1. Implement the Decision Tree

Step 1: calculate the entropy

計算傳入的data['Wait']有幾個是T，並算出是T的機率(p)，帶入entropy的公式，-p\*-(1-p)\*，就可以算出entropy。

Step 2: search for the best split

用(i+0.5)去切data，並計算出combination entropy，取最小的entropy，代表是最好的切法。

Step 3: build the decision tree

用dfs的方式去遞迴，直到目標深度或者是全部的case都被分類完，並記錄每一層的decision跟threshlod。

以下是我的tree:

['Patrons 1.5', ['Hungry 0.5', 'F', ['Friday 0.5', 'F', ['Price 0.5', 'F', 'T']]], 'T']

可以看到在第四層就已經完全分類完了。

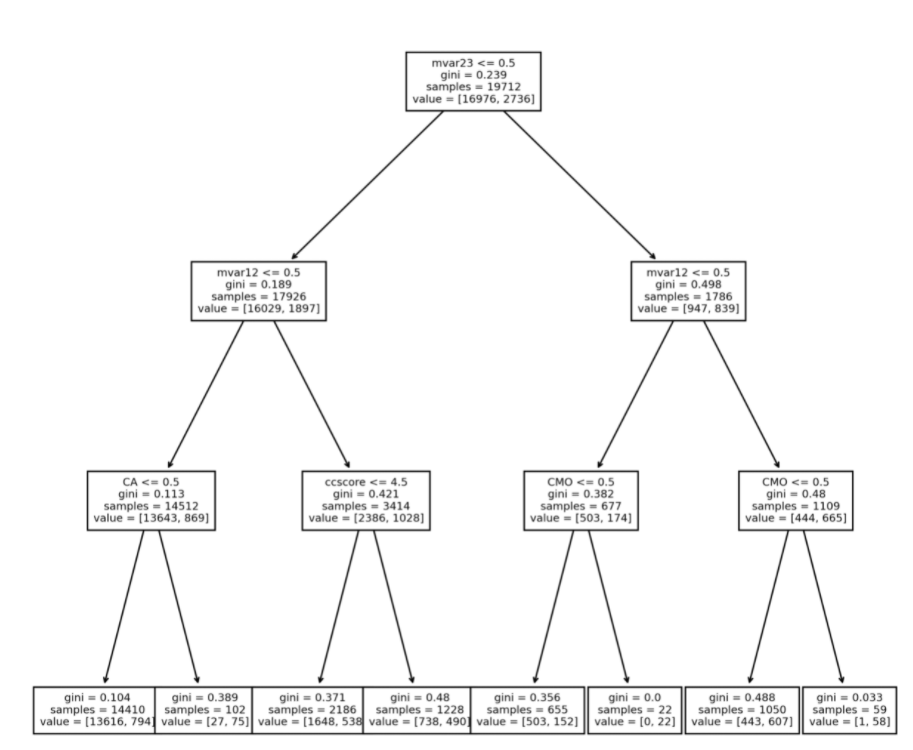
1. Classification with the MIMIC Dataset

我首先把['indextime']跟['subject\_id']從data中剃除，因為這兩個資料跟結果沒關係。

我用的方法是decision tree，我是從sklearn.tree import DecisionTreeClassifier這個model做使用，輸入x\_train跟y\_train做訓練。此外，decision tree的深度並不是越深越好，太深的話可能會出現overfit的狀況，所以我將深度設定為15。

至於評估訓練成果的score function，我是從sklearn.metrics import f1\_score，拿x\_validation輸入訓練好的model，將出來的prediction跟y\_validation輸入f1\_score，我得到的值約為0.48。

以下是我前三層features跟threshold:



1. Bonus

我從sklearn import tree，並使用以下function印出tree的前五層:

