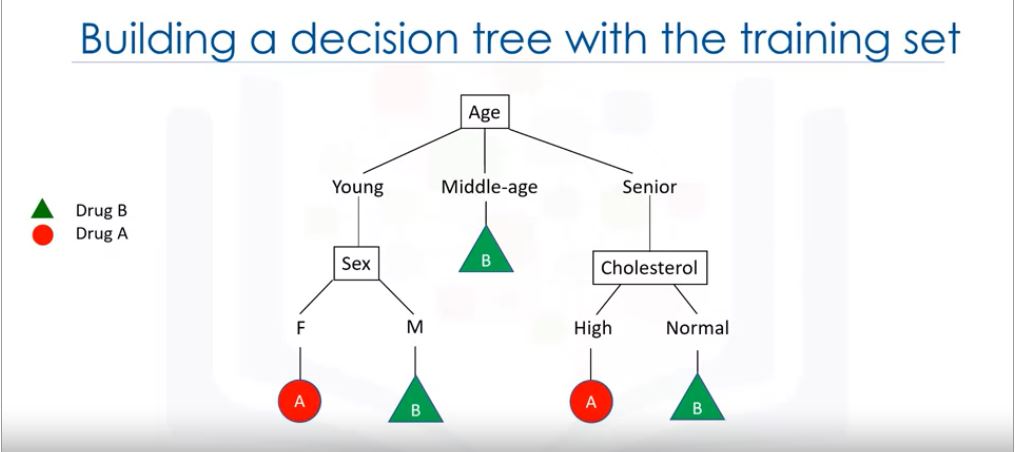
**Decision Trees:**

Suppose we take a example again to study how the decision trees are used. Let’s assume we have a data of patients with same health problems and different age ,sex , BP , cholesterol etc. Now it can be cured by wither drug A or drug B. Now you have a new patient with specific feature sets and you have to predict which drug will work the new patient. So now you will build a model of decision tree and pass the train set to it . With that you will enter the new data of the patient and predict the drug.



So in decision tree each internal node corresponds to a test

Each branch corresponds to a result of the test

Leaf assigns the class.

Procedure to make a decision tree:

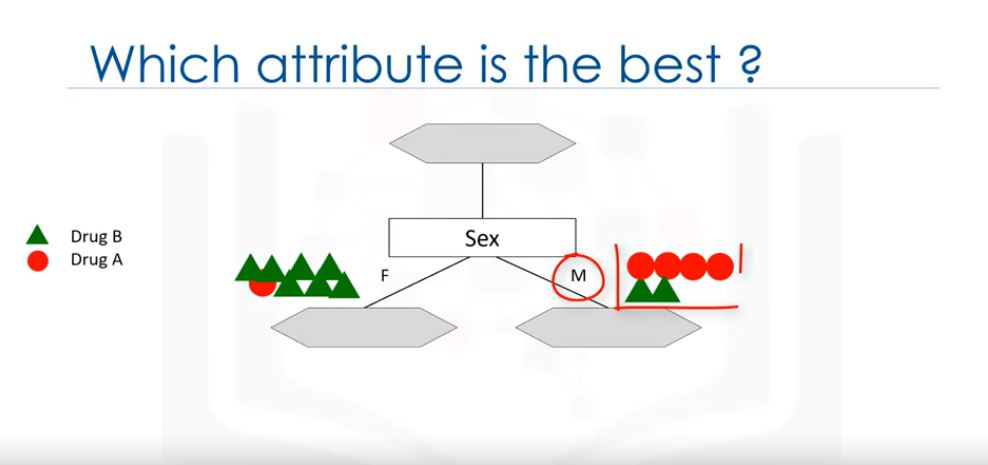
1. Choose an attribute from the data set
2. Calculate the significance of the attribute in splitting of data
3. Split data based on the value of the best attribute
4. Go to 1

