Classification

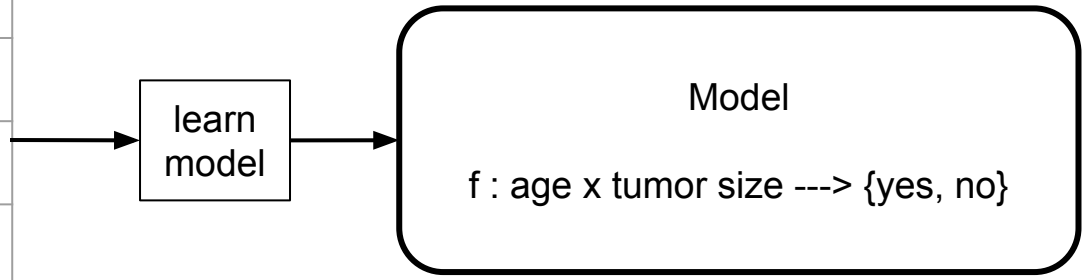
Boston University CS 506 - Lance Galletti

What is Classification?

- Given a **training set** where data is labeled with a special **attribute** called a **class** (a discrete value)
- We want to find a **model** describing the **class** attribute as a function of the values of the other attributes
- Goal: use this model on unlabeled data to assign a class as accurately as possible

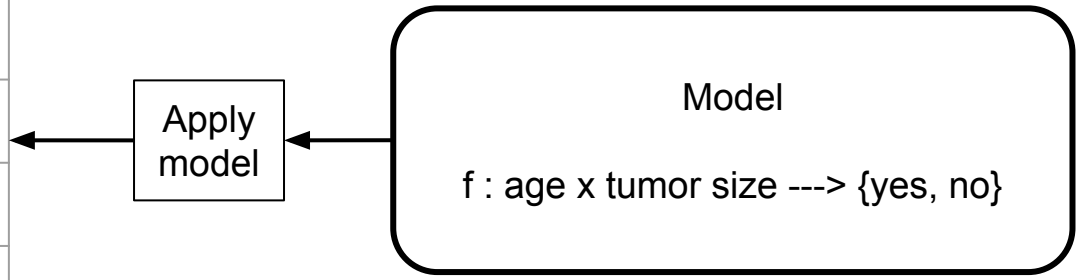
Example

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



Example

age	Tumor size	malignant?
25	5	?
35	10	?
45	25	?



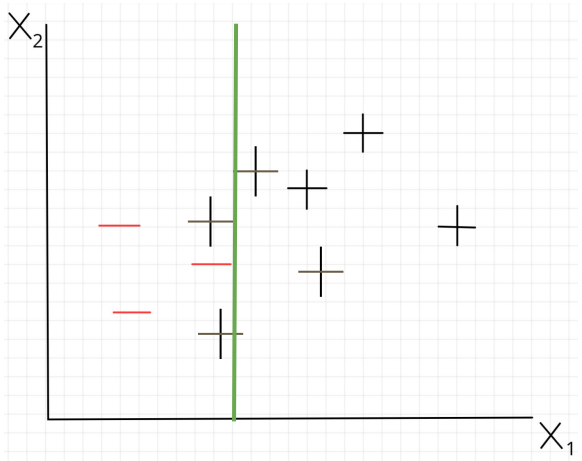
Classification Tasks

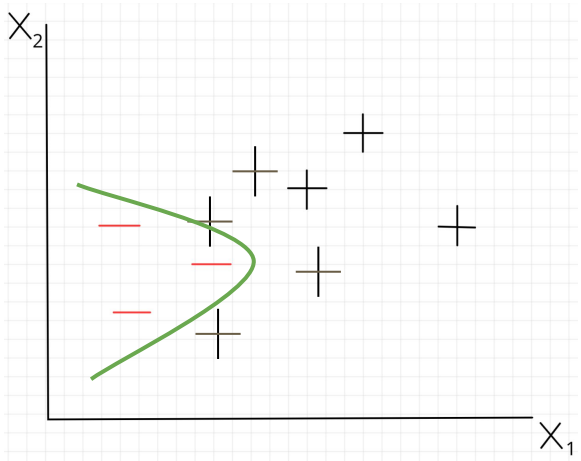
- Predicting tumor cells as benign or malignant
- Classifying images
- Classifying credit card transactions as being legitimate or fraudulent
- Many more

