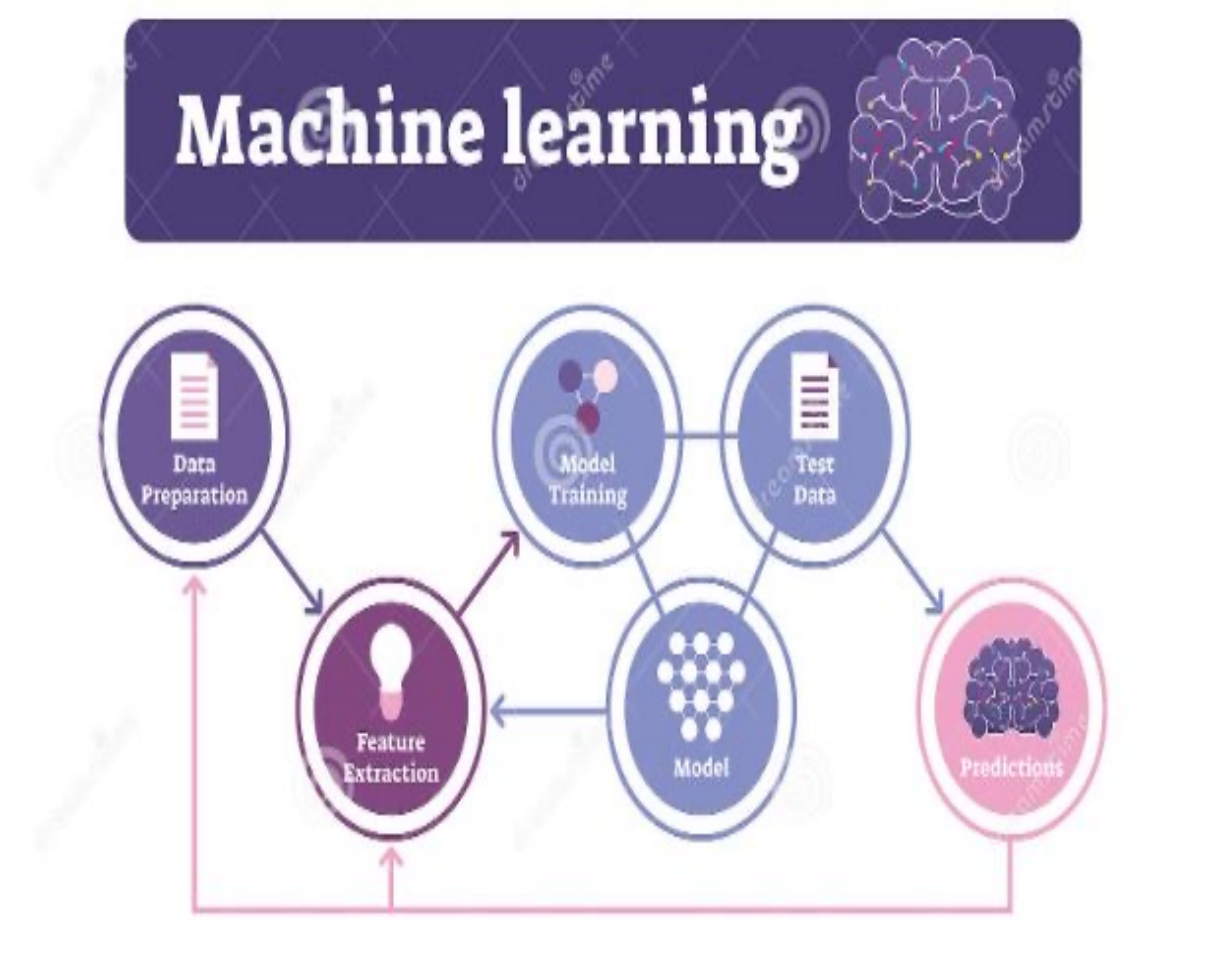
Used car finance fit and recommendation

- Harishankar Ganapathy
- Kunal Kumar
- Manisha Nallapati
- Pooja Vishwakarma
- Ravi Kavidayal

Agenda



Introduction

Used car price prediction

- Data cleaning and feature extraction
- Exploratory Data Analysis
- Model and Prediction

Customer Loan eligibility prediction

- Data cleaning
- Exploratory Data Analysis
- Model and Prediction

Car recommendation

Introduction

Our project aims to address the increasing demand for used cars in the market. According to market analysis, the demand for used cars has been steadily increasing in recent years. In terms of sales volume, the used car market in India sold approximately 4.4 million units in 2020, representing a 12.5% increase from the previous year. One of the primary drivers of this trend is the rising cost of new cars, which has led many consumers to opt for used vehicles instead. Additionally, the COVID-19 pandemic has had a significant impact on the automotive industry, with many people choosing to avoid public transportation and instead purchase their own personal vehicles.

The problem we are tackling is twofold: providing a reliable used car price prediction for sellers, and assisting customers in checking their loan eligibility, predicting their eligible loan amount, and recommending a suitable used car.

For sellers, we will analyse a variety of factors, including the vehicle's make, model, year, condition, mileage, etc. in order to provide an accurate price prediction for their used car. This will enable sellers to price their vehicles appropriately, leading to a smoother and more efficient sales process.

For customers, our solution will enable them to quickly and easily check their loan eligibility, predict the amount of loan they may qualify for, and receive personalized recommendations for used cars that fall within their budget.

