Security Analytics
Topic 2: Elements of Data Analysis

Purdue University

Prof. Ninghui Li

Based on slides by Prof. Jenifer Neville and Chris Clifton

Readings

- Reading
  - Chapter 2 of Principles of Data Mining
  - On kNN
    - **K-Nearest Neighbors for Machine Learning by Jason Brownlee**
    - A Complete Guide to K-Nearest-Neighbors with Applications in Python and R from Kevin Zakka's Blog

Key Issues in Data Mining/Machine Learning

- Task specification
- Data representation
- Knowledge representation
- Learning technique
  - Search + scoring
- Prediction and/or interpretation

Key Issues in Data Mining/Machine Learning

- **Task specification**
- Data representation
- Knowledge representation
- Learning technique
  - Search + scoring
- Prediction and/or interpretation

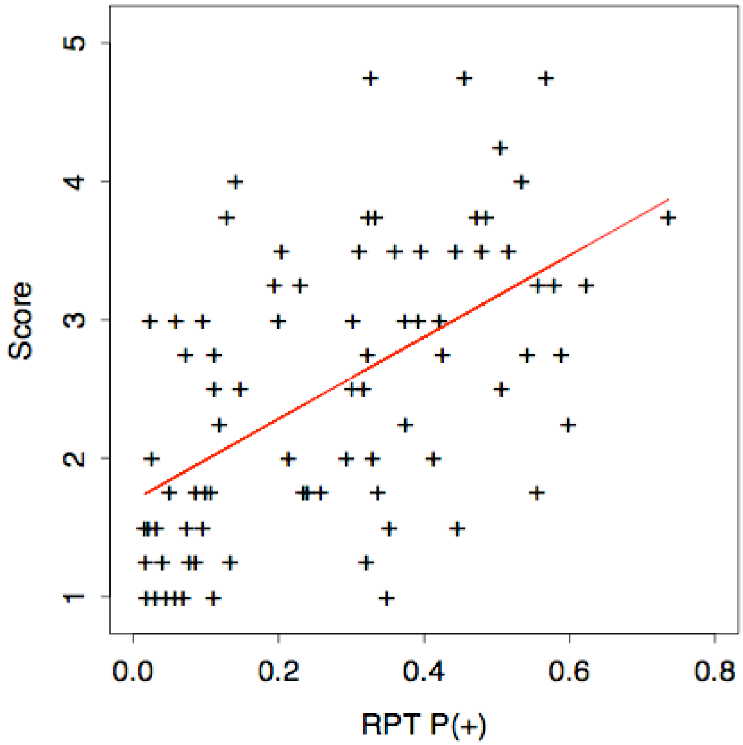
Task specification

- *Objective of the person who is analyzing the data*
- *Description of the characteristics of the analysis and desired result*

- Examples:
  - From a set of *labeled examples*, devise an *understandable model* that will *accurately predict* whether a stockbroker will commit fraud in the near future.
  - From a set of *unlabeled examples*, cluster stockbrokers into a *set of homogeneous groups* based on their demographic information

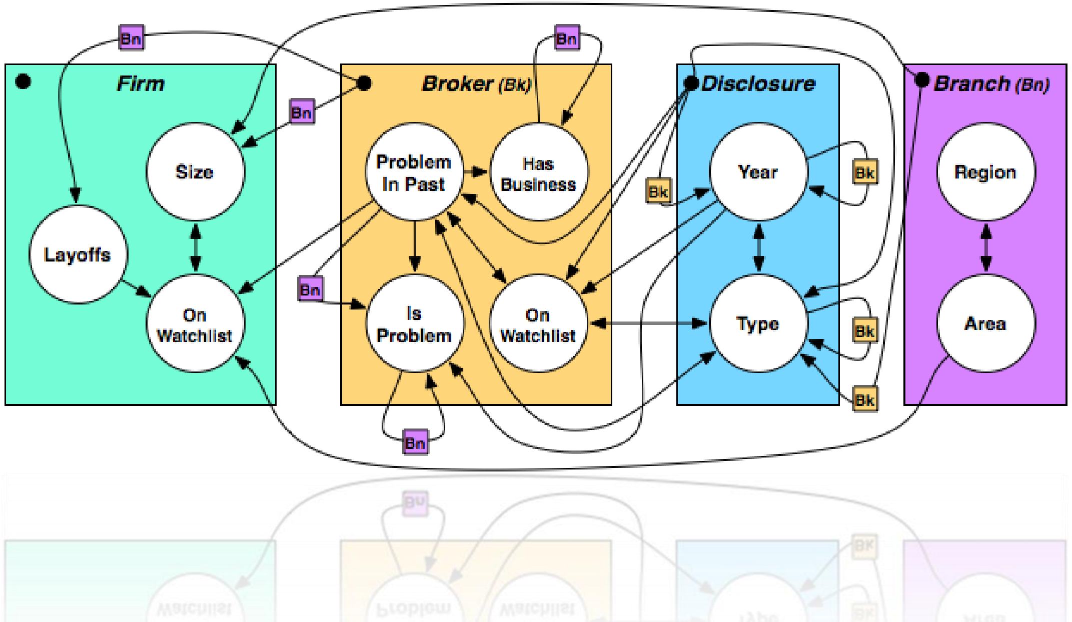
Exploratory data analysis

- Goal
  - Interact with data without clear objective
- Techniques
  - Visualization,
  - adhoc modeling
  - Adhoc querying/digging



Descriptive modeling

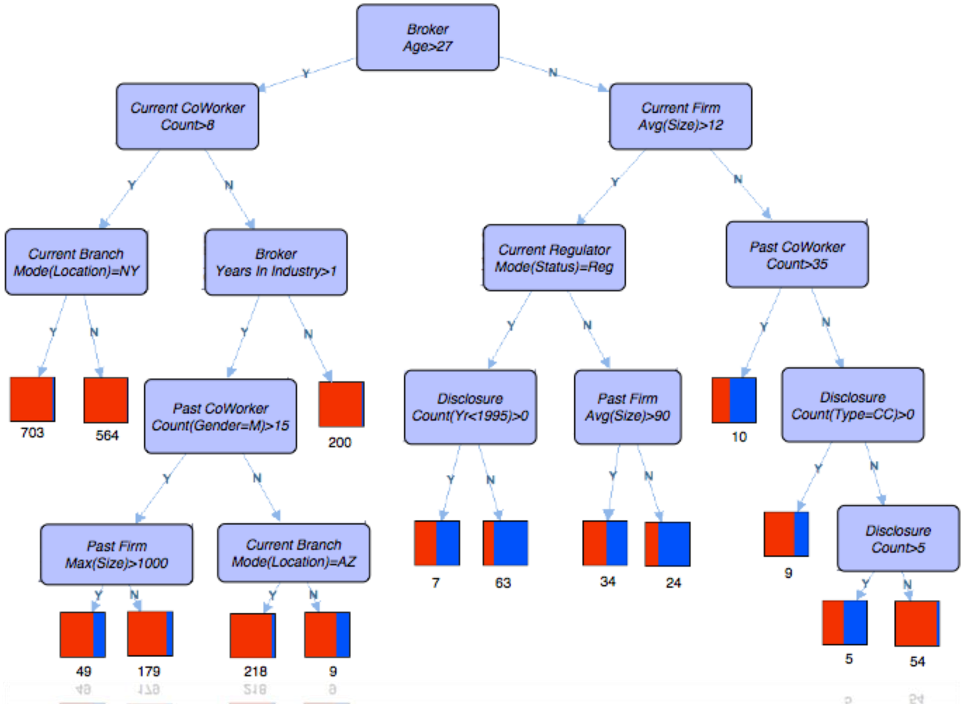
- Goal
  - Summarize the data
    or the underlying generative process
- Techniques
  - Density estimation, cluster analysis and segmentation



