FOUNDATIONS OF DATA SCIENCE

Assignment Report
CS F320 (FoDS)

Submitted By:

Siddhant Panda	2020A7PS0264H
Dev Bansal	2020A7PS2051H
Kartikey Goel	2020A7PS2070H

Assignment 2-A: Implementing PCA from Scratch and Applying it to Car Data

Principal component analysis, or PCA, is a dimensionality reduction method that is often used to reduce the dimensionality of large data sets by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

Data Understanding and Representation

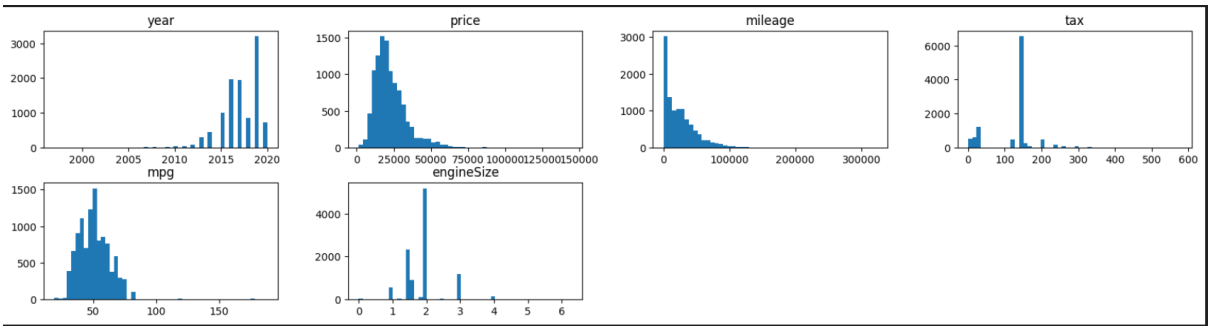
This task required us to load the dataset into a pandas dataframe and understand the features present in it.

- This dataset consists of 9 columns(features) in total. The target variable is “Price” which is a function of the other 8 variables given in the dataset.
- Out of the 8 variables (except the target variable), 3 are categorical variables: model, transmission, fuelType.
- The matrix form representation of the dataset is shown below:

	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
0	A1	2017	12500	Manual	15735	Petrol	150	55.4	1.4
1	A6	2016	16500	Automatic	36203	Diesel	20	64.2	2.0
2	A1	2016	11000	Manual	29946	Petrol	30	55.4	1.4
3	A4	2017	16800	Automatic	25952	Diesel	145	67.3	2.0
4	A3	2019	17300	Manual	1998	Petrol	145	49.6	1.0
...	...	...	...	...	...	...	...	...	...
10663	A3	2020	16999	Manual	4018	Petrol	145	49.6	1.0
10664	A3	2020	16999	Manual	1978	Petrol	150	49.6	1.0
10665	A3	2020	17199	Manual	609	Petrol	150	49.6	1.0
10666	Q3	2017	19499	Automatic	8646	Petrol	150	47.9	1.4
10667	Q3	2016	15999	Manual	11855	Petrol	150	47.9	1.4

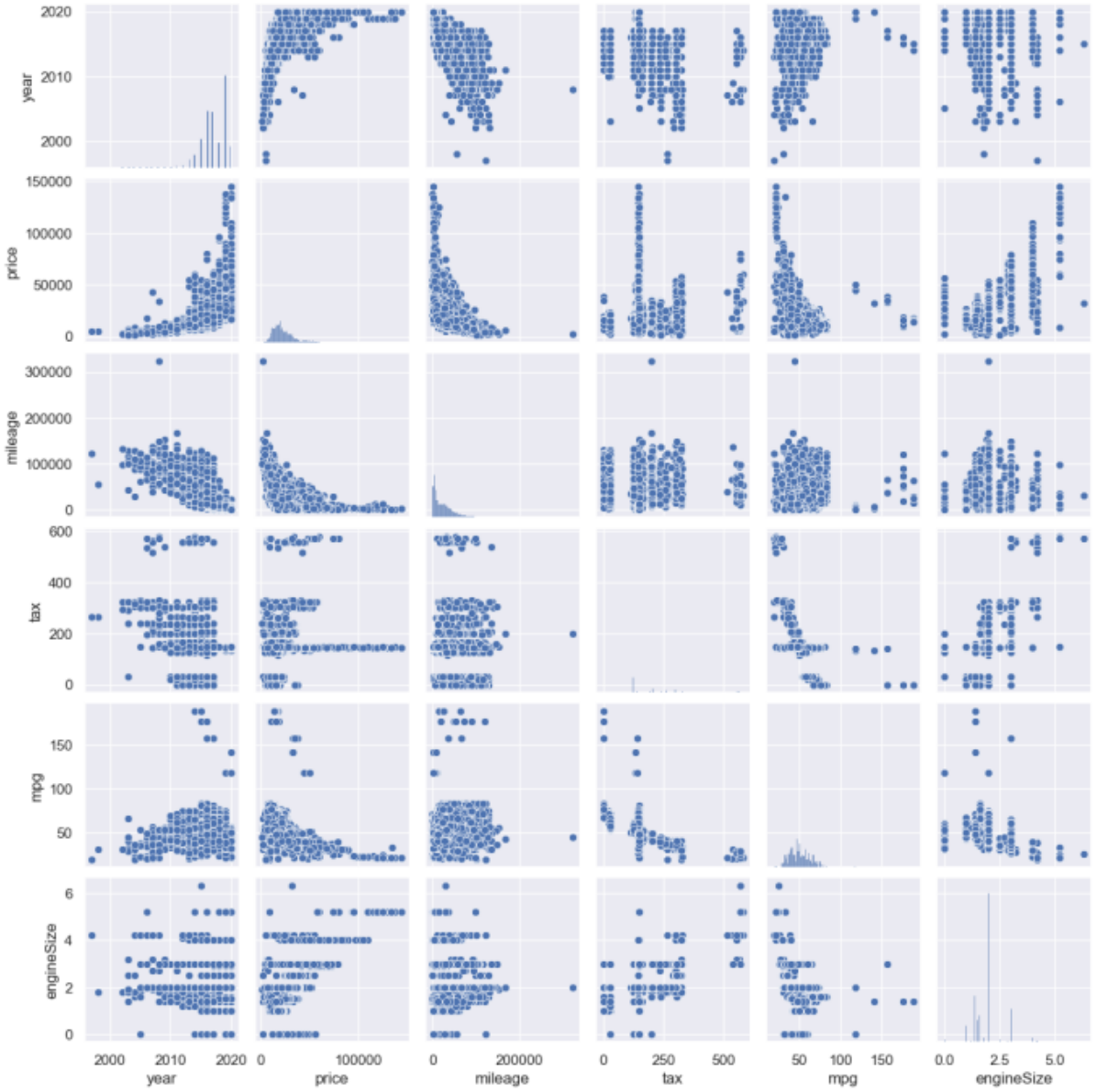
10668 rows × 9 columns

To get an understanding of what we were dealing, we drew various graphs as follows:



Each of these graphs represents the distribution of the values taken by each of the 6 continuous features in the dataset.

Next, we drew a graph for each possible pair of the 6 features, which would result in 6x6 = 36 graphs, attempting to visualize any possible correlation between them which is shown below:



- Next, we dropped all the categorical features from the dataset, namely ‘model’, .‘transmission’, ‘price’ and ‘fuelType’ because PCA is only suitable for continuous features.It tries to minimize variance (=squared deviations). The concept of squared deviations breaks down when you have binary variables.

Implementing PCA using the covariance matrix

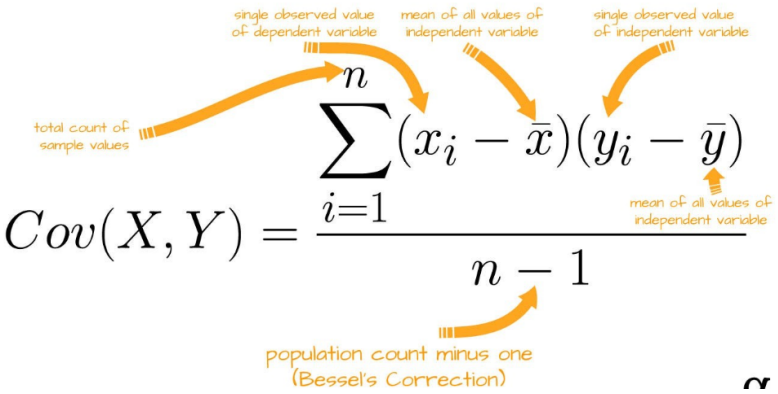
The aim of this step is to understand how the variables of the input data set are varying from the mean with respect to each other, or in other words, to see if there is any relationship between them. Because sometimes, variables are highly correlated in such a way that they contain redundant information. So, in order to identify these correlations, we compute the covariance matrix.

To achieve this, firstly, normalization was performed on the remaining features and then the features were centered around the mean.

	year	mileage	tax	mpg	engineSize
year	1.000000	-0.789667	0.093066	-0.351281	-0.031582
mileage	-0.789667	1.000000	-0.166547	0.395103	0.070710
tax	0.093066	-0.166547	1.000000	-0.635909	0.393075
mpg	-0.351281	0.395103	-0.635909	1.000000	-0.365621
engineSize	-0.031582	0.070710	0.393075	-0.365621	1.000000

Since we couldn't get any 2 variables with a high covariance value between them (assumed threshold is 90%), we were unable to remove any features based on the covariance matrix alone. Hence, we will be proceeding to the next method to perform PCA, which is through the concept of eigenvalues and eigenvectors.

The formula used for calculating the covariance matrix is:



Eigenvalue-Eigenvector equation and solving for principal components

1. If A is a square matrix and V is a column vector such that:

$$Av = \lambda v$$

then V is the eigenvector of A, and Lambda is the eigenvalue of A.

2. We used NumPy's *linalg.eig* function to compute the eigenvalues and eigenvectors of the previously calculated covariance matrix.
    - eigenvectors of the Covariance matrix are actually *the directions of the axes where there is the most variance* (most information) and that we call Principal Components.

- eigenvalues are simply the coefficients attached to eigenvectors, which give the *amount of variance carried in each Principal Component*.

3. Now we calculate the explained variance using Eigenvalues:

```
Eigenvalues:  [0.46259565 0.30993605 0.12400512 0.06310454 0.04035864]
Eigenvectors:
 [[-0.46373634  0.48249022 -0.26369364 -0.10806712  0.68624992]
 [ 0.48575669 -0.46716498  0.0732536   0.17892982  0.7130325 ]
 [-0.43356476 -0.43702648  0.49512711 -0.60322362  0.10954391]
 [ 0.5485194   0.22908421 -0.24802587 -0.76491314 -0.00616002]
 [-0.24522869 -0.55271001 -0.7864044  -0.08564328 -0.09277865]]
Eigenvectors shape:  (5, 5)
```

4. Next, we calculated the cumulative explained variance, which took the values as follows:

   [46.25956452, 77.25316956, 89.65368173, 95.96413556, 100.]

   Each of these values is the percentage of the variance explained by the principal components. Since the industrial norm is to take the features which lead to a cumulative explanation of at least 95% of the variance of the data, we will take the first **four** principal components.

   Therefore, the value of k is four

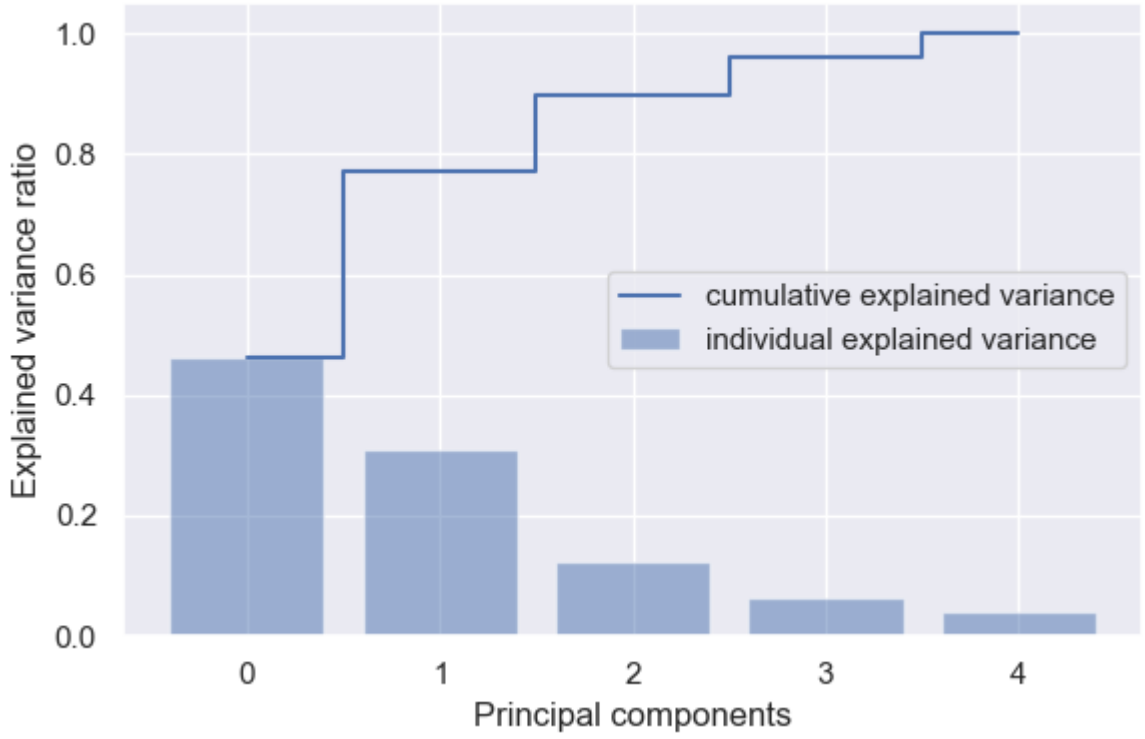
Sequential Variance Increase

Explained Variance: In the context of PCA, explained variance refers to the amount of variability in the original data that is "explained" by a specific principal component. Explained variance is often expressed as a ratio or percentage of the total variance.

Explained variance can be represented as a function of the ratio of related eigenvalues and the sum of eigenvalues of all eigenvectors.

