INSE 6220-Advanced Statistical Approaches to Quality

Principal Component Analysis of Predicting Miles per Gallon for Automobile

Concordia Institute for Information Systems Engineering, Concordia University

Student Name-Jay Shah
Student ID-40224404
https://github.com/jugal9795/INSE-6220

Abstract

The automobile industry is one of the booming industries in the whole world. Automobile production has increased in the last few decades, and a huge amount of automobile data is produced. This helps us to examine the country-wise fuel consumption in miles per gallon in terms of seven attributes (multi-valued and continuous). Various statistical methods and data visualization techniques are used to understand the dataset model and make certain predictions. In this project Principal component analysis (PCA) is performed on an automobile dataset using python code to understand fuel consumption in three types of vehicles based on their origin. The three types are type 1- American origin (Chevrolet, Buick, Plymouth, AMC, Ford, Dodge), type 2- German origin (Volkswagen, Peugeot, Audi, BMW, Saab, Opel, Fiat, Mazda, Volvo, Mercedes) and type 3- Japanese origin (Datsun, Toyota, Honda). It is one of the most used dimensionality reduction techniques that is used to transform data into uncorrelated data. In this paper, along with principal component analysis (PCA), we have implemented mac learning using classification models on the automobile dataset to observe the miles per gallon data for automobiles.

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

I. Introduction

The Automobile sector is going through a lot of changes in recent years. In the 1970s, cars were heavy and consumed more fuel. Back in the 70s, vehicles produced more fuel than new vehicles, which are quite light in weight and more fuel efficient. The automobile dataset is a collection of automobile records from the years 1970 to 1982. The information like the number of cylinders in the car, the car manufacturing year, weight of the car, fuel displacement, and acceleration. The data helps to predict the fuel consumption in miles per gallon in terms of two multivalued and five continuous attributes. The attributes cylinders and class are the two multivalued attributes while the other attributes are continuous. The weight and mpg are co-related. If the weight of the vehicle is more then its fuel efficiency is less, making the car consume more fuel per gallon. If the number of cylinders and the engine size is more, then the fuel displacement is more, resulting in more and more fuel consumption. The technology also improves, directly impacting the engine model used in cars, which plays an important role in determining whether the car will be fuel efficient or not. Because this report contains numerical variables, it is an observational study that will be supplemented by a case study with mileage as the dependent variable. The other variables are independent. The primary focus is on identifying the factors that actively influence fuel consumption.

Python programming and some machine learning principles were applied to achieve this. The goal is to develop a plan of action to address the problem statement. Data exploration (Measures of Spread, Box plot and whisker plot, Covariance matrix, Pair Plot), PCA algorithm implementation (Principal Components, Eigenvector matrix, Eigen value matrix, Scree plot, Pareto Charts, Scatter plots, Biplot, Control charts), classification algorithm implementation (Confusion matrices, Linear Regression, Gradient Boosting Classifier, K-nearest neighbors, Random Forest Classifier), project outcomes, and conclusion.

II. Dataset Description

The data in this report is drawn from the UCI Machine Learning library, which contains over 700 datasets. This dataset contains seven columns.

1. **MPG**- Abbreviation used for miles per gallon. The purpose of this paper is to predict the miles per gallon fuel consumption of vehicles.
2. **Cylinders**-A cylinder is an essential part of a vehicle. It is a chamber where fuel is introduced, burned, and ultimately power generation.
3. **Displacement**-Engine displacement is a measurement of the cylinder capacity that all pistons sweep through.
4. **Horsepower**-The power produced by the engine and used for getting high-speed vehicles and which increases vehicles performance.
5. **Weight**- The term "weight" refers to the car's overall weight, which includes the engine and the rest of the body.
6. **Acceleration**- Acceleration is the rate at which the car can pick up speed. It is measured in terms of the time it takes to reach a specific speed.
7. **Class**- This column indicates the type of vehicle, which is classified according to the manufacturer.

Below is the pie chart for the class column which shows the type of automobile manufactured.

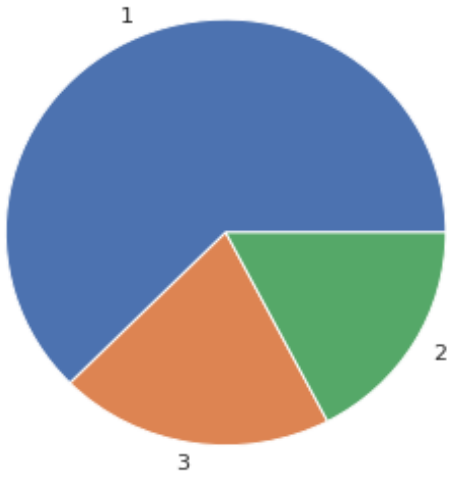

Fig 1. Pie Chart

III. Principal Component Analysis (PCA)

We need to know what is dimensionality reduction and the importance of the method before we perform principal component analysis. The transformation of multi-dimensional data into a representable and reduced dimension is called dimension reduction. The importance of this method is that the large set of variable data is reduced to a smaller dataset without losing any prevalent information. The reduced representation should have a dimensionality that matches the intrinsic dimensionality of the data. To handle such huge data effectively, its dimensionality must be reduced because high dimensionality can suggest a high processing cost while learning a model.

