

---

---

# Classification

— Boston University CS 506 - Lance Galletti —

---

---

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

**CLASS**

The diagram illustrates the components of a classification problem. A table with four rows and three columns is shown. The first two columns, 'age' and 'Tumor size', are highlighted in yellow and labeled as 'PREDICTORS / FEATURES / ATTRIBUTES' by an arrow pointing to them from a yellow box below. The third column, 'malignant?', is highlighted in orange and labeled as 'CLASS' by an arrow pointing to it from an orange box to its right. The data rows show that as age and tumor size increase, the likelihood of a malignant tumor also increases.

**PREDICTORS / FEATURES / ATTRIBUTES**

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

learn  
model

Model

$f : \text{age} \times \text{tumor size} \rightarrow \{\text{yes}, \text{no}\}$

# What is Classification?

age	Tumor size	malignant?
20	9	no
30	16	yes
40	18	no
50	28	yes

# What is Classification?

age	Tumor size	malignant?
20	9	no
40	18	no
30	16	yes
50	28	yes



# What is Classification?

age	Tumor size	malignant?
20	9	no
40	18	no
30	16	yes
50	28	yes

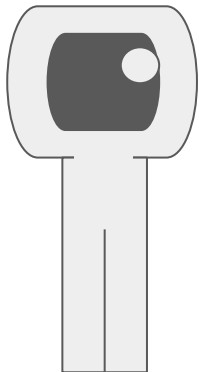
# What is Classification?

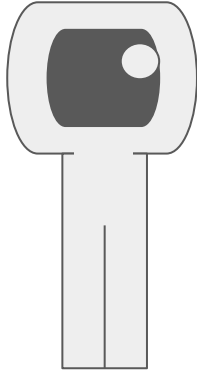
age	Tumor size	malignant?
20	9	no
40	18	no
30	16	yes
50	28	yes

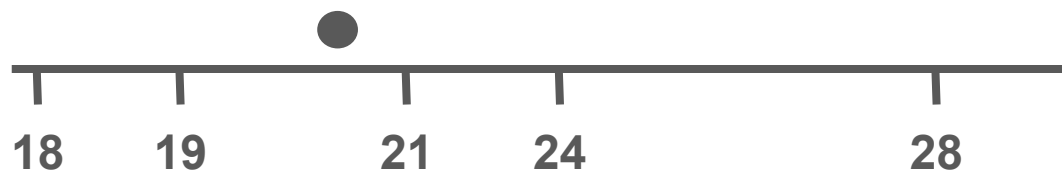
# What is Classification?

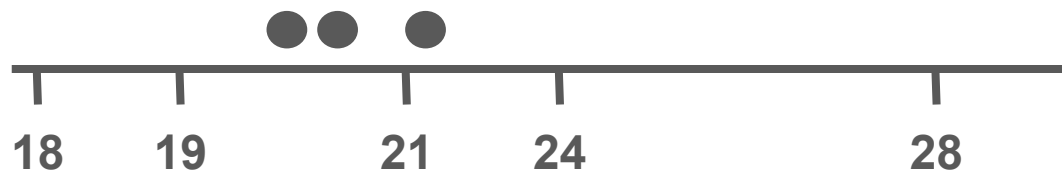
age	Tumor size	malignant?
20	9	no
40	18	no
30	16	yes
50	28	yes

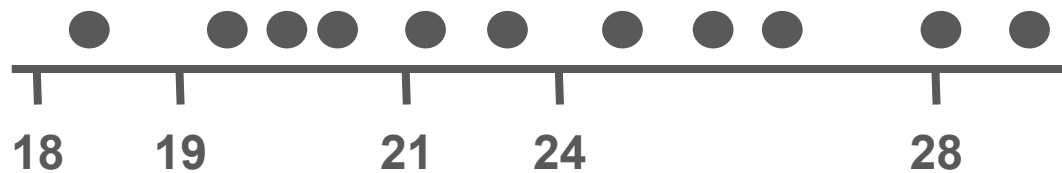
**What property of age and tumor size is unique to malignant tumors?**



