Laporan Praktikum
Pembelajaran Mesin SVM

Disusun oleh :
Kelompok 3

1. Gede Rangga Wira Aditya (082011633048)
2. Muhammad Rahmadhani Ferdiansyah (082011633068)
3. Arya Danu Triatmodjo (082011633069)
4. Mukhamad Ikhsanudin (082011633086)

Kelas I3

PROGRAM STUDI S1 SISTEM INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI

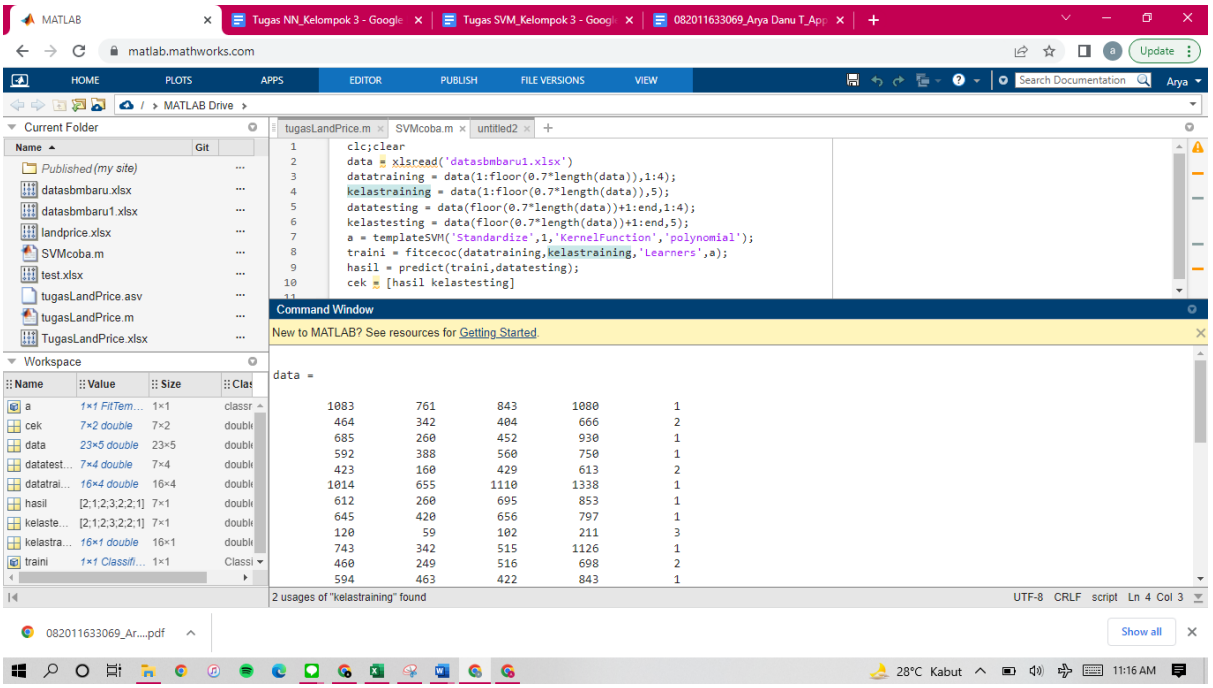
UNIVERSITAS AIRLANGGA

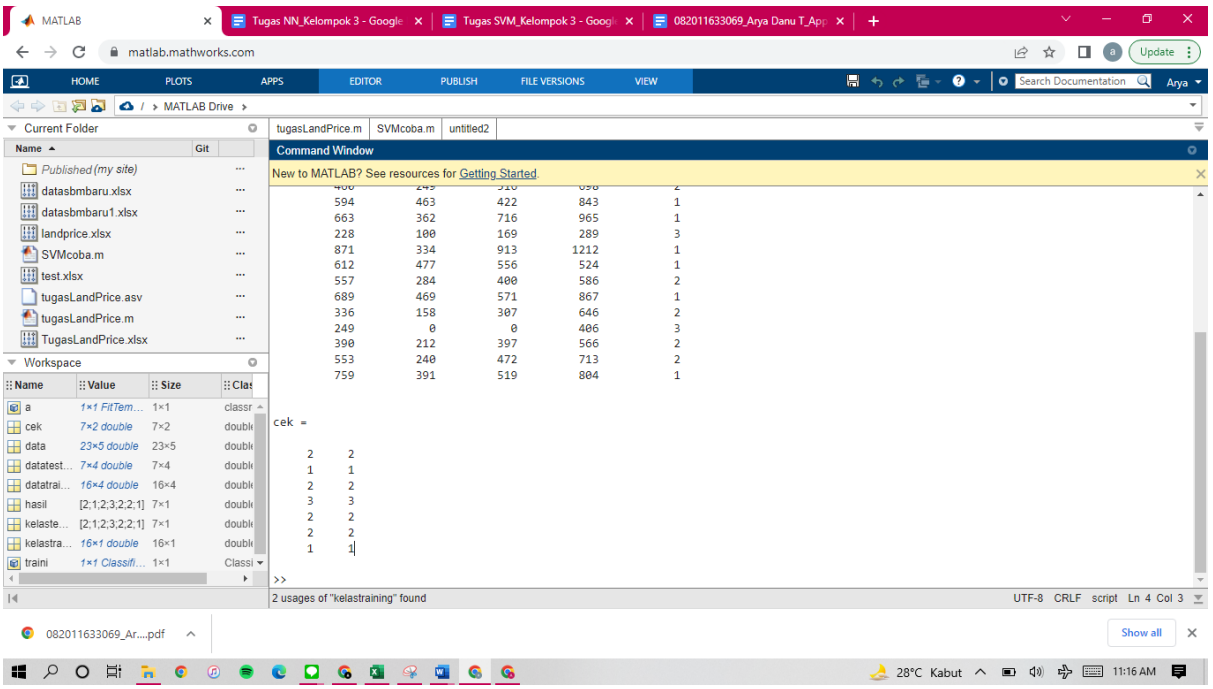
SURABAYA

2021/2022

