & 0.02237823 & 0.80021151 \\ -0.4603747 & 0.30837196 & 0.0907604 & -0.80424947 & -0.19469773 \\ 0.37851203 & 0.87453498 & -0.2641362 & 0.05540335 & 0.13812943 \end{bmatrix}$$

The variations in the data that PC records are called the eigenvalues. The amount of variance is shown using the scree plot and a pareto chart. The following equation can be used to calculate the percentage of variation accounted by the jth principal component:

$$\ell_j = \frac{\lambda_j}{\sum_j^p \lambda_j} \times 100\%, \quad for\ j = 1, \ldots, p$$

where λj is the jth element in the PC
The eigenvalue matrix is:

$$\begin{bmatrix} 4.3469 \\ 0.4965 \\ 0.1124 \\ 0.0625 \\ 0.0383 \end{bmatrix}$$

The following are the principal components are:

Z1 = 0.45052008***X1*** + (-0.4673529) ***X2*** + (-0.4726486) ***X3*** + (-0.4603747) ***X4*** + 0.37851203***X5***

Z2= (-0.36162259) ***X1*** + 0.09036072***X2*** + (-0.03404902) ***X3*** + 0.30837196***X4*** + 0.87453498***X5***

In first principal component, X1 and X5 have positive contribution whereas rest have negative contribution.

In the second principal component, X1 and X3 have negative contribution whereas rest others have positive contribution.

VII. Classification Results

In this section, we have used Pycaret library in python to shortlist the performance of various common classification algorithms on the automobile mpg dataset. Based on the list, we have selected four common classification algorithms that are Logistic Regression, Decision tree classifier, K Nearest Neighbours and Random Forest classifier. The models were trained and tested on the base dataset, on transformed dataset and on first two principal components dataset.

All the datasets were split into training and test dataset with 70-30 split. Then, we did a grid search on the training sets using k-fold cross validation to obtain the ideal hypermeter values for each algorithm. We have used 10-fold cross validation since the data is large. The purpose is to assess the impact of principal components analysis on different classifiers and to compare the performance of the models.

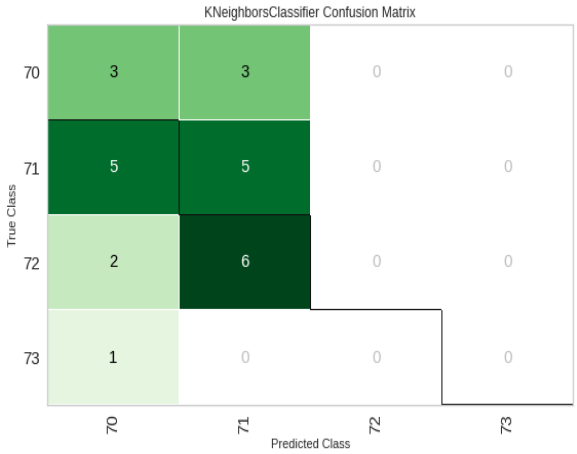
We analysed the original data using the following classification algorithms: -

1) Decision Tree Classifier

Decision tree learning or induction of decision trees is one of the predictive modelling approaches. It uses a decision tree (as a predictive model) to go from observations about an item to conclusions about the item's target value. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. [9]. Decision Tree Classifier performed best on the original dataset in predicting the miles per gallon for the automobiles. The confusion matrix of the decision tree classifier is shown further in the paper where the best model is discussed.

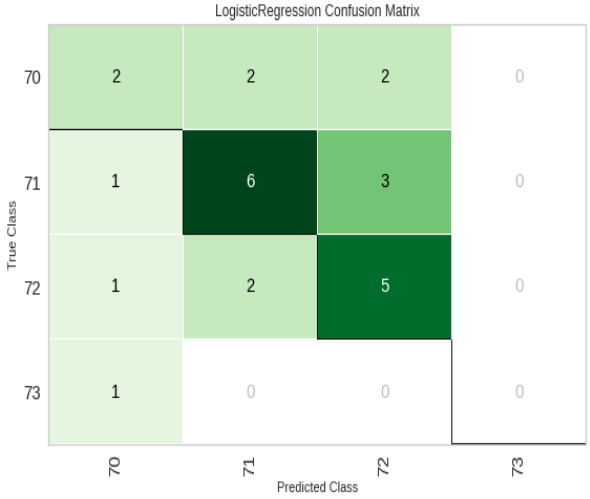
2) K Neighbours Classifier

The k-nearest neighbours (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems. It's easy to implement and understand but has a major drawback of becoming significantly slows as the size of that data in use grows. [10]. Below is the confusion matrix of the KNN algorithm.



