Week4: Deployment on Flask

Name: Walquer Xavier Valles Ruiz

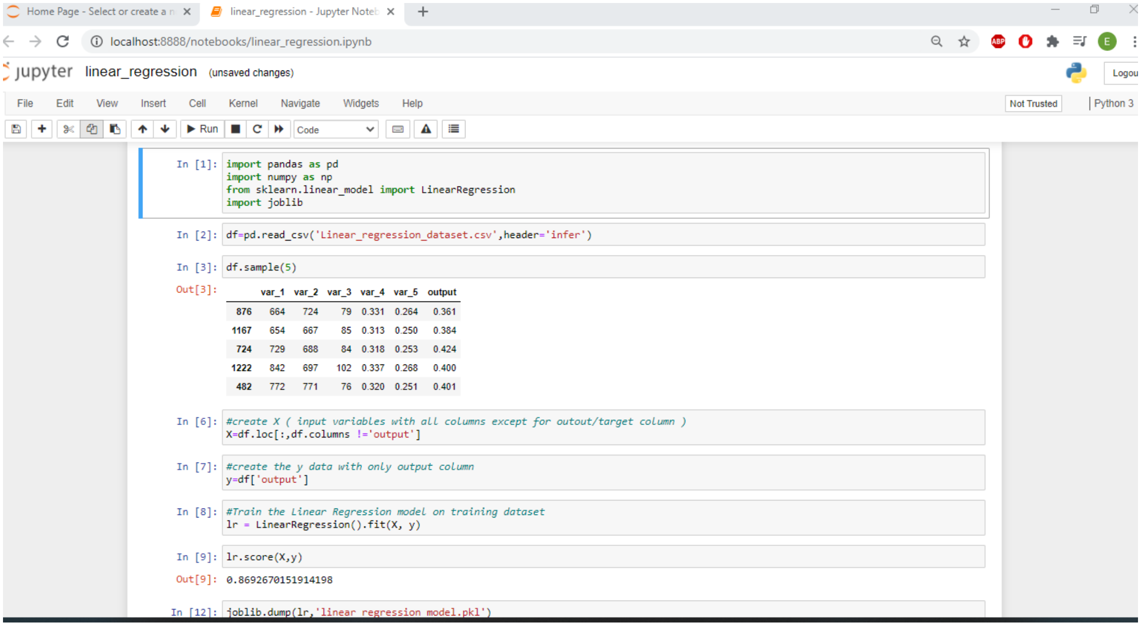
Batch code: LISP01

Submission date: 21/03/2021

Submitted to: GitHub

Snapshot of steps of deployment

1. Select any toy data (simple data).



Home Page - Select or create a n
linear_regression - Jupyter Noteb
localhost:8888/notebooks/linear_regression.ipynb
jupyter linear_regression (unsaved changes)
Logou
File Edit View Insert Cell Kernel Navigate Widgets Help
Not Trusted
Python 3
Run
Code

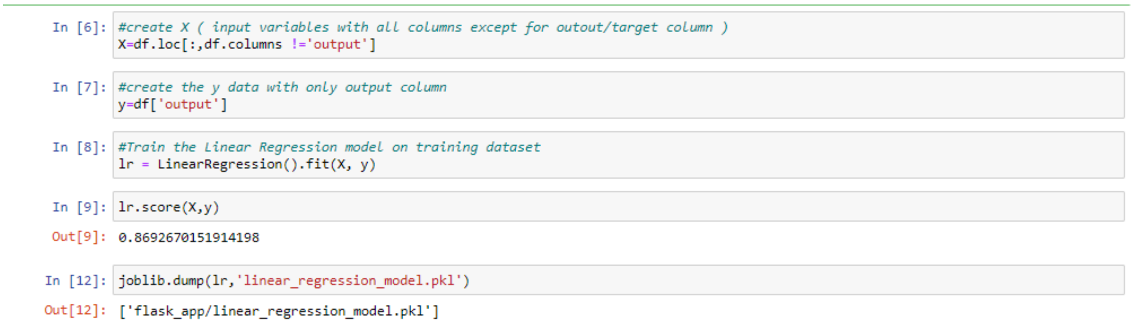
```
In [1]: import pandas as pd
        import numpy as np
        from sklearn.linear_model import LinearRegression
        import joblib

In [2]: df=pd.read_csv('Linear_regression_dataset.csv',header='infer')

In [3]: df.sample(5)
```

Out[3]:

	var_1	var_2	var_3	var_4	var_5	output
876	664	724	79	0.331	0.264	0.361
1167	654	667	85	0.313	0.250	0.384
724	729	688	84	0.318	0.253	0.424
1222	842	697	102	0.337	0.268	0.400
482	772	771	76	0.320	0.251	0.401

