Information Retrieval & Text Mining

Classification Techniques and Classifier Evaluation

Dr. Iqra Safder
Information Technology University

Term Project

- 3-4 group members
- Use any textual datasets.
  - Keggal
  - ACL
  - Github
  - Google Dataset
  - Elseveir (dataset along with papers are available)
  - Paperswithcode

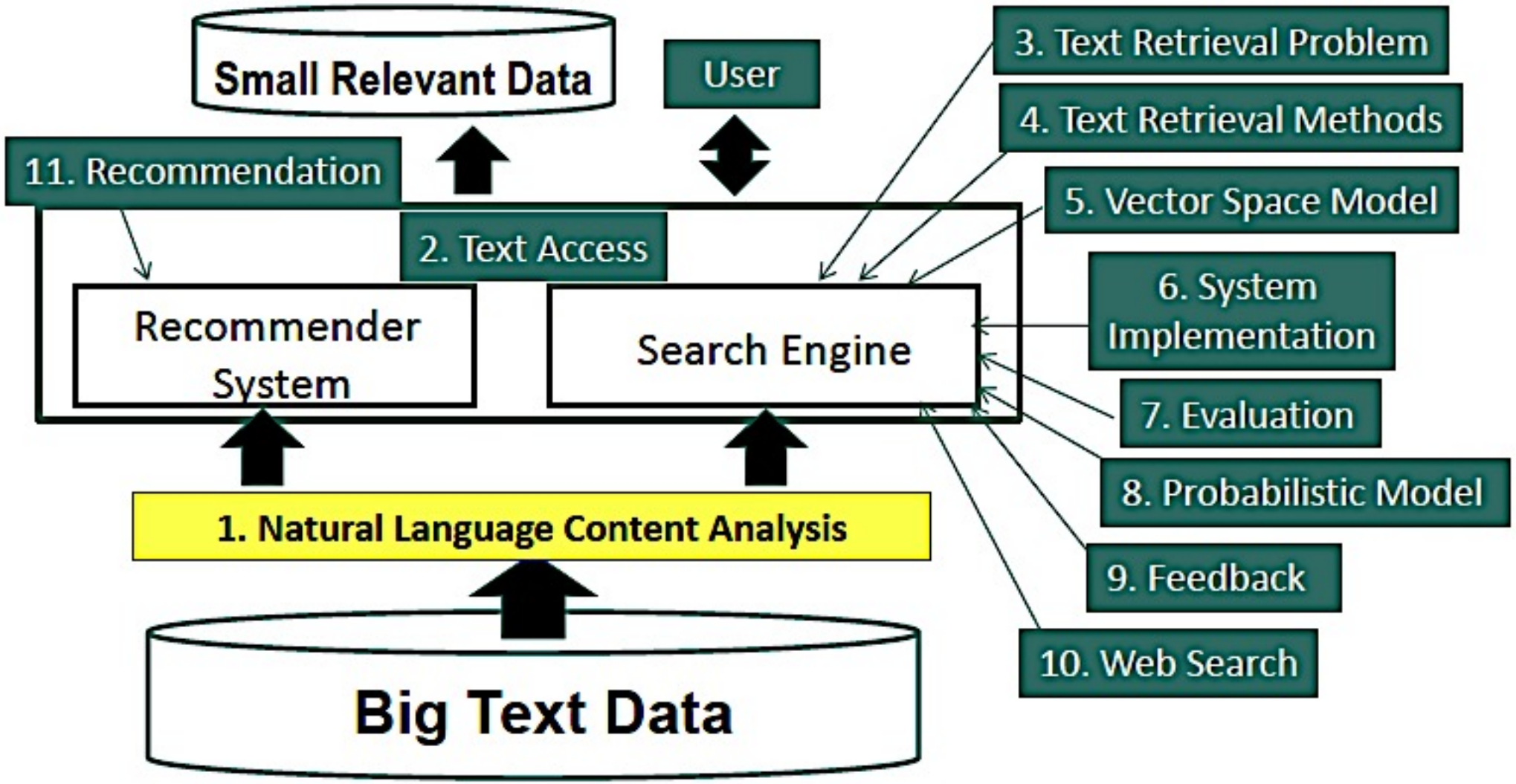
- Deadline to choose project ideas 18th November, 2021
- Please fill the online excel sheet.

Quiz #2

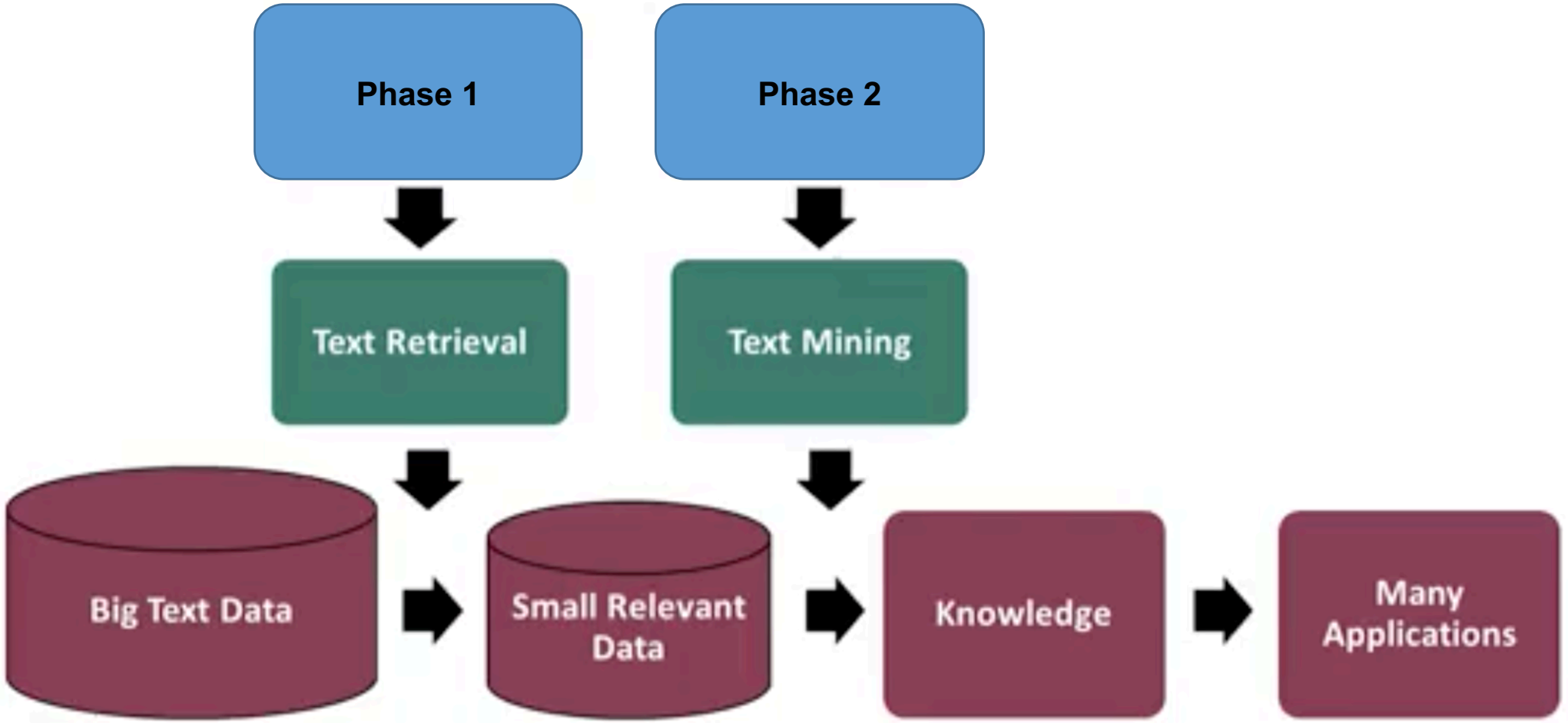
Thrusday, 18th November 2021

Content: Lecture 6- Lecture 13

Course Schedule



Main Techniques for Harnessing Big Data: Information Retrieval + Text Mining



Overview

- **Supervised and Unsupervised learning**
- **What is Classification?**
- **Classification Models**
  - Decision Tree Classifier, K-NN Classifier, Naïve Bayes Classifier, Support Vector Machines
- **Evaluating Classifiers**
  - Overfitting and Underfitting
  - The K-fold Cross-Validation Method

Supervised vs. Unsupervised Learning

- **Supervised learning (classification)**
  - Supervision: The training data (observations, measurements, etc.) are accompanied by **labels** indicating the class of the observations.
  - New data is classified based on the training set.

- **Unsupervised learning (clustering)**
  - The class labels of training data is unknown.
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data.

What is Classification?

- A machine learning task that deals with identifying the class to which an instance belongs.