COBA SEMUA SOURCE CODE SVM

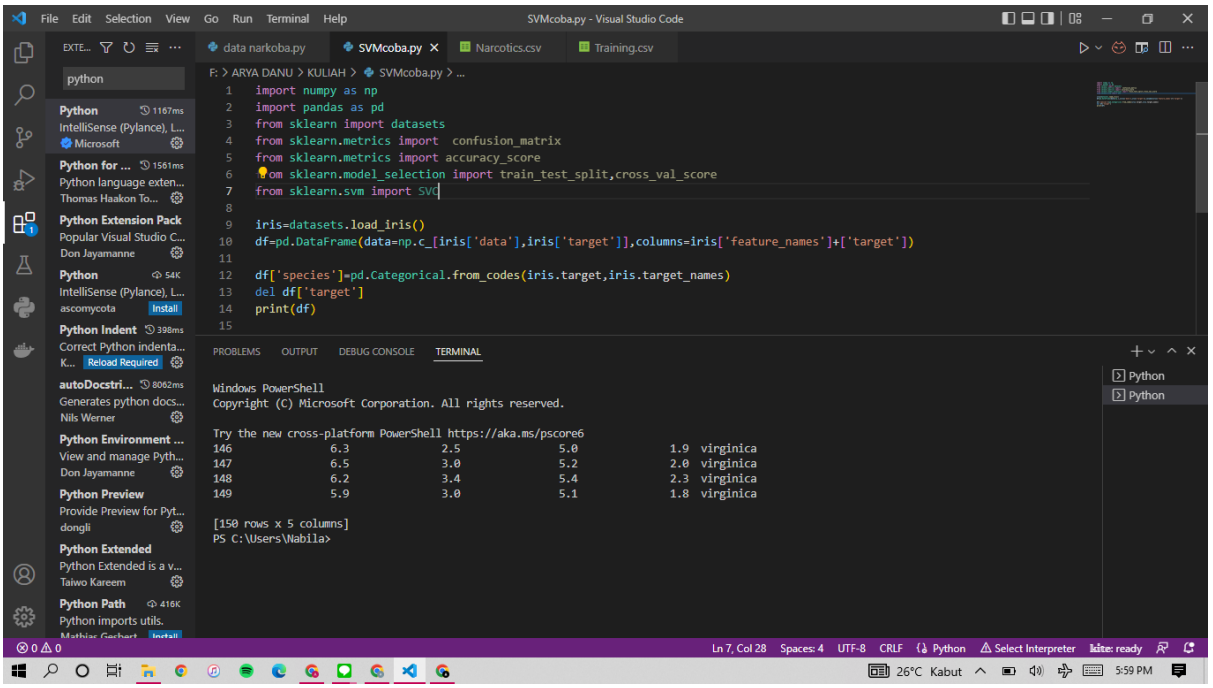
MATLAB



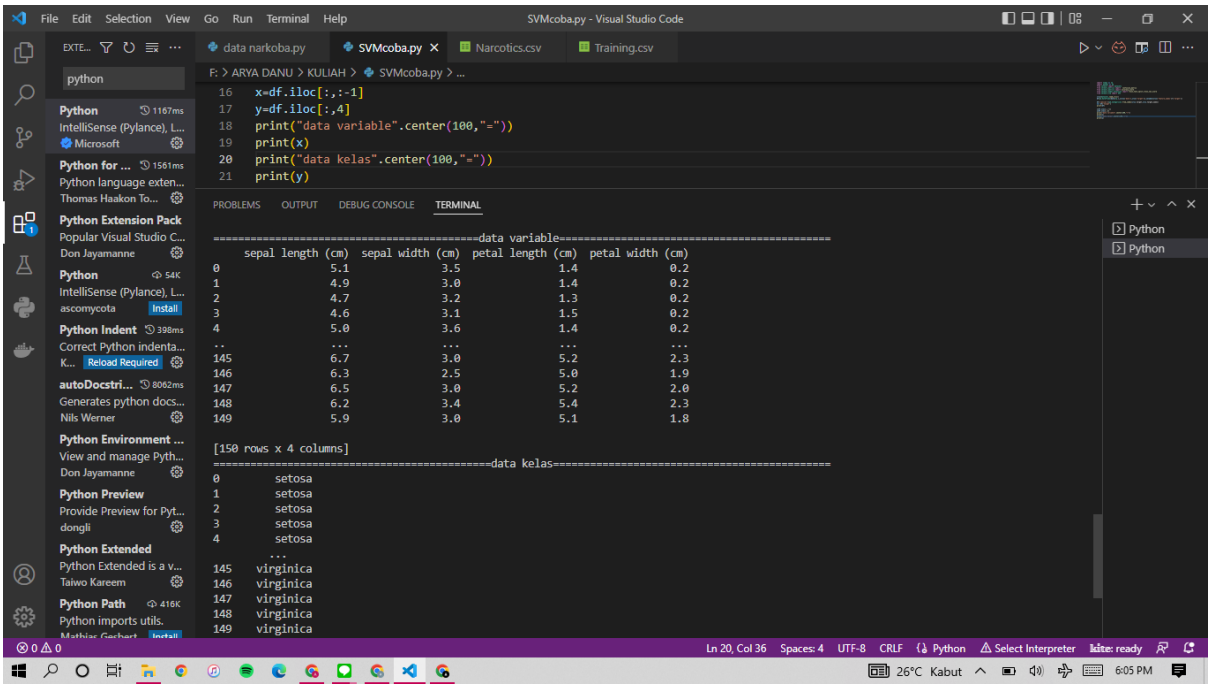


PYTHON

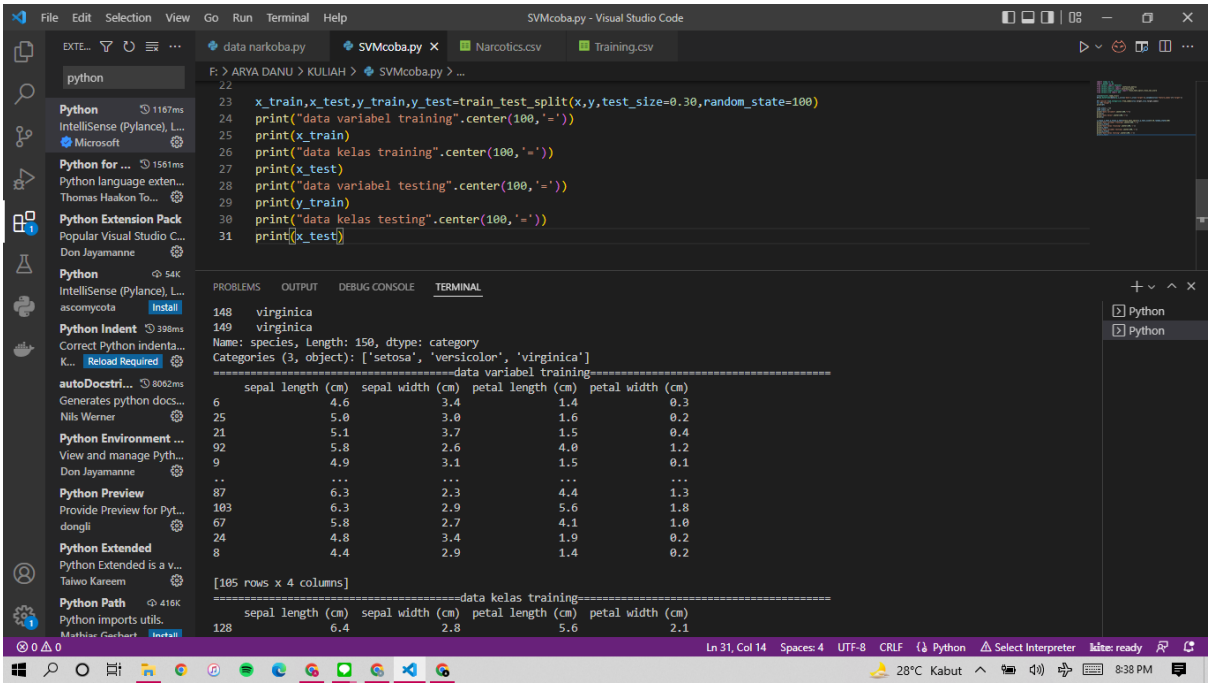
A. Import Library, Load dataset iris and change dataset

