
Classification

— Boston University CS 506 - Lance Galletti —

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

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 how the **class** attribute varies 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

learn
model

Model

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

Example

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

Apply
model

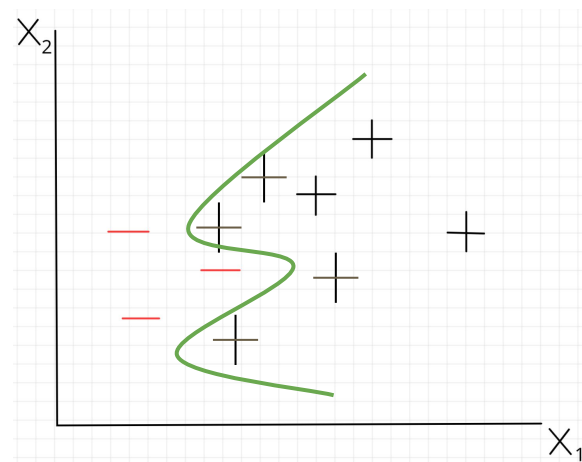
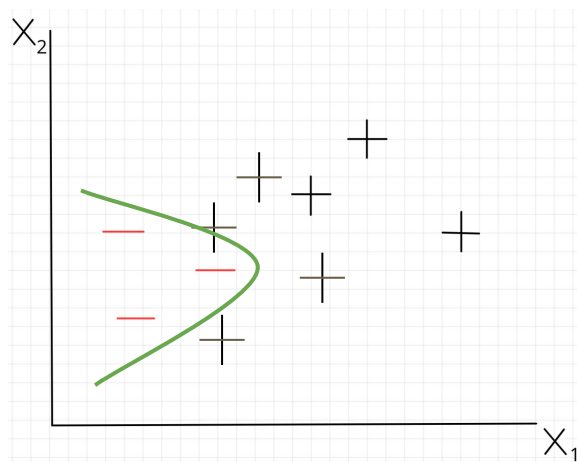
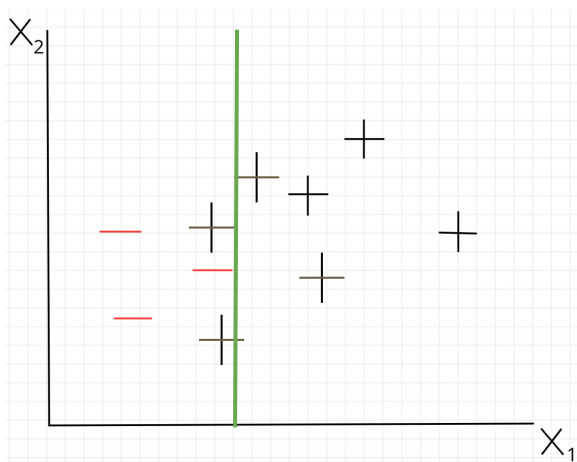
Model

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

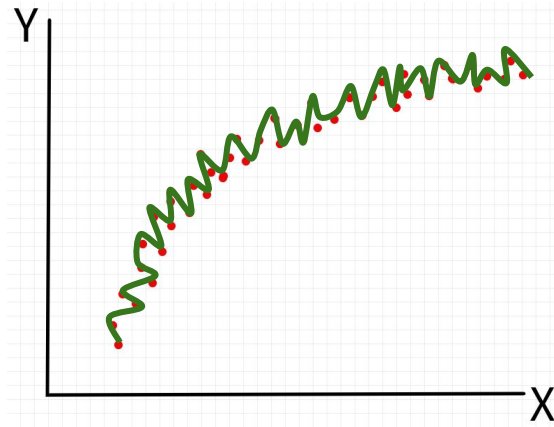
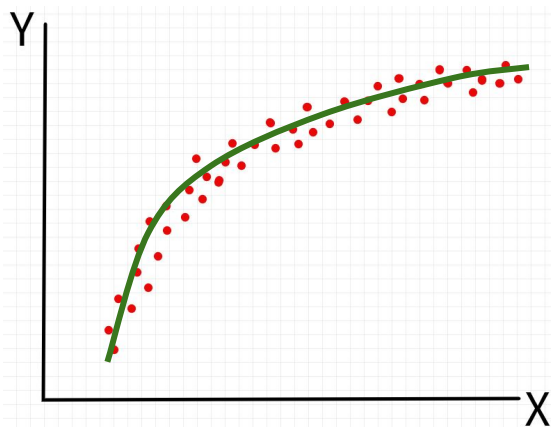
Modeling Philosophy

- What constitutes a good feature?
- What constitutes a good set of features?
 - Change in F_1, \dots, F_m means expect a change in Y
- Correlation vs causation
- Primary goal is to capture the general trend / relationship between class and features as simply as possible
 - Outliers
 - Noise
- Model performance / evaluations
 - Overfitting vs Underfitting
- All models are wrong but *some* are useful. What value does your model provide?

Underfitting VS Overfitting

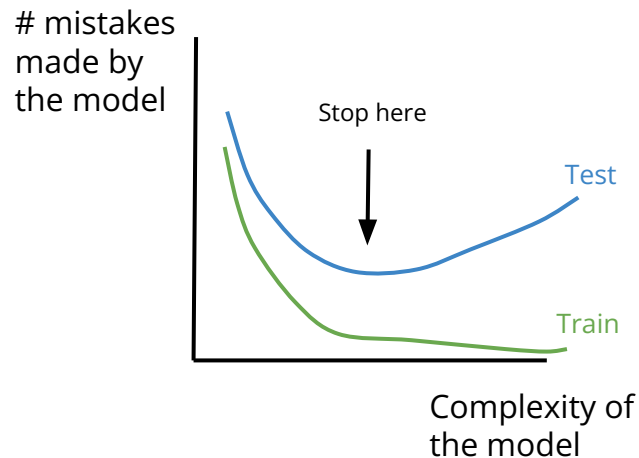


Underfitting VS Overfitting



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.



Worksheet Part 1

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

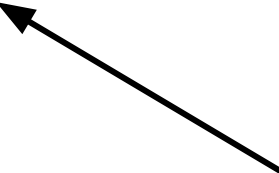
learn
model

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

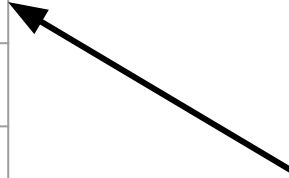
age	Tumor size	malignant?
20	10	?



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	no



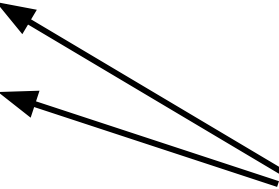
Instance-Based Classifiers

- 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

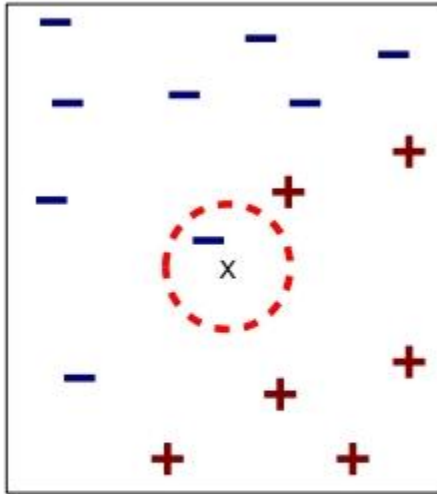
Requires:

- Training set
- Distance function
- Value for k

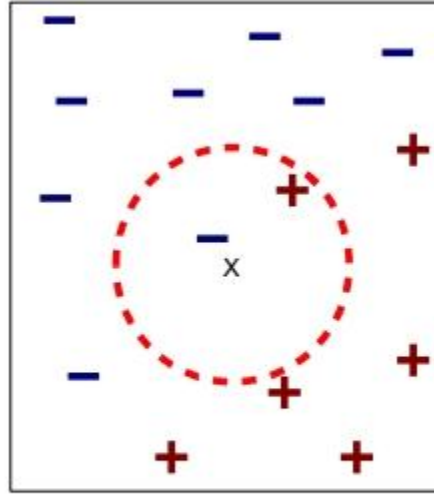
How to classify an unseen record:

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)

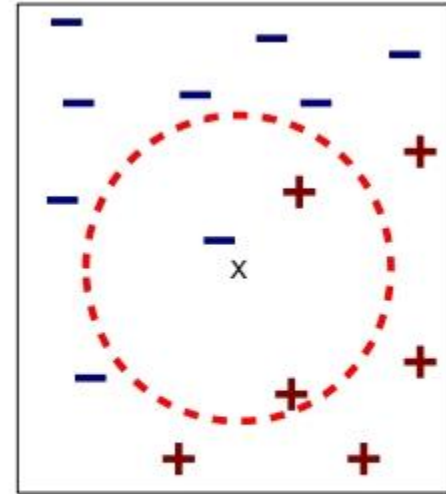
K Nearest Neighbor Classifier



(a) 1-nearest neighbor



(b) 2-nearest neighbor



(c) 3-nearest neighbor

K Nearest Neighbor Classifier

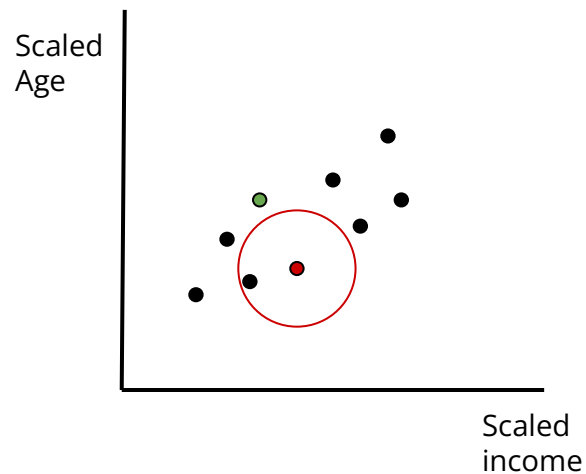
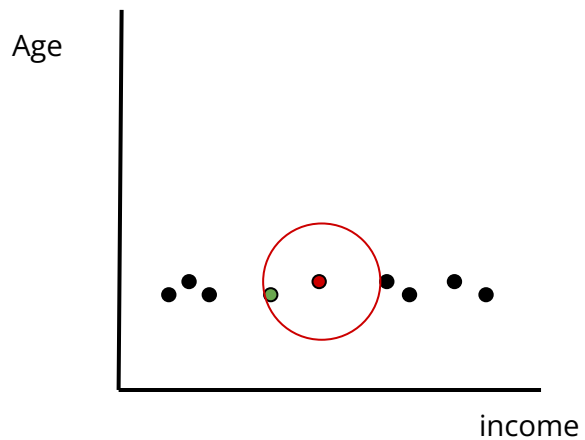
Aggregation methods:

- 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

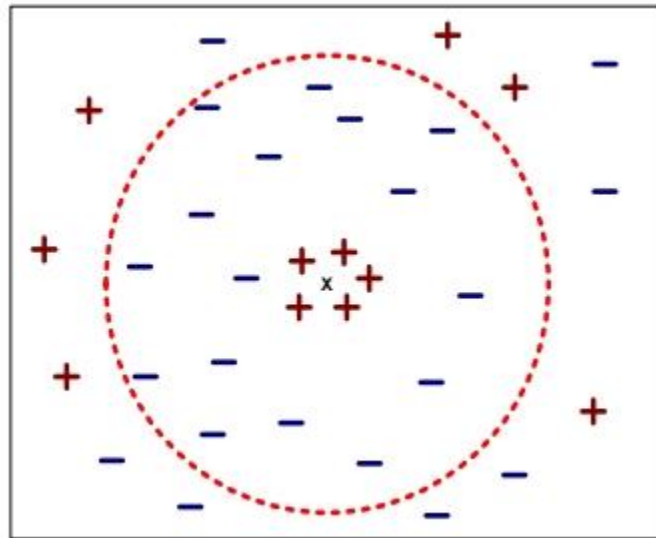
Scaling Attributes



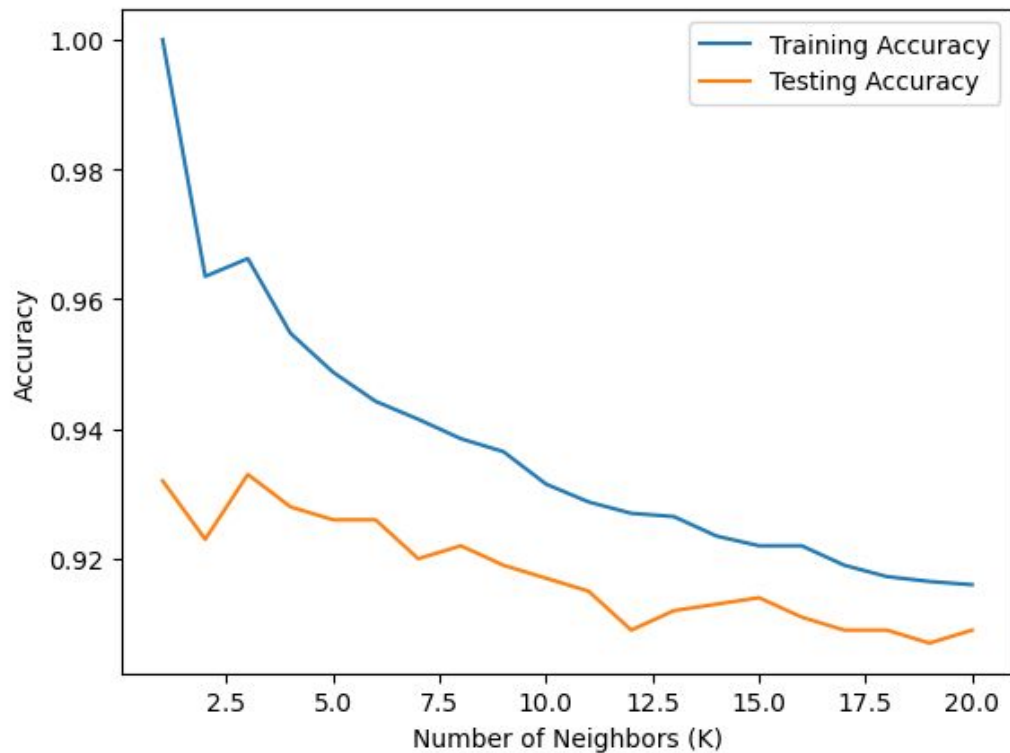
K Nearest Neighbor Classifier

Choosing the value of k:

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



How to choose k





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)