Decision Trees

Original Slides by:
Courtney Anne Ngo
Daniel Stanley Tan, PhD
Arren Antioquia

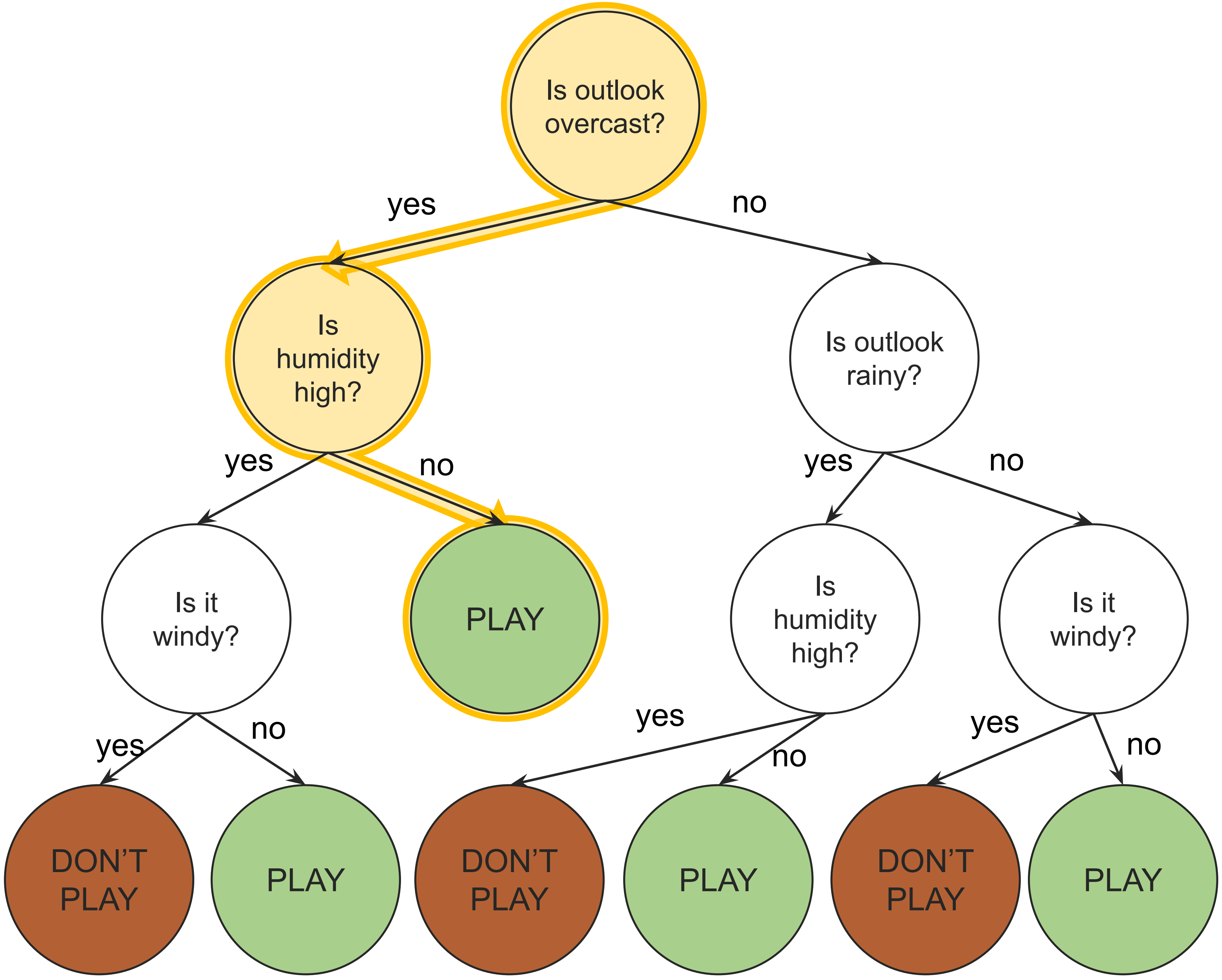
Updated (AY 2023 – 2024 T3) by:
Thomas James Tiam-Lee, PhD

Decision Trees

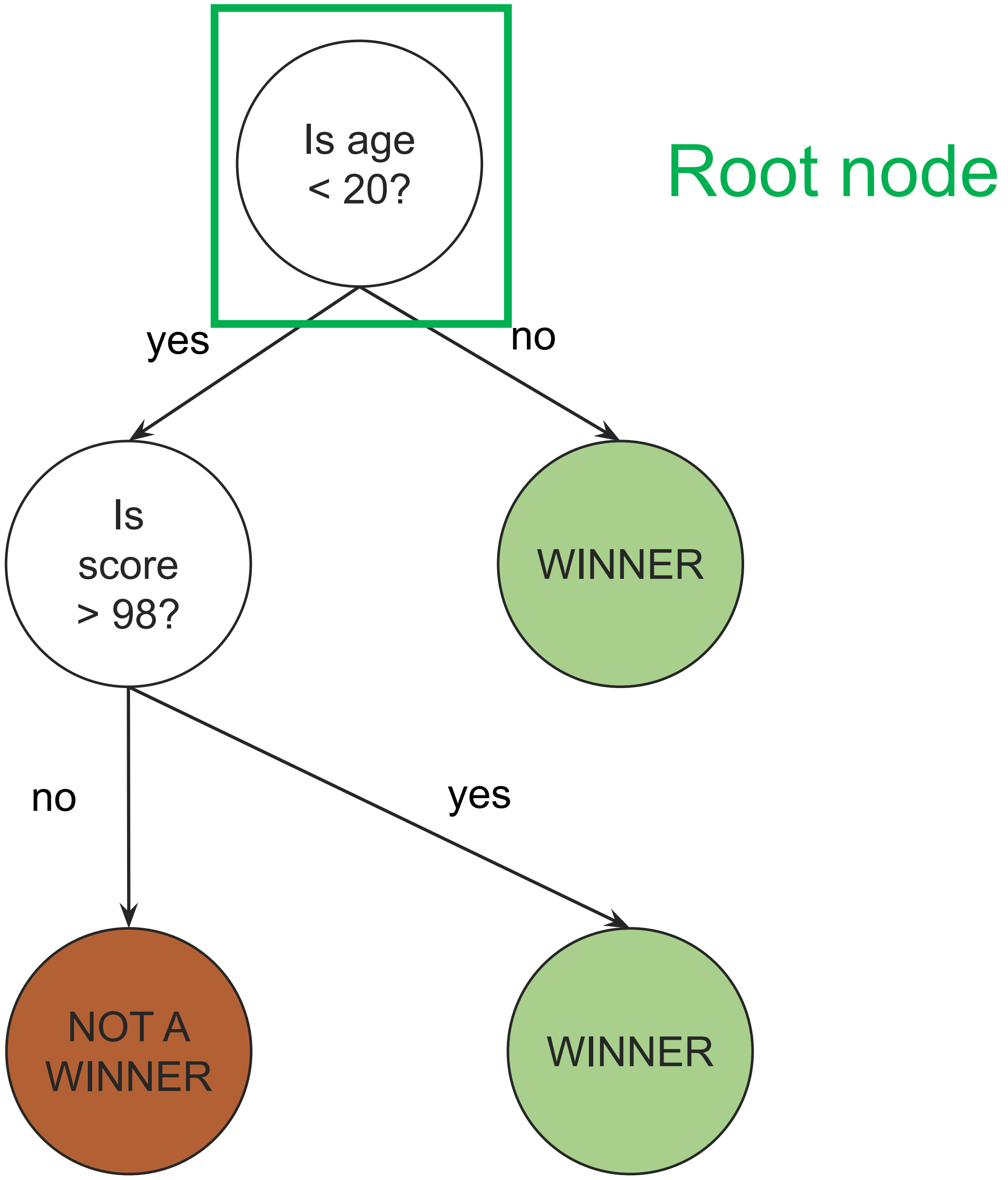
- **Supervised** machine learning algorithm for both classification and regression.

- Key idea: Make a prediction by **asking a series of yes / no questions**, based on the historical data.

X			y
Outlook	Humidity (Nominal)	Windy	Play
overcast	high	FALSE	yes
overcast	normal	TRUE	yes
overcast	high	TRUE	yes
overcast	normal	FALSE	yes
rainy	high	FALSE	yes
rainy	normal	FALSE	yes
rainy	normal	TRUE	no
rainy	normal	FALSE	yes
rainy	high	TRUE	no
sunny	high	FALSE	no
sunny	high	TRUE	no
sunny	high	FALSE	no
sunny	normal	FALSE	yes
sunny	normal	TRUE	yes

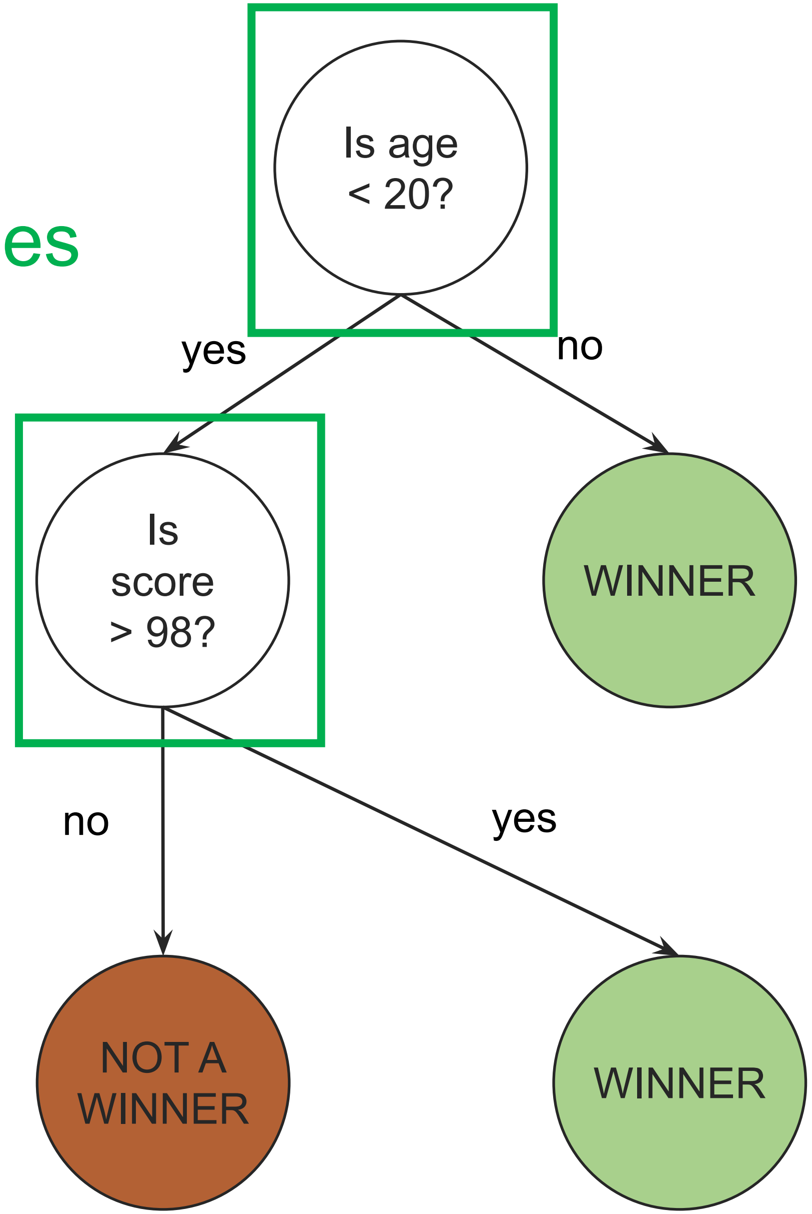


Terminology

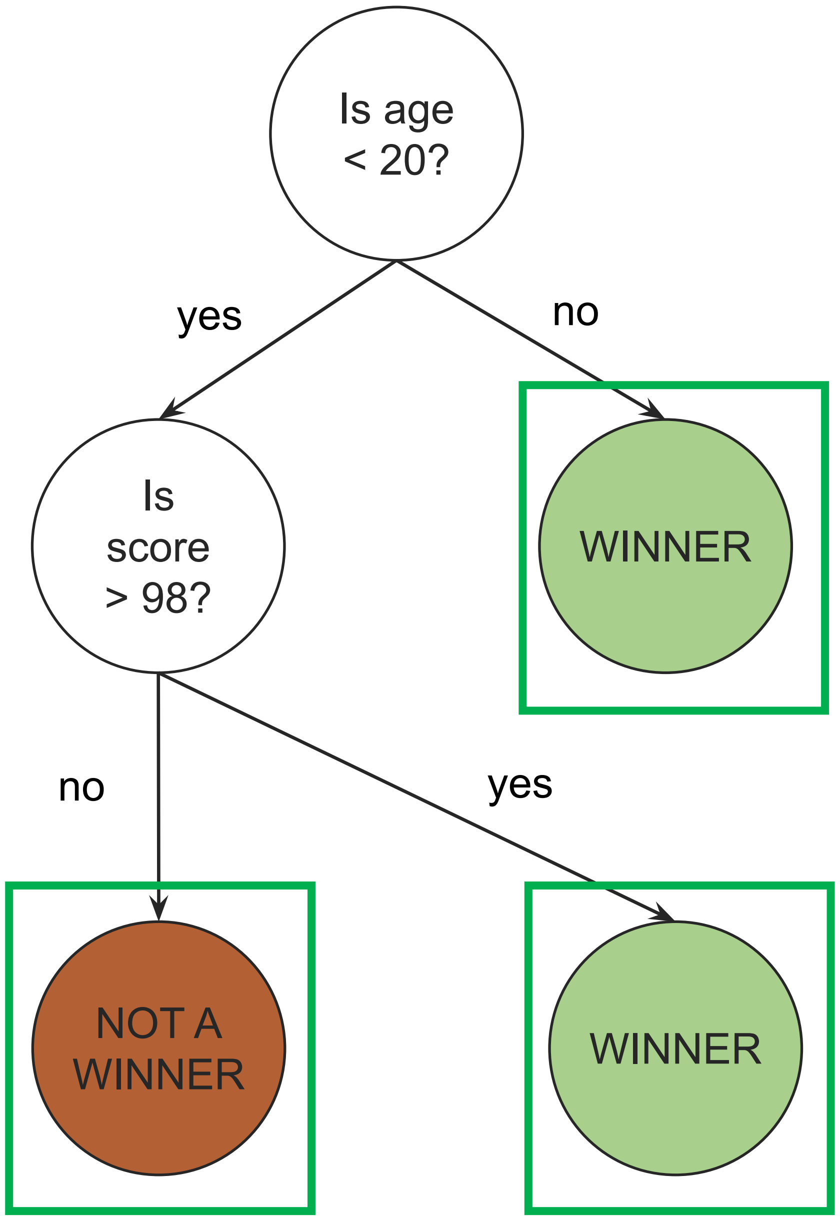


Terminology

Internal nodes
(queries)



Terminology



Leaf nodes
(decisions)

Example Data

X (features) y (label)

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

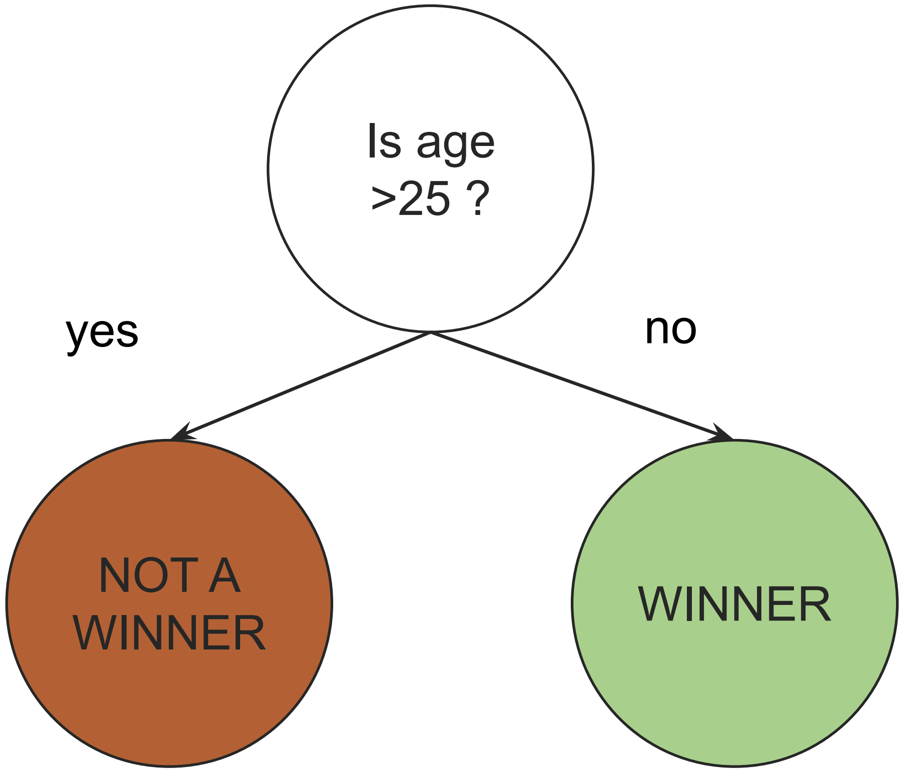
If we want to predict whether someone is a winner, what is the best yes/no question to ask?

