Machine Learning ZG565

Dr. Sugata Ghosal
Sugata.ghosal@pilani.bits-pilani.ac.in

BITS Pilani
Pilani Campus

BITS Pilani
Pilani Campus

Session 2
Date – 27th May 2023
Time – 2 PM to 4 PM

These slides are prepared by the instructor, with grateful acknowledgement of many others who made their course materials freely available online.

Session Content

- Data - Types, Attributes
- Data Quality
- Data Preprocessing
- Performance Metrics
- Challenges of ML
- Model Evaluation, Selection

- Homework
  - Review Lab Capsule 1 from the “Machine Learning” Virtual Labs (accessible via elearn portal)

ML in a Nutshell

- Tens of thousands of machine learning algorithms
  - Hundreds new every year

- Every ML algorithm has three components
  - Data Representation
  - Parameter Optimization
  - Model Evaluation, Selection

Slide credit: Pedro Domingos

ML in Practice

Loop

- Understand domain, prior knowledge, and goals
- Data integration, selection, cleaning, pre-processing, etc.
- Learn optimal parameter of the models
- Interpret results
- Consolidate and deploy discovered knowledge

Based on a slide by Pedro Domingos

Definition of Data

- Collection of ***data objects*** and their ***attributes***
- An ***attribute*** is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - *aka* variable, field, characteristic, dimension, or feature
- A collection of attributes describe an ***object***
  - Object is also known as record, point, case, sample, entity, or instance

Attributes

Objects

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Attribute Values

- ***Attribute values*** are numbers or symbols assigned to an attribute for a particular object

- Distinction between attributes and attribute values
  - Same attribute can be mapped to different attribute values
    - Example: height can be measured in feet or meters

  - Different attributes can be mapped to the same set of values
    - Example: Attribute values for ID and age are integers
  - But properties of attribute can be different than the properties of the values used to represent the attribute

Types of Attributes

- There are different types of attributes
  - Nominal
    - Examples: ID numbers, zip codes
  - Ordinal
    - Examples: rankings (e.g., taste of potato chips on a scale from 1-10), grades, height {tall, medium, short}
  - Interval
    - Examples: calendar dates, temperatures in Celsius or Fahrenheit.
  - Ratio
    - Examples: temperature in Kelvin, length, counts, elapsed time (e.g., time to run a race)

Properties of Attribute Values

- The type of an attribute depends on which of the following properties/operations it possesses:
  - Distinctness: $= \neq$
  - Order: $< >$
  - Differences are meaningful : $+ -$
  - Ratios are meaningful $* /$

  - Nominal attribute: distinctness
  - Ordinal attribute: distinctness & order
  - Interval attribute: distinctness, order & meaningful differences
  - Ratio attribute: all 4 properties/operations

Difference Between Ratio and Interval

- Is it physically meaningful to say that a temperature of 10 ° is twice that of 5° on
  - the Celsius scale?
  - the Fahrenheit scale?
  - the Kelvin scale?
- Consider measuring the height above average
  - If Bill's height is three inches above average and Bob's height is six inches above average, then would we say that Bob is twice as tall as Bill?
  - Is this situation analogous to that of temperature?

	Attribute Type	Description	Examples	Operations
Categorical Qualitative	Nominal	Nominal attribute values only distinguish. (=, ≠)	zip codes, employee ID numbers, eye color, sex: {*male, female*}	mode, entropy, contingency correlation, $\chi 2$ test
	Ordinal	Ordinal attribute values also order objects. (<, >)	hardness of minerals, {*good, better, best*}, grades, street numbers	median, percentiles, rank correlation, run tests, sign tests
Numeric Quantitative	Interval	For interval attributes, differences between values are meaningful. (+, -)	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, *t* and *F* tests
	Ratio	For ratio variables, both differences and ratios are meaningful. (*, /)	temperature in Kelvin, monetary quantities, counts, age, mass, length, current	geometric mean, harmonic mean, percent variation

This categorization of attributes is due to S. S. Stevens

	Attribute Type	Transformation	Comments
Categorical Qualitative	Nominal	Any permutation of values	If all employee ID numbers were reassigned, would it make any difference?
	Ordinal	An order preserving change of values, i.e., *new_value = f(old_value)* where *f* is a monotonic function	An attribute encompassing the notion of good, better best can be represented equally well by the values {1, 2, 3} or by { 0.5, 1, 10}.
Numeric Quantitative	Interval	*new_value = a * old_value + b* where a and b are constants	Thus, the Fahrenheit and Celsius temperature scales differ in terms of where their zero value is and the size of a unit (degree).
	Ratio	*new_value = a * old_value*	Length can be measured in meters or feet.

This categorization of attributes is due to S. S. Stevens

Discrete and Continuous Attributes

- Discrete Attribute
  - Has only a finite or countably infinite set of values
  - Examples: zip codes, counts, or the set of words in a collection of documents
  - Often represented as integer variables.
  - Note: binary attributes are a special case of discrete attributes
- Continuous Attribute
  - Has real numbers as attribute values
  - Examples: temperature, height, or weight.
  - Practically, real values can only be measured and represented using a finite number of digits.
  - Continuous attributes are typically represented as floating-point variables.

Asymmetric Attributes

- Only presence (a non-zero attribute value) is regarded as important
  - Words present in documents
  - Items present in customer transactions

- If we met a friend in the grocery store would we ever say the following?

  "I see our purchases are very similar since we didn't buy most of the same things."

Key Messages for Attribute Types

- The types of operations you choose should be "meaningful" for the type of data you have
  - Distinctness, order, meaningful intervals, and meaningful ratios are only four (among many possible) properties of data
  - The data type you see – often numbers or strings – may not capture all the properties or may suggest properties that are not present
  - Analysis may depend on these other properties of the data
    - Many statistical analyses depend only on the distribution
  - In the end, what is meaningful can be specific to domain

Important Characteristics of Data

- Dimensionality (number of attributes)
  - High dimensional data brings a number of challenges
- Sparsity
  - Only presence counts
- Resolution
  - Patterns depend on the scale
- Size
  - Type of analysis may depend on size of data

Types of data sets

- Record
  - Data Matrix
  - Document Data
  - Transaction Data
- Graph
  - World Wide Web
  - Molecular Structures
- Ordered
  - Spatial Data
  - Temporal Data
  - Sequential Data
  - Genetic Sequence Data

Record Data

Data that consists of a collection of records, each of which consists of a fixed set of attributes

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Data Matrix

- If data objects have the same fixed set of numeric attributes, then the data objects can be thought of as points in a multi-dimensional space, where each dimension represents a distinct attribute
- Such a data set can be represented by an *m* by *n* matrix, where there are *m* rows, one for each object, and *n* columns, one for each attribute

Projection of x Load	Projection of y load	Distance	Load	Thickness
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

Document Data

- Each document becomes a 'term' vector
  - Each term is a component (attribute) of the vector
  - The value of each component is the number of times the corresponding term occurs in the document.

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

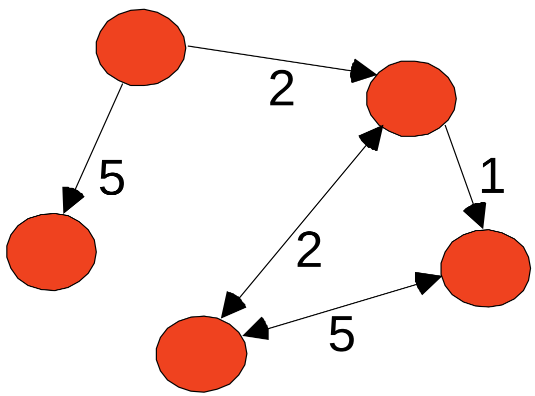
Transaction Data

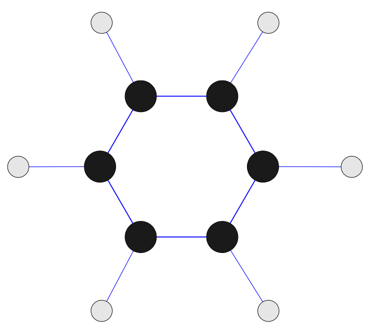
- A special type of data, where
  - Each transaction involves a set of items.
  - For example, consider a grocery store. The set of products purchased by a customer during one shopping trip constitute a transaction, while the individual products that were purchased are the items.
  - Can represent transaction data as record data

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

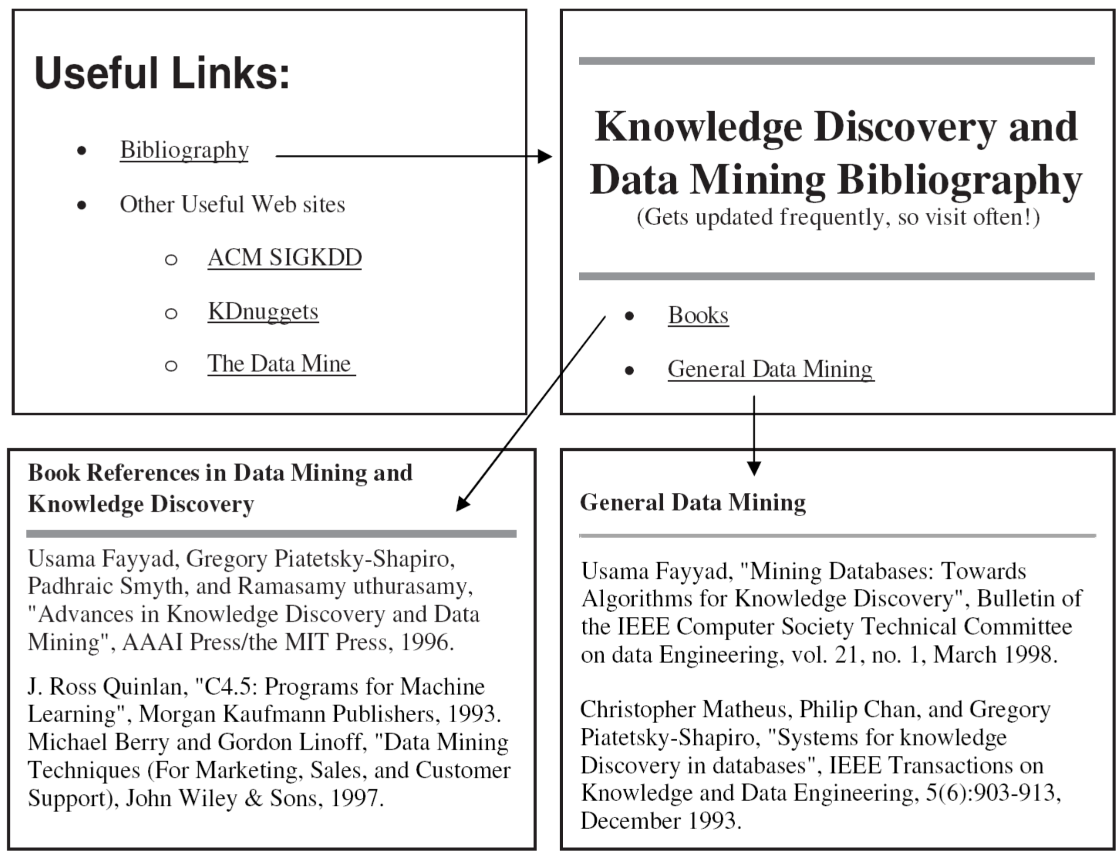
Graph Data

Examples: Generic graph, a molecule, and webpages





Benzene Molecule: C6H6


Useful Links:

- Bibliography
- Other Useful Web sites
  - ACM SIGKDD
  - KDnuggets
  - The Data Mine

Knowledge Discovery and Data Mining Bibliography

(Gets updated frequently, so visit often!)

- Books
- General Data Mining

Book References in Data Mining and Knowledge Discovery

Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy uthurasamy, "Advances in Knowledge Discovery and Data Mining", AAAI Press/the MIT Press, 1996.

J. Ross Quinlan, "C4.5: Programs for Machine Learning", Morgan Kaufmann Publishers, 1993. Michael Berry and Gordon Linoff, "Data Mining Techniques (For Marketing, Sales, and Customer Support), John Wiley & Sons, 1997.

General Data Mining

Usama Fayyad, "Mining Databases: Towards Algorithms for Knowledge Discovery", Bulletin of the IEEE Computer Society Technical Committee on data Engineering, vol. 21, no. 1, March 1998.

Christopher Matheus, Philip Chan, and Gregory Piatetsky-Shapiro, "Systems for knowledge Discovery in databases", IEEE Transactions on Knowledge and Data Engineering, 5(6):903-913, December 1993.