Customer loan amount prediction

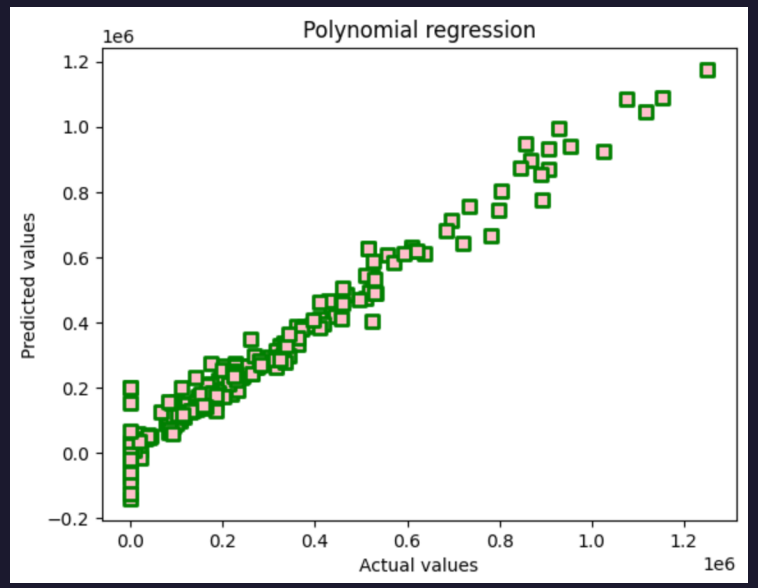
Subtitle

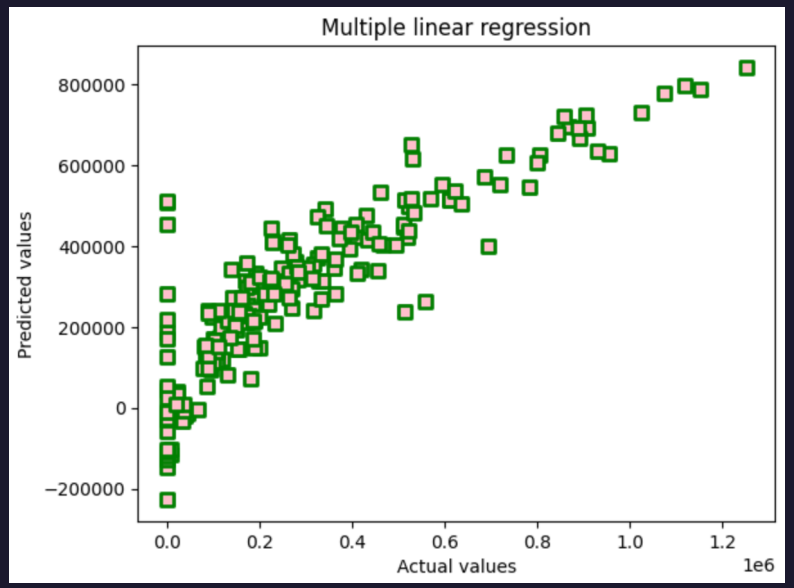
Prediction models - comparison

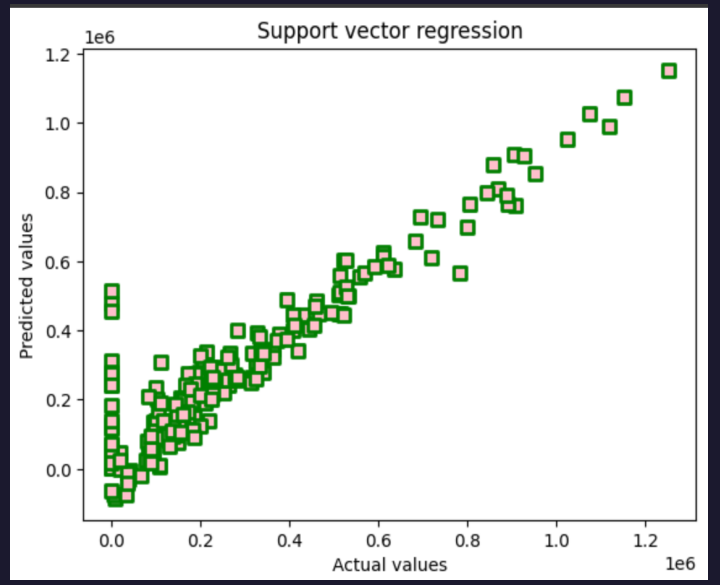
- **Features:** `Gender, Age, Married, No_Of_Dependents, Edu_Qualification, Employment_Status, Applicant_Annual_Income, Co_Applicant_Annual_Income, Applicant_Credit_History, Credit_Rating, Existing_No_of_Loans`
- **Label:** `Loan_Amount_Availed`
- We have normalized and encoded the categorical data.

- Polynomial regression model
  - Based on the features and labelled we tarin, polynomial regression model with degree of 2.
- Multiple regression model
  - Based on the features and labelled we tarin, multiple regression model.
- Support vector regression model
  - Based on the features and labelled we tarin, support vector regression model.

Observations







Observations

Type of Model	R2 score
Polynomial regression model	0.97030133160360823
Multiple regression model	0.7549258446460376
Support vector regression model	0.872788526395641

Used car price prediction

Subtitle

Data

1. S.No. : Serial Number
2. Name : Name of the car which includes Brand name and Model name
3. Location : The location in which the car is being sold or is available for purchase Citiesr> 4. Year : Manufacturing year of the car
4. Kilometers_driven : The total kilometers driven in the car by the previous owner(s) in KM.
5. Fuel_Type : The type of fuel used by the car. (Petrol, Diesel, Electric, CNG, LPG)
6. Transmission : The type of transmission used by the car. (Automatic / Manual)
7. Owner : Type of ownership
8. Mileage : The standard mileage offered by the car company in kmpl or km/kg
9. Engine : The displacement volume of the engine in CC.
10. Power : The maximum power of the engine in bhp.
11. Seats : The number of seats in the car.
12. New_Price : The price of a new car of the same model in INR Lakhs.(1 Lakh = 100, 000)
13. Price : The price of the used car in INR Lakhs (1 Lakh = 100, 000)

Data Cleaning and Feature Extraction

Features

#	Column	Non-Null Count	Dtype
0	S.No.	7253 non-null	int64
1	Name	7253 non-null	object
2	Location	7253 non-null	object
3	Year	7253 non-null	int64
4	Kilometers_Driven	7253 non-null	int64
5	Fuel_Type	7253 non-null	object
6	Transmission	7253 non-null	object
7	Owner_Type	7253 non-null	object
8	Mileage	7251 non-null	object
9	Engine	7207 non-null	object
10	Power	7207 non-null	object
11	Seats	7200 non-null	float64
12	New_Price	1006 non-null	object
13	Price	6019 non-null	float64

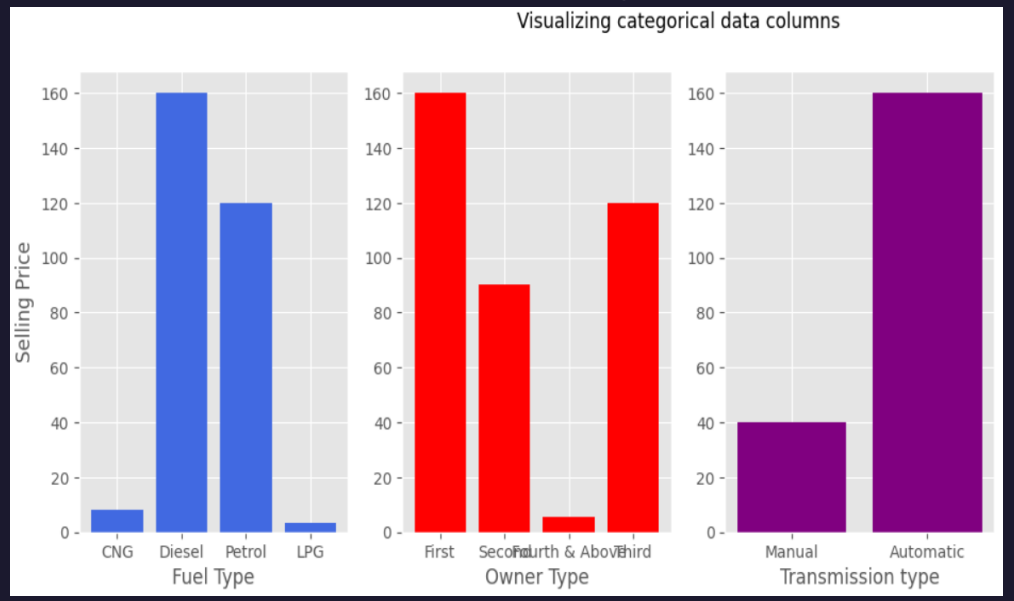
Null values

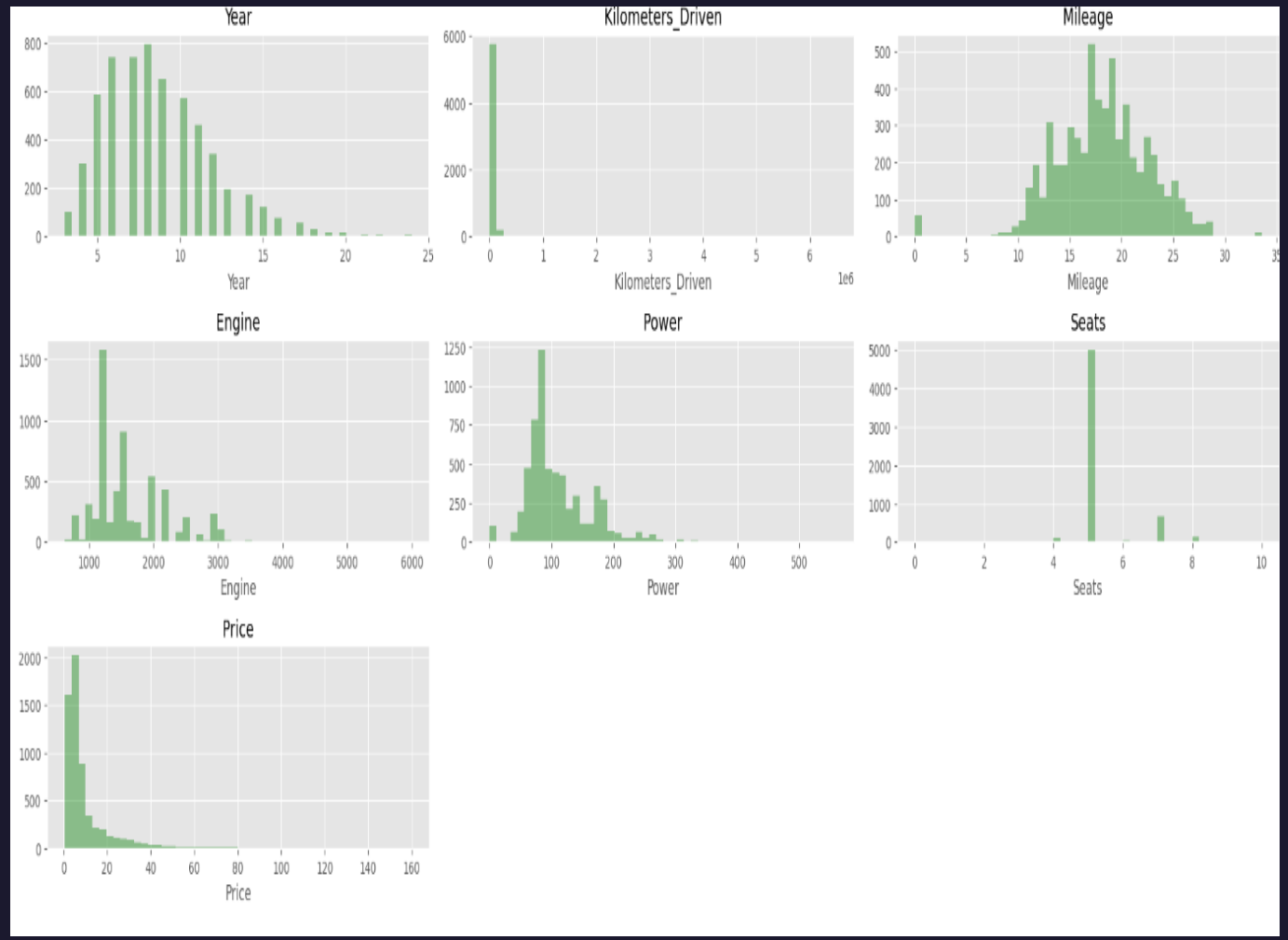
S.No.	0
Name	0
Location	0
Year	0
Kilometers_Driven	0
Fuel_Type	0
Transmission	0
Owner_Type	0
Mileage	2
Engine	46
Power	46
Seats	53
New_Price	6247
Price	1234

Extracting the Age feature from Year of manufacturing

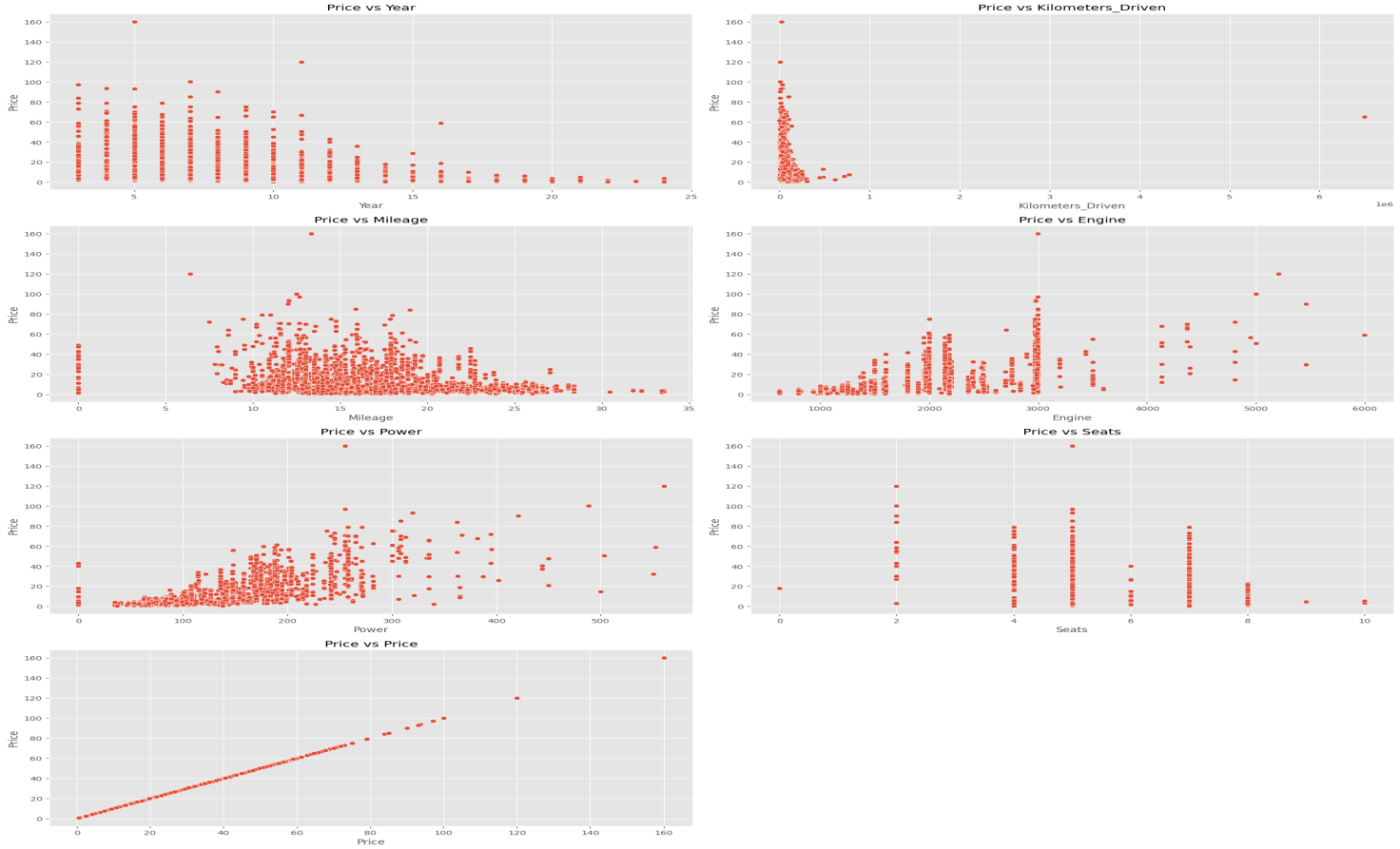
```
df['Current_year']=2022
df['Age']=df['Current_year']-df['Year']
df.drop('Current_year',axis=1,inplace=True)
df.head()
```

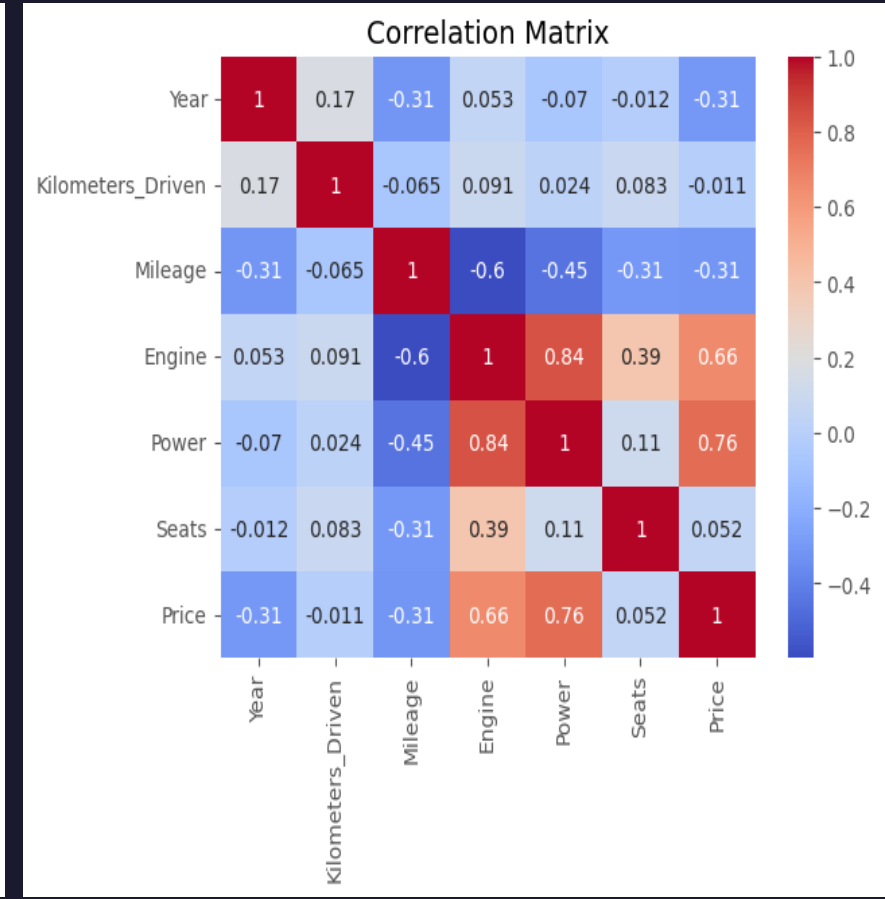
Exploratory Data Analysis





EDA Continued...





Data preparation

```
num_features = df.drop(['Price'], axis=1).select_dtypes(include=np.number).columns
cat_features = df.drop('Price', axis=1).select_dtypes(include=['object']).columns
```

```
numeric_transformer = Pipeline(steps=[
    ('scaler', StandardScaler()) ])

categorical_transformer = Pipeline(steps=[
    ('one_hot', OneHotEncoder(handle_unknown='ignore'))])

preprocessor = ColumnTransformer(
    transformers=[
        ('num', numeric_transformer, num_features),
        ('cat', categorical_transformer, cat_features)
    ])
```

Model Comparison and Prediction

Linear Regression Model

```
Training set score: 0.983518657107881
Test set score: 0.8534424085724738
Root Mean Square Error Linear Regression: 1.563933154754172
Root Mean Square Error Linear Regression: 5.080139621018562
```

Random Forest

```
Training set score: 0.9911319271435441
Test set score: 0.9434521320553896
Root Mean Square Error RF: 1.6989389771122052
Root Mean Square Error RF: 2.976238029547701
```

Gradient Boost Model

```
Training set score: 0.9708334735775364
Test set score: 0.9459681302503917
Root Mean Square Error Gradient Boost: 2.3696053797348866
Root Mean Square Error Gradient Boost: 2.852502154485467
```

XGBoost Model

```
Training set score: 0.9858413053501343
Test set score: 0.9545708921959206
Root Mean Square XGB: 1.5472298691594575
Root Mean Square XGB: 2.650681414051423
```

```
predicted_price = pipe_xg_test.predict(pd.DataFrame(columns=X_test.columns,data=np.array(['Hyundai Elite i20 Magna Plus','Kochi',11,23955,
                                                                                          'Petrol','Manual','First',18.6,1197,81.86,5]).reshape(1,11)))
```

```
The predicted car value for the test data taken: [3] lakhs
```

Customer car recommendation

Clustering

Data Cleaning and EDA

- Removal of –ve Values
- Removal of rows with NULL values
- Removal of rows with Car Purchased column as No Vehicle

