

섹션3 flask 웹 배포



Machine Learning Repository

Center for Machine Learning and Intelligent Systems

Check out the [beta version](#) of the new UCI Machine Learning Repository we are currently testing.

Iris Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Famous database; from Fisher, 1936



Data Set Characteristics:	Multivariate	Number of Instances:	150	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	4	Date Donated	1988-07-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	4482665



IRIS라 불리는 꽃들의 이파리 길이를 학습

각 잎사귀 길이를 입력하면 알맞은 품종이 나옴

```

1 import sqlite3
2 import csv
3
4 conn = sqlite3.connect('iris.db')
5 cur = conn.cursor()
6
7 cur.execute(['DROP TABLE IF EXISTS iris'])
8 cur.execute('''
9 CREATE TABLE "iris"(
10     first_value INT,
11     second_value INT,
12     third_value INT,
13     forth_value INT,
14     iris_name Text
15 )
16 ''')
17
18 pt = list()
19 with open('./iris.data', encoding='UTF8', newline='') as csvfile:
20     reader = csv.reader(csvfile)
21     i=0
22     for row in reader:
23         li = list(row)
24         li.insert(0,i-1)
25         tup = tuple(li)
26         pt.append(tup)
27         i += 1
28

```

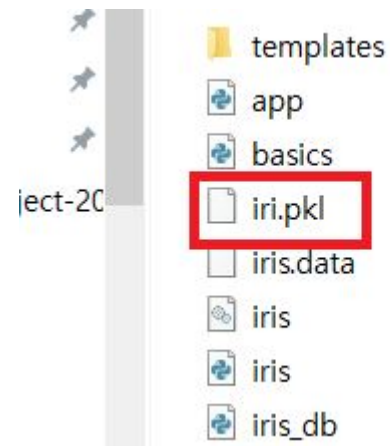
iris Enter a SQL expression to filter results (use Ctrl+Space)

	123 first_value	123 second_value	123 third_value	123 forth_value	Value
1	5.1	3.5	1.4	0.2	Iris-
2	4.9	3	1.4	0.2	Iris-
3	4.7	3.2	1.3	0.2	Iris-
4	4.6	3.1	1.5	0.2	Iris-
5	5	3.6	1.4	0.2	Iris-
6	5.4	3.9	1.7	0.4	Iris-
7	4.6	3.4	1.4	0.3	Iris-
8	5	3.4	1.5	0.2	Iris-
9	4.4	2.9	1.4	0.2	Iris-
10	4.9	3.1	1.5	0.1	Iris-
11	5.4	3.7	1.5	0.2	Iris-
12	4.8	3.4	1.6	0.2	Iris-
13	4.8	3	1.4	0.1	Iris-
14	4.3	3	1.1	0.1	Iris-
15	5.8	4	1.2	0.2	Iris-
16	5.7	4.4	1.5	0.4	Iris-
17	5.4	3.9	1.3	0.4	Iris-
18	5.1	3.5	1.4	0.3	Iris-
19	5.7	3.8	1.7	0.3	Iris-
20	5.1	3.8	1.5	0.2	Iris-

150 row(s) fetched - 1ms (+2m)

```
41 5.0,3.5,1.3,0.3,Iris-setosa
42 4.5,2.3,1.3,0.3,Iris-setosa
43 4.4,3.2,1.3,0.2,Iris-setosa
44 5.0,3.5,1.6,0.6,Iris-setosa
45 5.1,3.8,1.9,0.4,Iris-setosa
46 4.8,3.0,1.4,0.3,Iris-setosa
47 5.1,3.8,1.6,0.2,Iris-setosa
48 4.6,3.2,1.4,0.2,Iris-setosa
49 5.3,3.7,1.5,0.2,Iris-setosa
50 5.0,3.3,1.4,0.2,Iris-setosa
51 7.0,3.2,4.7,1.4,Iris-versicolor
52 6.4,3.2,4.5,1.5,Iris-versicolor
53 6.9,3.1,4.9,1.5,Iris-versicolor
54 5.5,2.3,4.0,1.3,Iris-versicolor
55 6.5,2.8,4.6,1.5,Iris-versicolor
56 5.7,2.8,4.5,1.3,Iris-versicolor
57 6.2,2.2,4.7,1.6,Iris-versicolor
```

```
1 import pandas as pd
2 import numpy as np
3 import pickle
4
5 df = pd.read_csv('iris.data')
6
7 x = np.array(df.iloc[:, 0:4])
8 y = np.array(df.iloc[:, 4:])
9
10 from sklearn.preprocessing import LabelEncoder
11 le = LabelEncoder()
12 y = le.fit_transform(y.reshape(-1))
13
14 from sklearn.model_selection import train_test_split
15 x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2)
16
17 from sklearn.svm import SVC
18 sv = SVC(kernel='linear').fit(x_train,y_train)
19
20
21 pickle.dump(sv, open('iri.pkl', 'wb'))
22
```



```
> Users > ChanHeeHwang > session3_project > flask_app > app.py > ...  
1  from flask import Flask, render_template, request  
2  import pickle  
3  import numpy as np  
4  
5  model = pickle.load(open('iri.pkl', 'rb'))  
6  
7  app = Flask(__name__)  
8  
9  
10  
11 @app.route('/')  
12 def man():  
13     return render_template('home.html')  
14
```

```

0
1 @app.route('/')
2 def man():
3     return render_template('home.html')
4
5
6 @app.route('/predict', methods=['POST'])
7 def home():
8     data1 = request.form['a']
9     data2 = request.form['b']
10    data3 = request.form['c']
11    data4 = request.form['d']
12    arr = np.array([[data1, data2, data3, data4]])
13    pred = model.predict(arr)
14    return render_template('after.html', data=pred)
15
16
17 if __name__ == "__main__":
18     app.run(debug=True)
19
20

