In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [ ]:
```
df = pd.read_csv('iris.csv')
```

In [ ]:
```
x=df[['sepal_length', 'sepal_width','petal_length','petal_width' ]]
y=df['species']
```

In [ ]:
```
# split X and y into training and testing sets
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)
```

In [ ]:
```
# import the class
from sklearn.linear_model import LogisticRegression

# instantiate the model (using the default parameters)
model = LogisticRegression()

# fit the model with data
model.fit(x_train,y_train)

#
y_pred=model.predict(x_test)
```

```
C:\Users\Haier\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\l
inear_model\_logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status
=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```

In [ ]:
```
# import the metrics class
from sklearn import metrics
cnf_matrix = metrics.confusion_matrix(y_test, y_pred)
cnf_matrix
```

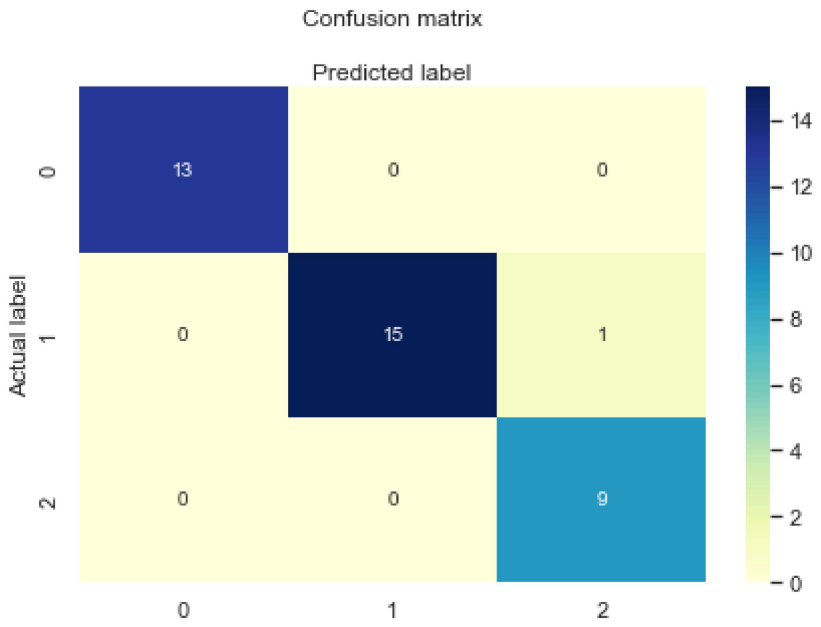
Out[ ]:
```
array([[13,  0,  0],
       [ 0, 15,  1],
       [ 0,  0,  9]], dtype=int64)
```

In [ ]:
```
# import required modules
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [ ]:
```
class_names=[0,1] # name  of classes
```

```
fig, ax = plt.subplots()
tick_marks = np.arange(len(class_names))
plt.xticks(tick_marks, class_names)
plt.yticks(tick_marks, class_names)
# create heatmap
sns.heatmap(pd.DataFrame(cnf_matrix), annot=True, cmap="YlGnBu" ,fmt='g')
ax.xaxis.set_label_position("top")
plt.tight_layout()
plt.title('Confusion matrix', y=1.1)
plt.ylabel('Actual label')
plt.xlabel('Predicted label')
```

Out[ ]:
```
Text(0.5, 257.44, 'Predicted label')
```



In [ ]:
```
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.9736842105263158
```

In [ ]:
```
y_pred_proba = model.predict_proba(x_test)[::,1]
fpr, tpr, _ = metrics.accuracy_score (y_test,  y_pred_proba)
auc = metrics.accuracy_score(y_test, y_pred_proba)
plt.plot(fpr,tpr,label="data 1, auc="+str(auc))
plt.legend(loc=4)
plt.show()
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_11100/3696992468.py in <module>
      1 y_pred_proba = model.predict_proba(x_test)[::,1]
----> 2 fpr, tpr, _ = metrics.accuracy_score (y_test,  y_pred_proba)
      3 auc = metrics.accuracy_score(y_test, y_pred_proba)
      4 plt.plot(fpr,tpr,label="data 1, auc="+str(auc))
      5 plt.legend(loc=4)

~\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\metrics\_class
ification.py in accuracy_score(y_true, y_pred, normalize, sample_weight)
    209
    210     # Compute accuracy for each possible representation
--> 211     y_type, y_true, y_pred = _check_targets(y_true, y_pred)
    212     check_consistent_length(y_true, y_pred, sample_weight)
    213     if y_type.startswith("multilabel"):

~\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\metrics\_class
ification.py in _check_targets(y_true, y_pred)
     91
     92     if len(y_type) > 1:
---> 93         raise ValueError(
     94             "Classification metrics can't handle a mix of {0} and {1} targ
ets".format(
     95                 type_true, type_pred

ValueError: Classification metrics can't handle a mix of multiclass and continuous
targets
```

In [ ]:

```
```

In [ ]:

```
from sklearn.linear_model import LogisticRegression
model=LogisticRegression()
```

In [ ]:

```
model.fit(x,y)
```

```
C:\Users\Haier\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\l
inear_model\_logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status
=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```

Out[ ]:

```
LogisticRegression()
```

In [ ]:

```
model.predict[[x]]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_11100/1309943151.py in <module>
----> 1 model.predict[[x]]

TypeError: 'method' object is not subscriptable
```

In [ ]:

```
```