Why?

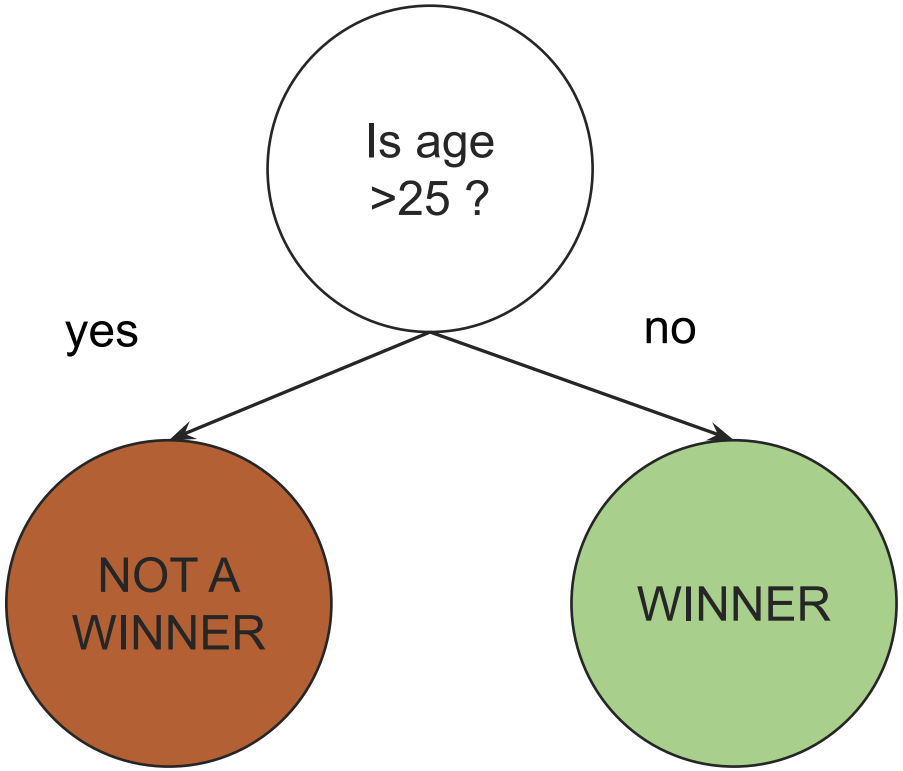
Example Data

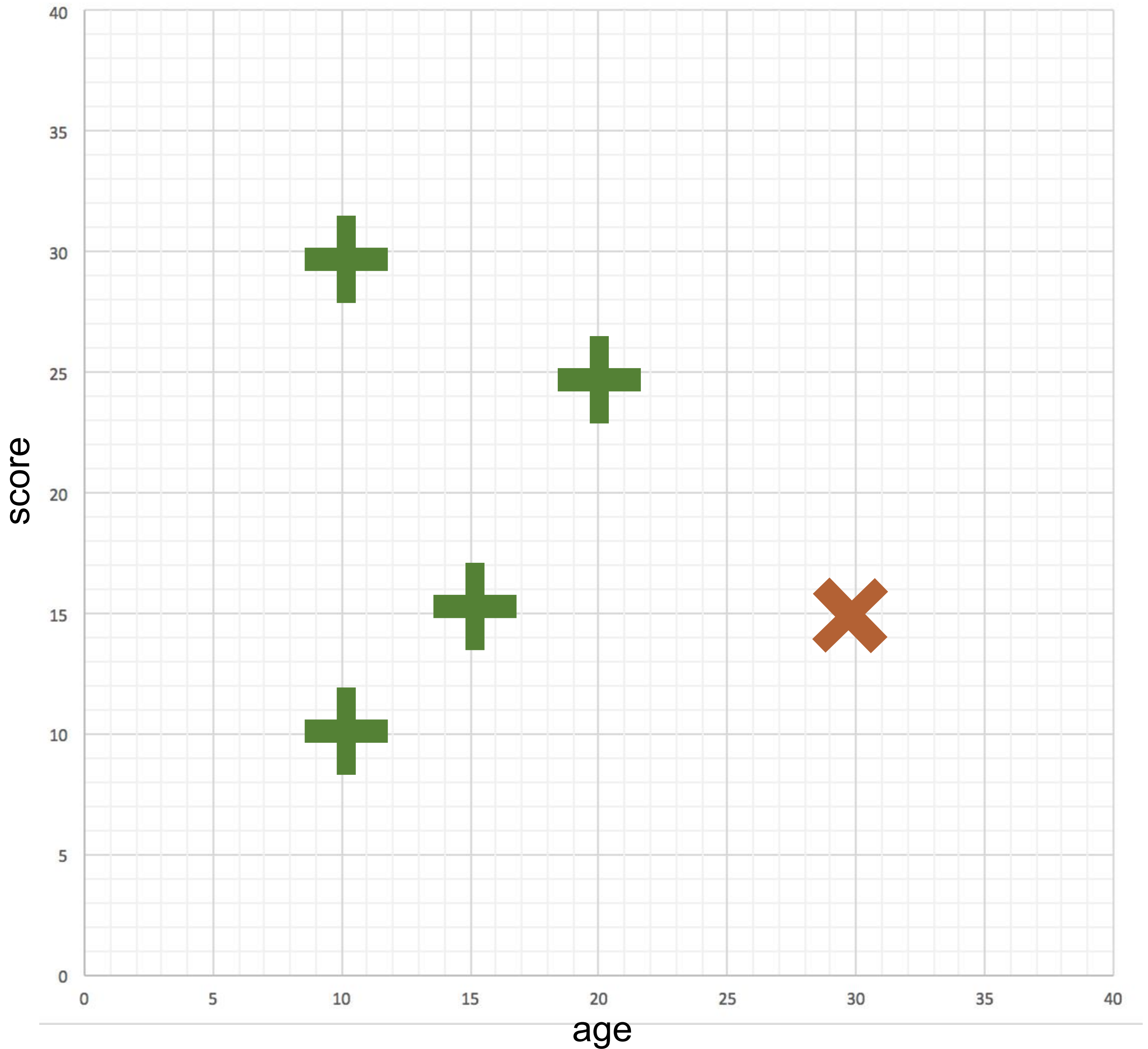
X (features) y (label)

