Name: Laâroussi Saâdeddine

Batch code: LISUM09

Submission date: 26/05/2022

Submitted to: Data Glacier

Mail: laar.saad.eddine@gmail.com

Country: Morocco

Steps taken for week 4: Deployment on Flask

1) Data intake report :

Chosen data is from kaggle: https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction

Data is about **flight price prediction**

- **Tabular data details: (Clean_DataSet)**

Total number of observations	300153
Total number of files	1
Total number of features	12
Base format of the file	csv
Size of the data	27.5+ MB

- **Features:**
  - **Unnamed:0** : Index of the original data, column was removed.
  - **Airline**: contains 6 different airlines:
    - 'SpiceJet', 'AirAsia', 'Vistara', 'GO_FIRST', 'Indigo', 'Air_India'
  - **Flight**: contains name of flight, was removed since not necessary for model creation.
  - **Source city** and **Destination city**: contains the cities:
    - 'Delhi', 'Mumbai', 'Bangalore', 'Kolkata', 'Hyderabad', 'Chennai'.
  - **Departure time** and **Arrival time**:
    - 'Evening', 'Early_Morning', 'Morning', 'Afternoon' 'Night', 'Late_Night'
  - **Stops**: number of times the flight will stop:
    - 'zero', 'one', 'two_or_more'
  - **Class**:
    - 'Economy', 'Business'
  - **Duration**: duration of flight
  - **Days left**: days left until the flight.
  - **Price**: price of the flight.

2) Model creation

The model will predict the price of a flight.

X values are: 'airline', 'source_city', 'departure_time', 'stops', 'arrival_time', 'destination_city', 'class', 'duration', 'days_left'.

Y value is 'price'.

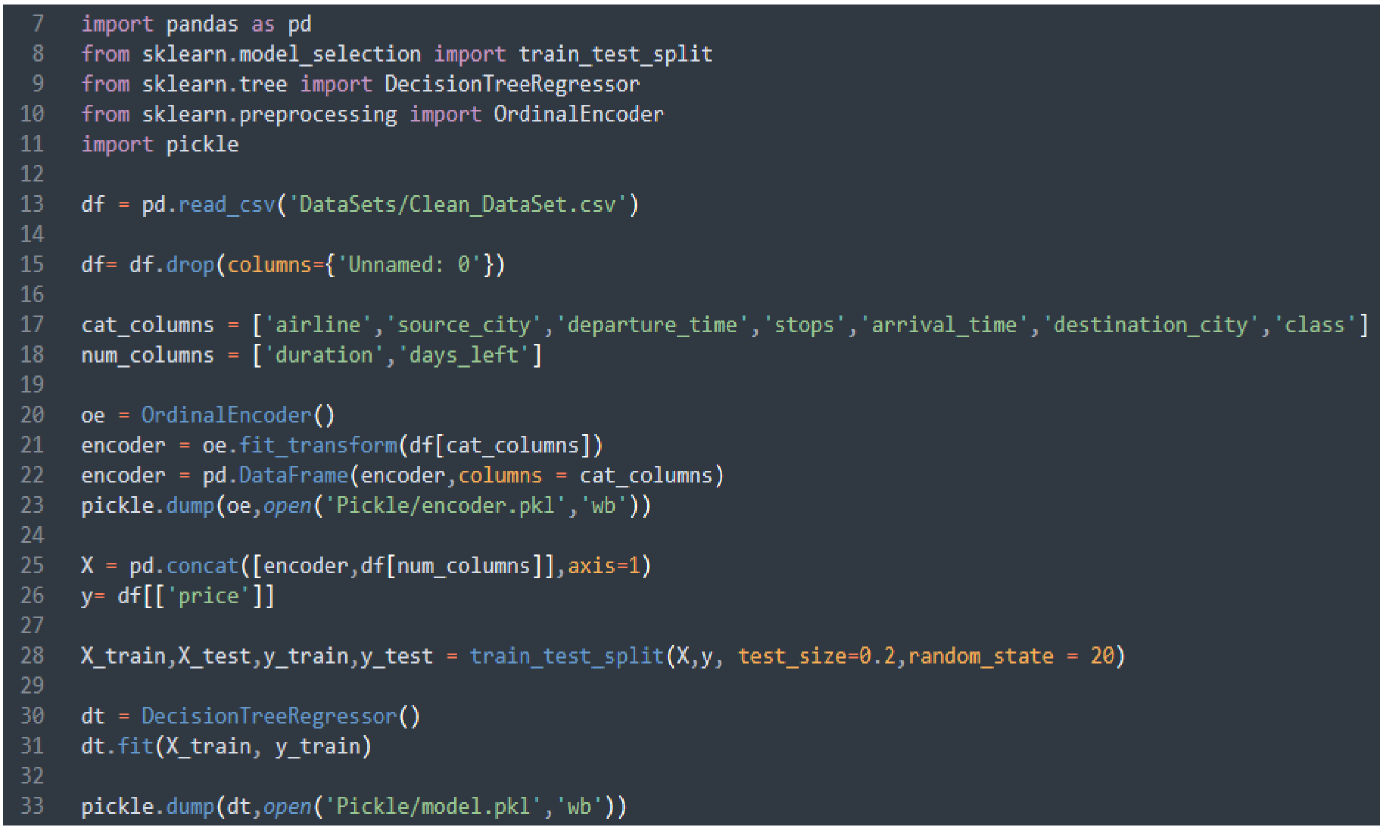
Model was tested with **linear regression** and **decision tree classifier**. Linear regression had an R2 score of **0.90** and decision tree classifier had an R2 score of **0.98**.