Ordered Data

Sequences of transactions

Items/Events

(A B) (D) (C E)
(B D) (C) (E)
(C D) (B) (A E)

An element of the sequence

Ordered Data

Genomic sequence data

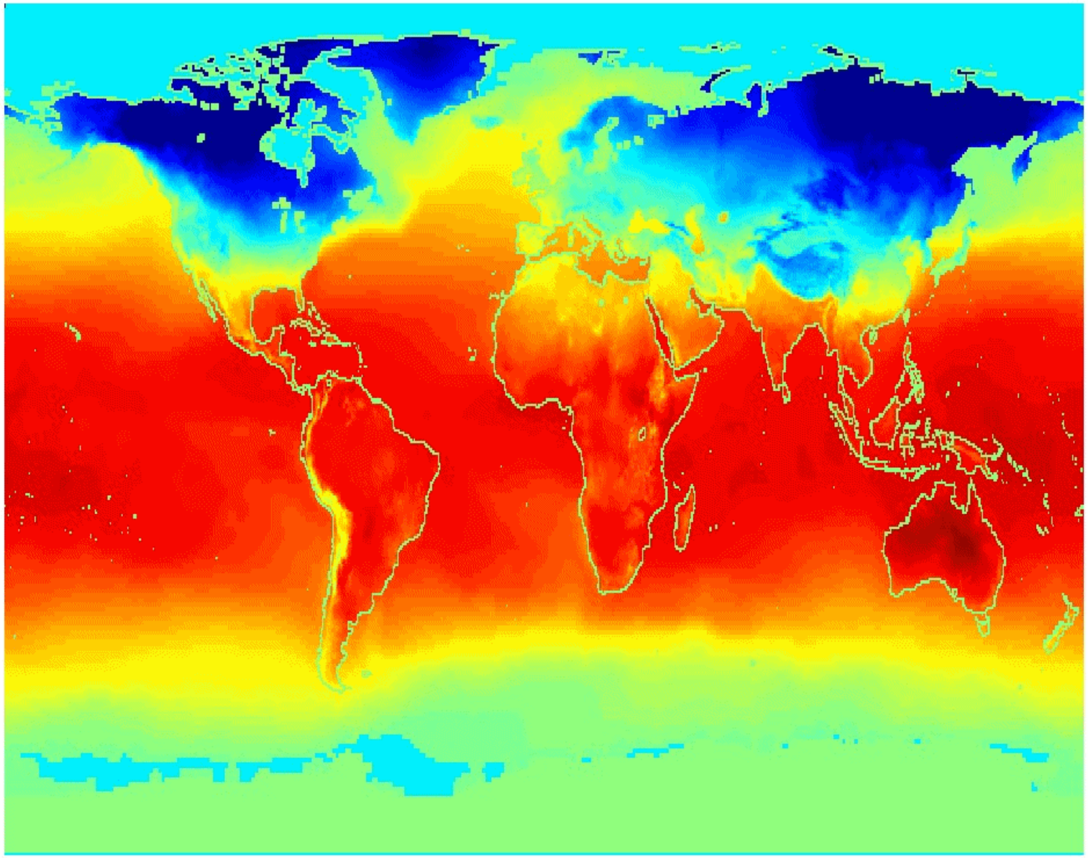
```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

Ordered Data

Spatiotemporal Data

Average Monthly Temperature of land and ocean

Jan



Data Quality

- Poor data quality negatively affects many data processing efforts

- ML example: a classification model for detecting people who are loan risks is built using poor data
  - Some credit-worthy candidates are denied loans
  - More loans are given to individuals that default

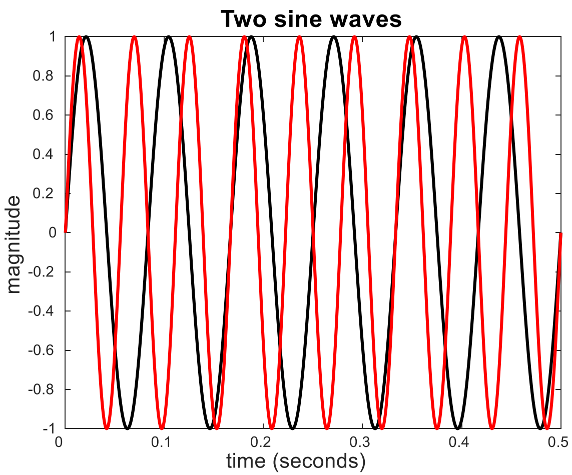
Data Quality …

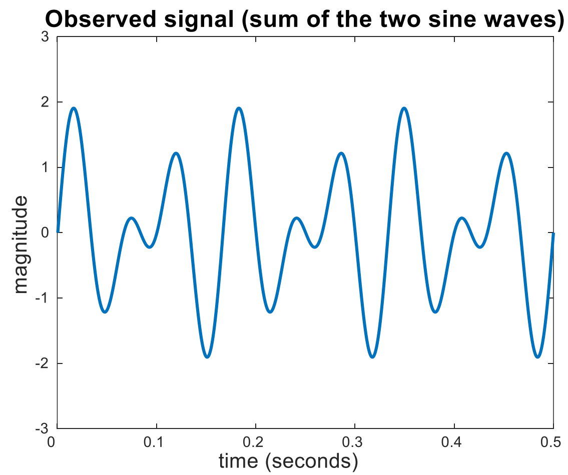
- What kinds of data quality problems?
- How can we detect problems with the data?
- What can we do about these problems?

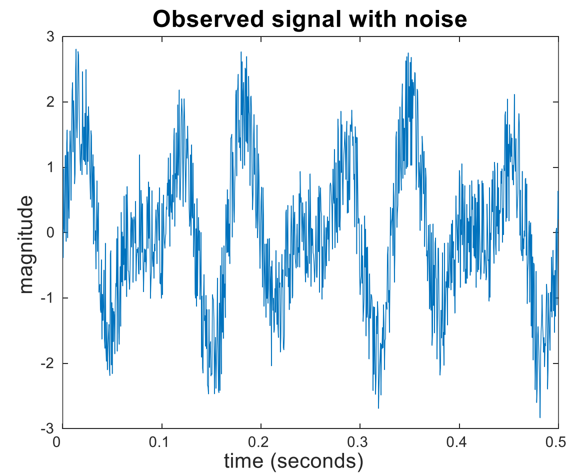
- Examples of data quality problems:
  - Noise and outliers
  - Wrong data
  - Fake data
  - Missing values
  - Duplicate data

Noise

- For objects, noise is an extraneous object
- For attributes, noise refers to modification of original values
  - Examples: distortion of a person’s voice when talking on a poor phone and “snow” on television screen
  - The figures below show two sine waves of the same magnitude and different frequencies, the waves combined, and the two sine waves with random noise
    - The magnitude and shape of the original signal is distorted

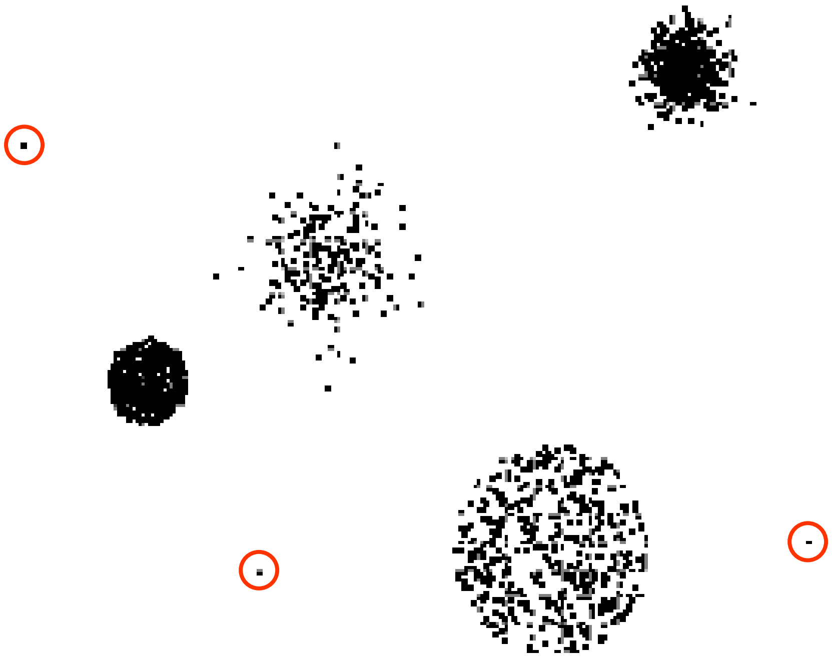






Outliers

- ***Outliers*** are data objects with characteristics that are considerably different than most of the other data objects in the data set
  - **Case 1:** Outliers are noise that interferes with data analysis
  - **Case 2:** Outliers are the goal of our analysis
    - Credit card fraud
    - Intrusion detection
- Causes?



Missing Values

- Reasons for missing values
  - Information is not collected
    (e.g., people decline to give their age and weight)
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)

- Handling missing values
  - Eliminate data objects or variables
  - Estimate missing values
    - Example: time series of temperature
    - Example: census results
  - Ignore the missing value during analysis

Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeneous sources
- Examples:
  - Same person with multiple email addresses
- Data cleaning
  - Process of dealing with duplicate data issues

Data Preprocessing

- Aggregation
- Sampling
- Discretization and Binarization
- Attribute Transformation
- Dimensionality Reduction
- Feature subset selection
- Feature creation

Aggregation

- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
  - Data reduction - reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc.
    - Days aggregated into weeks, months, or years
  - More “stable” data - aggregated data tends to have less variability

Table 2.4. Data set containing information about customer purchases.

Transaction ID	Item	Store Location	Date	Price	...
⋮	⋮	⋮	⋮	⋮	
101123	Watch	Chicago	09/06/04	\$25.99	...
101123	Battery	Chicago	09/06/04	\$5.99	...
101124	Shoes	Minneapolis	09/06/04	\$75.00	...
