

Supervised Learning-Classification Models\SVM-For-Social-Network-ads.py

```
1  # ----- Support Vector Machine (SVM) ----- #
2
3  # Importing the libraries
4  import numpy as np
5  import matplotlib.pyplot as plt
6  import pandas as pd
7  import pickle
8
9  # Importing the dataset
10 dataset = pd.read_csv(r"C:\Users\Jan Saïda\OneDrive\Documents\Desktop\Excel sheets\Social_Network_Ads.csv")
11 X = dataset.iloc[:, [2, 3]].values
12 y = dataset.iloc[:, -1].values
13
14 # Splitting the dataset into the Training set and Test set
15 from sklearn.model_selection import train_test_split
16 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)
17
18 # Feature Scaling
19 from sklearn.preprocessing import StandardScaler
20 sc = StandardScaler()
21 X_train = sc.fit_transform(X_train)
22 X_test = sc.transform(X_test)
23
24 # Training the SVM model on the Training set
25 from sklearn.svm import SVC
26 classifier = SVC()
27 classifier.fit(X_train, y_train)
28
29
30
31 # Predicting the Test set results
32 y_pred = classifier.predict(X_test)
33
34 # Making the Confusion Matrix
```

```
35 from sklearn.metrics import confusion_matrix
36 cm = confusion_matrix(y_test, y_pred)
37 print(cm)
38
39 # This is to get the Models Accuracy
40 from sklearn.metrics import accuracy_score
41 ac = accuracy_score(y_test, y_pred)
42 print(ac)
43
44 bias = classifier.score(X_train,y_train)
45 print(bias)
46
47 variance = classifier.score(X_test,y_test)
48 print(variance)
49
50 # This is to get the Classification Report
51 from sklearn.metrics import classification_report
52 cr = classification_report(y_test, y_pred)
53 cr
54
55 # Pickling the trained model to a file
56 with open('svm_model.pkl', 'wb') as model_file:
57     pickle.dump(classifier, model_file)
58     print("Model saved to 'svm_model.pkl'")
```

~\AppData\Roaming\Code\logs\20250125T225615\window1\exthost\output_logging_20250125T225618\10-Code.log

```
1 [Running] python -u "c:\VS Code\Supervised Learning-Classification Models\SVM-For-Social-Network-ads.py"
2 [[55 3]
3  [ 1 21]]
4 0.95
5 0.903125
6 0.95
7 Model saved to 'svm_model.pkl'
8
9 [Done] exited with code=0 in 4.864 seconds
10
11
```

Streamlit\SVM-Model-Socaial-Network-ads-app.py

```
1 # frontend_streamlit.py
2
3 import streamlit as st
4 import pickle
5 import numpy as np
6 from sklearn.preprocessing import StandardScaler
7
8 # Load the pickled SVM model
9 with open('svm_model.pkl', 'rb') as model_file:
10     classifier = pickle.load(model_file)
11     scaler = StandardScaler()
12
13 # Title of the Web App
14 st.title("Support Vector Machine (SVM) Prediction Web App")
15
16 # Description
17 st.write("""
18 This is a simple web app to predict the outcome of the Social Network Ads dataset using a trained Support Vector Machine (SVM) model.
19 You can input features such as Age and Estimated Salary, and the app will predict whether the person will buy the product (1) or not (0).
20 """)
21
22 # User inputs for Age and Estimated Salary
23 age = st.number_input("Enter Age", min_value=18, max_value=100, step=1)
24 salary = st.number_input("Enter Estimated Salary", min_value=10000, max_value=150000, step=100)
25
26 # Button to make a prediction
27 if st.button("Make Prediction"):
28     # Prepare the input data
29     user_input = np.array([[age, salary]])
30
31     # Feature Scaling (same as in training)
32     user_input_scaled = scaler.fit_transform(user_input)
33
34     # Predict the result using the loaded model
```

```
35 prediction = classifier.predict(user_input_scaled)
36
37 # Display the prediction
38 if prediction == 1:
39     st.write("Prediction: The person will buy the product (1).")
40 else:
41     st.write("Prediction: The person will not buy the product (0).")
42
43
```

Support Vector Machine (SVM) Prediction Web App

This is a simple web app to predict the outcome of the Social Network Ads dataset using a trained Support Vector Machine (SVM) model. You can input features such as Age and Estimated Salary, and the app will predict whether the person will buy the product (1) or not (0).

Enter Age

35

-

+

Enter Estimated Salary

60000

-

+

Make Prediction

Prediction: The person will not buy the product (0).