A. PCA Algorithm- Practically we have to apply the algorithm on an n*p dimension which is known as Data Matrix (X).

$$X(nxp)=\begin{bmatrix} x11 & x12 & \dots\dots & x1j & \dots.x1p \\ . & . & . & . & . \\ xi1 & xi2 & . & xij & xip \\ . & . & . & . & . \\ xn1 & xn2 & . & xnj & xnp \end{bmatrix}$$

The column represents variable data and the row denotes observations.

Step 1- Center Data Matrix- Here we subtract the column mean from each element of the data matrix and find the maximum spread. It is known as balanced data and the average of the centered data matrix is zero.

$$\bar{X}=\frac{1}{n}\sum_{i=1}^{n}X_i \qquad -(1)$$

The matrix with n*p dimensionality can be written as

$$Y=HX \qquad -(2)$$

Step 2- Covariance Matrix- In this step Covariance matrix (S) is calculated from the (Y) centered data matrix.

$$S=\frac{1}{n-1}Y^TY \qquad -(3)$$

The resulting covariance matrix has a p*p dimension which is a square matrix and can be decomposed using Eigenvalue Decomposition.

Step 3- Eigenvalue Decomposition- In this eigenvector matrix (A) and eigenvalue matrix (λ) of the covariance matrix (S) are computed using the:

$$S = A\lambda A^T = \sum_{j=1}^{p} \lambda_j a_j a_j^T \quad -(4)$$

The matrix A (p*p) is an orthogonal matrix that represents the PC directions of the eigenvector.

$$A\ (pxp) = \begin{bmatrix} a11 & . & a1i & ....a1p \\ . & . & . & . \\ ai1 & . & aii & aip \\ . & . & . & . \\ ap1 & . & api & app \end{bmatrix} \quad -(5)$$

The captured variance is given by each PC. It is represented as a diagonal matrix.

$$\lambda\ (pxp) = \begin{bmatrix} \lambda 1 & 0 & ....0 \\ . & . & . \\ 0 & \lambda i & 0 \\ . & . & . \\ 0 & 0 & \lambda p \end{bmatrix} \quad -(6)$$

NOTE- $\lambda 1 > \lambda 2 > \ldots \lambda i > \ldots \lambda p$.

Step 4- Principal Components- The transformed data matrix (Z) is calculated by the formula-

$$Z = YA \quad -(7)$$

The PCs are the same, as that of the original data matrix. They are represented as columns in the Z matrix.

IV. Classification Algorithms

Machine learning is an approach to teaching computers to deal with data more accurately and effectively, and it is involved with algorithms that gain knowledge from examples. Classification is the technique of clustering objects into prescribed groups that use ML algorithms to figure out how to apply a class label to samples from the problem area. This report compares four different classification algorithms: gradient boosting, K-nearest neighbors, logistic regression, and the tree-based Random Forest Classifier.

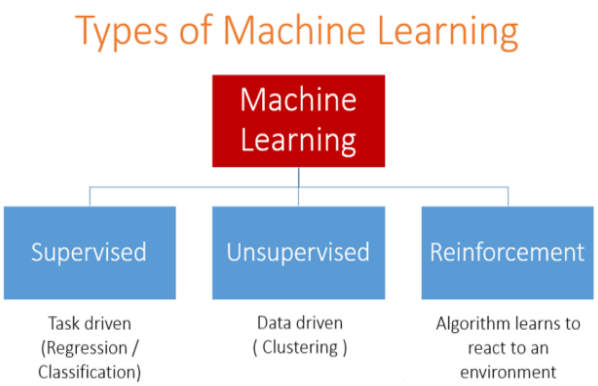

Fig 2. Types of Machine learning Problems

Gradient Boosting Classifier-

Both continuous and categorical target variables can be predicted using the gradient boosting approach as a Classifier. The cost function is Mean Square Error (MSE) when it is used as a regressor, while it is Log Loss when it is used as a classifier.

$$\tilde{y}_i = -\left[\frac{\partial L(y_i F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, i = 1, N$$

GBC models are powerful algorithms that can be used for classification as well as regression. On very complex datasets, the GBC model can perform incredibly well.

K-Nearest Neighbor(K-NN) Classifier-

Because it has no parameters, no training, and is based on instances, the K-NN algorithm is known as the lazy learner. Simply looking at the k closest points in the training set yields the new forecast. The k closest points in the dataset are determined by measuring the distance between the old and new points in the dataset. Data prediction is written as-

$$P\ (y=c \mid x, k) = \frac{1}{k} \sum_{i \in N\ k(x,D)} I(y_i = c)$$

K-NN builds a model of the target function using every labeled training instance. K-NN first chooses the number of neighbors, calculates the Euclidean distance of the chosen neighbors, and then chooses the K nearest neighbors based on the Euclidean distance. In the end, K-NN places the new sample in the class with the most samples.

Logistic Regression-

To establish a relationship between the class and features, LR attempts to create the best-fitting model. LR assigns the samples a value of 1 or 0. For example, if the sample's value is 0.49 or less, it is labeled as 0. If, on the other hand, the sample value is 0.5 or higher, the LR classifies the sample as 1. The logistic function is written as-

$$S(z) = \frac{1}{1+e^{-z}},$$

The primary benefit of using LR is that it is very simple to implement and can handle the classification problem of Vehicle type, such as whether it's type 1, 2, or 3. However, in high-dimensional data sets, LR is prone to overfitting. It is also sensitive to outliers, which means that if any sample deviates significantly from the expected range, the classification may produce incorrect results. This issue, however, can be solved with proper feature scaling.

