import pandas as pd

import seaborn as sns

%**matplotlib** inline

import numpy as npimport matplotlib.pyplot as plt

iris = pd.read\_csv('Iris\_kaggle.csv')

print(iris.shape)

print(iris.columns)

iris.head()

iris["species"].value\_counts()

iris.plot(kind = 'scatter' , x='sepal\_length' , y = 'sepal\_width');

sns.set\_style("whitegrid");  
sns.FacetGrid(iris, hue="species", size=4).map(plt.scatter, "sepal\_length", "sepal\_width").add\_legend();  
plt.show();

sns.pairplot(iris , hue="species", palette = 'Dark2')

**from** **sklearn.model\_selection** **import** train\_test\_split

x = iris.drop('species', axis=1)  
y = iris['species']  
x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y,test\_size=0.3)

from sklearn.svm import SVC

svc\_model = SVC()

svc\_model.fit(x\_train,y\_train)

[ Output--- SVC(C=1.0, cache\_size=200, class\_weight=None, coef0=0.0, decision\_function\_shape='ovr', degree=3, gamma='auto\_deprecated', kernel='rbf', max\_iter=-1, probability=False, random\_state=None, shrinking=True, tol=0.001, verbose=False)

]

predictions = svc\_model.predict(x\_test)

from sklearn.metrics import classification\_report,confusion\_matrix

print(confusion\_matrix(y\_test,predictions))

print(classification\_report(y\_test,predictions))

training\_acc = svc\_model.score(x\_train,y\_train)

validation\_acc = svc\_model.score(x\_test,y\_test)

print("training accuracy: **%2f**"%**training\_acc**)

print("test accuracy: **%2f**"%**validation\_acc**)

y\_pred = svc\_model.predict(x\_test)

print(y\_test)

print(y\_pred)