With a higher R2 score value the **decision tree classifier** was chosen.

Using pickle the model was saved.

The ordinal encoder that was used on the string X values was also saved using pickle.

Model code:



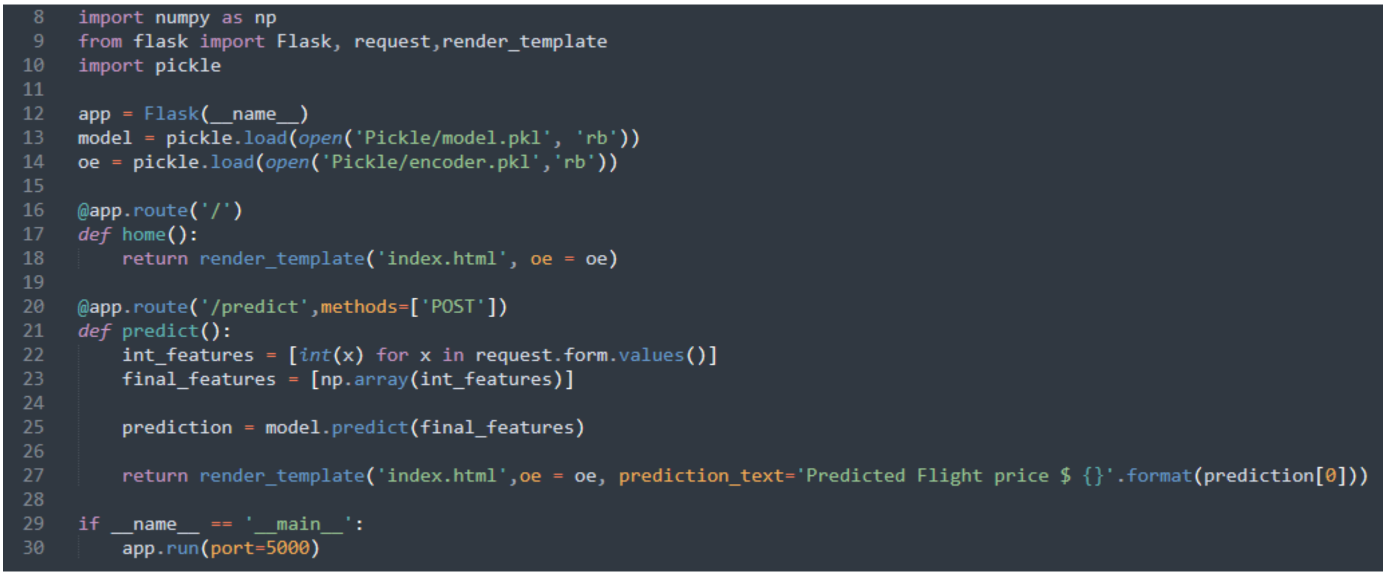
```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor
from sklearn.preprocessing import OrdinalEncoder
import pickle

df = pd.read_csv('DataSets/Clean_DataSet.csv')

df= df.drop(columns={'Unnamed: 0'})

cat_columns = ['airline','source_city','departure_time','stops','arrival_time','destination_city','class']
num_columns = ['duration','days_left']

oe = OrdinalEncoder()
encoder = oe.fit_transform(df[cat_columns])
encoder = pd.DataFrame(encoder,columns = cat_columns)
pickle.dump(oe,open('Pickle/encoder.pkl','wb'))

X = pd.concat([encoder,df[num_columns]],axis=1)
y= df[['price']]

X_train,X_test,y_train,y_test = train_test_split(X,y, test_size=0.2,random_state = 20)

dt = DecisionTreeRegressor()
dt.fit(X_train, y_train)

pickle.dump(dt,open('Pickle/model.pkl','wb'))
```

3) Deployment on Flask

Flask was installed using pip.