$$\frac{\lambda_i}{\lambda_1+\lambda_2+\ldots+\lambda_n}$$

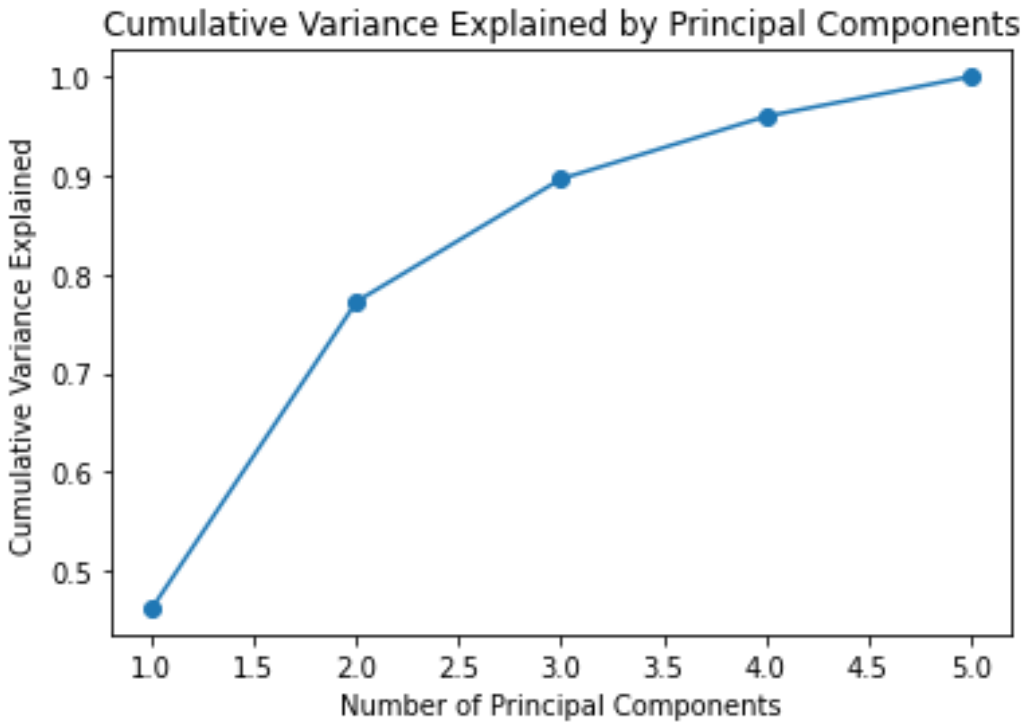
The graph for the same is given below:



The graph tells the first principal component captures about 44% of the information present in the original mathematical space, i.e., explains ~ 44% of the variation in the data.

On taking the first two components together, the total explained variation is close to 78% and when taking the first three components, the cumulative explained variation is ~ 90% and ~ 99% of the variation is captured by the first four principal components.

The other principal components are insignificant as they do not make any contribution, suggesting these do not explain much information and hence can be dropped.



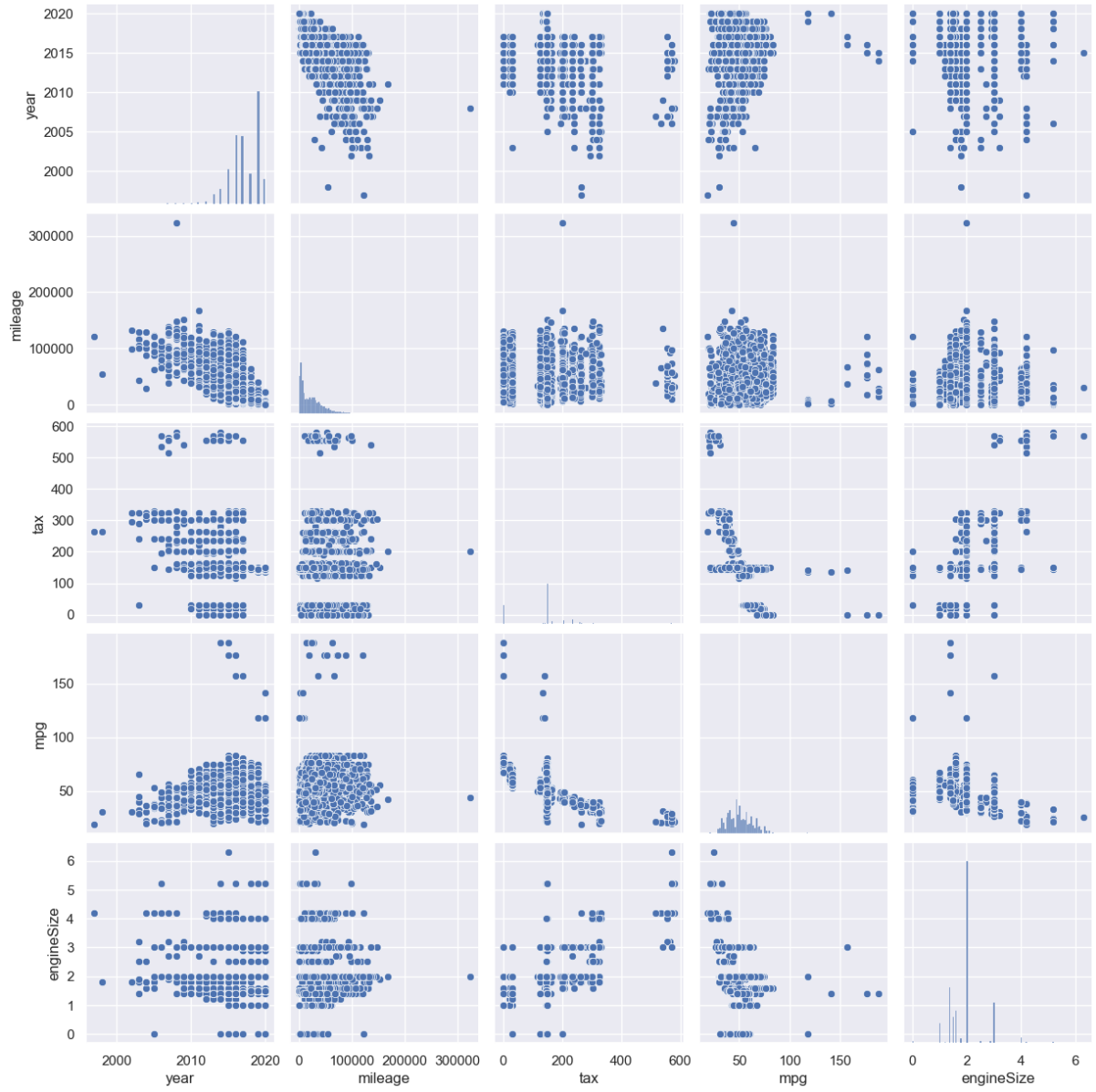
Analysis:

Initial Components: Initially, with a small number of principal components, the cumulative explained variance increases rapidly. This indicates that these initial components capture a significant portion of the total variability in the data.

Diminishing Returns: As more components are added, the rate of increase in cumulative explained variance starts to diminish. Each additional component contributes less to the overall explanation of variance.

