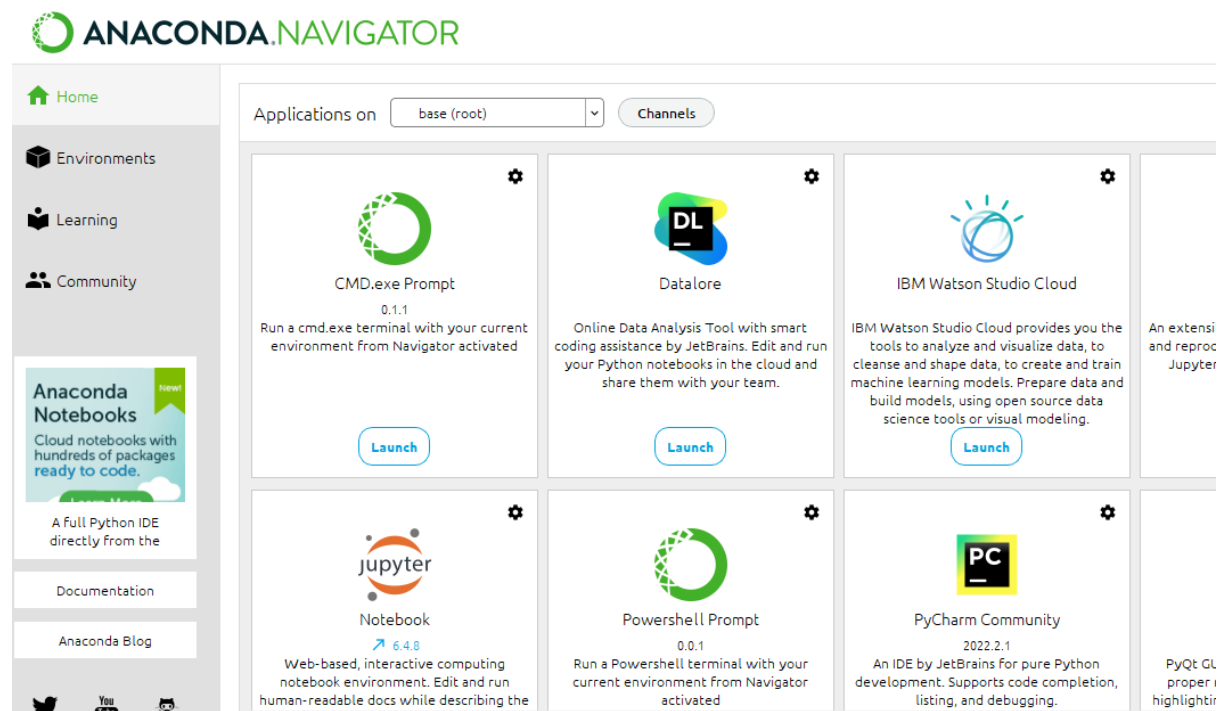


Date	26 November 2022
Team ID	PNT2022TMID30206
Project Name	Car Resale Value Prediction

Pre – Requisites:



Collect the dataset:

The dataset is collected from kaggle.com website

Pre – Process the dataset:

Importing required libraries and read the dataset

```
##import libraries
import pandas as pd
import numpy as np
import matplotlib as plt
from sklearn.preprocessing import LabelEncoder
import pickle
import seaborn as sns

car=pd.read_csv(r"C:\Users\narma\OneDrive\Desktop\Project\autos.csv",header=0,sep=',',encoding='Latin1',)
```

Cleaning the dataset

```
##remove unwanted columns  
car.drop(['Unnamed: 0', 'car_name', 'min_cost_price', 'max_cost_price', 'seller_type'], axis='columns', inplace=True)  
car.sample()
```

```
##saving cleaned dataset  
car2.to_csv("autos_preprocessed.csv")
```

```
car2['fuel_type'].unique()
```

```
array(['Petrol', 'Diesel', 'CNG', 'LPG'], dtype=object)
```

Splitting the dataset

```
##split the data set  
X=labeled.iloc[:,1:].values  
Y=labeled.iloc[:,0].values  
Y=Y.reshape(-1,1)
```

```
np.isnan(X)
```

```
array([[False, False, False, ..., False, False, False],  
       [False, False, False, ..., False, False, False],  
       [False, False, False, ..., False, False, False],  
       ...,  
       [False, False, False, ..., False, False, False],  
       [False, False, False, ..., False, False, False],  
       [False, False, False, ..., False, False, False]])
```

```
##train the dataset  
from sklearn.model_selection import cross_val_score, train_test_split
```

Model building

Choosing the appropriate model

```
##model building  
##random forest  
from sklearn.ensemble import RandomForestRegressor  
regressor=RandomForestRegressor()  
regressor.fit(X_train, np.ravel(Y_train, order='C'))
```

```
RandomForestRegressor()
```

Check the metrics of the model

```
pred_test=regressor.predict(X_test)
pred_test
```

```
array([[205820.          , 472791.66666667, 523470.          , ...,
        697800.          , 239897.5          , 514500.          ]])
```

```
Y_test
```

```
array([[280000],
       [330000],
       [591000],
       ...,
       [650000],
       [200000],
       [600000]], dtype=int64)
```

Saving the model

```
#saving the model
import pickle
filename='random.pkl'
pickle.dump(regressor,open(filename,"wb"))
```

Application building

```
from flask import Flask, render_template, request, redirect, url_for, session, redirect
import requests

GOOGLE_CLIENT_ID = "564634383443-f47nsem7k4kl0julaj8jlbn1fkcf3t71.apps.googleusercontent.com"
GOOGLE_CLIENT_SECRET = "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"
REDIRECT_URI = '/google/auth'

app=Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

@app.route("/main")
def main():
    return render_template("main.html")

@app.route("/google")
def google():
    return redirect(f"https://accounts.google.com/o/oauth2/v2/auth?scope=https://www.googleapis.com/auth/

@app.route("/google/auth")
def google_auth():
    r = requests.post("https://oauth2.googleapis.com/token",
    data={
```