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

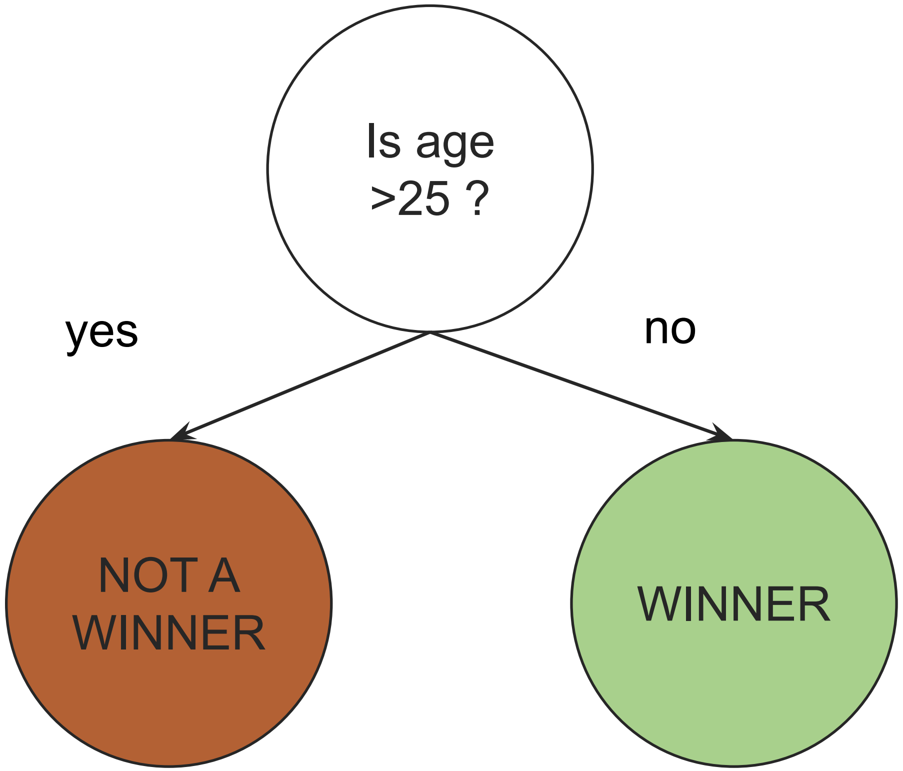


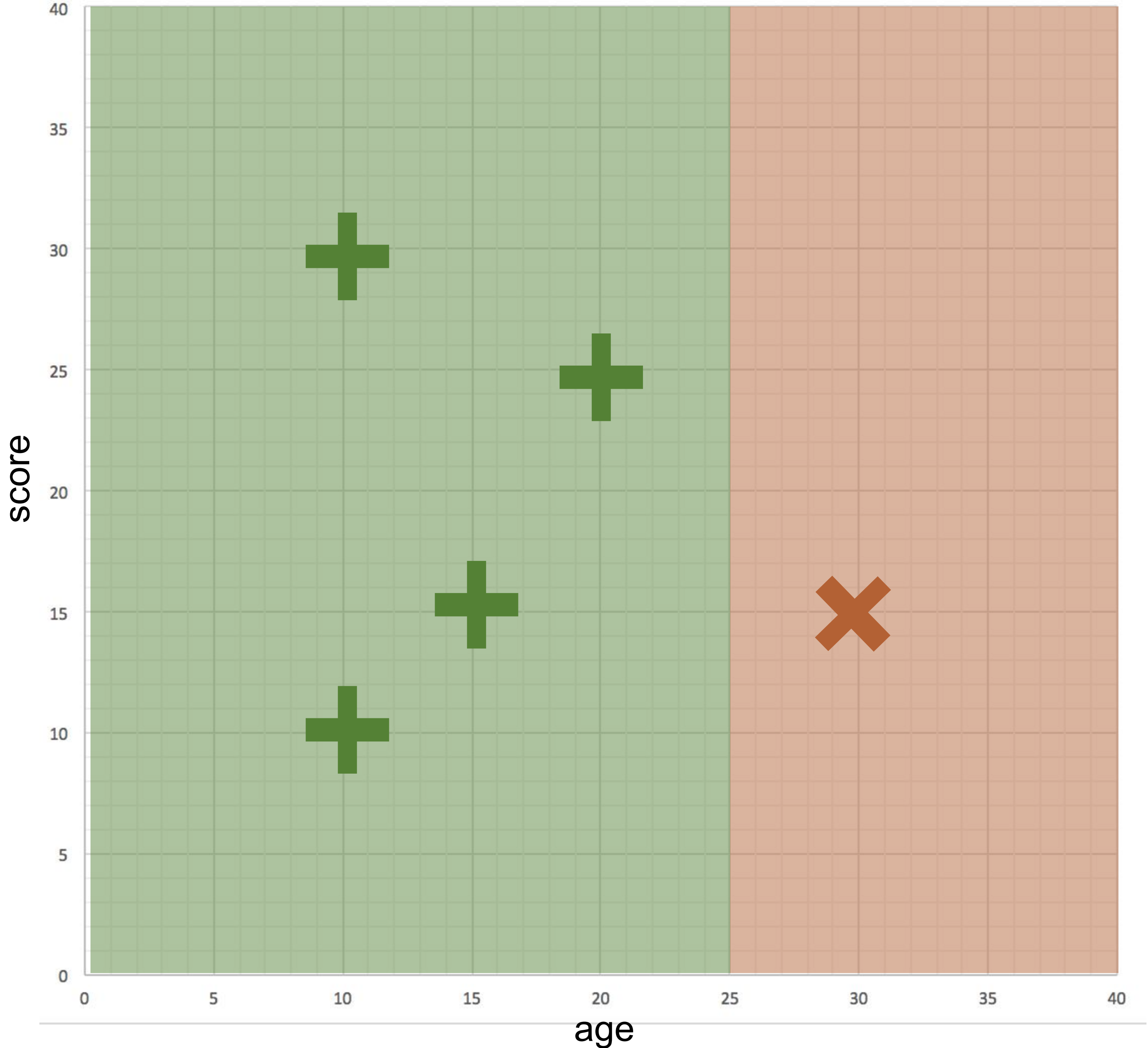
Example Data





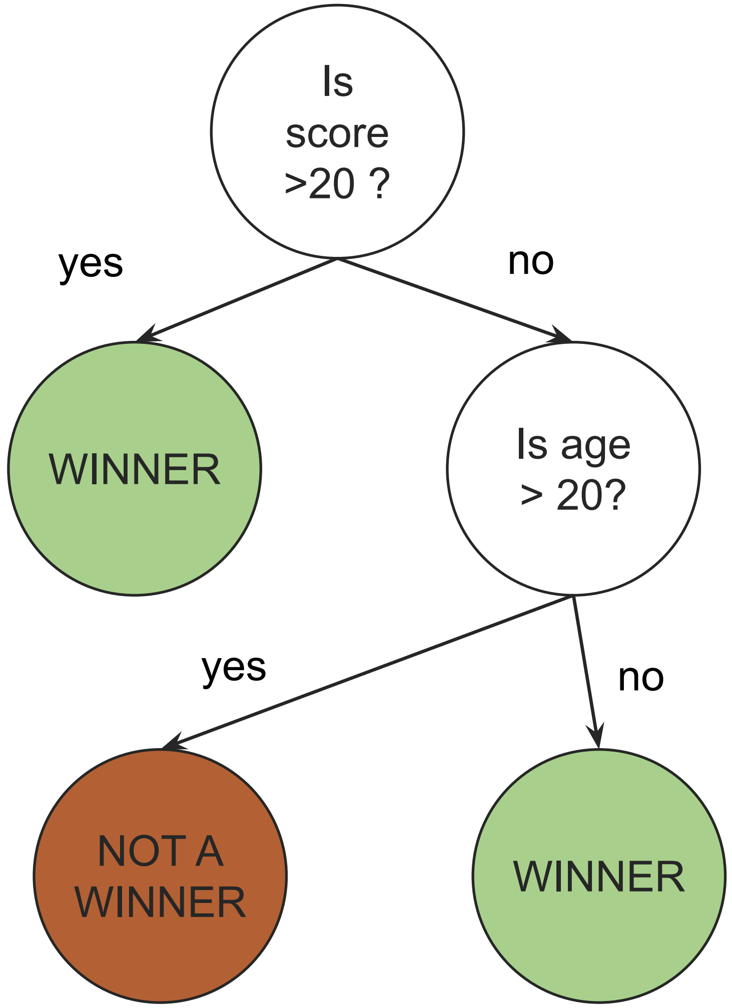
Example Data



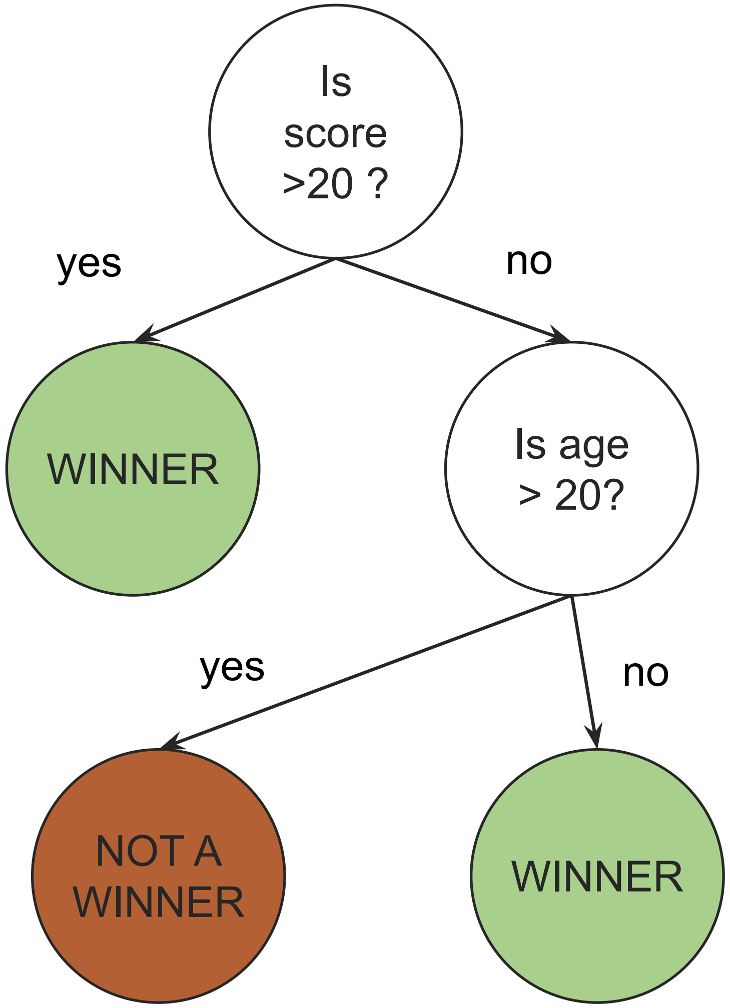


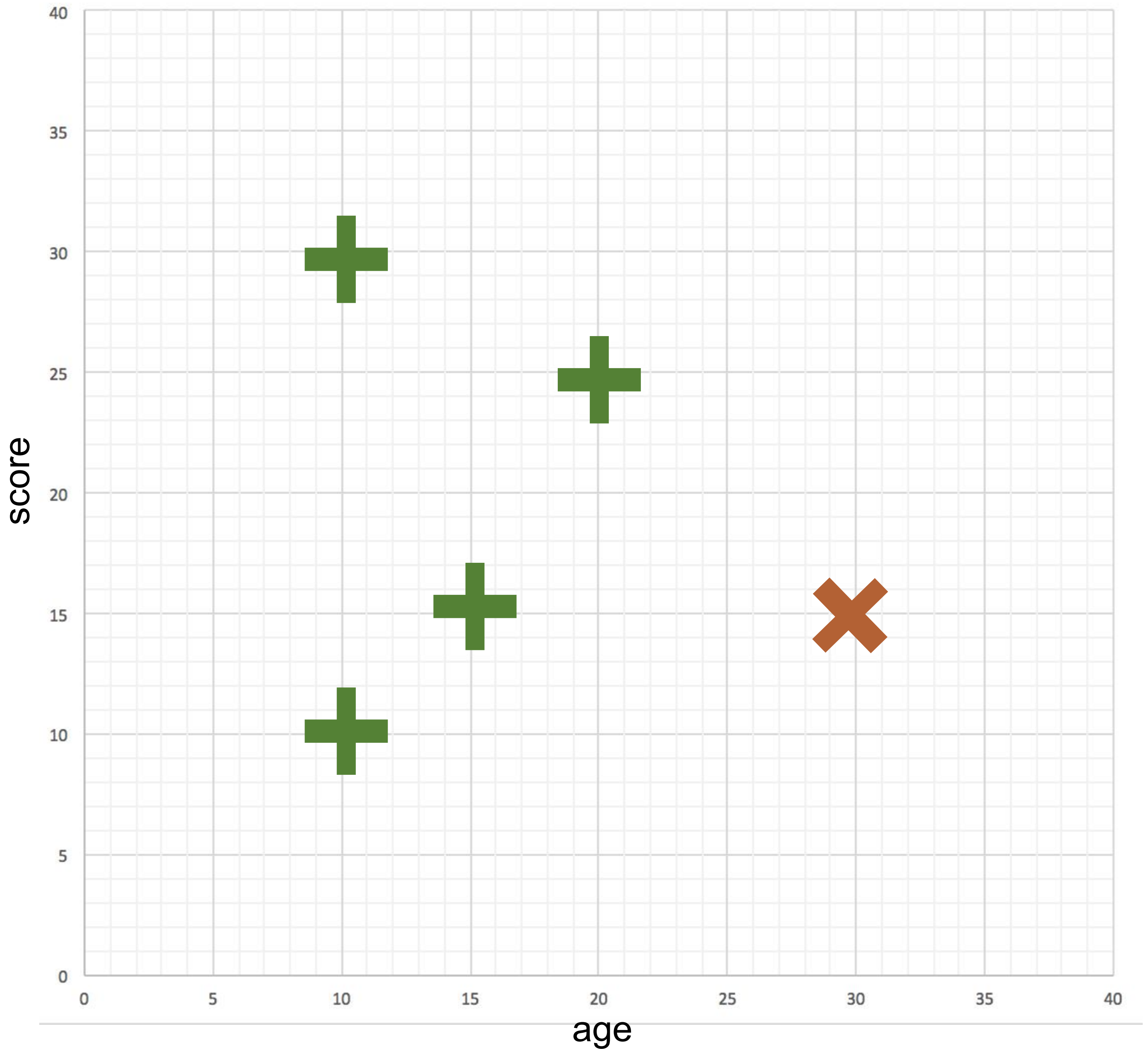
Example Data

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