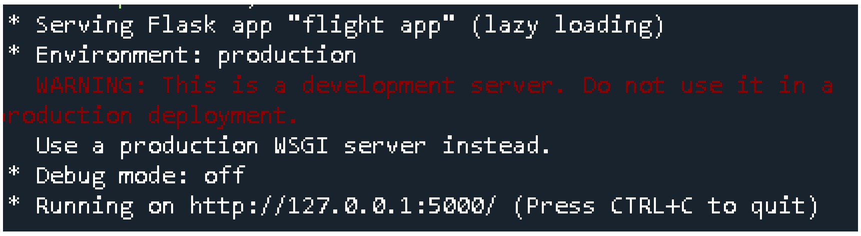
An HTML page and css file were created.

Flask app code :



```python
import numpy as np
from flask import Flask, request,render_template
import pickle

app = Flask(__name__)
model = pickle.load(open('Pickle/model.pkl', 'rb'))
oe = pickle.load(open('Pickle/encoder.pkl','rb'))

@app.route('/')
def home():
    return render_template('index.html', oe = oe)

@app.route('/predict',methods=['POST'])
def predict():
    int_features = [int(x) for x in request.form.values()]
    final_features = [np.array(int_features)]

    prediction = model.predict(final_features)

    return render_template('index.html',oe = oe, prediction_text='Predicted Flight price $ {}'.format(prediction[0]))

if __name__ == '__main__':
    app.run(port=5000)
```