⋮	⋮	⋮	⋮	⋮	

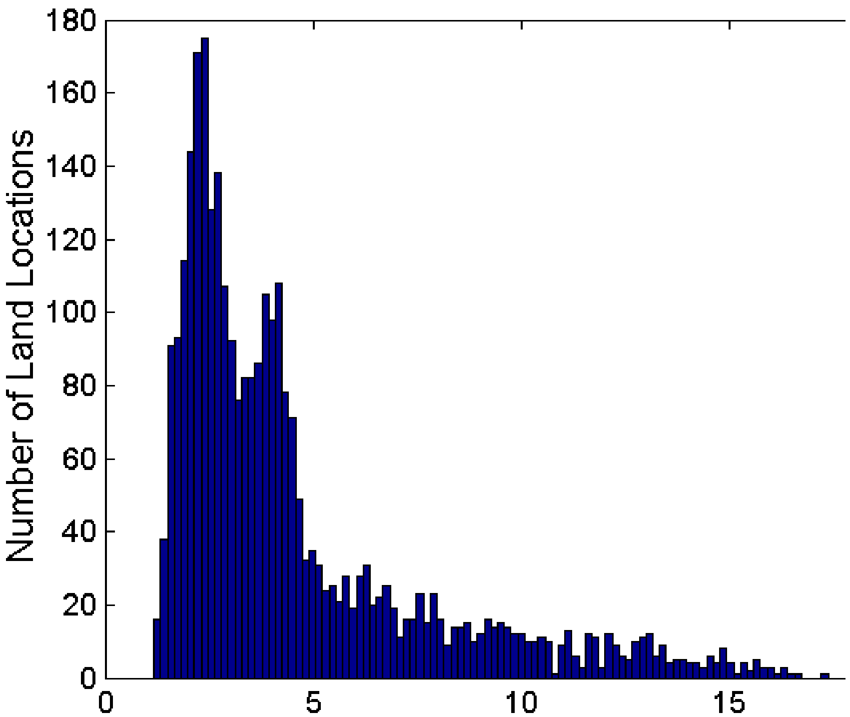
Example: Precipitation in Australia

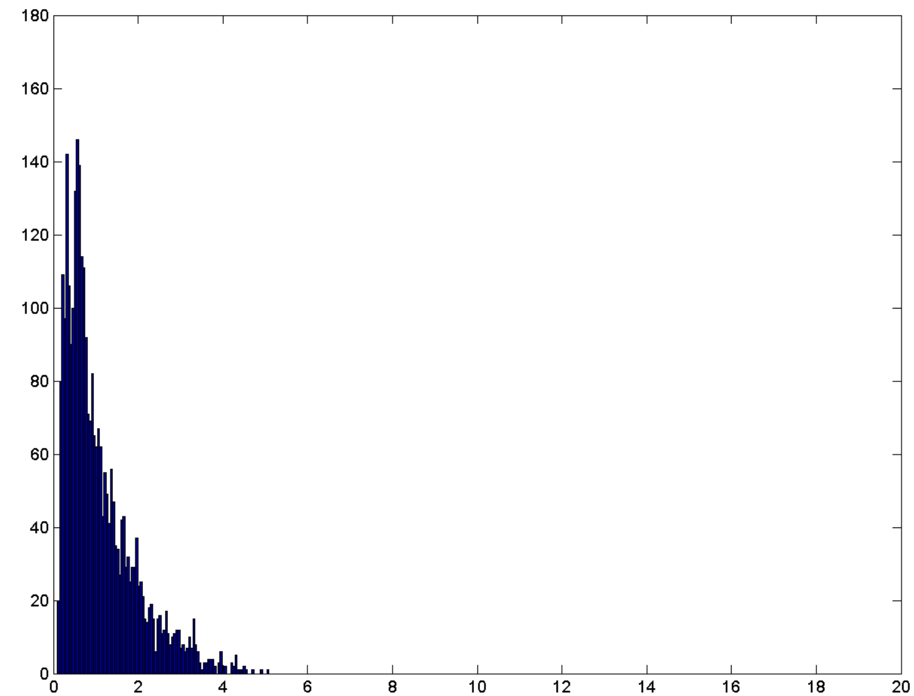
- This example is based on precipitation in Australia from the period 1982 to 1993.

  The next slide shows
  - A histogram for the standard deviation of average monthly precipitation for 3,030 0.5° by 0.5° grid cells in Australia, and
  - A histogram for the standard deviation of the average yearly precipitation for the same locations.
- The average yearly precipitation has less variability than the average monthly precipitation.
- All precipitation measurements (and their standard deviations) are in centimeters.

Example: Precipitation in Australia (contd)

Variation of Precipitation in Australia





Standard Deviation of Average Monthly Precipitation

Standard Deviation of Average Yearly Precipitation

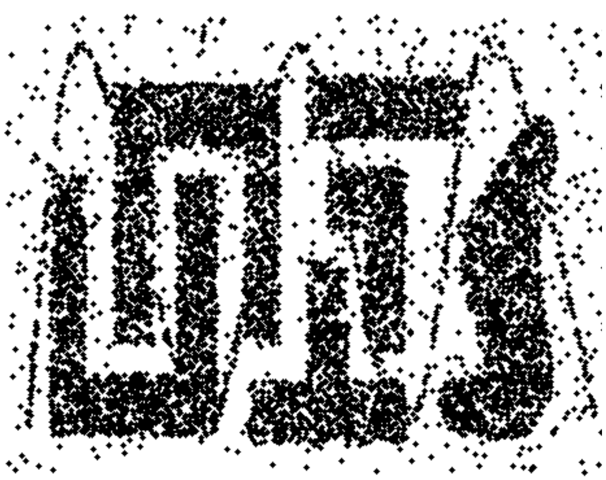
Sampling

- Sampling is the main technique employed for data reduction.
  - It is often used for both the preliminary investigation of the data and the final data analysis.
- Statisticians often sample because obtaining the entire set of data of interest is too expensive or time consuming.
- Sampling is typically used in data mining because processing the entire set of data of interest is too expensive or time consuming.

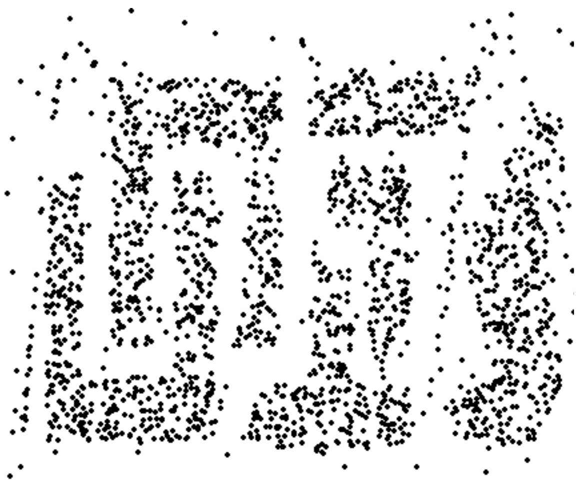
Sampling …

- The key principle for effective sampling is the following:
  - Using a sample will work almost as well as using the entire data set, if the sample is representative
  - A sample is representative if it has approximately the same properties (of interest) as the original set of data

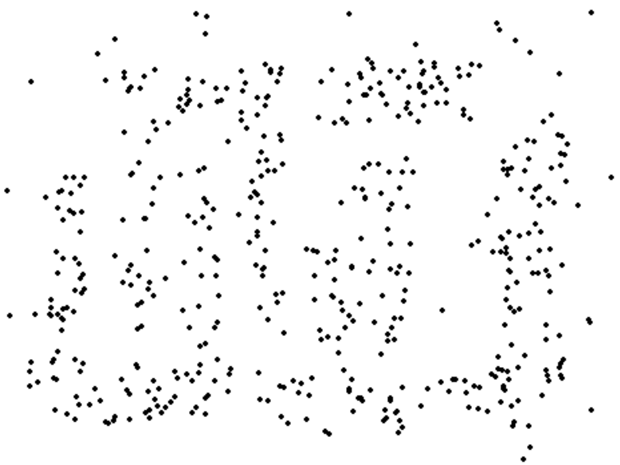
Sample Size



8000 points



2000 Points

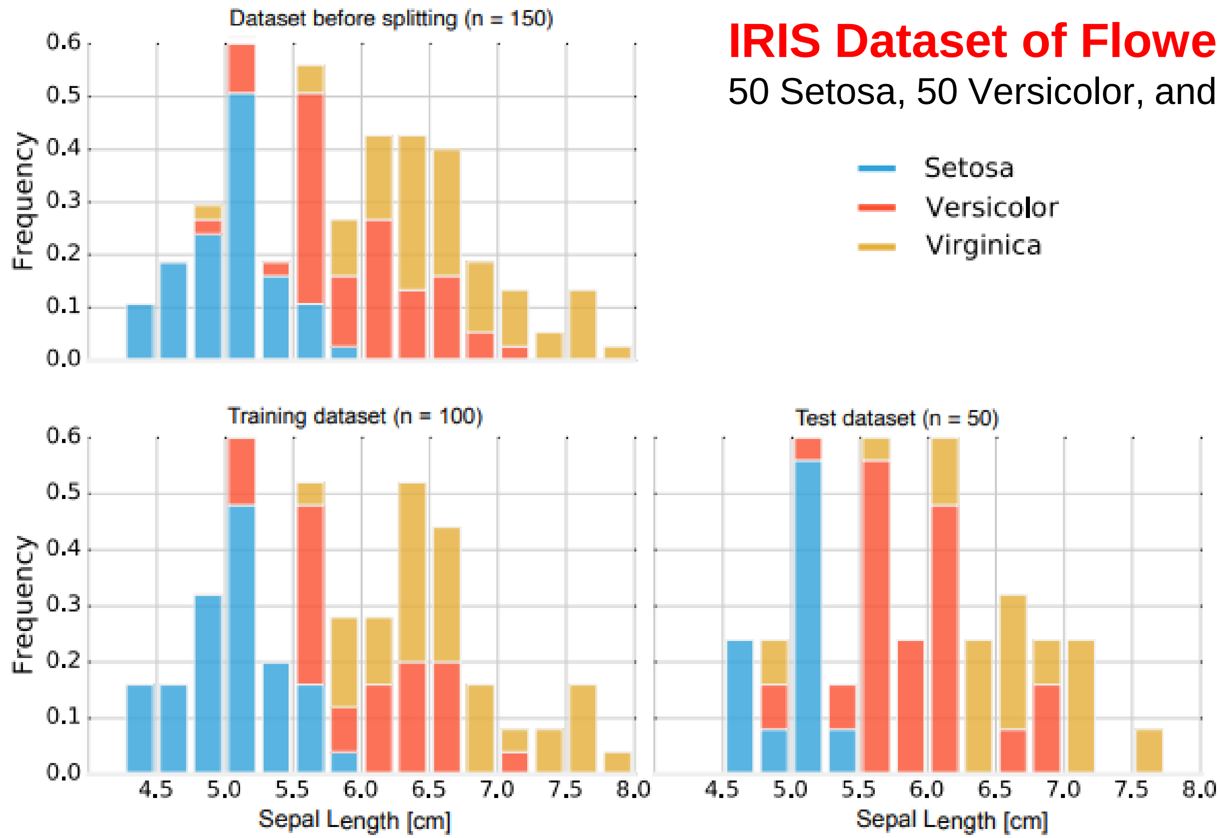


500 Points

Issues with Subsampling (Independence Violation)

IRIS Dataset of Flowers

50 Setosa, 50 Versicolor, and 50 Virginica



- **Random subsampling can assign 2/3 (100) to training set and 1/3 (50) to the test set**
- **Training set → 38 x Setosa, 28 x Versicolor, 34 x Virginica**
- **Test set → 12 x Setosa, 22 x Versicolor, 16 x Virginica**

Types of Sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
  - Sampling without replacement
    - As each item is selected, it is removed from the population
  - Sampling with replacement
    - Objects are not removed from the population as they are selected for the sample.
    - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition

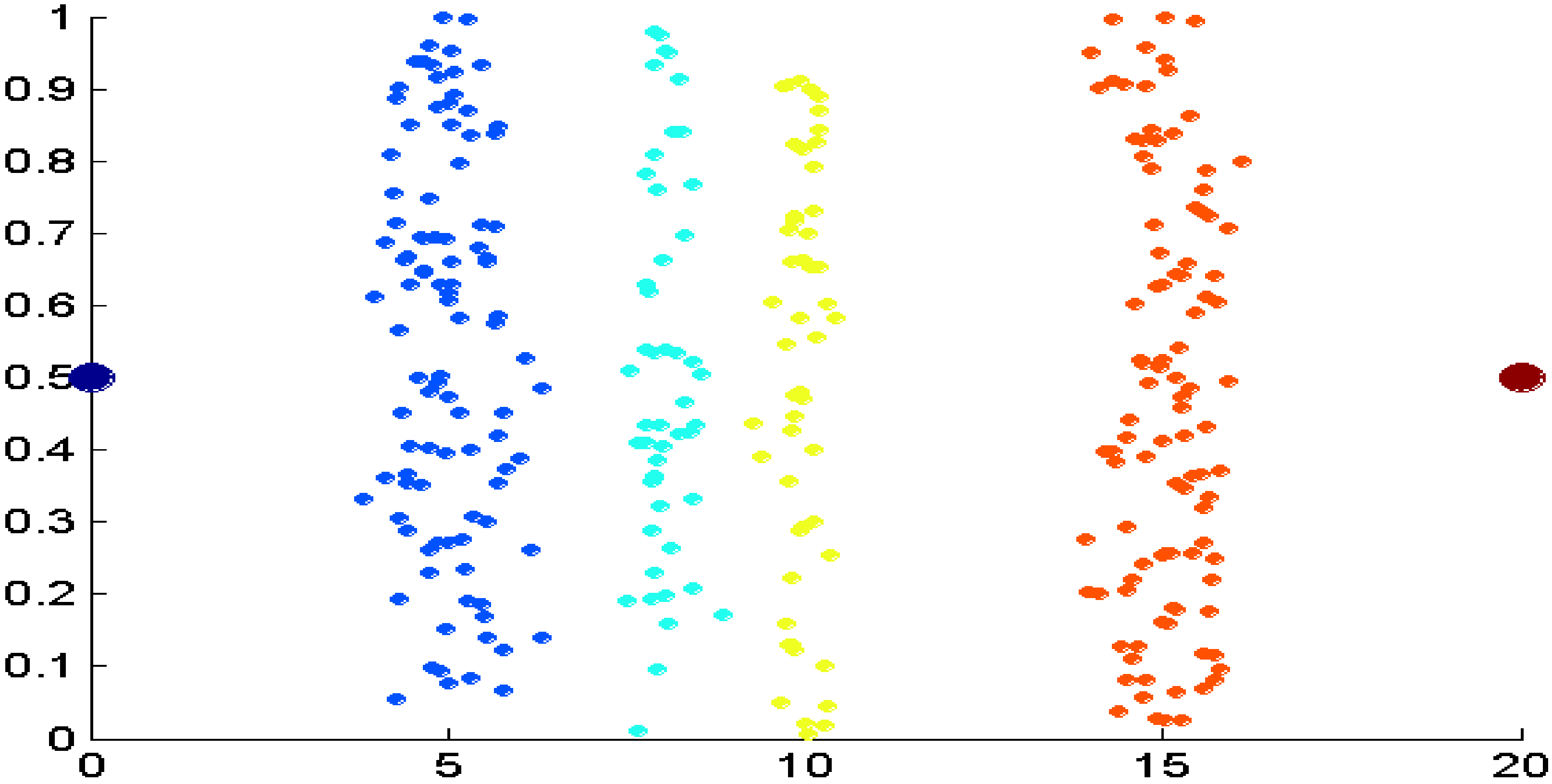
Building Classifiers with Imbalanced Training Set

- Modify the distribution of training data so that rare class is well-represented in training set
  - Undersample the majority class
  - Oversample the rare class

Discretization

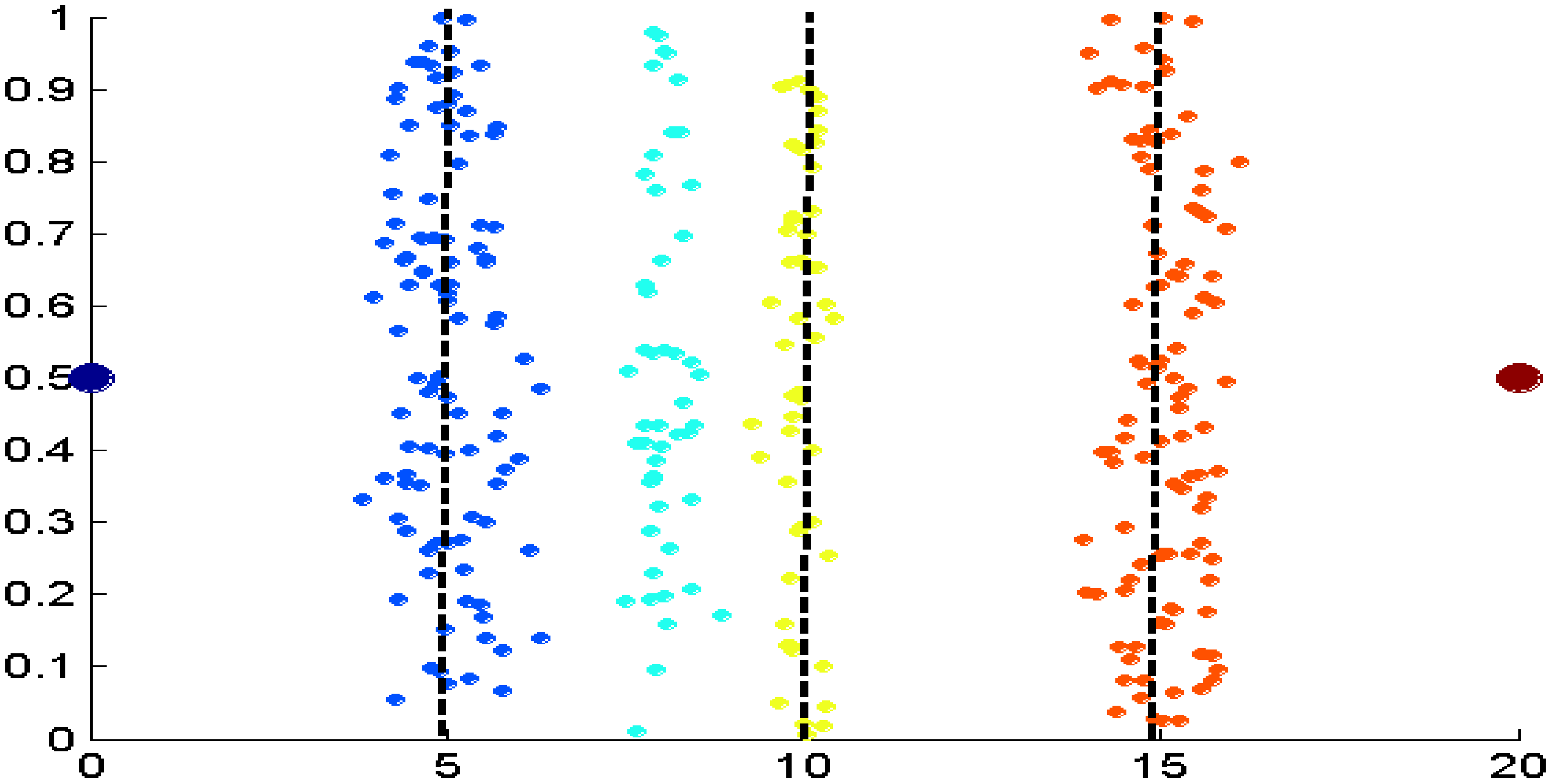
- Discretization is the process of converting a continuous attribute into an ordinal attribute
  - A potentially infinite number of values are mapped into a small number of categories
  - Discretization is used in both unsupervised and supervised settings

Unsupervised Discretization



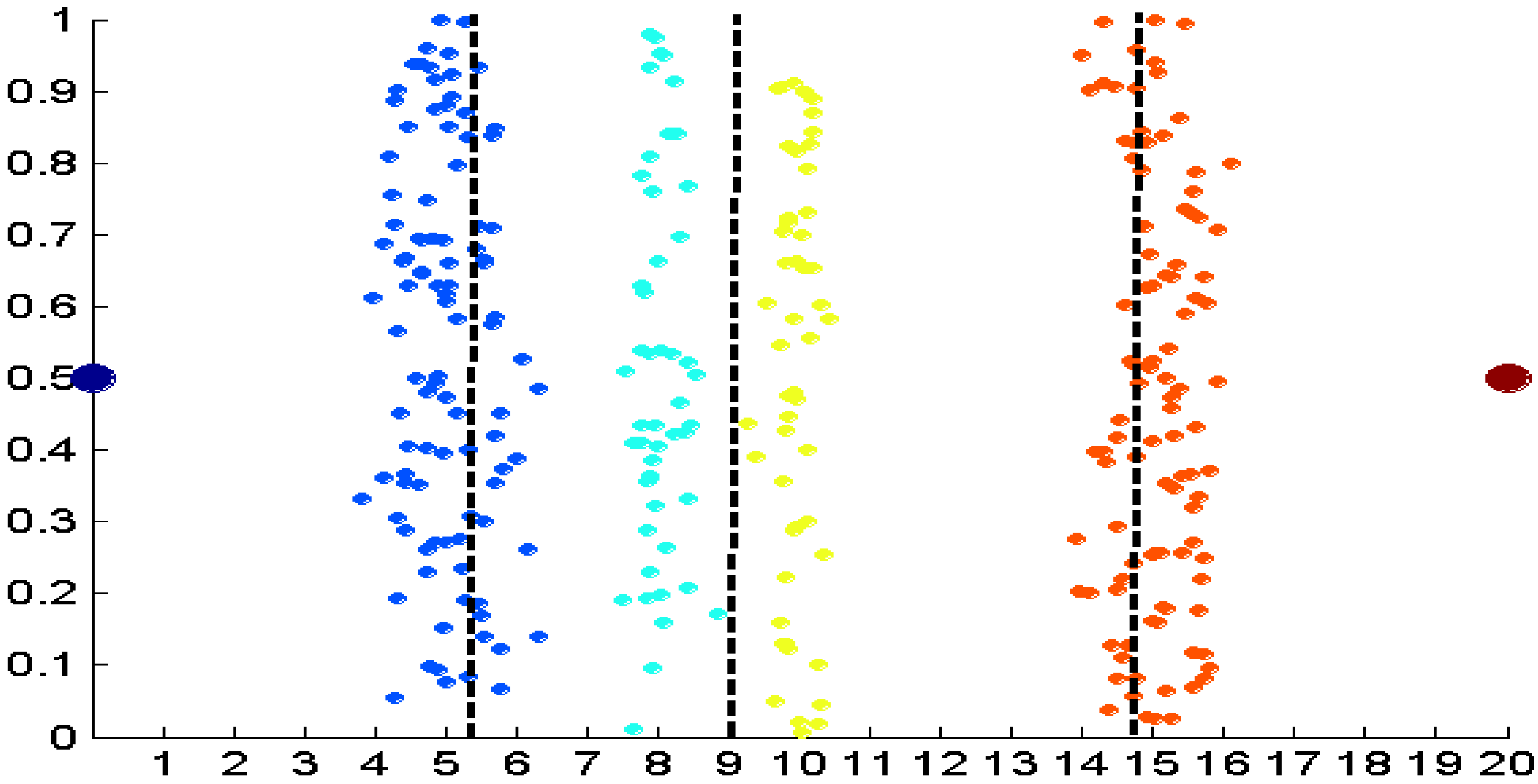
