

Name : Gloria Jepkurui
Batch code : LISUM35
Submission Date : 2 august 2024
Submitted To : Dashboard

Heroku Deployment steps:

1. Used insurance data to train and test a model(project i did 3 months ago). I cloned the insurance repository. The dataset and model from week 4.
2. Saved the model into a file using pickle(serialization)









```
[81]: # saving the trained model to a file using pickle(serilization)

with open('multi_model.pkl', 'wb') as multimodel_file:
    pickle.dump(multi_model, multimodel_file)
```


3. Created an app using flask and loaded the model(deserilization) in a python script. Saving everything in the same directory.

```
10 # deserilization
11 with open(multimodel_path, 'rb') as multimodel_file:
12     multi_model = pickle.load(multimodel_file)
13
14 # creating the flask application
15 in_app = Flask(__name__)
16
17 # defining the prediction endpoints
18 new *
19 @in_app.route(rule: '/predict', methods = ['POST'])
20 def predictin():
21     data = request.get_json(force_= True)
22     prediction = multi_model.predict(np.array(data['features']).reshape(1, -1))
23     return jsonify({'prediction': int(prediction[0])})
24
25 if __name__ == '__main__':
26     in_app.run(debug = True)
27
```





4. Create a repository(Heroku_trial) and uploaded all the files


main	Go to file	+	<> Code
 ruyo003	Add files via upload		d42b576 · 30 minutes ago
 Insurance_app.py	Add files via upload	30 minutes ago	
 README.md	Initial commit	31 minutes ago	
 insurance-2.csv	Add files via upload	30 minutes ago	
 insurance.ipynb	Add files via upload	30 minutes ago	
 multi_model	Add files via upload	30 minutes ago	
 multi_model.pkl	Add files via upload	30 minutes ago	
 requirements.txt	Add files via upload	30 minutes ago	








5. Created Heroku account and linked to it to my github repository


HEROKU

[Jump to Favorite](#)




expense-predictor

[Open app](#)
[More](#)


ruyo003/heroku_trial

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features

Pipelines let you connect

Pipelines connected to GitHub can

6. Deployed my expense predictor model using heroku

Only enable this option if you have a Continuous Integration service configured on your repo.

Enable Automatic Deploys

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more.](#)

Choose a branch to deploy

main

Deploy Branch

7. Got the link below for my prediction.

Choose a branch to deploy

main

Deploy Branch

Receive code from GitHub

Build **main** d42b5767

```
-----> Compressing...
         Done: 215.7M
-----> Launching...
         Released v4
         https://expense-predictor-71f9cc412a86.herokuapp.com/ deployed to Heroku
This app is using the Heroku-22 stack, however a newer stack is available.
To upgrade to Heroku-24, see:
https://devcenter.heroku.com/articles/upgrading-to-the-latest-stack
```

☒ Autoscroll with output

[View build](#)

note : i used the dataset and model from the previous week.