(Textual features : Ngrams)

(Training inputs)

(Age, Marital status, Health status, Salary)

Test instance

Attributes

(a1, a2,… an)

Classifier

Discrete-valued

Issue Loan? {Yes, No}
Class label

Category of document?
{Politics, Movies,
Biology}

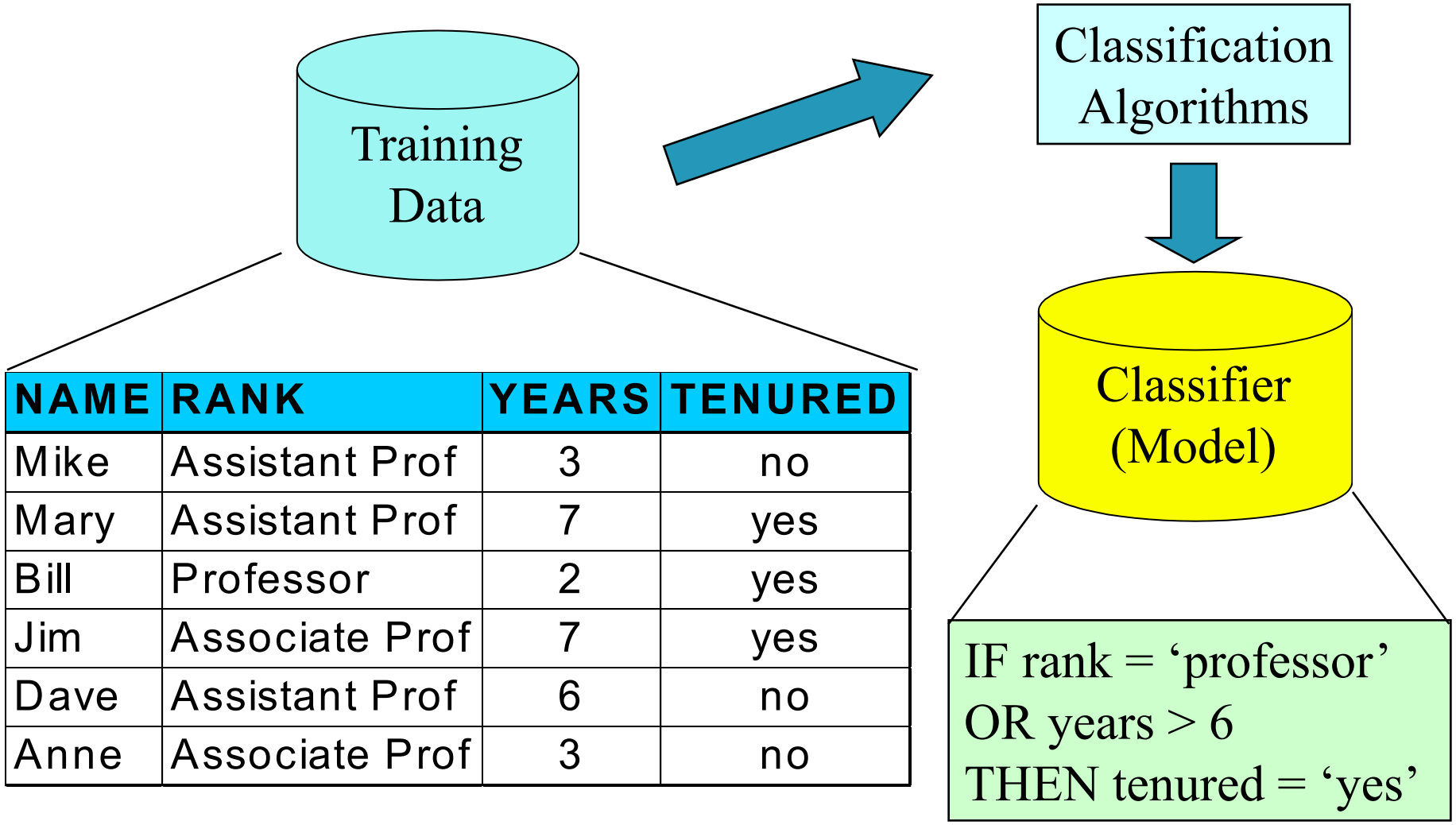
Classification - A Two-Step Process

- **Model construction (Step1):** describing a set of predetermined classes
  - Each tuple/sample is assumed to belong to a predefined class, as determined by the class label attribute.
  - The set of tuples used for model construction is training set.
  - The model is represented as classification rules, decision trees, or mathematical formulae.

Classification - A Two-Step Process

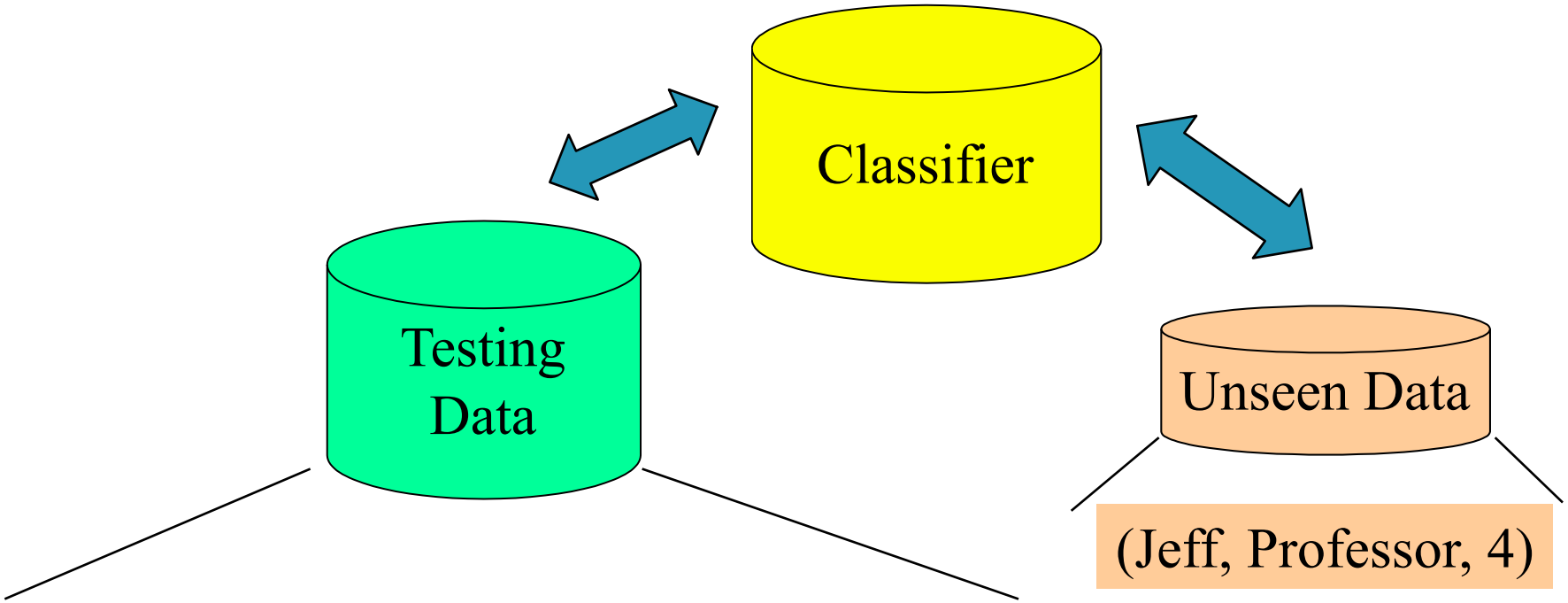
- **Model Usage (Step2):** For classifying future or unknown objects
  - Estimate accuracy of the model
    - The known label of test sample is compared with the classified result from the model
    - Accuracy rate is the percentage of test set samples that are correctly classified by the model
    - Test set is independent of training set (otherwise overfitting)
  - If the accuracy is acceptable, use the model to classify new data
- Note: If *the test set* is used to select models, it is called validation (test) set

Step1: Model Construction



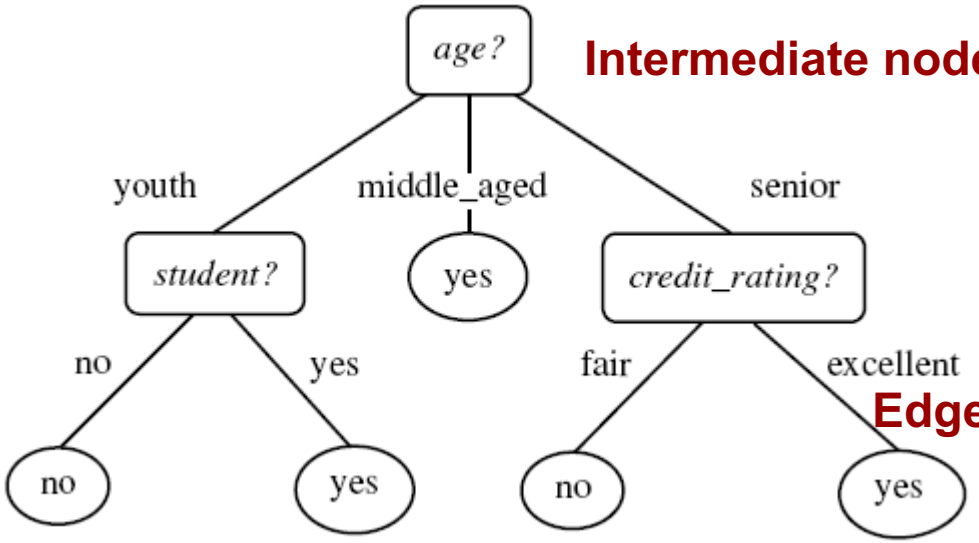
NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Step2: Using the Model in Prediction