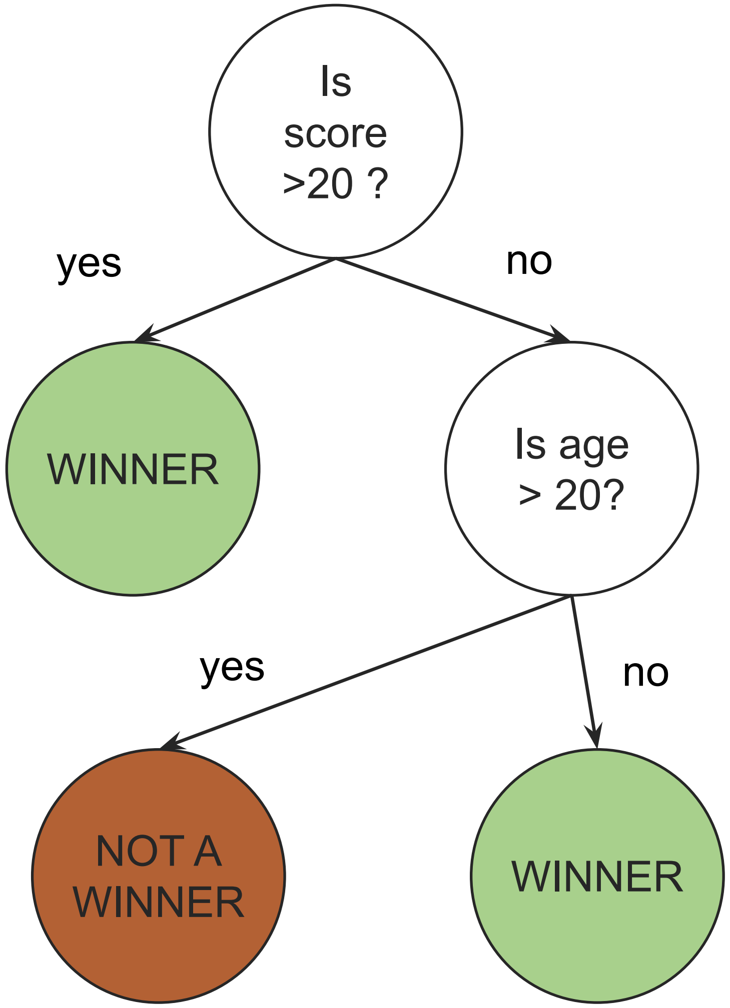


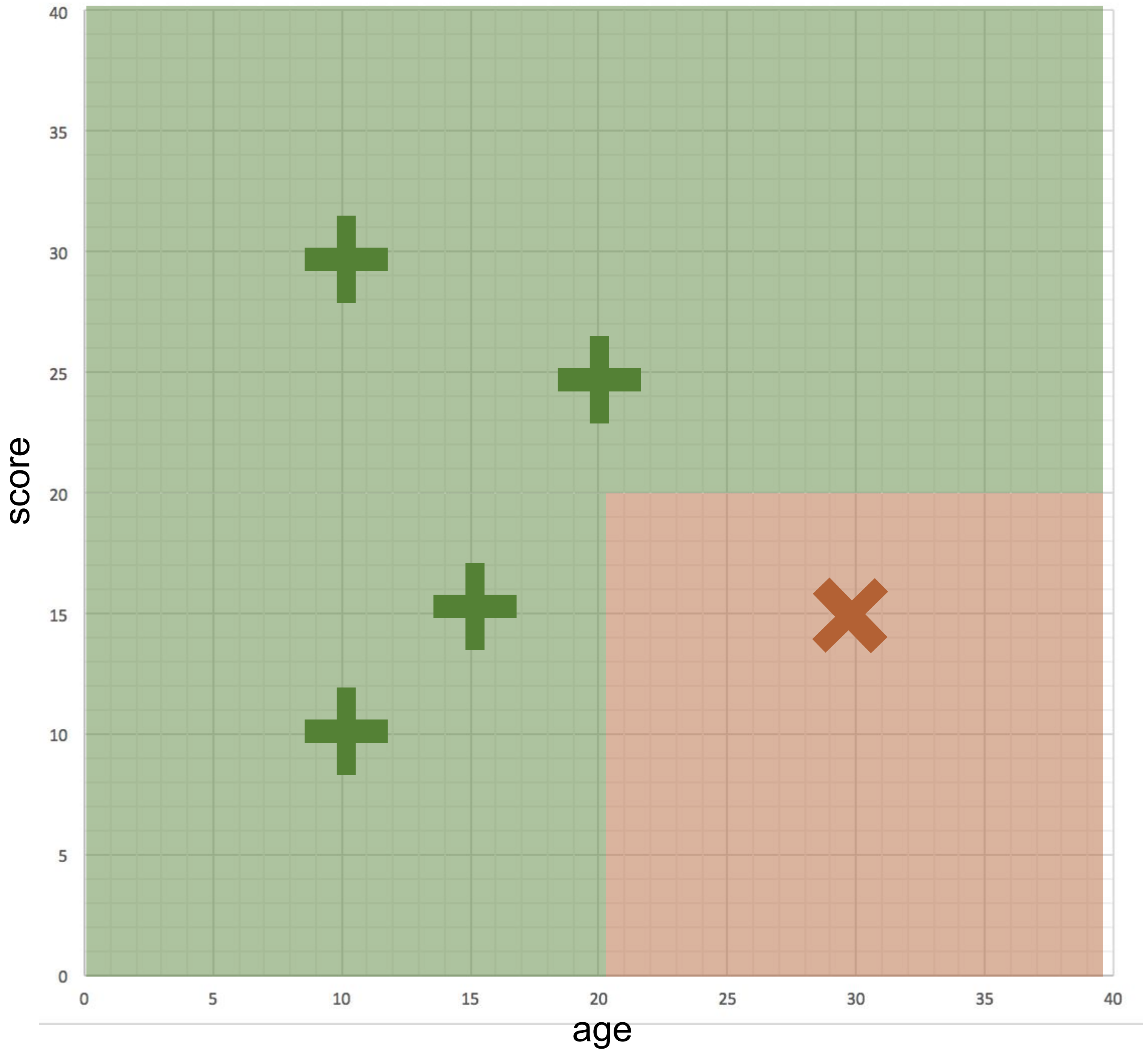
Example Data





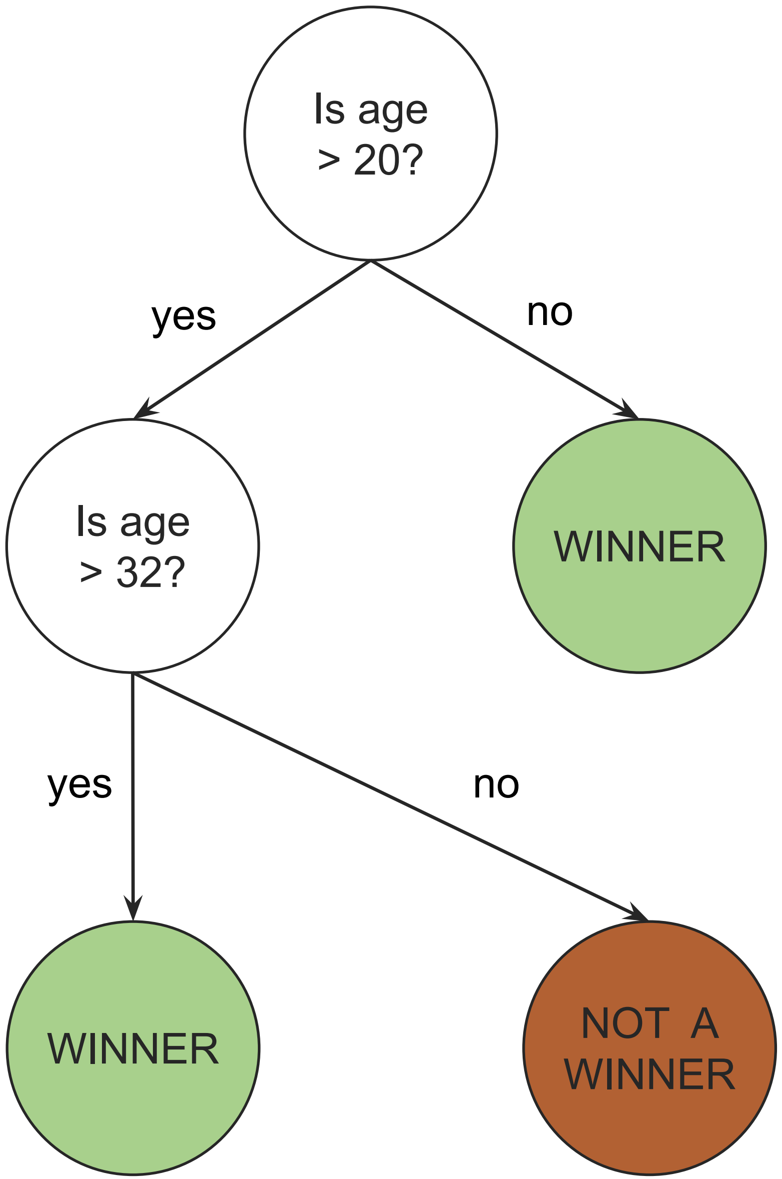
Example Data



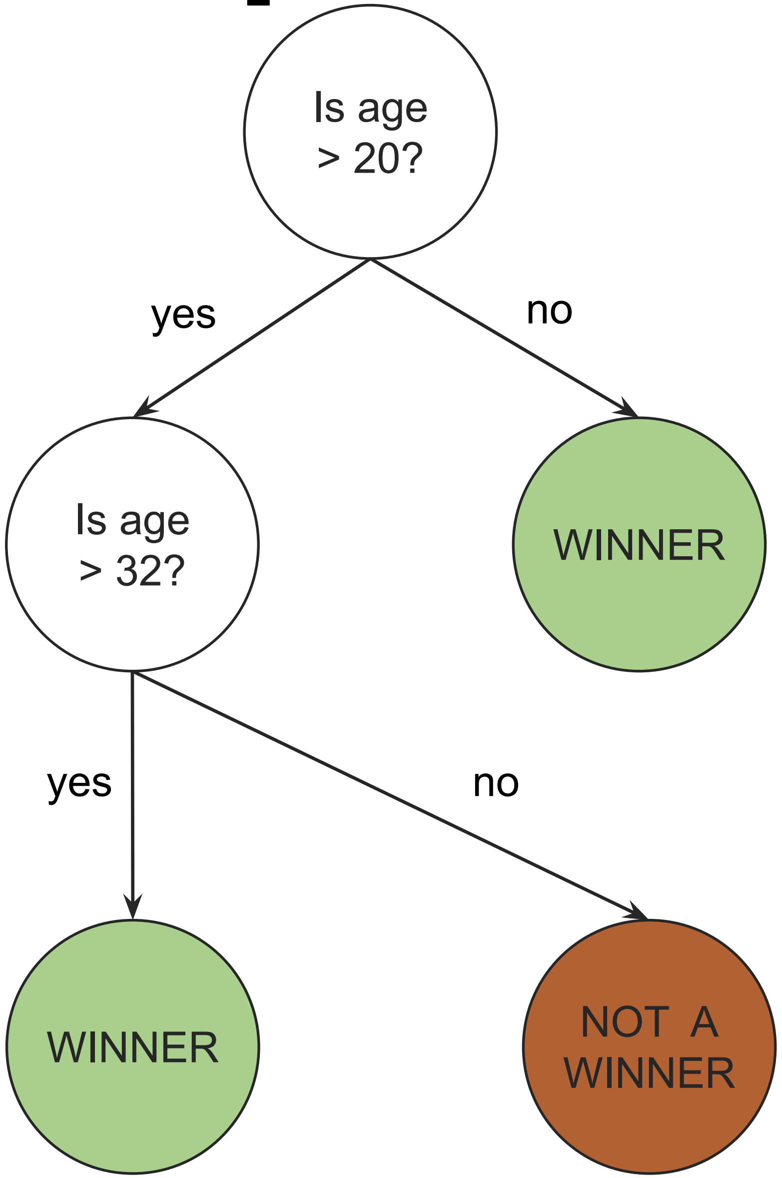


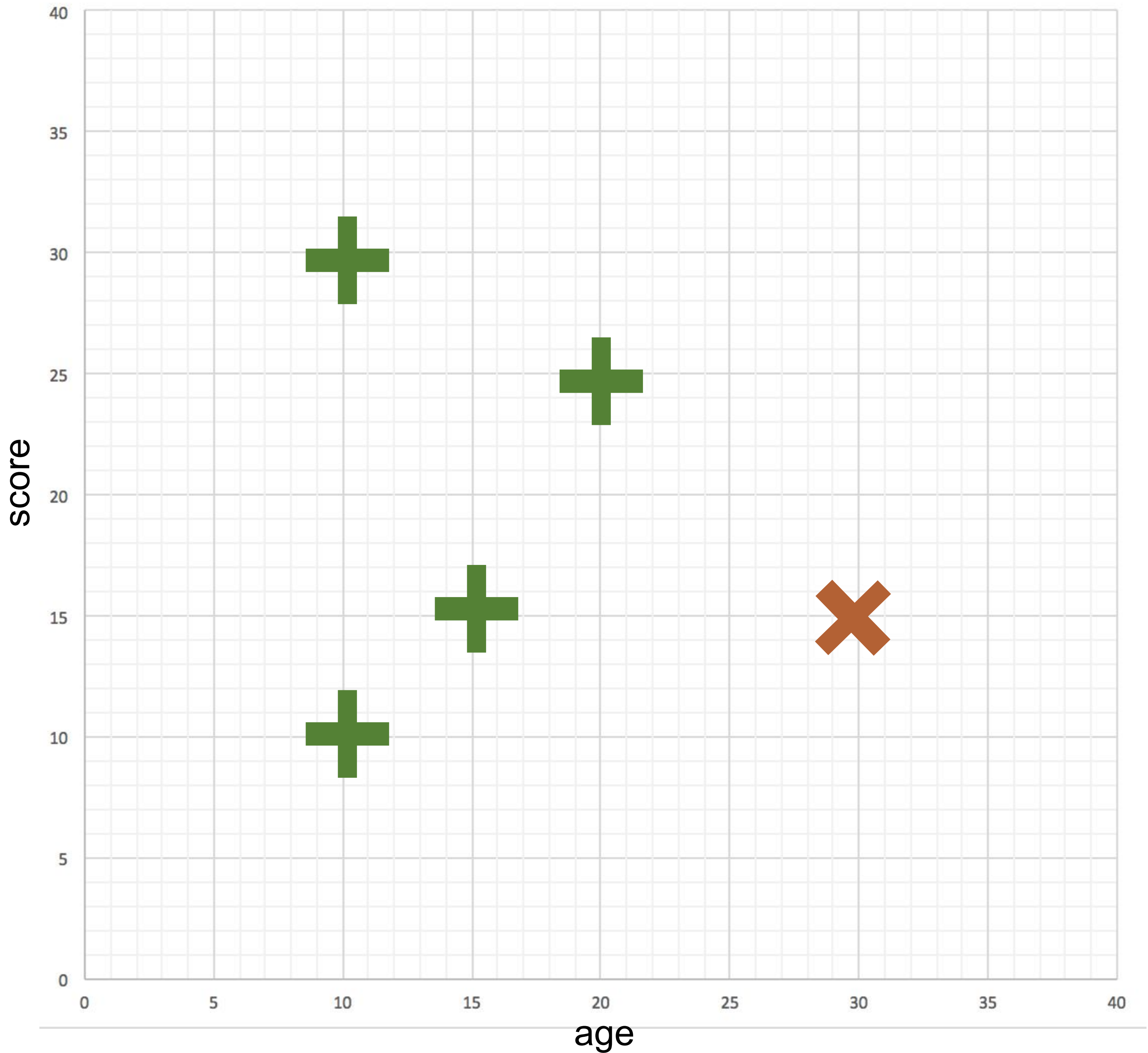
Example Data

X (features)		y (label)
Age	**Score**	**Winner**
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

