



SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
ASPproved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

23AMB201 - MACHINE LEARNING

II YEAR IV SEM

UNIT II – SUPERVISED LEARNING ALGORITHMS

TOPIC 8 – Implementation of SVM

Redesigning Common Mind & Business Towards Excellence



Build an Entrepreneurial Mindset Through Our Design Thinking FrameWork

SVM with Python

```
import numpy as nm

import matplotlib.pyplot as mtp

import pandas as pd

data_set= pd.read_csv('user_data.csv')

x= data_set.iloc[:, [2,3]].values

y= data_set.iloc[:, 4].values

from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test= train_test_split(x, y,

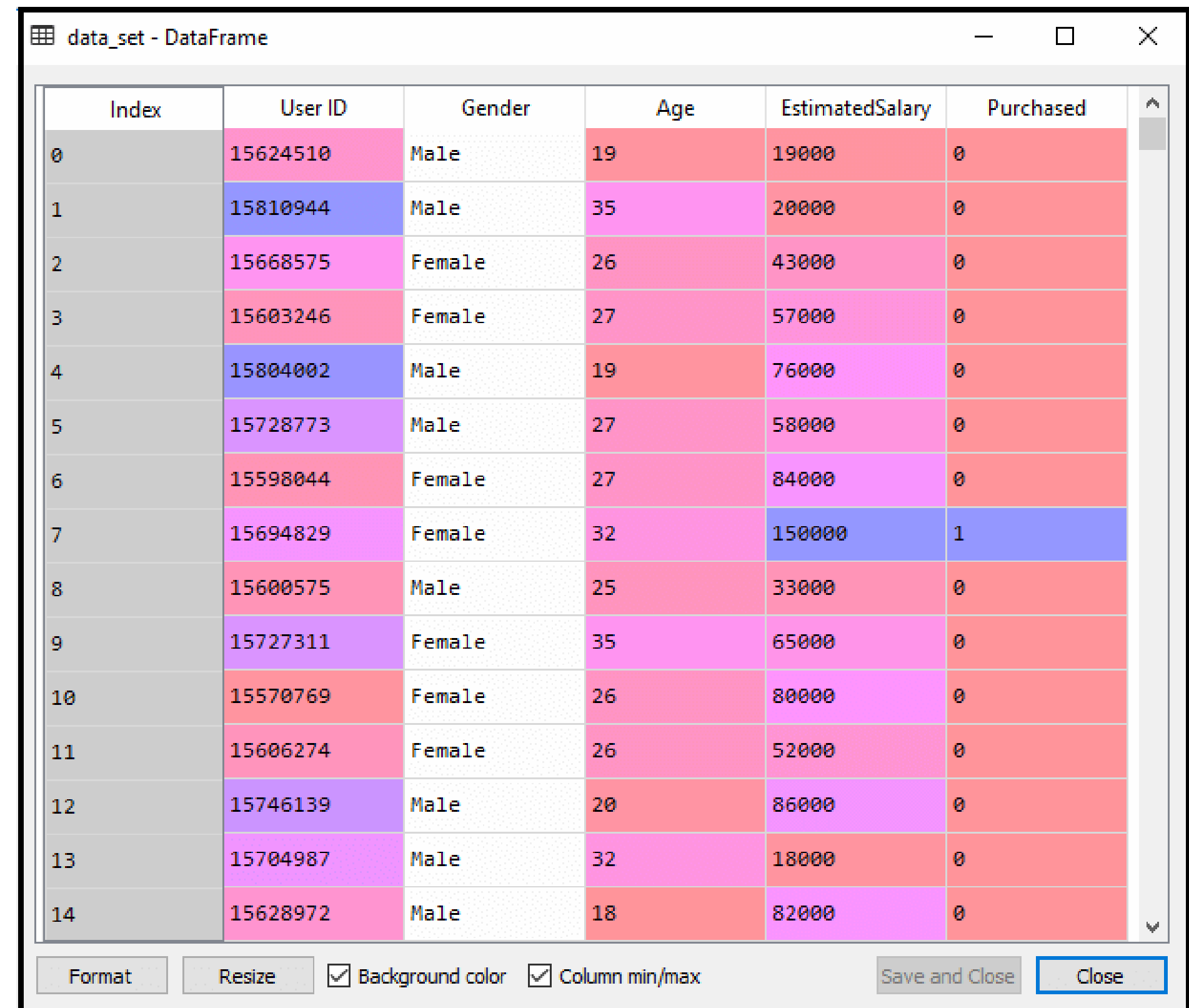
                                                    test_size= 0.25, random_state=0)

from sklearn.preprocessing import StandardScaler

st_x= StandardScaler()

x_train= st_x.fit_transform(x_train)

x_test= st_x.transform(x_test)
```



