

Model Deployment

Name: expense_prediction

Report date: 07/27/2022

Internship Batch: LISUM11: 30

Version: 1.0

Data intake by: Priyadarshani Kamble

Data intake reviewer: NA

Data storage location:

https://bitbucket.org/pk_projects/work/src/week4/Income_Expense_Data.csv

Tabular data details:

Total number of observations	14
Total number of files	1
Total number of features	3
Base format of the file	csv
Size of the data	1KB

Note: Replicate same table with file name if you have more than one file.

Proposed Approach:

A: Create and Pickle a Machine Learning Model

I used Income_Expense dataset and created a model using LinearRegression model. This model will predict the expense of the person using income and age as input features.

Please refer to

https://bitbucket.org/pk_projects/work/src/week4/expense_prediction.ipynb

This model performs below operations:

- Step 1 - Import data
 - o Explore the data
- Step 2-Data Cleaning:
 - o Looks for missing values, replaces missing data with appropriate values using imputation technique.
 - o Checking Outlier by definition and treating outliers
- Step 3- Exploratory data analysis

- Check how Expense is varying with income
- Check how Expense is varying with Age
- Check correlation matrix - to check the strength of variation between two variables
- Step 4-feature engineering
 - Normalization/scaling of data - understanding scaling
 - Converting data back to pandas dataframe
 - Separating features and response
 - Dividing data in test and train
 - Importing necessary packages
 - Fitting linear regression model
 - Checking accuracy on test data
 - Predict values on test data

This creates a model that can predict the expense of person with ~68% accuracy.

B Model Deployment prep:

Model can then be pickled using

```
import pickle
pickle.dump(model, open('expense_prediction.pkl','wb'))
```

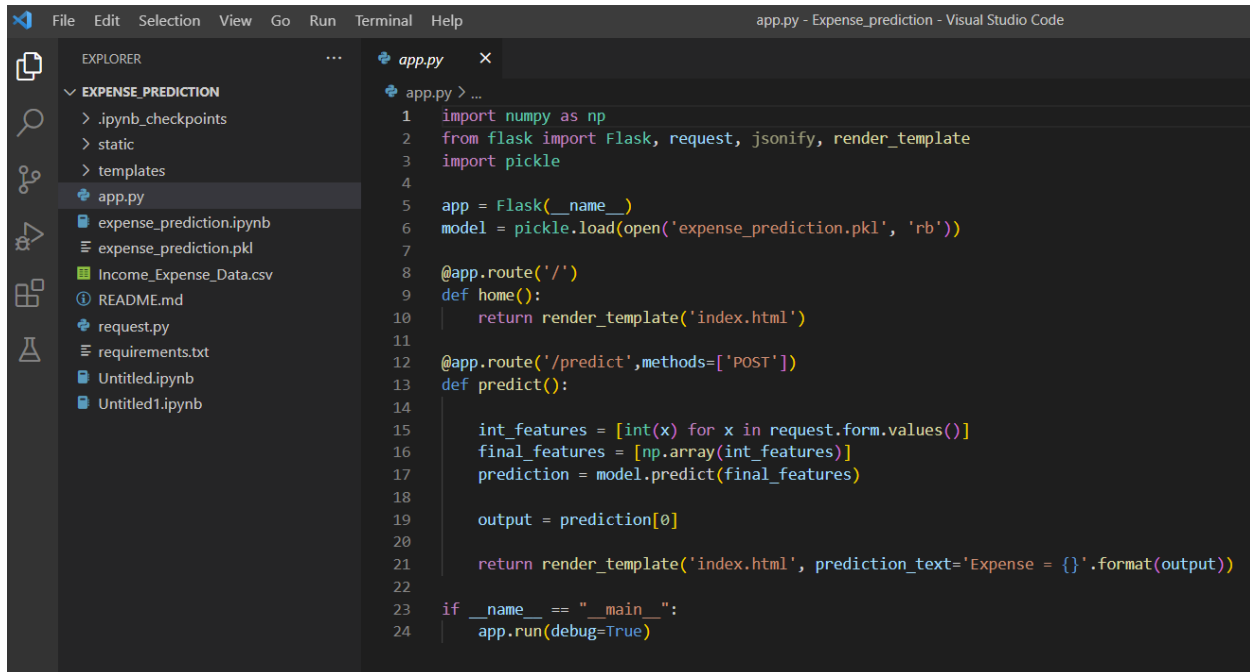
The pickle file can be found inside the same directory as the Jupyter notebook.
https://bitbucket.org/pk_projects/work/src/week4/expense_prediction.pkl