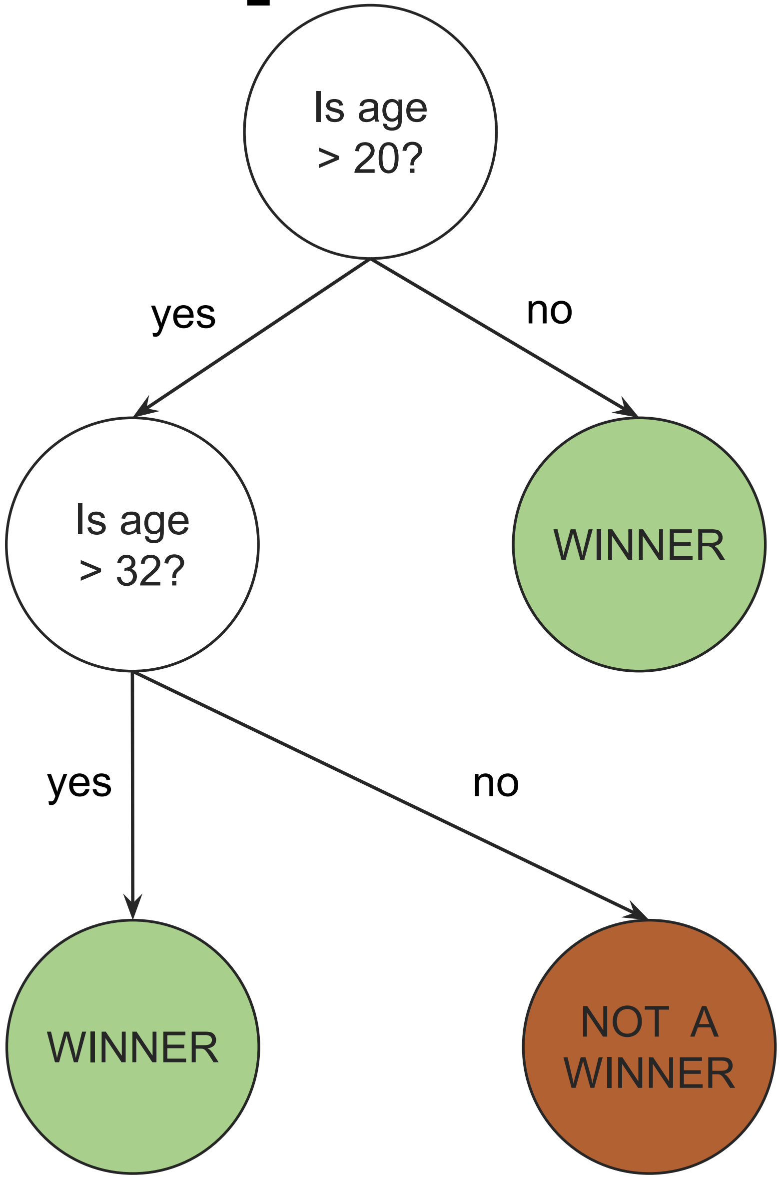


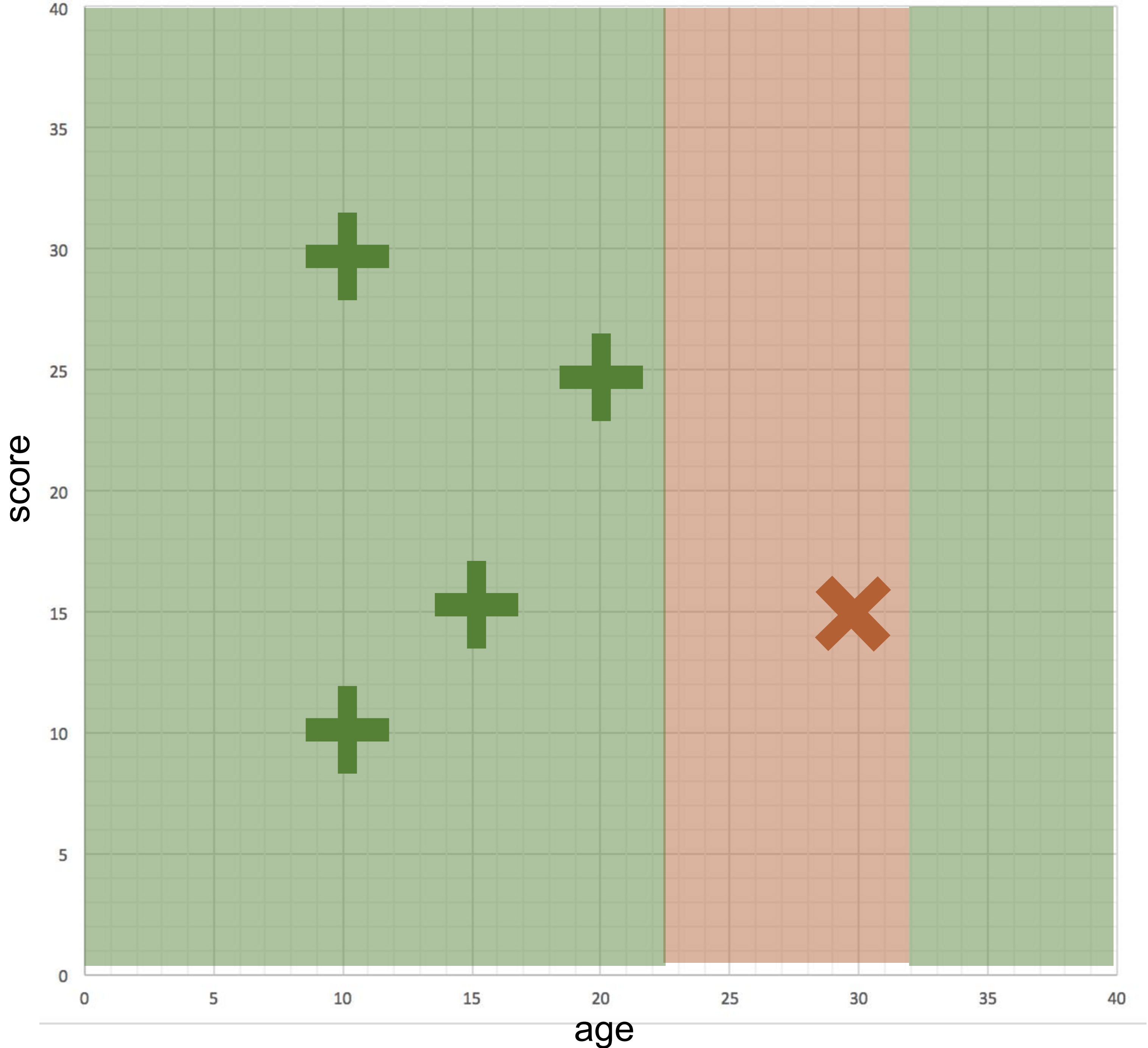
Example Data





Example Data



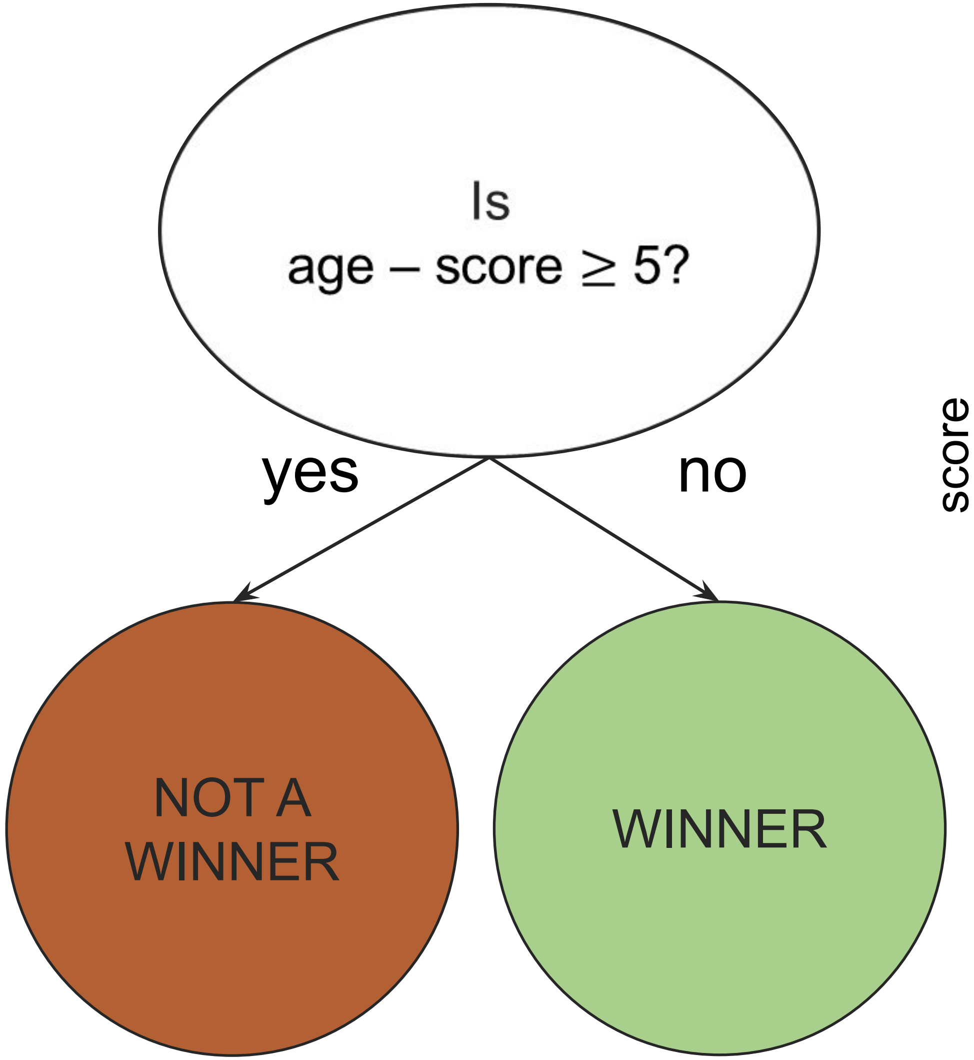


Decision Trees

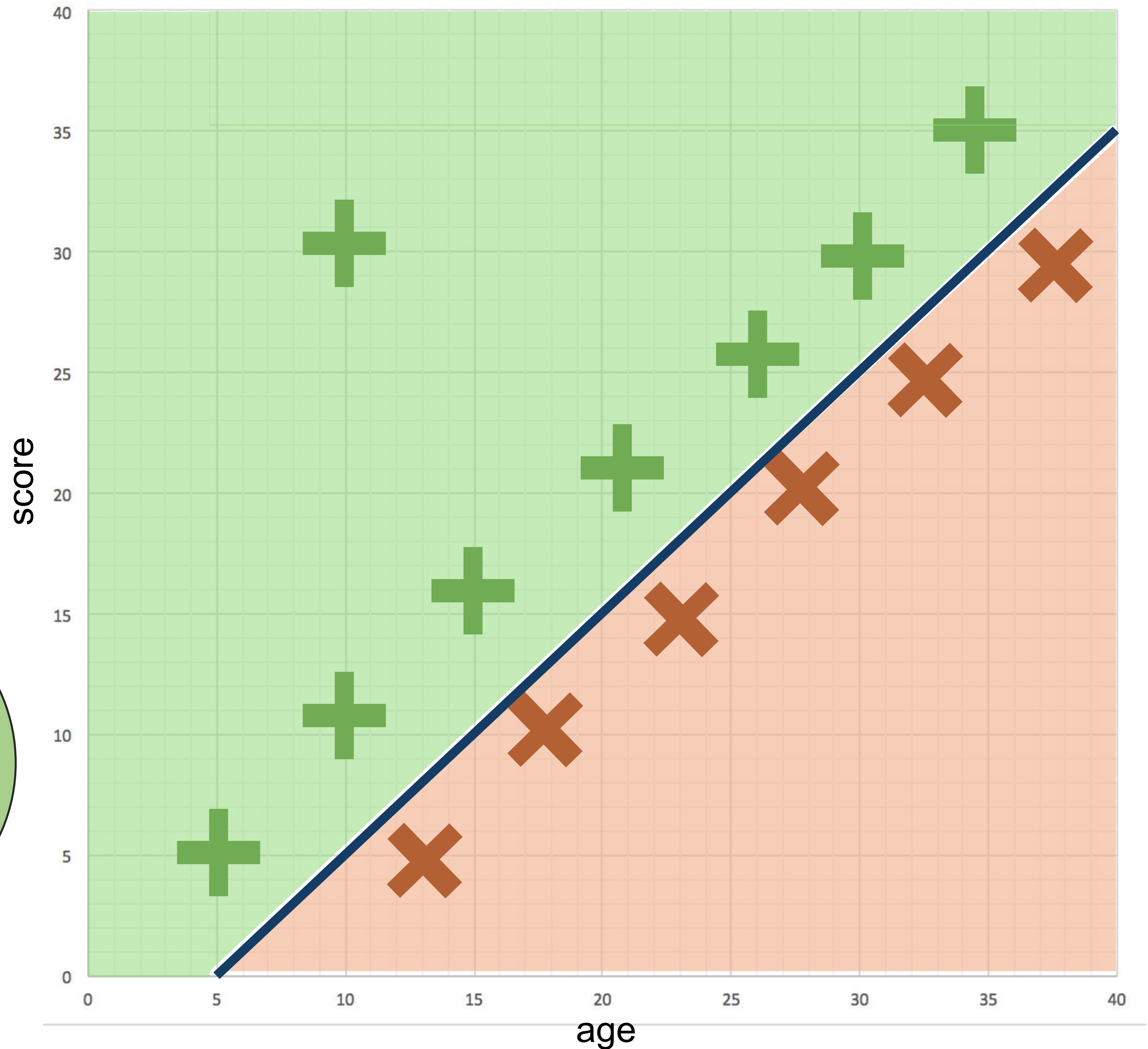
- Only yes/no questions are allowed.
- Each question **must allows follow the format of:**

Is [feature] equal to / greater than / less than [value]?

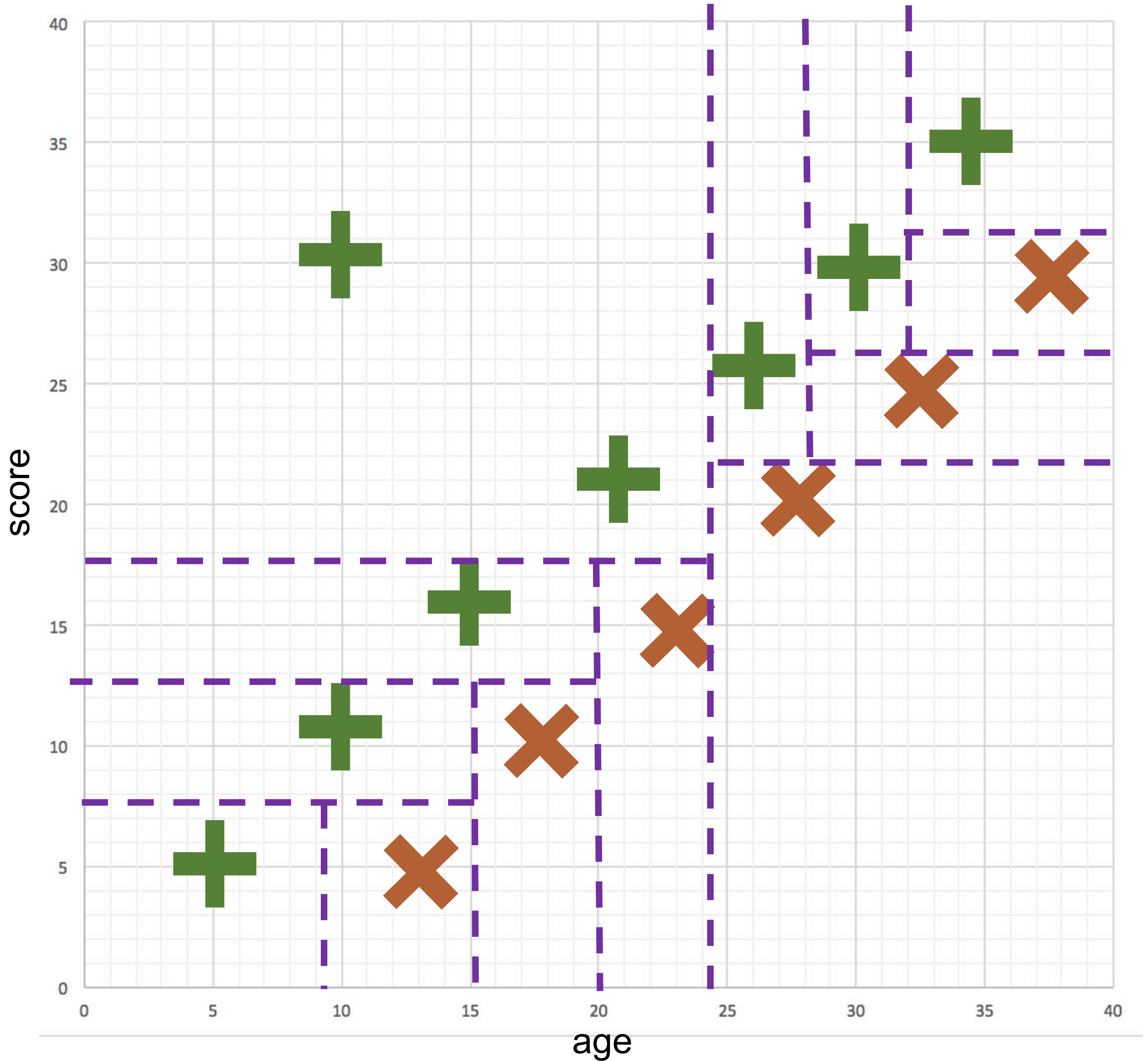
Not Allowed



If you want something like this, use logistic regression!



For this data, decision trees will be forced to do this instead.



Allowable Questions

Weather	Temp.	Suspended
sunny	32	no
rainy	32	no
rainy	20	yes
sunny	35	no
snowy	24	yes

- For categorical variables, equality with each possible value can be asked.
- For continuous variables, comparison with every value present in the data can be asked (just choose one of $\geq, \leq, >, <$)

Allowable Questions

Weather	Temp.	Suspended
sunny	32	no
rainy	32	no
rainy	20	yes
sunny	35	no
snowy	24	yes

Is the weather sunny?
Is the weather rainy?
Is the weather snowy?

Allowable Questions

Weather	Temp.	Suspended
sunny	32	no
rainy	32	no
rainy	20	yes
sunny	35	no
snowy	24	yes

Is the weather sunny?
Is the weather rainy?
Is the weather snowy?

Is the temperature > 35?
Is the temperature > 32?
Is the temperature > 24?
Is the temperature > 20?

Allowable Questions

Weather	Temp.	Suspended
sunny	32	no
rainy	32	no
rainy	20	yes
sunny	35	no
snowy	24	yes

Is the weather sunny?
Is the weather rainy?
Is the weather snowy?

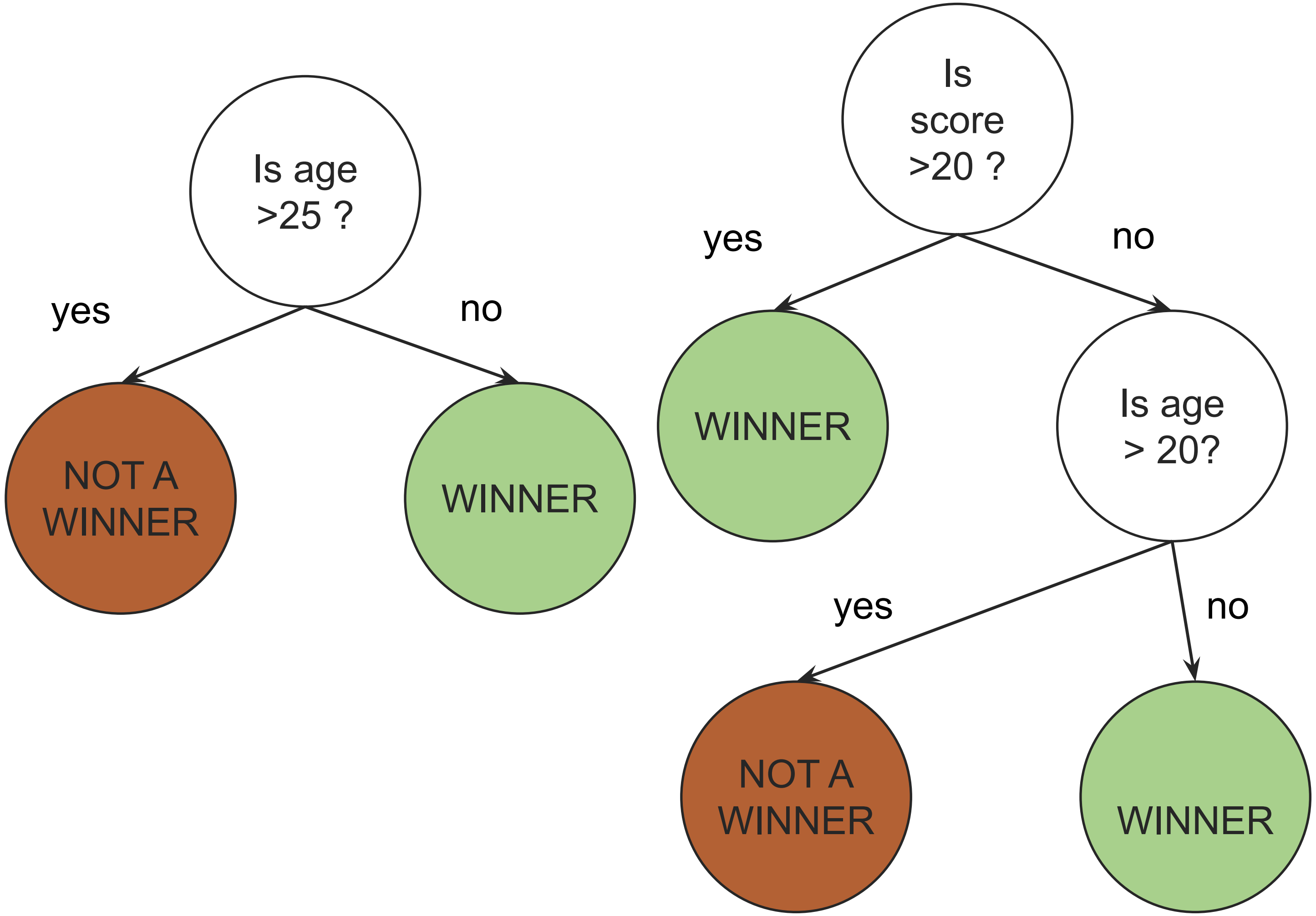
Is the temperature > 35?
Is the temperature > 32?
Is the temperature > 24?
Is the temperature > 20?

- Number of possible questions / splits is affected by the training data

Better Tree?

