

Arda Baris Basaran

Batch code: LISUM10

Data: Iris.csv

I used logistic regression

Codes can be found on <https://github.com/ardaxz99/Data-Glacier-Week-4.git>

```
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
import pickle
from sklearn.linear_model import LogisticRegression

# Load the csv file
df = pd.read_csv("iris.csv")

X = df[["Sepal_Length", "Sepal_Width", "Petal_Length", "Petal_Width"]]
y = df["class"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=50)

sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

model = LogisticRegression(solver='lbfgs', multi_class='ovr')

model.fit(X_train, y_train)

pickle.dump(model, open("model.pkl", "wb"))
```

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle

flask_app = Flask(__name__)
model = pickle.load(open("model.pkl", "rb"))

@flask_app.route("/")
def Home():
    return render_template("index.html")

@flask_app.route("/predict", methods = ["POST"])
def predict():
    float_features = [float(x) for x in request.form.values()]
    features = [np.array(float_features)]
    prediction = model.predict(features)
    return render_template("index.html", prediction_text = "The flower species is {}".format(prediction))

if __name__ == "__main__":
    flask_app.run(debug=True)
```



Flower Class Prediction

The flower species is ['Virginica']