Untitled5

February 12, 2019

In [2]:
```
from sklearn.datasets import load_iris
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split

iris_data = load_iris()
X = iris_data.data
y = iris_data.target
X_train, X_test, y_train, Y_test = train_test_split(X,y,random_state=42)

model = DecisionTreeClassifier(max_leaf_nodes=3, random_state=0)
model.fit(X_train, y_train)

from sklearn.tree import _tree

def find_rules(tree, features):
  dt = tree.tree_
  def visitor(node, depth):
    indent = ' ' * depth
    if dt.feature[node] != _tree.TREE_UNDEFINED:
      print('{} if <{}> <= {}:'.format(indent, features[node], round(dt.threshold[node]
      visitor(dt.children_left[node], depth + 1)
      print('{} else:'.format(indent))
      visitor(dt.children_right[node], depth + 1)
    else:
      print('{} return {}'.format(indent, dt.value[node]))
  visitor(0,1)

find_rules(model, iris_data.feature_names)
```

```
if <sepal length (cm)> <= 0.8:
 return [[ 35.   0.   0.]]
else:
 if <petal length (cm)> <= 4.75:
  return [[  0.  34.   1.]]
 else:
  return [[  0.   5.  37.]]
```