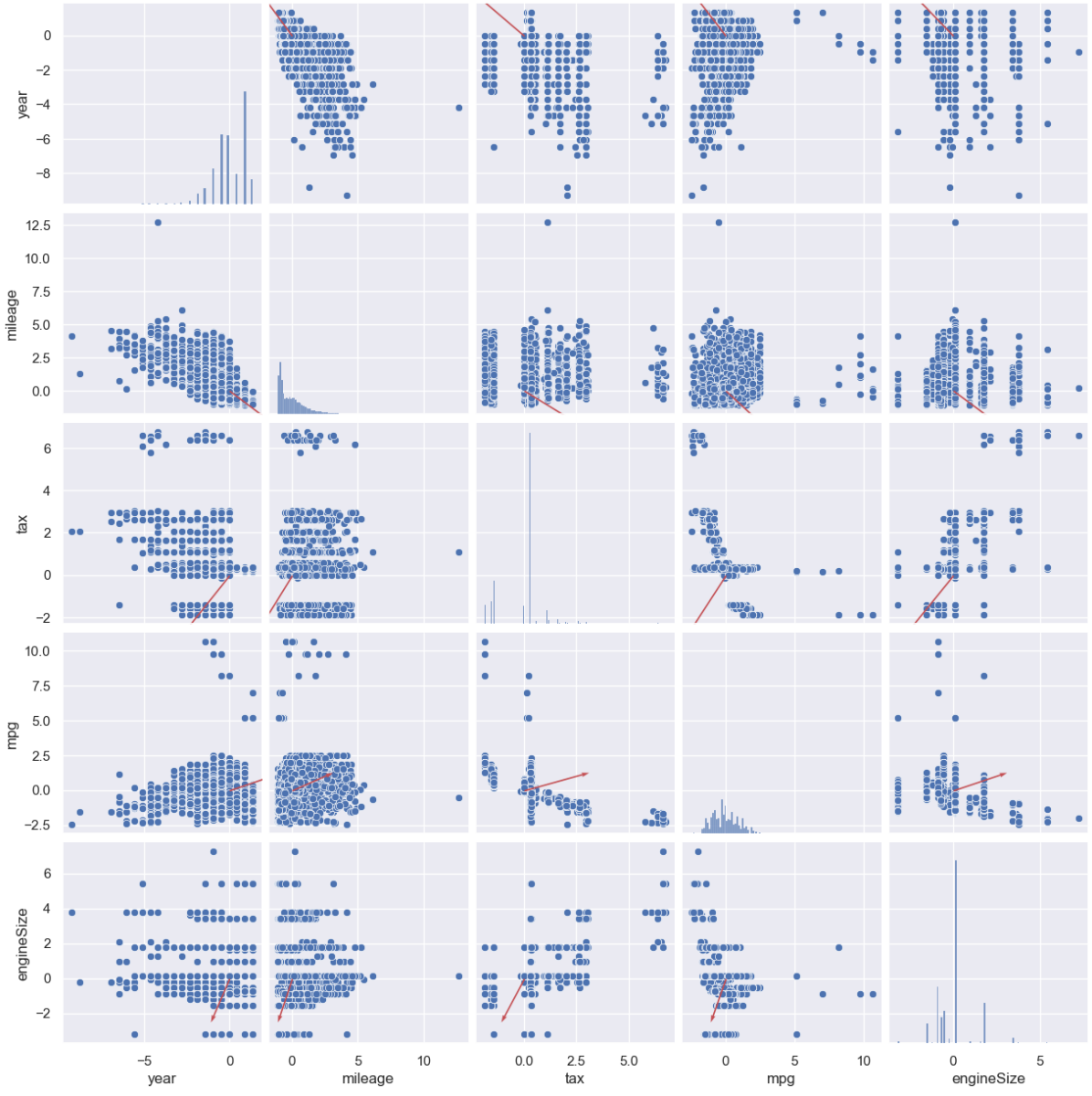
Visualization using pair plots

This task required us to project the principal components on the pair plots, here, we have five principal components, which means 25 pair plots in number.



Analysis:

- Initial observation of the plotted principal components might suggest a linear correlation. However, this visual inference of collinearity doesn't accurately represent the relationship among these components.
- In a three-dimensional view, it becomes clear that the principal components form an angle between them and are actually perpendicular to each other in a higher-dimensional space.
- To confirm their orthogonality, we plan to incorporate the third principal component into these graphical representations. Introducing the third principal component into the graphs aims to validate further and reinforce the concept of orthogonality among the principal components.



Conclusions and Interpretations

- In PCA, it is crucial to exclude categorical features because the method is specifically designed to minimize variance by considering squared deviations. However, the notion of squared deviations becomes problematic when dealing with categorical variables.

- In this dataset, approximately 46% of the variance is accounted for by the initial principal component. As we consider more principal components, the cumulative variance explained increases: the first two components cover about 77%, the first three components explain roughly 89%, and the first four components encompass around 95% of the variance. Ultimately, when all principal components are considered together, they collectively explain the entire 100% of the variance in the dataset.

- High-dimensional datasets often suffer from the curse of dimensionality, leading to increased computational complexity and the risk of overfitting in machine learning models. PCA addresses this by allowing for the reduction of dimensionality. The transformation provided by PCA enables one to retain most of the dataset's variability in a smaller number of dimensions, reducing noise and redundancy.

- For this experiment, by performing PCA on the given dataset, we got the optimal value of k to be **Four**.

Assignment 2-B: PCA Analysis and Determining Optimal Number of Components

The objective of this assignment is to conduct Principal Component Analysis (PCA) on the 'Hitters.csv' dataset, determine the optimal number of components for efficient prediction using Mean Squared Error (MSE) or Root Mean Squared Error (RMSE), and test the most efficient model.

Exploratory Data Analysis (EDA):

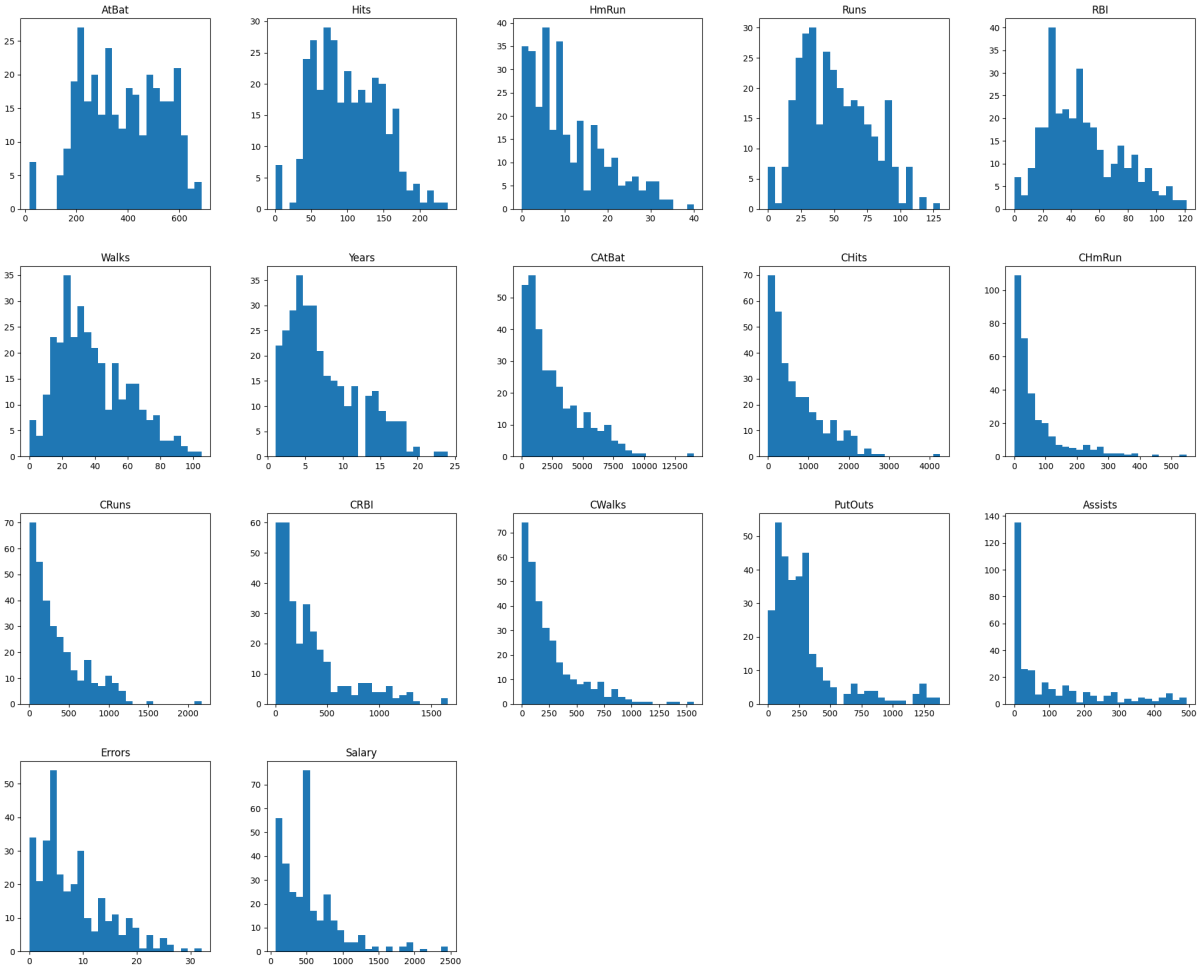
- Data Examination:
  - Identified NULL values only in the 'salary' column among all features.
  - 'Salary' column contained continuous data.

