

# PROJECT 3

DUE: 11:59PM, MARCH 29.

## 1 Introduction

In this project, you will need to turn one of your predictor into a web application. This web application should be able to get input from users and make a prediction which class the given unlabeled data belongs to. No matter what your predictor is, you should explain well in your application to the user what you are predicting and what kind of input data you are expecting from the user.

**Bonus question.** You will get an extra 5 points if you can implement the following function in your application. After the prediction output is given, the user can decide whether this prediction is correct. Using the decision of the user, your application will adjust your prediction model so as to make more accurate predictions in the future.

## 2 Tools

### 2.1 Flask

Flask is a framework to develop your web applications. After Armin Ronacher's initial release of Flask in 2010, the framework has gained huge popularity over the years and examples of popular applications that make use of Flask include LinkedIn and Pinterest. Since Flask is written in Python, it provides us Python programmers with a convenient interface for embedding existing Python code such as our predictors. For more details, you are encouraged to visit the official document of Flask (<http://flask.pocoo.org/>).

To install Flask, simply typing

```
$ pip install flask
```

in your terminal. To avoid clutter, I suggest you use Python virtual environment, for example, *virtualenv* or *Anaconda*.

To give you a general idea how Flask works, I will follow the tutorial in the book to implement an easy application. First, we create a directory tree:

```
flash_flask_app_1/  
    flask_app.py  
        templates/  
            first_app.html
```

The *flask\_app.py* file will contain the main code that will be executed by the Python interpreter to run the Flask web application. The templates directory is the directory in which Flask will look for static HTML files for rendering in the web browser. Let's now take a look at the contents of *flask\_app.py*:

```
from flask import Flask, render_template
app = Flask(__name__)

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

if __name__ == '__main__':
    app.run()
```

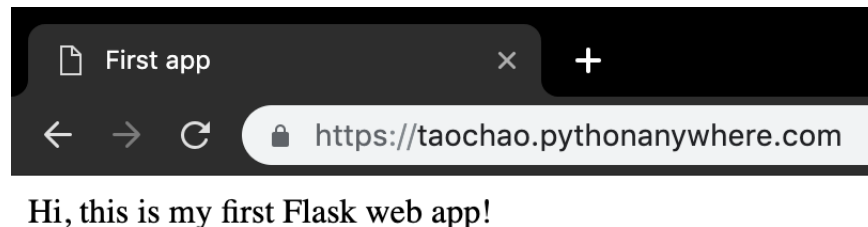
Now, let's take a look at the contents of the *first\_app.html* file:

```
<!doctype html>
<html>
  <head>
    <title>First app</title>
  </head>
  <body>
    <div>Hi, this is my first Flask web app!</div>
  </body>
</html>
```

To run this application, simply run

```
$ python flask_app.py
```

in your terminal and then open <http://127.0.0.1:5000/> in one of your favorite internet explorer.



For more details on how to implement a web application using Flask, please read the tutorial in Section 3-5 of Python Machine Learning textbook (Pg. 257-276). The next step is just deploying the web application to a public server.

## 2.2 PythonAnywhere

PythonAnywhere (<https://www.pythonanywhere.com/>) is such a server we can utilize. The good thing is it allows everyone to hold one free Flask application. Note that by default the server runs *flask\_app.py* file to start the application. Make sure to use this name before uploading your files. It seems that you can only upload files one by one instead of using *ssh*.

## 3 Submission

You will need to submit one **zip** file which includes all the source code before the due date. Fail to do so will make your final grade deducted. Apart from that, put website name (in the form of *XXX.pythonanywhere.com*) in the comment of your submission so that we can test it. We will check the website right after the deadline. So make sure your website is working before that time. In the report, you should specify your model details when necessary. Try to write your code clearly so that someone else reading the code can understand it without significant effort (i.e. structure it and put enough documentation).

## 4 Collaboration

Note that this is an **independent** project, which means you are not allowed to make a group. However, discussion is allowed. If you have discussed with someone or got any help from others, you need to clearly specify their names in acknowledgement.