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Classification Model: Decision Tree



Intermediate nodes : Attributes

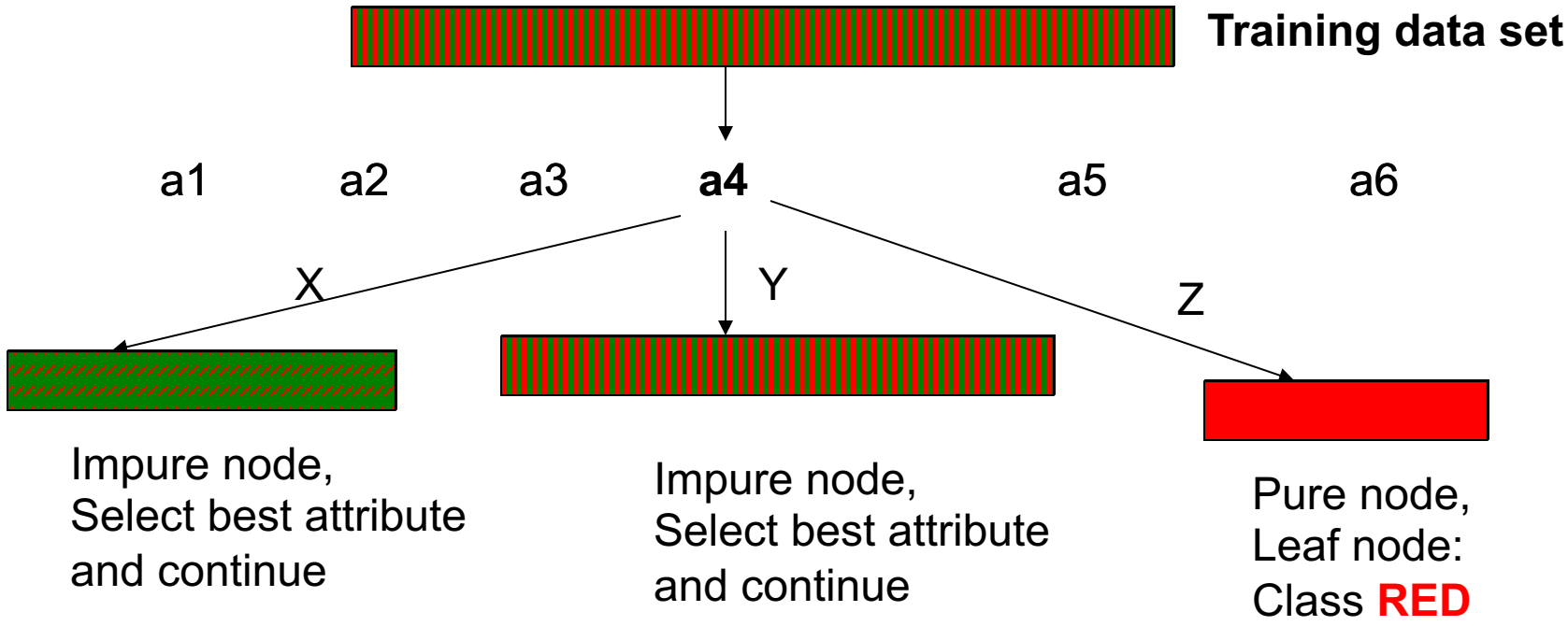
Edges : Attribute value tests

Leaf nodes : Class predictions

Example algorithms: ID3, C4.5, SPRINT, CART

Diagram from Han-Kamber

Decision Tree Schematic



A Decision to split at each node is made according to the metric called purity. A node is 100% impure if it splits equally into 50/50 and 100% pure when all of its data belongs to the single class

Decision Tree Issues

How to determine the attribute for split?

Alternatives:

- Information Gain

  Gain (A, S) = Entropy (S) – **Σ** ((Sj/S)*Entropy(Sj))

Other options:

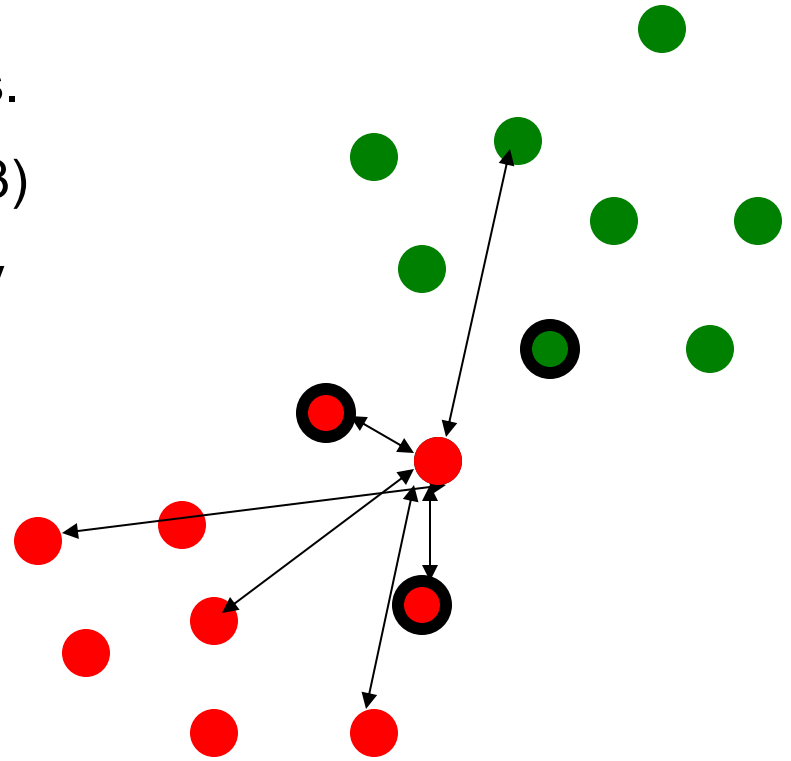
Gain ratio, etc.

The other metric used is information gain, which is used to decide what feature to split at each step in the tree.

Classification Model: K-NN Classifier

For a test instance,

1) Calculate distances from training pts.
2) Find K-nearest neighbors (say, K = 3)
3) Assign class label based on majority



$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}.$$

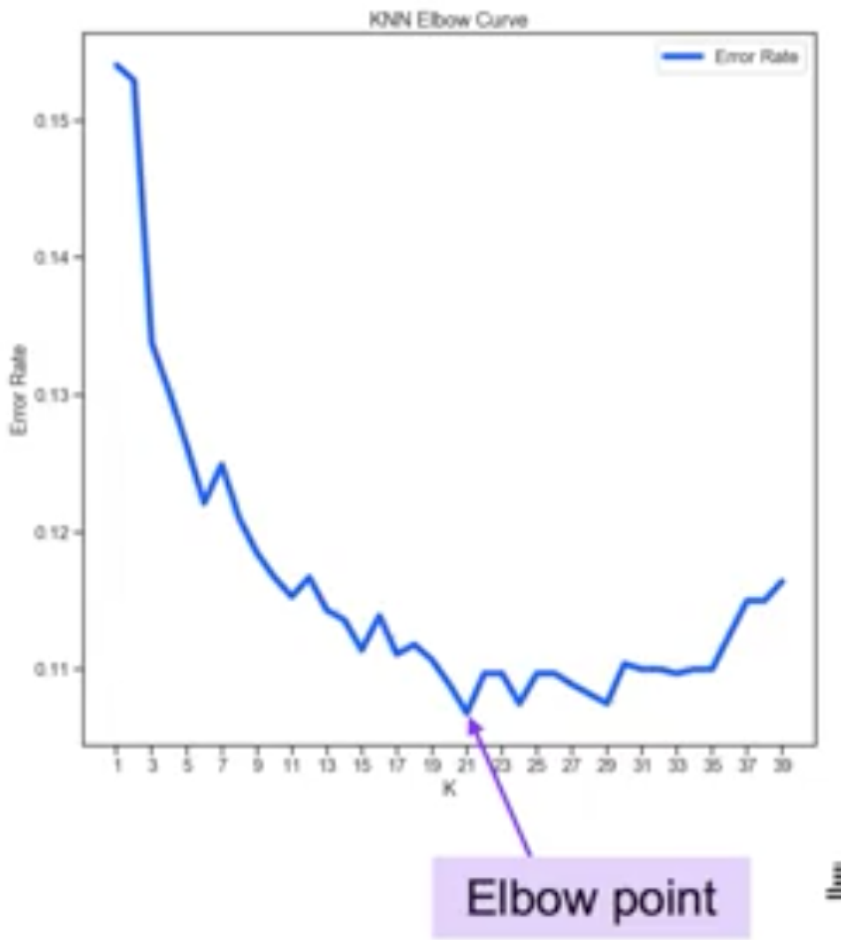
Feature Scaling
Min max scaling

$$v' = \frac{v - min_A}{max_A - min_A},$$

K Nearest Neighbors Decision Boundary

Choosing the right value for K

- KNN does not provide a 'correct' *K*
- The right value depends on which error metric is most important
- A common approach is to use an 'elbow method' approach
- This emphasizes kinks in a curve of the error rate as a function of K
- Beyond this point, the rate of improvement slows or stops



K-NN Classifier Issues

How to determine distances between values of categorical attributes?