X (features) y (label)

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

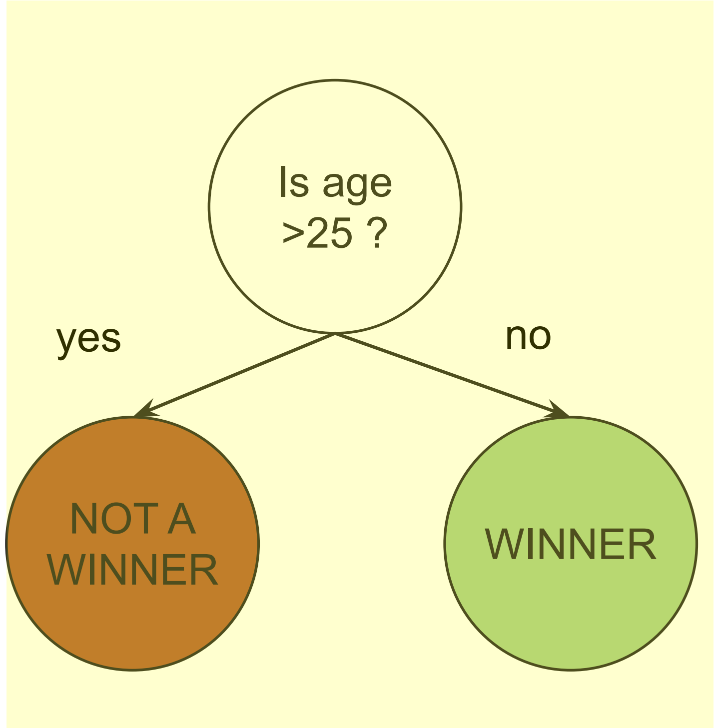


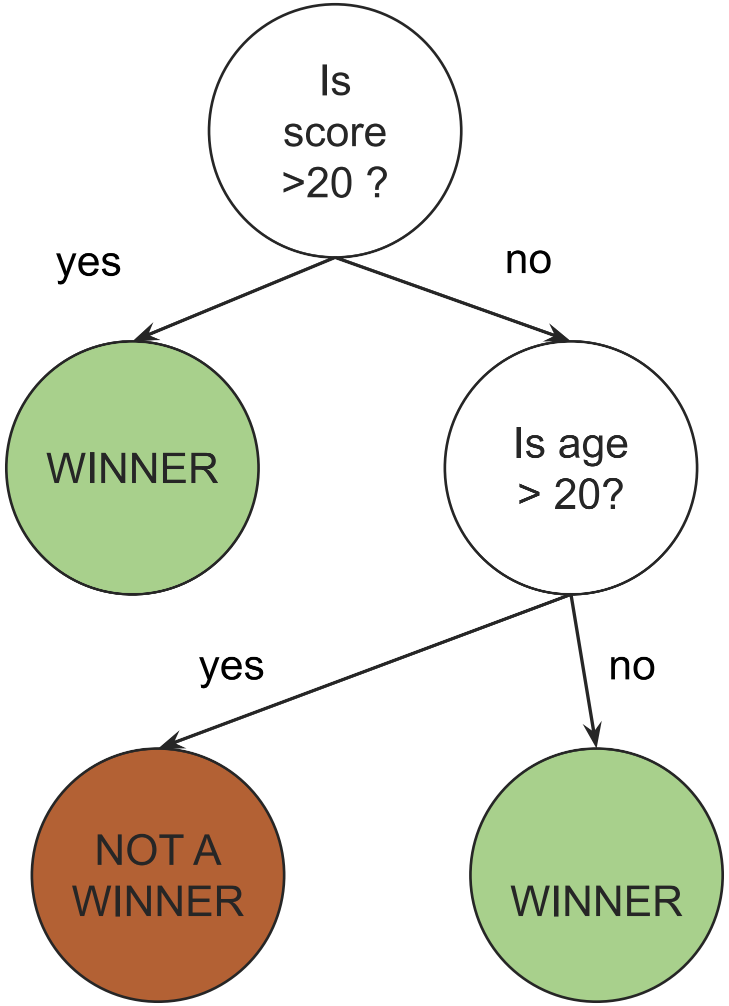
Both are valid trees, but which one is better? Why?

Better Tree?

X (features) | y (label)

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner





Tree on the left is better! Lesser memory for the same performance.
How do we make good trees?

Key Idea

X (features)		y (label)
Age	**Score**	**Winner**
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

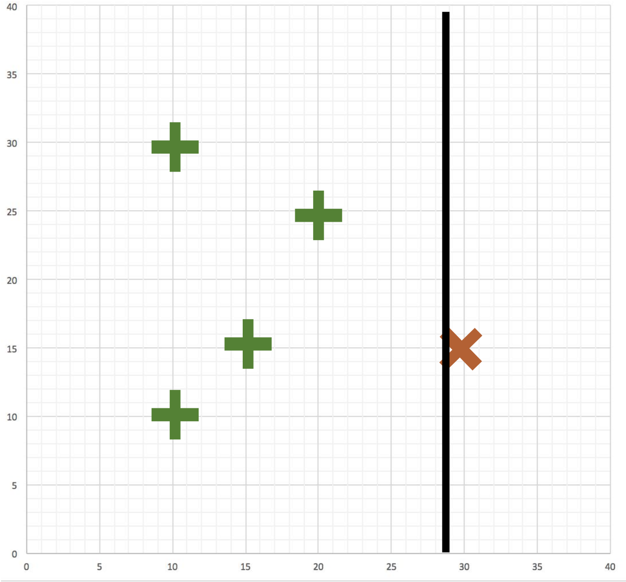
- **To make a better tree, we need to ask good questions.**

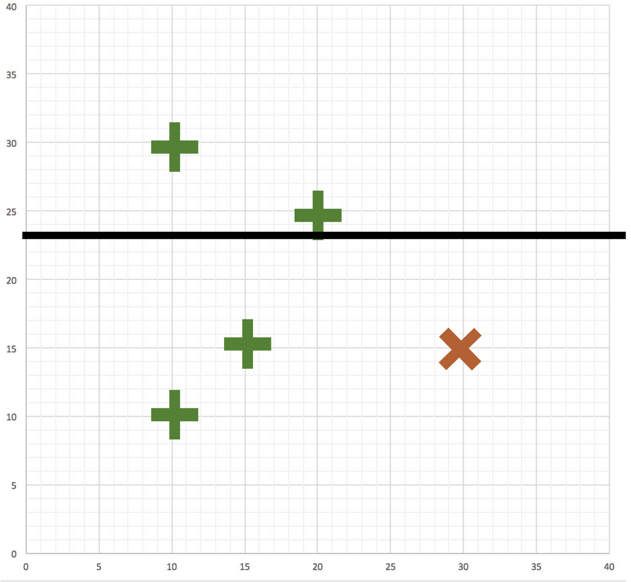
- What makes a question good?
  - How do we quantify this mathematically?

Key Idea

Age	Score	Winner
10	30	winner
10	10	winner
15	15	winner
20	25	winner
30	15	not a winner

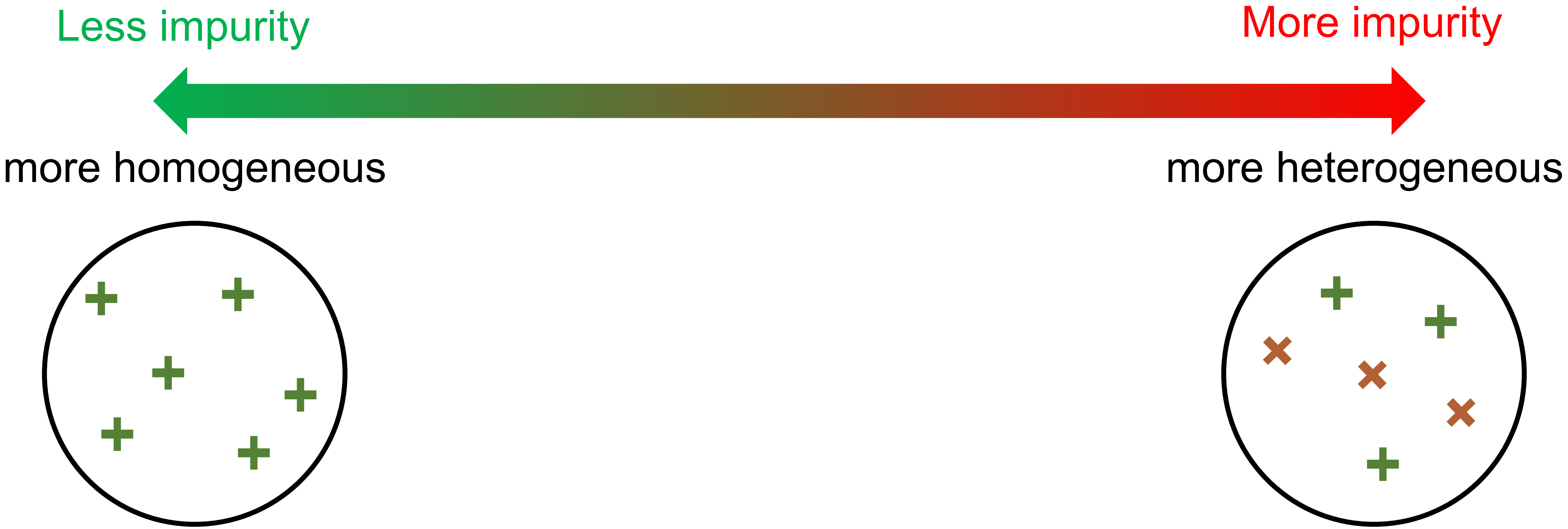
- Each question "splits" the data into two.
- What makes a good split?





Measure of Impurity

- Given a set of objects, “impurity” measures how homogeneous or heterogeneous the objects are.



Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

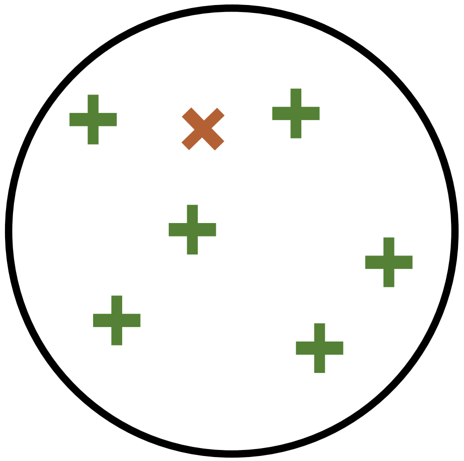
$$p_i = \frac{\text{number of class } i}{|S|}$$

Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

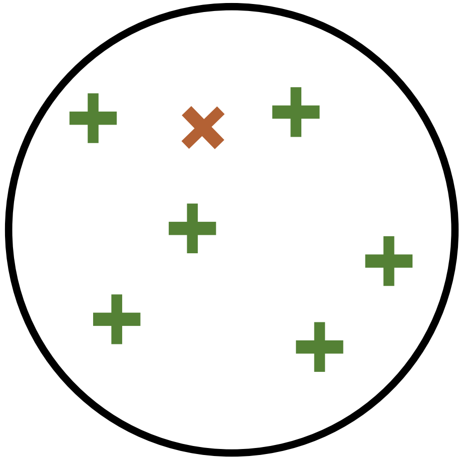


Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = (-7) \times \left(\left(\frac{6}{7}\right) \log_2 \left(\frac{6}{7}\right) + \left(\frac{1}{7}\right) \log_2 \left(\frac{1}{7}\right) \right)$$

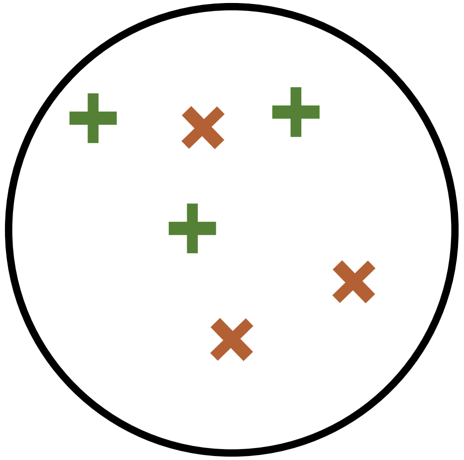
$$L(S) = 4.141709$$

Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

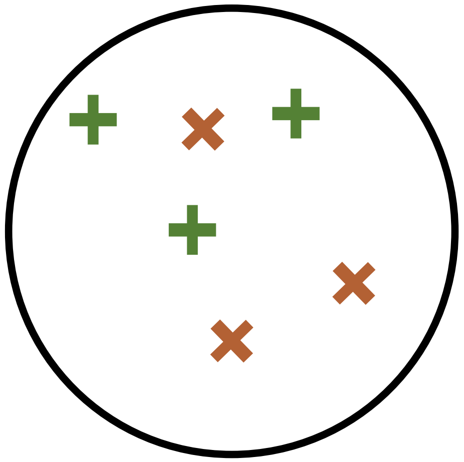


Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = (-6) \times \left(\left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right) + \left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right) \right)$$

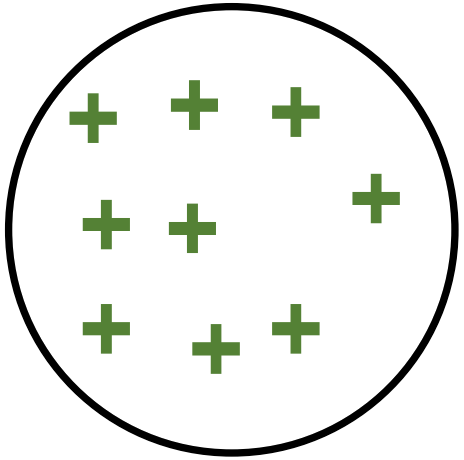
$$L(S) = 6$$

Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

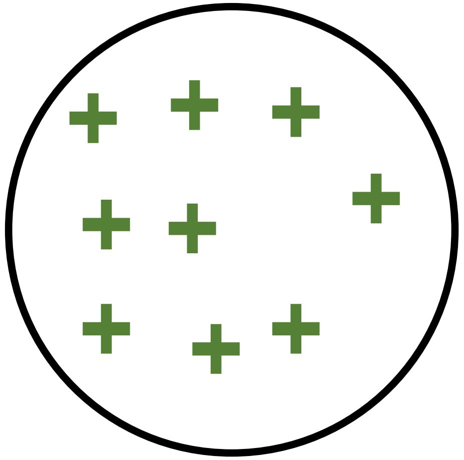


Common Measures of Impurity

- Shannon's Entropy

$$L(S) = -|S| \sum_{i}^{y} p_i \log_2(p_i)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = (-9) \times \left(\left(\frac{9}{9}\right) \log_2 \left(\frac{9}{9}\right) + \left(\frac{0}{9}\right) \log_2 \left(\frac{0}{9}\right) \right)$$

$$L(S) = 0$$

Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

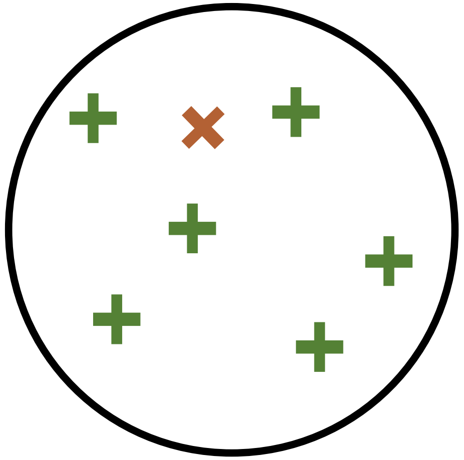
$$p_i = \frac{\text{number of class } i}{|S|}$$

Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

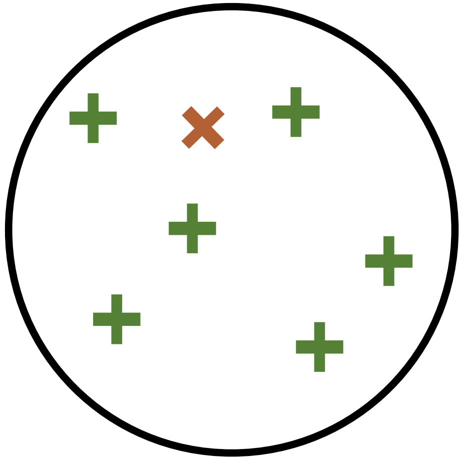


Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = 7 \times \left(1 - \left(\left(\frac{6}{7}\right)^2 + \left(\frac{1}{7}\right)^2\right)\right)$$