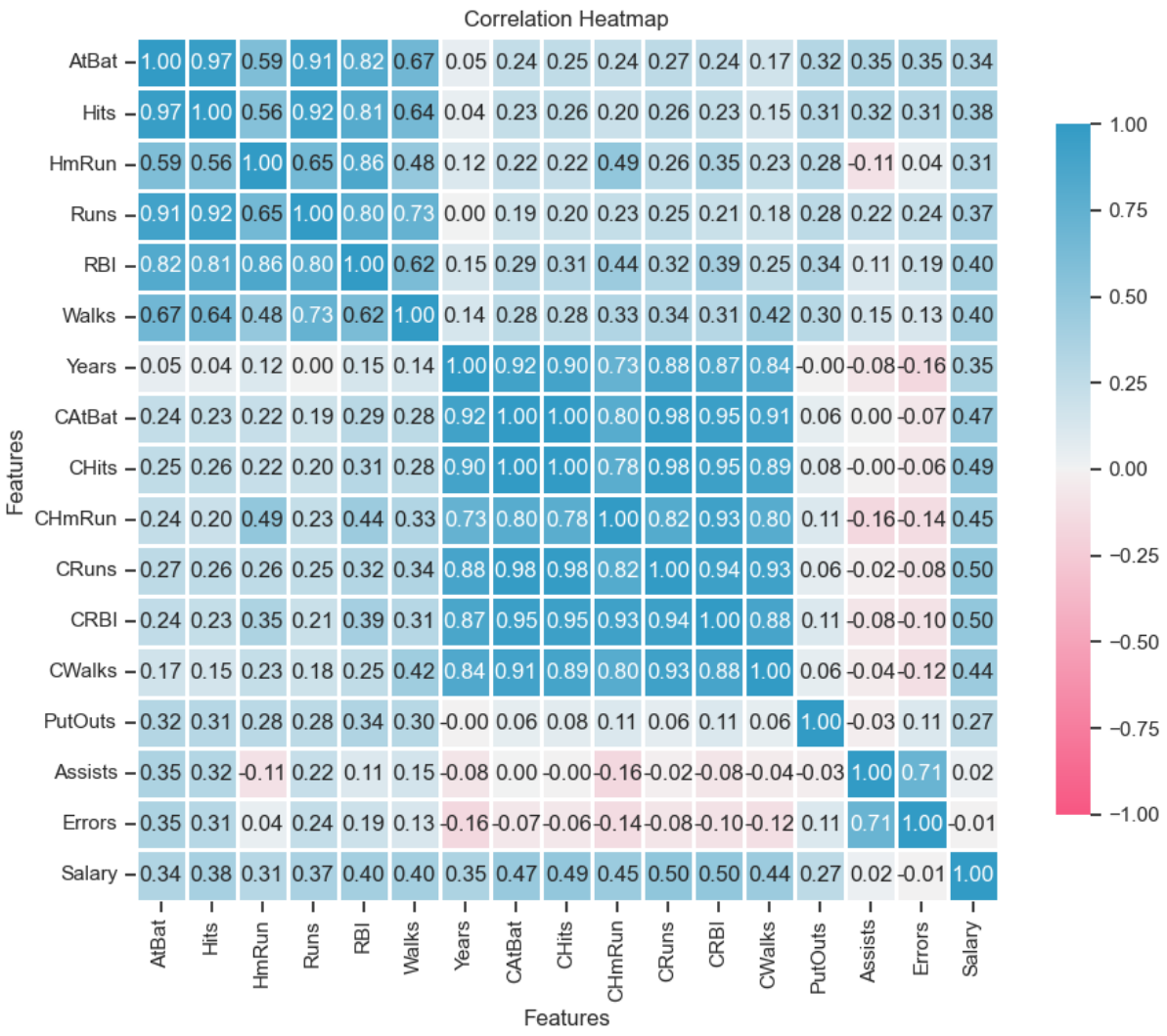
- Handling NULL Values:
  - Specifically addressed NULL values in the 'salary' column.
  - Imputed the NULL values by replacing them with the mean value calculated from the existing non-NULL values in the 'salary' column.