Alternatives:

1. Boolean distance (1 if same, 0 if different)
2. Differential grading (e.g. weather – ‘drizzling’ and ‘rainy’ are closer than ‘rainy’ and ‘sunny’)

Classification Model: Naïve Bayes (Probabilistic Classifier)

- Based on Bayes rule
- Naïve Bayes : Conditional independence assumption

$$C = \underset{C}{\operatorname{argmax}}\, P(C_i \mid X) = \frac{P(X \mid C_i) \cdot P(C_i)}{P(X)}$$

$$P(X \mid C_i) = \prod_{k=1}^{d} P(x_k \mid C_i)$$

Naïve Bayes Issues

- **Problems due to sparsity of data?**

- Problem: Probabilities for some values may be zero

  Solution: Laplace smoothing

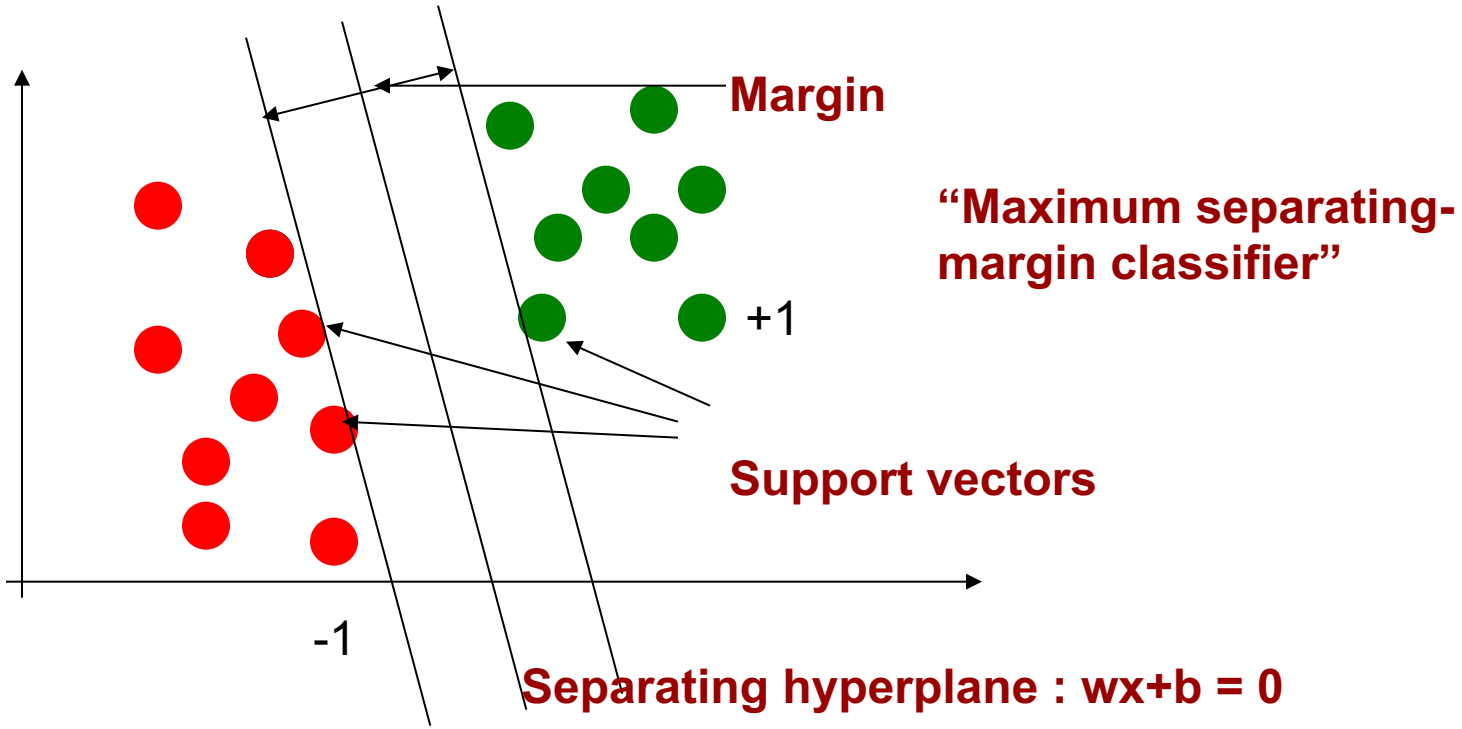
- For each attribute value,

  update probability m / n as : (m + 1) / (n + k)

  where k = domain of values

Classification Model: Support Vector Machines

Basic Idea:



“Maximum separating-margin classifier”

SVM Issues

- **What if n-classes are to be predicted?**

- Problem: SVMs deal with two-class classification

  Solution: Have multiple SVMs each for one class

Evaluating Classifiers

Outcome:

- Accuracy
- Confusion matrix
- If cost-sensitive, the expected cost of classification (attribute test cost + misclassification cost)

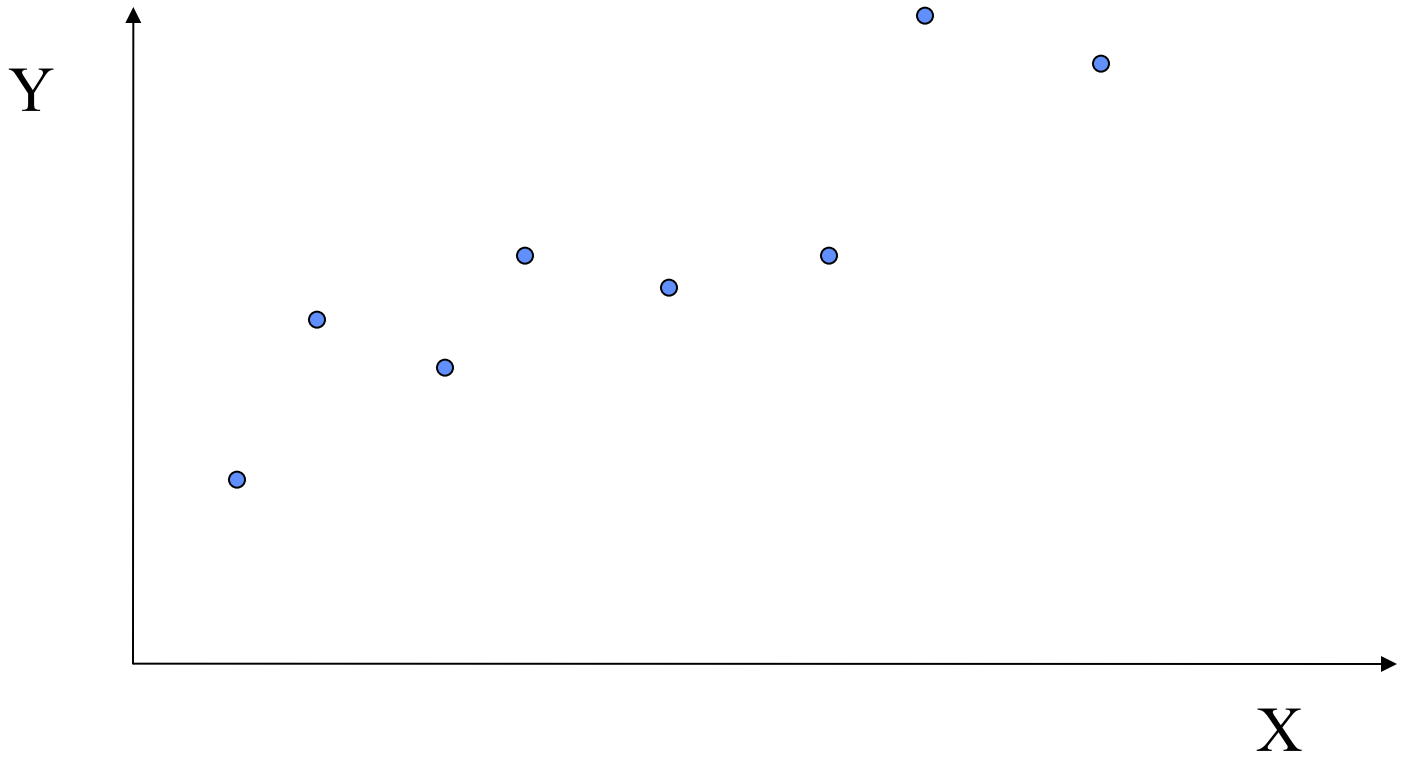
etc.