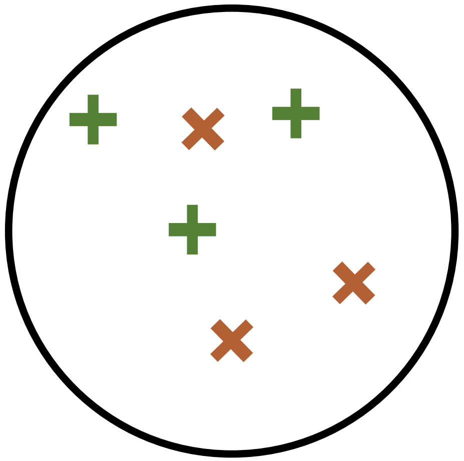
$$L(S) = 1.714$$

Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

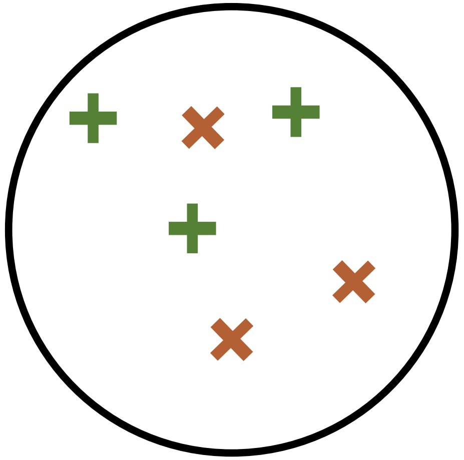


Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = 6 \times \left(1 - \left(\left(\frac{3}{6}\right)^2 + \left(\frac{3}{6}\right)^2\right)\right)$$

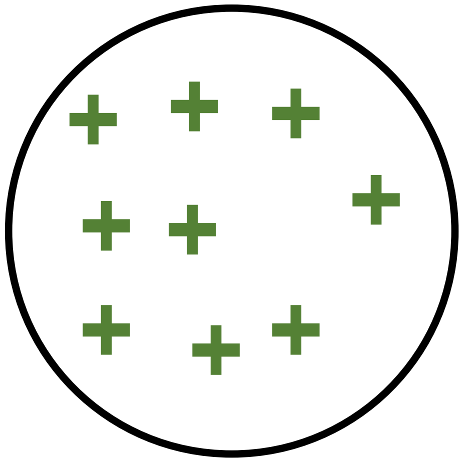
$$L(S) = 3$$

Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

$$p_i = \frac{\text{number of class } i}{|S|}$$

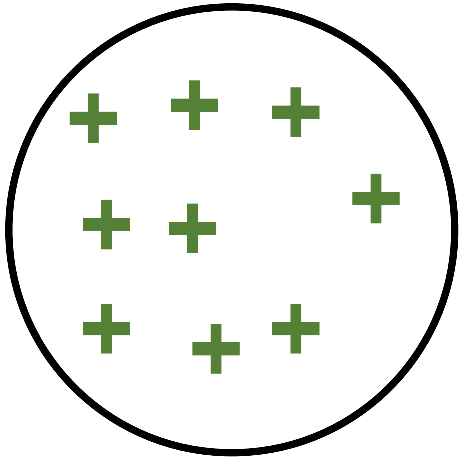


Common Measures of Impurity

- Gini Index

$$L(S) = |S| \times \left(1 - \sum_{i}^{y} p_i^2\right)$$

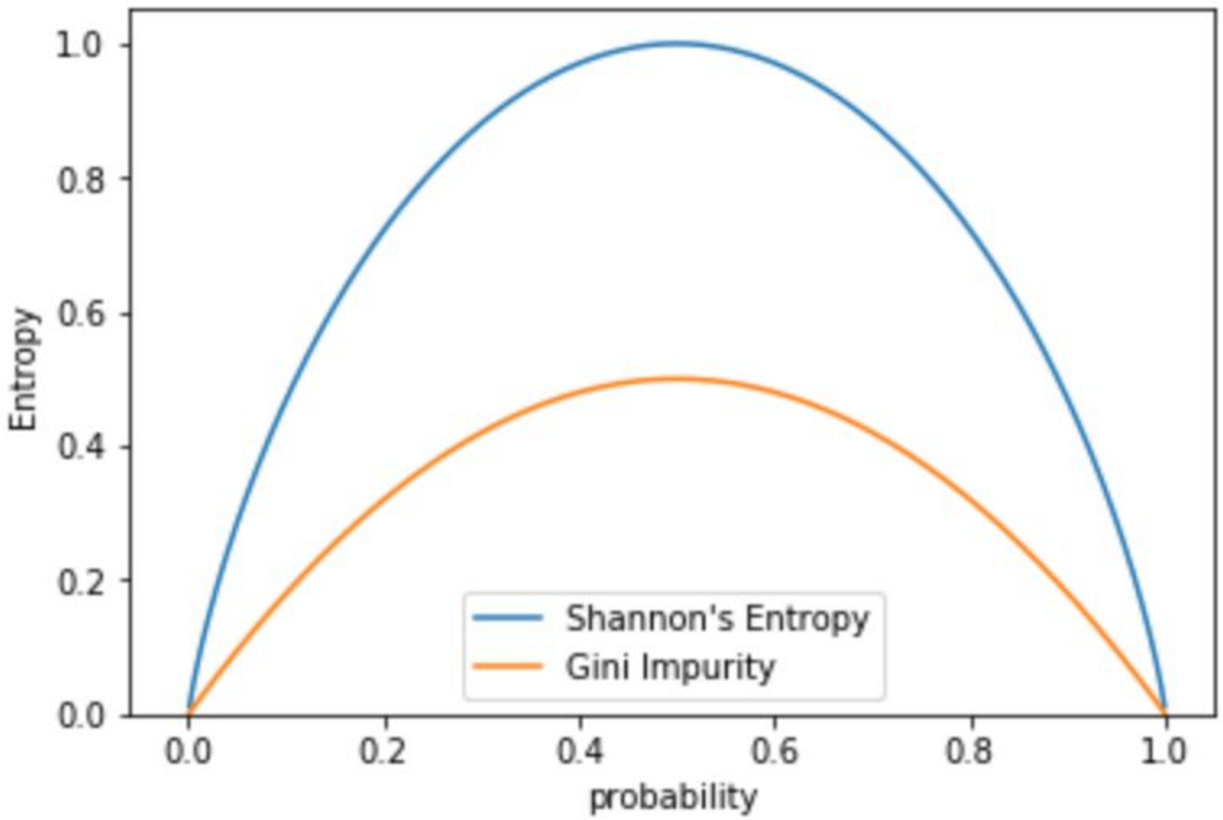
$$p_i = \frac{\text{number of class } i}{|S|}$$



$$L(S) = 9 \times \left(1 - \left(\left(\frac{9}{9}\right)^2 + \left(\frac{0}{9}\right)^2\right)\right)$$

$$L(S) = 0$$

Shannon Vs. Gini Index



- Note that the impurity still must be scaled with the number of instances in the set to give more importance to the impurity of larger groups.

Impurity for Continuous Labels

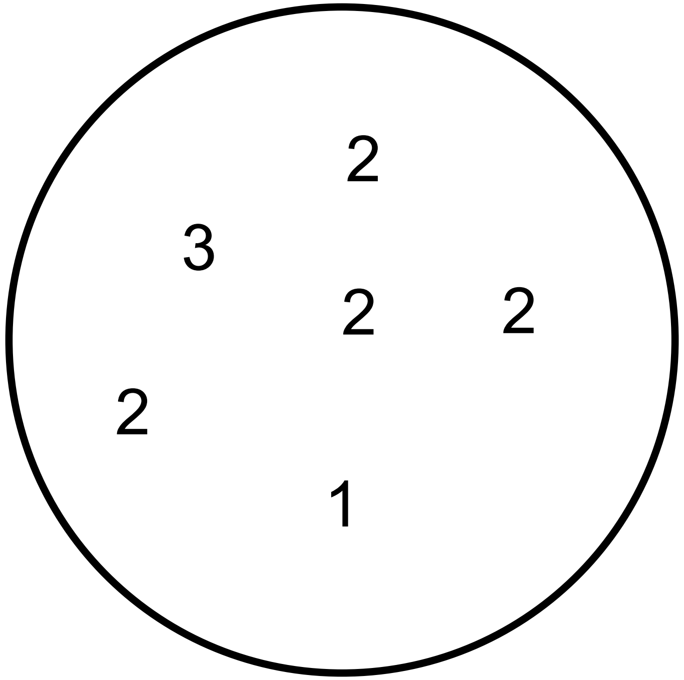
- Variance

$$L(S) = |S| \times \frac{\sum (x - \mu)^2}{n - 1}$$

Impurity for Continuous Labels

- Variance

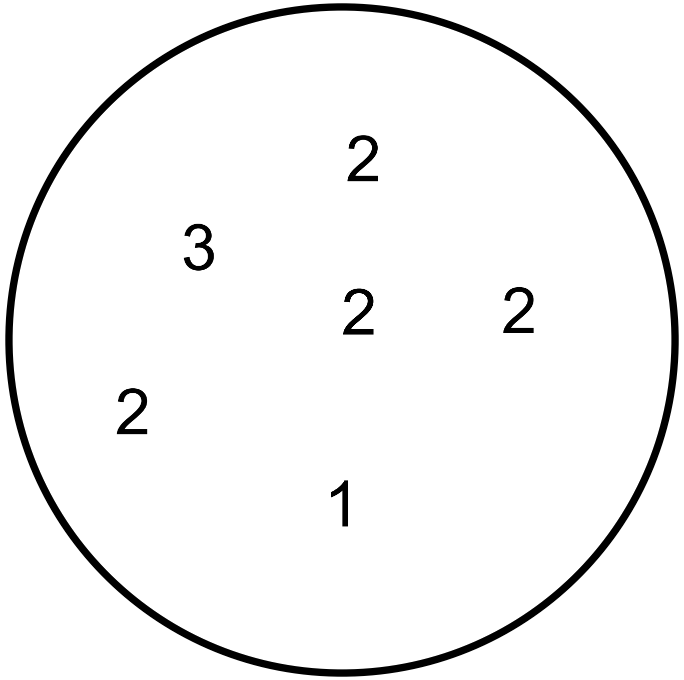
$$L(S) = |S| \times \frac{\sum (x-\mu)^2}{n-1}$$



Impurity for Continuous Labels

- Variance

$$L(S) = |S| \times \frac{\sum (x-\mu)^2}{n-1}$$

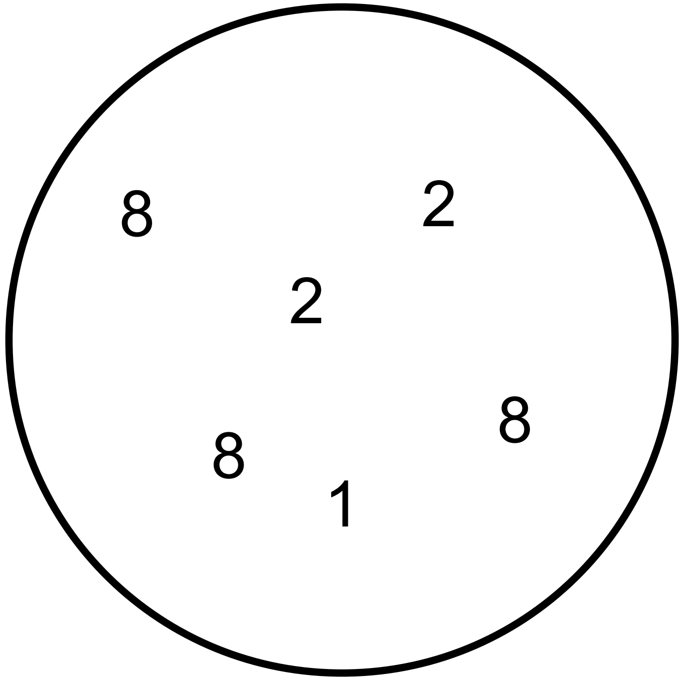


$$L(S) = 6 \times 0.4 = 2.4$$

Impurity for Continuous Labels

- Variance

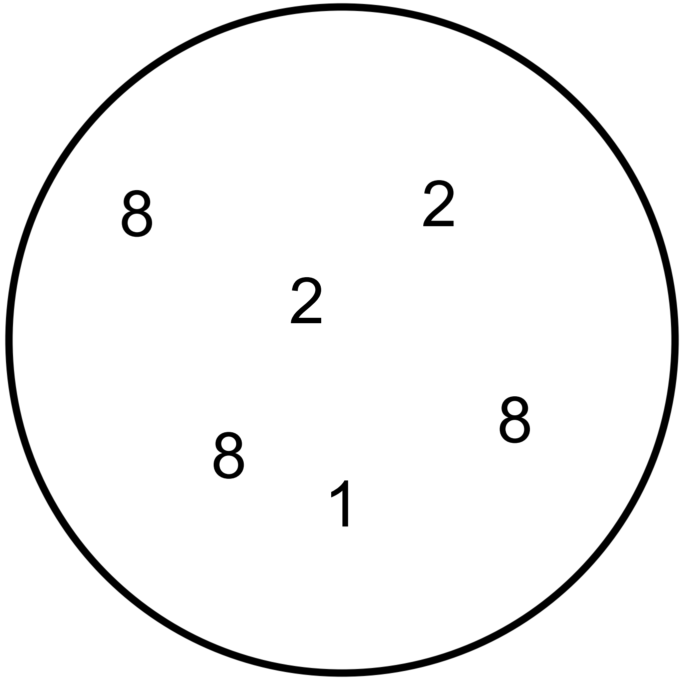
$$L(S) = |S| \times \frac{\sum (x-\mu)^2}{n-1}$$



Impurity for Continuous Labels

- Variance

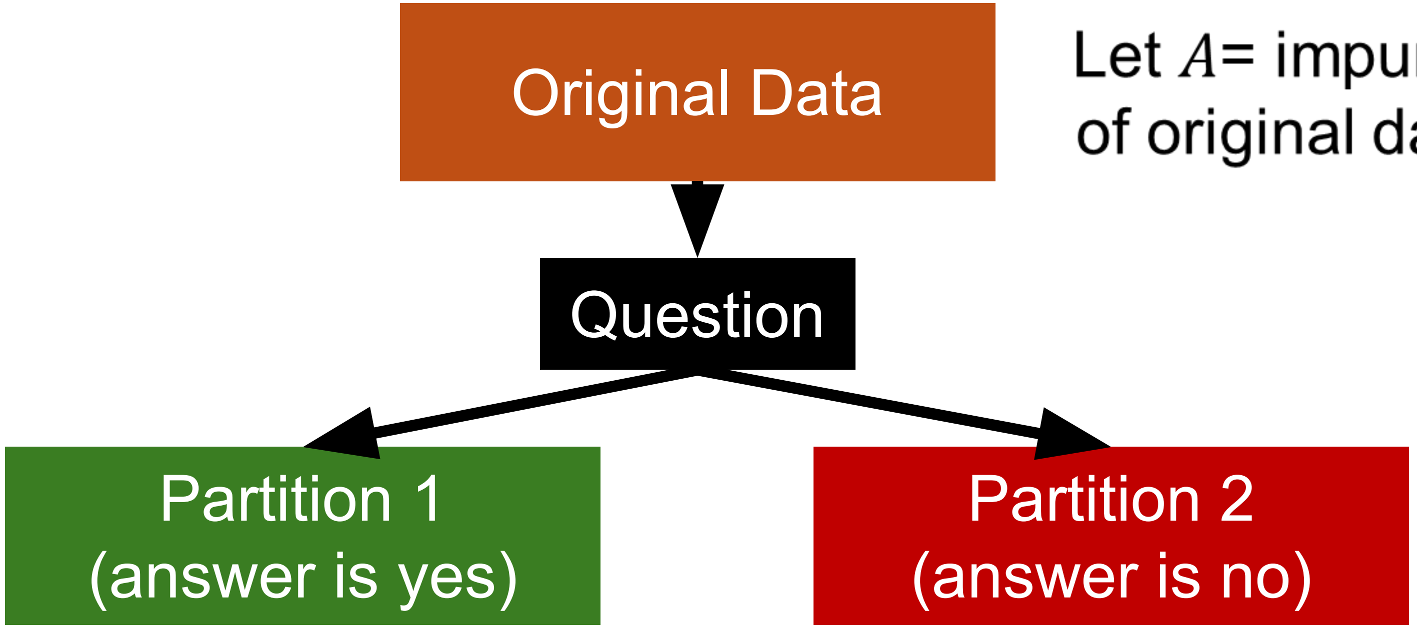
$$L(S) = |S| \times \frac{\sum (x-\mu)^2}{n-1}$$



$$L(S) = 6 \times 12.16667 = 73$$

Information Gain

- Amount of impurity that was lost when splitting the dataset through a question.