Random Forest Classifier:

Random Forest creates a forest of classification or regression trees instead of just one single tree for classification or regression. Based on a set of traits, each of them generates a type. This tree may be seen as a vote, with the categorization being determined by the number of votes received.

$$Gain(t,x) = E\,(t) - E\,(t,x)$$
$$E(t) = \sum_{i=1}^{c} - P_i \log_2 P_i$$
$$E(t,x) = \sum_{c \in X} P(c)E(c)$$

Random forests are made up of multiple single trees, each based on a random sample of the training data. They are usually more accurate than single decision trees. As more trees are added, the decision boundary becomes more accurate and stable.

V. PCA Implementation

Measure of Spread

Measures of spread show how the collected data varies about a particular variable. Measures of spread include meaning, variance, standard deviation, range, and interquartile regions. The mean is the sum of all the points. Variance describes how your data is distributed around the mean and is very useful for data visualization. It is the standard deviation squared. The range is defined as the difference between the maximum and minimum data points. The data is standardized, as we can see below figure depicts the data spread measures:

	count	mean	std	min	25%	50%	75%	max
mpg	392.0	23.445918	7.805007	9.0	17.000	22.75	29.000	46.6
cylinders	392.0	5.471939	1.705783	3.0	4.000	4.00	8.000	8.0
displacement	392.0	194.411990	104.644004	68.0	105.000	151.00	275.750	455.0
horsepower	392.0	104.469388	38.491160	46.0	75.000	93.50	126.000	230.0
weight	392.0	2977.584184	849.402560	1613.0	2225.250	2803.50	3614.750	5140.0
acceleration	392.0	15.541327	2.758864	8.0	13.775	15.50	17.025	24.8

Fig 3. The measure of the spread of data

Box Plot

A visual schematic depiction of the distribution of the quantitative data is produced via a box plot. The data is divided into quartiles and whiskers, and any outliers may be displayed.

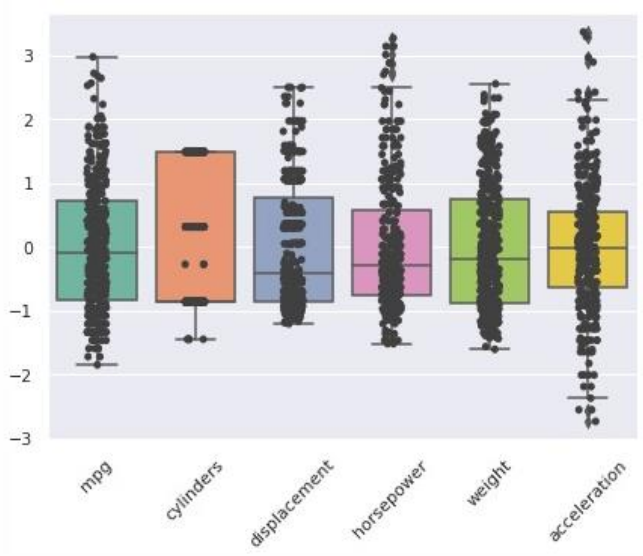

Fig 4. Box and Whisker Plots

Box plots and Whisker plots, which are graphical representations of numerical data points incorporating the qualities mentioned above, such as maximum, minimum, outliers, and distinct interquartile zones, can be used to show data dispersion. the dataset's median for the first quartile (Q1), the second quartile (Q2), and the third quartile (Q3). It also demonstrates how skewed the numbers are.

Covariance Matrix

The covariance matrix, also known as the pairwise analysis matrix, is used to examine the relationship between two variables. This is used to determine the magnitude of the relationship between the variables. Each variable has a relationship coefficient ranging from - 1 to +1. The diagonal members of the covariance matrix are always one because the relationship between the same variables is always unity.

	mpg	cylinders	displacement	horsepower	weight	acceleration	Class
mpg	1	-0.78	-0.81	-0.78	-0.83	0.42	0.57
cylinders	-0.78	1	0.95	0.85	0.9	-0.51	-0.57
displacement	-0.81	0.95	1	0.9	0.94	-0.55	-0.62
horsepower	-0.78	0.85	0.9	1	0.87	-0.69	-0.46
weight	-0.83	0.9	0.94	0.87	1	-0.42	-0.59
acceleration	0.42	-0.51	-0.55	-0.69	-0.42	1	0.21
Class	0.57	-0.57	-0.62	-0.46	-0.59	0.21	1

Fig 5. Covariance matrix

Pair Plot

A pair plot allows us to see the distribution of a single variable as well as the relationships between two variables. A pair plot is a scatterplot matrix that shows the pairwise relationship between different variables in a dataset.

Fig 6. Pair Plot

Scatter Plot

A scatter plot or scatter chart uses dots to represent the values of two distinct variables, with the position of each dot on the horizontal and vertical axes indicating the value of a single data point. The scatter plot depicts the various groups of vehicles based on their type.