- EDA Outcomes:
  - After handling NULLs, proceeded with exploratory analysis.
  - Explored relationships, distributions, and statistical summaries post NULL value imputation in the 'salary' column.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 322 entries, 0 to 321
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   AtBat       322 non-null    int64
 1   Hits        322 non-null    int64
 2   HmRun       322 non-null    int64
 3   Runs        322 non-null    int64
 4   RBI         322 non-null    int64
 5   Walks       322 non-null    int64
 6   Years       322 non-null    int64
 7   CAtBat      322 non-null    int64
 8   CHits       322 non-null    int64
 9   CHmRun      322 non-null    int64
 10  CRuns       322 non-null    int64
 11  CRBI        322 non-null    int64
 12  CWalks      322 non-null    int64
 13  League      322 non-null    object
 14  Division    322 non-null    object
 15  PutOuts     322 non-null    int64
 16  Assists     322 non-null    int64
 17  Errors      322 non-null    int64
 18  Salary      322 non-null    float64
 19  NewLeague   322 non-null    object
dtypes: float64(1), int64(16), object(3)
memory usage: 50.4+ KB
```

- We then move ahead with histogram plots of the non-categorical variables to understand the data better.



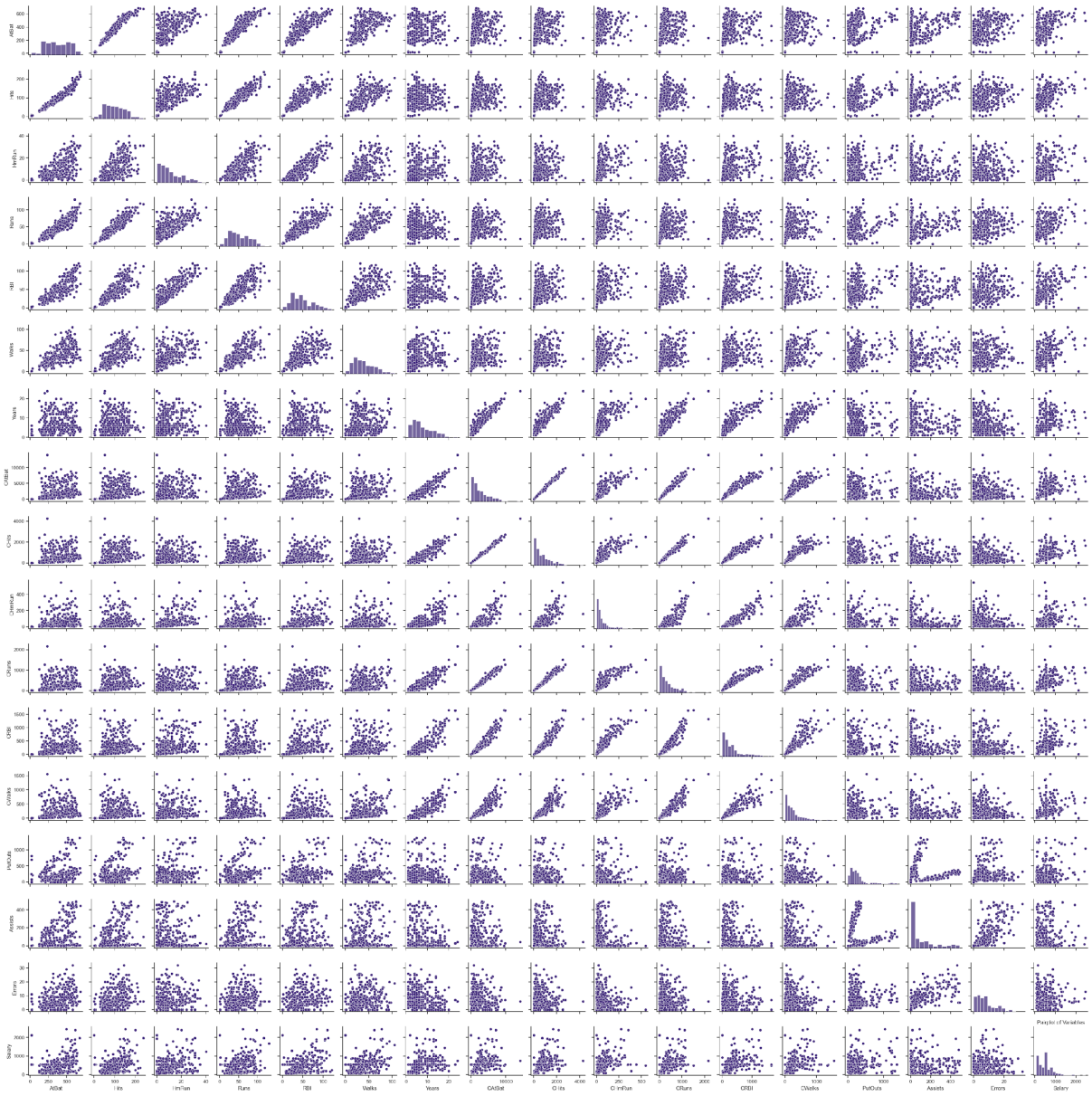


- Distribution Analysis:
  - Noticeable deviation from a normal (Gaussian) distribution specifically observed in the ASSISTS plot.
  - This deviation signals a significant skewness in the data, hinting at potential outliers influencing the distribution.

- Potential Outliers Impacting Skewness:
  - Skewness in the ASSISTS plot is likely attributed to the presence of outliers within the data.

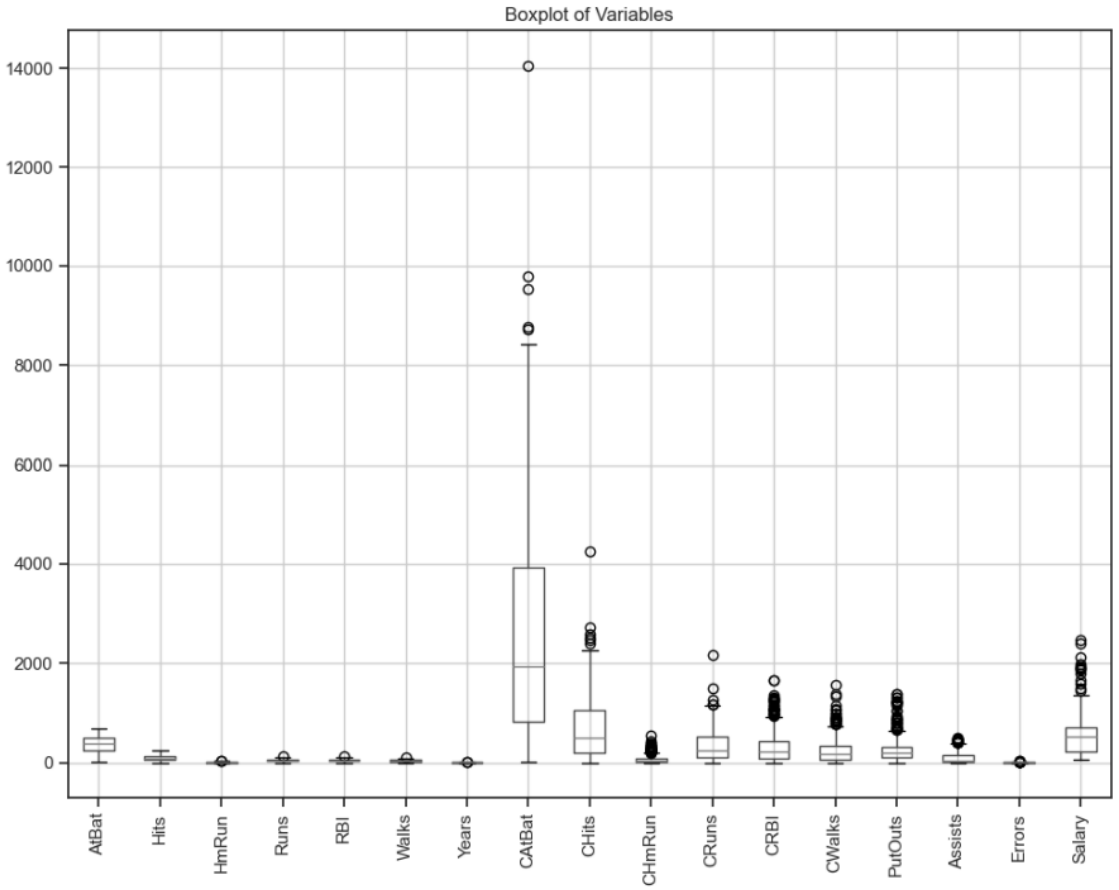
- Similarity in Distributions:
  - Moreover, there appears to be a resemblance in the distributions of CAtBat, CHits, CHmRUN, CRuns, CRBI, and CWalks.
  - Notable similarity or high correlation patterns identified in these variables' distributions.

- Planned PCA Analysis:
    - Intention to further investigate the potential correlations observed among these variables through Principal Component Analysis (PCA).
    - PCA analysis aims to confirm and quantify the relationships and similarities observed in the distributions of these variables.



As observed in our correlation heatmap, our pairplot seems to validate the same. Highly correlated parameters exhibit a clearly noticeable linear pattern.

Identifying the outliers :



The above is a box plot that helps in identifying the outlying data points, which might have a negative impact on any further data analysis that we might perform with the dataset.

PCA Analysis:

- Normalization Process:
    - Implemented normalization on the remaining features for several reasons:
    - Minimized redundant data in the dataset.
    - Ensured consistent data throughout the database.

- Facilitated a more adaptable database design.
- Strengthened database security measures.
- Enhanced execution speed and efficiency.
- Improved overall database organization.

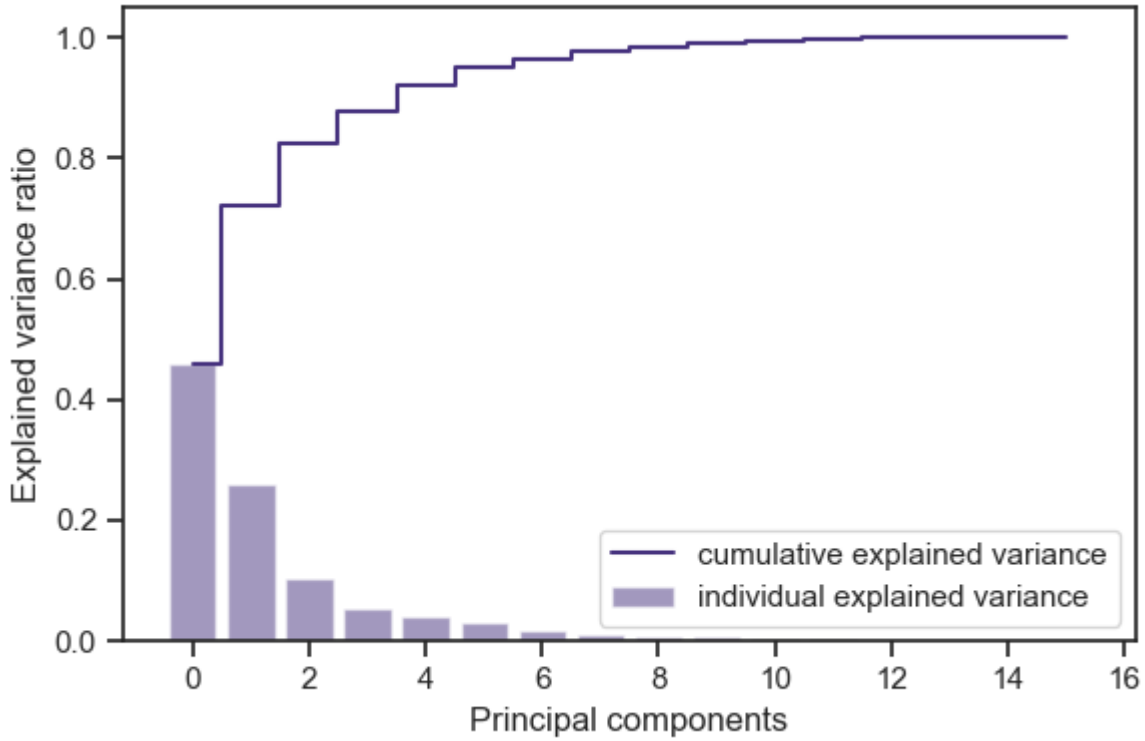
- Covariance Matrix Computation :
  - Computed the covariance matrix from the centered and normalized dataset.
  - The resulting covariance matrix was a symmetrical 16x16 matrix, representing the relationships between the remaining 16 independent continuous variables.

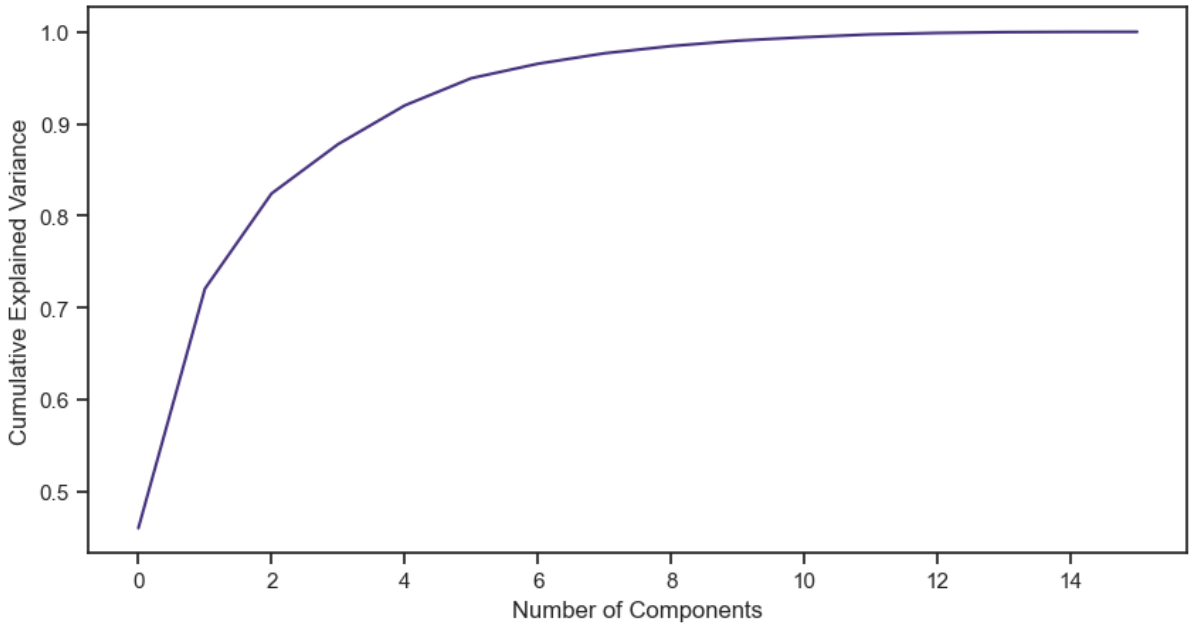
- Eigenvalue and Eigenvector Calculation :
  - Utilized NumPy to compute the eigenvalues and eigenvectors.
  - These eigenvalues and eigenvectors are pivotal in characterizing linear transformations, finding applications in diverse fields from geology to quantum mechanics.

- Significance of Eigenvalues and Eigenvectors:
  - Eigenvalues and eigenvectors serve essential roles in systems involving feedback loops or iterative transformations.
  - The largest eigenvalue notably governs the long-term behavior of the system, determining the steady state, especially after multiple iterations of the linear transformation.

- Explained Variance Calculation:
  - Calculated the explained variance based on the eigenvalues obtained.
  - This step provided insight into how much variance each principal component explains within the dataset.

- Cumulative Explained Variance:
    - Determined the cumulative explained variance percentages:
    - These values indicated the cumulative contribution of each principal component to the overall variance in the dataset.
    - Ranged from 46.04% to 100%, showcasing the cumulative impact of each subsequent principal component on the dataset's variance capture.



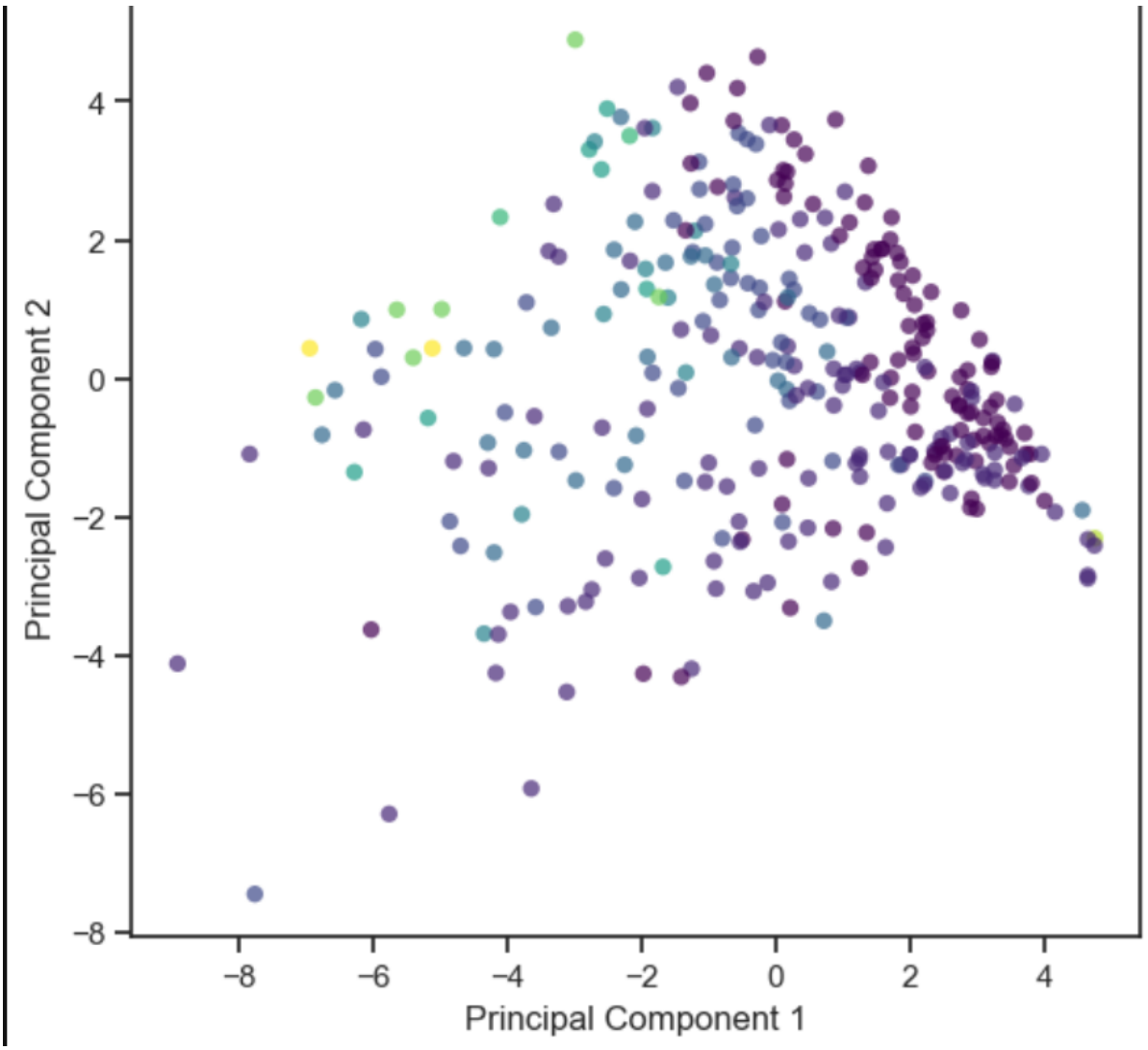


Elbow Point: The point where the curve starts to flatten is often referred to as the "elbow point." It represents the optimal number of principal components to retain. Beyond this point, adding more components provides marginal gains in explained variance.

- In our case, the elbow point appears to be at number of components = 5

Projection onto best PCA components

- Next, we projected the data points onto a graph whose axes are represented by the first two principal components since they would provide the most intuitive understanding of the data analysis.



Model Training and MSE/RMSE Calculation:

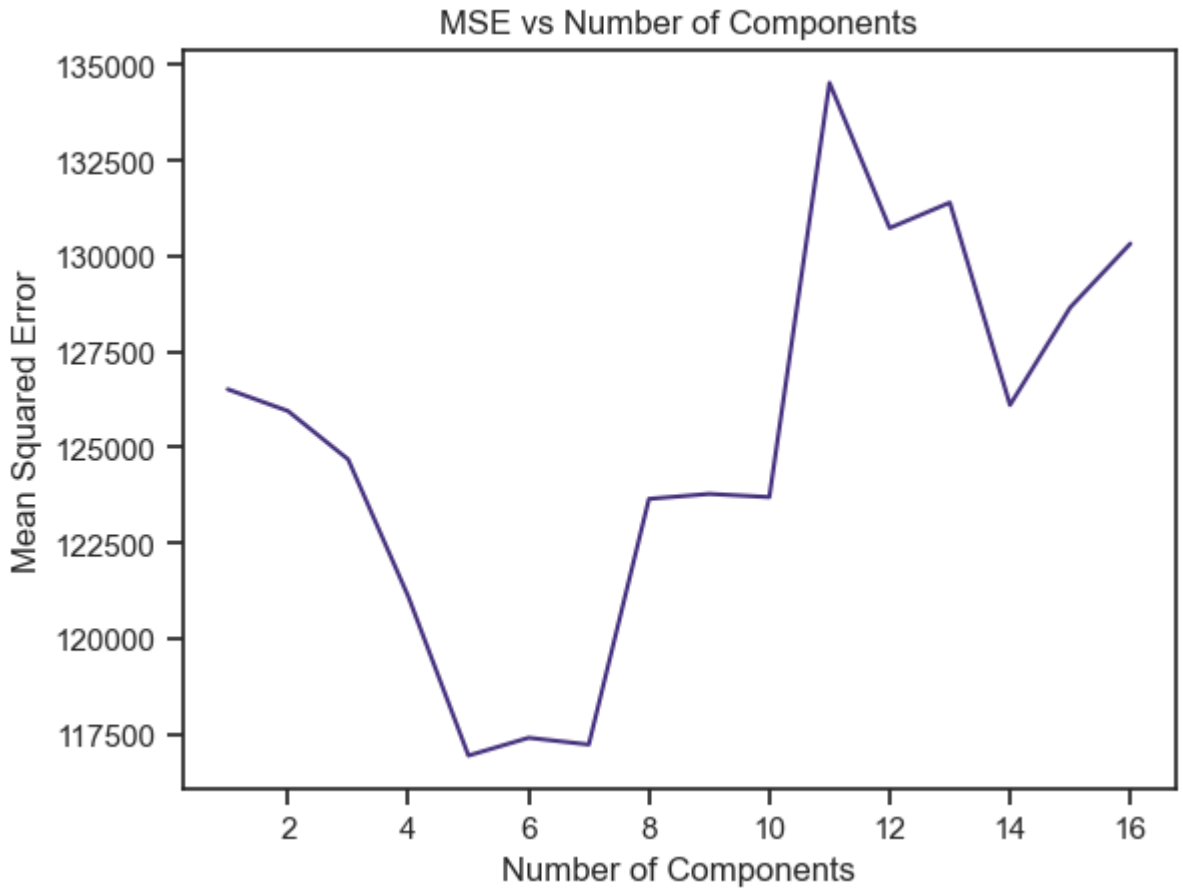
- Dataset Split:
  - Segmented the normalized dataset into training and testing sets, maintaining an 80-20 split ratio.

- Model Building Approach:
  - Developed a sequence of 16 models using stochastic gradient descent methodology.
  - Each model iteration followed a distinct strategy:
    - Utilized a varying number of principal components as independent features for training.
    - Initiated with one principal component and systematically increased the count up to 16.

- Examined the influence of principal component quantity on the models' predictive performance.