Clustering models - comparison

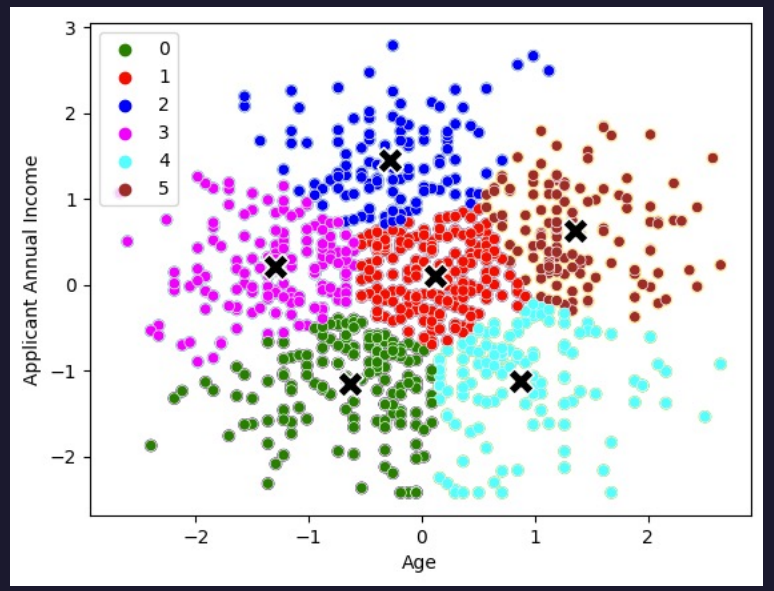
- K Means Clustering
  - Used ELBOW Method to identify the number of clusters
  - Used Customers *Age*, *Applicant Annual Income*, *Credit Score* as features to create the cluster and plot them in 2D and 3D plots
  - With new customer data, we get the cluster to which customer belongs using kmeans.predict
  - Once the cluster is known, the TOP THREE CARS in the given cluster is recommended to the Customer as part of RECOMMENDER SYSTEM

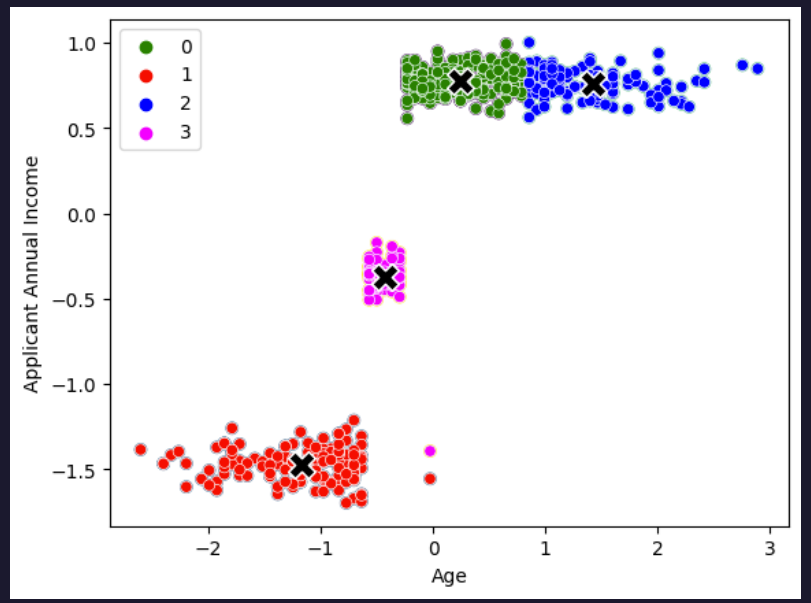
Hierarchical Clustering

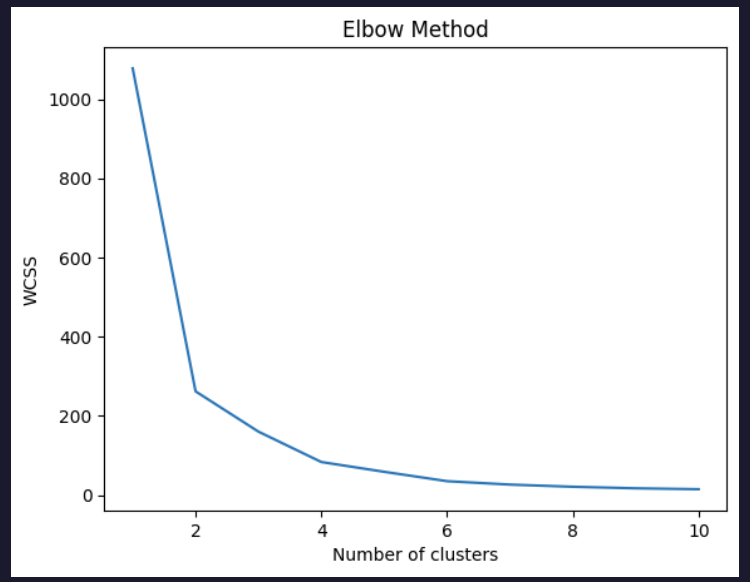
- Tried Hierarchical clustering to identify the number of clusters using the same set of features
- Implemented the basic clustering for comparison