**How to build a decision tree?**

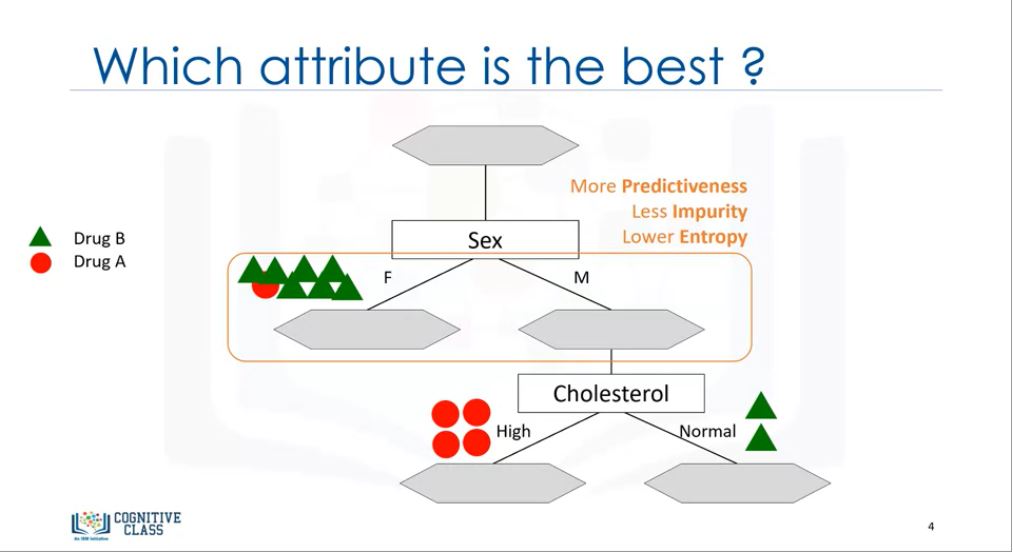
Let’s say we have the same case of example for making a decision tree.

First of all we have to find the attribute which we can use at the starting of the classification

First we pick cholesterol as the attributes. So now that we have to split the data into two branches i.e. Normal or high. But if we divide the train set and assign values , then also we cannot decide directly from the attribute cholesterol that person has to be given drug A or B.Hence cholesterol is a bad attribute.

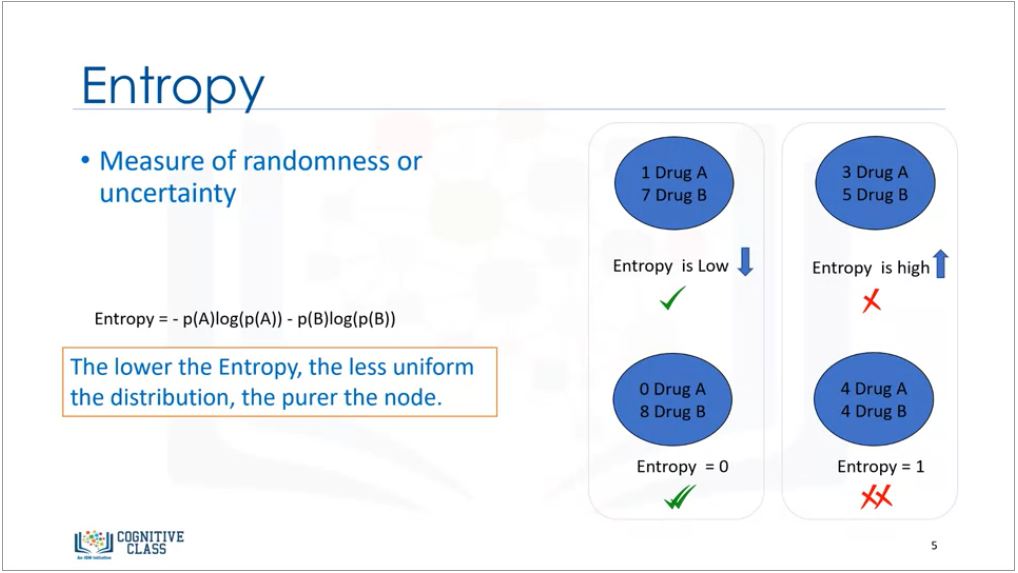
Now we will take sex attribute. We will split into two parts male and female.

Clearly we can see that if sex is female its high probability that the drug A will work .But in male we cannot say so.

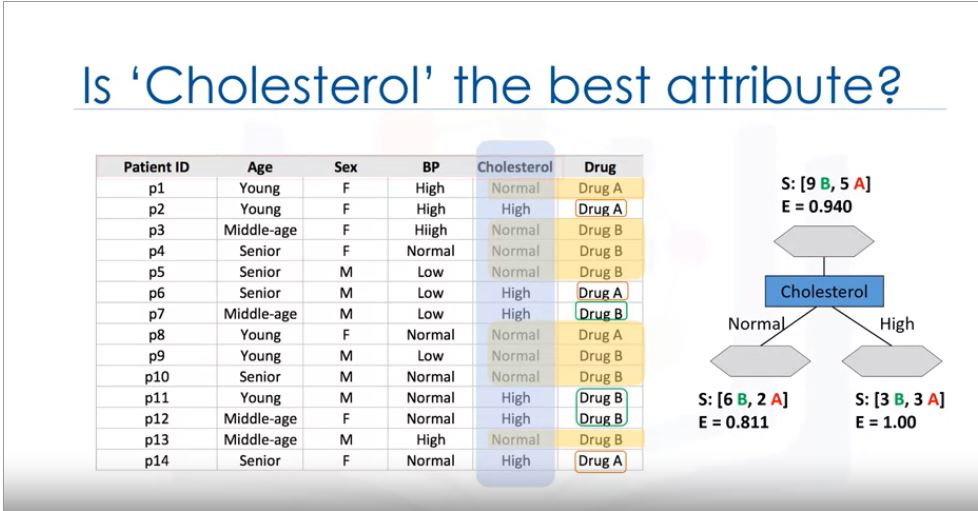
But as compared to the cholesterol attribute sex can generalised the stuff that which drug is to be selected. Hence we can say that the sex attribute is more significant as compared to the cholesterol attriute. Hence more predictiveness,less impurity of nodes. Now for the male branch we again split it to the subtree of cholesterol type i.e. high and normal