```
Number of components: 1, Mean Squared Error: 126522.32931534418
Number of components: 2, Mean Squared Error: 125953.9622313186
Number of components: 3, Mean Squared Error: 124689.51338093201
Number of components: 4, Mean Squared Error: 121104.71034098488
Number of components: 5, Mean Squared Error: 116946.20339750305
Number of components: 6, Mean Squared Error: 117411.31574001852
Number of components: 7, Mean Squared Error: 117234.79658105991
Number of components: 8, Mean Squared Error: 123656.78933846092
Number of components: 9, Mean Squared Error: 123786.85048416484
Number of components: 10, Mean Squared Error: 123703.24580459
Number of components: 11, Mean Squared Error: 134532.6581193975
Number of components: 12, Mean Squared Error: 130737.63463464964
Number of components: 13, Mean Squared Error: 131401.52160775455
Number of components: 14, Mean Squared Error: 126111.12402610519
Number of components: 15, Mean Squared Error: 128660.17917523271
Number of components: 16, Mean Squared Error: 130330.02039232652
```

The output provides the MSE for different numbers of components. The goal is to analyze this output and identify the number of components that yields the lowest MSE, which happens to be 5 components.



The graph confirms the same.

Testing the Most Efficient Model:

- Model Prediction Task:
  - Subsequent to previous task's results analysis, conducted prediction to assess the dataset's reliability and applicability.

- Model Construction Approach:
  - Employed stochastic gradient descent technique, integrating the first five principal components to build the predictive model.

