

Week 5: Cloud and API deployment

Data Glacier Virtual Internship

Name: Junfei Liu

Batch Code: LISUM16

Submission Date: Jan 5 2022

Submitted to: Data Glacier

1. Introduction

1.1 Data

The dataset I used to train the model is a csv file containing 50 pairs of randomly generated x, y values, which can be found in Week4/dataset/A.

1.2 Model

I imported the linear regression model in Scikit-learn and trained the model on the data in csv file introduced above. The model is saved to model.pkl using Pickle. This part is in model.py.

1.3 Deployment

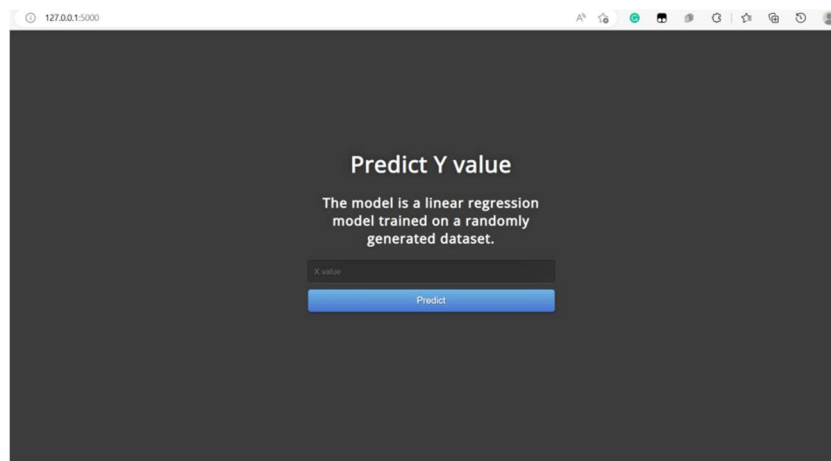
The ML model is deployed through Flask. This part is mainly implemented in app.py.

2. Snapshots

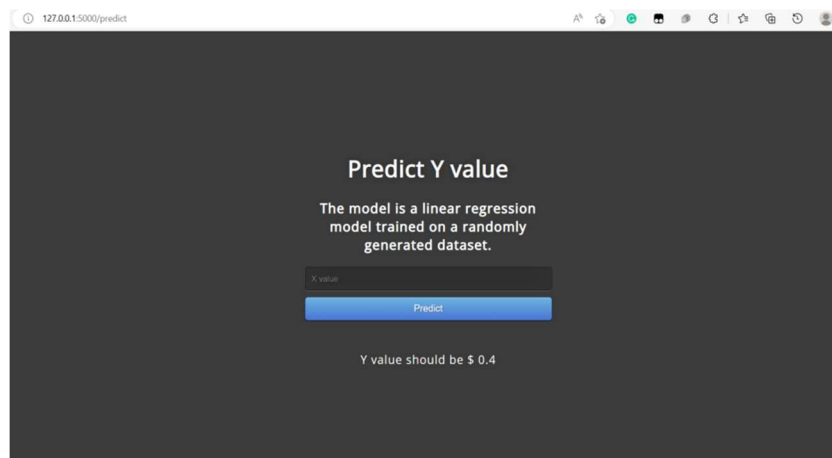
First, start running the app in command line:

```
PS C:\Users\23566\Desktop\Data_Glacier> cd Week4
PS C:\Users\23566\Desktop\Data_Glacier\Week4> python 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: 128-669-135
```

Now it is successfully running. The web page can be accessed on local host at <https://127.0.0.1:5000/>. The page is functioning like below:



By entering 1 in the input line, the model will successfully predict the result and present the result to users as below:



3. Deployment using Fly.io

Because Heroku has switched to no longer providing free plan, I will use Fly.io to deploy the app created in week4.

Fly.io allows to deploy containerized applications using Docker. It is required to have Python, Flask, and Docker installed on the machine to deploy an app using Fly.io. In this task, the app deployed is the linear prediction model created in Week4 using Flask.

