

Deployment using Flask onto Heroku

Name: Jeffery Su

Batch Code: LISUM11:30

Submission Data: August 3, 2022

Submitted to: Canvas/Github

Selecting Toy Data

7.1.2. Iris plants dataset

Data Set Characteristics:

Number of Instances:	150 (50 in each of three classes)
Number of Attributes:	4 numeric, predictive attributes and the class
Attribute Information:	<ul style="list-style-type: none">• sepal length in cm• sepal width in cm• petal length in cm• petal width in cm• class:<ul style="list-style-type: none">◦ Iris-Setosa◦ Iris-Versicolour◦ Iris-Virginica

The toy data I'm going to use is dataset of Iris plants from scikit-learn. The dataset have 150 instances and 4 attributes (sepal length, sepal width, petal length, and petal width).

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

Model Development

Im going to create a model that uses sepal length, sepal width and petal width to predict the petal length.

First I'm using Jupyter notebook to create the model

```
In [4]: # sepal length, sepal width and petal width
X = df.iloc[:, [0,1,3]]
# petal length
Y = df.iloc[:, [2]]
# splitting the data
xtrain, xtest, ytrain, ytest = train_test_split(X, Y, test_size = 0.2, random_state = 0)
```

```
In [5]: # building model
regressor = LinearRegression()
regressor.fit(xtrain, ytrain)
```

```
Out[5]: LinearRegression()
```

```
In [6]: print('The intercept is {} and the coef are {}'.format(regressor.intercept_, regressor.coef_))

The intercept is [-0.26533136] and the coef are [[ 0.7005897 -0.61042077  1.48167646]]
```

```
In [7]: ypred = regressor.predict(xtest).tolist()
ypred_ = [val for sublist in ypred for val in sublist]

ytest['predicted'] = ypred_
ytest
```

Out[10]:




	petal length (cm)	predicted
114	5.1	5.644934
62	4.0	4.076958
33	1.4	1.320480
107	6.3	5.745771
7	1.5	1.458522
100	6.0	5.838186
40	1.3	1.545647
86	4.7	4.758830
76	4.8	4.863848
71	4.0	4.225267
134	5.6	4.495519
51	4.5	4.487611
73	4.7	4.077099
54	4.6	4.801838
63	4.7	4.312393
37	1.4	1.118211
78	4.5	4.390501
90	4.4	3.778830
45	1.4	1.710740
16	1.3	1.729883
121	4.9	4.912146
66	4.5	4.049223
24	1.9	1.318404
8	1.4	1.343378
126	4.8	5.036164
22	1.0	1.056202
44	1.9	1.580748
97	4.3	4.234284
93	3.3	3.315326
26	1.6	1.754857

I see the predicted petal length is pretty similar to the actual petal length, so I know the model is correct.

Then I'm going to pack the model with pickle

```
In [17]: # pack the model with pickle
with open('model.pkl', 'wb') as files:
    pickle.dump(regressor, files)
```

Now there is a file with the model call model.pkl

 .ipynb_checkpoints	2022-07-26 3:55 PM	File folder	
 linear_regression_model_of_petal_lengt...	2022-07-28 3:50 PM	Jupyter Source File	15 KB
 model.pkl	2022-07-28 3:50 PM	PKL File	1 KB

Deployment on Flask

I'm using pycharm to code flask.

First I need to build a website with html. It's just a very simple website with 3 text box and a submit button.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <title>Web form</title>
5  </head>
6  <body>
7      <form action="{{ url_for('predict') }}" method="POST">
8          <p>Please enter your iris information below.</p>
9          <table border="0">
10             <tr>
11                 <td>Sepal Length (cm) </td>
12                 <td><input type="text" placeholder="5.1" name="sl" required></td>
13             </tr>
14             <tr>
15                 <td>Sepal Width (cm) </td>
16                 <td><input type="text" placeholder="3.5" name="sw" required></td>
17             </tr>
18             <tr>
19                 <td>Petal Width (cm) </td>
20                 <td><input type="text" placeholder="0.2" name="pw" required></td>
21             </tr>
22         </table>
23         <button type="submit">Submit</button>
24     </form>
25     <br>
26     {{ prediction_text }}
27 </body>
28 </html>
```

Please enter your iris information below.

Sepal Length (cm) :

Sepal Width (cm) :

Petal Width (cm) :

Now its time for flask.