Overfitting and Underfitting

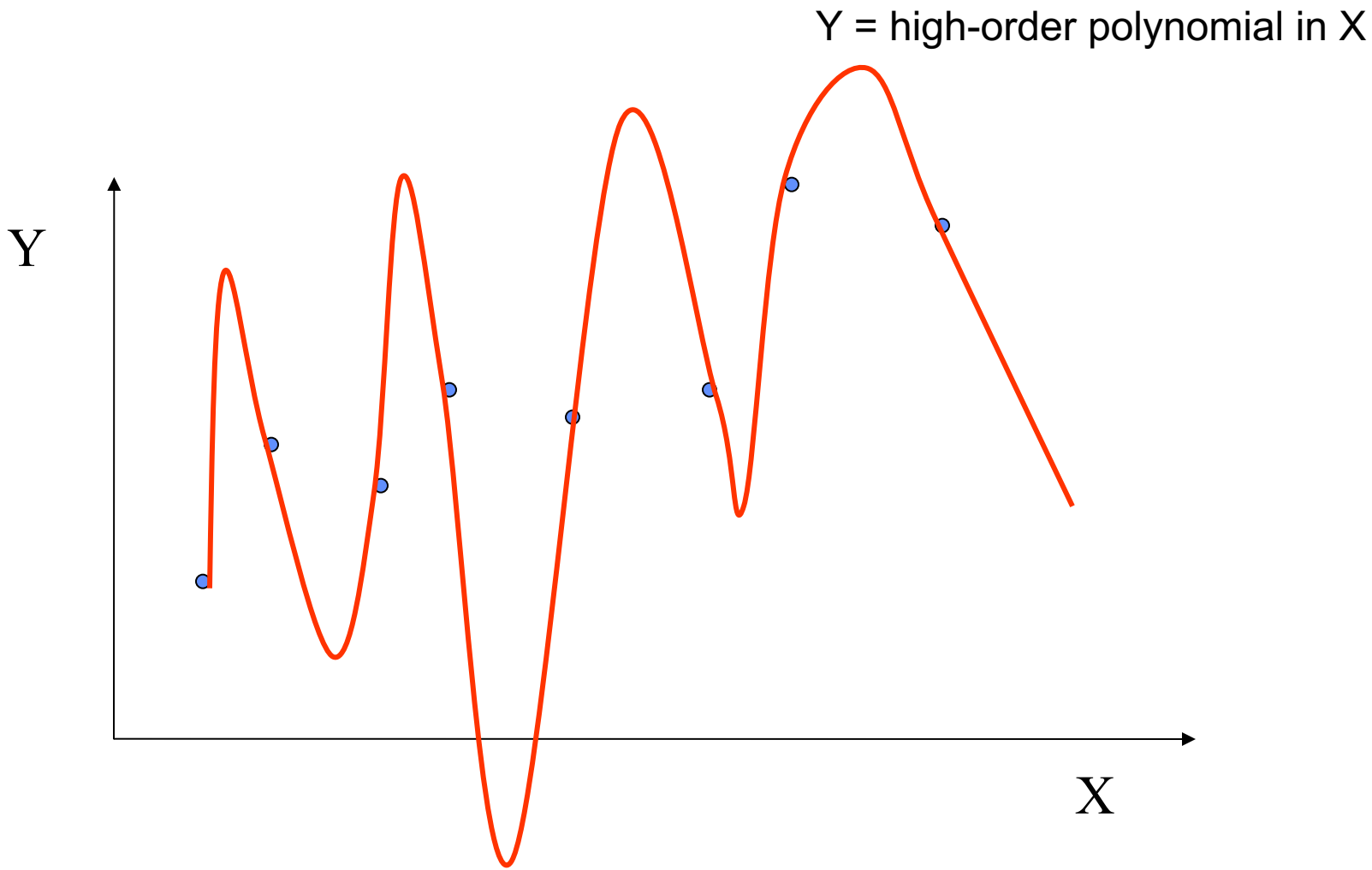
- A model is **overfitting** the training data when the model performs well on the training data but does not perform well on the evaluation data.

- A model is **underfitting** the training data when the model performs poorly on the training data.