Let A= impurity of original data

$$\mathrm{IG} = A - (B + C)$$

Let B= impurity of partition 1

Let C= impurity of partition 2

Training a Decision Tree

function DTL (*data*)

if all *data* have the same class
    return class

No need to ask more questions

else

best ← Choose-Attribute-and-Threshold(*data*)

Choose the best question based on the highest IG

tree.left ← DTL(*data* matching best)

tree.right ← DTL(*data* not matching best)

return *tree*

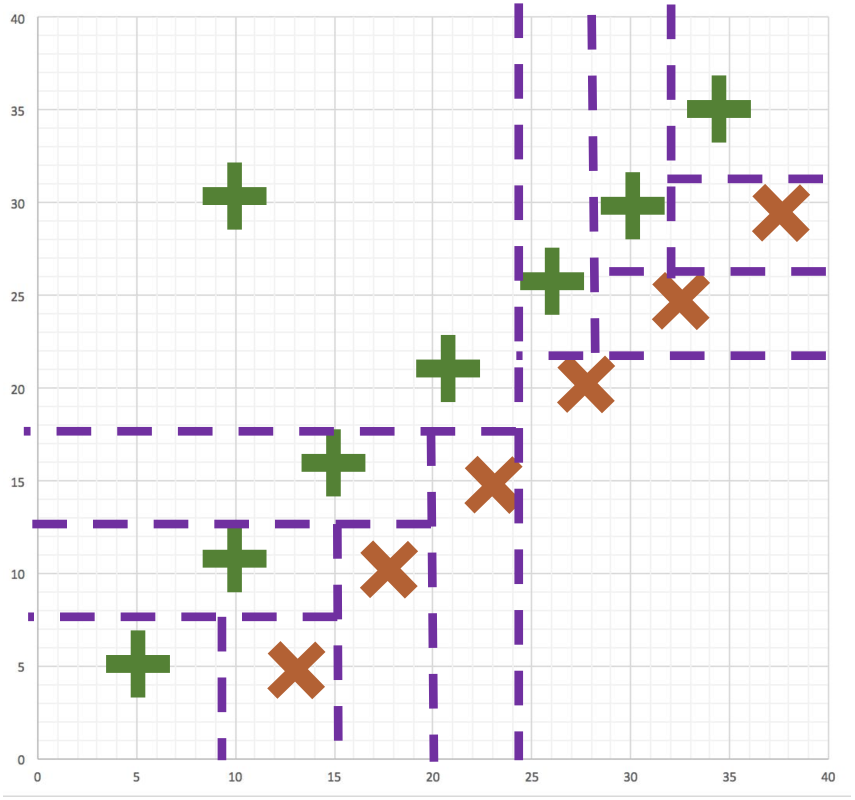
Left and right partition are also trees, so they can be generated recursively

DT are High Variance Models

- DT is designed to keep asking questions until all classes have been completely separated from one another.

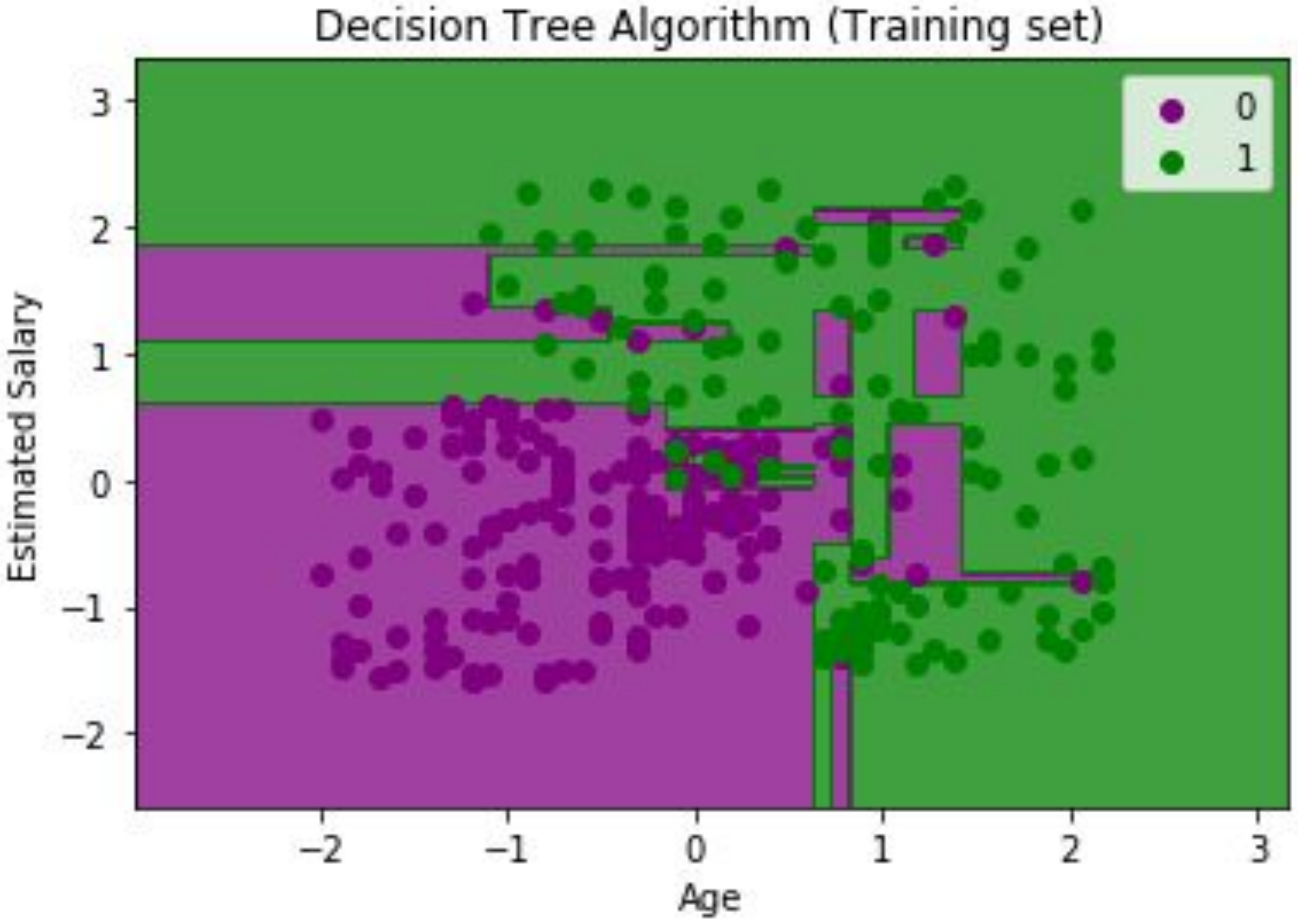
- Unless there are overlapping classes on the same point, **DT will always achieve 100% accuracy on the training set!**

- Is this always a good thing?



DT are High Variance Models

- Decision trees are prone to overfitting!

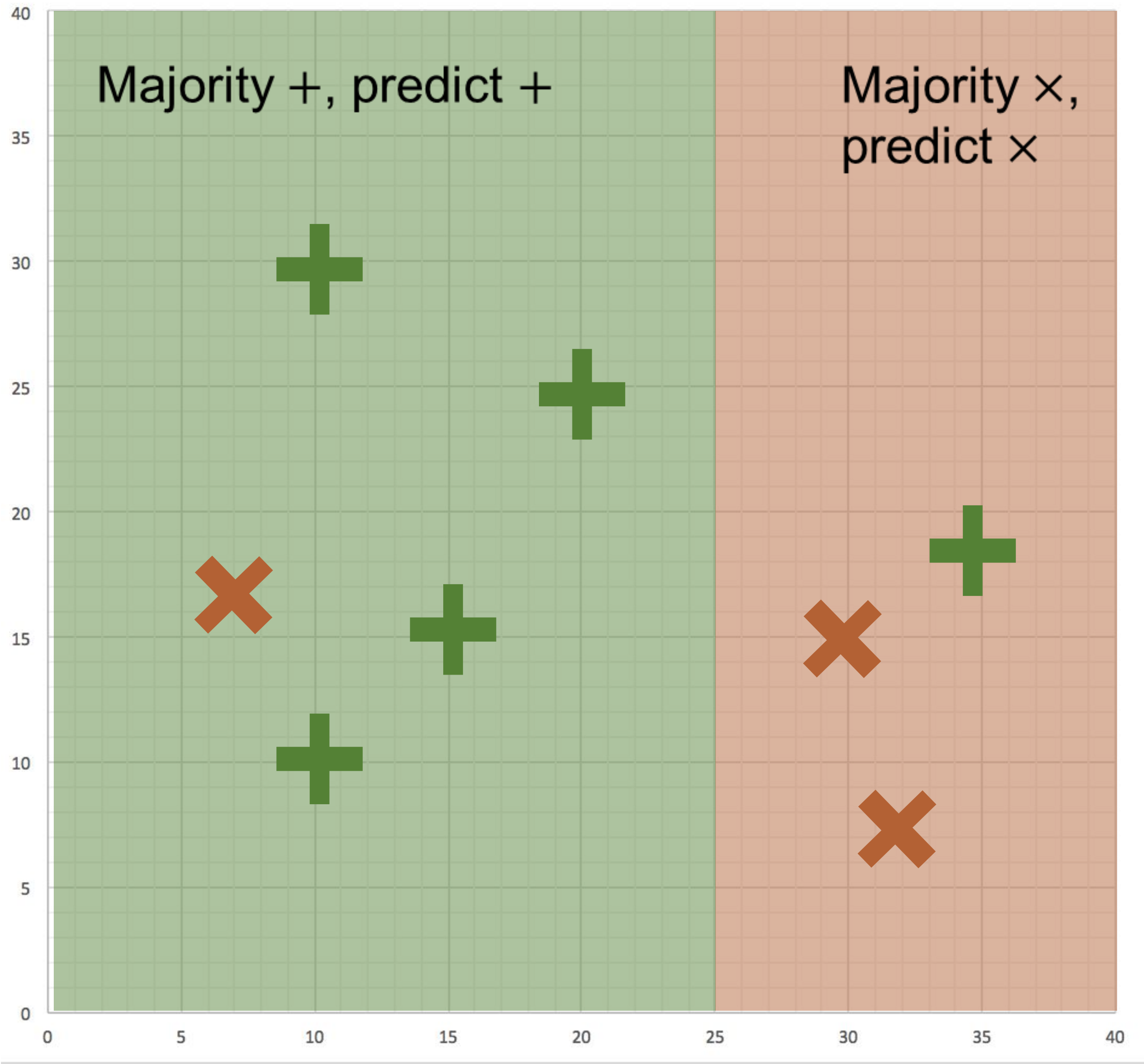


Regularization Techniques

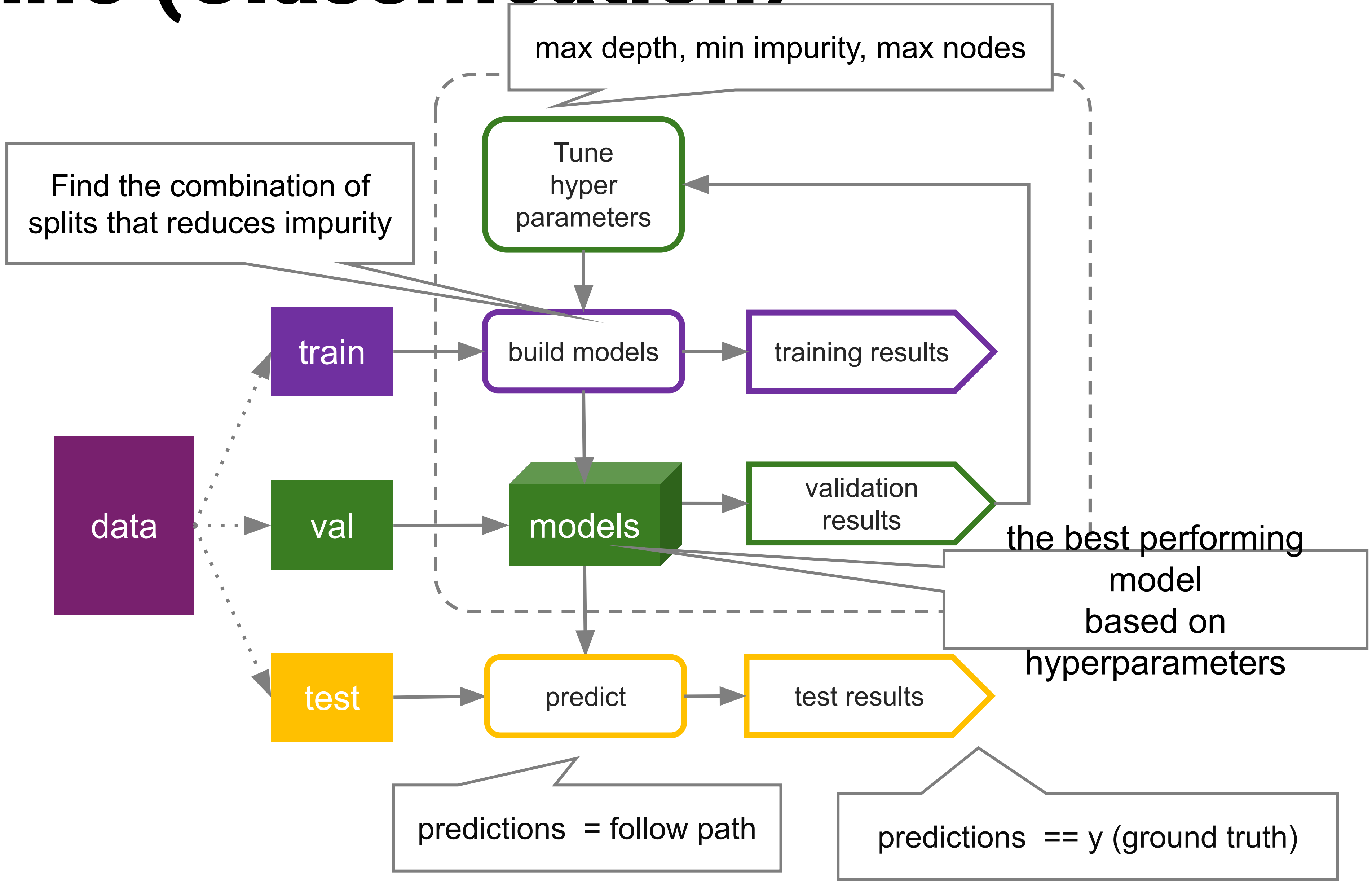
- **Stopping Criterion** (stop asking questions once a certain criteria is reached)
  - **Minimum batch size**
    - If number of data points to split < min
  - **Tree depth (height)**
    - If height > max
  - **Number of nodes**
    - If number of nodes > max
  - **Impurity reduction percentage**
    - If impurity < min, return mode/avg

Regularization Techniques

- When we stop asking questions, and the dataset is still not homogeneous, how do we make the prediction?
  - For classification: **pick the majority**
  - For regression: **get the average**



Pipeline (Classification)



Pipeline (Regression)