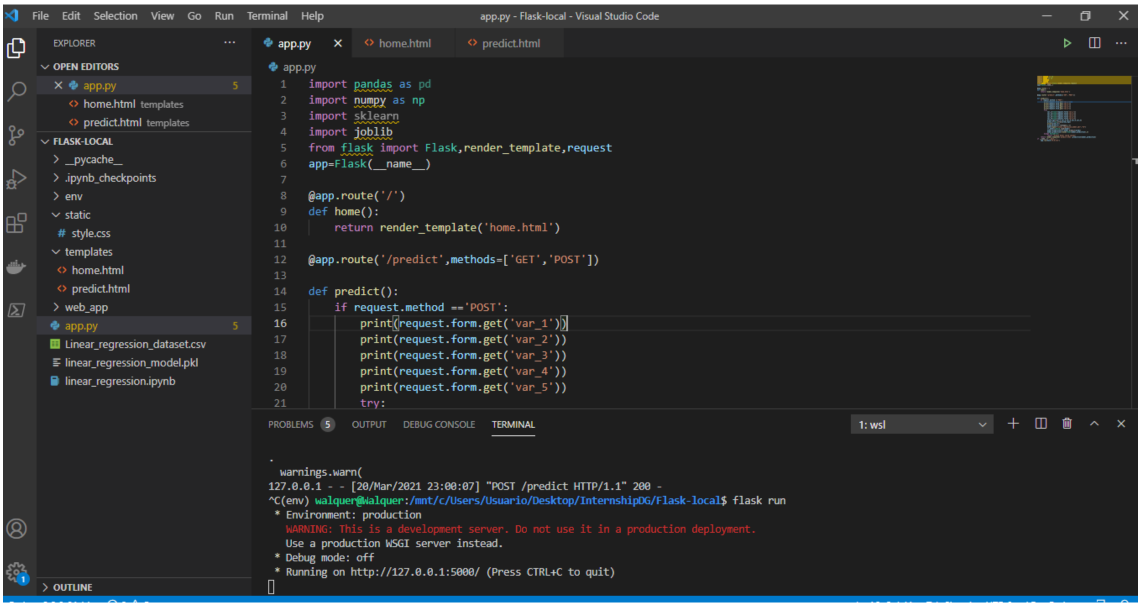
```
In [6]: #create X ( input variables with all columns except for outout/target column )
        X=df.loc[:,df.columns !='output']

In [7]: #create the y data with only output column
        y=df['output']

In [8]: #Train the Linear Regression model on training dataset
        lr = LinearRegression().fit(X, y)

In [9]: lr.score(X,y)
Out[9]: 0.8692670151914198

In [12]: joblib.dump(lr,'linear_regression_model.pkl')
```

2. Save the model



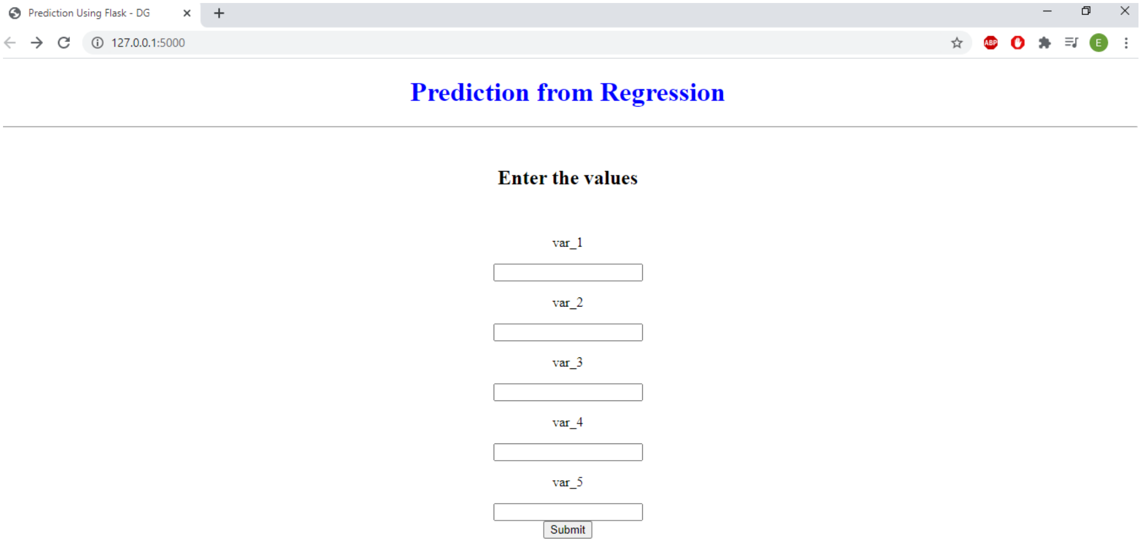
```
In [6]: #create X ( input variables with all columns except for outout/target column )
        X=df.loc[:,df.columns !='output']

In [7]: #create the y data with only output column
        y=df['output']

In [8]: #Train the Linear Regression model on training dataset
        lr = LinearRegression().fit(X, y)

In [9]: lr.score(X,y)
Out[9]: 0.8692670151914198

In [12]: joblib.dump(lr,'linear_regression_model.pkl')
Out[12]: ['flask_app/linear_regression_model.pkl']
```