Index	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
5	15728773	Male	27	58000	0
6	15598044	Female	27	84000	0
7	15694829	Female	32	150000	1
8	15600575	Male	25	33000	0
9	15727311	Female	35	65000	0
10	15570769	Female	26	80000	0
11	15606274	Female	26	52000	0
12	15746139	Male	20	86000	0
13	15704987	Male	32	18000	0
14	15628972	Male	18	82000	0

SVM with Python

```
from sklearn.svm import SVC
```

```
classifier = SVC(kernel='linear', random_state=0)
```

```
classifier.fit(x_train, y_train)
```

```
#Predicting the test set result
```

```
y_pred= classifier.predict(x_test)
```

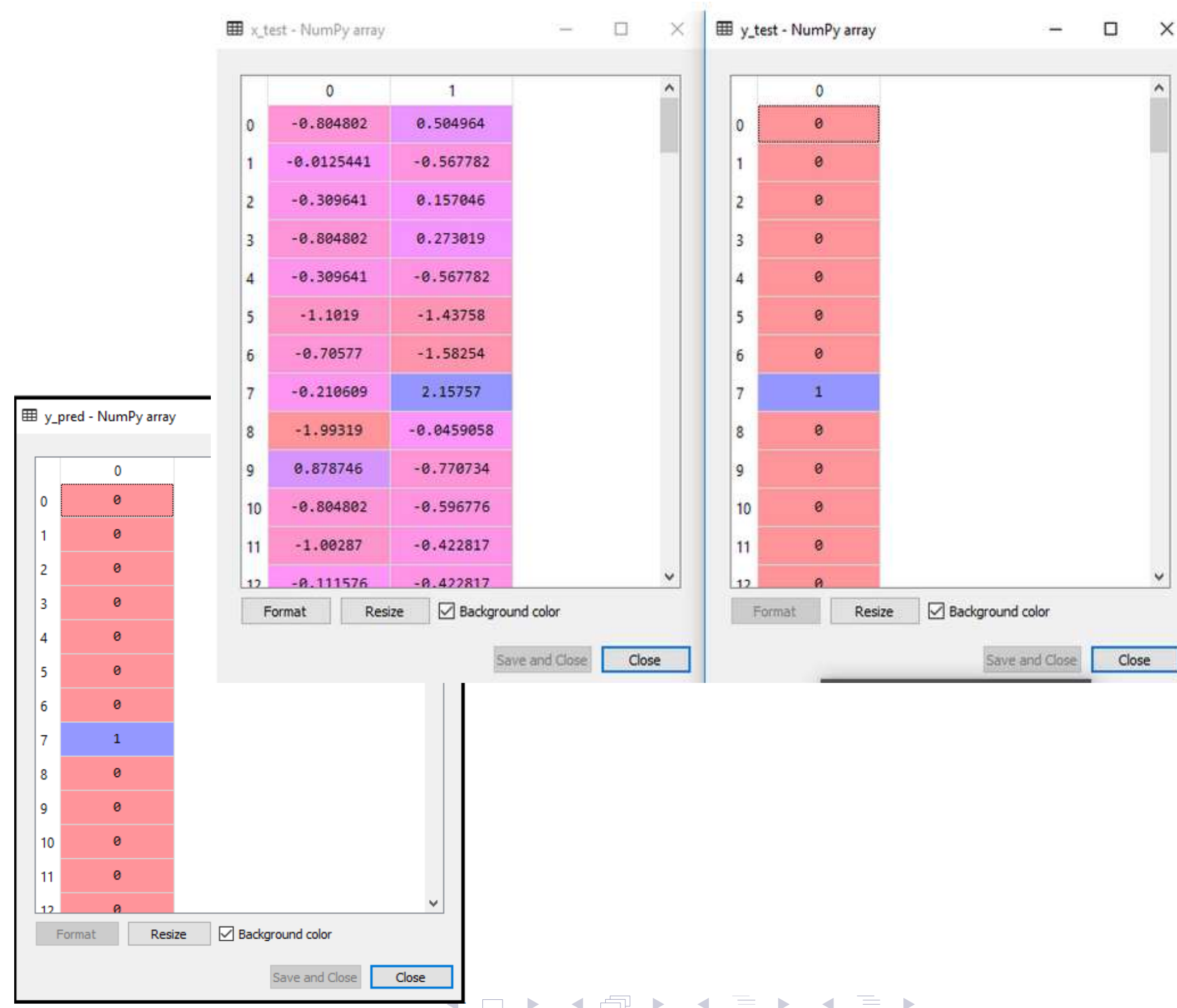
```
#Creating the Confusion matrix
```

```
from sklearn.metrics import confusion_matrix
```

```
cm= confusion_matrix(y_test, y_pred)
```