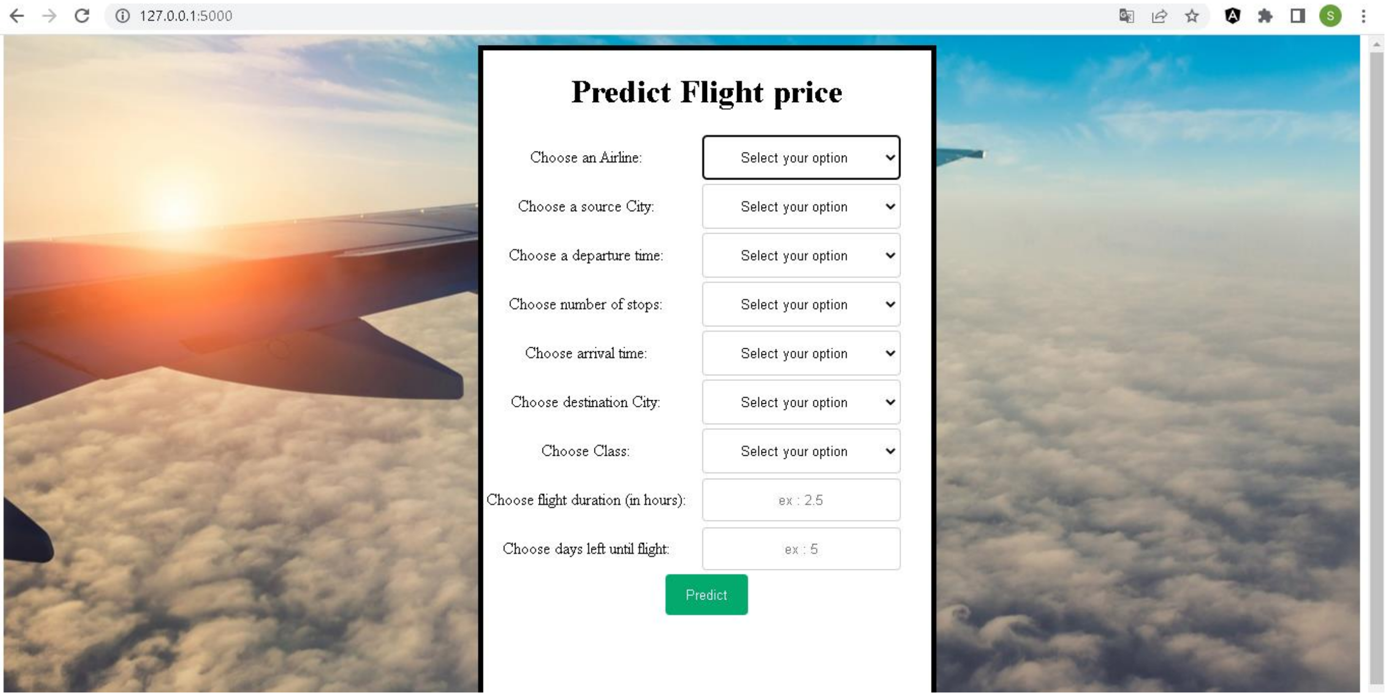
Flask was deployed:



```
* Serving Flask app "flight app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a
roduction deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

HTML page:

The options on the select HTML tags were obtained from the saved ordinal encoder



127.0.0.1:5000

Predict Flight price

Choose an Airline: Select your option

Choose a source City: Select your option

Choose a departure time: Select your option

Choose number of stops: Select your option

Choose arrival time: Select your option

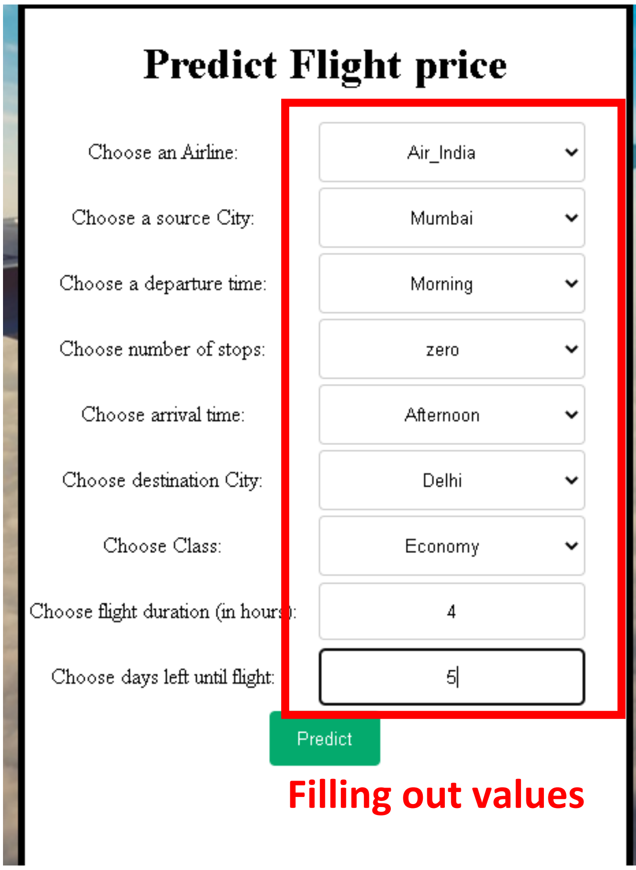
Choose destination City: Select your option

Choose Class: Select your option

Choose flight duration (in hours): ex : 2.5

Choose days left until flight: ex : 5

Getting a prediction:



Predict Flight price

Choose an Airline: Air_India

Choose a source City: Mumbai

Choose a departure time: Morning

Choose number of stops: zero

Choose arrival time: Afternoon

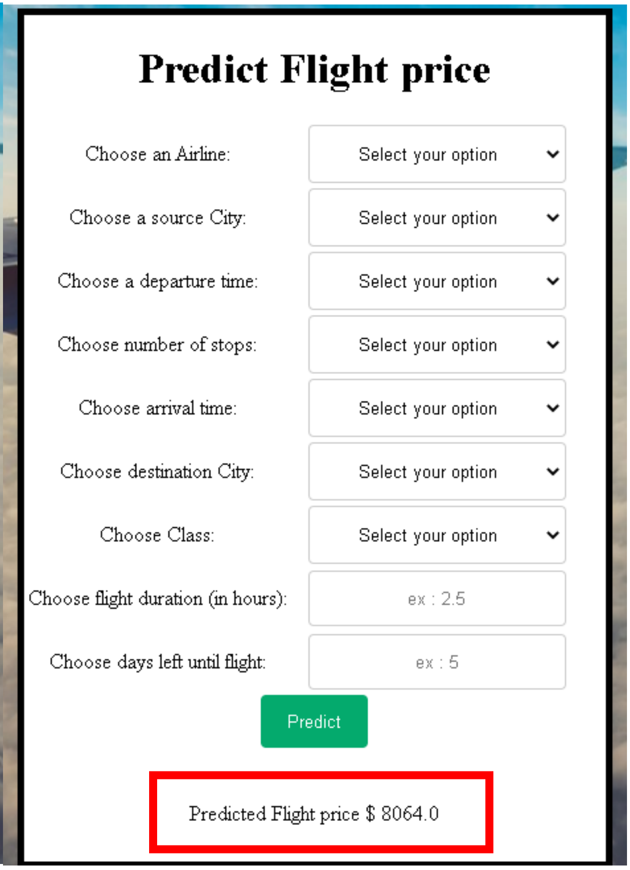
Choose destination City: Delhi

Choose Class: Economy

Choose flight duration (in hours): 4

Choose days left until flight: 5

Filling out values



Predict Flight price

Choose an Airline: Select your option

Choose a source City: Select your option

Choose a departure time: Select your option

Choose number of stops: Select your option

Choose arrival time: Select your option

Choose destination City: Select your option

Choose Class: Select your option

Choose flight duration (in hours): ex : 2.5

Choose days left until flight: ex : 5

Predicted Flight price $ 8064.0

Getting prediction