C Write Flask App:

Using IDE or visual studio code, create a new .py file inside the working directory named app.py. which looks like the snapshot below.

The structure of the code follows:

- Load pickled model
- Name flask app
- Create a route that receives inputs through a html page, uses the trained model to make a prediction, and returns the prediction in a html format, which can be accessed through the API endpoint.



```
1 import numpy as np
2 from flask import Flask, request, jsonify, render_template
3 import pickle
4
5 app = Flask(__name__)
6 model = pickle.load(open('expense_prediction.pkl', 'rb'))
7
8 @app.route('/')
9 def home():
10     return render_template('index.html')
11
12 @app.route('/predict', methods=['POST'])
13 def predict():
14
15     int_features = [int(x) for x in request.form.values()]
16     final_features = [np.array(int_features)]
17     prediction = model.predict(final_features)
18
19     output = prediction[0]
20
21     return render_template('index.html', prediction_text='Expense = {}'.format(output))
22
23 if __name__ == "__main__":
24     app.run(debug=True)
```

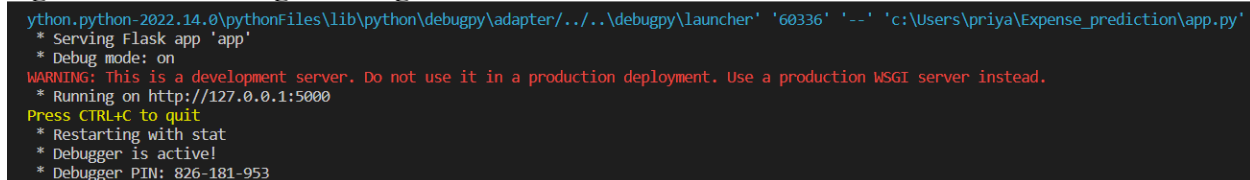
The app file uses render_template function to load index.html
Index.html is nothing but a user interface which accepts input variables for income and age and returns expense as output.
This template can be found in
https://bitbucket.org/pk_projects/work/src/week4/templates/index.html

app file further uses predict() function to predict the output.

D:

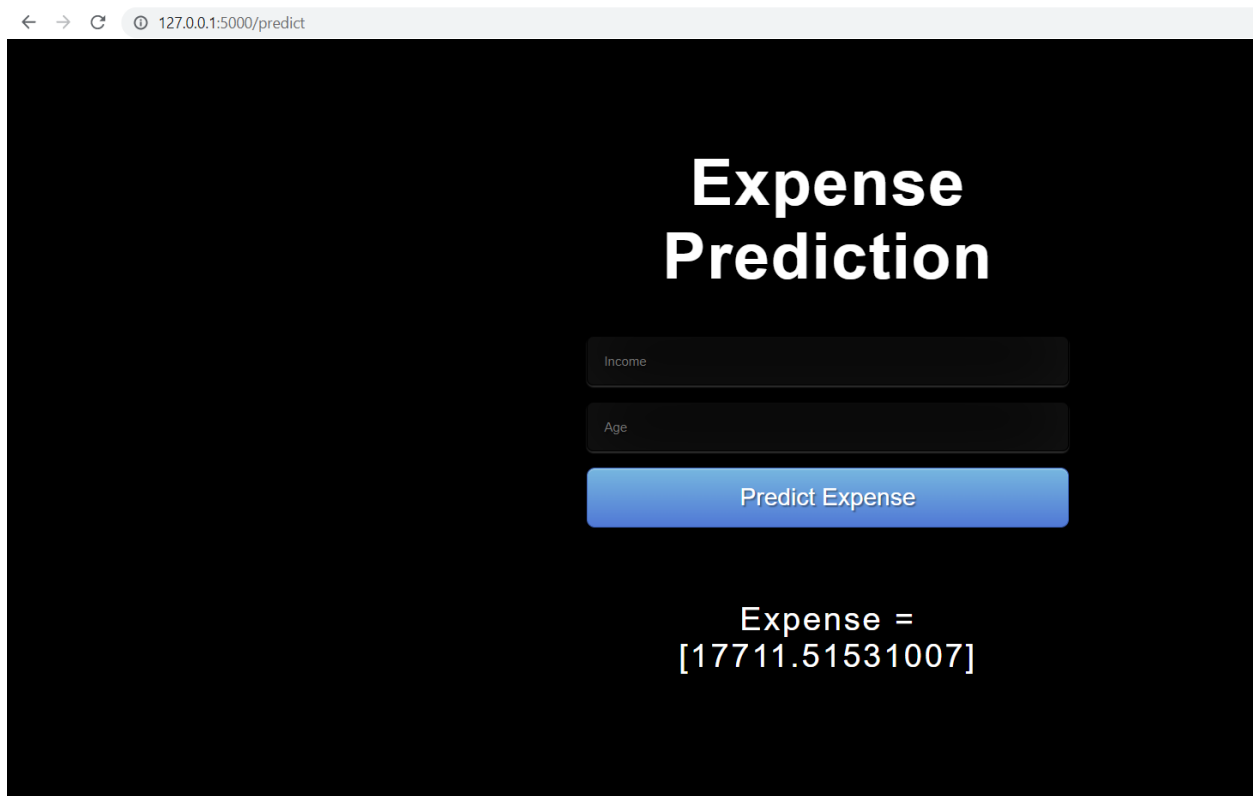
Once the app.py file is ready, I ran the flask app from the command line as below
python app.py

I got the following message with the address:



```
ython.python-2022.14.0\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher '60336' '--' 'c:\Users\priya\Expense_prediction\app.py'
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 826-181-953
```

The app can now be tested using url <http://127.0.0.1:5000>



A screenshot of a web browser showing a web application titled "Expense Prediction". The browser's address bar displays "127.0.0.1:5000/predict". The application has a dark background with white text. It features two input fields labeled "Income" and "Age", followed by a blue button labeled "Predict Expense". Below the button, the predicted expense is displayed as "Expense = [17711.51531007]".

The model predicted 17711 as expense for input variables income = 30000 and age = 24.












Step E : Create requirements.txt

The requirements.txt file contains all the dependencies for the flask app. This can be created using the below command:

```
pip freeze > requirements.txt
```

The folder contains the following files.

;) > Users > priya > Expense_prediction >

<input type="checkbox"/> Name ^	Date modified	Type	Size
 .git	8/27/2022 5:53 PM	File folder	
 .ipynb_checkpoints	8/27/2022 4:05 PM	File folder	
 static	8/27/2022 3:49 PM	File folder	
 templates	8/27/2022 3:49 PM	File folder	
 app	8/27/2022 3:58 PM	PY File	1 KB
 expense_prediction	8/27/2022 3:58 PM	IPYNB File	51 KB
 expense_prediction	8/27/2022 3:58 PM	PKL File	1 KB
 Income_Expense_Data	8/27/2022 3:48 PM	Microsoft Excel Co...	1 KB
 README	8/27/2022 1:28 PM	Markdown Source ...	1 KB
 request	8/27/2022 3:48 PM	PY File	1 KB
 requirements	8/27/2022 5:51 PM	Text Document	1 KB