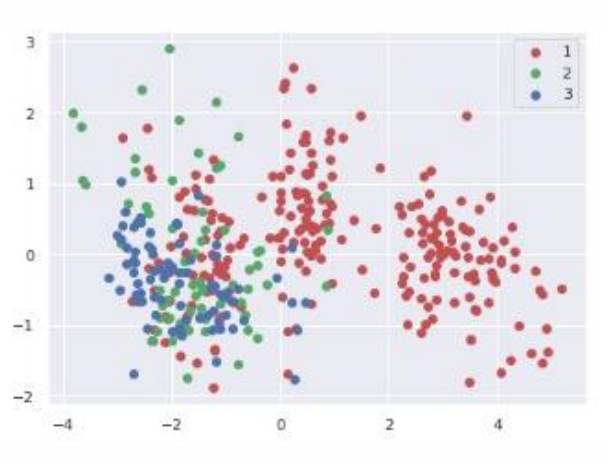

Fig 7. Scatter Plot

Eigen Vectors

One of the most important methods for determining the contribution of each attribute to the principal components is principal component analysis. PC coefficients are plotted against each other for this purpose.

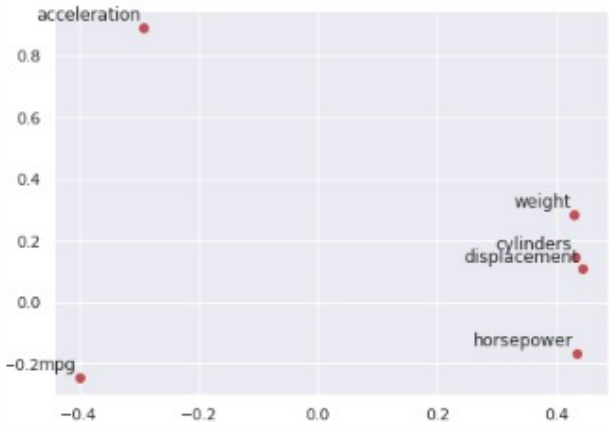

Fig 8. PC Coefficients

Scree Plot

A scree plot is a graphical tool used to choose the number of pertinent components or factors to take into account in a factor analysis or principal components analysis. We can conclude that the first two components, which are shown in a Scree plot, account for more than 90% of the variance based on the analysis. As a result, following data reduction, we determine that r=2.

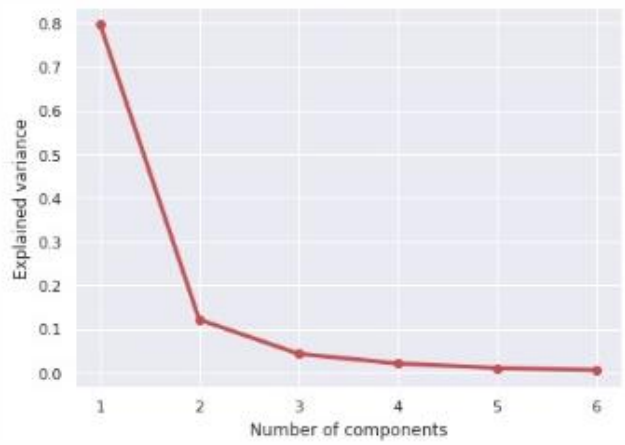

Fig 9. Scree Plot

Pareto Chart

The Pareto plots demonstrate that you can condense low-dimensional space to two dimensions considering that the first two elements are more than 90% of the variance.

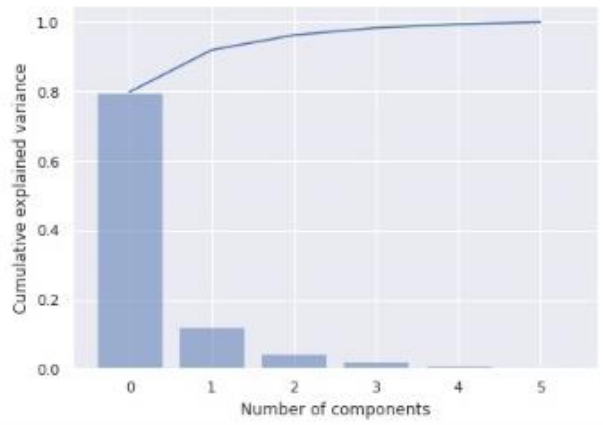

Fig 10. Pareto Chart

Biplot

Biplot displays the PC coefficients for each vector as well as the principal component scores for each observation. Points can be used to represent observations, and vectors can be used to represent sectors. Each vector's position and length indicate its contribution to the two main components. As we can see, the majority of the data points are clustered around the zero lines.