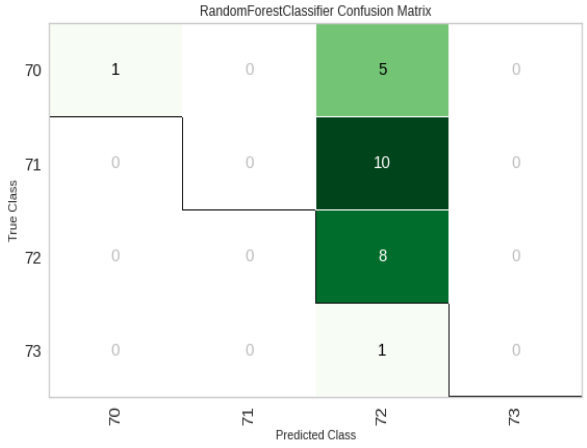
3) Logistic Regression

Logistic regression models the probabilities for classification problems with two possible outcomes. It's an extension of the linear regression model for classification problems. [11] Logistic regression models have a particular decided number of parameters that rely on the number of input features, and they output categorical prediction. For instance, whether a plant belongs to a certain species or not. Below is the confusion matrix of Logistic regression.

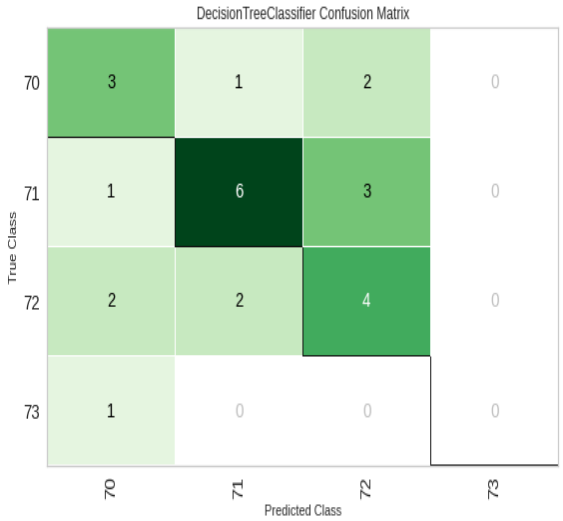


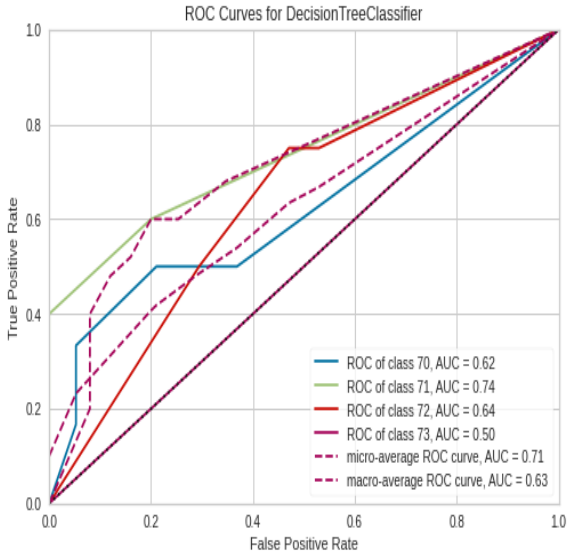
4) Random Forest Classifier

Random forest, like its name implies, consists of many individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. [12]. Below is the confusion matrix of Random Forest classifier.

