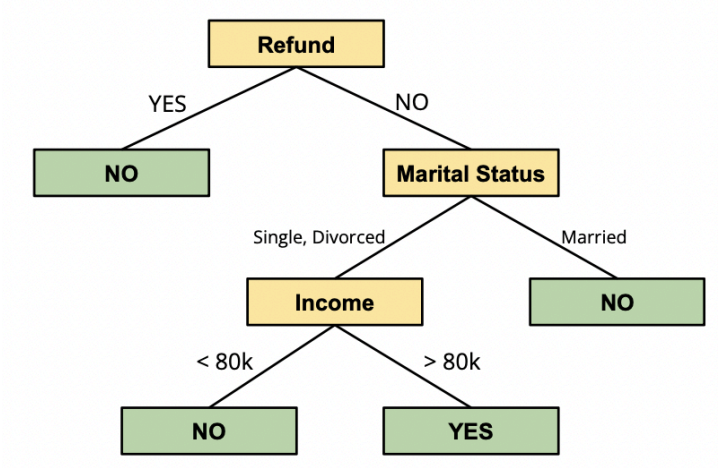
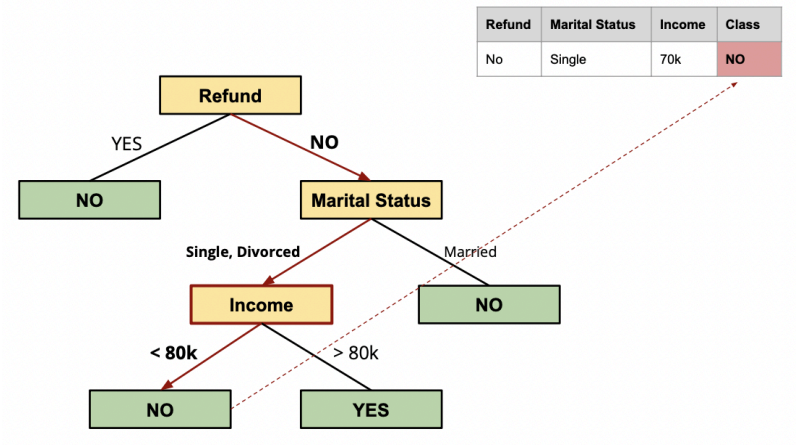
CAS CS 506
Lec 12

Decision Trees

1. What a Decision Tree Looks Like
   a. 

2. How it Works
   a. Given information, we narrow down the decision tree after starting at the root node
   b.

Refund	Marital Status	Income	Class
No	Single	70k	?

   c. 

3. Hunt's Algorithm
   a. Recursive Algorithm
      i. Repeatedly split the dataset based on attributes
   b. Base cases
      i. IF Split and all data points in the same class
         1. Great! Predict that class

ii. IF Split and no data points
    1. No Problem! Predict a reasonable default

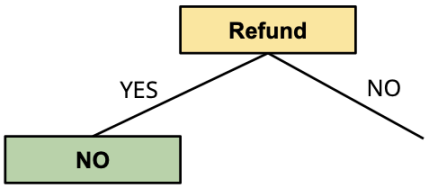
c. The recursion (IF split and data points belong to more than one class)
    i. Find the attribute (and best way to split that attribute) that best splits the data

4. Example

a.

Refund	Marital Status	Income	Class
Yes	Single	125k	No
No	Married	100k	No
No	Single	70k	No
Yes	Married	120k	No
No	Divorced	90k	Yes
No	Married	60k	No
Yes	Divorced	220k	No
No	Single	85k	Yes
No	Married	75k	No
No	Single	90k	Yes

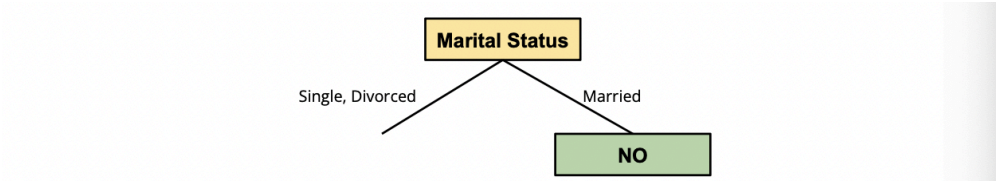
b.



Refund	Marital Status	Income	Class
Yes	Single	125k	No
Yes	Married	120k	No
Yes	Divorced	220k	No

Refund	Marital Status	Income	Class
No	Married	100k	No
No	Single	70k	No
No	Divorced	90k	Yes
No	Married	60k	No
No	Single	85k	Yes
No	Married	75k	No
No	Single	90k	Yes

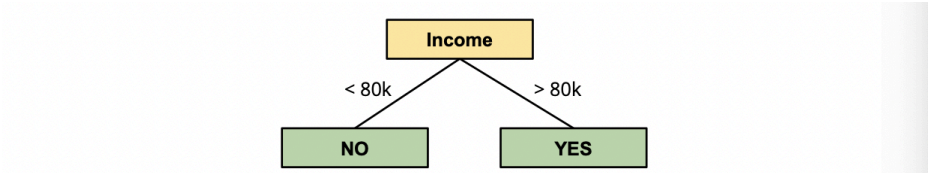
c.



Refund	Marital Status	Income	Class
No	Single	70k	No
No	Divorced	90k	Yes
No	Single	85k	Yes
No	Single	90k	Yes

Refund	Marital Status	Income	Class
No	Married	100k	No
No	Married	60k	No
No	Married	75k	No

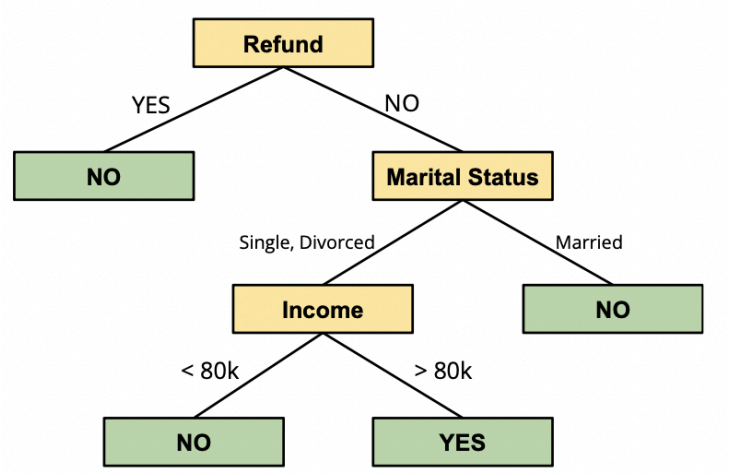
d.



Refund	Marital Status	Income	Class
No	Single	70k	No

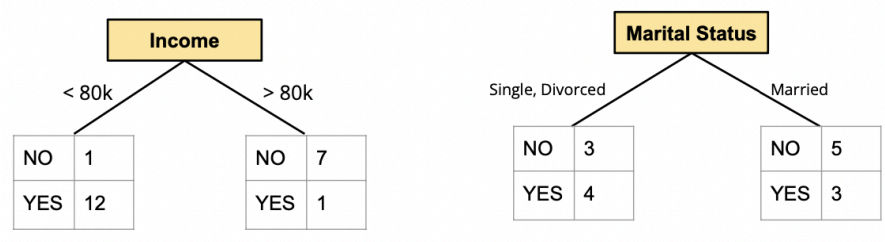
Refund	Marital Status	Income	Class
No	Divorced	90k	Yes
No	Single	85k	Yes
No	Single	90k	Yes

e.



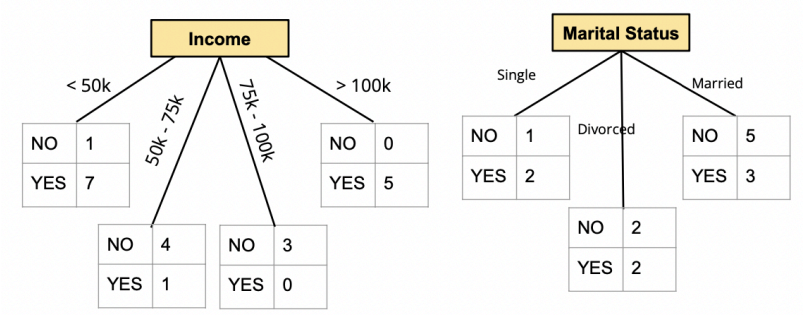
5. Many Ways to Split a Given Attribute
   a. Binary Split

i.



b. Multi-Way Split

i.



6. Continuous Variable
   a. Use binning before running the decision tree
      i. Can use clustering for that for example
   b. Compute a threshold while building the tree
      i. A > t VS A < t
7. Need a Metric
   a. That favors nodes like
      i. NO: 1
      ii. YES: 7
   b. Over nodes like
      i. No: 4
      ii. YES: 4
8. GINI Index
   a. Denote p(j|t) as the relative frequency of class j at node t

b.

NO	1
YES	7

p(NO | t) = ⅛
p(YES | t) = ⅞

NO	4
YES	3

p(NO | t) = 4/7
p(YES | t) = 3/7

c.

$$GINI(t) = 1 - \sum_j p(j|t)^2$$

NO	1
YES	7

p(NO | t) = ⅛
p(YES | t) = ⅞

GINI(t) = 1 - 1/64 - 49/64 = 14/64

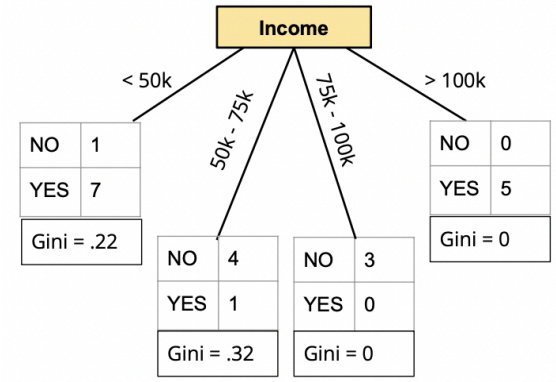
NO	4
YES	3

p(NO | t) = 4/7
p(YES | t) = 3/7

GINI(t) = 1 - 16/49 - 9/49 = 24/49

9. GINI of the Split

a.



b.

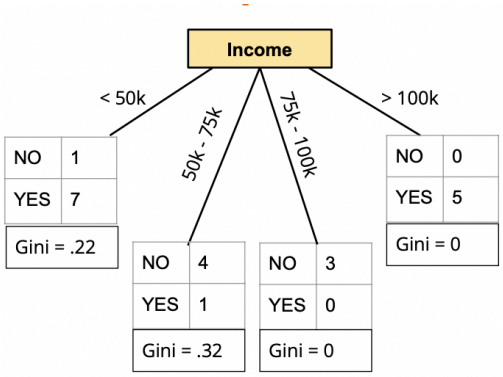
$$GINI_{split} = \sum_{t=1}^{k} \frac{n_t}{n} GINI(t)$$

where:

n_t = number of data points at node t

n = number of data points before the split (parent node)

c.



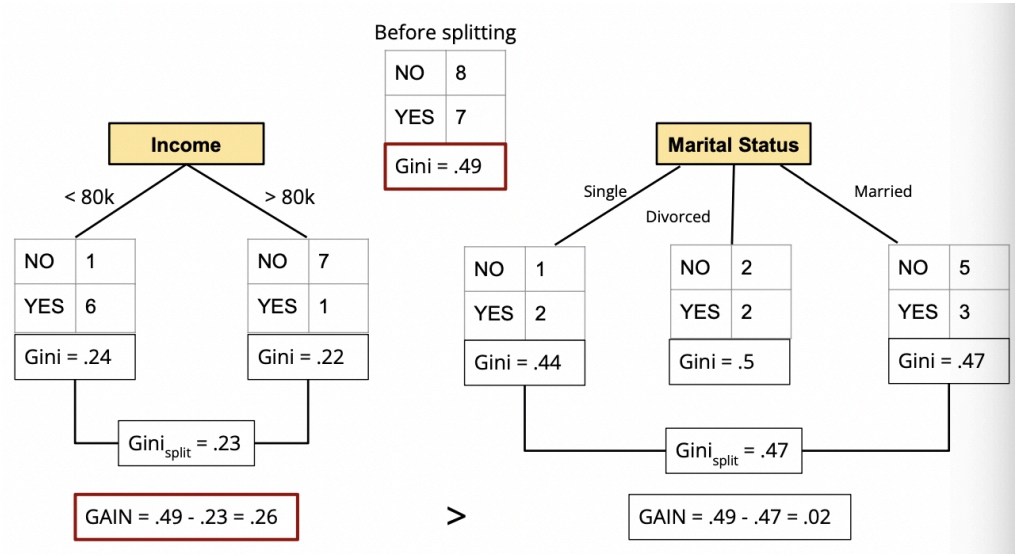
$$GINI_{split} = \sum_{t=1}^{k} \frac{n_t}{n} GINI(t)$$

n = 21

$GINI_{split}$ = .22 * 8/21
+ .32 * 5/21
+ 0 * 3/21
+ 0 * 5/21
= .16

10. Putting It All Together

a.



11. Limitations
    a. Easy to construct a tree that is too complex and overfits the data
    b. Solutions:
        i. Early termination (stop before tree is fully grown - use majority vote at leaf node)
            1. Stop at some specified depth
            2. Stop if size of node is below some threshold
            3. Stop if gini does not improve
        ii. Pruning (create fully grown tree then trim)
12. Extensions
    a. Entropy
        i. $$\text{Entropy}(t) = -\sum_j p(j|t) \log(p(j|t))$$
    b. Misclassification Error
        i. $$\text{Error}(t) = 1 - \max_j (p(j|t))$$
13.