Also known as: **unsupervised** learning

Predictive modeling

- Goal
  - Learn model to predict unknown class label values given observed attribute values
- Techniques
  - Classification,

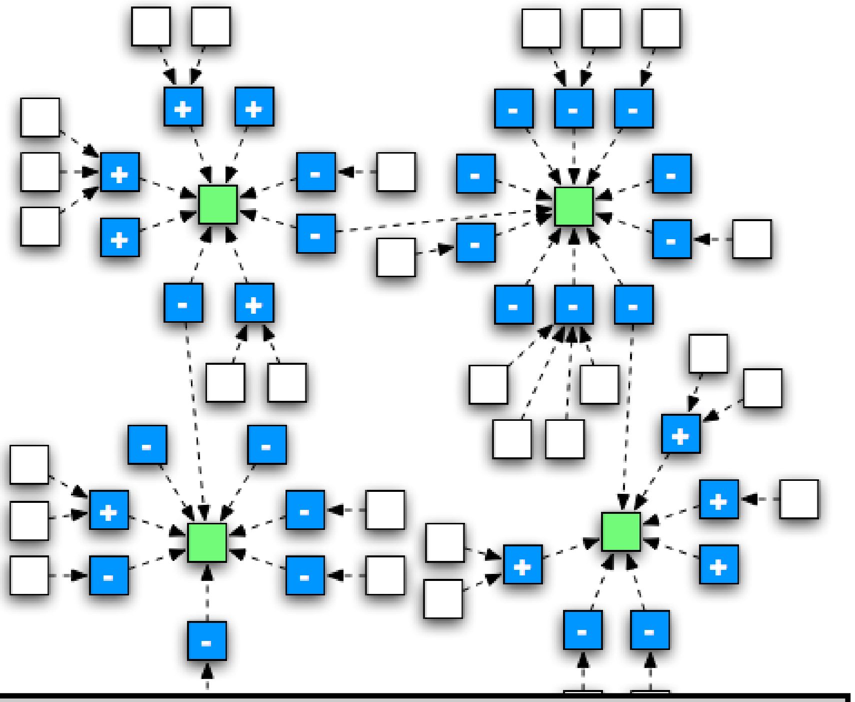


Broker Age>27
Y
N
Current CoWorker Count>8
Current Firm Avg(Size)>12
Current Branch Mode(Location)=NY
Broker Years In Industry>1
Current Regulator Mode(Status)=Reg
Past CoWorker Count>35
703
564
Past CoWorker Count(Gender=M)>15
200
Disclosure Count(Yr<1995)>0
Past Firm Avg(Size)>90
10
Disclosure Count(Type=CC)>0
Past Firm Max(Size)>1000
Current Branch Mode(Location)=AZ
7
63
34
24
9
Disclosure Count>5
49
179
218
9
5
54

Also known as: **supervised** learning

Pattern discovery

- Goal
  - Detect patterns and rules that describe sets of examples
- Techniques
  - Association rules, graph mining, anomaly detection



Model: global summary of a data set
Pattern: local to a subset of the data

Key Issues in
Data Mining/Machine Learning

- Task specification
- **Data representation**
- Knowledge representation

- Learning technique
  - Search + scoring
- Prediction and/or interpretation

Data representation

- *Choice of **data structure** for representing individual and collections of measurements*

- Individual measurements: single observations (e.g., person’s date of birth, product price)
- Collections of measurements: sets of observations that describe an instance (e.g., person, product)
- Choice of representation determines applicability of algorithms and can impact modeling effectiveness
- Additional issues: data sampling, data cleaning, feature construction

Individual measurements

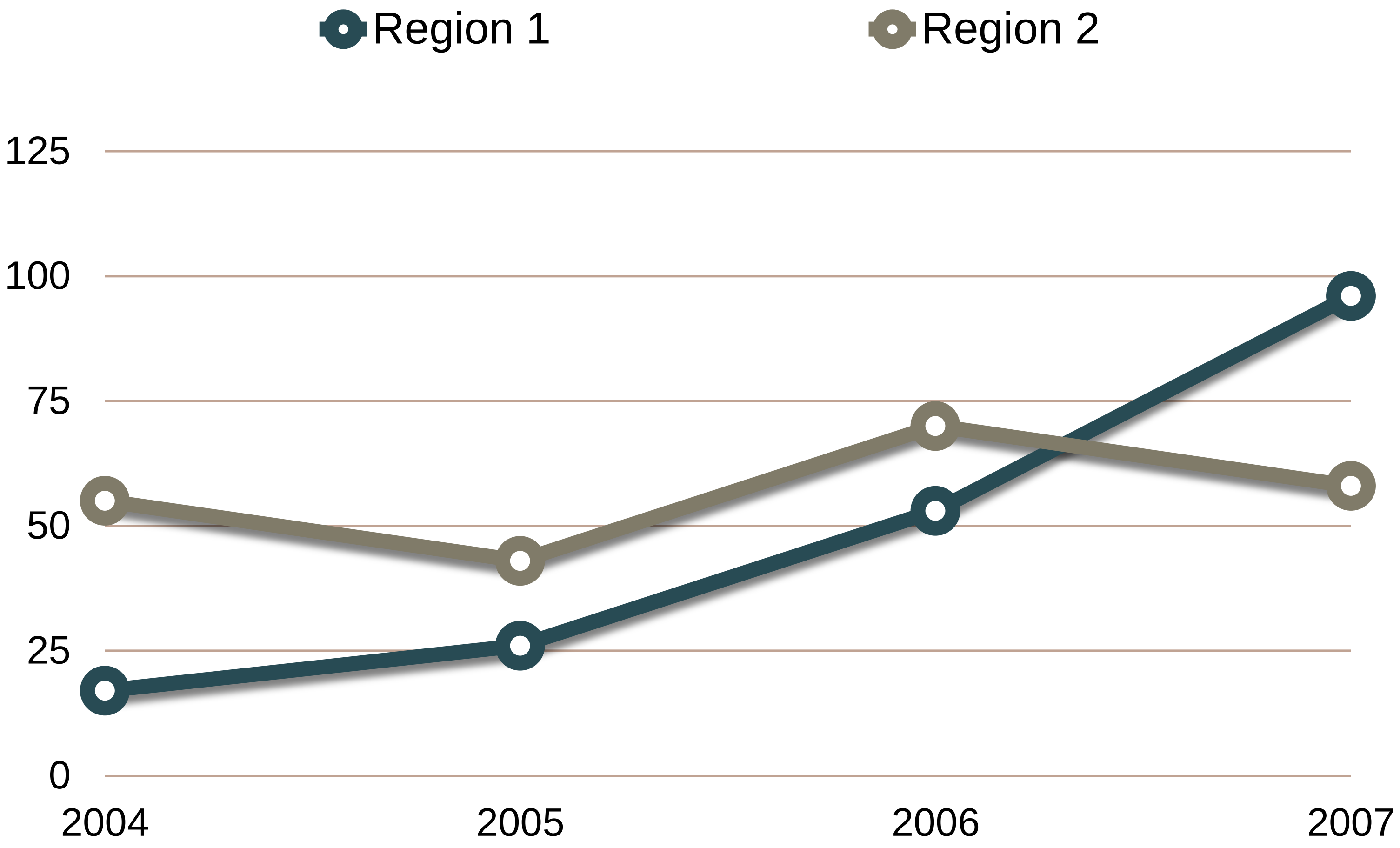
- Unit measurements:
  - Discrete values — categorical or ordinal variables
  - Continuous values — interval and ratio variables
- Compound measurements:
  - < x, y >
  - < value, time >

Data representation: Table/vectors

Fraud	**Age**	**Degree**	**StartYr**	**Series7**
+	22	Y	2005	N
-	25	N	2003	Y
-	31	Y	1995	Y
-	27	Y	1999	Y
+	24	N	2006	N
-	29	N	2003	N

N instances X *p* attributes

Data representation: Time series/sequences



Key Issues in
Data Mining/Machine Learning

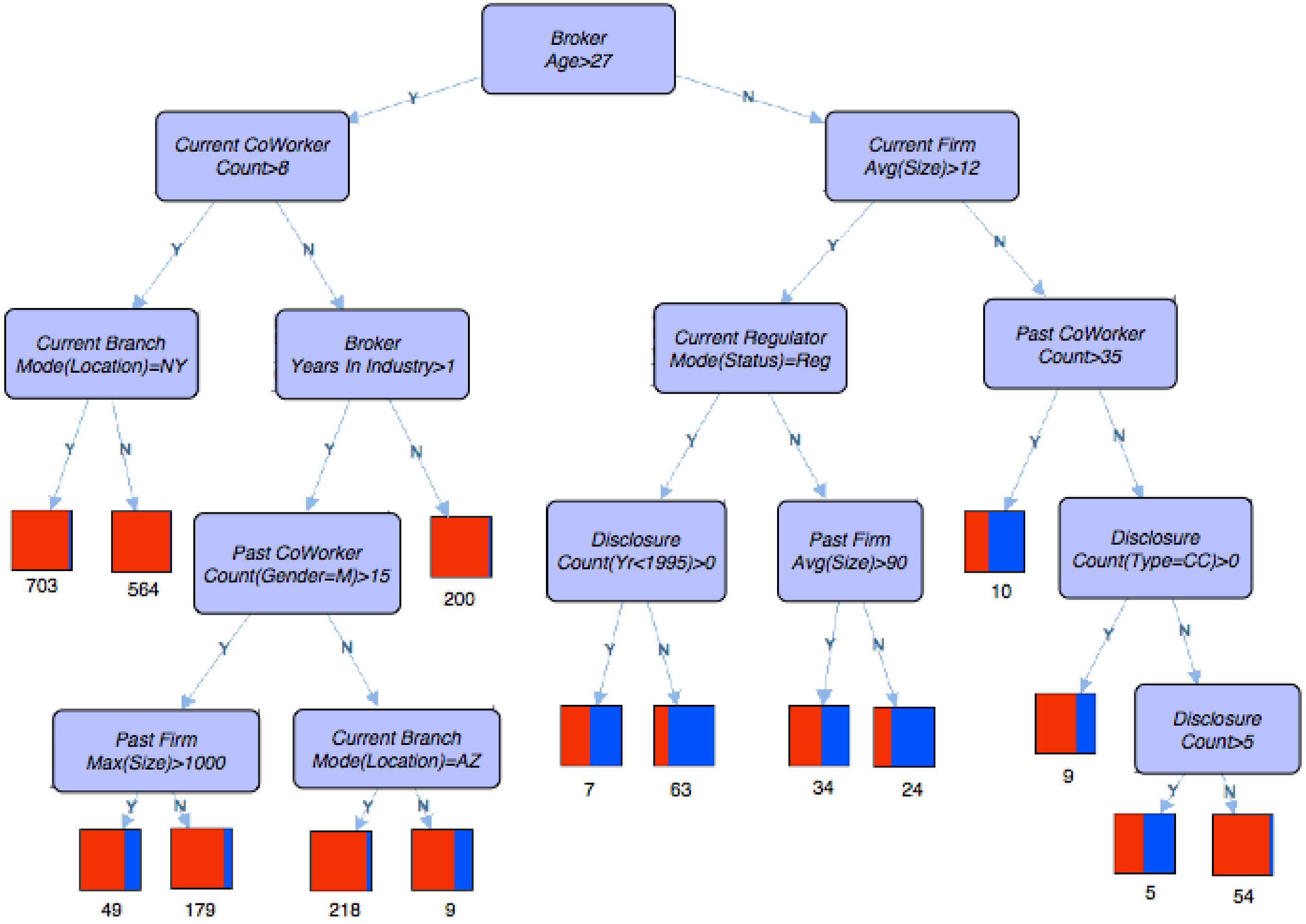
- Task specification
- Data representation
- **Knowledge representation**
- Learning technique
  - Search + scoring
- Prediction and/or interpretation

Knowledge representation

- *Underlying structure of the model or patterns that we seek from the data*
  - Specifies the models/patterns that could be returned as the results of the data mining algorithm
  - Defines the **model space** that algorithms search over (i.e., all possible models/patterns)

- Examples:
  - **If-then rule**
    If short closed car then toxic chemicals
  - **Conditional probability distribution**
    P(fraud | age, degree, series7, startYr)
  - **Decision tree**

Knowledge representation: Classification tree



Broker Age>27
Y
N
Current CoWorker Count>8
Current Firm Avg(Size)>12
Y
N
Y
N
Current Branch Mode(Location)=NY
Broker Years In Industry>1
Current Regulator Mode(Status)=Reg
Past CoWorker Count>35
Y
N
Y
N
Y
N
Y
N
703
564
Past CoWorker Count(Gender=M)>15
200
Disclosure Count(Yr<1995)>0
Past Firm Avg(Size)>90
10
Disclosure Count(Type=CC)>0
Y
N
Y
N
Y
N
Y
N
Past Firm Max(Size)>1000
Current Branch Mode(Location)=AZ
7
63
34
24
9
Disclosure Count>5
Y
N
Y
N
Y
N
49
179
218
9
5
54