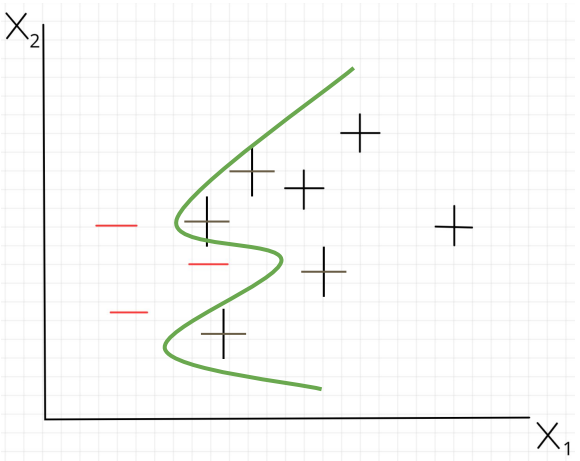
Classification Techniques

- Instance-Based Classifiers
- Decision Trees
- Naive Bayes
- Support Vector Machines
- Neural Networks

Underfitting VS Overfitting



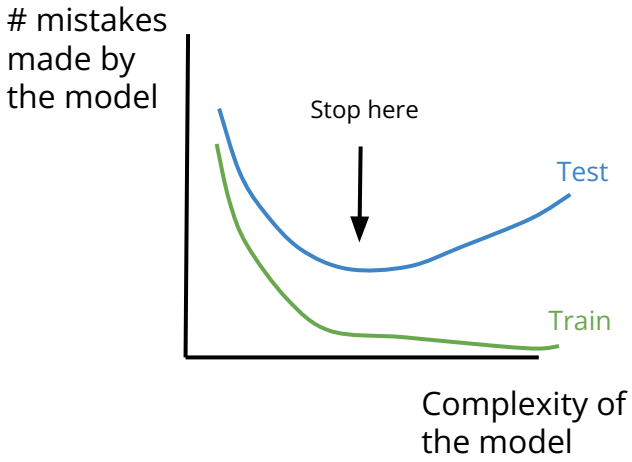




Underfitting VS Overfitting

Model Evaluation (simply)

- Evaluating a model on the data it was trained on is cheating - can just memorize.
- Distinction between data used for training and data left out used for testing / evaluation.

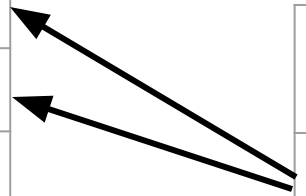


Instance-Based Classifiers

- 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
- Nearest Neighbor:
  - Use the k closest records to perform classification

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	?

K Nearest Neighbor Classifier

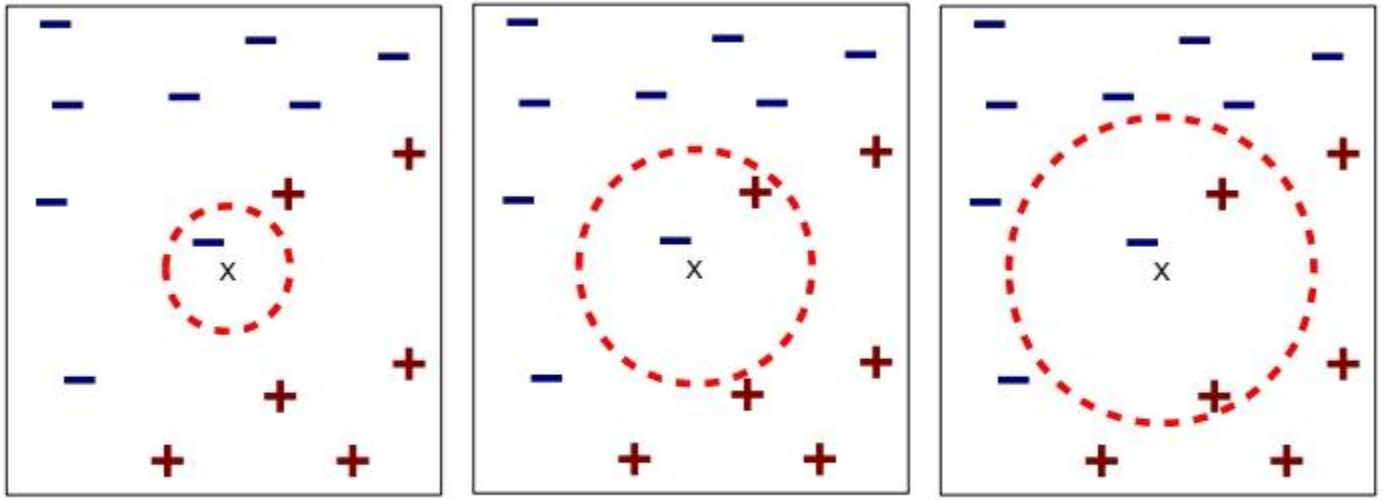
Requires:

- 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:
  - Height: 1m -> 2m
  - Income: 10k -> 1million

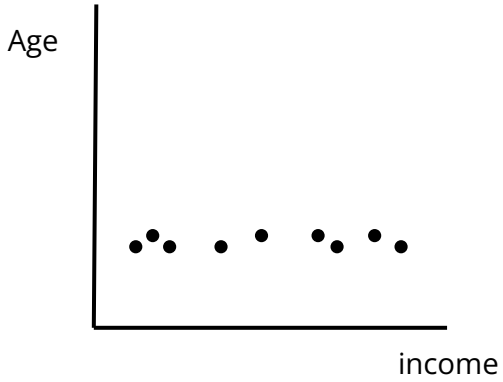
K Nearest Neighbor Classifier

Aggregation methods:

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

Scaling issues:

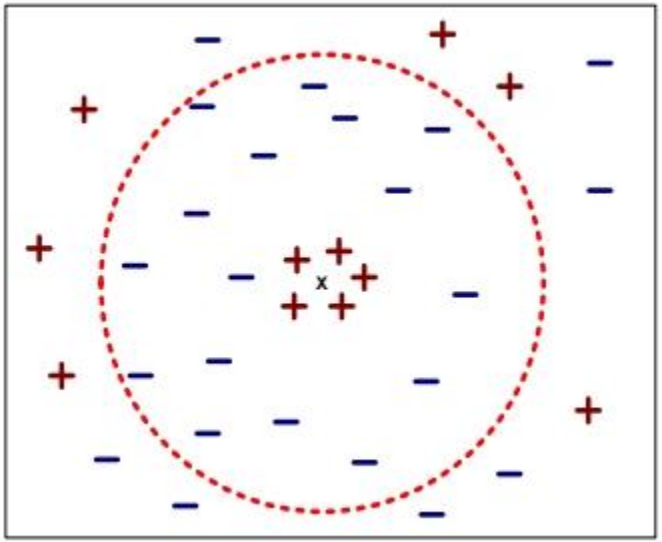
- Attributes should be scaled to prevent distance measures from being dominated by one attribute. Example:
  - Age: 1 -> 100
  - Income: 10k -> 100million