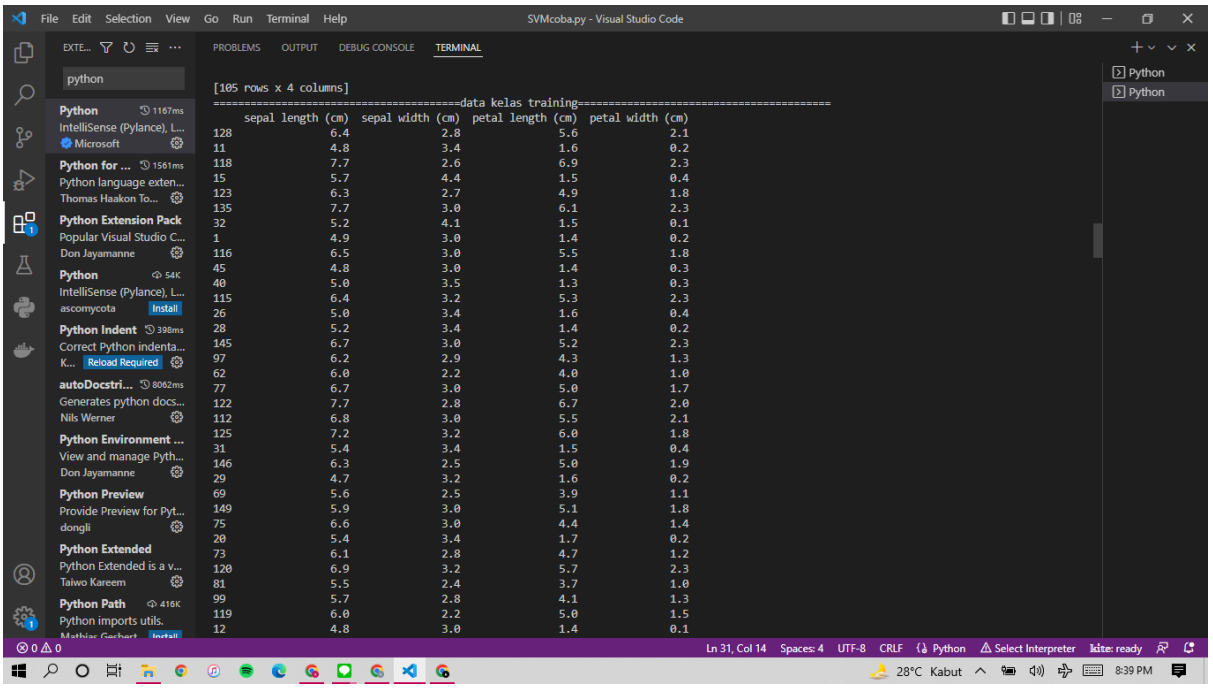


B. Separate Variables (X) and class instance (Y)

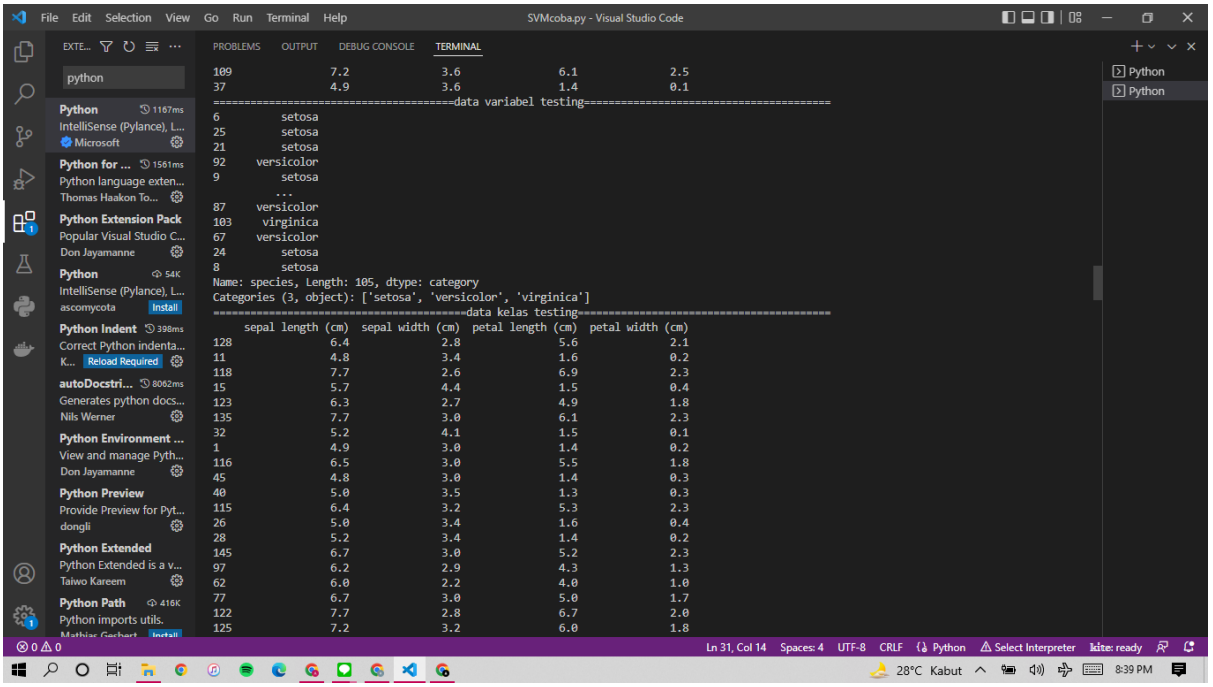


C. Split data, 70% training and 30% testing



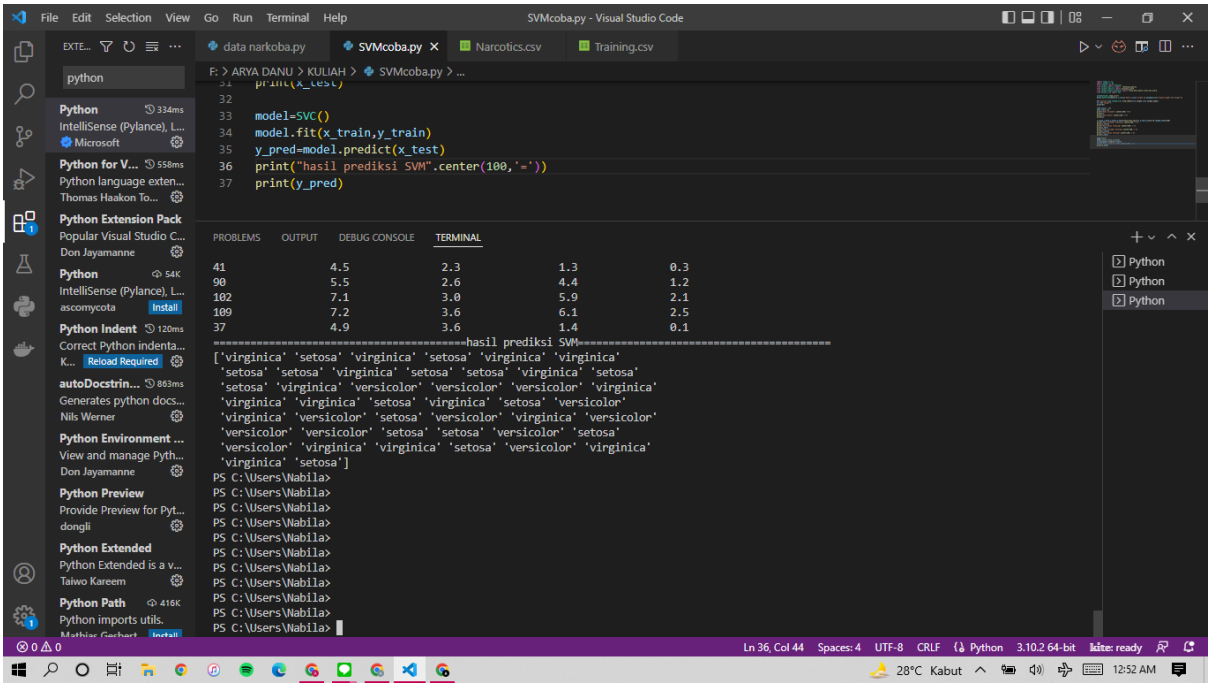


Training data

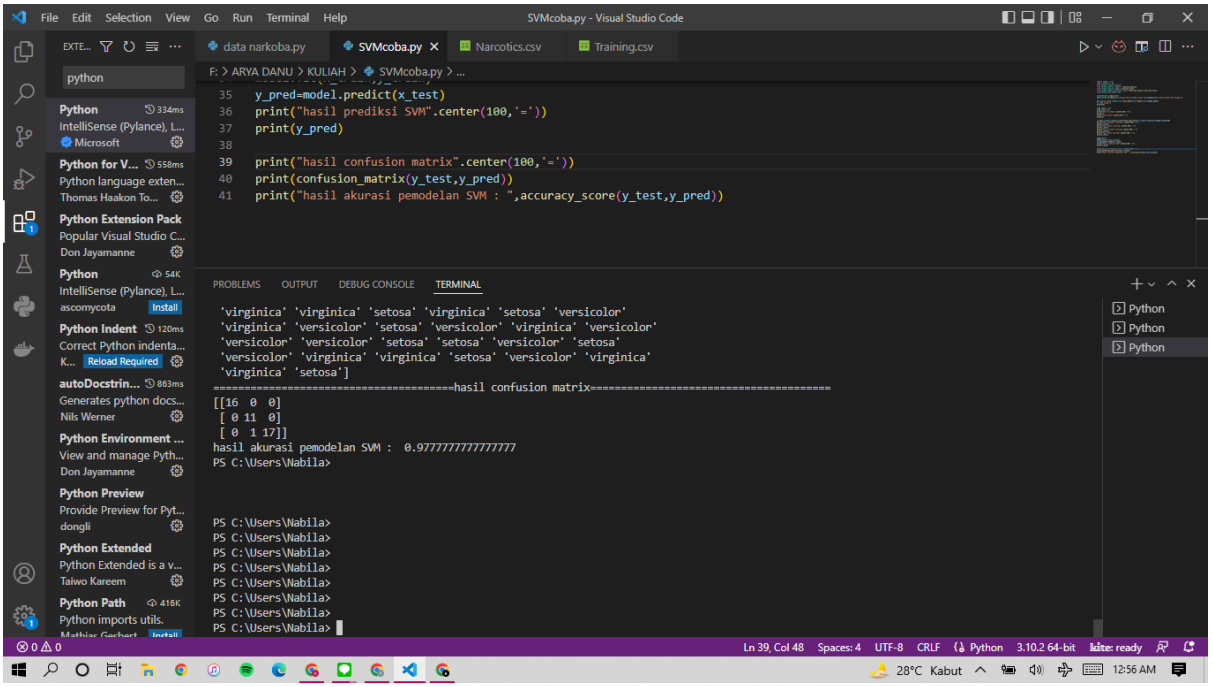


Testing Data

D. SVM Modelling



E. Calculate accuracy matrix


File Edit Selection View Go Run Terminal Help
SVMcoba.py - Visual Studio Code
data narkoba.py
SVMcoba.py
Narcotics.csv
Training.csv
F: > ARYA DANU > KULIAH > SVMcoba.py > ...
35 y_pred=model.predict(x_test)
36 print("hasil prediksi SVM".center(100,'='))
37 print(y_pred)
38
39 print("hasil confusion matrix".center(100,'='))
40 print(confusion_matrix(y_test,y_pred))
41 print("hasil akurasi pemodelan SVM : ",accuracy_score(y_test,y_pred))
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
'virginica' 'virginica' 'setosa' 'virginica' 'setosa' 'versicolor'
'virginica' 'versicolor' 'setosa' 'versicolor' 'virginica' 'versicolor'
'versicolor' 'versicolor' 'setosa' 'setosa' 'versicolor' 'setosa'
'versicolor' 'virginica' 'virginica' 'setosa' 'versicolor' 'virginica'
'virginica' 'setosa']
==========hasil confusion matrix==========
[[16 0 0]
[0 11 0]
[0 1 17]]
hasil akurasi pemodelan SVM : 0.9777777777777777
PS C:\Users\Nabila>
Ln 39, Col 48 Spaces: 4 UTF-8 CRLF Python 3.10.2 64-bit
28°C Kabut
12:56 AM

