Support Vector Machine

In [2]:
```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

In [3]:
```python
dataset = pd.read_csv(r"C:\Users\JANHAVI\Desktop\Social_Network_Ads.csv")
X = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, -1].values
```

In [4]:
```python
# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_
```

In [5]:
```python
# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

In [6]:
```python
# Training the SVM model on the Training set
from sklearn.svm import SVC
classifier = SVC()
classifier.fit(X_train, y_train)
```

Out[6]:
```
▼  SVC  ⓘ ❓

▶ Parameters
```

In [16]:
```python
# Predicting the Test set results
y_pred = classifier.predict(X_test)
```

In [17]:
```python
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)
```

```
[[55  3]
 [ 1 21]]
```

In [18]:
```python
# This is to get the Models Accuracy
from sklearn.metrics import accuracy_score
ac = accuracy_score(y_test, y_pred)
print(ac)
```

```
0.95
```

In [19]:
```python
bias = classifier.score(X_train,y_train)
print(bias)
```

```
0.903125
```

In [20]:
```python
variance = classifier.score(X_test,y_test)
print(variance)
```

```
0.95
```

In [21]:
```python
# This is to get the Classification Report
from sklearn.metrics import classification_report
cr = classification_report(y_test, y_pred)
cr
```

Out[21]:
```
'              precision    recall  f1-score   support\n\n           0       0.98      0.95      0.96        58\n           1       0.88      0.95      0.91        22\n\n    accuracy                           0.95        80\n   macro avg       0.93      0.95      0.94        80\nweighted avg       0.95      0.95      0.95        80\n'
```

In [22]:
```python
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv(r"C:\Users\JANHAVI\Desktop\Social_Network_Ads.csv")
X = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, -1].values

# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_

# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# Training the SVM model on the Training set
from sklearn.svm import SVC
classifier = SVC()
classifier.fit(X_train, y_train)


# Predicting the Test set results
y_pred = classifier.predict(X_test)

# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)

# This is to get the Models Accuracy
from sklearn.metrics import accuracy_score
ac = accuracy_score(y_test, y_pred)
print(ac)

bias = classifier.score(X_train,y_train)
print(bias)

variance = classifier.score(X_test,y_test)
print(variance)

# This is to get the Classification Report
from sklearn.metrics import classification_report
cr = classification_report(y_test, y_pred)
cr
```

```
[[55  3]
 [ 1 21]]
0.95
0.903125
0.95
```

Out[22]:
```
'              precision    recall  f1-score   support\n\n           0       0.98
0.95      0.96        58\n           1       0.88      0.95      0.91        22\n
\n    accuracy                           0.95        80\n   macro avg       0.93
0.95      0.94        80\nweighted avg       0.95      0.95      0.95        80\n'
```

In [ ]: