10-701: Introduction to Machine Learning Lecture 2 – Decision Trees

Henry Chai & Zack Lipton

8/30/23

Front Matter

- Announcements:
  - Recitations will be held on Fridays, at the same time and place as lecture
    - No recitation Friday, September 1st
  - Office hours will start next week
- Recommended Readings:
  - Mitchell, Chapter 3: Decision Tree Learning
  - Daumé III, Chapter 1: Decision Trees

Recall: Our second Machine Learning Classifier

- Alright, let's actually (try to) extract a pattern from the data

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

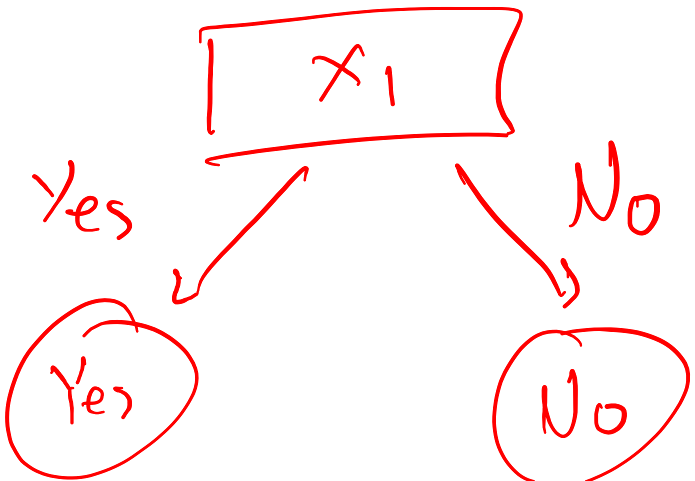
- Decision stump on x_1:

$$h(\boldsymbol{x}') = h(x_1', \ldots, x_D') = \begin{cases} \text{“Yes” if } x_1' = \text{“Yes”} \\ \text{“No” otherwise} \end{cases}$$

Recall: Our second Machine Learning Classifier

- Alright, let’s actually (try to) extract a pattern from the data

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes



Decision Stumps: Questions

1. How can we pick which feature to split on?

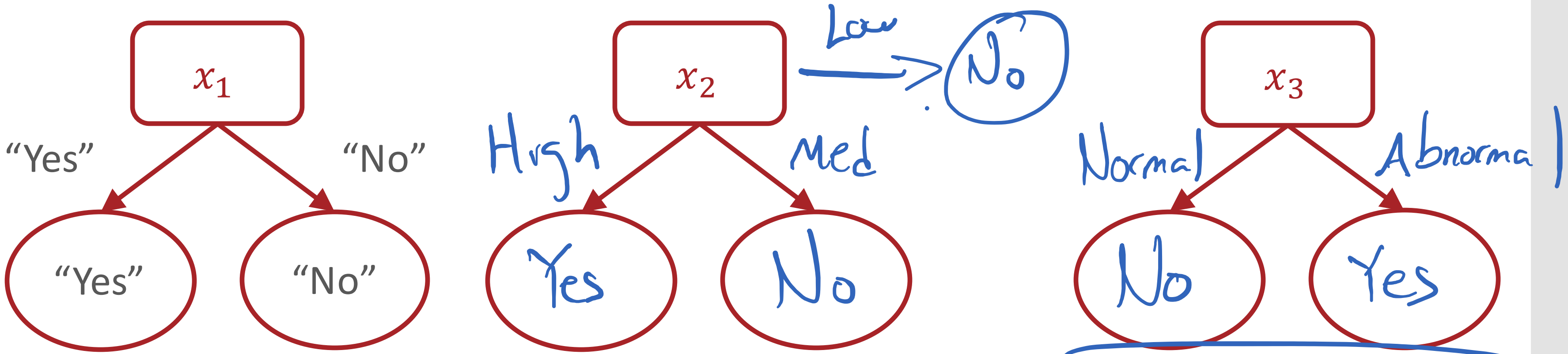
2. Why stop at just one feature?

Splitting Criterion

- A **splitting criterion** is a function that measures how good or useful splitting on a particular feature is *for a specified dataset*
- Insight: use the feature that optimizes the splitting criterion for our decision stump.

Training error rate as a Splitting Criterion

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes



Training error rate: 2/5

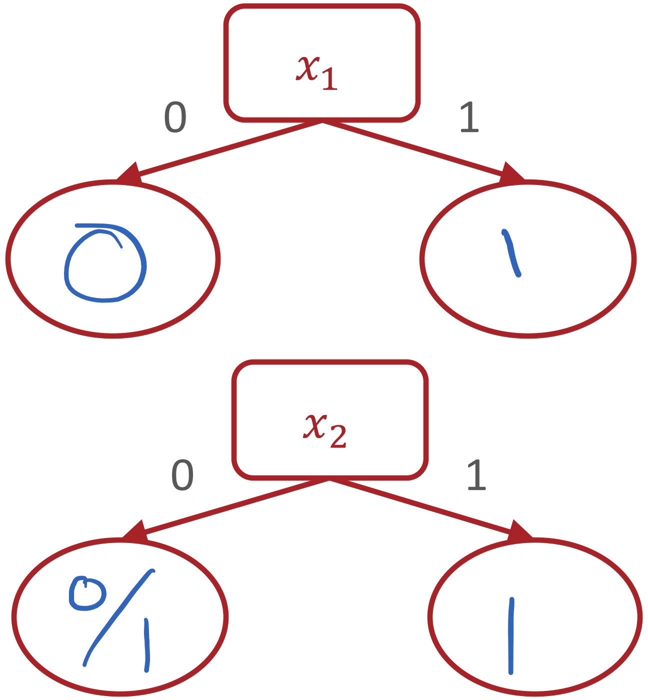
Training error rate: 2/5

Training error rate: 1/5

Training error rate as a Splitting Criterion?

x_1	x_2	y
1	0	0
1	0	0
1	0	1
1	0	1
1	1	1
1	1	1
1	1	1
1	1	1

- Which feature would you split on using training error rate as the splitting criterion?



both have training error rate = 2/8

Splitting Criterion

- A **splitting criterion** is a function that measures how good or useful splitting on a particular feature is *for a specified dataset*
- Insight: use the feature that optimizes the splitting criterion for our decision stump.
- Potential splitting criteria:
  - Training error rate (minimize)
  - Gini impurity (minimize) → CART algorithm
  - Mutual information (maximize) → ID3 algorithm

Splitting Criterion

- A **splitting criterion** is a function that measures how good or useful splitting on a particular feature is *for a specified dataset*
- Insight: use the feature that optimizes the splitting criterion for our decision stump.
- Potential splitting criteria:
  - Training error rate (minimize)
  - Gini impurity (minimize) → CART algorithm
  - Mutual information (maximize) → ID3 algorithm

Entropy

- Entropy describes the purity or uniformity of a collection of values: the lower the entropy, the more pure

size of

$$H(S) = -\sum_{v \in V(S)} \frac{|S_v|}{|S|} \log_2\left(\frac{|S_v|}{|S|}\right)$$

where S is a collection of values,

$V(S)$ is the set of unique values in S

S_v is the collection of elements in S with value v

- If all the elements in S are the same, then

$$H(S) = -\frac{N}{N}\log_2\left(\frac{N}{N}\right) = -1\log_2(1) = 0$$

Entropy

- Entropy describes the purity or uniformity of a collection of values: the lower the entropy, the more pure

$$H(S) = -\sum_{v \in V(S)} \frac{|S_v|}{|S|} \log_2\left(\frac{|S_v|}{|S|}\right)$$

where S is a collection of values,

$V(S)$ is the set of unique values in S

S_v is the collection of elements in S with value v

- If S is split fifty-fifty between two values, then

$$H(S) = -\frac{(N/2)}{N}\log_2\left(\frac{N/2}{N}\right) - \frac{(N/2)}{N}\log_2\left(\frac{N/2}{N}\right)$$

$$= -1/2\left(\log_2(1/2)\right) - 1/2\left(\log_2(1/2)\right) = 1/2 + 1/2 = 1$$

Mutual Information

- Mutual information describes how much information or clarity a particular feature provides about the label

$$I(x_d;Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$$

where x_d is a feature

Y is the collection of all labels

$V(x_d)$ is the set of unique values of x_d

f_v is the fraction of inputs where $x_d = v$

$Y_{x_d=v}$ is the collection of labels where $x_d = v$

Mutual Information: Example

x_d	y
1	1
1	1
0	0
0	0

2/4

2/4

$$I(X_d; Y) = H(Y) - \frac{1}{2} H(Y_{x_d=1}) - \frac{1}{2} H(Y_{x_d=0})$$

$$= 1 - \frac{1}{2}(0) - \frac{1}{2}(0) = 1$$

Mutual Information: Example

x_d	y
1	1
0	1
1	0
0	0

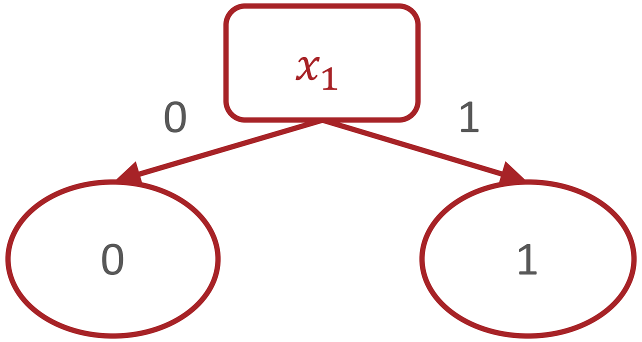
$$I(x_d; y) = H(Y) - \frac{1}{2}H(Y_{x_d=1}) - \frac{1}{2}H(Y_{x_d=0})$$

$$= 1 - \frac{1}{2}(1) - \frac{1}{2}(1) = 0$$

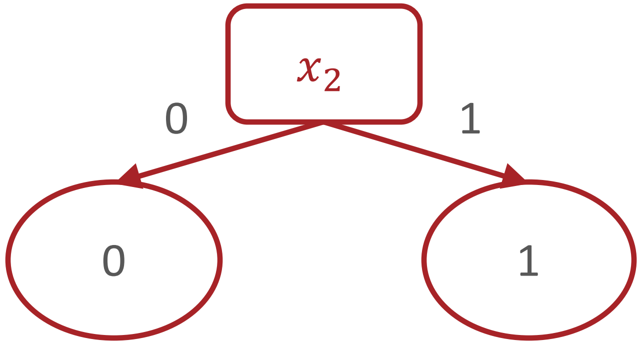
Mutual Information as a Splitting Criterion

x_1	x_2	y
1	0	0
1	0	0
1	0	1
1	0	1
1	1	1
1	1	1
1	1	1
1	1	1

- Which feature would you split on using mutual information as the splitting criterion?



Mutual Information: 0



Mutual Information: $-\frac{2}{8}\log_2\frac{2}{8}-\frac{6}{8}\log_2\frac{6}{8}-\frac{1}{2}(1)-\frac{1}{2}(0)\approx 0.31$

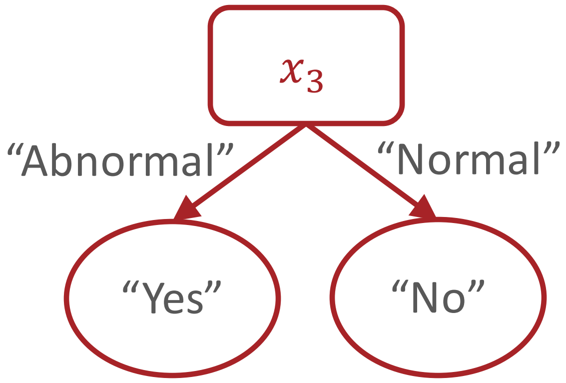
Decision Stumps: Questions

1. How can we pick which feature to split on?

2. Why stop at just one feature?

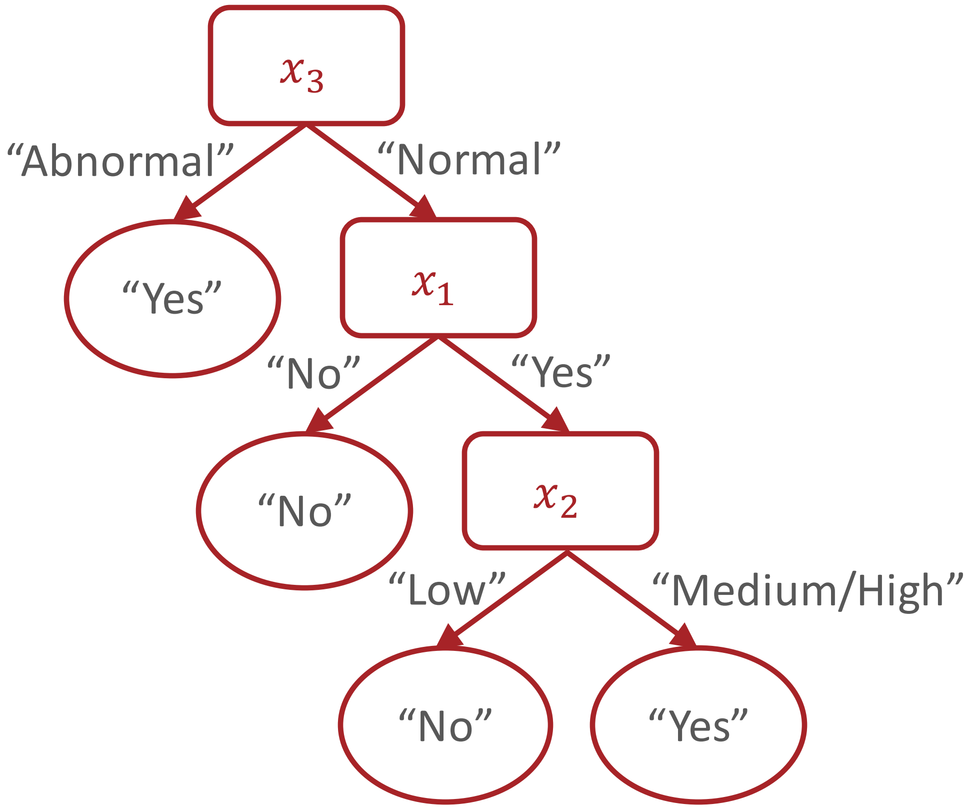
From Decision Stump …

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes



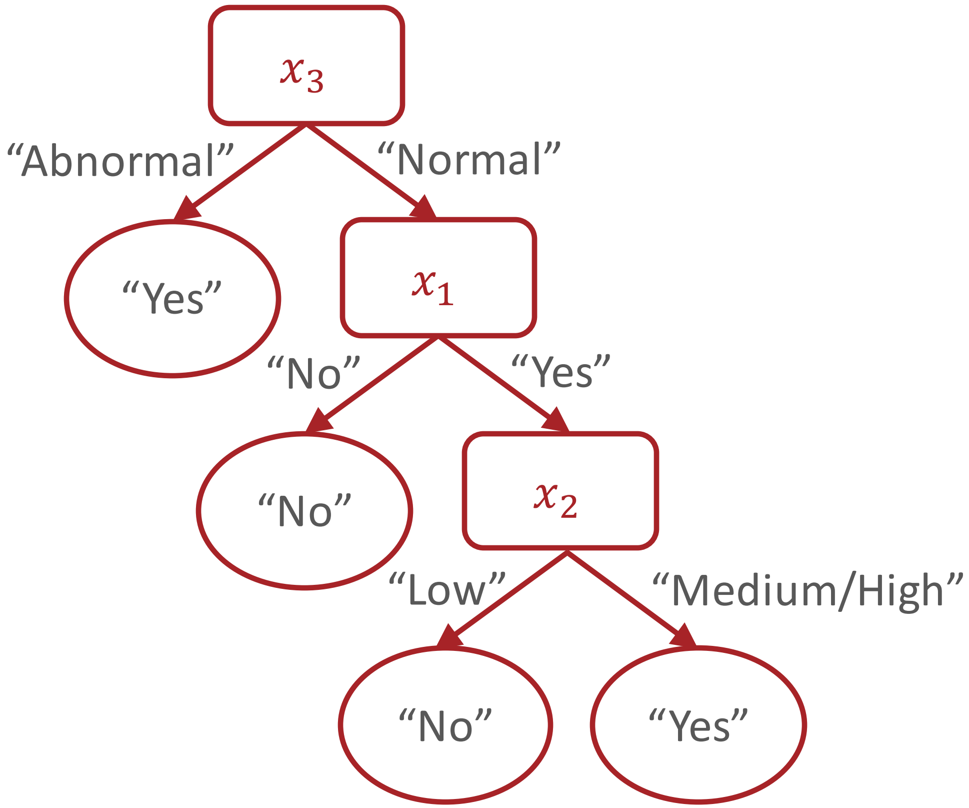
From Decision Stump to Decision Tree

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes



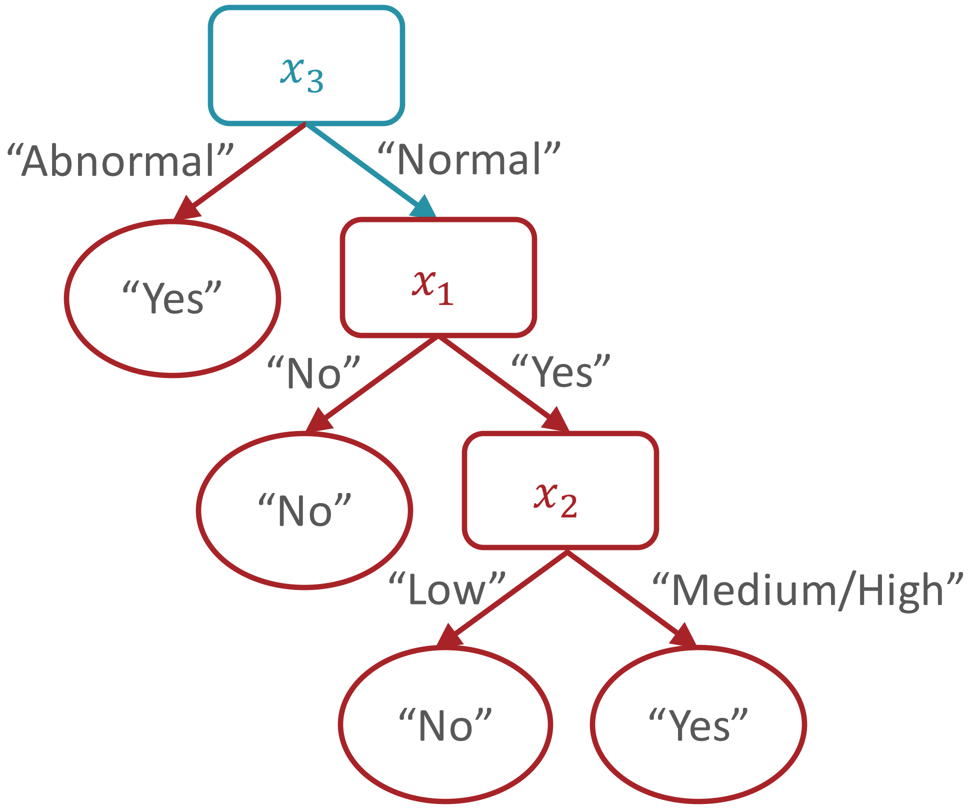
From Decision Stump to Decision Tree

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes
No	High	Normal	No



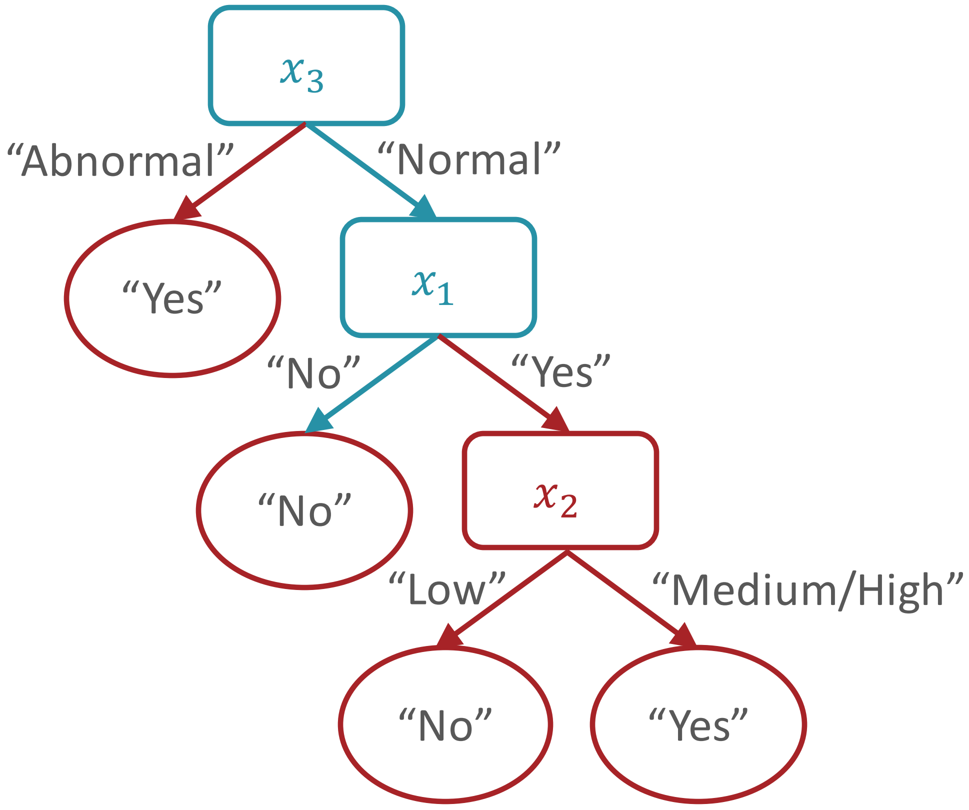
From Decision Stump to Decision Tree

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes
No	High	Normal	No



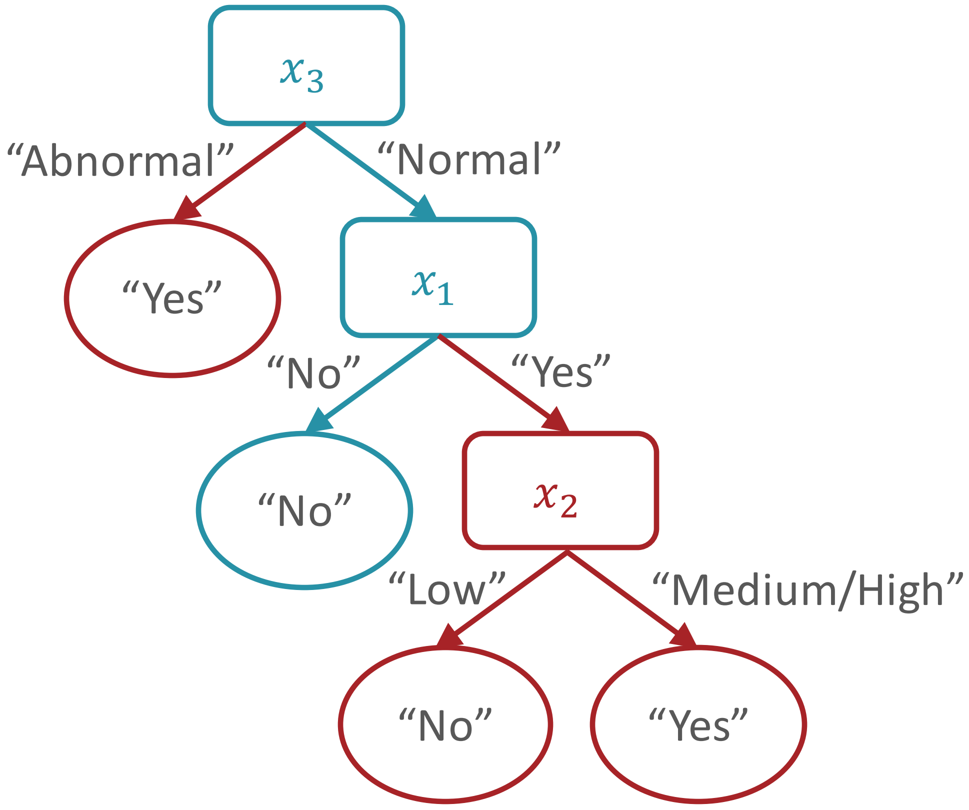
From Decision Stump to Decision Tree

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes
No	High	Normal	No



From Decision Stump to Decision Tree

x_1 Family History	x_2 Resting Blood Pressure	x_3 Cholesterol	y Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes
No	High	Normal	No



Decision Tree Prediction: Pseudocode

$[x'_1, x'_2, \ldots, x'_D]$

```
def predict(x'):
    # walk from the root node to a leaf node
    while(true):
        if current_node =/= leaf
            check associated feature, x_d
            go down branch corresponding to x'_d
        if current_node == leaf
            return label stored in the node
```

Decision Tree Learning: Pseudocode

```
def train(D):
    store root = tree_recurse(D)

def tree_recurse(D'):
    q = new node()
    base case - if (SOME CONDITION):
    recursion - else:
        find best feature to split on, x_d
        q.split = x_d
        for v ∈ V(x_d) (all possible values of x_d)
            D_v = {(x^(n), y^(n)) ∈ D' | x_d^(n) = v}
            q.child(v) = tree_recurse(D_v)
    return q
```

Decision Tree: Pseudocode

```
def train(𝒟):


def tree_recurse(𝒟′):
    q = new node()
    base case – if (all labels in D' are the same
                    OR all features have been split on
                       (all feature vectors in D' are identical)
                    OR D' is empty (or just very small))
        q.label = majority_vote(D')

    recursion – else:

    return q
```

Decision Tree: Example (~~Iteratively~~)

- How is Henry getting to work?
- Label: mode of transportation
  - $y \in \mathcal{Y} = \{\text{Bike}, \text{Drive}, \text{Bus}\}$
- Features: 4 categorial features
  - Is it raining? $x_1 \in \{\text{Rain}, \text{No Rain}\}$
  - When am I leaving (relative to rush hour)? $x_2 \in \{\text{Before}, \text{During}, \text{After}\}$
  - What am I bringing? $x_3 \in \{\text{Backpack}, \text{Lunchbox}, \text{Both}\}$
  - Am I tired? $x_4 \in \{\text{Tired}, \text{Not Tired}\}$

Data

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Which feature would we split on first using mutual information as the splitting criterion?

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall:

$$H(S) = -\sum_{v \in V(S)} \frac{|S_v|}{|S|} \log_2\left(\frac{|S_v|}{|S|}\right)$$

$H(Y)$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall:

$$H(S) = -\sum_{v \in V(S)} \frac{|S_v|}{|S|} \log_2\left(\frac{|S_v|}{|S|}\right)$$

$$H(Y) = -\frac{3}{16}\log_2\frac{3}{16} - \frac{6}{16}\log_2\frac{6}{16} - \frac{7}{16}\log_2\frac{7}{16}$$

$$H(Y) \approx 1.5052$$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall: $I(x_d; Y) = H(Y)$

$$- \sum_{v \in V(x_d)} (f_v) \left(H\left(Y_{x_d = v}\right) \right)$$

$I(x_1, Y$

$$\mathrm{IG}(x_1, y) = -\frac{7}{16} \log_2 \left(\frac{7}{16}\right)$$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall: $I(x_d; Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$

$I(x_1, Y) \approx 1.5052$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

$IG(x_1, y$ … 16 … (16)

Recall: $I(x_d;Y) = H(Y)$

$$- \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$$

$I(x_1, Y) \approx 1.5052$

$$-\frac{6}{16}(1)$$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall: $I(x_d;Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H(Y_{x_d=v})\right)$

$I(x_1, Y) \approx 1.5052$

$-\frac{6}{16}(1)$

$-\frac{10}{16}\left(-\frac{3}{10}\log_2\left(\frac{3}{10}\right) - \frac{3}{10}\log_2\left(\frac{3}{10}\right) - \frac{4}{10}\log_2\left(\frac{4}{10}\right)\right)$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

$$\text{Recall: } I(x_d; Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$$

$$I(x_1, Y) \approx 1.5052 - \frac{6}{16}(1) - \frac{10}{16}(1.5710) \approx 0.1482$$

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall: $I(x_d; Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H(Y_{x_d=v})\right)$

$I(x_d, Y)$	
x_1	0.1482
x_2	0.1302
x_3	0.5358
x_4	0.5576

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Not Tired	Bus
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Not Tired	Bus
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Backpack	Not Tired	Bus
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Both	Tired	Drive
No Rain	After	Lunchbox	Not Tired	Bus

Recall: $I(x_d; Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$

	$I(x_d, Y)$
x_1	0.1482
x_2	0.1302
x_3	0.5358
x_4	0.5576

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Tired	Drive

Recall: $I(x_d; Y) = H(Y) - \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$

$I(x_d, Y)$	
x_1	0.1482
x_2	0.1302
x_3	0.5358
x_4	0.5576

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Tired	Metro
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Tired	Drive

Recall: $I(x_d; Y) = H(Y)$

$$- \sum_{v \in V(x_d)} (f_v)\left(H\left(Y_{x_d=v}\right)\right)$$

$I(x_d, Y)$	
x_1	0.1482
x_2	0.1302
x_3	0.5358
x_4	0.5576

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Tired	Drive

x_4

Not Tired

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus

Tired

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Tired	Drive

Decision Tree: Example

x_4

Not Tired

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus

Tired

x_1	x_2	x_3	x_4	y
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
Rain	After	Backpack	Tired	Bus
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Backpack	Tired	Bike
No Rain	Before	Lunchbox	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Backpack	Tired	Bike
No Rain	After	Both	Tired	Drive

x_4

Not Tired

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus

Tired

x_1	x_2	x_3	x_4	y
Rain	After	Backpack	Tired	Bus
No Rain	Before	Backpack	Tired	Bike
No Rain	After	Backpack	Tired	Bike
Rain	Before	Both	Tired	Drive
Rain	During	Both	Tired	Drive
No Rain	During	Both	Tired	Drive
No Rain	After	Both	Tired	Drive
Rain	After	Lunchbox	Tired	Drive
No Rain	Before	Lunchbox	Tired	Drive

$$I\left(x_1, Y_{x_4=\text{Tired}}\right) \approx 0.3244$$

$$I\left(x_2, Y_{x_4=\text{Tired}}\right) \approx 0.2516$$

$$I\left(x_3, Y_{x_4=\text{Tired}}\right) \approx \mathbf{0.9183}$$

x_4

Not Tired

Tired

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus

x_3

Backpack

Both, Lunchbox

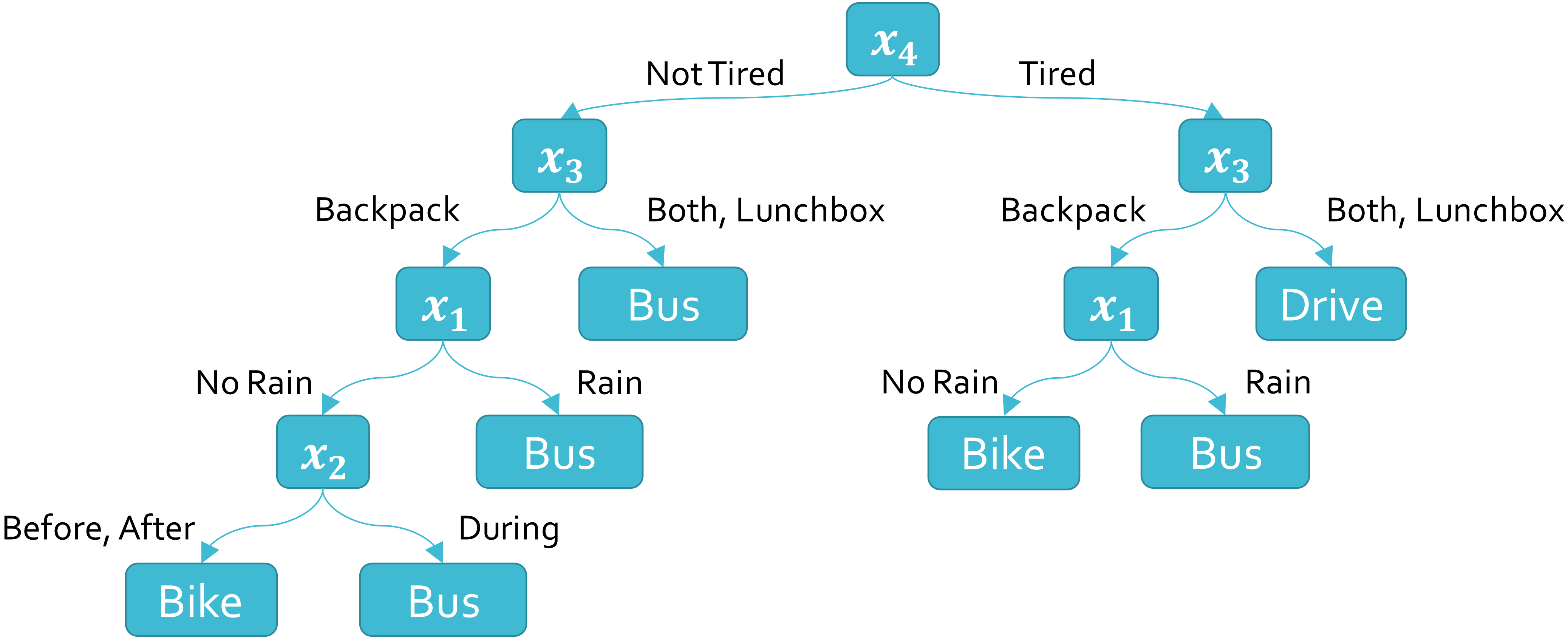
x_1	x_2	y
Rain	After	Bus
No Rain	Before	Bike
No Rain	After	Bike

Drive

$$I\left(x_1, Y_{x_4=\text{Tired}}\right) \approx 0.3244$$

$$I\left(x_2, Y_{x_4=\text{Tired}}\right) \approx 0.2516$$

$$I\left(x_3, Y_{x_4=\text{Tired}}\right) \approx \mathbf{0.9183}$$


x_4
Not Tired
Tired
x_3
x_3
Backpack
Both, Lunchbox
Backpack
Both, Lunchbox
x_1
Bus
x_1
Drive
No Rain
Rain
No Rain
Rain
x_2
Bus
Bike
Bus
Before, After
During
Bike
Bus

Decision Trees: Inductive Bias

- The **inductive bias** of a machine learning algorithm is the principal by which it generalizes to unseen examples
- What is the inductive bias of the ID3 algorithm i.e., decision tree learning with mutual information maximization as the splitting criterion?
  - Try to find the shortest tree that achieves zero (or the lowest possible) training error with high mutual information features at the top

Decision Trees: Pros & Cons

- Pros
  - Interpretable
  - Efficient (computational cost and storage)
  - Can be used for classification and regression tasks
  - Compatible with categorical and real-valued features
- Cons

Real-Valued Features: Example - x = Outside Temperature (°F)

x	y
74	Drive
55	Metro
63	Bike
33	Drive
80	Drive
81	Drive
44	Metro
45	Metro
78	Drive
51	Metro

x	y
33	Drive
44	Metro
45	Metro
51	Metro
55	Metro
63	Bike
74	Drive
78	Drive
80	Drive
81	Drive

$x < 38.5$

Real-Valued Features: Example - x = Outside Temperature (°F)

x	y
74	Drive
55	Metro
63	Bike
33	Drive
80	Drive
81	Drive
44	Metro
45	Metro
78	Drive
51	Metro

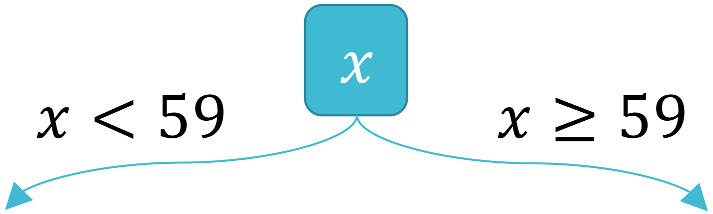
x	y
33	Drive
44	Metro
45	Metro
51	Metro
55	Metro
63	Bike
74	Drive
78	Drive
80	Drive
81	Drive

$x < 44.5$

Real-Valued Features: Example - x = Outside Temperature (°F)

x	y
74	Drive
55	Metro
63	Bike
33	Drive
80	Drive
81	Drive
44	Metro
45	Metro
78	Drive
51	Metro

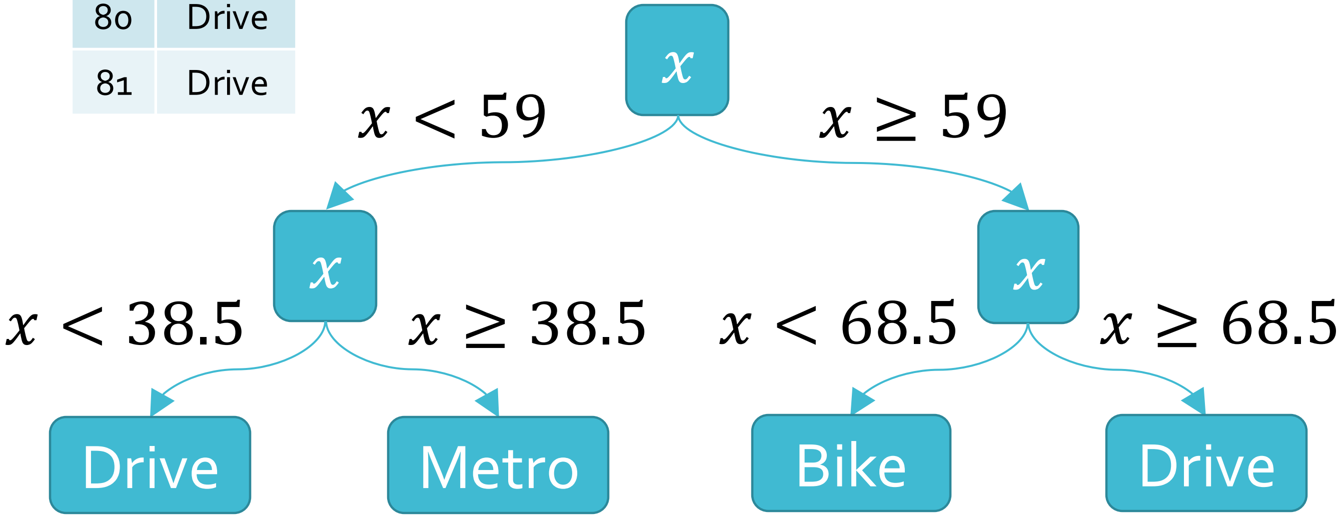
x	y
33	Drive
44	Metro
45	Metro
51	Metro
55	Metro
63	Bike
74	Drive
78	Drive
80	Drive
81	Drive



Real-Valued Features: Example - x = Outside Temperature (°F)

x	y
74	Drive
55	Metro
63	Bike
33	Drive
80	Drive
81	Drive
44	Metro
45	Metro
78	Drive
51	Metro

x	y
33	Drive
44	Metro
45	Metro
51	Metro
55	Metro
63	Bike
74	Drive
78	Drive
80	Drive
81	Drive



Decision Trees: Pros & Cons

- Pros
  - Interpretable
  - Efficient (computational cost and storage)
  - Can be used for classification and regression tasks
  - Compatible with categorical and real-valued features
- Cons
  - Learned greedily: each split only considers the immediate impact on the splitting criterion
    - Not guaranteed to find the smallest (fewest number of splits) tree that achieves a training error rate of 0.
  - Liable to overfit!

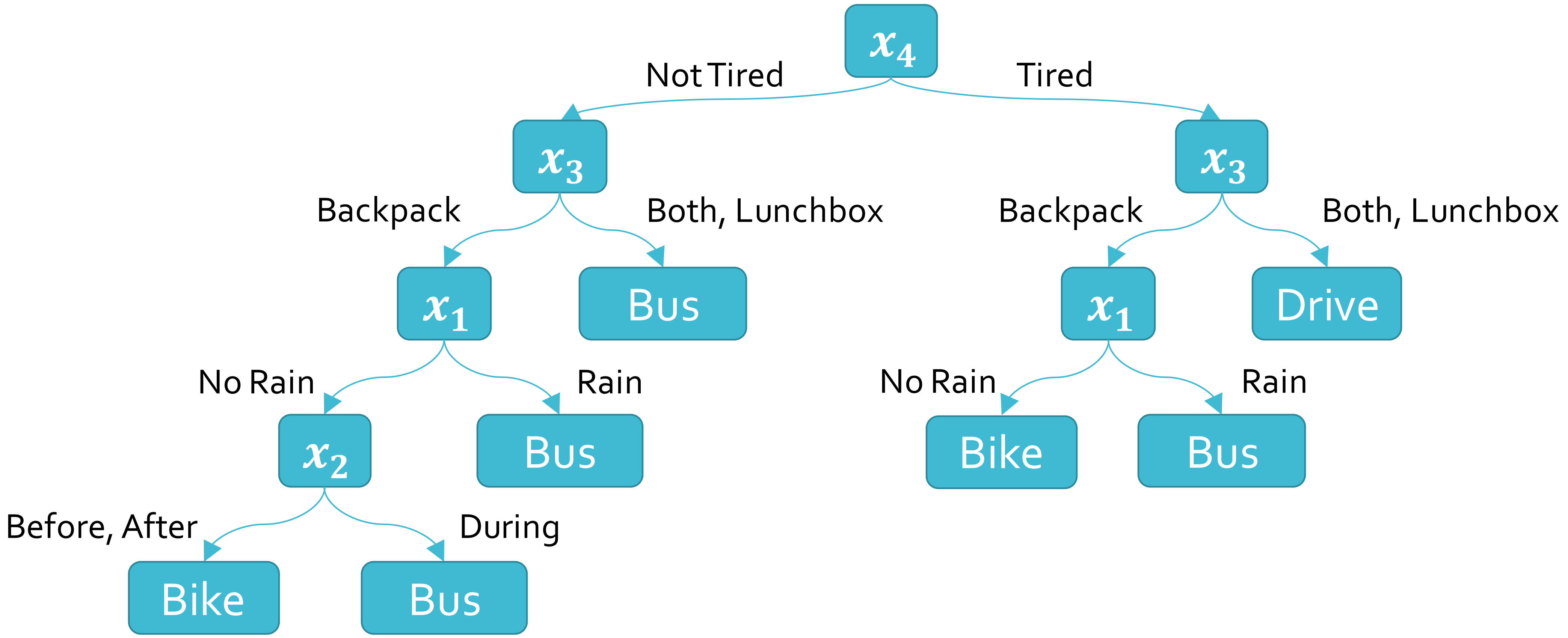
Overfitting

- Overfitting occurs when the classifier (or model)…
  - is too complex
  - fits noise or “outliers” in the training dataset as opposed to the actual pattern of interest
  - doesn’t have enough inductive bias pushing it to generalize
- Underfitting occurs when the classifier (or model)…
  - is too simple
  - can’t capture the actual pattern of interest in the training dataset
  - has too much inductive bias

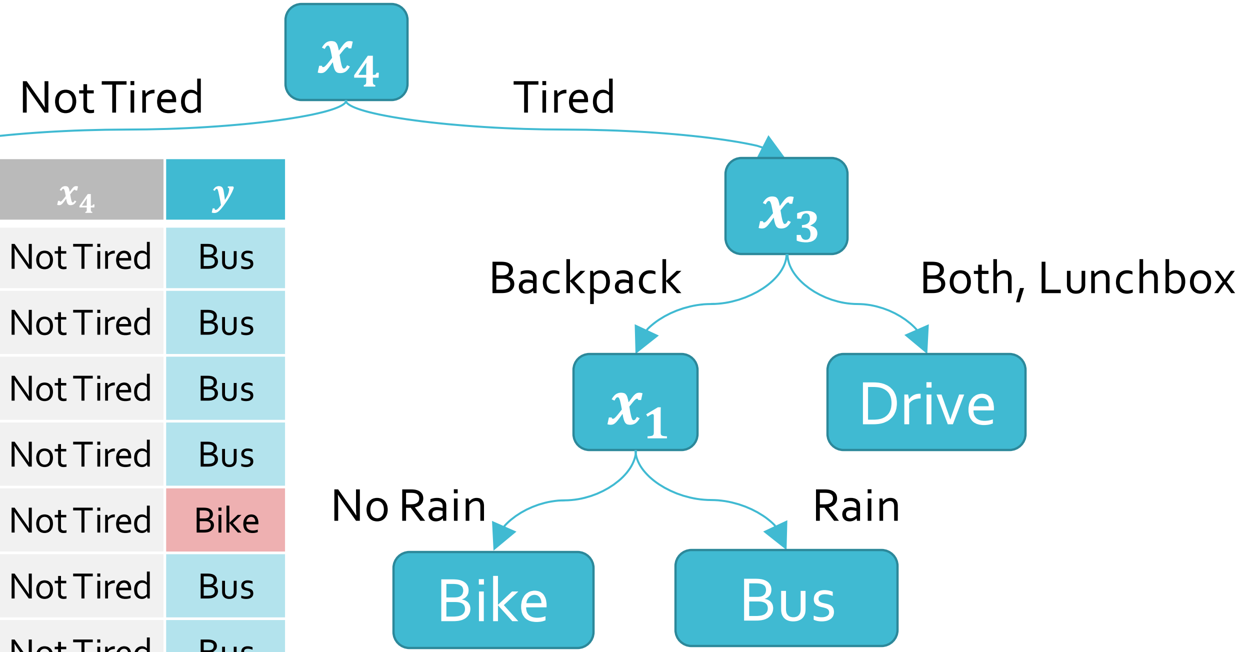
Different Kinds of Error

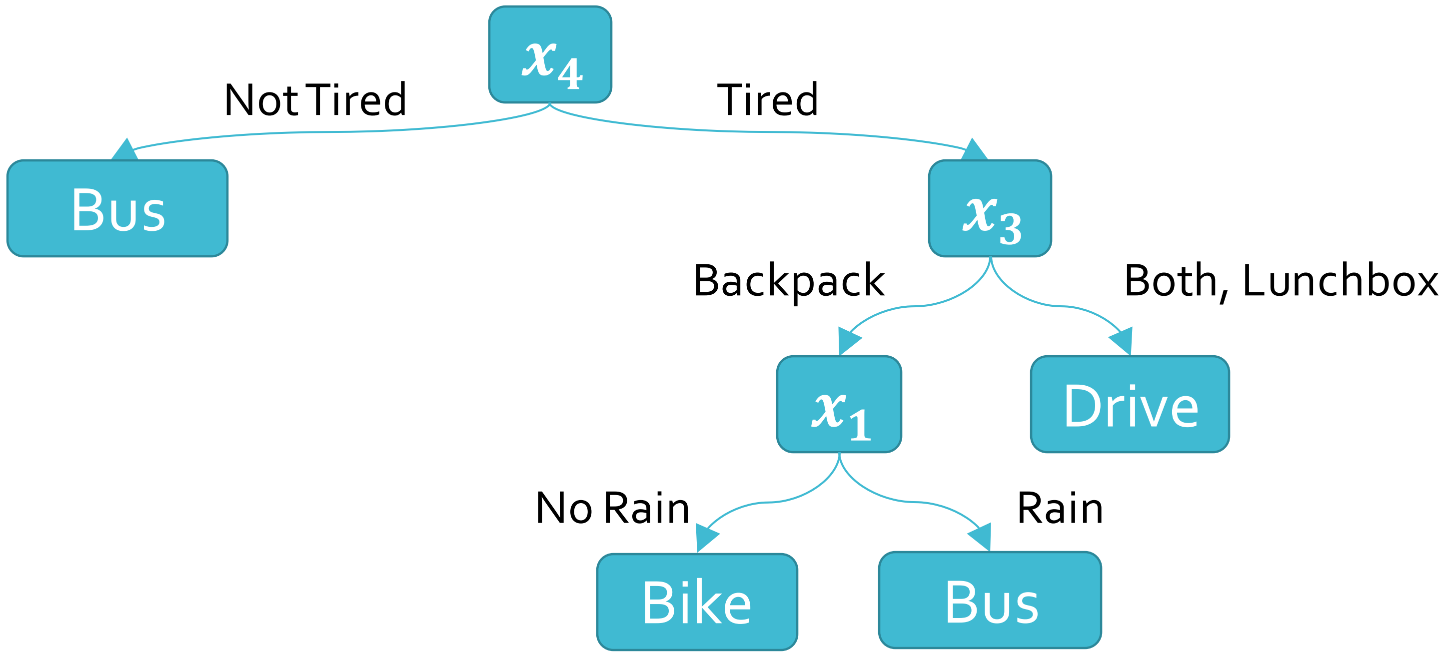
- Training error rate = $err(h, \mathcal{D}_{train})$
- Test error rate = $err(h, \mathcal{D}_{test})$
- True error rate = $err(h)$

  = the error rate of h on all possible examples
  - In machine learning, this is the quantity that we care about but, in most cases, it is unknowable.
- Overfitting occurs when $err(h) > err(h, \mathcal{D}_{train})$
  - $err(h) - err(h, \mathcal{D}_{train})$ can be thought of as a measure of overfitting


x_4
Not Tired
Tired
x_3
x_3
Backpack
Both, Lunchbox
Backpack
Both, Lunchbox
x_1
Bus
x_1
Drive
No Rain
Rain
No Rain
Rain
x_2
Bus
Bike
Bus
Before, After
During
Bike
Bus

x_1	x_2	x_3	x_4	y
Rain	During	Both	Not Tired	Bus
Rain	After	Backpack	Not Tired	Bus
No Rain	Before	Lunchbox	Not Tired	Bus
No Rain	During	Backpack	Not Tired	Bus
No Rain	After	Backpack	Not Tired	Bike
No Rain	After	Both	Not Tired	Bus
No Rain	After	Lunchbox	Not Tired	Bus





This tree only misclassifies one training data point!

Key Takeaways

- Decision tree prediction algorithm
- Decision tree learning algorithm via recursion
- Inductive bias of decision trees
- Overfitting vs. Underfitting
- How to combat overfitting in decision trees