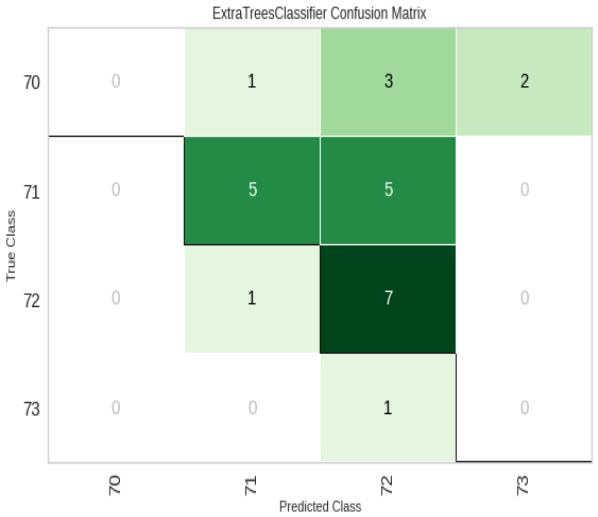


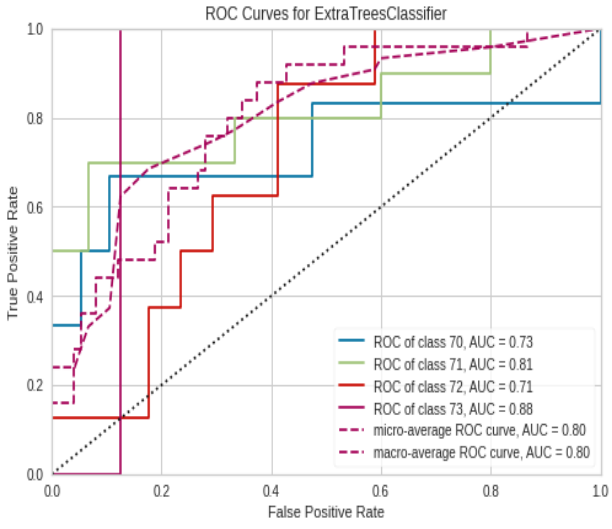
We selected the best algorithm out of the other classification algorithms that we tried.
The Decision Tree Classifier turned out to be the best model for the tuned dataset. The confusion matrix is a table which shows the actual positives, false positives and the actual negatives predicted for each class. It also highlights the most trouble making correct predictions for the model. ROC curves plot the percentage of incorrectly predicted negatives which is also called as false negatives versus the percentage of correctly predicted positives which are also called true positives for different threshold values. The Area under the curve which is also called as AUC is a single value summary for the ROC Curve. The models that have area under the curve as one is considered as perfect classifiers and models with area under the curve as 0.5 are termed as random classifiers.





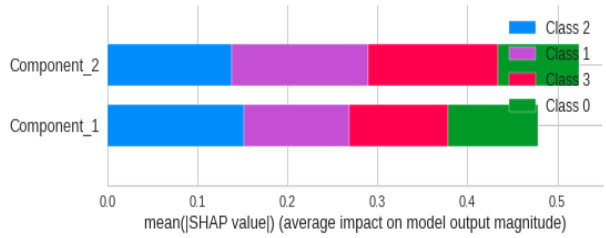
We further apply principal component analysis on the original dataset wherein we have reduced the dimension to two PC's dataset. After applying PCA, we notice that the best performing dataset for the tuned data that was Decision Tree Classifier is no longer the top performer, instead the best performing model is Extra Trees Classifier with higher accuracy than the Decision Tree Classifier.