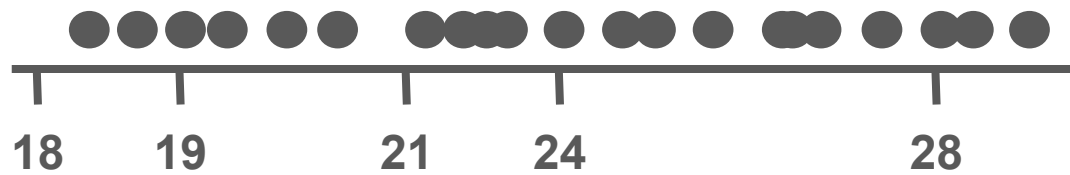


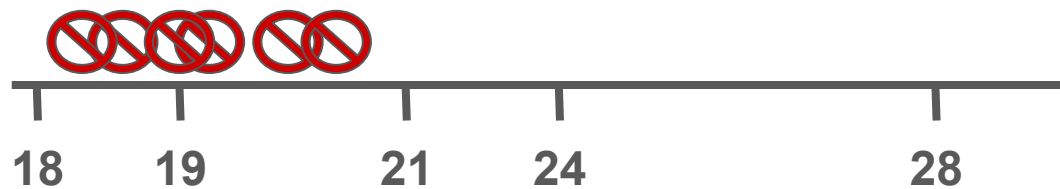


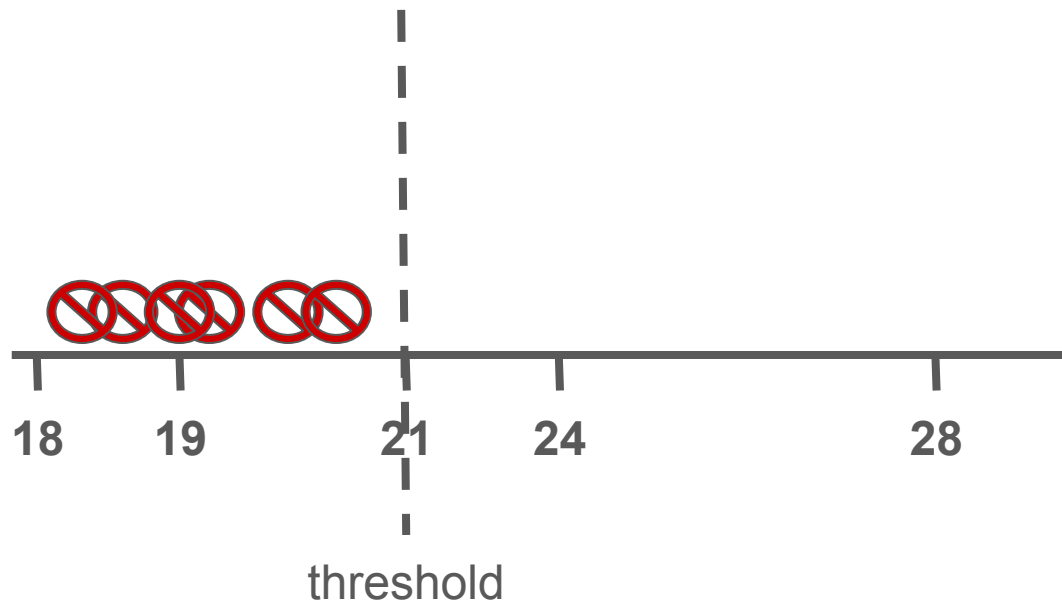












Sometimes there are **many** correct answers



Sometimes there are **many** correct answers



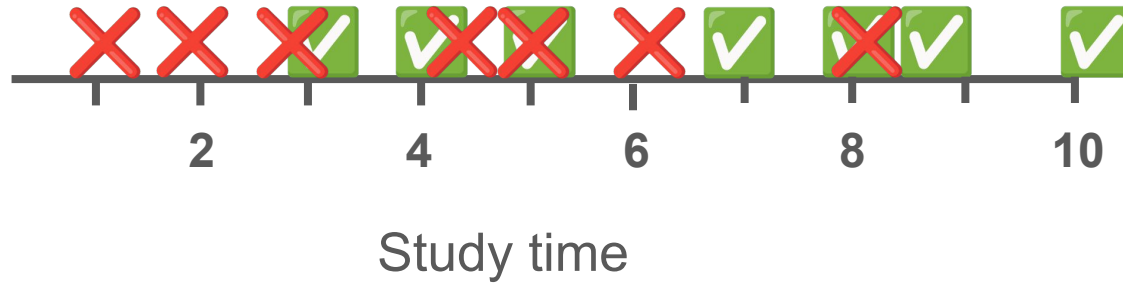
Sometimes there are **many** correct answers