Data consists of four groups of points and two outliers. Data is one-dimensional, but a random y component is added to reduce overlap.

Unsupervised Discretization



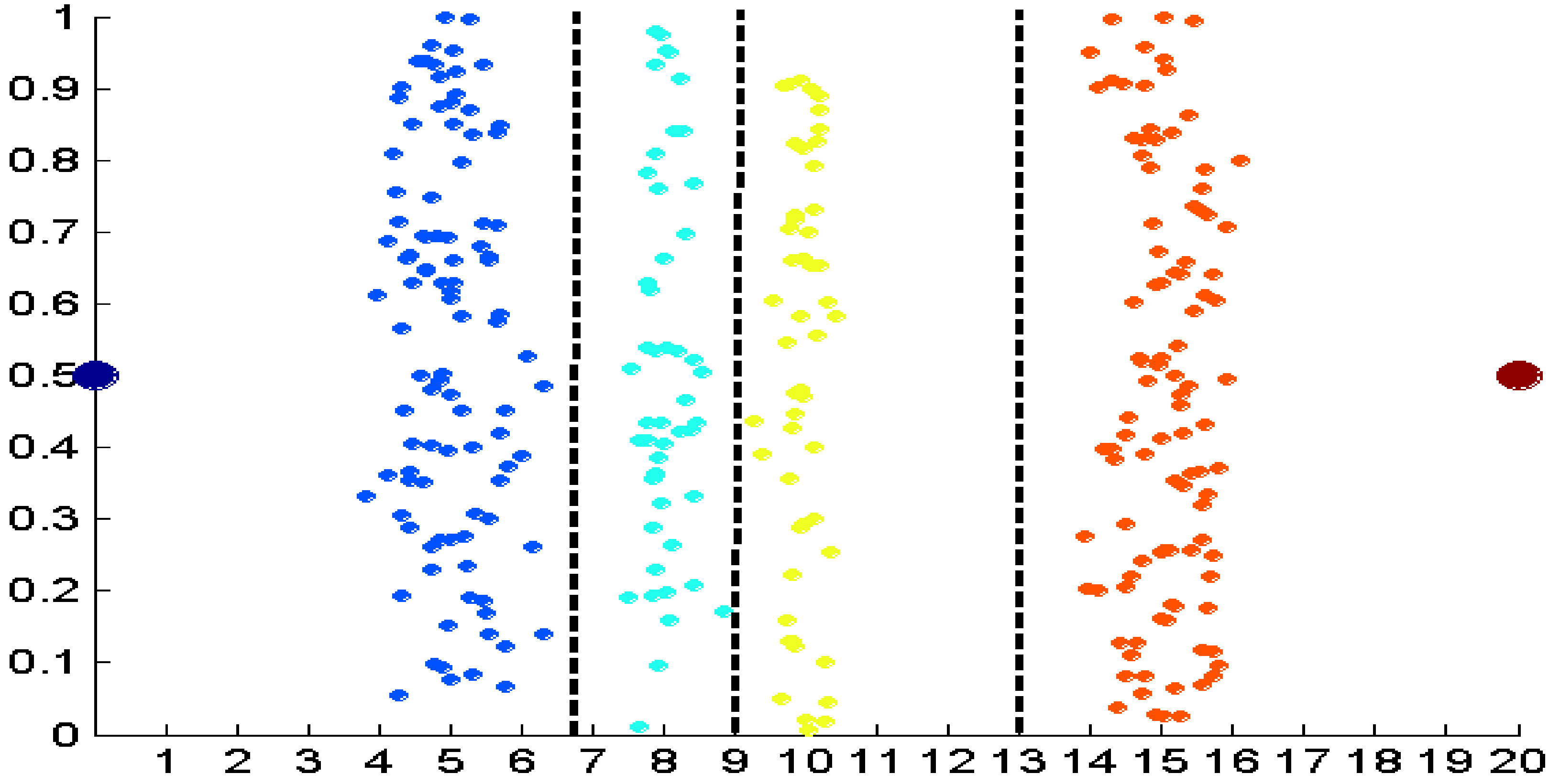
Equal interval width approach used to obtain 4 values.

Unsupervised Discretization



Equal frequency **approach used to obtain 4 values.**

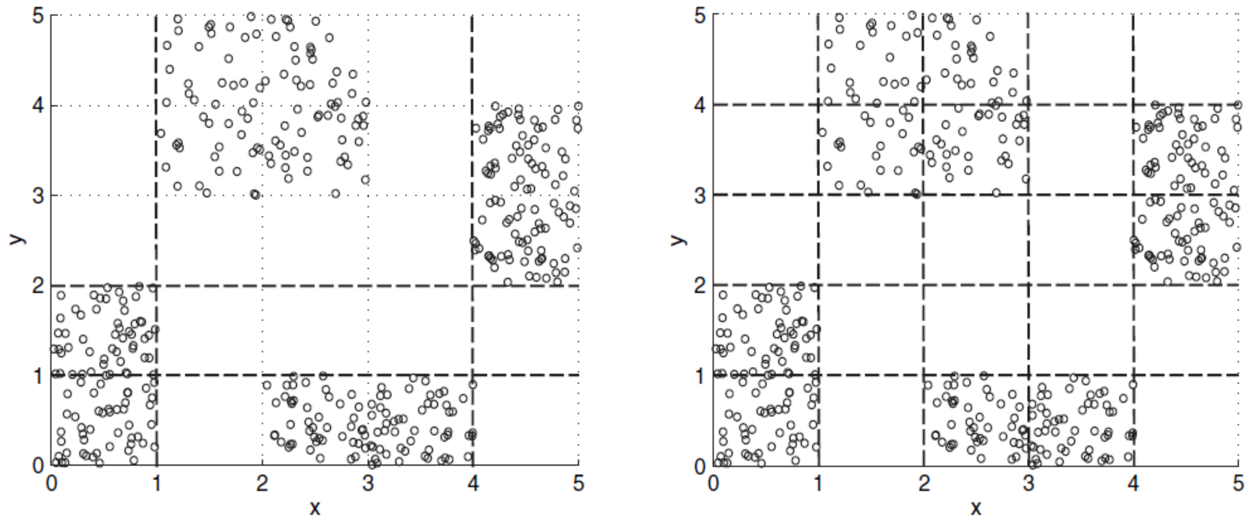
Unsupervised Discretization



K-means approach to obtain 4 values.

Discretization in Supervised Settings

- Many classification algorithms work best if both the independent and dependent variables have only a few values
- We give an illustration of the usefulness of discretization using the following example.



(a) Three intervals

(b) Five intervals

Figure 2.14. Discretizing x and y attributes for four groups (classes) of points.

Binarization

- Binarization maps a continuous or categorical attribute into one or more binary variables

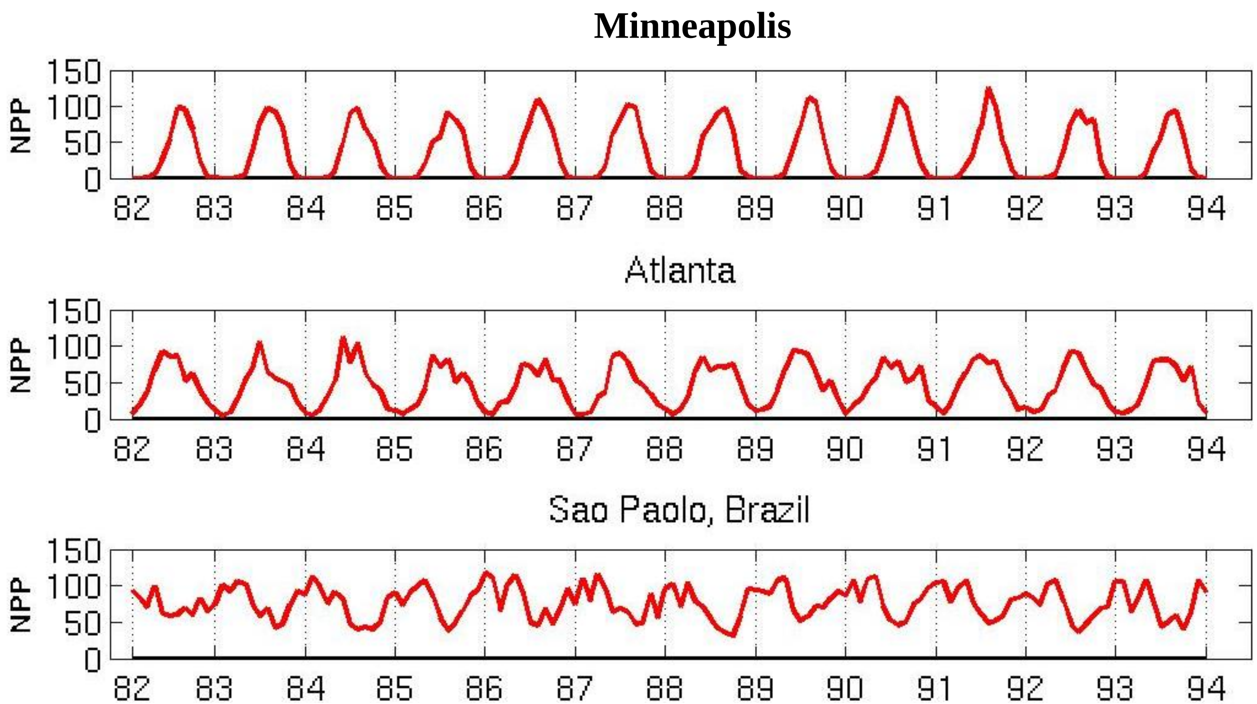
Table 2.6. Conversion of a categorical attribute to five asymmetric binary attributes.

Categorical Value	Integer Value	x_1	x_2	x_3	x_4	x_5
awful	0	1	0	0	0	0
poor	1	0	1	0	0	0
OK	2	0	0	1	0	0
good	3	0	0	0	1	0
great	4	0	0	0	0	1

Attribute Transformation

- An attribute transform is a function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
  - Simple functions: x^k, $\log(x)$, e^x, $|x|$
  - Normalization
    - Refers to various techniques to adjust to differences among attributes in terms of frequency of occurrence, mean, variance, range
    - Take out unwanted, common signal, e.g., seasonality
  - In statistics, standardization refers to subtracting off the means and dividing by the standard deviation

Example: Sample Time Series of Plant Growth

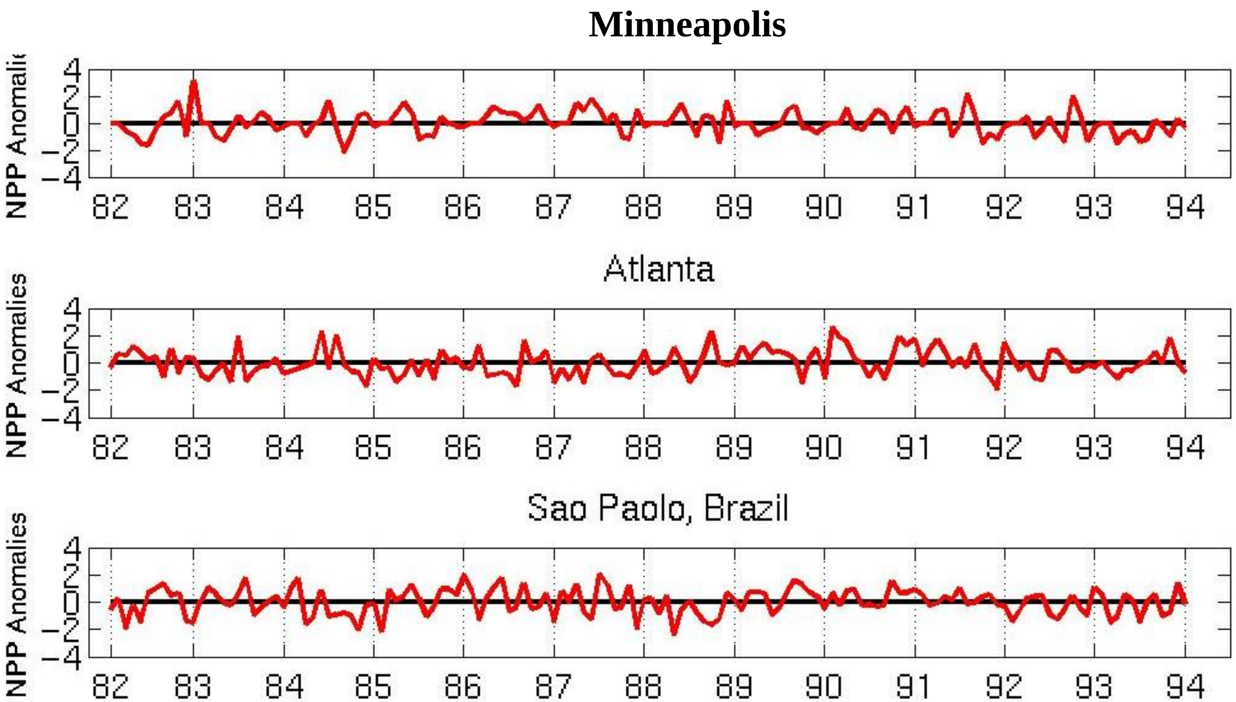


Net Primary Production (NPP) is a measure of plant growth used by ecosystem scientists.

Correlations between time series

	Minneapolis	Atlanta	Sao Paolo
Minneapolis	1.0000	0.7591	-0.7581
Atlanta	0.7591	1.0000	-0.5739
Sao Paolo	-0.7581	-0.5739	1.0000

Seasonality Accounts for Much Correlation



Normalized using monthly Z Score:

Subtract off monthly mean and divide by monthly standard deviation

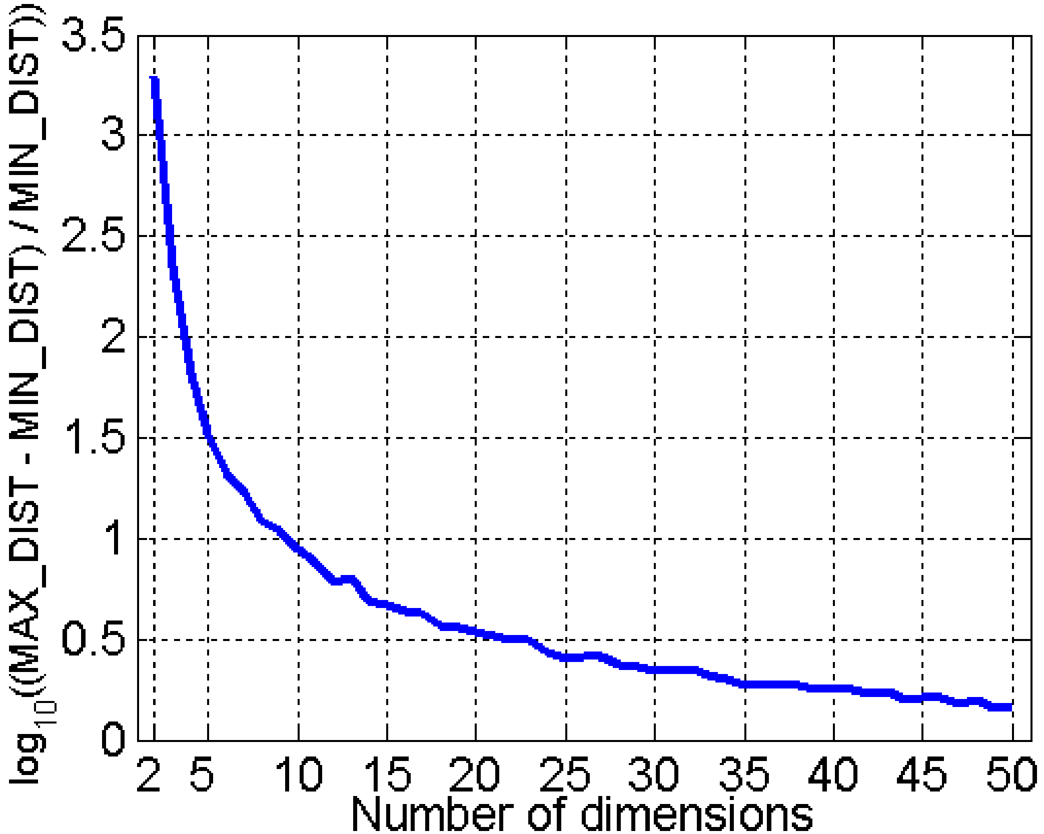
Correlations between time series

	Minneapolis	Atlanta	Sao Paolo
Minneapolis	1.0000	0.0492	0.0906
Atlanta	0.0492	1.0000	-0.0154
Sao Paolo	0.0906	-0.0154	1.0000

Curse of Dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies

- Definitions of density and distance between points, which are critical for clustering and outlier detection, become less meaningful



- **Randomly generate 500 points**
- **Compute difference between max and min distance between any pair of points**

Dimensionality Reduction

- Purpose:
  - Avoid curse of dimensionality
  - Reduce amount of time and memory required by data mining algorithms
  - Allow data to be more easily visualized
  - May help to eliminate irrelevant features or reduce noise

- Techniques
  - Principal Components Analysis (PCA)
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

Feature Subset Selection

- Another way to reduce dimensionality of data
- Redundant features
  - Duplicate much or all of the information contained in one or more other attributes