Each node corresponds to a feature; each leaf a class label or probability distribution

Knowledge representation: Regression model

$$y = \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_0$$

- X are predictor variables
- Y is response variable
- Example:
  - Predict number of disclosures given income and trading history

Key Issues in Data Mining/Machine Learning

- Task specification
- Data representation
- Knowledge representation
- **Learning technique**
  - Search + scoring
- Prediction and/or interpretation

Learning technique

- Method to construct model or patterns from data
- **Model space**
  - Choice of knowledge representation defines a set of possible models or patterns
- **Scoring function**
  - Associates a numerical value (score) with each member of the set of models/patterns
- **Search technique**
  - Defines a method for generating members of the set of models/patterns and determining their score

Scoring function

- *A numeric score assigned to each possible model in a search space,* ***given a reference/input dataset***
  - Used to judge the quality of a particular model for the domain

- Score function are statistics—estimates of a population parameter based on a sample of data
- Examples:
  - Misclassification
  - Squared error
  - Likelihood

Parameter estimation vs. structure learning

- Models have both **parameters** and **structure**
- Parameters:
  - Coefficients in regression model
  - Feature values in classification
  - Probability estimates in graphical model

Search: Convex/smooth optimization techniques

- Structure:
  - Variables in regression model
  - Nodes in classification tree
  - Edges in graphical model

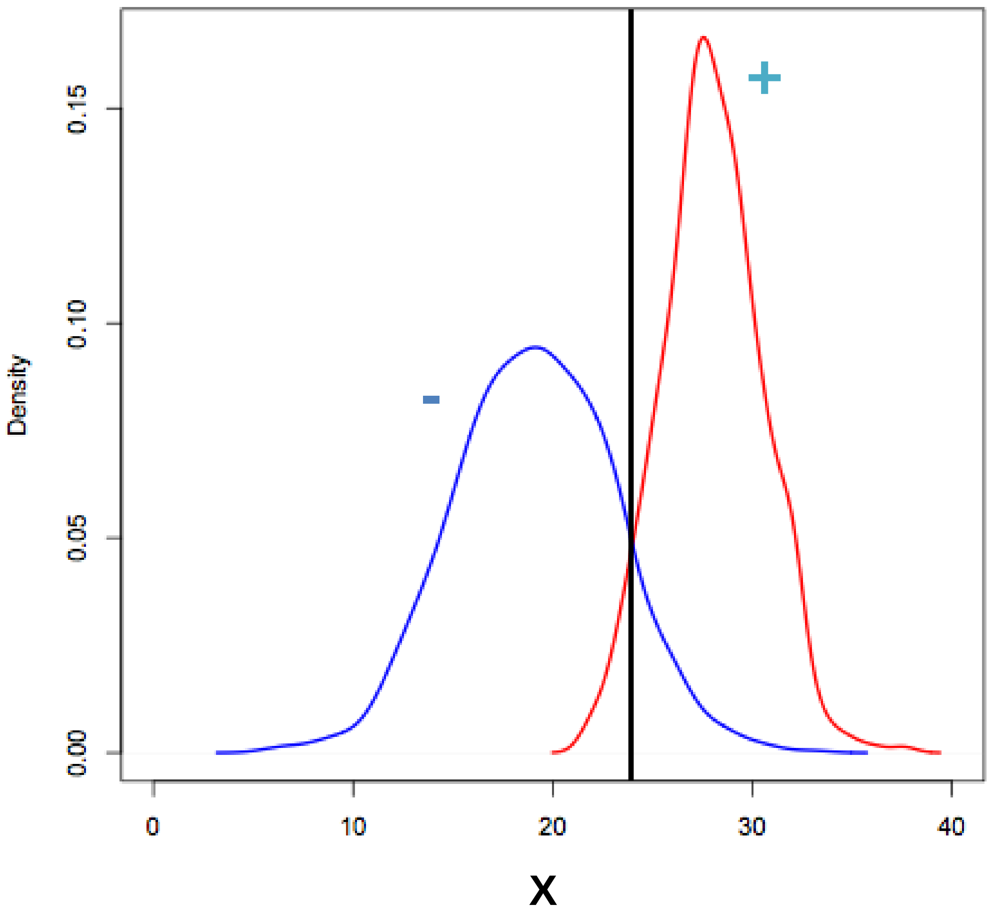
Search: Heuristic approaches for combinatorial optimization

Example learning problem

Task: Devise a rule to classify items based on the attribute ***X***

Knowledge representation: If-then rules

Example rule:
If x > 25 then +
Else -



What is the model space?

All possible thresholds

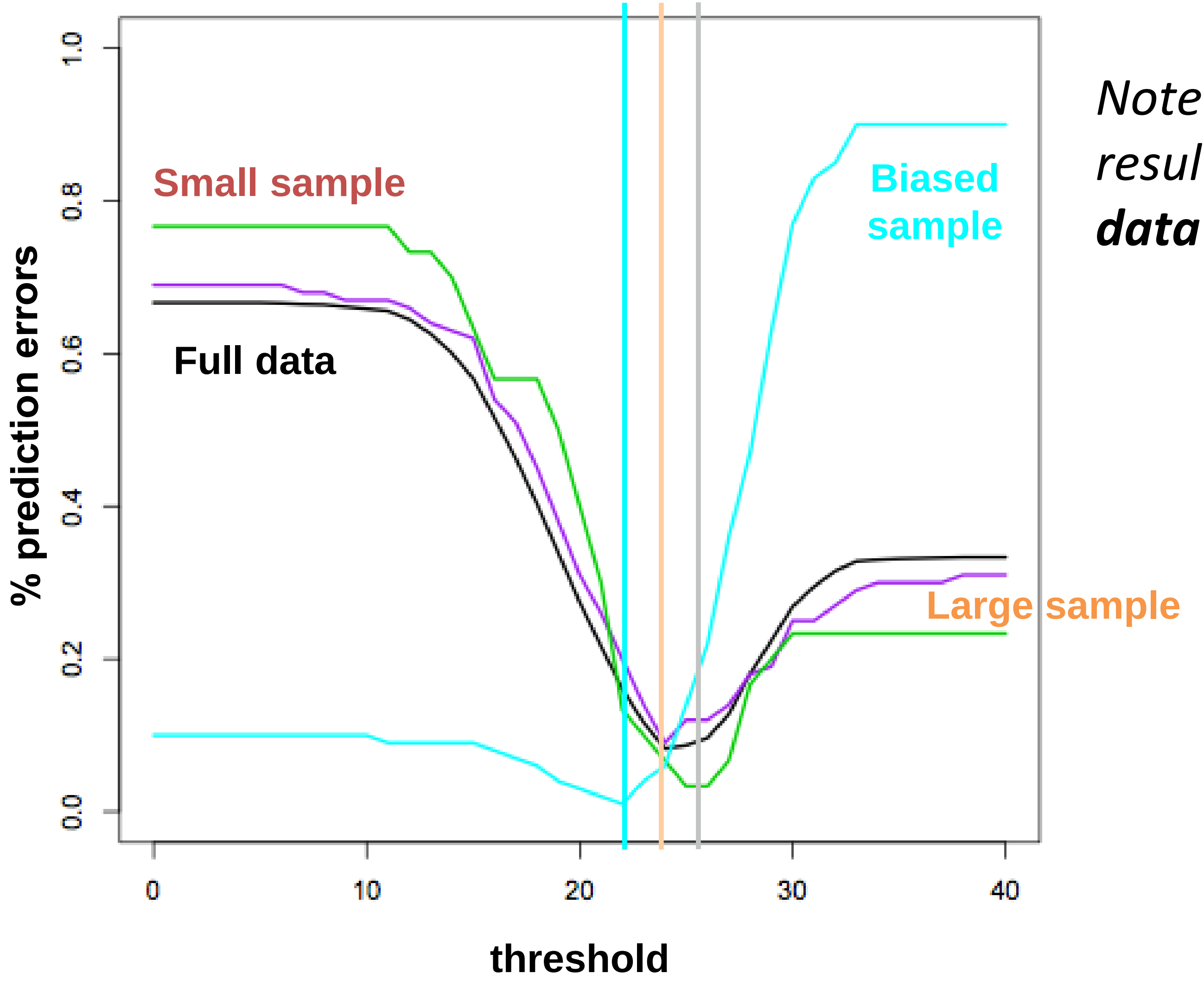
What score function?