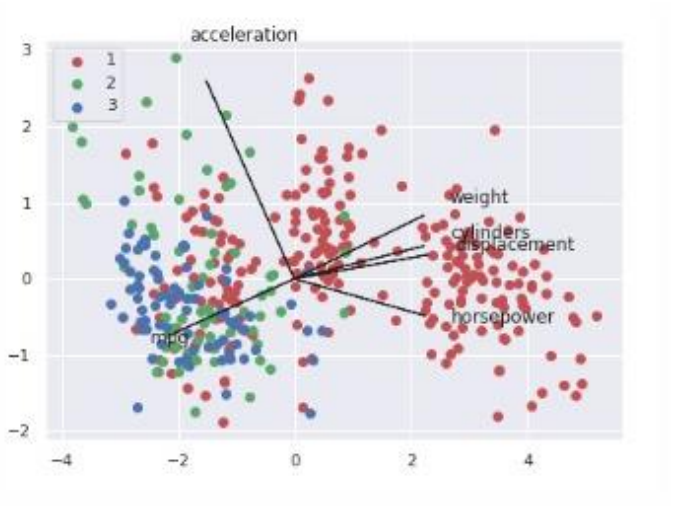

Fig 11. Biplot

PCA Results

PCA's primary objective is to decrease the dataset's dimension. The eigenvector matrix is used to condense the n x p dataset (A). After using PCA, the following eigenvector matrix is produced: -

$$A=\begin{bmatrix} -0.39 & 0.14 & 0.19 & 0.85 & 0.25 & -0.053 & -0.051 \\ -0.41 & -0.04 & -0.18 & 0.4 & -0.62 & 0.21 & -0.45 \\ 0.43 & -0.03 & -0.1 & 0.3 & -0.082 & 0.05 & 0.84 \\ 0.41 & -0.26 & -0.11 & 0.013 & -0.63 & 0.57 & -0.19 \\ 0.41 & -0.14 & -0.29 & 0.12 & 0.34 & 0.74 & -0.22 \\ -0.27 & -0.75 & -0.51 & 0.086 & 0.15 & 0.27 & 0.016 \\ -0.29 & 0.58 & -0.75 & -0.023 & -0.1 & -0.051 & 0.077 \end{bmatrix}$$

$$\lambda=\begin{bmatrix} 4.8005 \\ 0.7304 \\ 0.2591 \\ 0.1254 \\ 0.0633 \\ 0.0363 \end{bmatrix}$$

The first 2 principal components are given by

Thus, the first 2 principal components are given by

Z1 = -0.39X1-0.41X2+0.43X3+0.41X4+0.41X5-0.27X6-0.29X7

Z2 = 0.14X1-0.04X2-0.03X3-0.26X4-0.14X5-0.75X6+0.58X7

Control Charts

A well-known technique for monitoring multivariate data is the hoteling T2 control chart. When the effects of several variables are not independent of one another or when there is a correlation between them, multivariate control charts are used. If the variables are linked, multivariate control charts are used to determine whether the process is under control or not. Multivariate control, in addition to the features listed above, can be used to examine the stability of a process or system while taking multiple univariate factors into account at the same time. The following figures show the out-of-control points:

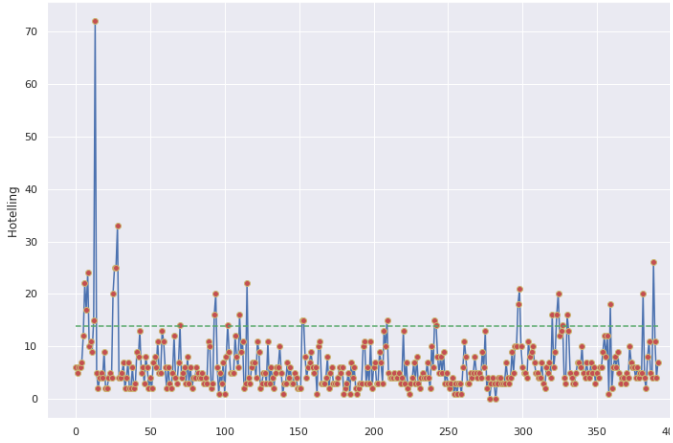

Fig 12. Hotelling T2 test Chart

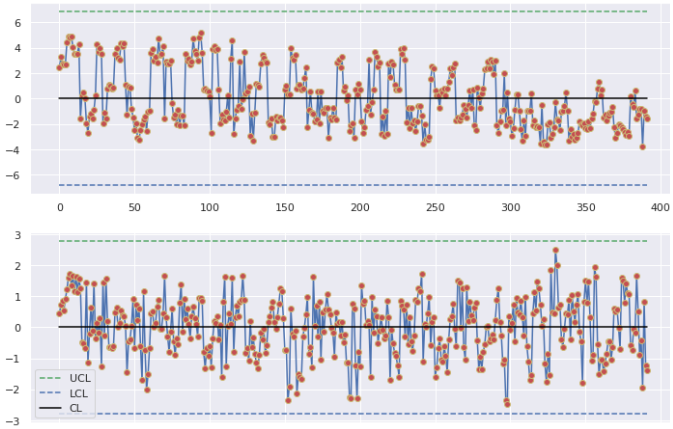

Fig 13. Control Charts

VI. Classification Results

In this study, we classified data using four classifiers: a gradient boosting, K nearest neighbors, logistic regression, and random forest classification. After analyzing the results, the effect of PCA on four main classifiers is investigated. The models are trained and tested on the initial data set, the updated data set, and the first three PCs data set. Confusion matrices and Receiver Operating Characteristic (ROC) curves are used to display the precision and recall outcomes. Confusion matrices are used to determine where the algorithm has difficulty distinguishing between classes. The model's predictions are shown on the x-axis, while the true label is shown on the y-axis. The remaining confusion matrices can be found in the accompanying Jupiter notebook. The confusion matrix outlines the true positives (TP), false positives (FP), and true negatives (TN) predicted in the table for each class. It is commonly known that all three algorithms anticipate the same result on the original data set. Each confusion matrix's accuracy reflects the same result (i.e., sum along the diagonal).

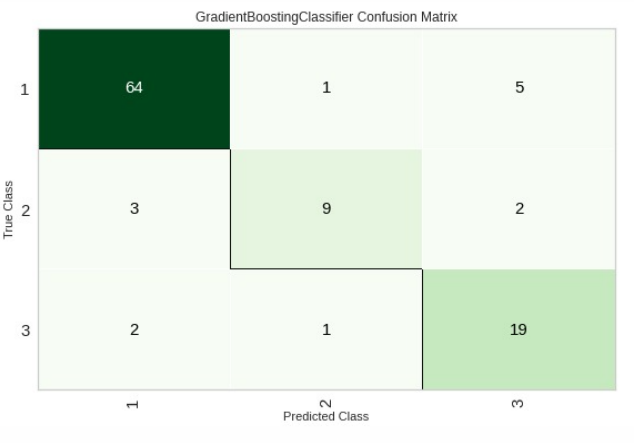

Fig 14. GBC confusion matrix

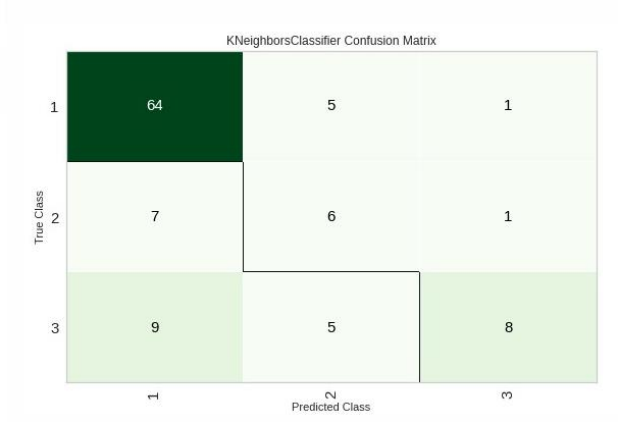

Fig 15. Knn Confusion matrix

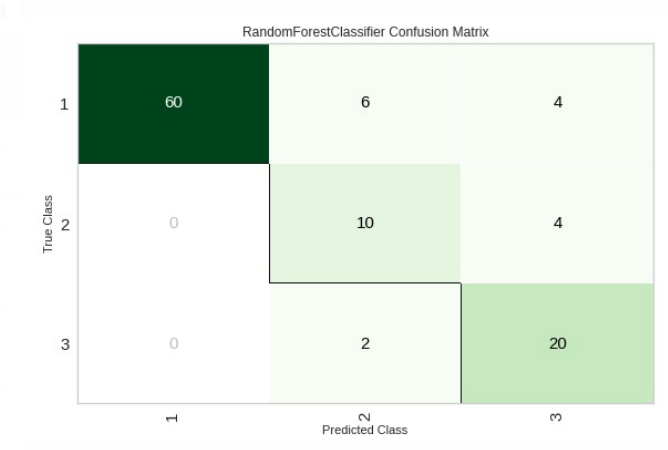

Fig 16. Random Forest Classification confusion matrix

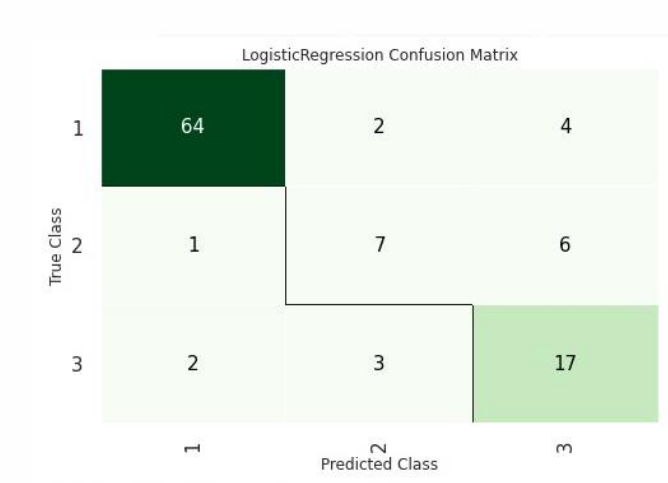

Fig 17. Logistic Regression Confusion matrix

Each confusion matrix's accuracy reflects the result. F-scores, which take into account both precision and recall, are used to determine an algorithm's performance and to compare the accuracy of various classifiers. The accuracy scale also supports the assertion that GBC is superior to the other algorithms: given the initial dataset, logistic regression has a 76% accuracy, GBC has an 80.95% accuracy, and Random Forest Classifier has a 78.52% accuracy.

On the following page, the decision boundaries of the original dataset for the classification methods are shown. Each of the four datasets has a total of seven features. The decision border position where this misclassification occurs can be plotted. The Logistic Regression Algorithm was discovered to be more accurate in classifying the training and test datasets. LR has the highest accuracy of all because it has the fewest mixed-up class types.