  - Example: purchase price of a product and the amount of sales tax paid
- Irrelevant features
  - Contain no information that is useful for the data mining task at hand
  - Example: students' ID is often irrelevant to the task of predicting students' GPA
- Many techniques developed, especially for classification

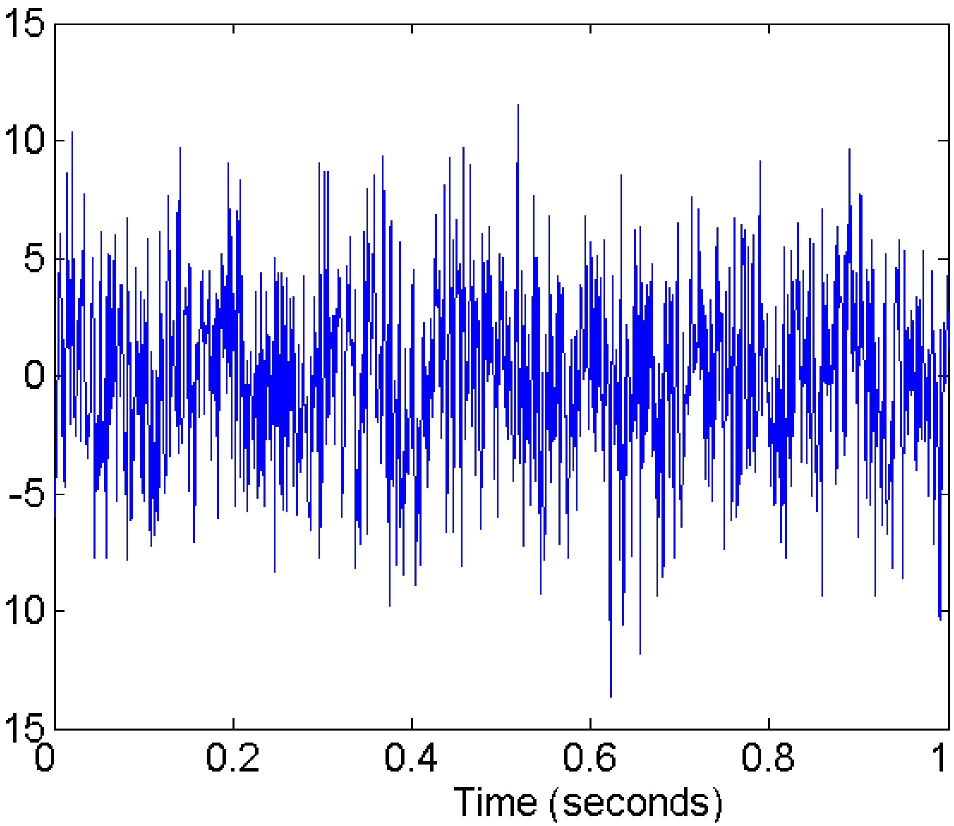
Feature Creation

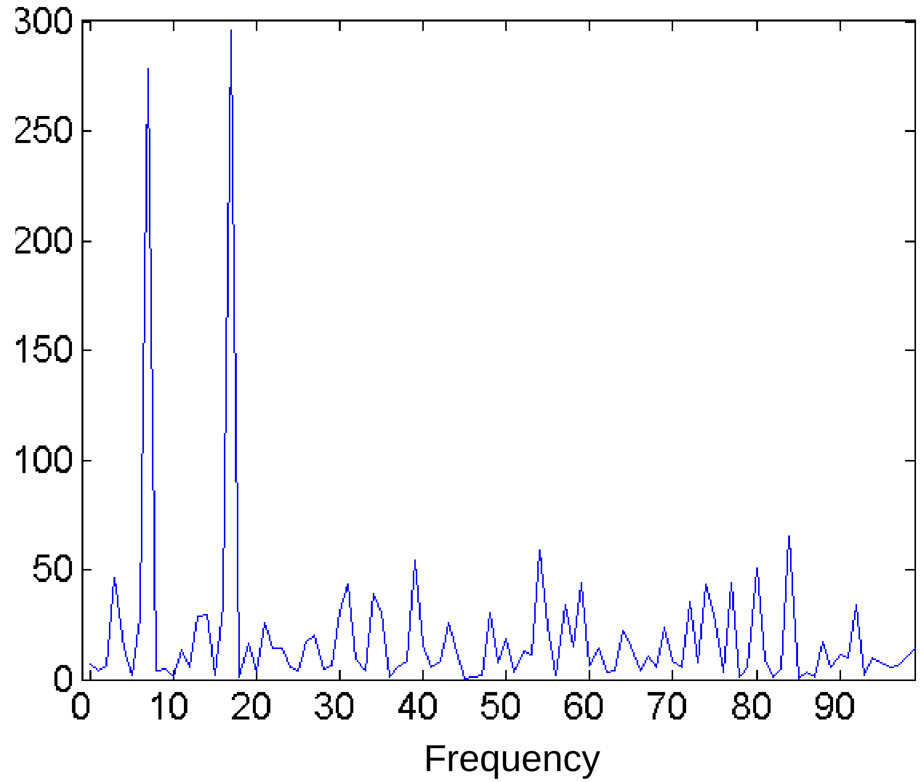
- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes

- Three general methodologies:
  - Feature extraction
    - Example: extracting edges from images
  - Feature construction
    - Example: dividing mass by volume to get density
  - Mapping data to new space
    - Example: Fourier and wavelet analysis

Mapping Data to a New Space

Fourier and wavelet transform





Two Sine Waves + Noise

Frequency

Evaluation Metrics: Confusion Matrix

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Evaluation Metrics: Accuracy

	PREDICTED CLASS	

ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Class Imbalance Problem

- Lots of classification problems where the classes are skewed (more records from one class than another)
  - Credit card fraud
  - Intrusion detection
  - Defective products in manufacturing assembly line
  - COVID-19 test results on a random sample

- **Key Challenge**:
  - Evaluation measures such as accuracy are not well-suited for imbalanced class

Problem with Accuracy

- Consider a 2-class problem
  - Number of Class NO examples = 990
  - Number of Class YES examples = 10
- If a model predicts everything to be class NO, accuracy is 990/1000 = 99 %
  - This is misleading because this trivial model does not detect any class YES example
  - Detecting the rare class is usually more interesting (e.g., frauds, intrusions, defects, etc)

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	0	10
	Class=No	0	990

Which model is better?

A

	PREDICTED	
ACTUAL	Class=Yes	Class=No
Class=Yes	0	10
Class=No	0	990

Accuracy: 99%

B

	PREDICTED	
ACTUAL	Class=Yes	Class=No
Class=Yes	10	0
Class=No	500	490

Accuracy: 50%

Which model is better?

A

	PREDICTED	
ACTUAL		Class=Yes	Class=No
	Class=Yes	5	5
	Class=No	0	990

B

	PREDICTED	
ACTUAL		Class=Yes	Class=No
	Class=Yes	10	0
	Class=No	500	490

Alternative Measures

	PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

$$\text{F-measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

Alternative Measures

	PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	10	0
	Class=No	10	980

$$\text{Precision (p)} = \frac{10}{10+10} = 0.5$$

$$\text{Recall (r)} = \frac{10}{10+0} = 1$$

$$\text{F-measure (F)} = \frac{2*1*0.5}{1+0.5} = 0.62$$

$$\text{Accuracy} = \frac{990}{1000} = 0.99$$

Alternative Measures

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	10	0
	Class=No	10	980

$$\text{Precision (p)} = \frac{10}{10+10} = 0.5$$

$$\text{Recall (r)} = \frac{10}{10+0} = 1$$

$$\text{F - measure (F)} = \frac{2*1*0.5}{1+0.5} = 0.62$$

$$\text{Accuracy} = \frac{990}{1000} = 0.99$$

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	1	9
	Class=No	0	990

$$\text{Precision (p)} = \frac{1}{1+0} = 1$$

$$\text{Recall (r)} = \frac{1}{1+9} = 0.1$$

$$\text{F - measure (F)} = \frac{2*0.1*1}{1+0.1} = 0.18$$

$$\text{Accuracy} = \frac{991}{1000} = 0.991$$

Which of these classifiers is better?

A

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	40	10
Class=No	10	40

$$\text{Precision (p)} = 0.8$$
$$\text{Recall (r)} = 0.8$$
$$\text{F - measure (F)} = 0.8$$
$$\text{Accuracy} = 0.8$$

B

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	40	10
Class=No	1000	4000

$$\text{Precision (p)} =\sim 0.04$$
$$\text{Recall (r)} = 0.8$$
$$\text{F - measure (F)} =\sim 0.08$$
$$\text{Accuracy} =\sim 0.8$$

Measures of Classification Performance

	PREDICTED CLASS	
	Yes	No
ACTUAL CLASS Yes	TP	FN
ACTUAL CLASS No	FP	TN

α is the probability that we reject the null hypothesis when it is true. This is a Type I error or a false positive (FP).

β is the probability that we accept the null hypothesis when it is false. This is a Type II error or a false negative (FN).

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

$$ErrorRate = 1 - accuracy$$

$$Precision = Positive\ Predictive\ Value = \frac{TP}{TP + FP}$$

$$Recall = Sensitivity = TP\ Rate = \frac{TP}{TP + FN}$$

$$Specificity = TN\ Rate = \frac{TN}{TN + FP}$$

$$FP\ Rate = \alpha = \frac{FP}{TN + FP} = 1 - specificity$$

$$FN\ Rate = \beta = \frac{FN}{FN + TP} = 1 - sensitivity$$

$$Power = sensitivity = 1 - \beta$$

Alternative Measures

A	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	40	10
	Class=No	10	40

$$\text{Precision (p)} = 0.8$$
$$\text{TPR} = \text{Recall (r)} = 0.8$$
$$\text{FPR} = 0.2$$
$$\text{F−measure (F)} = 0.8$$
$$\text{Accuracy} = 0.8$$

$$\frac{\text{TPR}}{\text{FPR}} = 4$$

B	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	40	10
	Class=No	1000	4000

$$\text{Precision (p)} = 0.038$$
$$\text{TPR} = \text{Recall (r)} = 0.8$$
$$\text{FPR} = 0.2$$
$$\text{F−measure (F)} = 0.07$$
$$\text{Accuracy} = 0.8$$

$$\frac{\text{TPR}}{\text{FPR}} = 4$$

Which of these classifiers is better?

A	PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	10	40
	Class=No	10	40

$$\text{Precision } (p) = 0.5$$
$$\text{TPR} = \text{Recall } (r) = 0.2$$
$$\text{FPR} = 0.2$$
$$F - \text{measure} = 0.28$$

B	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	25	25
	Class=No	25	25

$$\text{Precision } (p) = 0.5$$
$$\text{TPR} = \text{Recall } (r) = 0.5$$
$$\text{FPR} = 0.5$$
$$F - \text{measure} = 0.5$$

C	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	40	10
	Class=No	40	10

$$\text{Precision } (p) = 0.5$$
$$\text{TPR} = \text{Recall } (r) = 0.8$$
$$\text{FPR} = 0.8$$
$$F - \text{measure} = 0.61$$

ROC (Receiver Operating Characteristic)

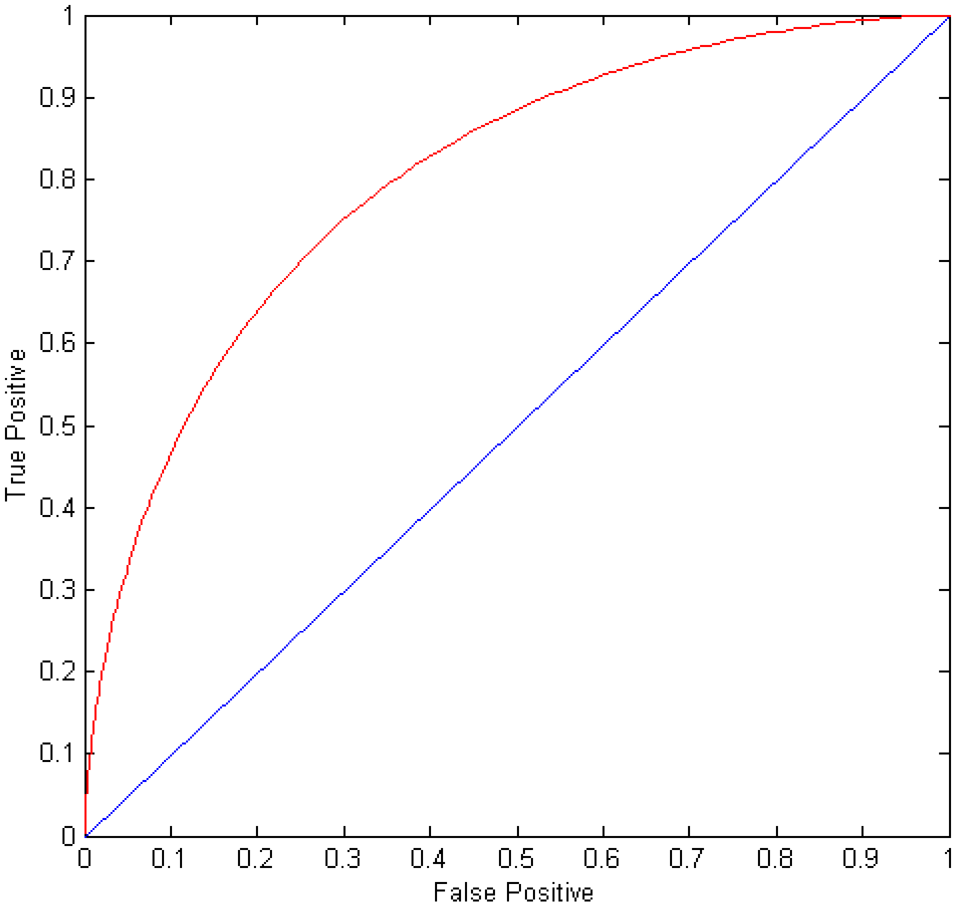
- A graphical approach for displaying trade-off between detection rate and false alarm rate
- Developed in 1950s for signal detection theory to analyze noisy signals
- ROC curve plots TPR against FPR
  - Performance of a model represented as a point in an ROC curve

ROC Curve

(TPR,FPR):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (1,0): ideal

- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class

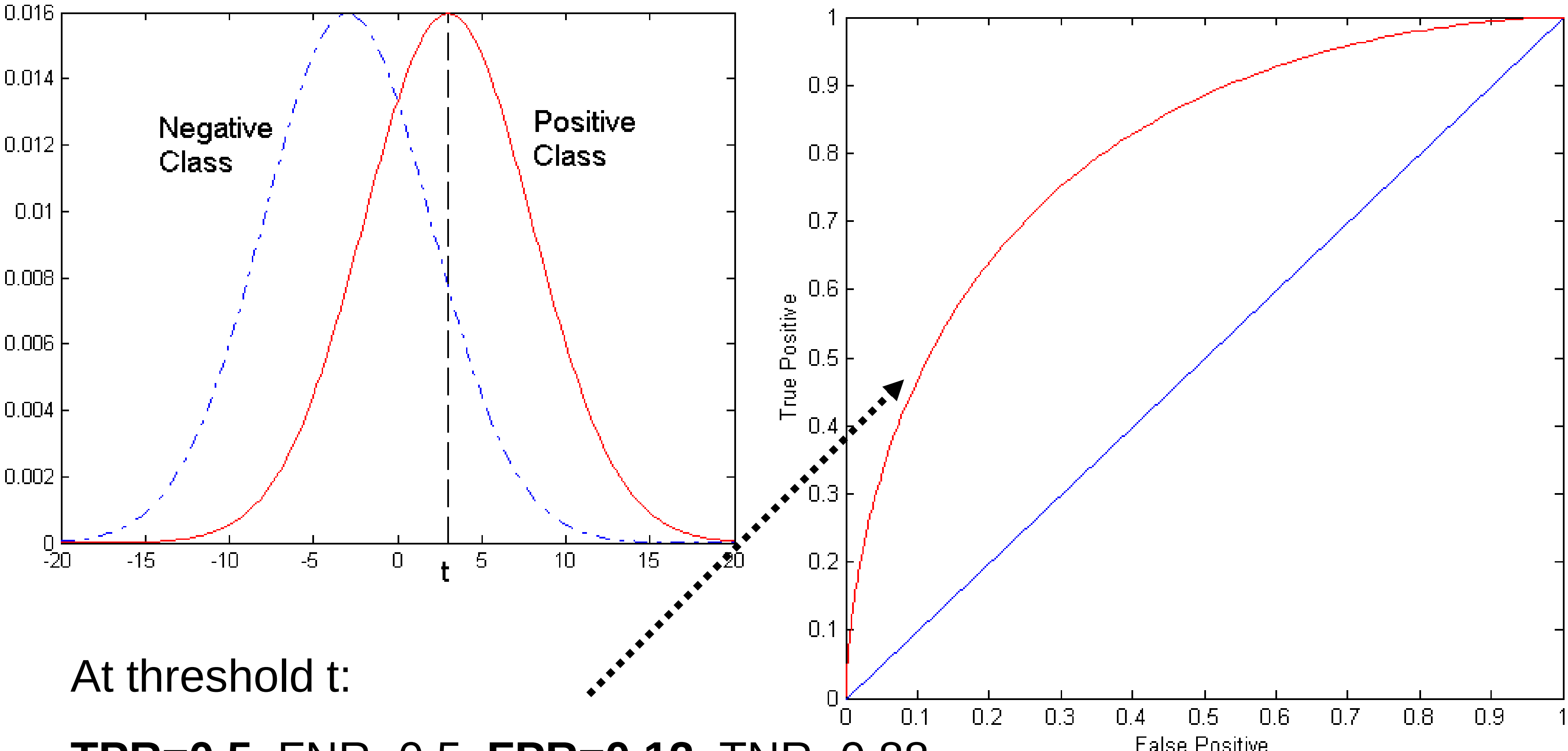


ROC (Receiver Operating Characteristic)