```
poly = PolynomialFeatures(degree=2)
X_train_poly = poly.fit_transform(X_train_pca)
X_test_poly = poly.transform(X_test_pca)
```

- Regularization Implementation:
    - Introduced regularization techniques, specifically Lasso and Ridge regressions, to counteract potential issues of underfitting and overfitting within the models.

Lasso Regression Training:

```
lasso_model = Lasso()
lasso_model.fit(X_train_poly, y_train)
```

Ridge Regression Training:

```
ridge_model = Ridge()
ridge_model.fit(X_train_poly, y_train)
```

- Prediction Outcome:
    - Selected a random data point from the dataset and generated predictions.
    - Results of the prediction for the chosen point were as follows:

```
The predicted y value for the selected point using Lasso is 352.5755733147337
The predicted y value for the selected point using Ridge is 351.3540295417232
The actual y value for the selected point is 350.0
```

Conclusion :

- Optimal Principal Components Determination:
    - Identified the ideal count of principal components leading to the model with the minimal mean squared error, which stood at five.
    - Determined through the creation of a graphical plot:
    - Plotted mean squared error on the y-axis.
    - Plotted the number of principal components used to construct the model on the x-axis.

    - Observed a pivotal point, denoted as the knee point, occurring at the value of five on the x-axis. This infers that employing five principal components yielded the most effective model.

- Regularization Techniques Implemented:
    - Utilized stochastic gradient descent in conjunction with L1 Norm (Lasso) and L2 Norm (Ridge) for regularization purposes.

- Prediction Outcome:
    - Randomly selected a specific data point for prediction assessment.
    - The actual value of the target variable for the chosen point was 350.

- Predicted values:
    - Lasso prediction resulted in 352.57.
    - Ridge prediction yielded 351.35.

This detailed analysis involved determining the optimal principal components through graphical assessment, employing regularization techniques, and evaluating the predictive performance on a chosen data point, providing insights into the model's accuracy.