Decision tree

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from sklearn.preprocessing import LabelEncoder  
from sklearn.model\_selection import train\_test\_split  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.metrics import classification\_report,confusion\_matrix  
from sklearn.tree import plot\_tree  
df = sns.load\_dataset('iris')  
print(df.head())  
print(df.info())  
df.isnull().any()  
print(df.shape)  
sns.pairplot(data=df,hue = 'species')  
plt.savefig("pne.png")  
sns.heatmap(df.corr())  
plt.savefig("one.png")  
target = df['species']  
df1 = df.copy()  
df1 = df1.drop('species',axis =1)  
print(df1.shape)  
print(df1.head())  
X = df1  
print(target)  
le = LabelEncoder()  
target = le.fit\_transform(target)  
print(target)  
y = target  
X\_train, X\_test,y\_train,y\_test = train\_test\_split(X,y,test\_size = 0.2,random\_state = 42)  
print("training split input",X\_train.shape)  
print("testing split input" , X\_test.shape)  
dtree = DecisionTreeClassifier()  
dtree.fit(X\_train,y\_train)  
print("decision tree classifier created")  
y\_pred = dtree.predict(X\_test)  
print("classification report \n", classification\_report(y\_test,y\_pred))  
cm = confusion\_matrix(y\_test,y\_pred)  
plt.figure(figsize=(5,5))  
sns.heatmap(data=cm,linewidths=.5,annot=True,square=True,cmap='Blues')  
plt.ylabel('Actual label')  
plt.xlabel('predicted label')  
all\_sample\_title = 'accuracy score :{0}'.format(dtree.score(X\_test , y\_test))  
plt.title(all\_sample\_title,size =15)  
plt.savefig("two.png")  
plt.figure(figsize = (20,20))  
dec\_tree = plot\_tree(decision\_tree=dtree,feature\_names= df1.columns,  
 class\_names=["setosa","vercicolor","verginica"],filled = True, precision =4 ,rounded =True)  
plt.savefig("three.png")

output

