

import pandas as pd

df=pd.read\_csv("./dataset.csv")

print (df)

y=df['PLAY']

df=pd.get\_dummies(df)

df=df.drop("PLAY\_ Don't Play",axis=1)

df=df.drop("PLAY\_ Play",axis=1)

print(df,None)

print(y)

from sklearn.tree import DecisionTreeClassifier,plot\_tree

model=DecisionTreeClassifier()

model.fit(df,y)

import matplotlib.pyplot as plt

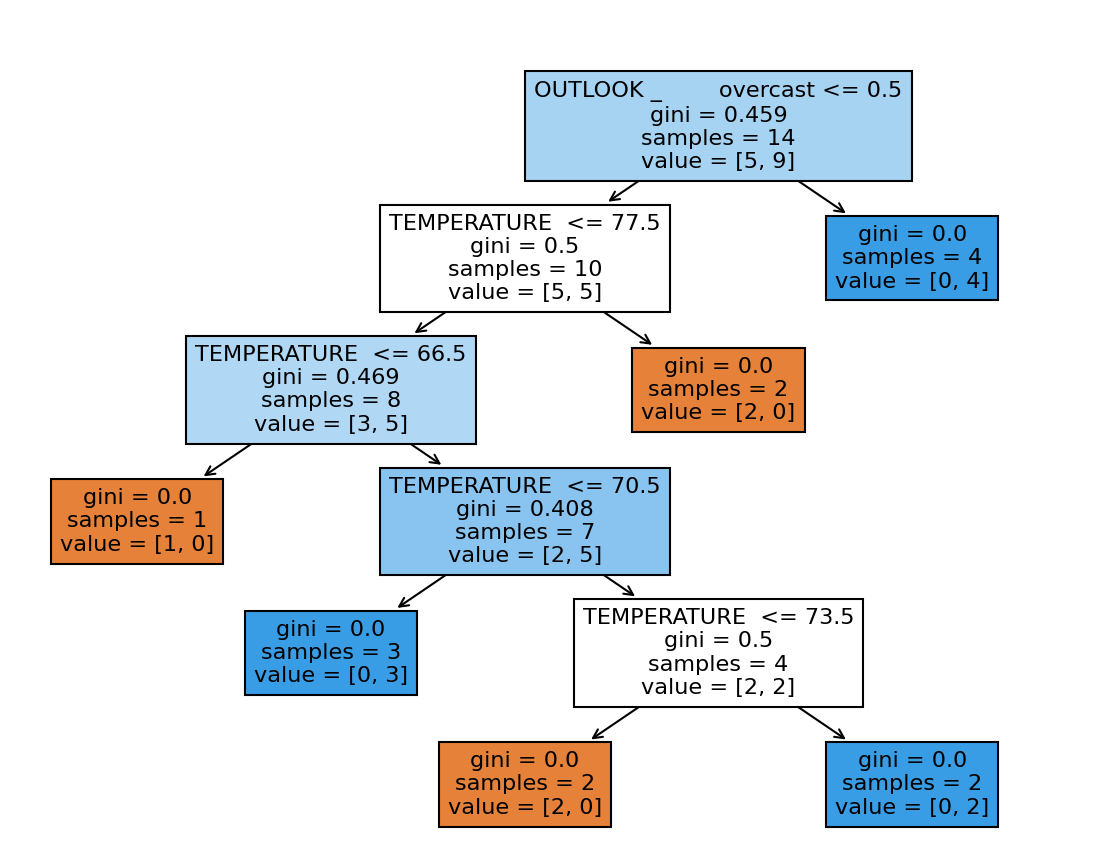
from matplotlib import figure

plt.figure(figsize=(10,8),dpi=150)

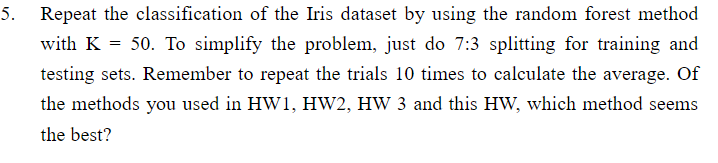
plot\_tree(model,feature\_names=df.columns,filled=True)

plt.show()

# https://data36.com/coding-a-decision-tree-in-python-classification-tree-gini-impurity/



Sunny, high temperature, low humidity, no wind -> play



Knn(0.97) is better , random forest=0.955555

from sklearn import datasets

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

from sklearn.metrics import accuracy\_score

iris = datasets.load\_iris()

#print(iris.feature\_names)

#print(iris.target\_names)

data\_x=iris.data

data\_y=iris.target

x\_train,x\_test,y\_train,y\_test=train\_test\_split(data\_x,data\_y,test\_size=0.3)

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

x\_train = sc.fit\_transform(x\_train)

x\_test = sc.transform(x\_test)

from sklearn.ensemble import RandomForestClassifier

classifier = RandomForestClassifier(n\_estimators=50)

classifier.fit(x\_train, y\_train)

#avg accuracy for 10 times

count=0

for i in range(10):

    y\_pred = classifier.predict(x\_test)

    count+=accuracy\_score(y\_test, y\_pred)

count=count/10

print ("Accuracy : ", count)