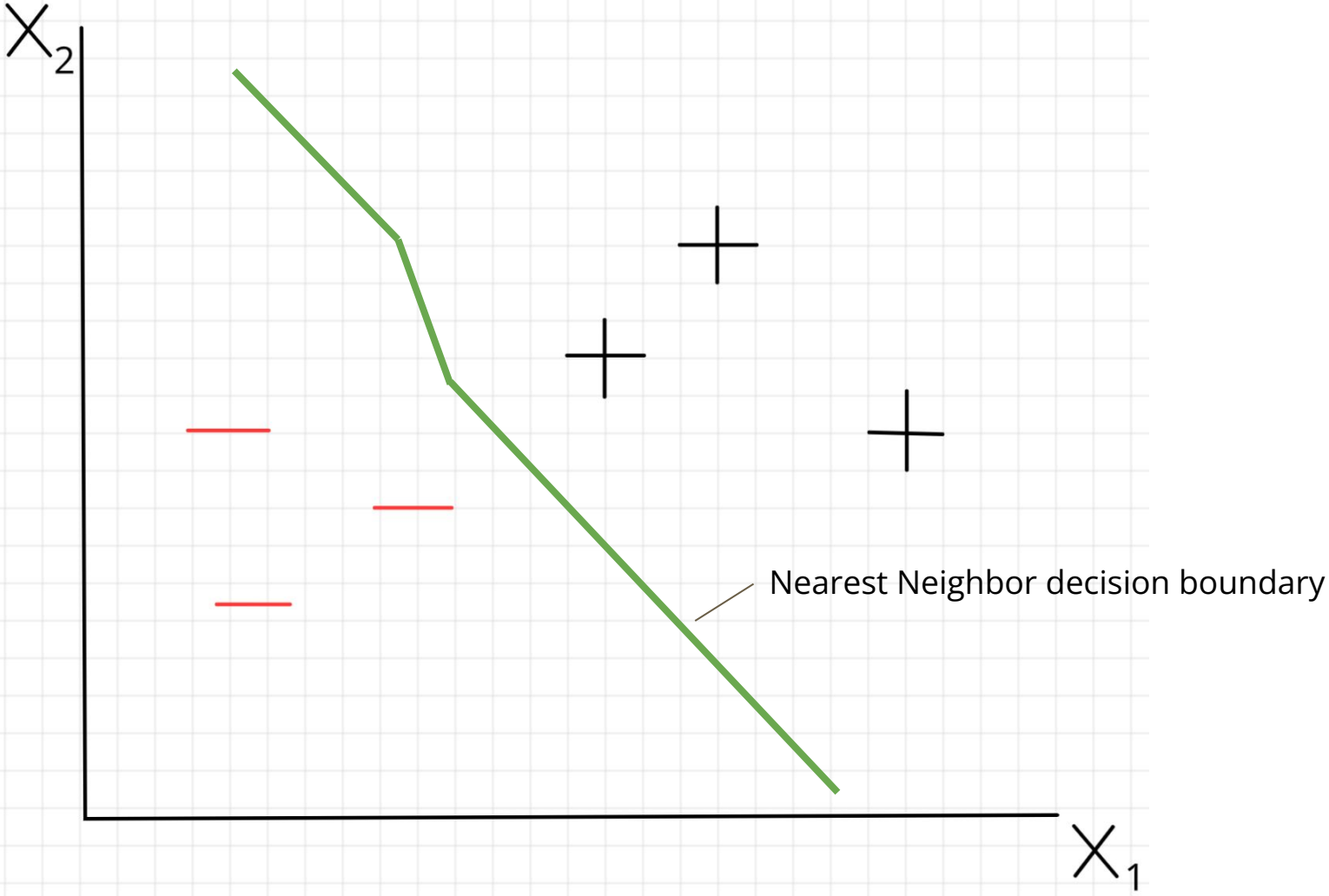


K Nearest Neighbor Classifier

Choosing the value of k:

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




X_2
Nearest Neighbor decision boundary
X_1

K Nearest Neighbor Classifier

Pros:

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

Cons:

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