Prediction error rate

Score function over model space

Search procedure?

Try all thresholds, select one with lowest score



Note: learning result depends on ***data***

Key Issues in Data Mining/Machine Learning

- Task specification
- Data representation
- Knowledge representation

- Learning technique
  - Search + Evaluation
- **Prediction and/or interpretation**

Inference and interpretation

- Prediction technique
  - Method to apply learned model to new data for prediction/analysis
  - Only applicable for predictive and some descriptive models
  - Prediction is often used during **learning** (i.e., search) to determine value of scoring function
  - Generalization: how well a learned model perform on previously unseen data

Interpretation

- Interpretation of results from tasks such as Pattern Discovery
  - Objective: significance measures
  - Subjective: importance, interestingness, novelty

Recap: Key Issues in Data Mining/Machine Learning

- Task specification
- Data representation
- Knowledge representation

- Learning technique
  - Search + scoring
- Prediction and/or interpretation

Your First Classifier!

- Let's consider one of the simplest classifiers out there.
- Assume we have a training set $(x_1,y_1)\ldots(x_n,y_n)$
- Now we get a new instance x_{new} , how can we classify it?
  - Example: Can you recommend a movie, based on user's movie reviews?
- **Simple Solution**:
  - Find the most similar example (x,y) in the training data and predict the same
    - If you liked "*Fast and Furious*" you'll like "*2 fast 2 furious*"
- One key decision is needed: distance metric to compute similarity

On Distance Metrics

- Distance (or equivalently, similarity) measures are used by many data analysis tasks
  - Clustering, nearest neighbors
- How to measure similarity/distance
  - From humans/experts.
  - From data characteristics
- What is a metric?
  - Non-negativity: $d\left(x^{(i)}, x^{(j)}\right) \geq 0$
  - Identity: $d\left(x^{(i)}, x^{(i)}\right) = 0$
  - Symmetry: $d(x^{(i)}, x^{(j)}) = d(x^{(j)}, x^{(i)})$
  - Triangle inequality: $d\left(x^{(i)}, x^{(j)}\right) \leq d\left(x^{(i)}, x^{(k)}\right) + d\left(x^{(k)}, x^{(i)}\right)$

Euclidean (L_2) Distance

- Euclidean distance
  - Assume each data point is a n-dimensional vector
  - Given two vectors $\langle x^{(i)}{}_{1,} \cdots, x^{(i)}{}_{n} \rangle$, $\langle x^{(j)}{}_{1,} \cdots, x^{(j)}{}_{n} \rangle$,
  - Euclidean Distance is $\sqrt{\sum_{k=1}^{n} (x^{(i)}{}_{k} - x^{(j)}{}_{k})^2}$
- What are the implied assumptions?
  - There are some degree of *commensurability* between the different variables (including units)

Euclidean Distance

- What if different variables are not commensurable
  - Dividing each variable by its standard deviation
  - Adding weights to the different variables
  - Normalize using covariance
  - Use dimensionality reduction techniques such as Principal Component Analysis

Minkowski or L_p metric

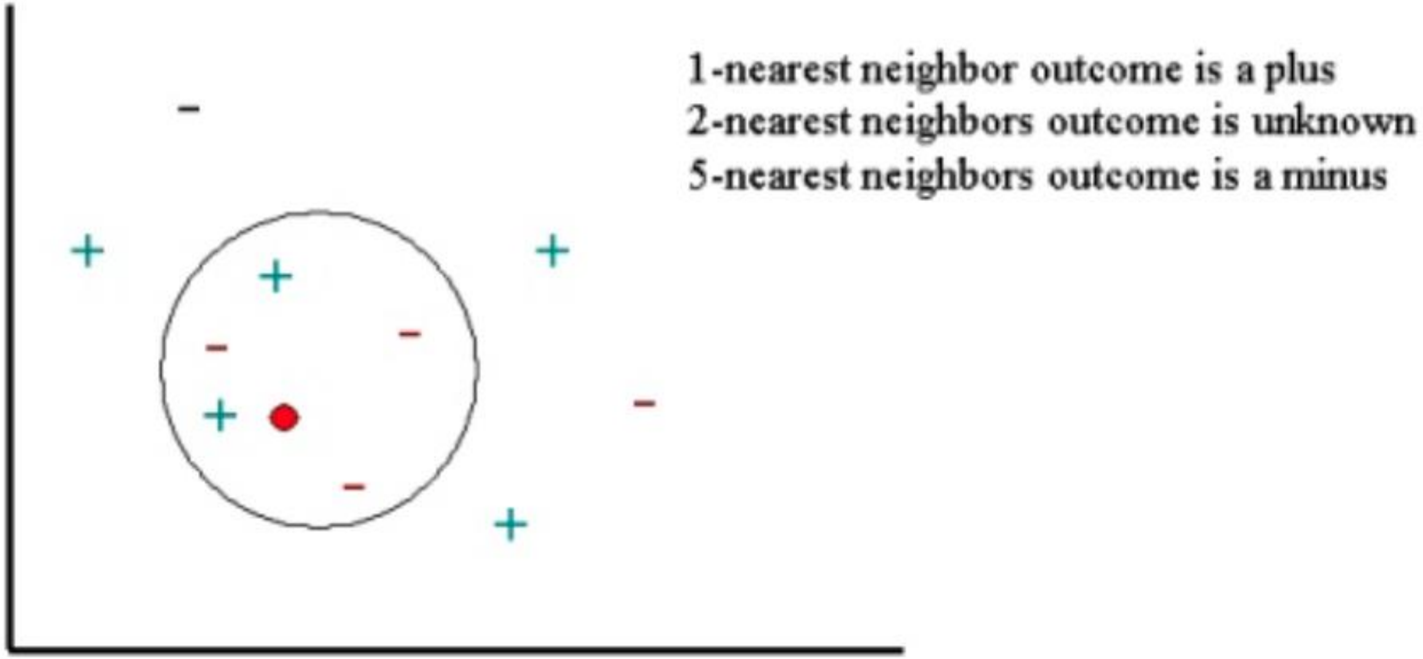
- Given two vectors $\langle x^{(i)}{}_{1,} \cdots, x^{(i)}{}_{n} \rangle$, $\langle x^{(j)}{}_{1,} \cdots, x^{(j)}{}_{n} \rangle$,
- Minkowski Distance is a family of defined as $\sqrt[p]{\sum_{k=1}^{n} |x^{(i)}{}_{k} - x^{(j)}{}_{k}|^{p}}$
- What if p=1?
- **Manhattan distance** or city-block distance
- What other p are often used?