3. Deploy the model on flask (web app)



```python
import pandas as pd
import numpy as np
import sklearn
import joblib
from flask import Flask,render_template,request
app=Flask(__name__)

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

@app.route('/predict',methods=['GET','POST'])

def predict():
    if request.method =='POST':
        print(request.form.get('var_1'))
        print(request.form.get('var_2'))
        print(request.form.get('var_3'))
        print(request.form.get('var_4'))
        print(request.form.get('var_5'))
        try:
```

```
.
  warnings.warn(
127.0.0.1 - - [20/Mar/2021 23:00:07] "POST /predict HTTP/1.1" 200 -
^C(env) walquer@Walquer:/mnt/c/Users/Usuario/Desktop/InternshipDG/Flask-local$ flask run
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```



Prediction from Regression

Enter the values

var_1

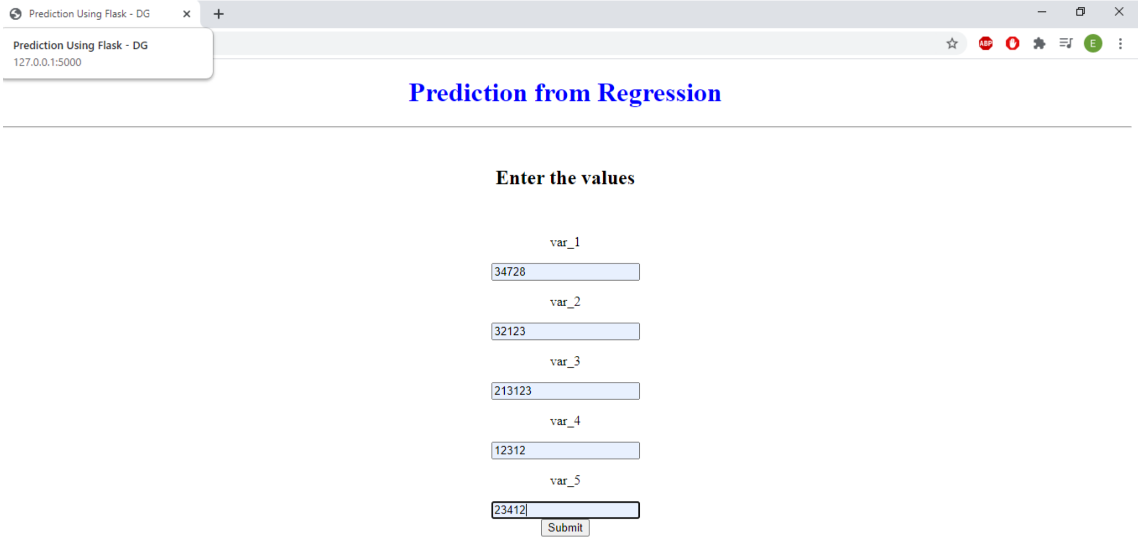
var_2

var_3

var_4

var_5

Submit



Prediction from Regression

Enter the values

var_1

34728

var_2

32123

var_3

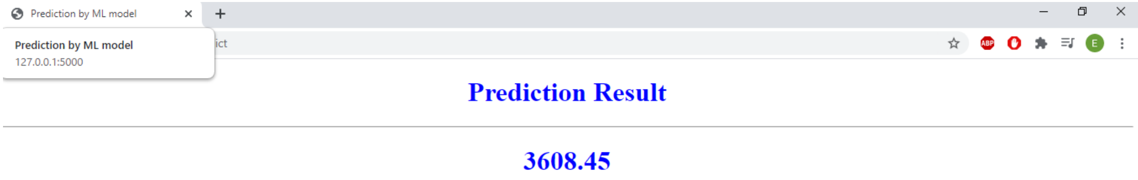
213123

var_4

12312

var_5

23412



Prediction Result

3608.45