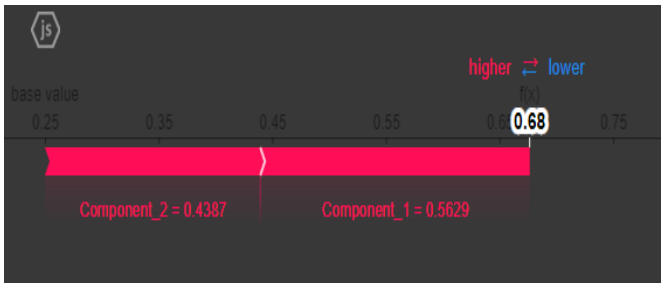


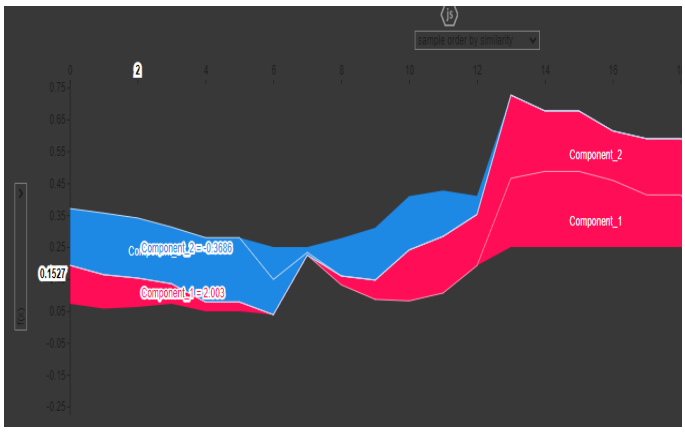


VIII. Explained AI with Shapley Values

This section explains machine learning models with shapley values. Shapley values are a widely used approach from cooperative game theory that come with desirable properties. [14]. in which the results of the solution can be understood by humans. It contrasts with the concept of the "black box" in machine learning where even its designers cannot explain why an AI arrived at a specific decision.[13] We have used Random Forest Classifier along with the principal component analysis to tune the model and predict the miles per gallon for the automobiles. In the SHAP summary plot, features are sorted by the sum of the SHAP value and its magnitudes across all the samples. In the prediction visualization, the size of each principal component shows the impact of it on the model.







IX. Conclusion

In this report, we successfully predicted the miles per gallon(mpg) attribute of automobiles manufactures in the 70s using Decision Tree Classifier as it was the best performing classifier for the tuned data. We applied Principal Component Analysis on the original dataset and found that the more than 90% of the variance is given by the first two principal components. As a result, we worked on those two components. We checked the impact of PCA and various classification algorithms and we found that Extra Trees Classifier was the best performer among other algorithms. Decision Tree Classifier performed better on the original and transformed dataset, but Extra Trees Classifier performed best on the first two datasets.

References

[1] https://rstudio-pubs static.s3.amazonaws.com/496255_92d2051015464b62aed8abb82a4ab219.html

[2] Quinlan,R. (1993). Combining Instance-Based and Model-Based Learning. In Proceedings on the Tenth International Conference of Machine Learning, 236-243, University of Massachusetts, Amherst. Morgan Kaufmann.

[3] https://www.linkedin.com/pulse/exploring-visualizing-auto-mpg-data-set-watson-analytics-joseph-true/

[4] https://en.wikipedia.org/wiki/Principal_component _analysis

[5] I. Jolliffe, “Principal component analysis,” Technometrics, vol. 45, no. 3, p. 276, 2003

[6] G. H. Dunteman, Principal components analysis. Sage, 1989, no. 69

[7] https://www.statology.org/pairs-plot-in-python/#:~:text=A%20pairs%20plot%20is%20a,different%20variables%20in%20a%20dataset.

[8] https://methods.sagepub.com/reference/the-sage-encyclopedia-of-educational-research-measurement-and-evaluation/i18507.xml#:~:text=A%20scree%20plot%20is%20a,analysis%20or%20a%20factor%20analysis

[9] https://en.wikipedia.org/wiki/Decision_tree_learning

[10] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761#:~:text=Summary,that%20data%20in%20use%20grows

[11] https://christophm.github.io/interpretable-ml-book/logistic.html

[12] https://towardsdatascience.com/understanding-random-forest-58381e0602d2

[13] https://en.wikipedia.org/wiki/Explainable_artificial_intelligence

[14] https://shap.readthedocs.io/en/latest/example_notebooks/overviews/An%20introduction%20to%20explainable%20AI%20with%20Shapley%20values.html

[15] INSE 6220 Lecture notes