Jaccard Distance

- When attributes are binary, how to measure distance?
- Jaccard distance
- $d_J(A, B) = 1 - J(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}$
- Where $J(A, B)$ is also known as **Intersection over Union** and the **Jaccard similarity coefficient**

K Nearest Neighbors

- We can make the decision by looking at several near examples, not just one. **Why would it be better**?



1-nearest neighbor outcome is a plus
2-nearest neighbors outcome is unknown
5-nearest neighbors outcome is a minus

K Nearest Neighbors

- **Learning**: just storing the training examples
- **Prediction**:
  - Find the K training example closest to **x**
- **Predict a label:**
  - Classification: majority vote
  - Regression: mean value
- KNN is a type of *instance based learning*
  - Store instances seen in training, and in prediction time compare new instances to the stored ones.
- This is called *lazy* learning, since most of the computation is done at prediction time

Let’s analyze KNN

- ***What are the advantages and disadvantages of KNN?***
  - *What should we care about when answering this question?*
- ***Complexity***
  - ***Space*** *(how memory efficient is the algorithm?)*
    - *Why should we care?*
  - ***Time*** *(computational complexity)*
    - *Both at training time and at test (prediction) time*
- ***Expressivity***
  - *What kind of functions can we learn?*

Let’s analyze KNN

- ***What are the advantages and disadvantages of KNN?***
  - *What should we care about when answering this question?*
- ***Complexity***
  - ***Space*** *(how memory*
    - *Why should we care*

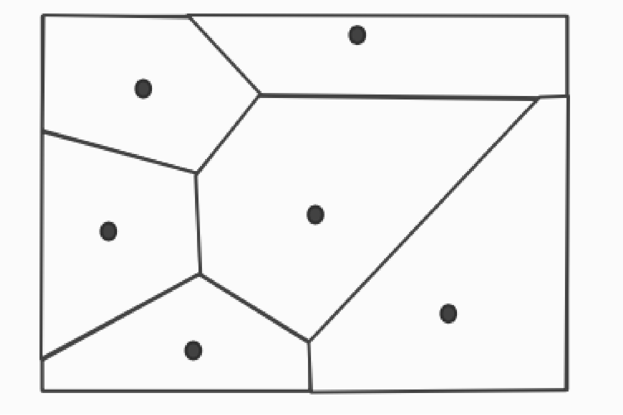
KNN needs to maintain all training examples!
-Datasets can be HUGE

  - ***Time*** *(computational com*
    - *Both at training time and*

Training is very fast! But *prediction is slow*
- O(dN) for N examples with d attributes
- *increases* with the number of examples!

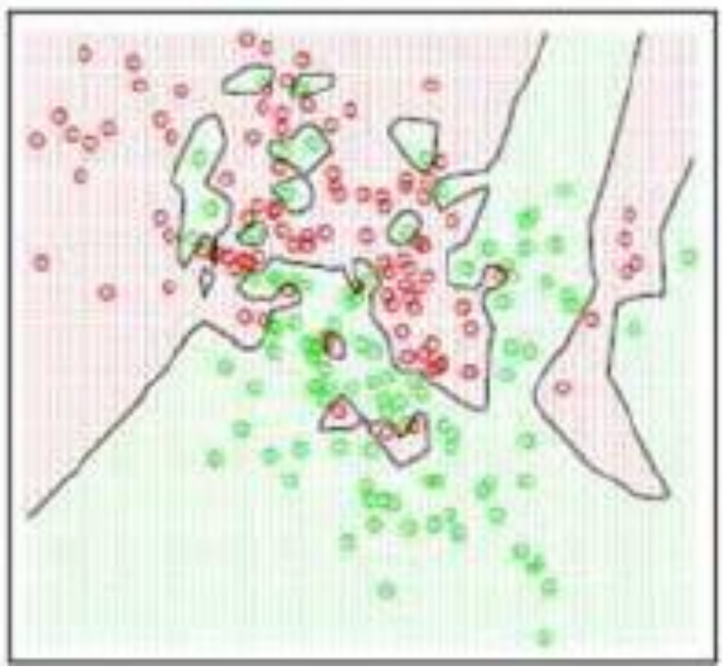
- ***Expressivity***
  - *What kind of functions can we learn?*

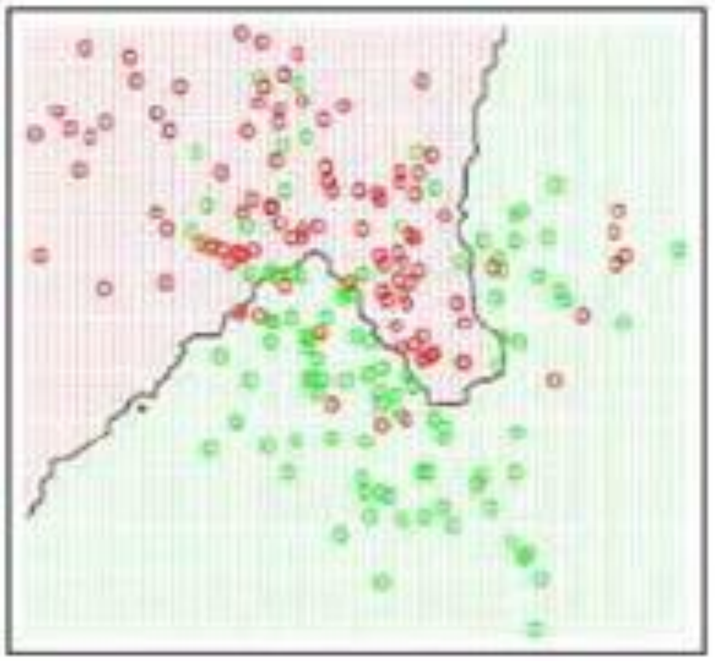
Analyzing K Nearest Neighbors

- We discussed the importance of finding a good model space
  - Expressive (we can represent the right model)
  - Constrained (we can search effectively, using the data we have)
- Let  erize the model space, by loo **sion boundary**
- **Ho if K=1?**

Analyzing K Nearest Neighbors

- Which model has a higher K value?
- Which model is more sensitive to noise?
- Which model is better?