Sometimes there are **many** correct answers

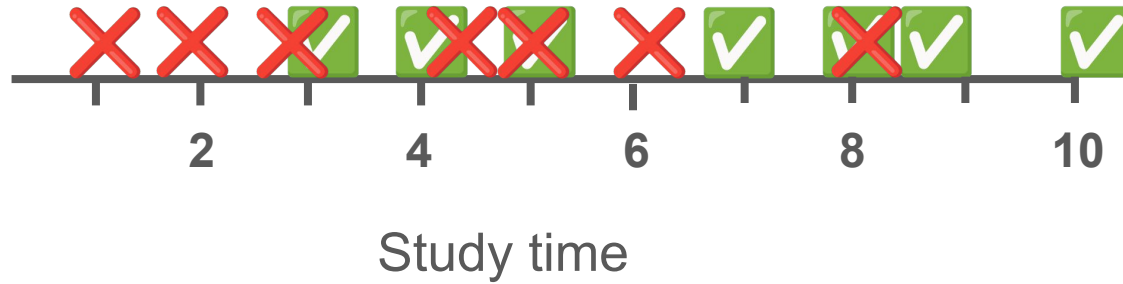


Sometimes there are **no** correct answers

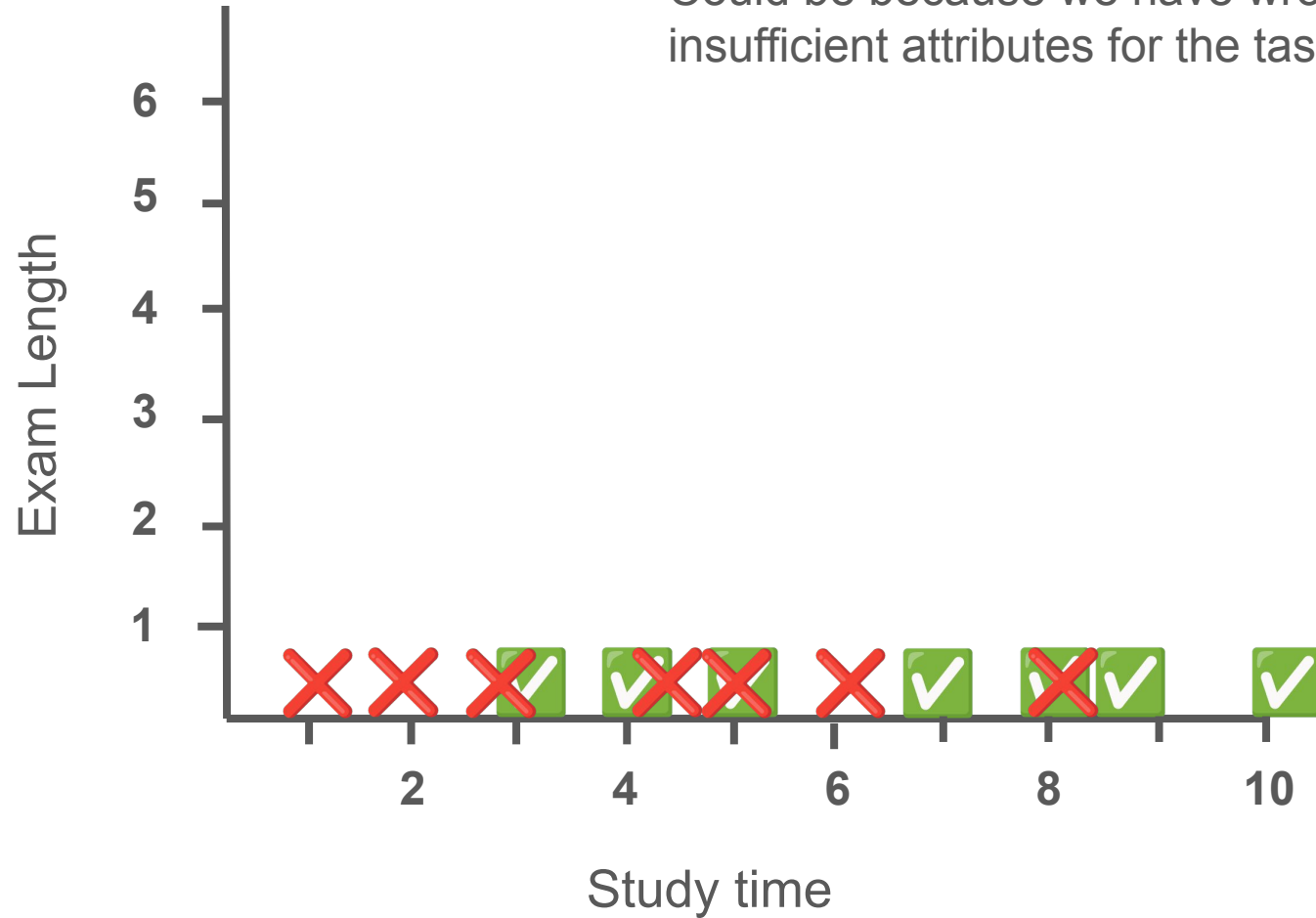


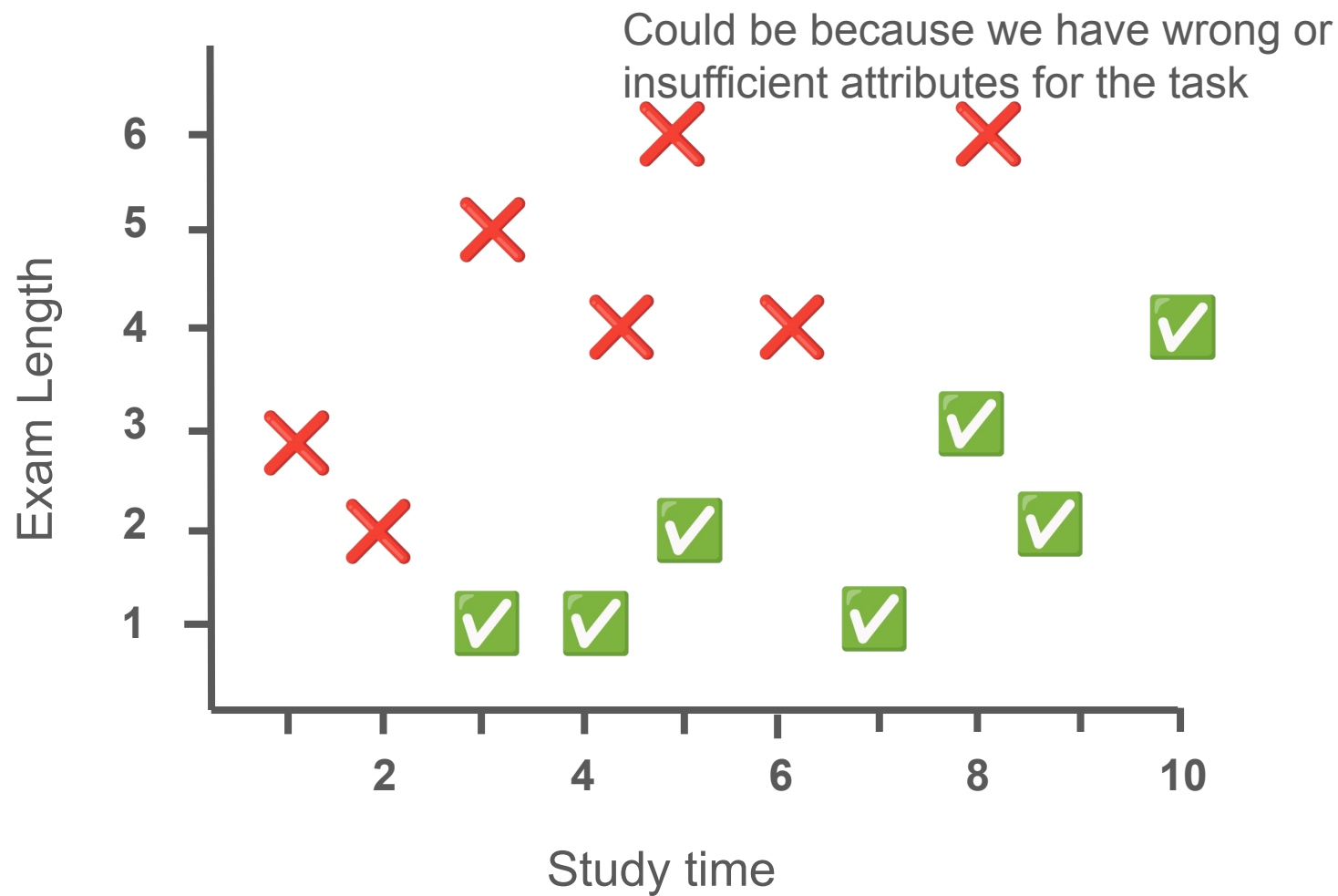


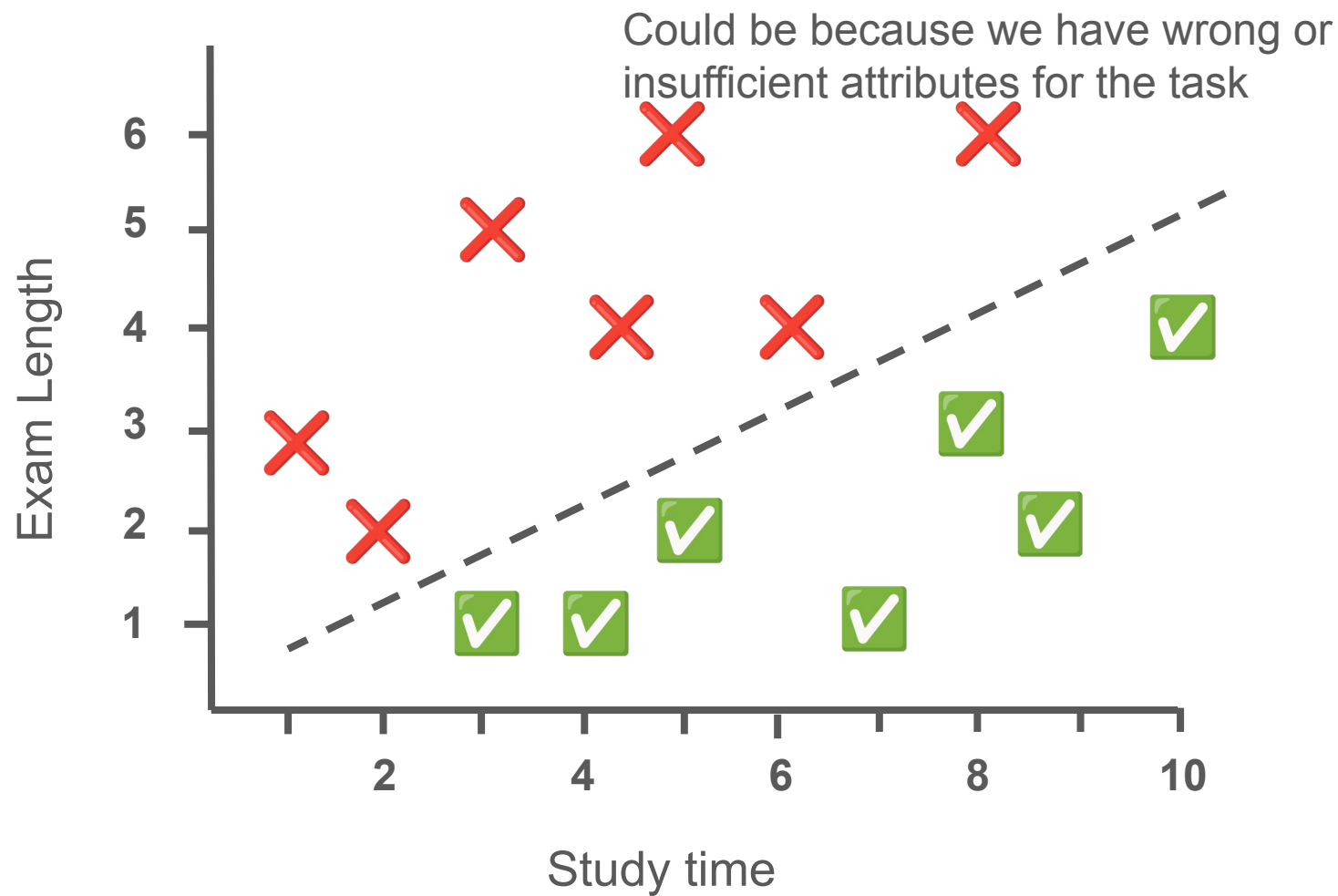
Could be because we have wrong or  
insufficient attributes for the task