```
1  import numpy as np
2  import pickle
3  from flask import Flask, request, render_template, url_for
4
5  app = Flask(__name__, template_folder='templates')
6  model = pickle.load(open('model.pkl', 'rb'))
7
8
9  @app.route('/')
10 def index():
11     return render_template('index.html')
12
13
14 @app.route('/predict', methods=['POST'])
15 def predict():
16     sepal_length = request.form['sl']
17     sepal_width = request.form['sw']
18     petal_width = request.form['pw']
19
20     prediction = model.predict(np.asarray([sepal_length, sepal_width, petal_width], dtype=float).reshape(-1,3))
21
22     output = np.round(prediction[0, 0], 2)
23
24     return render_template('index.html', prediction_text='Predicted Petal Length is {} with sepal length: {}, sepal'
25                                     ' width: {} and petal width: {}'.format(output,
26                                     sepal_length,
27                                     sepal_width,
28                                     petal_width))
```

Here is a test run:

Please enter your iris information below.

Sepal Length (cm) :	<input type="text" value="5.1"/>
Sepal Width (cm) :	<input type="text" value="3.5"/>
Petal Width (cm) :	<input type="text" value="0.2"/>
<input type="button" value="Submit"/>	

Predicted Petal Length is 1.47 with sepal length: 5.1, sepal width: 3.5 and petal width: 0.2

The flask server is up and can be access at <http://127.0.0.1:5000/>.

Deploy App onto Heroku

First, login onto Heroku or sign up for a account.



Log in to your account

Email address

Password

Log In

New to Heroku? [Sign Up](#)

[Log in via SSO](#)

[Forgot your password?](#)

Then, create a new app

Create New App

App name

Choose a region



United States



Add to pipeline...

Create app

Now I need to connect the app to my GitHub

Connected to  [sujeffer1999/DG_Deployment](#) by  [sujeffer1999](#)

Disconnect...

🔗 Releases in the [activity feed](#) link to GitHub to view commit diffs



You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

Enable automatic deploys from GitHub

Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more](#).

Choose a branch to deploy

 main

☐ Wait for CI to pass before deploy

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

Enable Automatic Deploys

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

Before we deploy, I first need to add some required files into my GitHub repository.


I need to add a requirements.txt file that tells Heroku what version my python code is dependent on.

main

DG_Deployment / requirements.txt

Go to file

...

 **sujeffer1999** change req




Latest commit 3a1423f 21 hours ago

History

1 contributor

11 lines (11 sloc) | 176 Bytes

RawBlame



```
1 Flask==1.1.1
2 gunicorn==19.9.0
3 itsdangerous==1.1.0
4 Jinja2==2.10.1
5 MarkupSafe==1.1.1
6 Werkzeug==0.15.5
7 numpy>=1.9.2
8 scipy>=0.15.1
9 scikit-learn>=0.18
10 matplotlib>=1.4.3
11 pandas>=0.19
```


And I need a Procfile to tell Heroku how to run my code with the Procfile file.

main

DG_Deployment / Procfile

Go to file

...

 **sujeffer1999** changed procfile




Latest commit d67bee2 21 hours ago

History

1 contributor

1 lines (1 sloc) | 22 Bytes

RawBlame



```
1 web: gunicorn app:app
```


Lastly I need to specify the python version my code can run in with the runtime.txt file.

main

DG_Deployment / runtime.txt

Go to file

...

 **sujeffer1999** added runtime.txt




Latest commit cad643c 21 hours ago

History

1 contributor

1 lines (1 sloc) | 12 Bytes

RawBlame



```
1 python-3.9.0
```

Now, we can finally deploy my app on Heroku.

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

Receive code from GitHub	✓
Build main cad643c2	✓
Release phase	✓
Deploy to Heroku	✓

Your app was successfully deployed.

 View

Click on “View” to view the app.

Please enter your iris information below.

Sepal Length (cm) :	<input type="text" value="5.1"/>
Sepal Width (cm) :	<input type="text" value="3.5"/>
Petal Width (cm) :	<input type="text" value="0.2"/>
<input type="button" value="Submit"/>	

Fill in the sepal length, sepal width and petal width to predict the petal length.

The app can be viewed on <https://dg-deployment-iris.herokuapp.com/>