https://www.tutorialspoint.com/artificial\_intelligence\_with\_python/index.htm

Decision Tree Classifier

A decision tree is a binary tree flowchart where each node splits a group of observations according to some feature variable.

Here, we are building a Decision Tree classifier

* Predict male or female
* 19 samples
* 2 features: height and length of hair

Prerequisite

For building the following classifier, we need to install **pydotplus** and **graphviz**. Basically, graphviz is a tool for drawing graphics using dot files and **pydotplus** is a module to Graphviz’s Dot language. It can be installed with the package manager or pip.

from sklearn import tree

from sklearn.datasets import load\_iris

from sklearn.metrics import classification\_report

from sklearn.tree import DecisionTreeClassifier

data\_feature\_names = ['height','length of hair']

X = [[165,19],[175,32],[136,35],[174,65],[141,28],[176,15],[131,32],

[166,6],[128,32],[179,10],[136,34],[186,2],[126,25],[176,28],[112,38],

[169,9],[171,36],[116,25],[196,25]]

Y = ['Man','Woman','Woman','Man','Woman','Man','Woman','Man','Woman',

'Man','Woman','Man','Woman','Woman','Woman','Man','Woman','Woman','Man']

#Create mode and train data

clf = tree.DecisionTreeClassifier()

clf = clf.fit(X,Y)

#Accuracy

#from sklearn.metrics import accuracy\_score

#print(accuracy\_score(TEST\_DATA, TRAIN\_DATA))

#Predict, if height 133 and hair length 47 if he is a man or woman

prediction = clf.predict([[133,37]])

print(prediction)

(Woman)

Generar grafico

import pydotplus

import collections

dot\_data = tree.export\_graphviz(clf,feature\_names = data\_feature\_names,

            out\_file = None,filled = True,rounded = True)

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

colors = ('orange', 'yellow')

edges = collections.defaultdict(list)

for edge in graph.get\_edge\_list():

    edges[edge.get\_source()].append(int(edge.get\_destination()))

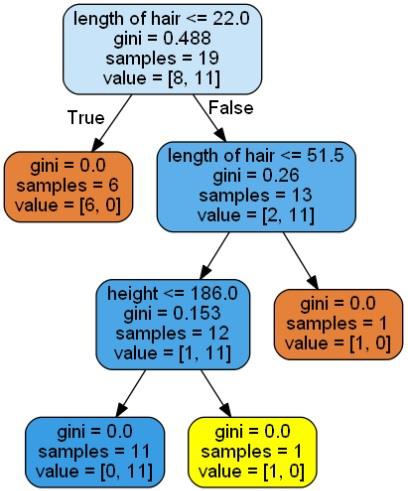
for edge in edges: edges[edge].sort()

for i in range(2):dest = graph.get\_node(str(edges[edge][i]))[0]

dest.set\_fillcolor(colors[i])

graph.write\_png('Decisiontree16.png')

Aparecera en la seccion de files



We can change the values of features in prediction to test it.