Questions

- We know higher K values result in a smoother decision boundary.
  - Less ”jagged” decision regions
  - Total number of regions will be smaller
  - “Simpler, less expressive”

What will happen if we keep increasing K, up to the point that K=n ?

n = is the number of examples we have

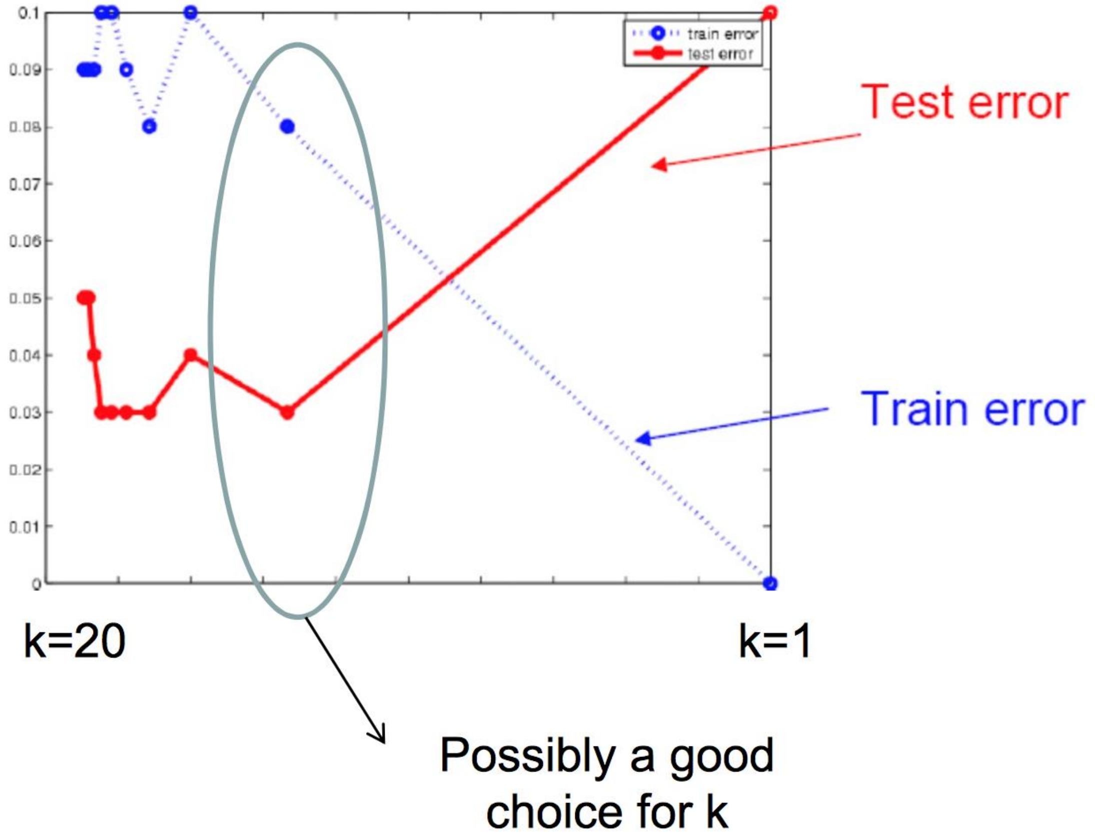
How should we determine the value of K?

- Higher K values result in less complex functions (less expressive)
- Lower K values are more sensitive to noises in data
- **How can we find the right balance between the two?**
- Option 1: Find the K that minimizes the training error.
  - Training error: after learning the classifier, what is the number of errors we get on the training data.
  - What will be this value for k=1, k=n, k=n/2?

Is this a good idea?

- Option 2: Find K that minimizes the **validation error**.
  - Validation error: set aside some of the data (validation set). what is the number of errors we get on the validation data, after training the classifier.

How should we determine the value of K?



