Predicting Used Car Prices in San Francisco

Design

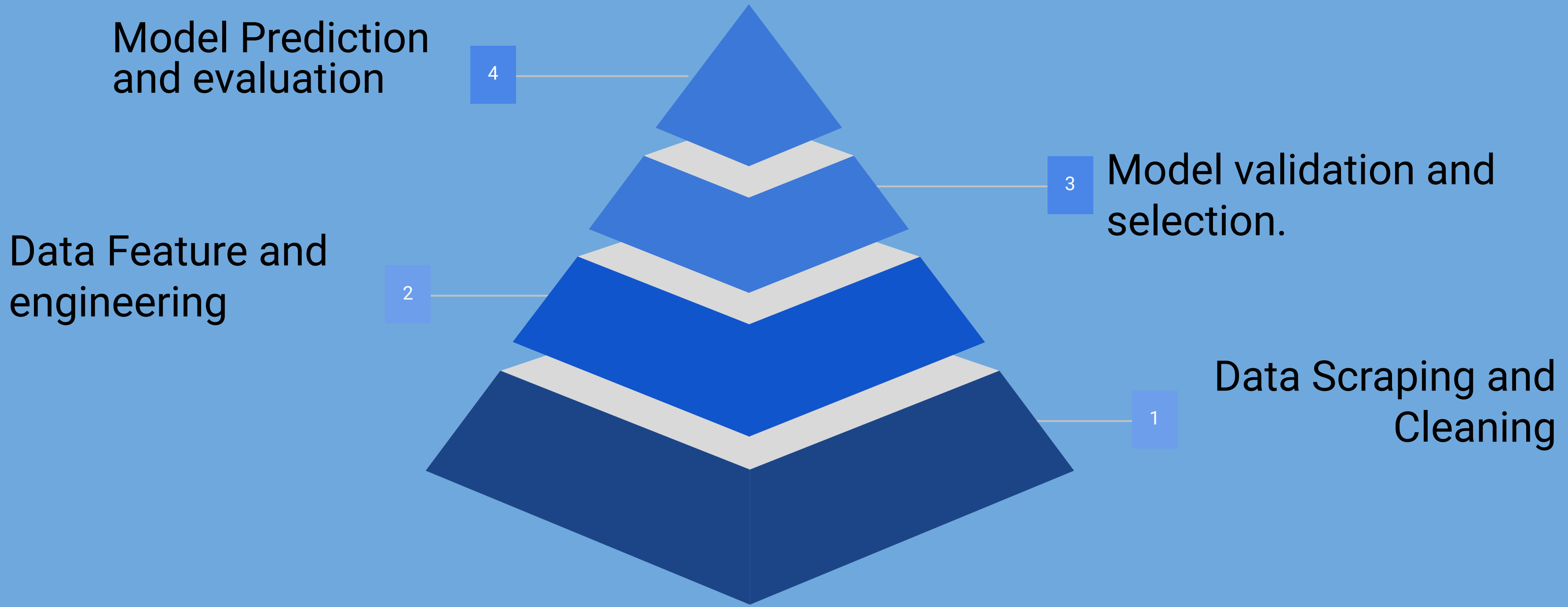
Client:

- Buyers of used cars
- sellers of used cars

Objective: Explore whether the sale price of a Car can be modeled against other car features.

Goal: Produce a regression model that can best interpret a relationship for sale price with other features of car and best predict Car sale price in San Francisco region.

PROJECT STRATEGY



Scraping

1. Scraped ~13000 car listing from cars.com
2. Patience is the key!
3. Expressvpn to the rescue.

Data Cleaning and EDA

1. Dealing with missing MPG
2. Extracted “make/brand” from the vehicle name and encoded using one hot encoding
3. Extracted engine_volume from engine:
   a. 3.6L V6 24V MPFI DOHC
   b. Intercooled Turbo Regular Unleaded I-4 1.5 L/91
   c. Regular Unleaded V-6 3.6 TFSI/220
4. Extracted and encoded transmission
   a. 50+ transmission types (1 speed, 2 speed, CVT etc.)
   b. Reduced them to 4 labels
5. Cleaned up fuel_type
   a. 6+ Fuel Types
   b. Converted them to 3 categories: gasoline, Diesel and hybrid.

Iterations - EDA

Take 1: R-Square of .55

1. 6000 car entries from 2016 -2021
2. Only SUV’s

Take 2: R-Square of .62

1. Added sedan’s car entries from 2016 -2021
2. Total ~9000 cars.

Take 3: R-Square of .71

1. Added sedan’s and suv’s from 2009
2. Total ~13000 cars.

Feature Engineering

1. Dropped cars for less popular brands:

```
car_brand_mask= (
    (df7["brand"] == "Chrysler") |
    (df7["brand"] == "FIAT") |
    (df7["brand"] == "Genesis") |
    (df7["brand"] == "Jaguar") |
    (df7["brand"] == "Land") |
    (df7["brand"] == "MINI") |
    (df7["brand"] == "Maserati") |
    (df7["brand"] == "Mitsubishi") |
    (df7["brand"] == "Porsche") |
    (df7["brand"] == "RAM") |
    (df7["brand"] == "Rolls-Royce") |
    (df7["brand"] == "Scion") |
    (df7["brand"] == "Bentley") |
    (df7["brand"] == "Hummer") |
    (df7["brand"] == "Pontiac") |
    (df7["brand"] == "Saturn")

)
```

2. Removed outliers using interquartile range:

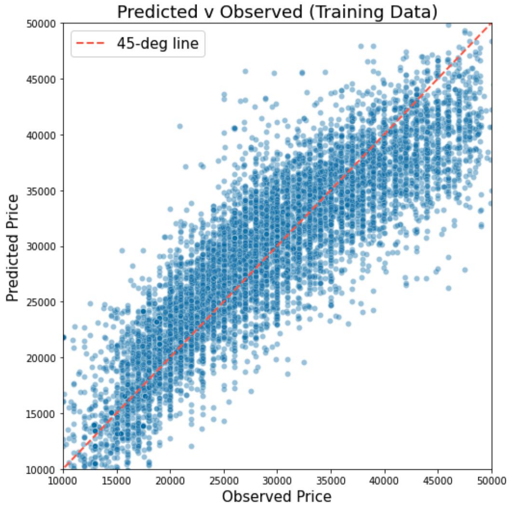
```
q3 = df8.quantile(0.75)
q1 = df8.quantile(0.25)
iqr = q3-q1
maxm, minm = q3 + 1.5*iqr, q1 - 1.5*iqr
```

3. R-square increased to : .77

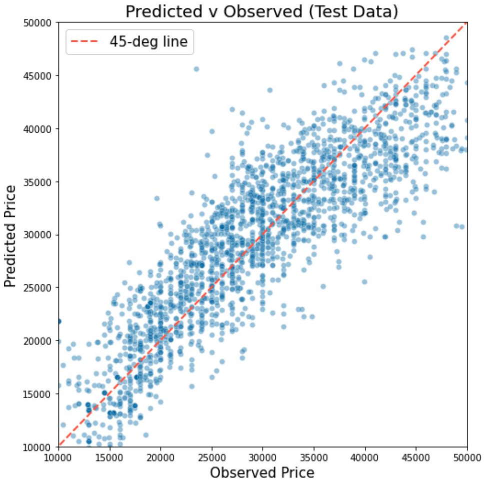
Baseline Model

Correlation Matrix for Car features

Linear Regression On Train Data

1. Using OLS: Adj r-square was .77
2. Using Sklearn: Adj r-square was .77
3. Evaluated the training data:
   a. Mean residual was near to zero: 1.0561957136808735e-11
   b. 

Linear Regression on test data

1. Using Sklearn: Adj r-square was .76.
2. Evaluated the training data:
   a. Mean residual jumped to around 151.
   b. 

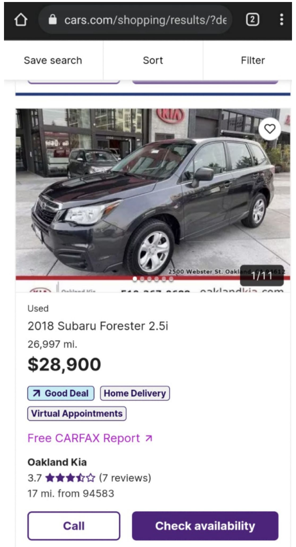
Model in action

1. Testing it on a car bought in 2018 of make subaru, with 30000 miles and 2.5 L engine

   model.predict([[2018,30000,0.000,0,1,2.5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0]])

   The model predicted the price as: array([27488.04852201])

2. Cars.com listing for similar model:



0	year
1	miles
2	transmission
3	fuel_type
4	drivetrain
5	engine_volume
6	Acura
7	Alfa
8	Audi
9	BMW
10	Buick
11	Cadillac
12	Chevrolet
13	Dodge
14	Ford
15	GMC
16	Honda
17	Hyundai
18	INFINITI
19	Jeep
20	Kia
21	Lexus
22	Mazda
23	Mercedes-Benz
24	Nissan
25	Subaru
26	Toyota
27	Volkswagen
28	Volvo

Learning and next steps

1. Add more features
2. Explore polynomial regression for some of the features