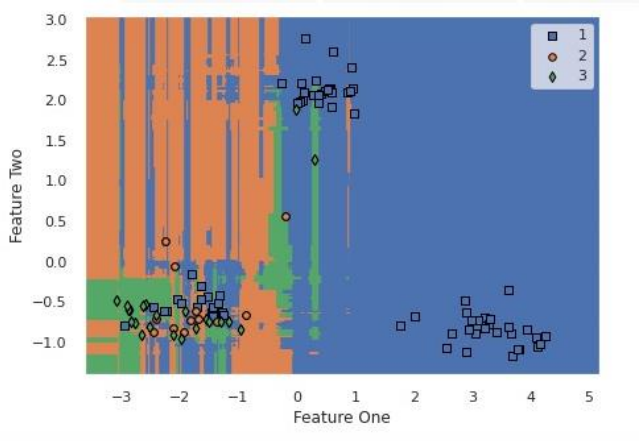

Fig 18. GBC Decision Boundary

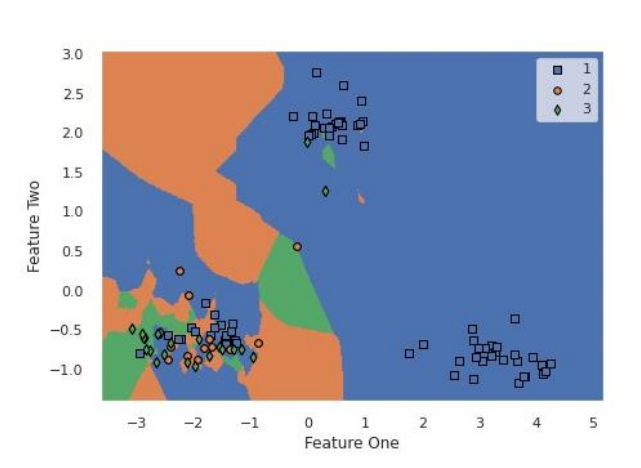

Fig 19. K-nn Decision Boundary

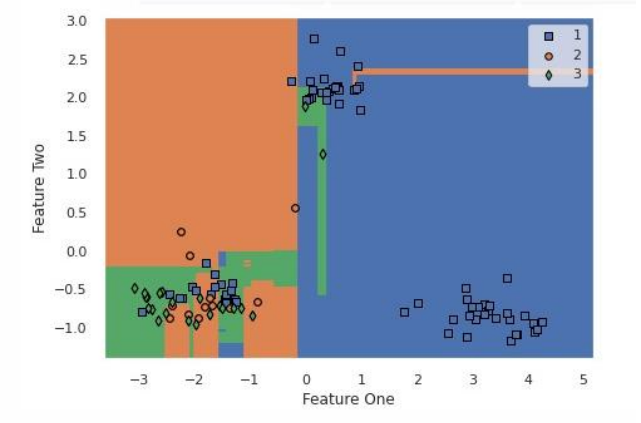

Fig 20. Random Forest Decision Boundary

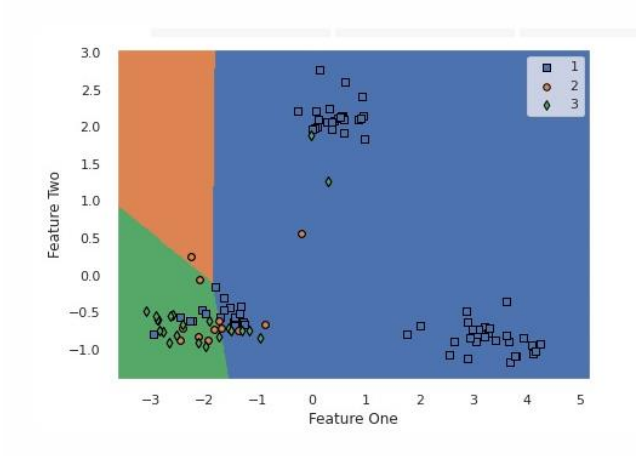

Fig 21. Logistic regression Decision Boundary

The algorithm's final step is to compute Receiver Operating Characteristic curves (ROC curves). The ROC curves for various classifiers for original data are depicted in the figures below. The blue line belongs to Class 1, the orange line to Class 2, the green line to Class 3, and the dotted lines are the micro and macro average of ROC curves that have more than 90% area, implying that the original data set is more consistent than others. The other ROC curves can be interpreted in the same way.