- To draw ROC curve, classifier must produce continuous-valued output
  - Outputs are used to rank test records, from the most likely positive class record to the least likely positive class record
  - By using different thresholds on this value, we can create different variations of the classifier with TPR/FPR tradeoffs
- Many classifiers produce only discrete outputs (i.e., predicted class)
  - How to get continuous-valued outputs?
    - Decision trees, rule-based classifiers, neural networks, Bayesian classifiers, k-nearest neighbors, SVM

ROC Curve Example

- 1-dimensional data set containing 2 classes (positive and negative)
- Any points located at x > t is classified as positive



At threshold t:

TPR=0.5, FNR=0.5, **FPR=0.12**, TNR=0.88

How to Construct an ROC curve

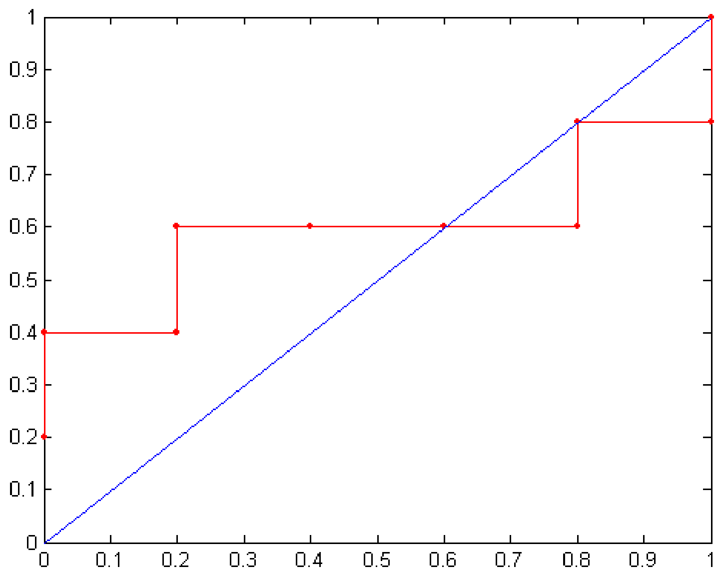
Instance	Score	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use a classifier that produces a continuous-valued score for each instance
  - The more likely it is for the instance to be in the + class, the higher the score
- Sort the instances in decreasing order according to the score
- Apply a threshold at each unique value of the score
- Count the number of TP, FP, TN, FN at each threshold
  - TPR = TP/(TP+FN)
  - FPR = FP/(FP + TN)

How to construct an ROC curve

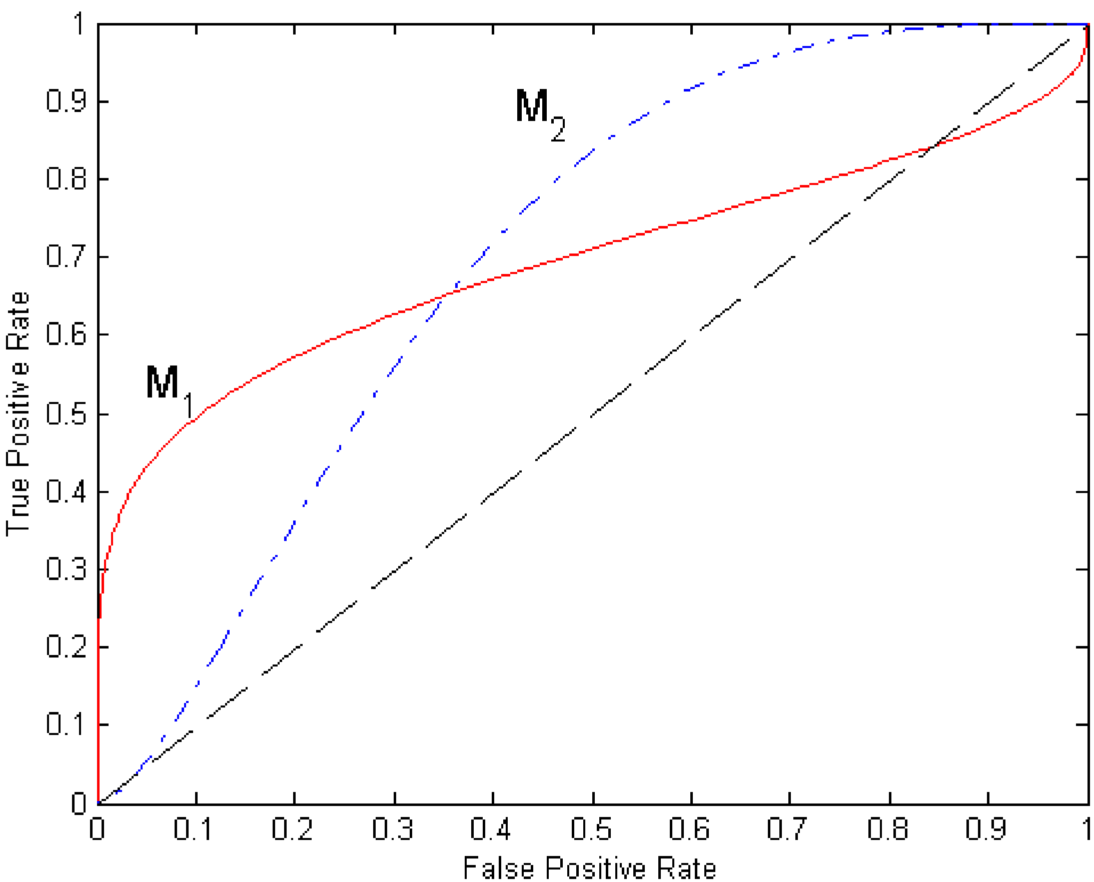
Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve:



Instance	Score	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

Using ROC for Model Comparison



- No model consistently outperforms the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve (AUC)
  - Ideal:
- Area = 1
  - Random guess:
- Area = 0.5

Dealing with Imbalanced Classes - Summary

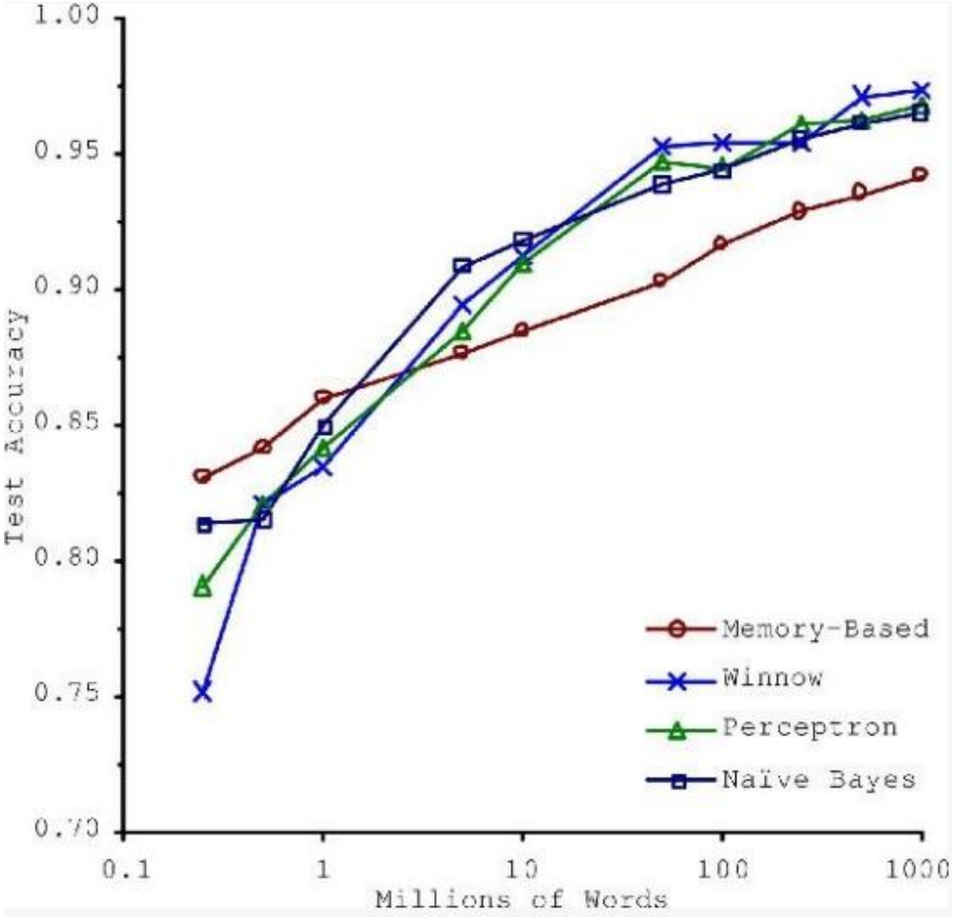
- Many measures exists, but none of them may be ideal in all situations
  - Random classifiers can have high value for many of these measures
  - TPR/FPR provides important information but may not be sufficient by itself in many practical scenarios
  - Given two classifiers, sometimes you can tell that one of them is strictly better than the other
    - C1 is strictly better than C2 if C1 has strictly better TPR and FPR relative to C2 (or same TPR and better FPR, and vice versa)
  - Even if C1 is strictly better than C2, C1's F-value can be worse than C2's if they are evaluated on data sets with different imbalances
  - Classifier C1 can be better or worse than C2 depending on the scenario at hand (class imbalance, importance of TP vs FP, cost/time tradeoffs)

Challenges of Machine Learning

- Training Data
  - Insufficient
  - Non representative
- Model Selection
  - Overfitting
  - Underfitting
- Validation and Testing

Insufficient Training Data

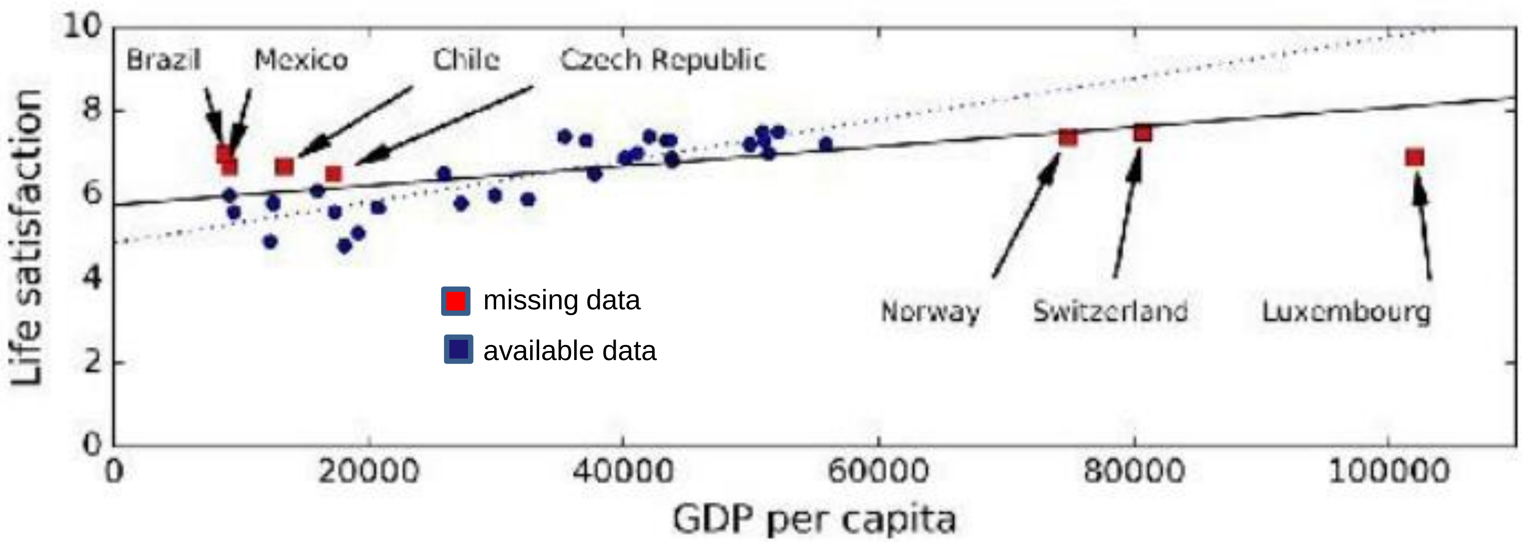
Consider trade-off Between Algorithm development & training data capture



Non-representative Training Data

Training Data be representative of the new cases we want to generalize

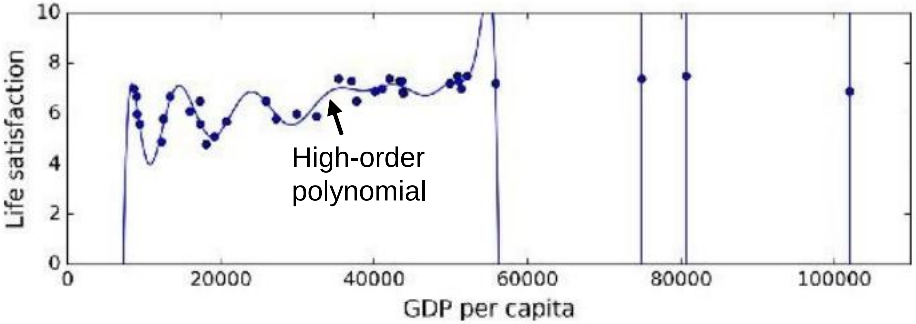
- Small sample size leads to sampling noise
  - Missing data over emphasizes the role of wealth on happiness
- If sampling process is flawed, even large sample size can lead to sampling bias