	0	1
0	66	2
1	8	24



	0	1
0	-0.804802	0.504964
1	-0.0125441	-0.567782
2	-0.309641	0.157046
3	-0.804802	0.273019
4	-0.309641	-0.567782
5	-1.1019	-1.43758
6	-0.70577	-1.58254
7	-0.210609	2.15757
8	-1.99319	-0.0459058
9	0.878746	-0.770734
10	-0.804802	-0.596776
11	-1.00287	-0.422817
12	-0.111576	-0.422817

	0
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	0
9	0
10	0
11	0
12	0

	0
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	0
9	0
10	0
11	0
12	0

SVM with Python

Visualizing the training set result:

```
from matplotlib.colors import ListedColormap

x_set, y_set = x_train, y_train

x1, x2 = nm.meshgrid(nm.arange(start = x_set[:, 0].min() - 1, stop = x_set[:, 0].max() + 1, step = 0.01),
nm.arange(start = x_set[:, 1].min() - 1, stop = x_set[:, 1].max() + 1, step = 0.01))

mtp.contourf(x1, x2, classifier.predict(nm.array([x1.ravel(), x2.ravel()]).T).reshape(x1.shape),
alpha = 0.75, cmap = ListedColormap(('red', 'green'))) mtp.xlim(x1.min(), x1.max())

mtp.ylim(x2.min(), x2.max()) for i, j in enumerate(nm.unique(y_set)):

mtp.scatter(x_set[y_set == j, 0], x_set[y_set == j, 1], c = ListedColormap(('red', 'green'))(i), label = j)

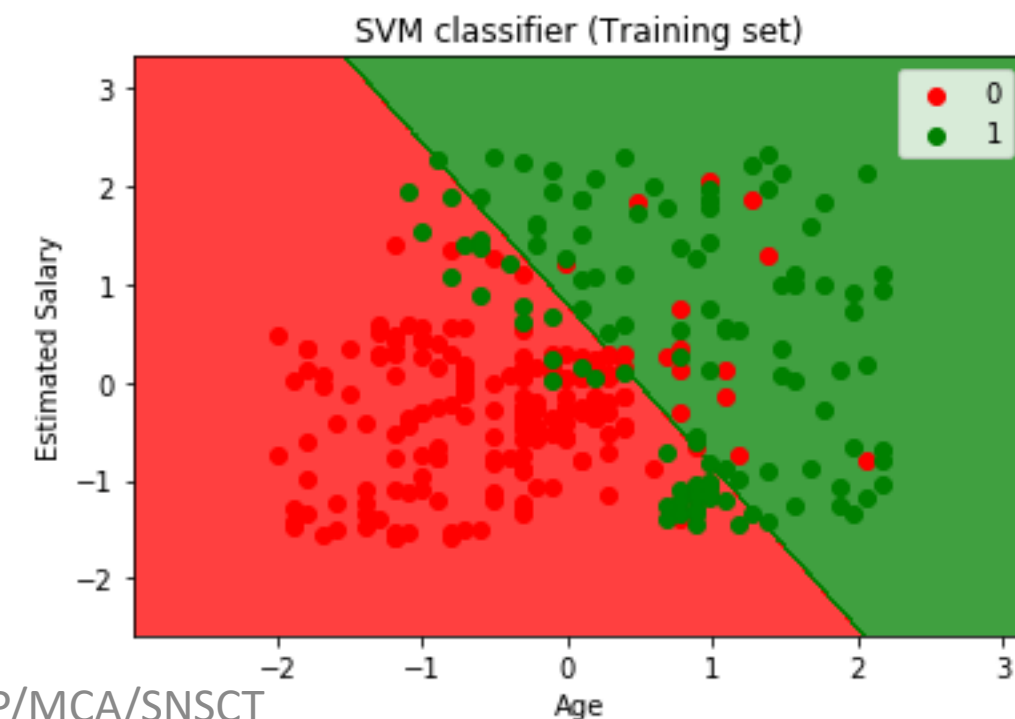
mtp.title('SVM classifier (Training set)')

mtp.xlabel('Age')

mtp.ylabel('Estimated Salary')

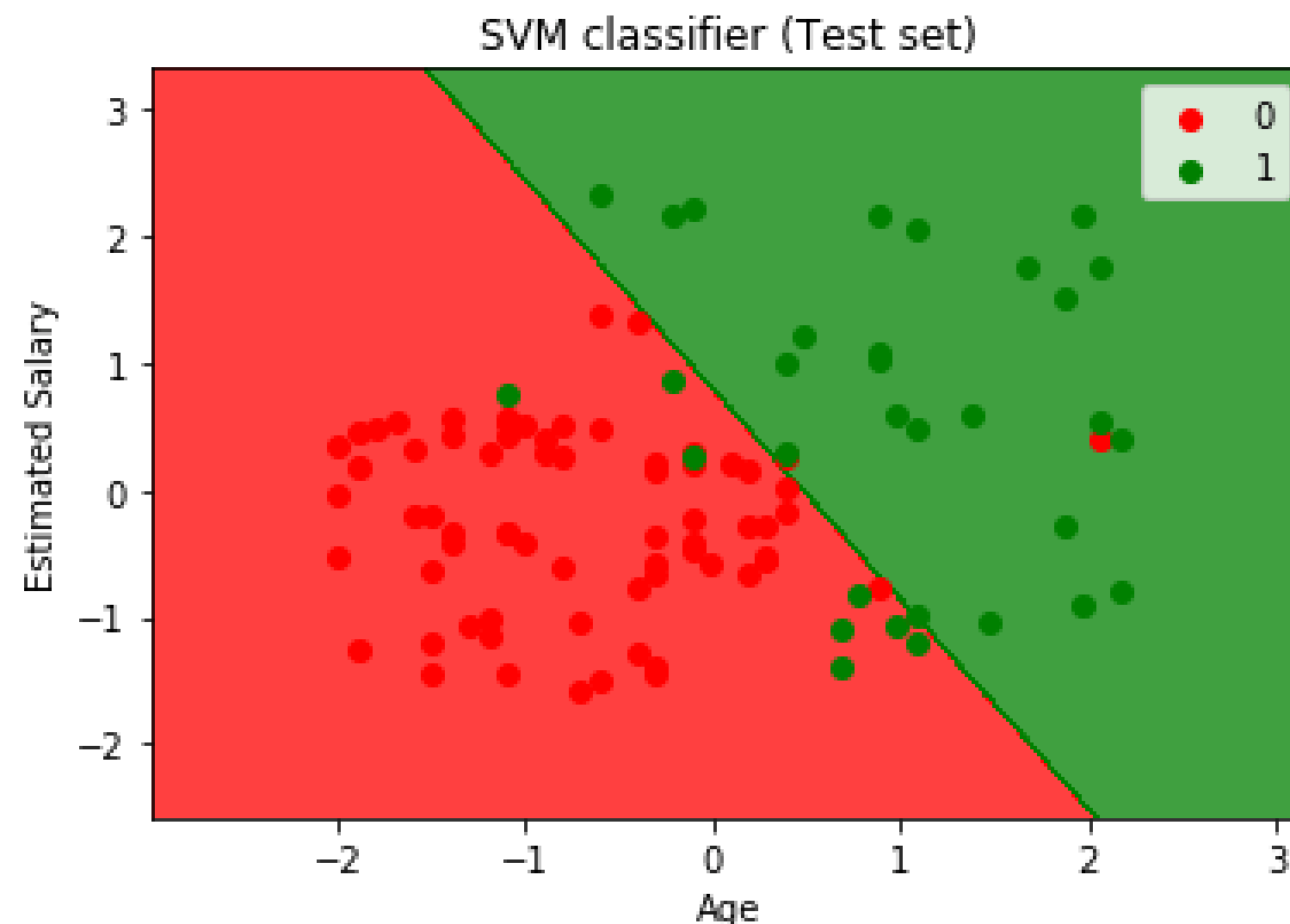
mtp.legend()

mtp.show()
```