Clustering models - KMeans

- K Means Clustering 2D Sample Plots

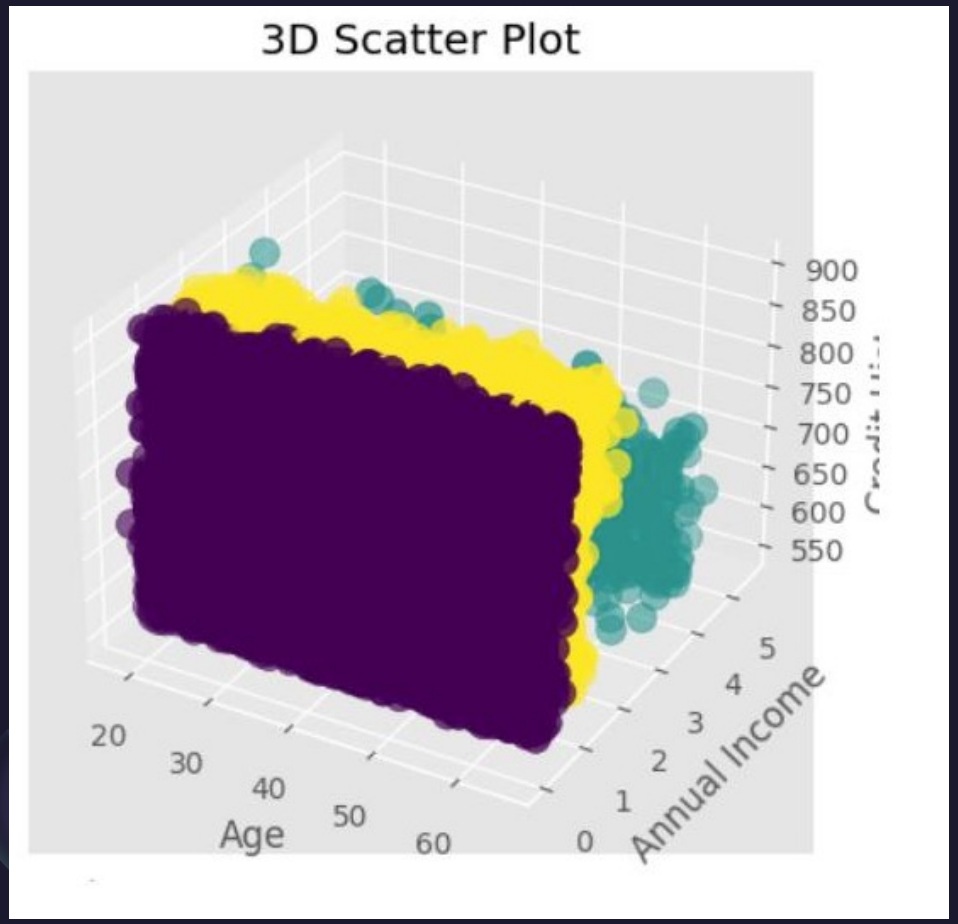


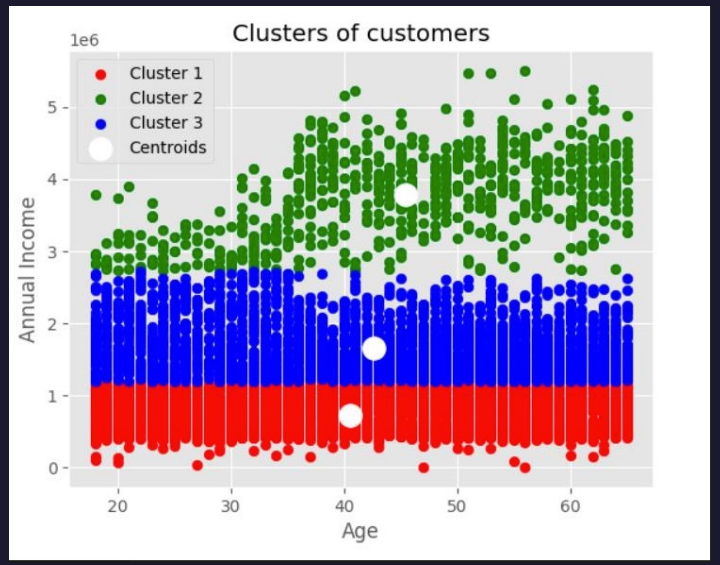


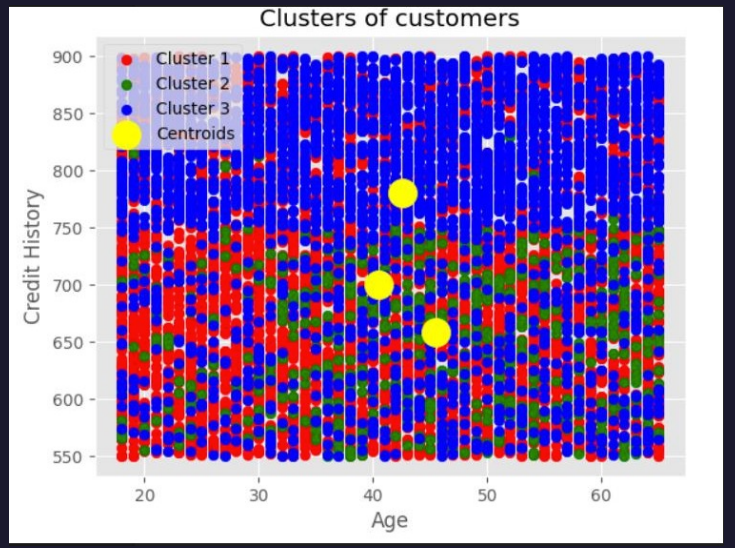


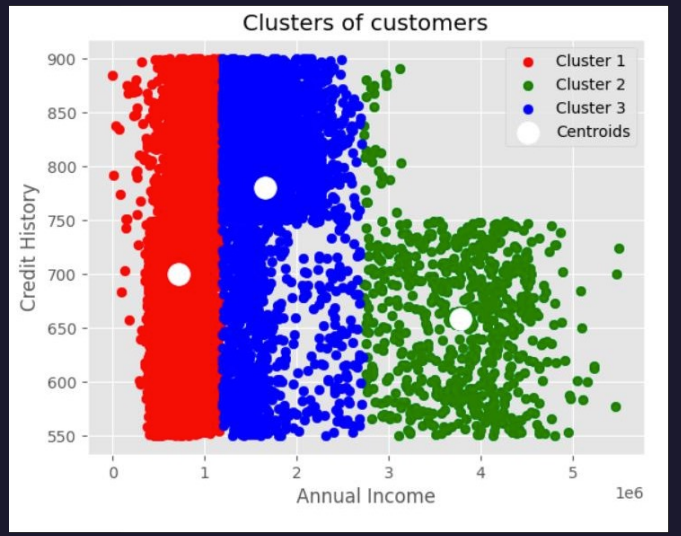
Clustering models - KMeans

- K Means Clustering 3D Sample Plots



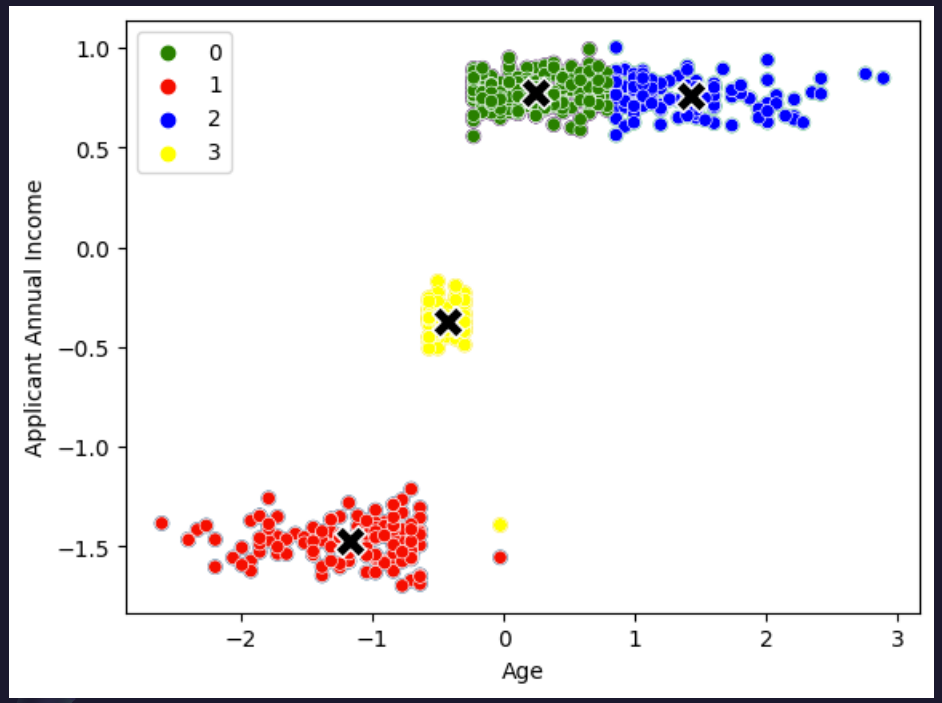


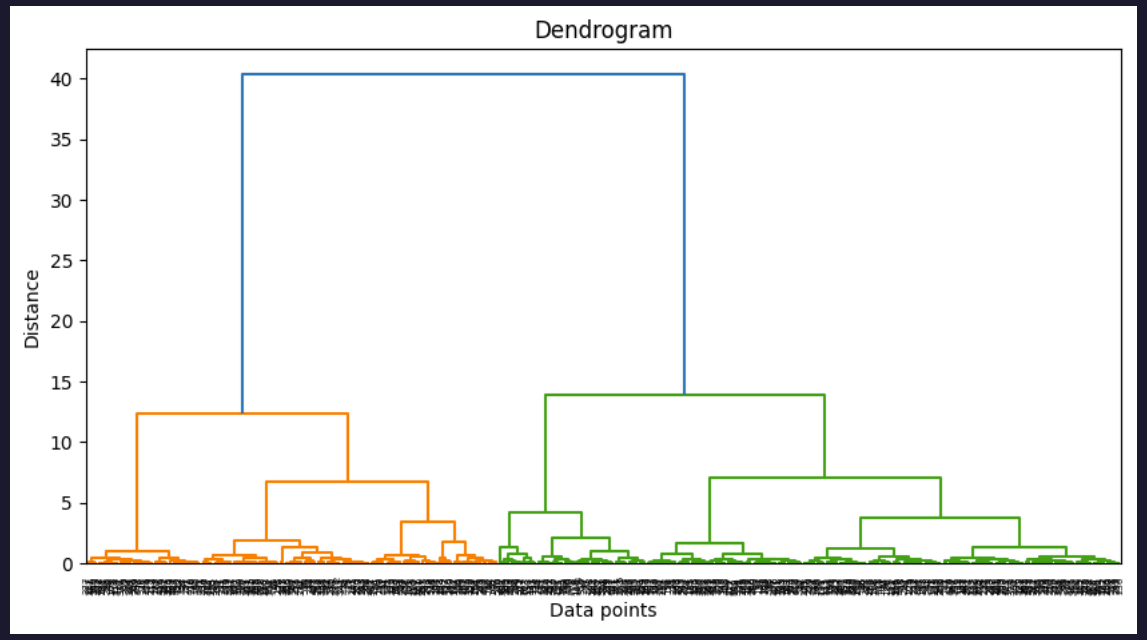




Clustering models - Hierarchical

- Hierarchical Clustering Sample Plots





Kmeans Clustering models - Recommendation

- *After we were able identify the cluster, we labelled each data in the user dataframe to the cluster it belongs to.*
- Output: *Identify the cluster for the new user*
- *Recommend the top 3 cars in the cluster*

```
   Age  Applicant_Annual_Income
0   35                  4550000
The new data point belongs to cluster: 3
```

Car Recommendation

```
Recommended Top 3 Cars for cluster 3 is ['Toyota Etios', 'Maruti Suzuki Desire', 'Maruti Suzuki Swift']
```

Summary

References and Data sources

Thank You