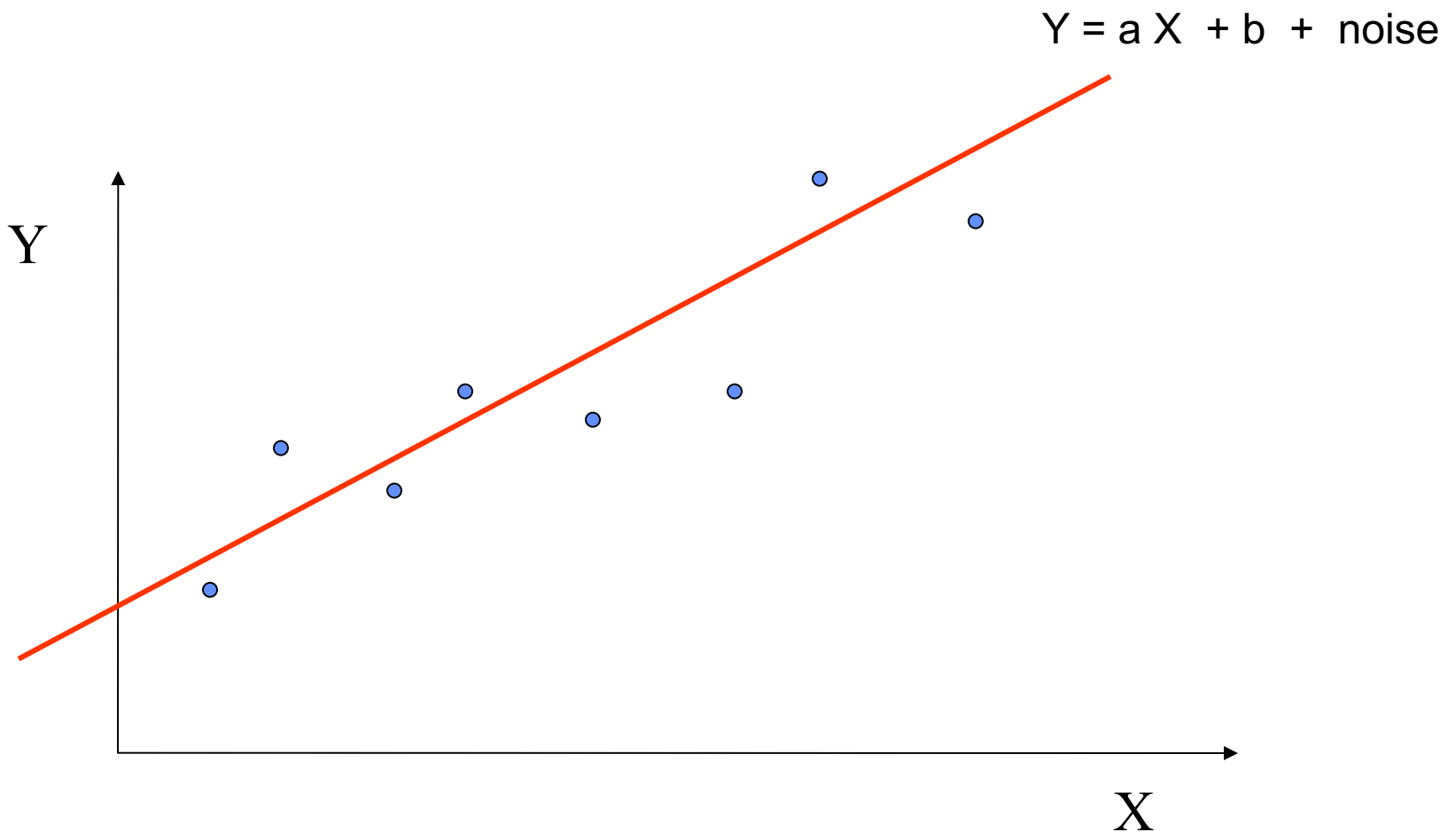
Overfitting and Underfitting



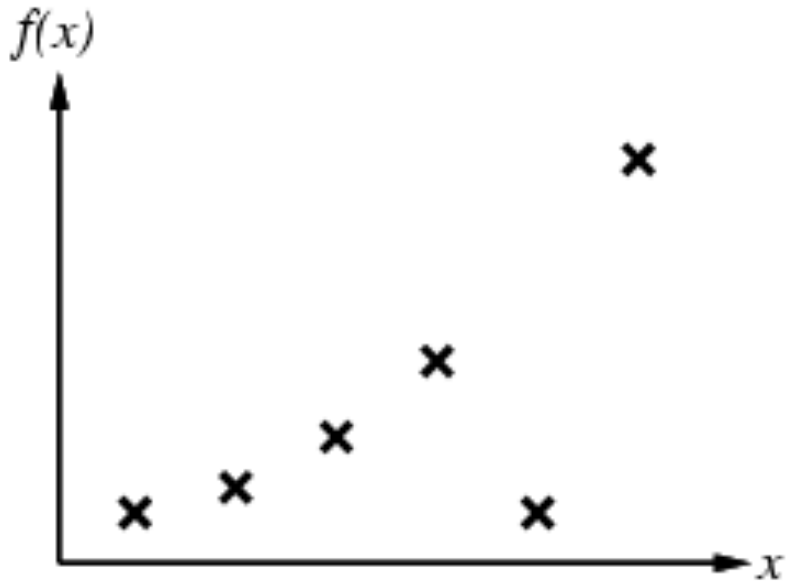
A Complex Model



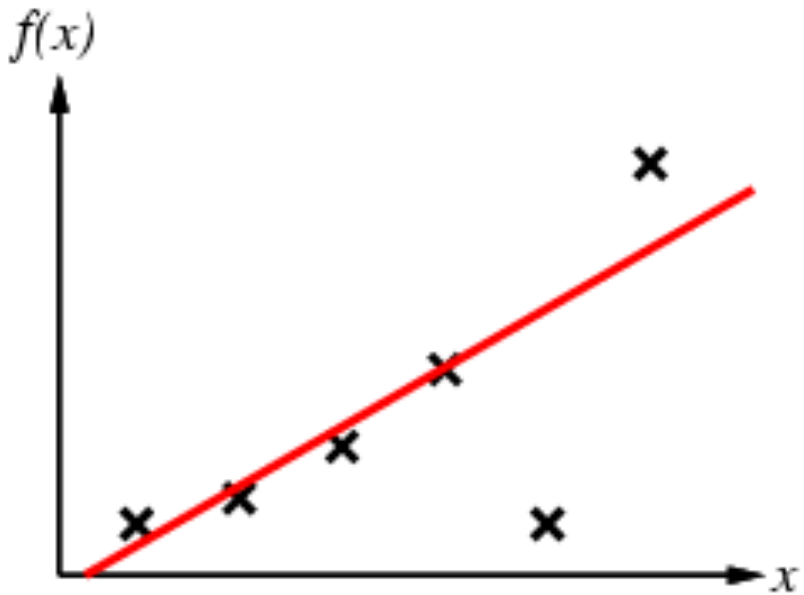
A Much Simpler Model



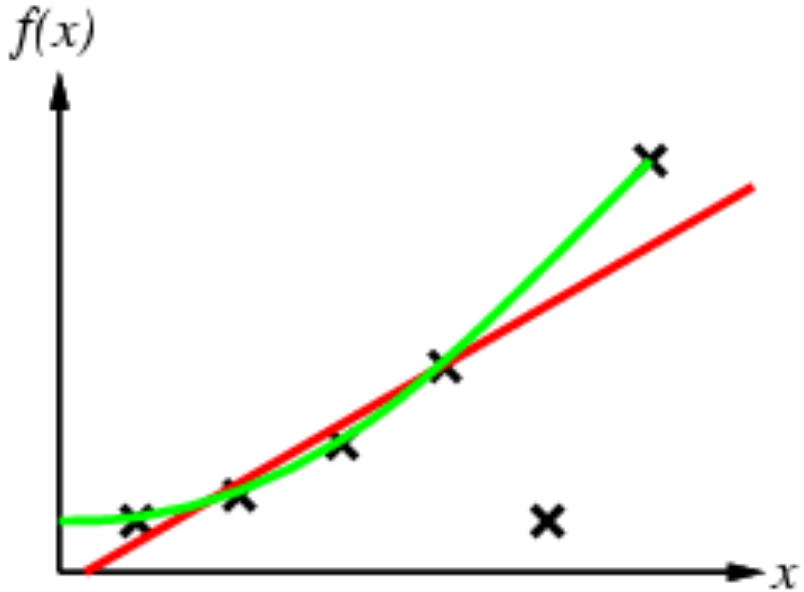
Example 2



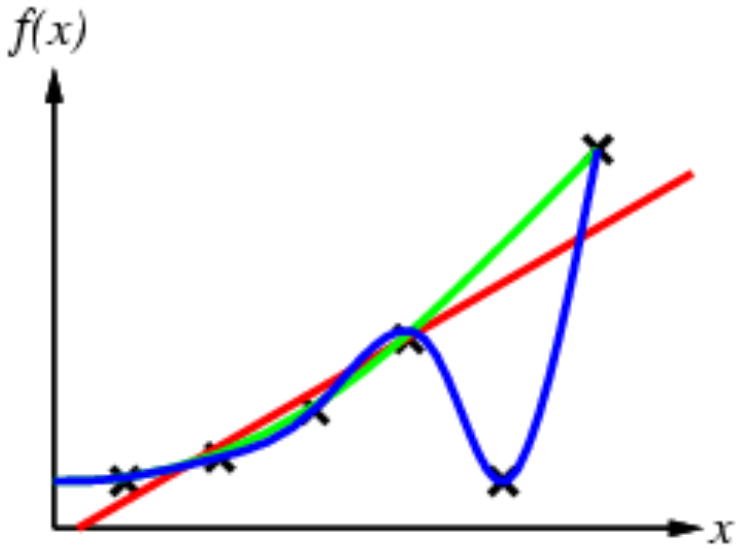
Example 2



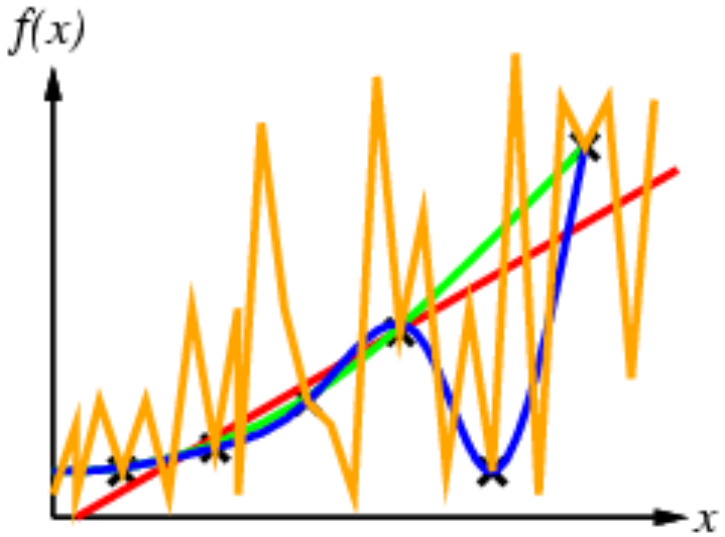
Example 2



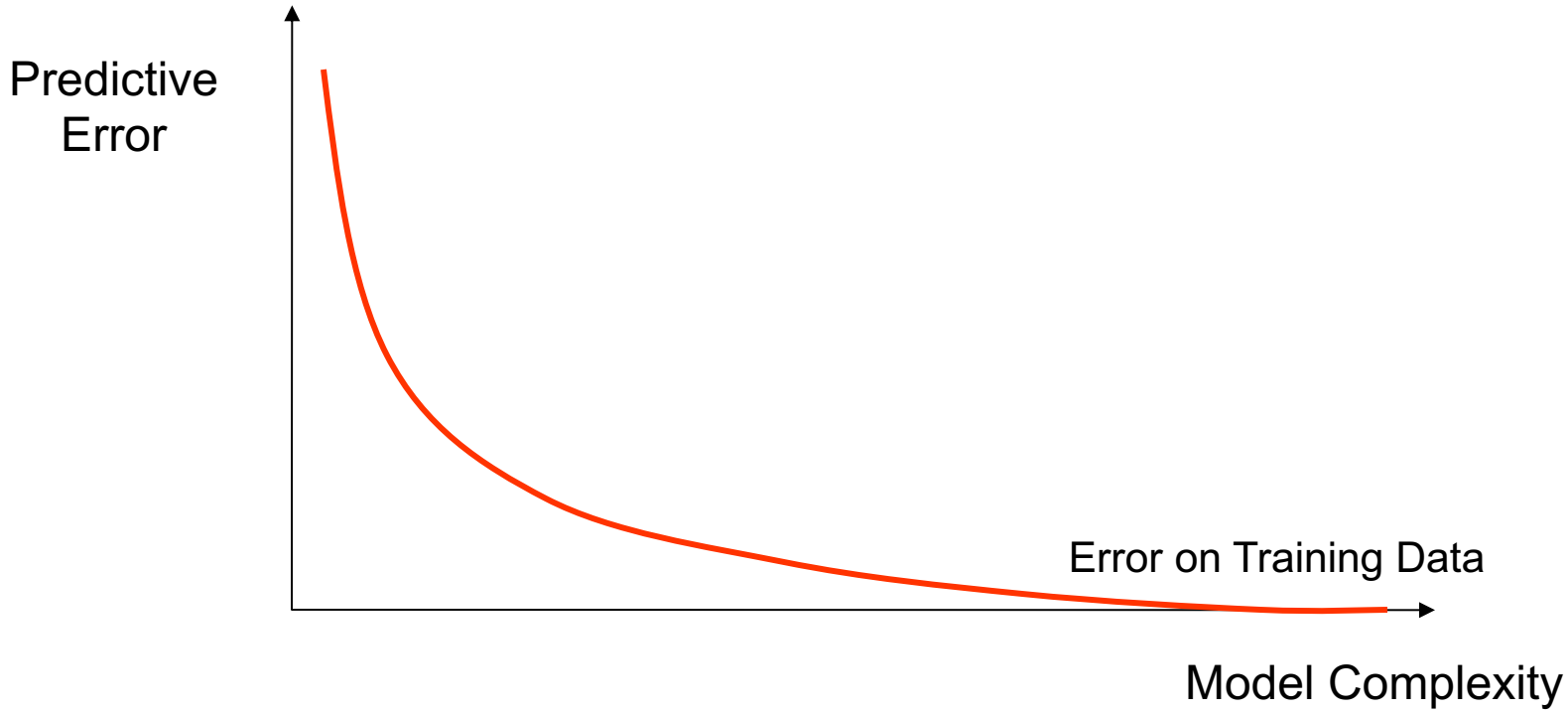
Example 2



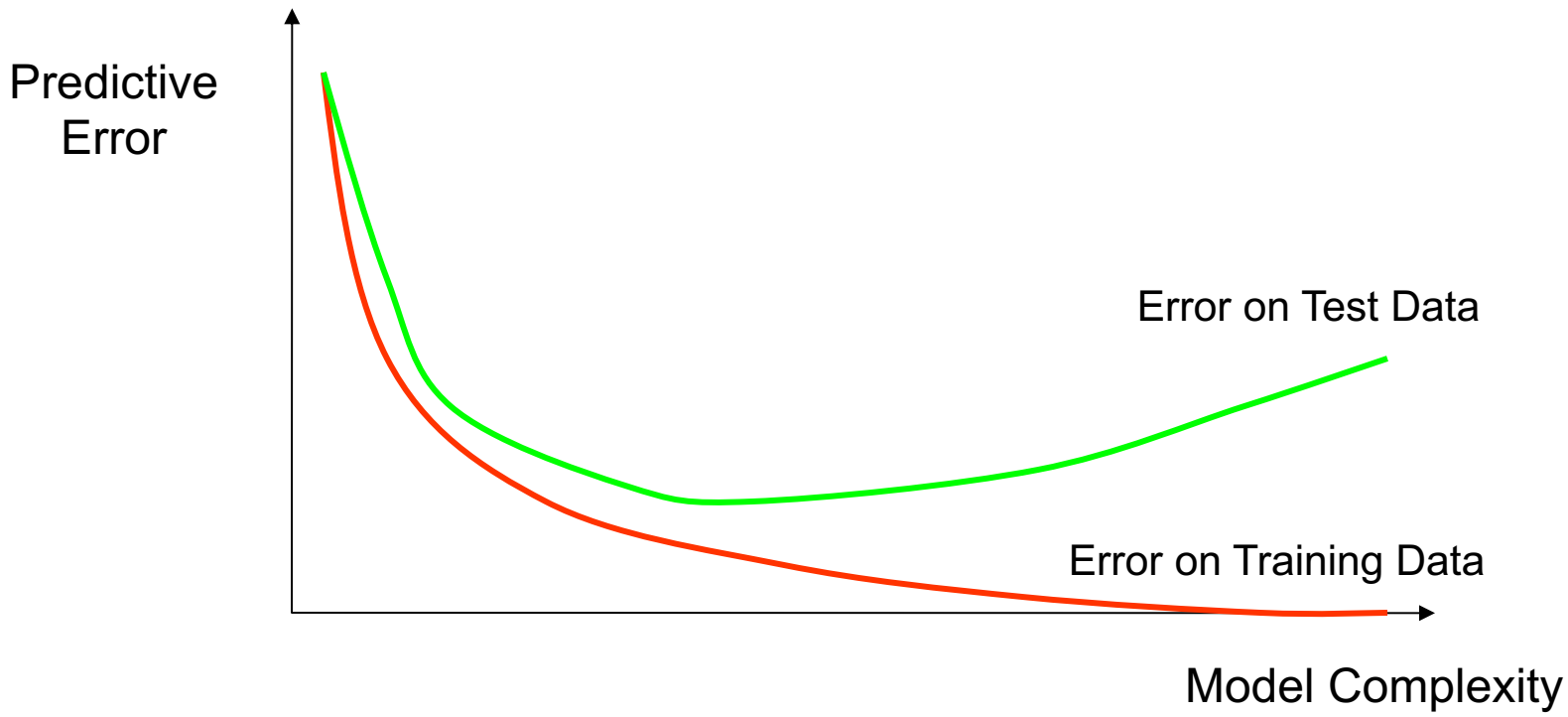
Example 2