COBA CODE DENGAN DATA SENDIRI

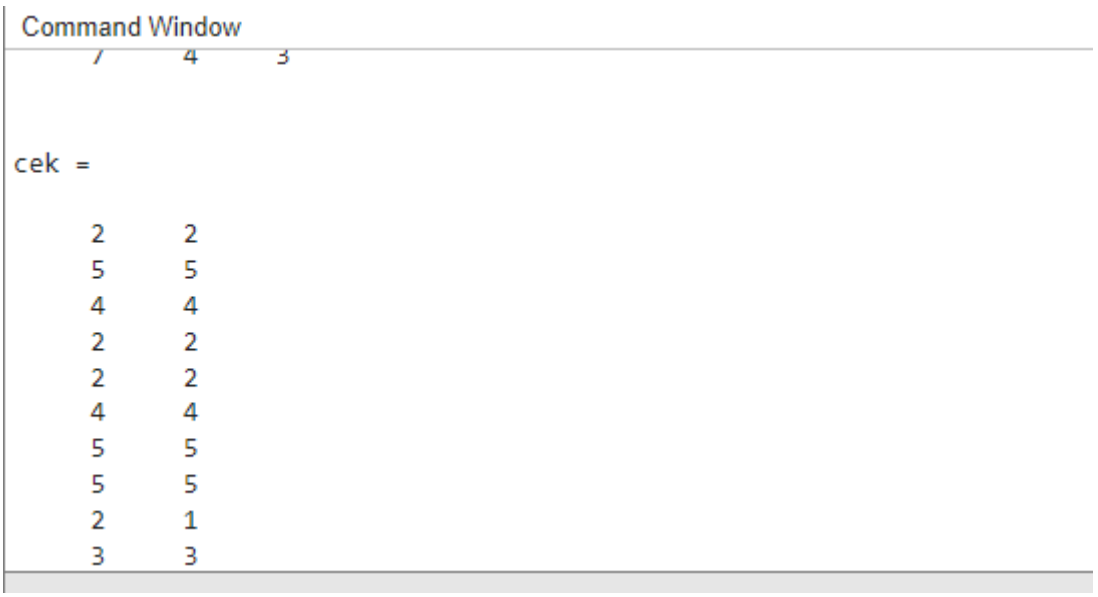
MATLAB

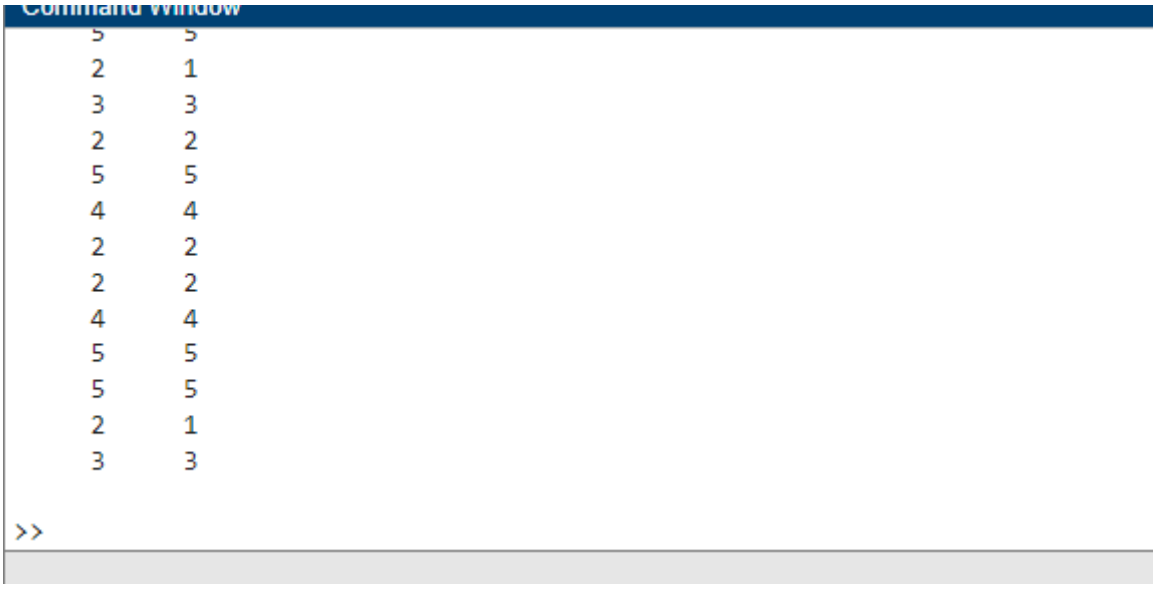
	A	B	C
	RATING	EFFECTIV...	SIDEEFFE...
	Number	Number	Number
1	RATING	EFFECTIV...	SIDE EFFE...
2	1	5	2
3	10	5	5
4	3	2	4
5	2	2	2
6	1	1	2