Could be because we have wrong or  
insufficient attributes for the task

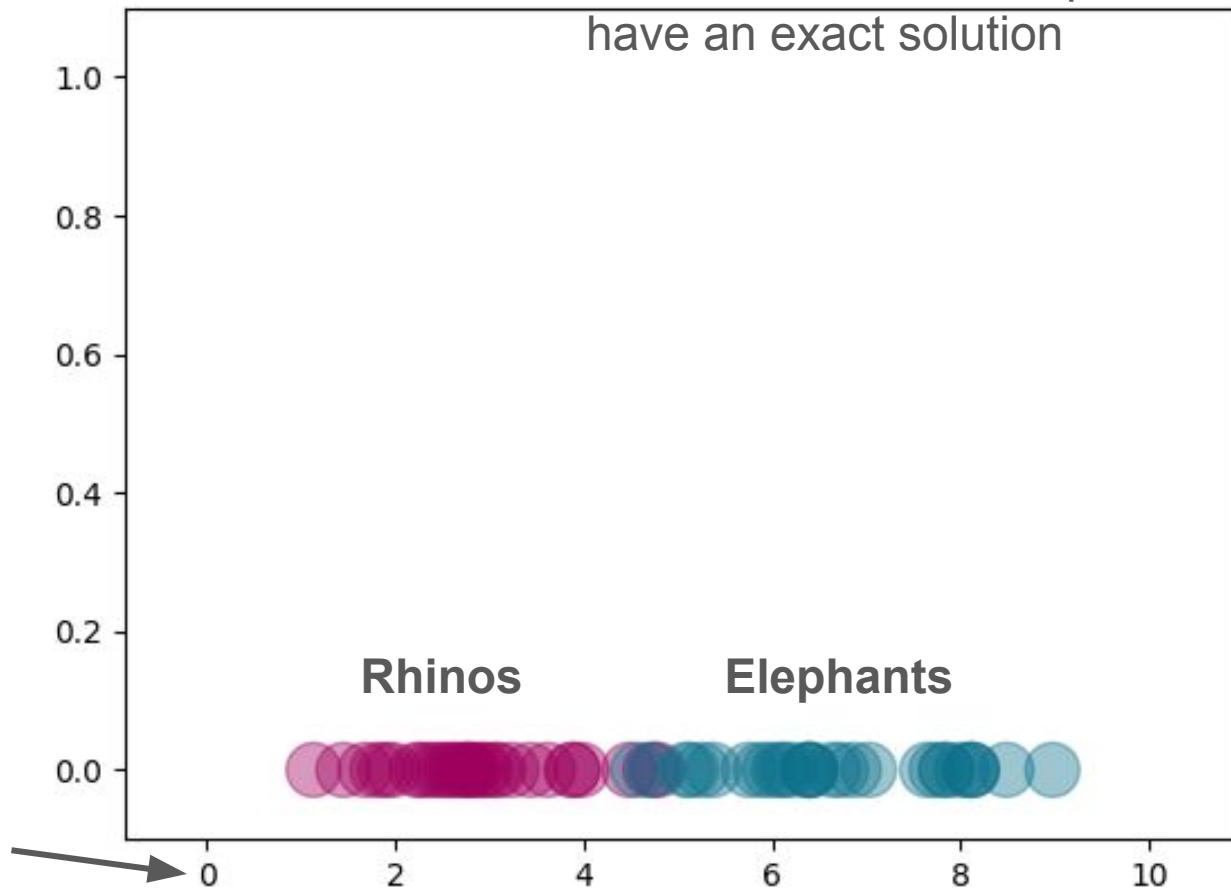




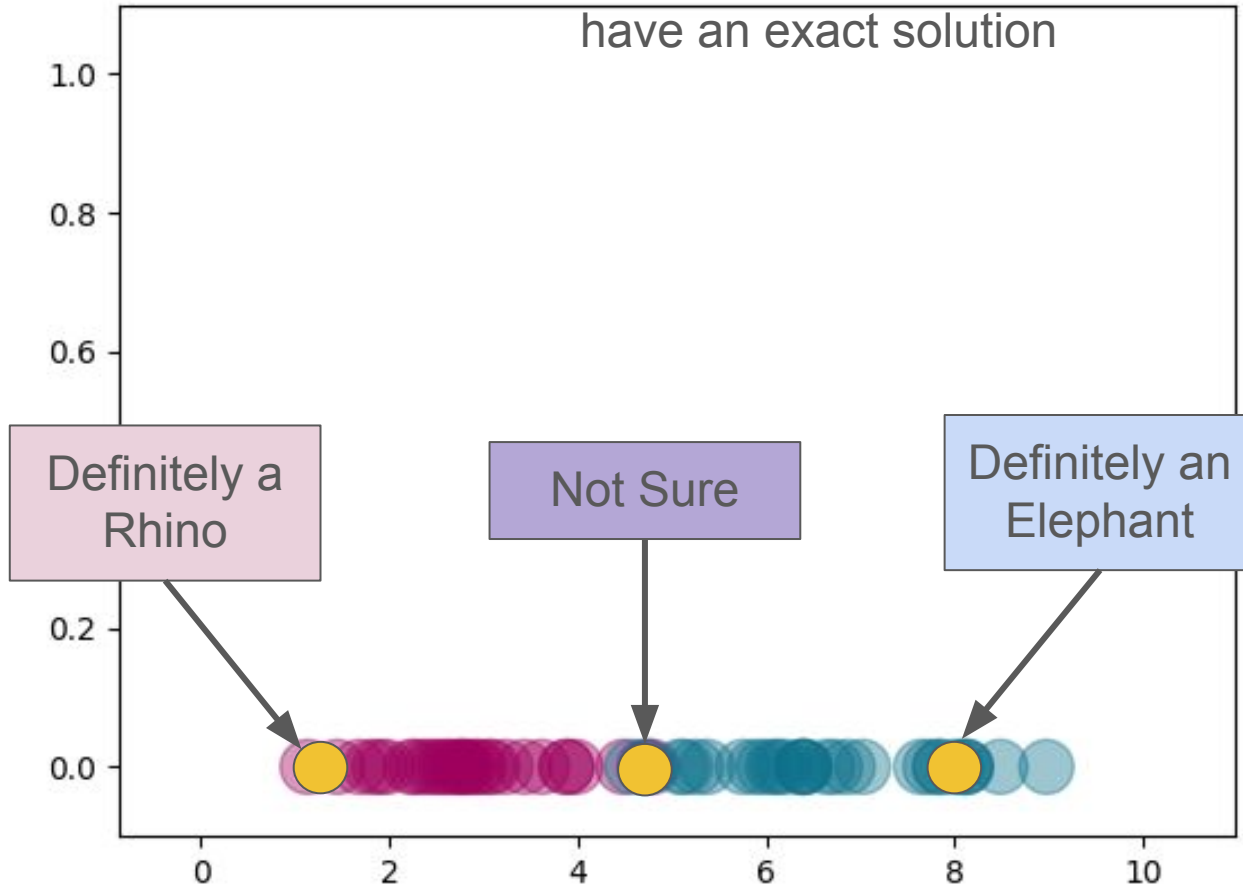


Could be because the problem just doesn't  
have an exact solution

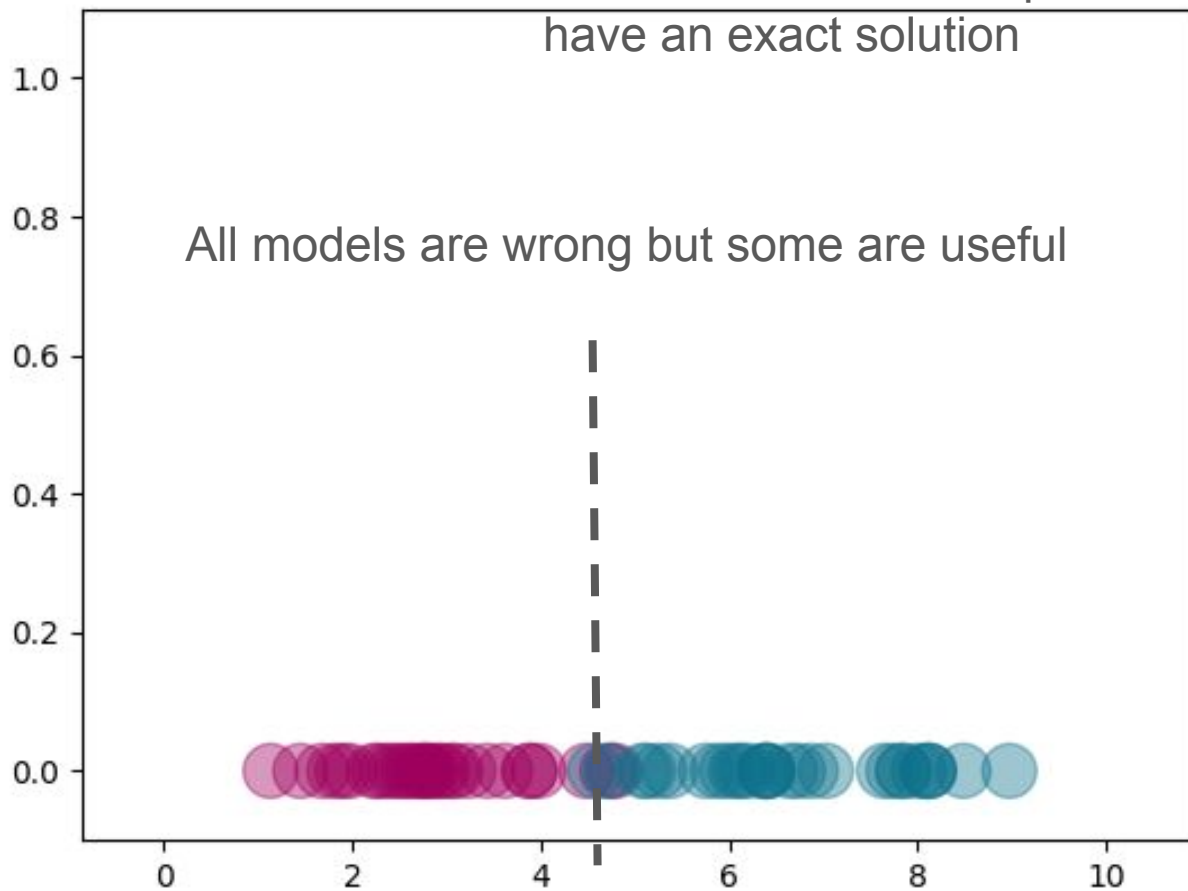
**Weight  
(in Tons)**



Could be because the problem just doesn't  
have an exact solution



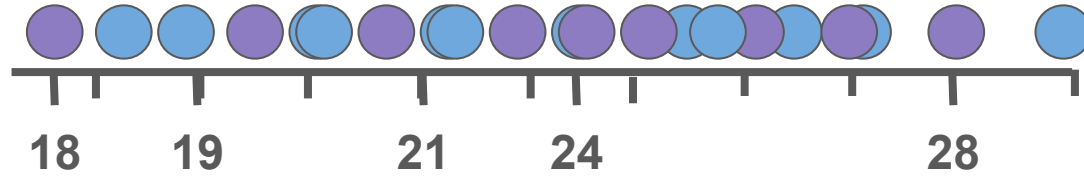
Could be because the problem just doesn't  
have an exact solution



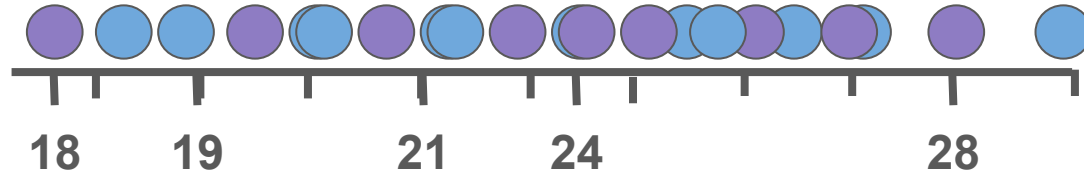
The feasibility of a classification task completely depends on the relationship between the attributes (or predictors) and the class.



For example if we used age instead of weight for elephants and rhinos



Age cannot distinguish rhinos and elephants