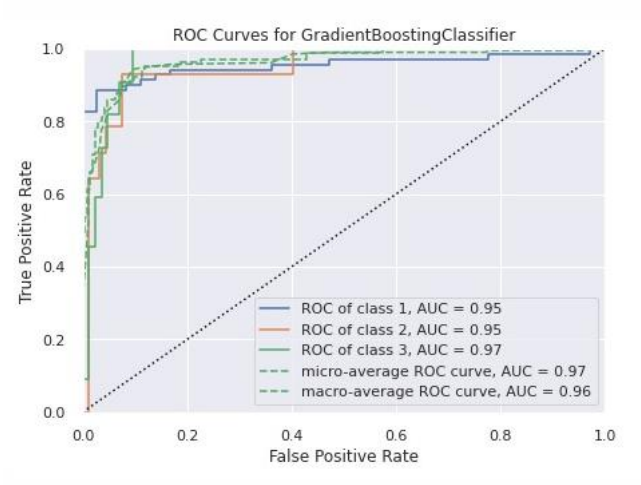

Fig 22. GBC ROC curves

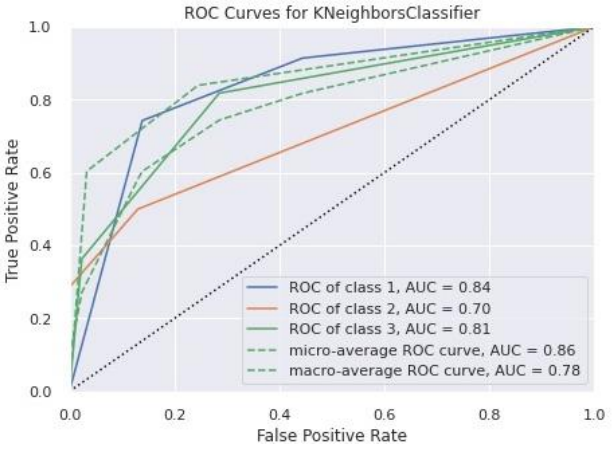

Fig 23. KNN ROC curves

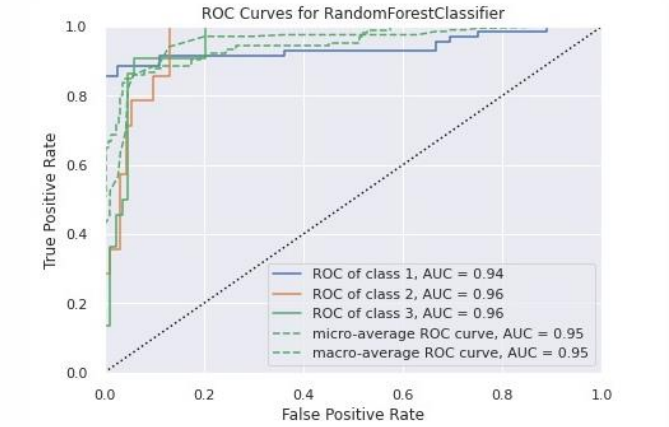

Fig 24. Random Forest ROC curves

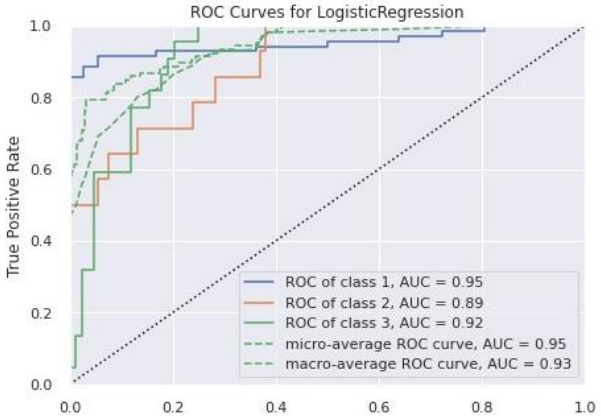

Fig 25. Logistic Regression ROC curves

VII. Explained AI with Shapley Values

Shapley-valued machine learning models are described in this section. A technique from a cooperative game theory known as Shapley values is popular because it has the desirable property of allowing humans to understand the solution's outcomes. In contrast, the "black box" in machine learning refers to a situation in which even the technology's creators are unable to explain how an AI came to a particular conclusion. To fine-tune the model and forecast the miles per gallon for the cars, we combined the Random Forest Classifier with principal component analysis. Features are arranged according to the total SHAP value and its magnitudes across all samples in the SHAP summary graphic. Each primary component's impact is depicted in the prediction visualization by its size.

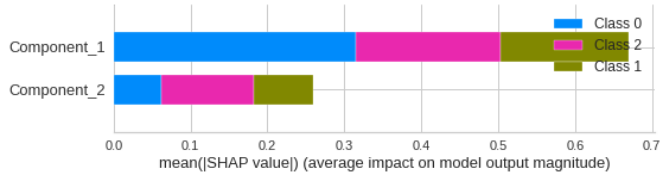

Fig 26. Summary Plot

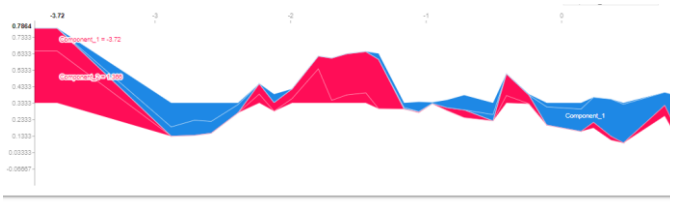

Fig 27. Continuous Reason Plot

VIII. Conclusion

In conclusion, the paper can be split into two parts. A vehicle dataset with seven different attributes is reduced in dimension using principle component analysis in the first segment, and the findings show that the first two PCs account for more than 80% of the data variability. To spot out-of-control areas and assess PC performance, control charts were developed. In the second phase, the data from the original dataset, the first two PCs, and the full PC dataset were categorized into Class 1, Class 2, and Class 3 using four different machine learning techniques (GBC, K-NN, Random Forest Classification, and Logistic Regression). The confusion matrix was used to assess the classification's accuracy and precision, and it was found that the GBC outperformed the others classification methods. Finally, several interpretation plots are generated using explainable AI Shapley values to improve the model's interpretability.

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