	A	B	C
	RATING	EFFECTIV...	SIDEEFFE...
	Number	Number	Number
996	1	1	2
997	9	5	4
998	10	4	5
999	10	5	5
1000	1	1	1
1001	7	4	3

tugasLandPrice.m | SVMcoba.m | SVMdata.m

```
1   clc;clear
2   data = xlsread('Narcotics.xlsx')
3   datatraining = data(1:floor(0.7*length(data)),1:2);
4   kelastraining = data(1:floor(0.7*length(data)),3);
5   datatesting = data(floor(0.7*length(data))+1:end,1:2);
6   kelastesting = data(floor(0.7*length(data))+1:end,3);
7   a = templateSVM('Standardize',1,'KernelFunction','polynomial');
8   traini = fitcecoc(datatraining,kelastraining,'Learners',a);
9   hasil = predict(traini,datatesting);
10  cek = [hasil kelastesting]
11
```





The dataset used this time is Narcotics.xlsx. This dataset consists of 1001 instances and the variables are divided into 2 data variables and 1 class variable. The comparison between training data and testing data is 7:3. After the SVM classifier has been carried out, a comparison/checking is carried out using 20 instances between the predicted data and the actual data. From these results found 2 differences in data. So the accuracy of the 20 data is 90%.

PYTHON

A. Import Library and Load dataset

```
1  import pandas as pd
2  from sklearn.metrics import confusion_matrix
3  from sklearn.metrics import accuracy_score
4  from sklearn.model_selection import train_test_split
5  from sklearn.svm import SVC
6
7  data=pd.read_csv('Social_Network_Ads.csv')
8  df=pd.DataFrame(data)
9  print(df)
```

There are 2 types of libraries used, namely pandas and sklearn. The dataset used is Social Network Ads which is the result of a survey from several people, whether these people will buy items that are sold based on several attributes.

```
      User ID  Gender   Age  EstimatedSalary  Purchased
0    15624510       0  19.0            19000          0
1    15810944       0  35.0            20000          0
2    15668575       1  26.0            43000          0
3    15603246       1  27.0            57000          0
4    15804002       0  19.0            76000          0
..        ...     ...   ...              ...        ...
395  15691863       1  46.0            41000          1
396  15706071       0  51.0            23000          1
397  15654296       1  50.0            20000          1
398  15755018       0  36.0            33000          0
399  15594041       1  49.0            36000          1

[400 rows x 5 columns]
```

B. Separate Variables (X) and class instance (Y)

```
11  x=df.iloc[:,1:3]
12  y=df.iloc[:,4]
```

Attribute variables used are gender, age, and salary. While the target variable is a categorical attribute between true and false.

C. Split data, 70% training and 30% testing

```
14  x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=100)
```

The dataset is divided where the training data gets a share of 70% and the testing data gets a share of 30%.

D. SVM Modelling

```
16      model=SVC()
17      model.fit(x_train, y_train)
18      y_pred=model.predict(x_test)
19      print("HASIL PREDIKSI SVM")
20      print(y_pred)
```

```
HASIL PREDIKSI SVM
[0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 1 1 1 1 1 0
 0 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 1 0 1 1 0 1 0 1 0 0 0 0 1 0 0 0 0 1 1 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 1
 0 0 0 1 0 0 1 0 1]
```

Furthermore, predictions are made using SVM modeling and the results formed are in the form of a 2-dimensional array.

E. Calculate confusion matrix and accuracy score

```
21      print("HASIL CONFUSION MATRIX")
22      print(confusion_matrix(y_test, y_pred))
23      print("HASIL AKURASI PEMODELAN SVM:", accuracy_score(y_test, y_pred))
```

```
HASIL CONFUSION MATRIX
[[72  3]
 [14 31]]
HASIL AKURASI PEMODELAN SVM: 0.8583333333333333
```

The final results obtained are in the form of a confusion matrix and an accuracy score of: (72+31)/(72+31+14+3) = **0.8583**.