# Takeaways

- There could be **many correct answers**
- There could be **no correct answers**
  - But the model could **still be useful** if it's more or less correct most of the time
- Whether a task is feasible depends on:
  - The relationship between the predictors and the class

# Lots of Questions

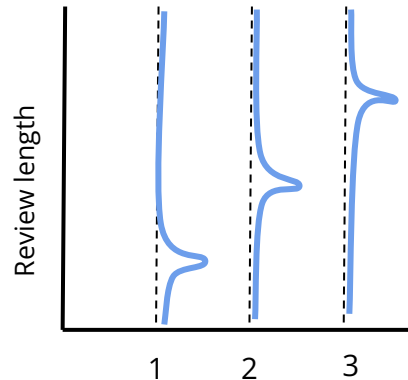
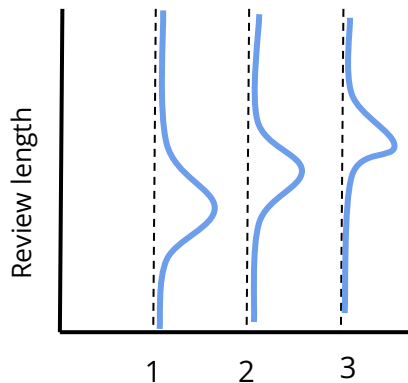
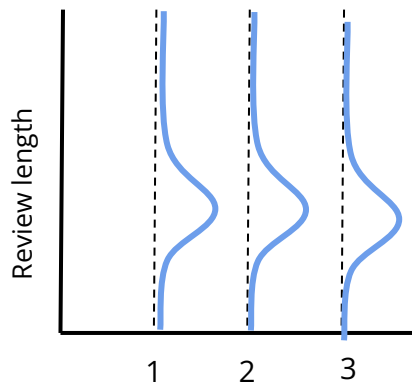
- How do we know if we have good predictors for a task?
- How do we know we have done a good job at classification?

# Lots of Questions

- **How do we know if we have good predictors for a task?**
- How do we know we have done a good job at classification?

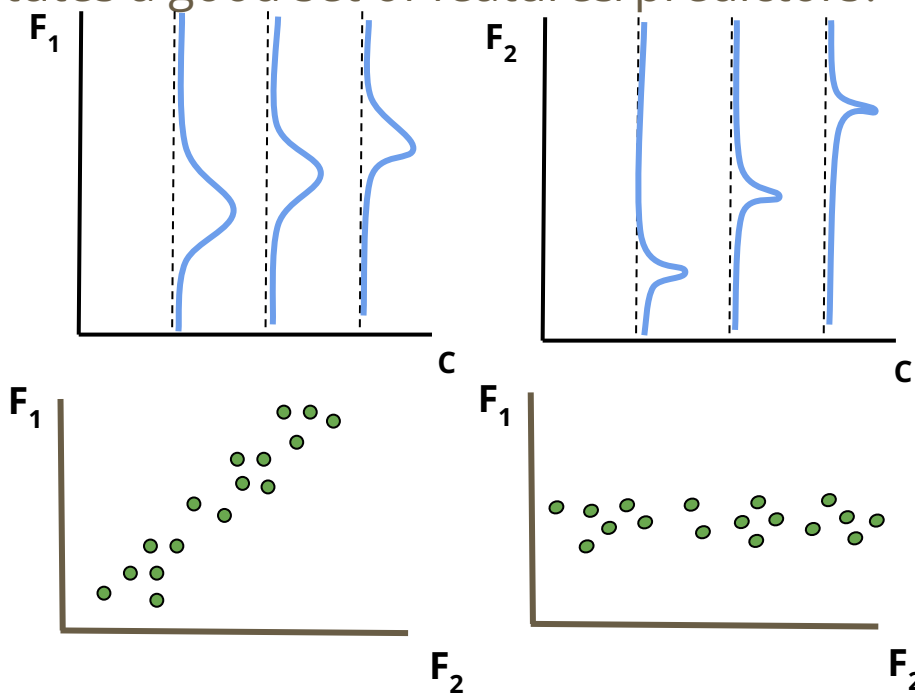
# How do we know we have good predictors?