```

```

<html>
  <body bgcolor=#d4a3ae>

    <center>

      <h1> IRIS FLOWER DETECTION </h1><br>

      <form method="POST", action="{{url_for('home')}}">
        <b> First value : <input type="text", name="a", placeholder="enter 1"> <br><br>
        Second value : <input type="text", name='b', placeholder="enter 2"> <br><br>
        Third value : <input type="text", name='c', placeholder="enter 3"> <br><br>
        Fourth value : <input type="text", name='d', placeholder="enter 4"> <br><br><br></b>
        <input type="submit" , value='predict!' >
      </form>

      <img src='static\flower1.jpg' alt="flower">

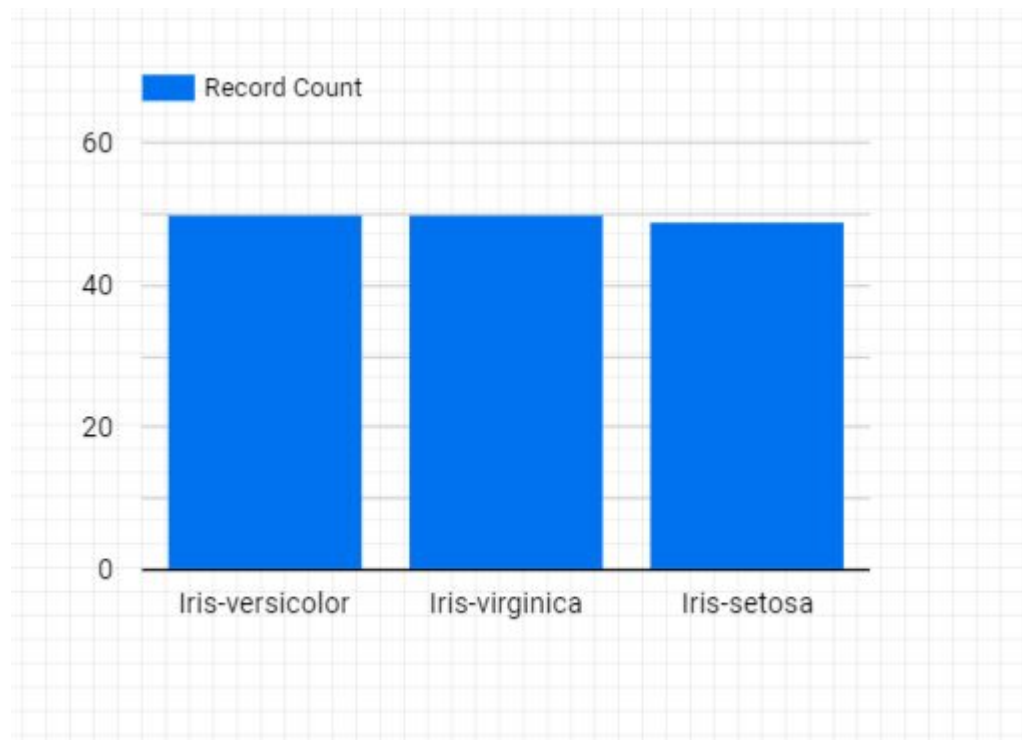
    </center>

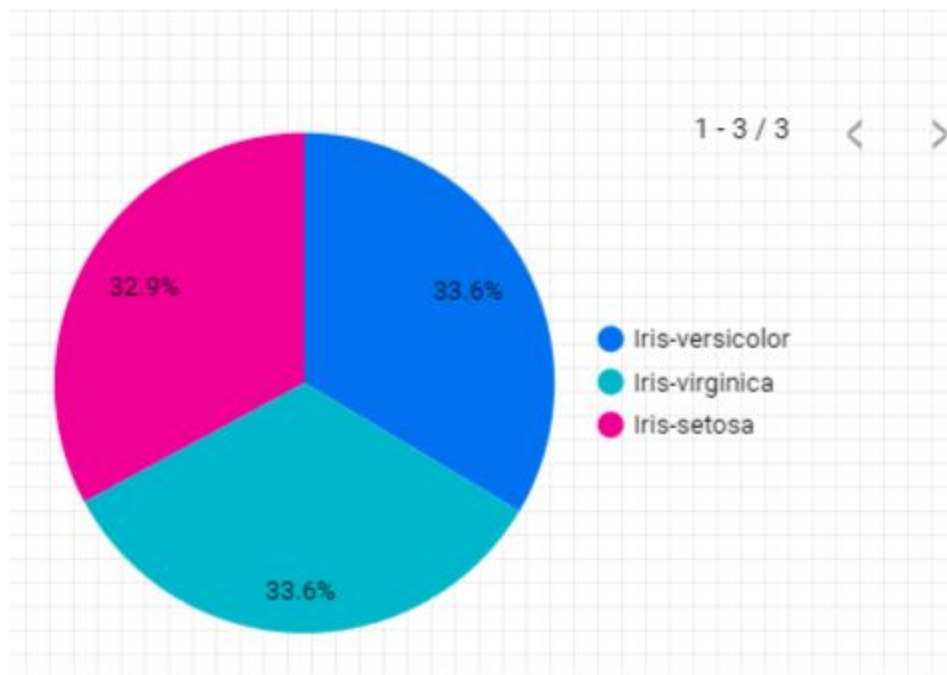
  </body>
</html>

```

구글 데이터 스튜디오 분석결과

Iris-setosa		Record Count ▾
1.	Iris-versicolor	50
2.	Iris-virginica	50
3.	Iris-setosa	49





프로젝트 시현