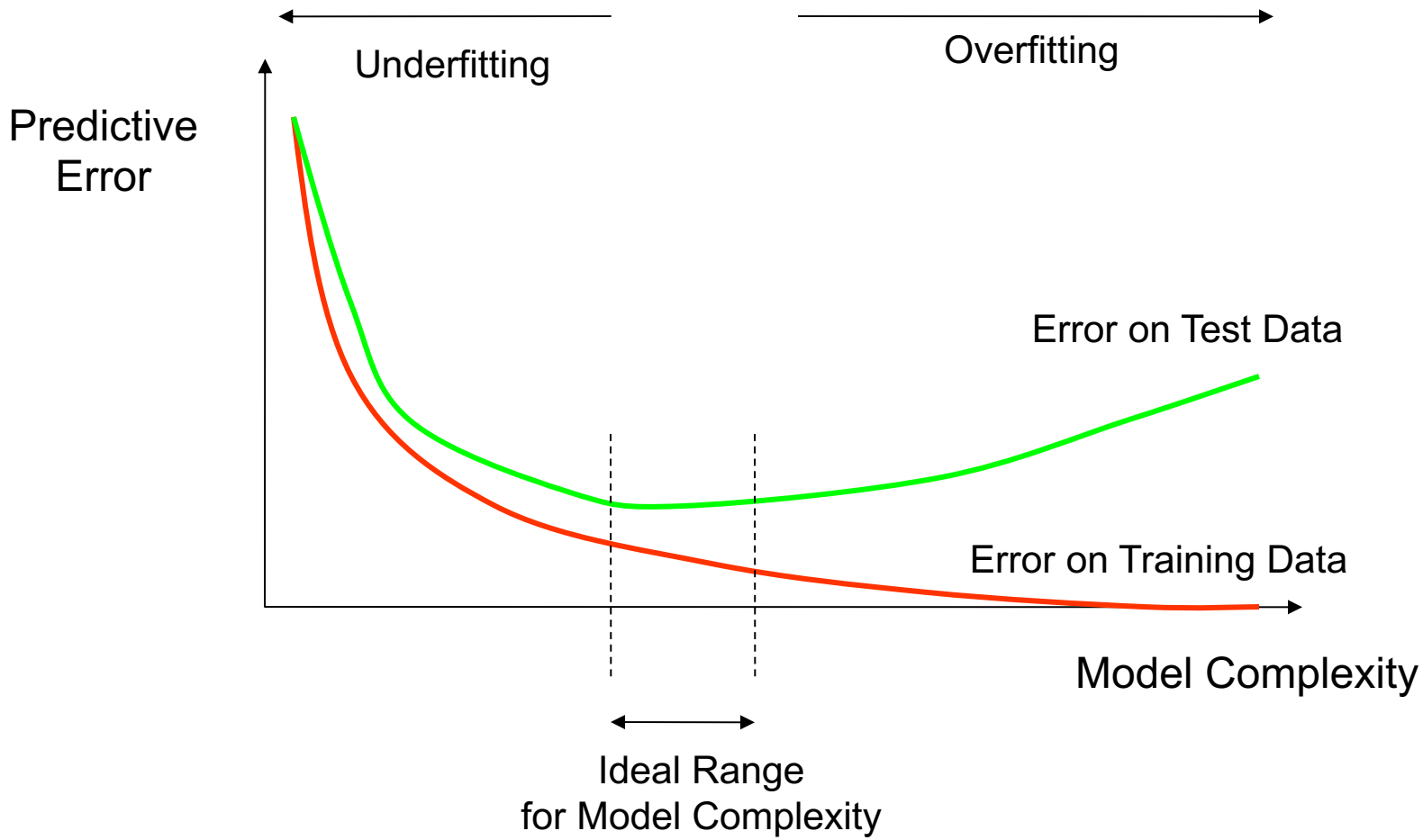
How Overfitting affects Prediction



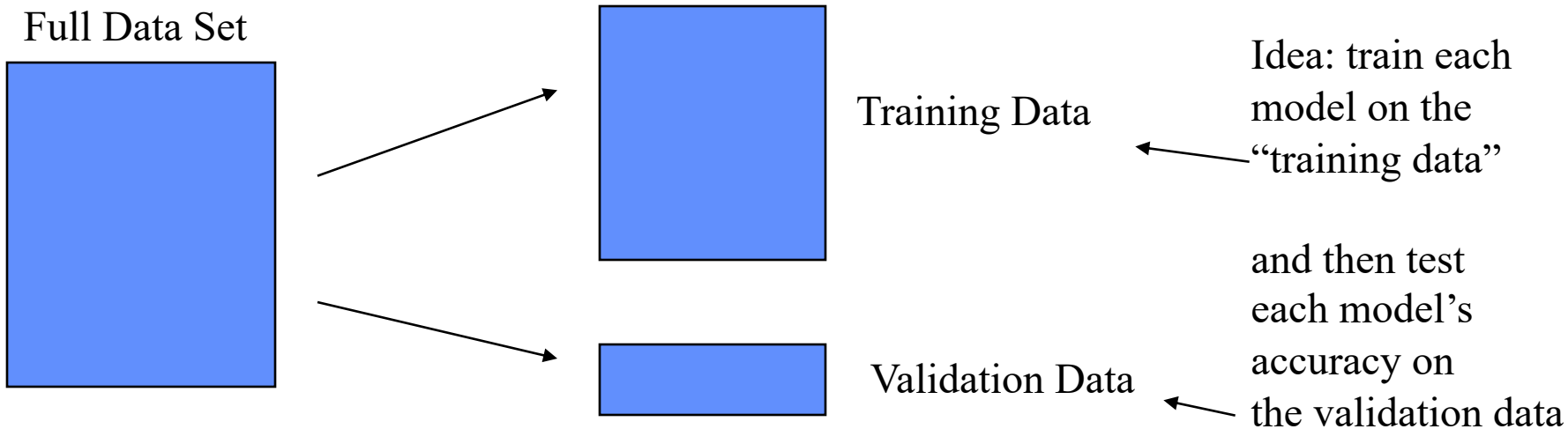
How Overfitting affects Prediction



How Overfitting affects Prediction



Training and Validation Data



The K-fold Cross-Validation Method

- Why just choose one particular 90/10 “split” of the data?
  - In principle we could do this multiple times

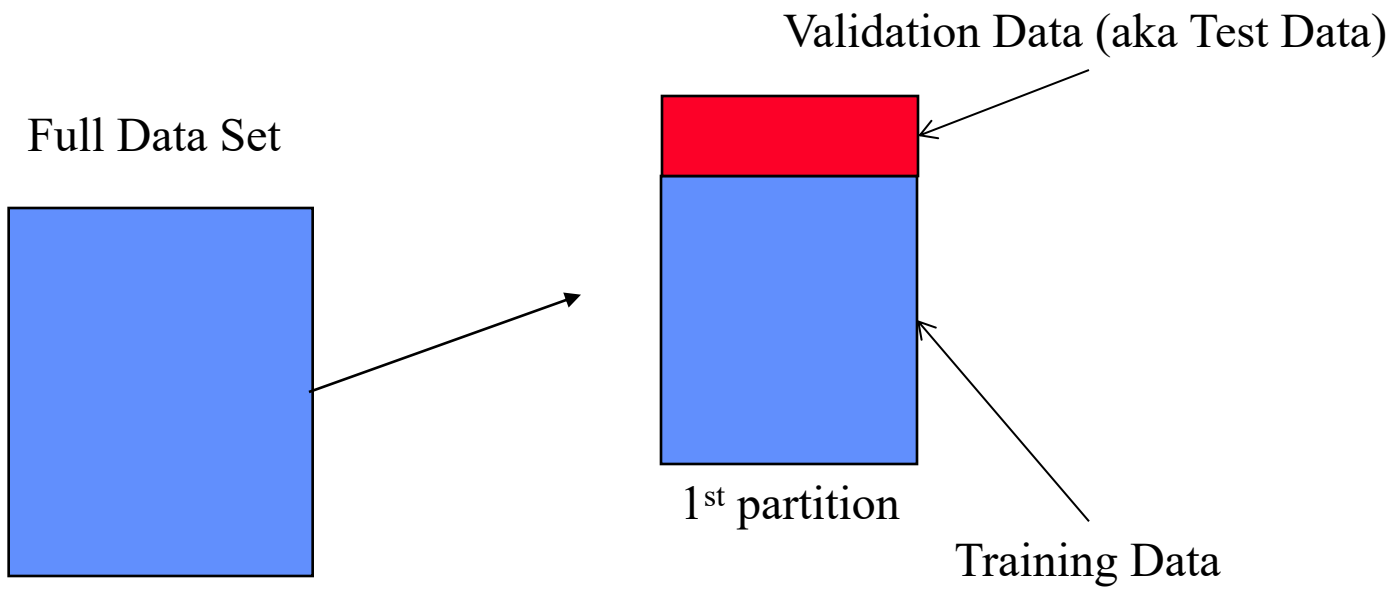
- “K-fold Cross-Validation” (e.g., K=10)