- What constitutes a good feature/predictor?



# How do we know we have good predictors?

- What constitutes a good feature/predictor?
- What constitutes a good set of features/predictors?



# How do we know we have good predictors?

- What constitutes a good feature/predictor?
- What constitutes a good set of features/predictors?
- BUT...

Correlation is not causation.



# How do we know we have good predictors?

## Correlation VS Causation

1. Temperature and ice cream sales are positively correlated

# How do we know we have good predictors?

## Correlation VS Causation

1. Temperature and ice cream sales are positively correlated
  - a. Temperature increases cause ice cream sales to spike
  - b. Ice cream sale increases do not cause the temperature to rise

# How do we know we have good predictors?

## Correlation VS Causation

1. Temperature and ice cream sales are positively correlated
  - a. Temperature increases cause ice cream sales to spike
  - b. Ice cream sale increases do not cause the temperature to rise
2. Sleeping with shoes on is strongly correlated with waking up with a headache.

# How do we know we have good predictors?

## Correlation VS Causation

1. Temperature and ice cream sales are positively correlated
  - a. Temperature increases cause ice cream sales to spike
  - b. Ice cream sale increases do not cause the temperature to rise
2. Sleeping with shoes on is strongly correlated with waking up with a headache.
  - a. But neither causes the other...
  - b. There's a third common factor causing this correlation: going to bed drunk.

# How do we know we have good predictors?

Testing for causality requires specific testing / experimentation with a control group

# Lots of Questions

- How do we know if we have good predictors for a task?
- **How do we know we have done a good job at classification?**

**How do we know we've done well at classification?**

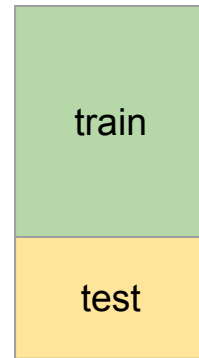
# How do we know we've done well at classification?

- Testing without cheating. Learning not memorizing.



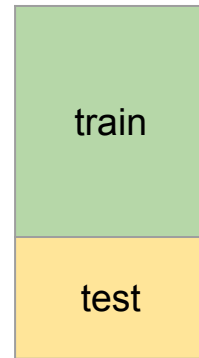
# How do we know we've done well at classification?

- Testing without cheating. Learning not memorizing.
  - Split up our data into a training set and a separate testing set
  - Use the training set to find patterns and create a model
  - Use the testing set to evaluate the model on data it has not seen before



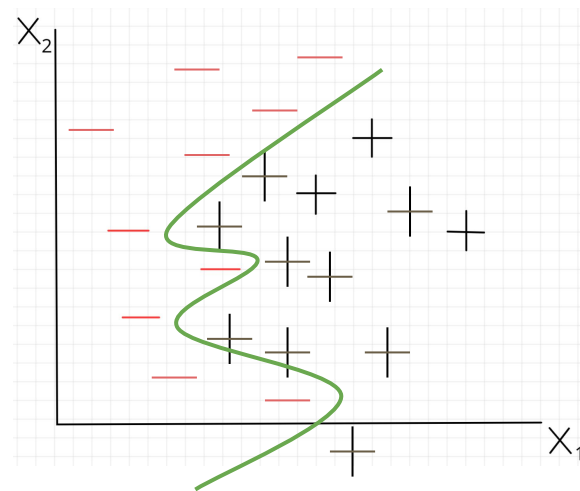
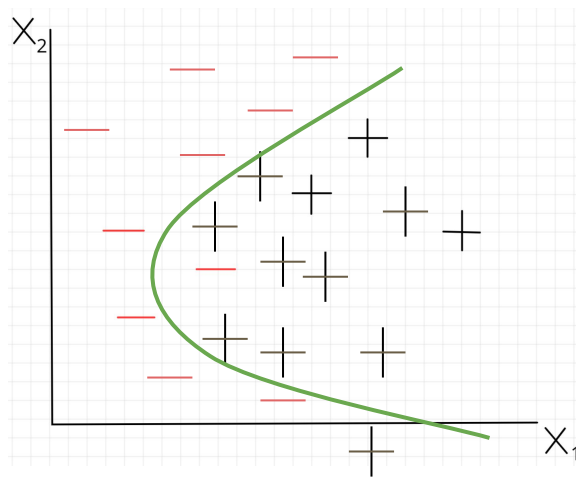
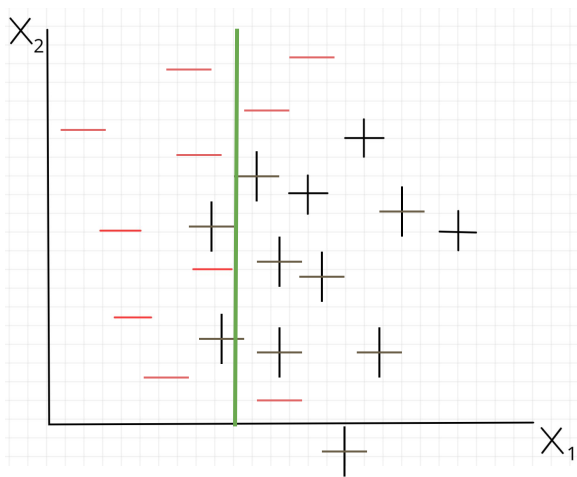
# How do we know we've done well at classification?

- Testing without cheating. Learning not memorizing.
  - Split up our data into a training set and a separate testing set
  - Use the training set to find patterns and create a model
  - Use the testing set to evaluate the model on data it has not seen before
- Also allows us to check that we have not learned a model TOO SPECIFIC to the dataset
  - Overfitting vs underfitting



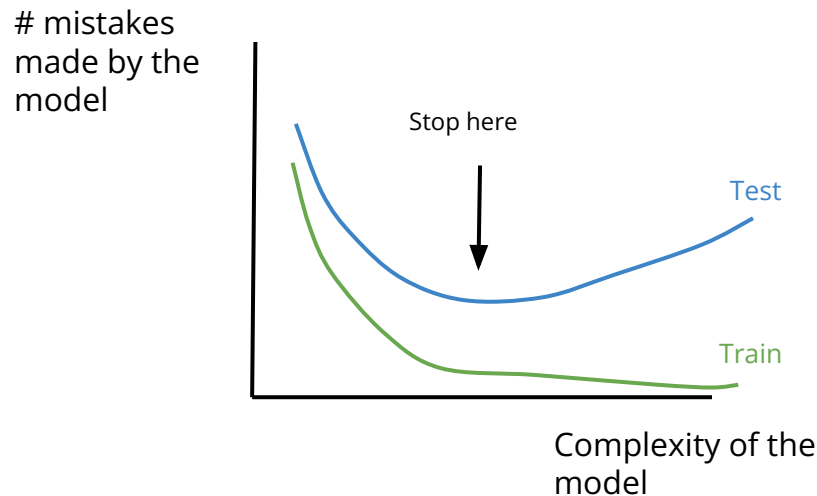
# How do we know we've done well at classification?

## Underfitting VS Overfitting



# How do we know we've done well at classification?

## Underfitting VS Overfitting

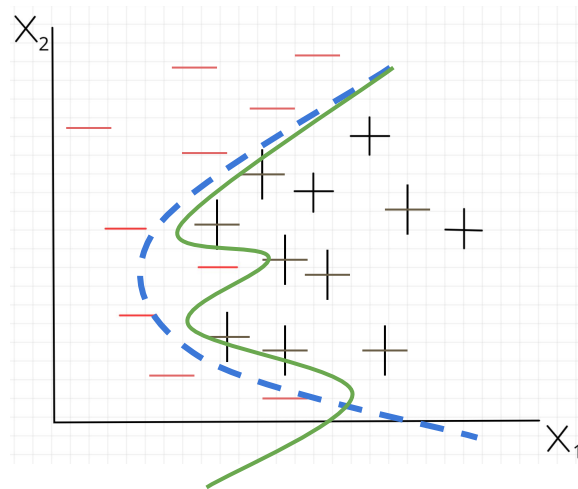
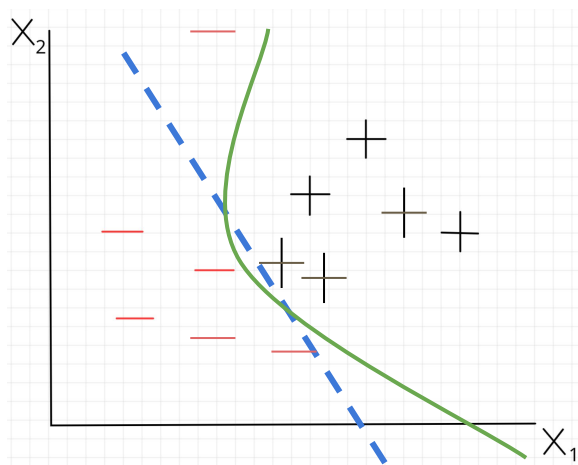


# How do we know we've done well at classification?

- Testing without cheating:
  - Split up our data into a training set and a separate testing set
  - Use the training set to find patterns and create a model
  - Use the testing set to evaluate the model on data it has not seen before
- Also allows us to check that we have not learned a model TOO SPECIFIC to the dataset
  - Overfitting vs underfitting
  - Goal is to capture general trends
    - Watch out for outliers and noise

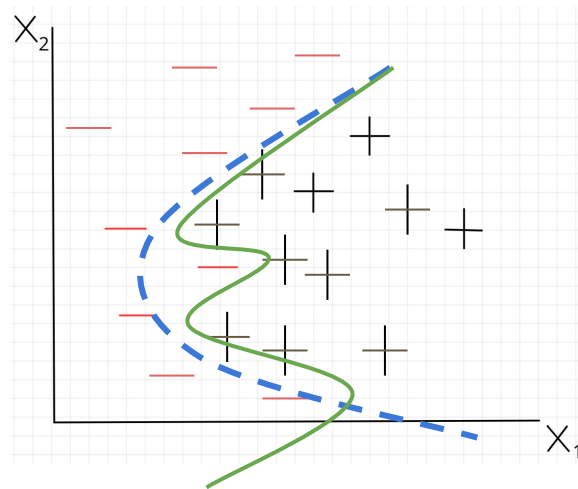
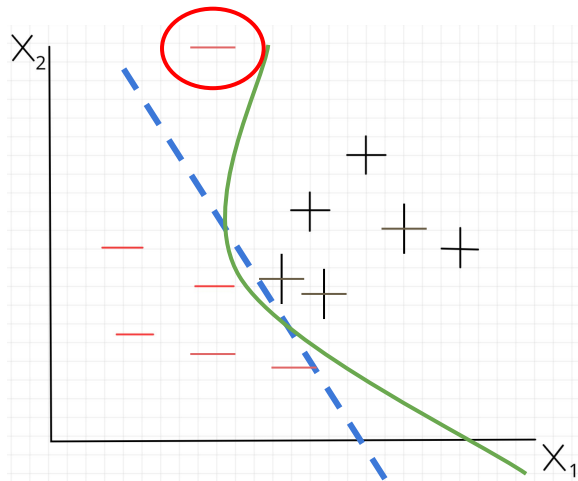
# How do we know we've done well at classification?

## Outliers and Noise



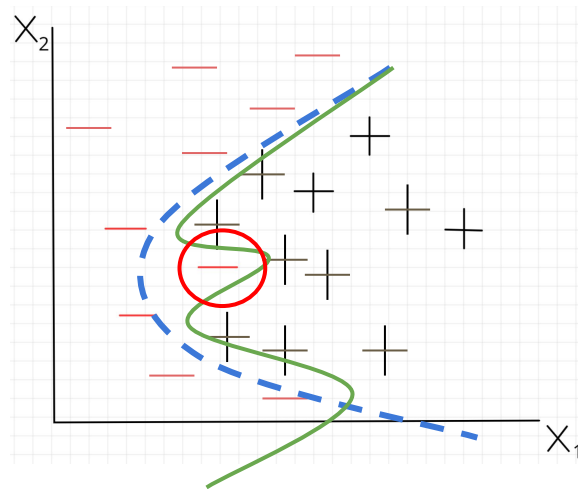
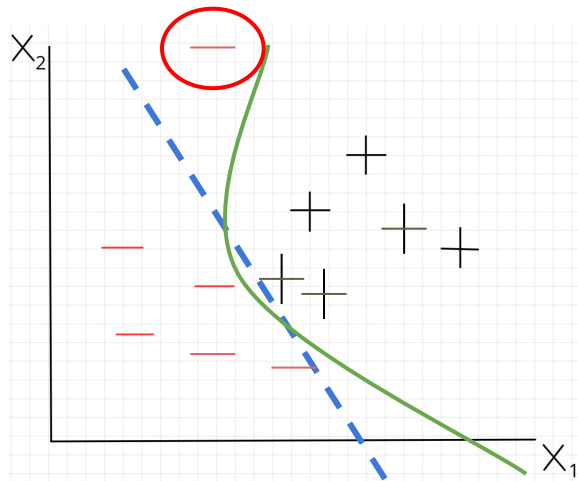
# How do we know we've done well at classification?

## Outliers and Noise



# How do we know we've done well at classification?

## Outliers and Noise





# How do we know we've done well at classification?

- Testing without cheating:
  - Split up our data into a training set and a separate testing set
  - Use the training set to find patterns and create a model
  - Use the testing set to evaluate the model on data it has not seen before
- Also allows us to check that we have not learned a model TOO SPECIFIC to the dataset
  - Overfitting vs underfitting
  - Goal is to capture general trends
    - Watch out for outliers and noise
- The types of mistakes made matters

# How do we know we've done well at classification?

## Types of mistakes

- Testing for a rare disease
  - Out of 1000 data points, only 10 have this rare disease. A model that simply tells folks they don't have the disease will have an accuracy of 99%.

# Part 1

# Classification

- Training Step
  - Create the model based on the examples / data points in the training set
- Testing Step
  - Use the model to fill in the blanks of the testing set
  - Compare the result of the model to the true values

# Instance-Based Classifiers

- Use the stored training records to predict the class label of unseen cases
- Rote-learners:
  - Perform classification only if the attributes of the unseen record exactly match a record in our training set

# Instance-Based Classifiers: Training Step

age	Tumor size	malignant?
20	10	no
30	15	yes
40	20	no
50	25	yes

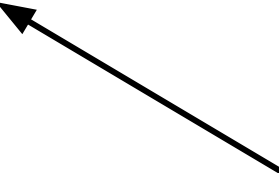
learn  
model

There is no training step per se. The dataset itself is the model.