SVM with Python

```
#Visualaizing the test set result
from matplotlib.colors import ListedColormap
x_set, y_set = x_test, y_test
x1, x2 = nm.meshgrid(nm.arange(start = x_set[:, 0].min() - 1, stop = x_set[:, 0].max() + 1, step = 0.01),
nm.arange(start = x_set[:, 1].min() - 1, stop = x_set[:, 1].max() + 1, step = 0.01))
mtp.contourf(x1, x2, classifier.predict(nm.array([x1.ravel(), x2.ravel()]).T).reshape(x1.shape),
alpha = 0.75, cmap = ListedColormap(('red', 'green' )))
mtp.xlim(x1.min(), x1.max())
mtp.ylim(x2.min(), x2.max())
for i, j in enumerate(nm.unique(y_set)):
    mtp.scatter(x_set[y_set == j, 0], x_set[y_set == j, 1],
        c = ListedColormap(('red', 'green'))(i), label = j)
mtp.title('SVM classifier (Test set)')
mtp.xlabel('Age')
mtp.ylabel('Estimated Salary')
mtp.legend()
mtp.show()
```





References

1. Tom M. Mitchell, "Machine Learning", McGraw-Hill Education (India) Private Limited, 2013.
2. Sebastian Raschka , Yuxi (Hayden) Liu Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python Packt Publishing Limited (23 December 2022).