'requirements.txt', 'Dockerfile', '.platform.app.yaml', and 'fly.toml' are created as the app deployment configuration with necessary information such as python version, app name, etc. Then, the app can be deployed with 'flyctl apps create <app-name>' and 'flyctl deploy'.

```
Windows PowerShell
PS C:\Users\23566\OneDrive\桌面\data-glacier-week5-cloud-deploy> flyctl apps create testmodel1145
automatically selected personal organization: Junfei Liu
New app created: testmodel1145
PS C:\Users\23566\OneDrive\桌面\data-glacier-week5-cloud-deploy> flyctl deploy
==> Verifying app config
--> Verified app config
==> Building image
Remote builder fly-builder-divine-glitter-3527 ready
==> Creating build context
--> Creating build context done
==> Building image with Docker
--> docker host: 20.10.12 linux x86_64
[+] Building 0.0s (0/1)
[+] Building 6.9s (9/9) FINISHED
  => [internal] load remote build context                                0.0s
  => copy /context /                                                    0.0s
  => [internal] load metadata for docker.io/library/python:3.10.7-slim  1.2s
  => [1/5] FROM docker.io/library/python:3.10.7-slim@sha256:f2ee145f3bc4e061f8dfe7e6ebd427a410121495a0bd26e7622136  2.6s
  => => resolve docker.io/library/python:3.10.7-slim@sha256:f2ee145f3bc4e061f8dfe7e6ebd427a410121495a0bd26e7622136  0.0s
  => => sha256:28358ad706063a4fab219d8bcb048124e6dbc6245015b4267b978363af064d42  7.50kB / 7.50kB  0.0s
  => => sha256:bd159e379b3b1bc0134341e4ffdeab5f966ec422ae04818bb69ecef08a823b05  31.42MB / 31.42MB  0.9s
  => => sha256:de08aeb7fd50562d57cef1a49d6197d619df0b4ce52e4caeba2402c27c6e536b  1.08MB / 1.08MB  0.2s
  => => sha256:30527e10f55af538c1464ce755f9e9fcaae68bb639f9b8211cfaf3c11e83ba42  12.11MB / 12.11MB  0.9s
  => => sha256:693e7a5ba2a8a42880d04042159d67aa9ebb2d67864b7b419fe2847294c088a2  234B / 234B  0.4s
  => => sha256:c7b6f7685fa51adb5da18c9a4fd647bc9cb24af571c32a824206e8a9277d7b09  3.34MB / 3.34MB  0.7s
  => => sha256:f2ee145f3bc4e061f8dfe7e6ebd427a410121495a0bd26e7622136db060c59e0  1.86kB / 1.86kB  0.0s
  => => sha256:f8fbb2370c6314c806b2ddb6c8d94375987e16bc122379bef979c6fc5e962920  1.37kB / 1.37kB  0.0s
  => => extracting sha256:bd159e379b3b1bc0134341e4ffdeab5f966ec422ae04818bb69ecef08a823b05  0.8s
```

```
Windows PowerShell
=> => extracting sha256:c7b6f7685fa51adb5da18c9a4fd647bc9cb24af571c32a824206e8a9277d7b09  0.2s
=> [2/5] WORKDIR /app 0.1s
=> [3/5] COPY requirements.txt . 0.0s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt 2.8s
=> [5/5] COPY . . 0.0s
=> exporting to image 0.2s
=> exporting layers 0.1s
=> writing image sha256:593818ec4be76e980f9174e3cd2354c7bae61447de85f6451d812ae6cb31baa7 0.0s
=> naming to registry.fly.io/testmodel1145:deployment-01GVYS74RXS55WEX88VFVSAM4R 0.0s
--> Building image done
==> Pushing image to fly
The push refers to repository [registry.fly.io/testmodel1145]
7bebc80a014: Pushed
fee82e8016f6: Pushed
6c33624d181a: Pushed
ab0105a77058: Pushed
31f6a8e43a88: Pushed
55e0611d9389: Pushed
ff12e0cbd8e0: Pushed
1169b1563e05: Pushed
fe7b1e9bf792: Pushed
deployment-01GVYS74RXS55WEX88VFVSAM4R: digest: sha256:674f4052aa36c06625e359071606f50ff969baeffd7c62c04abf68524a9e4596 s
ize: 2204
--> Pushing image done
image: registry.fly.io/testmodel1145:deployment-01GVYS74RXS55WEX88VFVSAM4R
image size: 137 MB
==> Creating release
--> release v2 created
--> You can detach the terminal anytime without stopping the deployment
```

The deployment was successful as the docker image has been created.

Deployment related configuration files are included in git repo.

4. Summary

The deployment of model using Flask is successful in general as the web page can normally function to take inputs and the model can yield predictions.