# Instance-Based Classifiers: Applying the model

age	Tumor size	malignant?
20	10	no
30	15	yes
40	20	no
50	25	yes

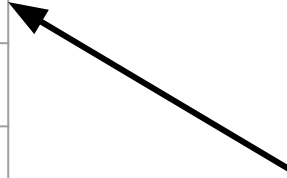
age	Tumor size	malignant?
20	10	?



# Instance-Based Classifiers: Applying the model

age	Tumor size	malignant?
20	10	no
30	15	yes
40	20	no
50	25	yes

age	Tumor size	malignant?
20	10	<b>no</b>





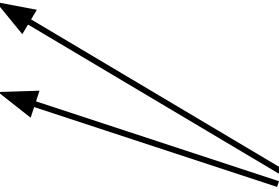
# Instance-Based Classifiers

- Use the stored training records to predict the class label of unseen cases
- Rote-learners:
  - Perform classification only if the attributes of the unseen record exactly match a record in our training set

# Instance-Based Classifiers

age	Tumor size	malignant?
20	10	no
30	15	yes
40	20	no
50	25	yes

age	Tumor size	malignant?
25	5	?



# Nearest Neighbor Classifier

Use **SIMILAR** records to perform classification

# K Nearest Neighbor Classifier

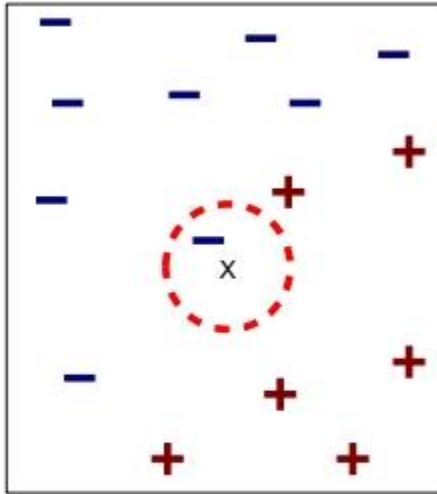
Requires:

- Training set
- Distance function
- Value for  $k$

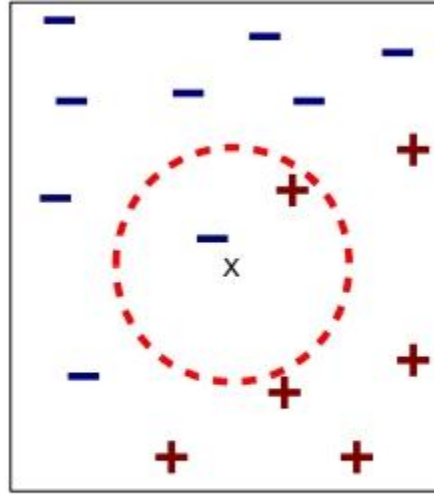
How to classify an unseen record:

1. Compute distance of unseen record to all training records
2. Identify the  $k$  nearest neighbors
3. Aggregate the labels of these  $k$  neighbors to predict the unseen record class (ex: majority rule)

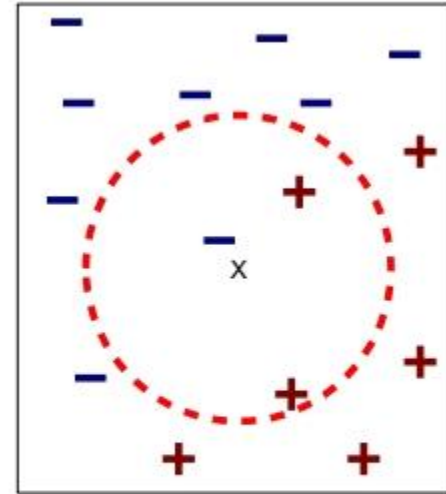
# K Nearest Neighbor Classifier



(a) 1-nearest neighbor



(b) 2-nearest neighbor



(c) 3-nearest neighbor

# K Nearest Neighbor Classifier

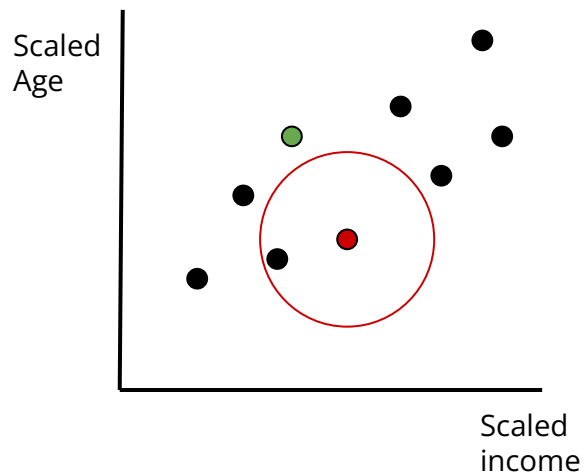
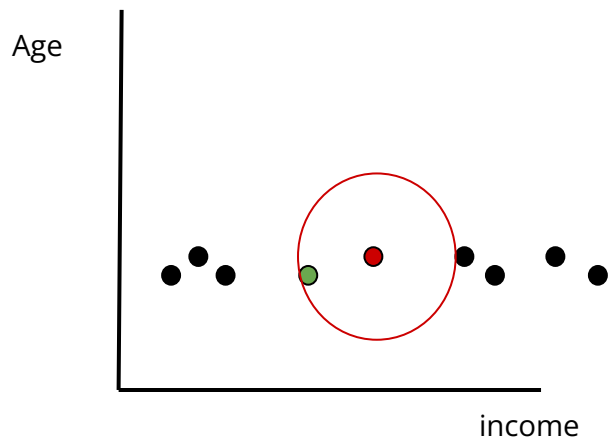
Aggregation methods:

- Majority rule
- Weighted majority based on distance ( $w = 1/d^2$ )

Scaling issues:

- Attributes should be scaled to prevent distance measures from being dominated by one attribute. Example:
  - Age: 0 -> 100
  - Income: 10k -> 1million

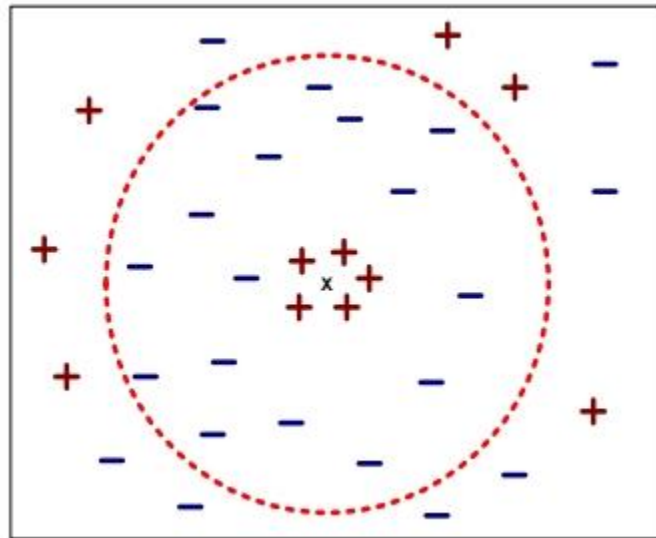
# Scaling Attributes



# K Nearest Neighbor Classifier

Choosing the value of k:

- If k is too small ->
  - sensitive to noise points + doesn't generalize well
- If k is too big ->
  - neighborhood may include points from other classes







# K Nearest Neighbor Classifier

Pros:

- Simple to understand why a given unseen record was given a particular class

Cons:

- Expensive to classify new points
- KNN can be problematic in high dimensions (curse of dimensionality)

## Part 2

A B C

Where would the KNN decision boundary be for  $K=1$ ?