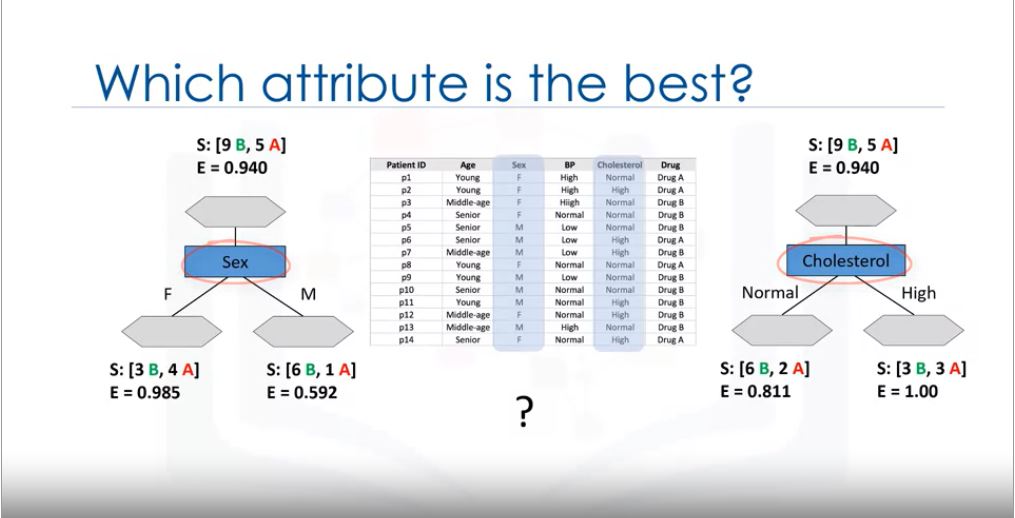
As we can see here even more pure leaves are there. Hence we can predict easily. Suppose the patient is male and has high cholesterol then we can easily suggest drug A to it with high confidence. Hence the attribute selection and its sub-tree selection is very important to achieve the goal of getting purest leaves for accurate predictions. If node is having only specific categorised values then it is said to be pure nodes. Here are high and normal cholesterol.

Entropy is the measurement of the randomness or uncertainty. Lower the entropy purer the node. I the node is pure entropy is 0 and if node has completely divided entropy is 1.



Here we can easily calculate the entropy by using the formula given above. p is for the proportion or ratio of a category such as drug A or B.

Now we will calculate the entropy after dividing it with cholesterol.

Similiarly for sex.

So which one is to be selected as the best attribute . For that we will calculate the information gain.

Information gain is the information that can increase the level of certainty after splitting.

Information gain=(Entropy before split ) - (weighted entropy after split )

Information gain has reverse relation with the entropy. As the entropy decreases i.e. randomness decreases ,the information gain increases and vice versa.

Gain(s,Sex)=0.940 - [(7/14)0.985 + (7/14)0.592]=0.151

Gain(s,cholesterol)=0.940 - [(8/14)0.811 + (6/14)1.0]=0.048

So tree with higher information gain after splitting is considered as best attribute . So here it is sex attribute.

So we will select sex as first classifier and repeat the whole process for second one.

**CODING FOR DECISION TREES:**

#First of all import all the libraries used

import numpy as np

import pandas as pd

from sklearn.tree import DecisionTreeClassifier

#Understanding the dataset

You have a set of data of the patients suffering from the same illness and there are 5 type of medications Drug A,B,C,x and y. The past data has a specific drug used to cure the illness . The feature sets are AGE , SEX , CHOLESTEROL , BLOOD PRESSURE and Drug. Now you have to train model such that if we enter age ,sex, BP and cholesterol of the new patient ,then it will predict which drug is to be used to cure the illness of the new patient.