  - Randomly partition full data set into k disjoint subsets (each roughly of size n/v, n = total number of training data points).

for i = 1:10 (here k = 10)
train on 90% of data,
Acc(i) = accuracy on other 10%
end

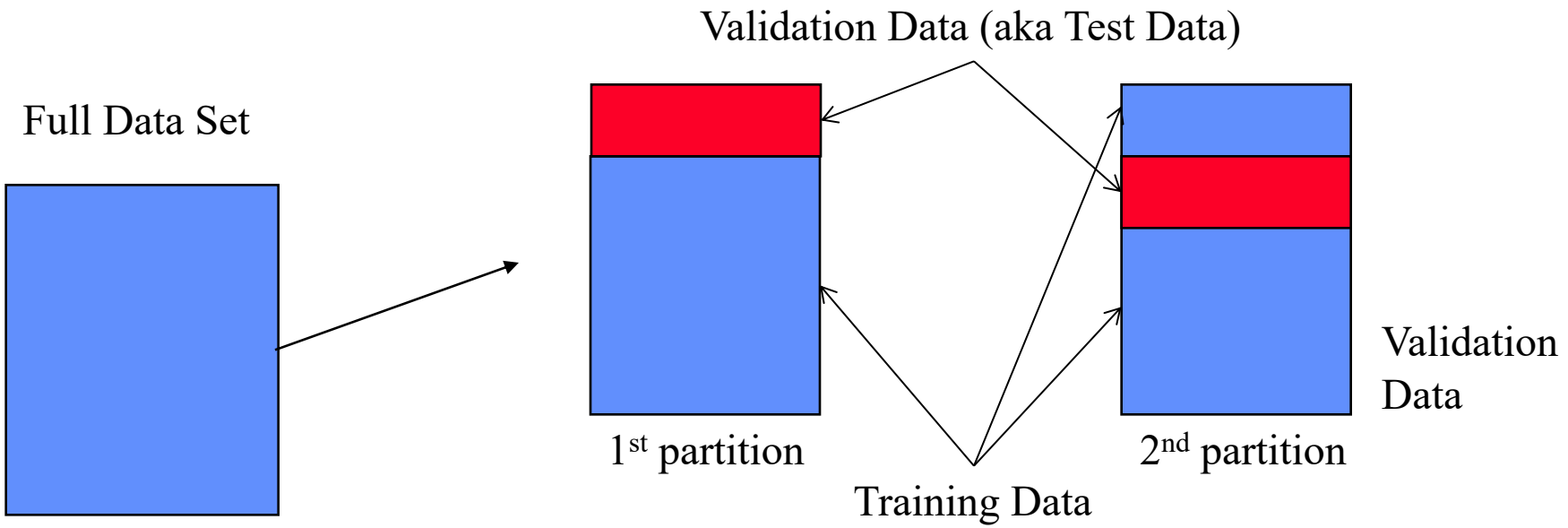
$$\textit{Cross-Validation-Accuracy} = 1/k \sum_i \textit{Acc}(i)$$

  - Choose the method with the highest cross-validation accuracy
  - Common values for k are 5 and 10

Disjoint Validation Data Sets



Disjoint Validation Data Sets



Disjoint Validation Data Sets