Test error

Train error

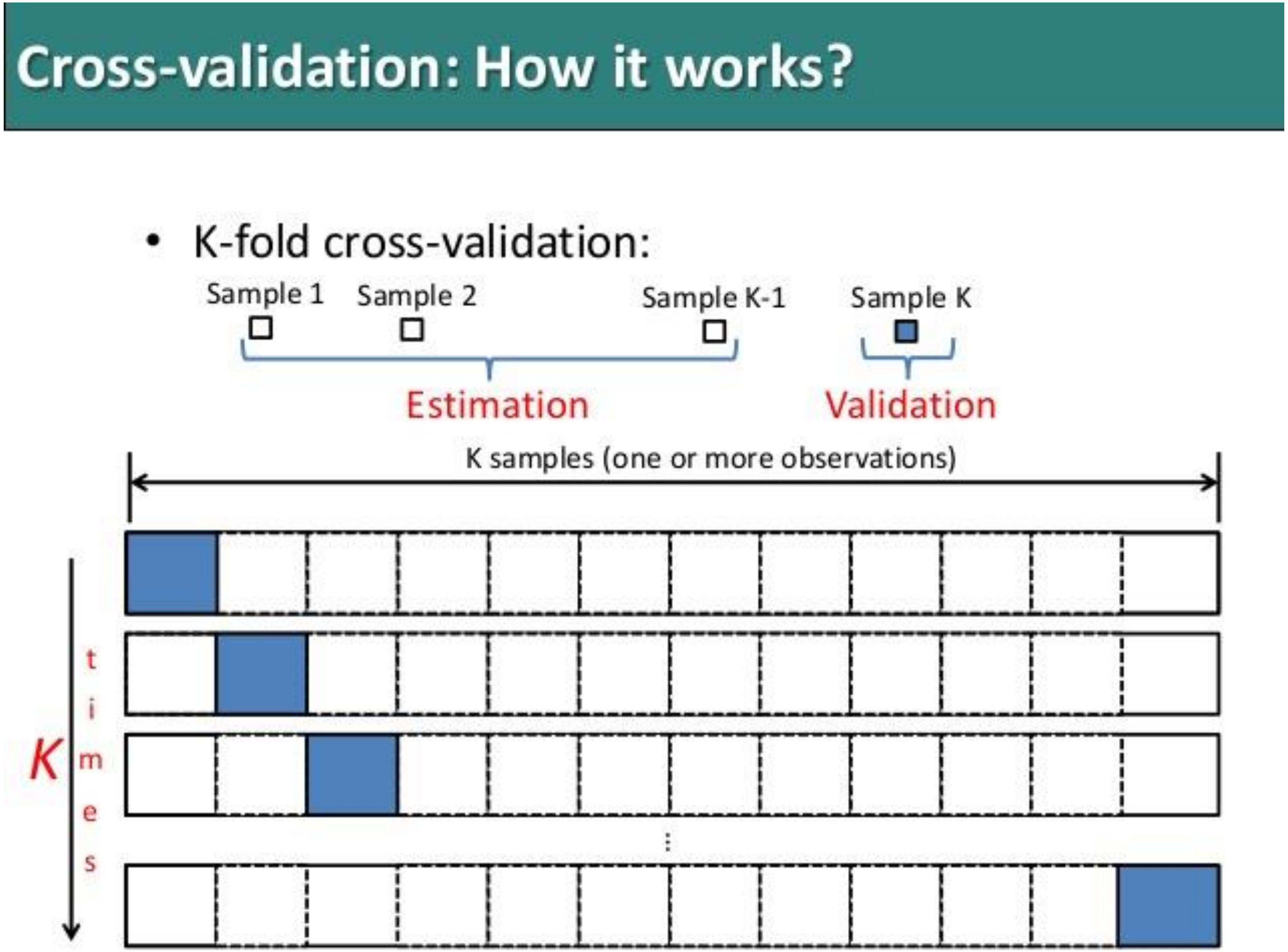
Possibly a good choice for k

In general – using the training error to tune parameters will always result in a more complex hypothesis! **(why?)**

Training-Validation-Testing Datasets

- Training set: data for learning a model
- Test set: data used to assess and strength of learned model (evaluate)
- Validation set:
  - Used to learn hyper parameters, such as the value k in kNN, choosing among different models
  - Hold-out method: leave about 30% of data from training set for validation

Cross-Validation



Cross-validation: How it works?

- K-fold cross-validation:

Usage of Cross-Validation when Evaluating an Algorithm

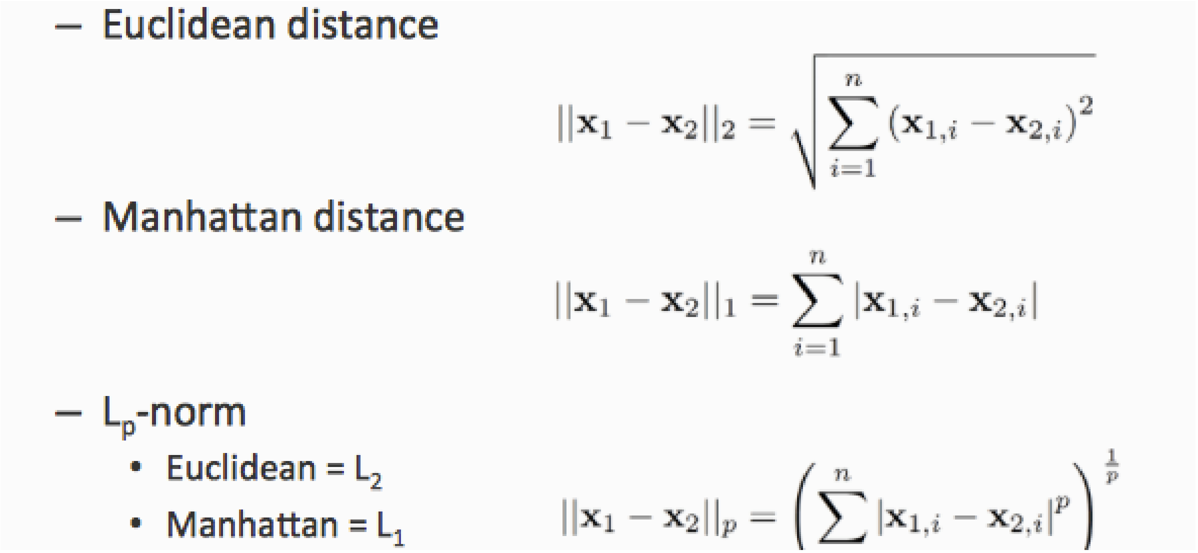
- Given a dataset that needs to be divided to be used for training and testing.
- No Cross-Validation: Divide the data into Training and Testing, use the Training set to train a model, and then use the Testing set to get accuracy numbers
- With Cross-Validation: Dividing data into c equal-size subset.
  - Each time use c-1 subsets to train and the remaining subset to evaluate to get accuracy numbers
  - Repeat c times, and take average

Usage of Cross-Validation to Select Hyper-Parameter

- When selecting hyper-parameter of an algorithm.
  - In Machine Learning, a **hyperparameter** is a parameter whose value is set before the learning process begins. By contrast, the values of other parameters are derived via training.
- No Cross-Validation: Divide the Training data into Training and Validation. For each **hyperparameter** value, use the Training set to train a model, and then evaluate using the Validation set, choose the hyper parameter value that gives the best result.
- With Cross-Validation: dividing data into c equal-size subset: For each **hyperparameter** value,
  - For each j in [1,c], all except set j to train, and subset j to evaluate, average the results from c runs.
  - Choose the hyperparameter that has the best average result

KNN Practical Consideration

- Finding the right representation is key
  - KNN is very sensitive to irrelevant attributes
- Choosing the right distance metric is important
  - Many options!
  - Popular choices:



- Euclidean distance

$$||\mathbf{x}_1 - \mathbf{x}_2||_2 = \sqrt{\sum_{i=1}^{n} (\mathbf{x}_{1,i} - \mathbf{x}_{2,i})^2}$$

- Manhattan distance

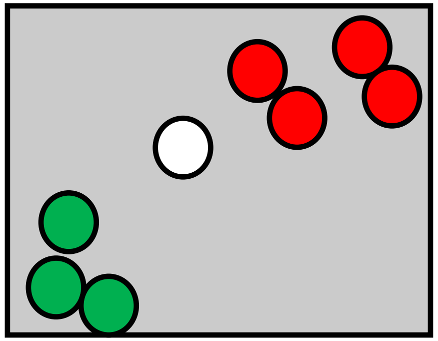
$$||\mathbf{x}_1 - \mathbf{x}_2||_1 = \sum_{i=1}^{n} |\mathbf{x}_{1,i} - \mathbf{x}_{2,i}|$$

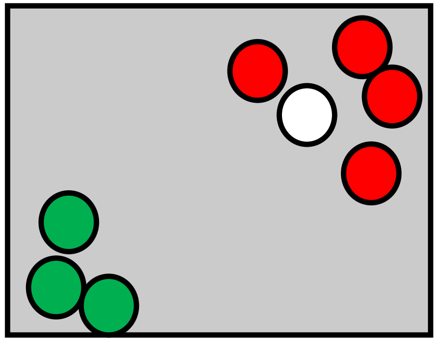
- L_p-norm
  - Euclidean = L_2
  - Manhattan = L_1

$$||\mathbf{x}_1 - \mathbf{x}_2||_p = \left(\sum^{n} |\mathbf{x}_{1,i} - \mathbf{x}_{2,i}|^p\right)^{\frac{1}{p}}$$

Beyond KNN

- KNN is not a statistical classifier.
- It memorizes the training data, and makes a majority vote over the K closest points.
- For example, these two cases make the same decision for the white data point:





- What is the difference between the two scenarios?
- How can we reason about it?