#Download dataset

!wget -O drug200.csv <https://s3-api.us-geo.objectstorage.softlayer.net/cf-courses-data/CognitiveClass/ML0101ENv3/labs/drug200.csv>

#Read the dataset

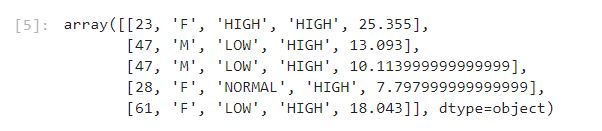
df=pd.read\_csv(“drug200.csv”,delimeter=”,”)

df[0:5]

#Preprocessing the data-X(Feature matrix),Y(Target variable)

X=df[[‘Age’,’Sex’,’BP’,’Cholesterol’,’Na\_to\_K’]].values

X[0:5]



#Now we have labels in the data- Example-In Sex we have M and F, in cholesterol LOW and HIGH etc.So we will process this data to 0,1 by using pre-processing library.

from sklearn import preprocessing

le\_sex=preprocessing.LabelEncoder()

le\_sex.fit([‘F’,’M’])

X[:,1]=le\_sex.transform(X[:,1])

le\_bp=preprocessing.LabelEncoder()

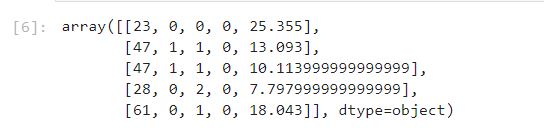
le\_bp.fit([‘LOW’,’NORMAL’,’HIGH’])

X[:,2]=te\_bp.transform(X[:,2})

le\_chol=preprocessing.LabelEncoder()

le\_chol.fit(‘NORMAL’,’HIGH’)

X[:,3]=le\_chol.transform(X[:,3])



#declare taget variable

Y=df[[‘drug’]].values

Y[0:5]

#We use train\_test\_split in the decision tree

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

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,test\_size=0.3,random\_state=3)

#Modelling

drugTree=DecisionTreeClassifier(criterion=’entropy’,max\_depth=4)

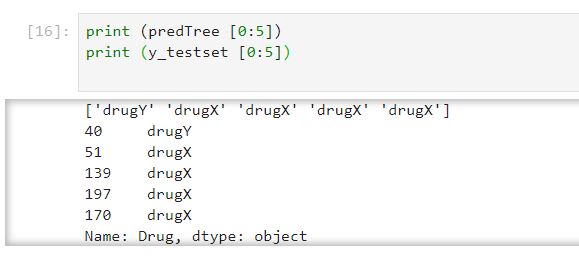
drugTree

drugTree.fit(X\_train,Y\_train)

#Prediciton

ptree=drugTree.predict(X\_test)

ptree[0:5]

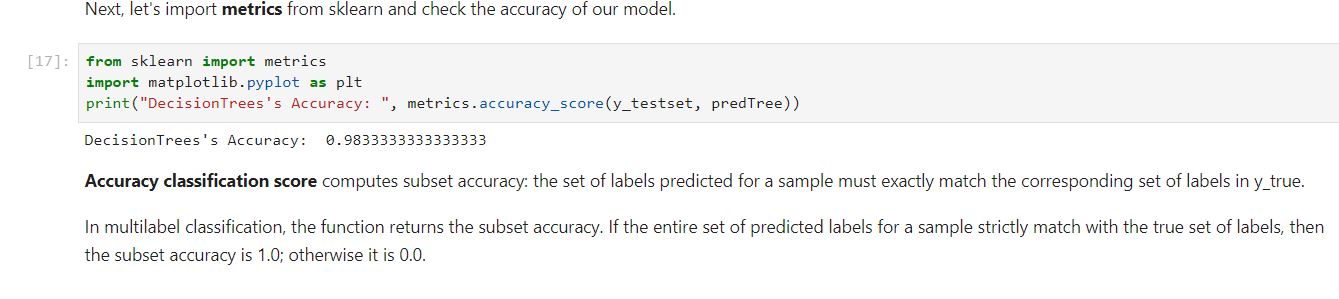


#Evaluation of the metrics

from sklearn import metrics

import malplotlib.pyplot as plt

print(‘Decision tree accuracy:’,metrics.accuracy\_score(y\_text,ptee))



#Visualizing the tree.

# Notice: You might need to uncomment and install the pydotplus and graphviz libraries if you have not installed these before

# !conda install -c conda-forge pydotplus -y

# !conda install -c conda-forge python-graphviz –y

from sklearn.externals.six import StringIO

import pydotplus

import matplotlib.image as mpimg

from sklearn import tree

%matplotlib inline

dot\_data = StringIO()

filename = "drugtree.png"

featureNames = my\_data.columns[0:5]

targetNames = my\_data["Drug"].unique().tolist()

out=tree.export\_graphviz(drugTree,feature\_names=featureNames, out\_file=dot\_data, class\_names= np.unique(y\_trainset), filled=True, special\_characters=True,rotate=False)

graph = pydotplus.graph\_from\_dot\_data(dot\_data.getvalue())

graph.write\_png(filename)

img = mpimg.imread(filename)

plt.figure(figsize=(100, 200))

plt.imshow(img,interpolation='nearest')