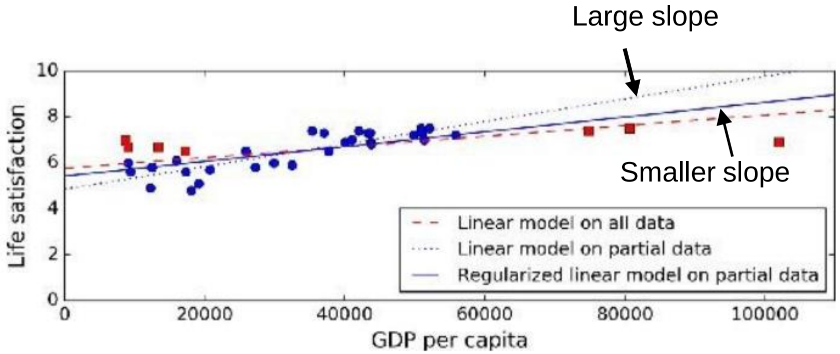


Model Selection

Overfitting or Underfitting

- Overfitting leads to high performance in training set but performs poorly on new data
  - e.g., a high-degree polynomial life satisfaction model that strongly overfits the training data
  - Small training set or sampling noise can lead to model following the noise than the underlying pattern in the dataset
  - Solution: Regularization to constrain the values of parameters





- Underfitting when the model is too simple to learn the underlying structure in the data
  - Select a more powerful model, with more parameters
  - Feed better features to the learning algorithm
  - Reduce regularization

Choice of Hyperparameters

Modern ML models often use a lot of model parameters

- Known as *hyperparameters*
- Model performance depends on choice of parameters
- Each parameter can assume a number of values
  - Real numbers or *categories*
- Exponential number of hyperparameter combinations possible
- Best model correspond to best cross validation performance over the set of hyperparameter combinations
- Expensive to perform
- Some empirical frameworks available for hyperparameter optimization
  - Grid search
  - Random search
  - Bayesian

Evaluating Predictive Performance of a Model

- Want to estimate the generalization performance, the predictive performance of our model on future (unseen) data.

- Want to increase the predictive performance by tweaking the learning algorithm and selecting the best performing model from a given hypothesis space.

- Want to identify the ML algorithm that is best-suited for the problem at hand; thus, we want to compare different algorithms, selecting the best-performing one as well as the best performing model from the algorithm’s hypothesis space.

Evaluation and Validation

Performance of ML algorithms is statistical / predictive

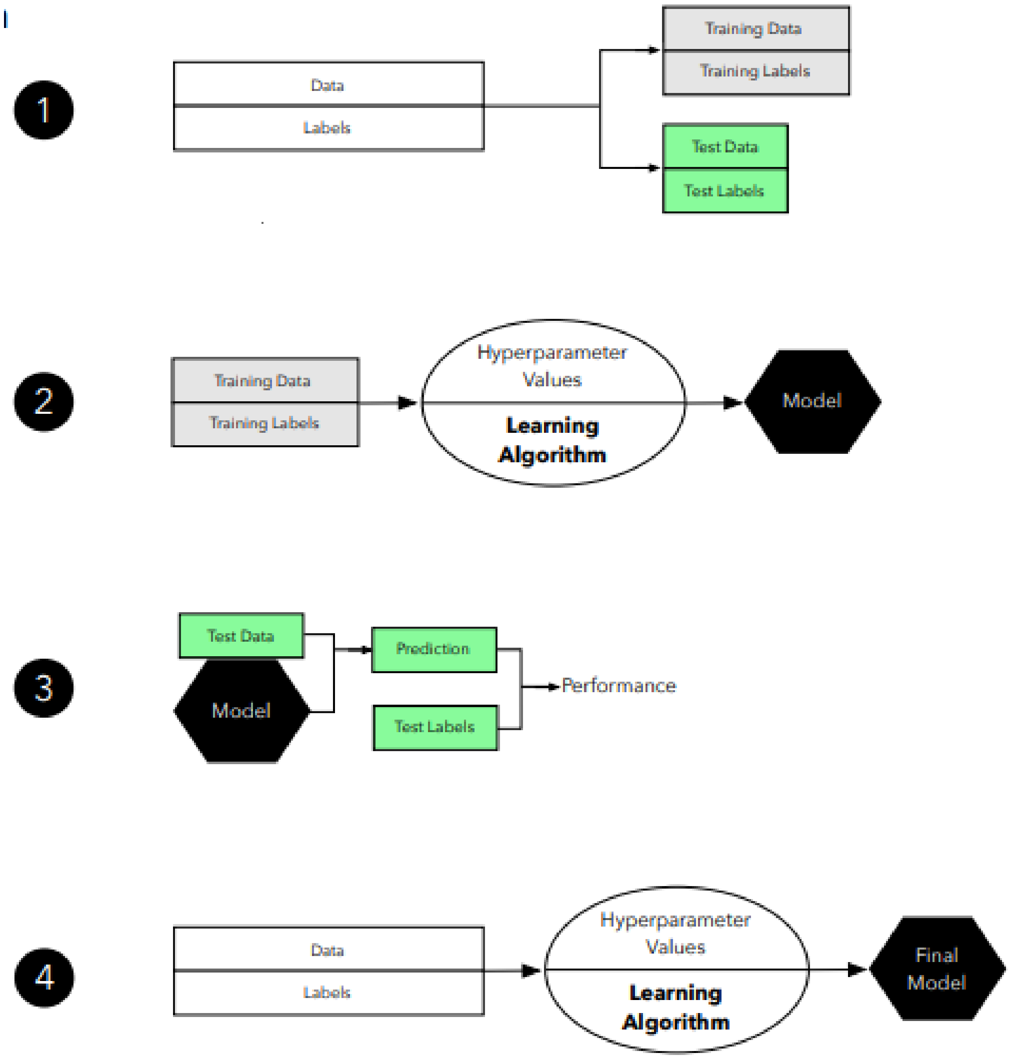
- Good ML algorithms need to work well on test data
  - But test data is often not accessible to the provider of the algorithm
- Common assumption is training data is representative of test data
- Randomly chosen subset of the training data is held out as validation set, *aka* dev set
- Once ML model is trained, its performance is evaluated on validation data
  - Expectation is ML model working well on validation set will work well on unknown test data
- Typically 20-30% of the data is randomly held out as validation data

Cross Validation

K-fold validation is often performed

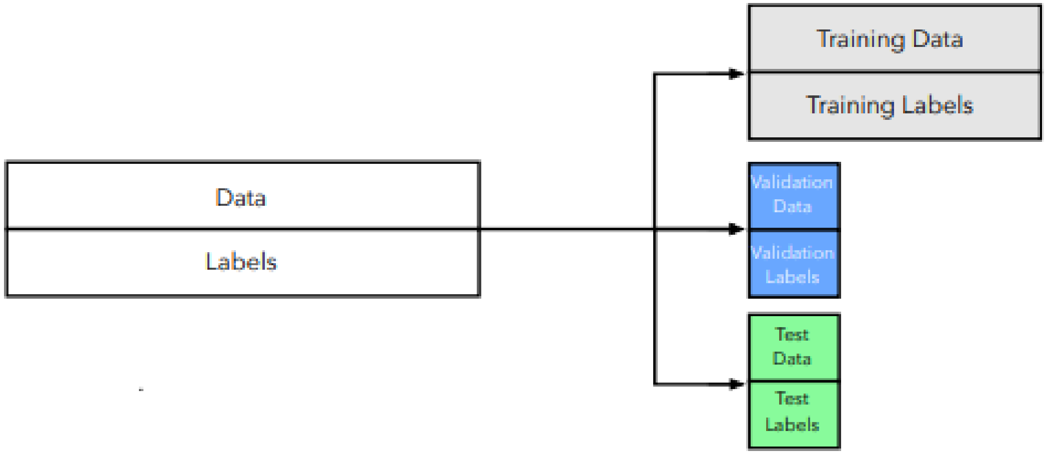
- To reduce the bias of validation set selection process
- Often K is chosen as 10
  - *aka* 10 fold cross validation
- 10 fold cross validation involves
  - randomly selecting the validation set 10 times
  - model generation with 10 resulting training set
  - Evaluate the performance of each on that validation set
  - averaging the performance over the validation sets

Holdout for Model Evaluation

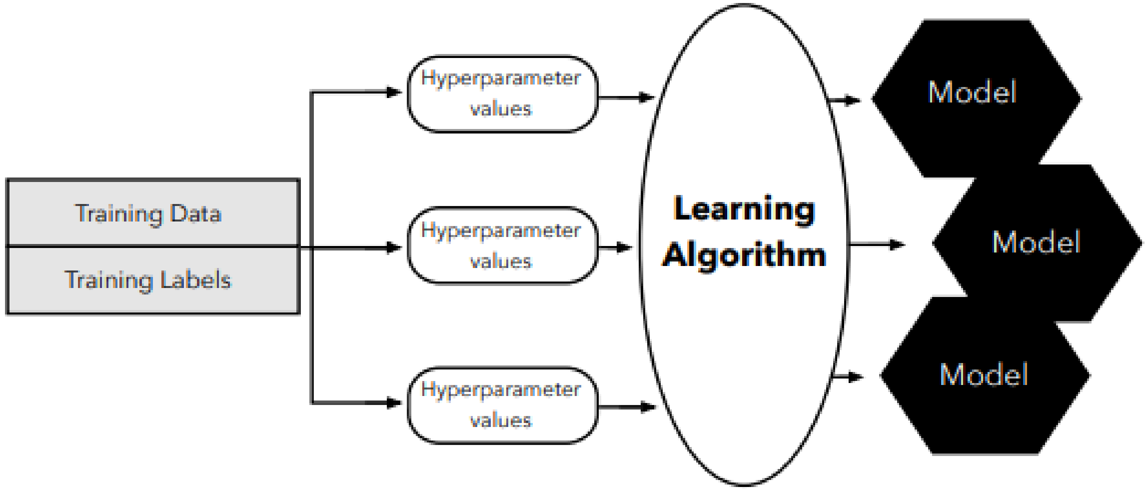


Holdout for Model Selection (1)

1

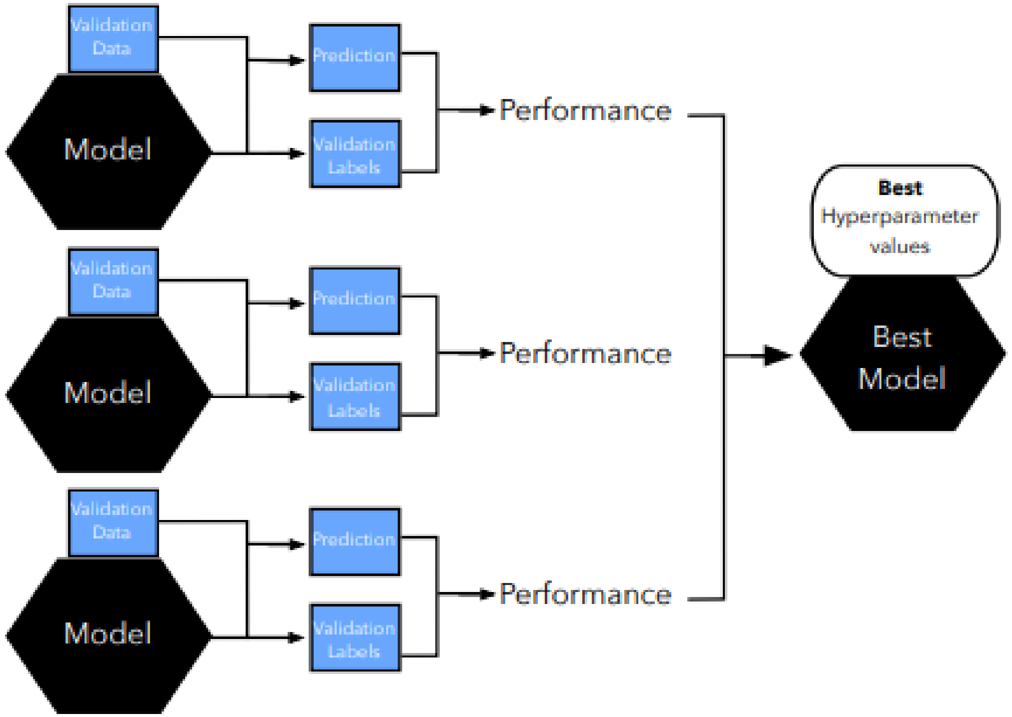


2

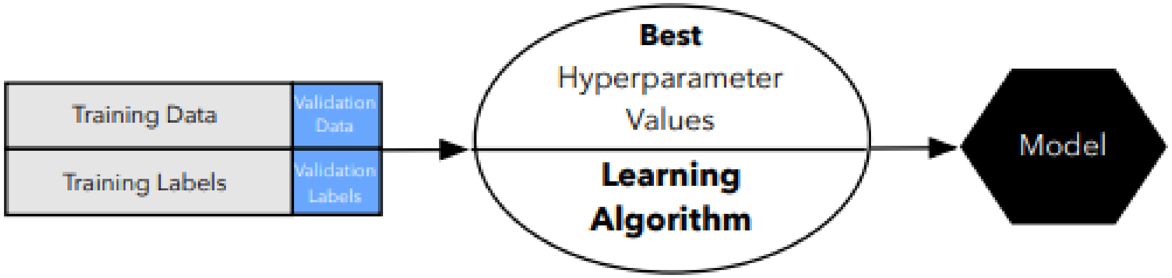


Holdout for Model Selection (2)

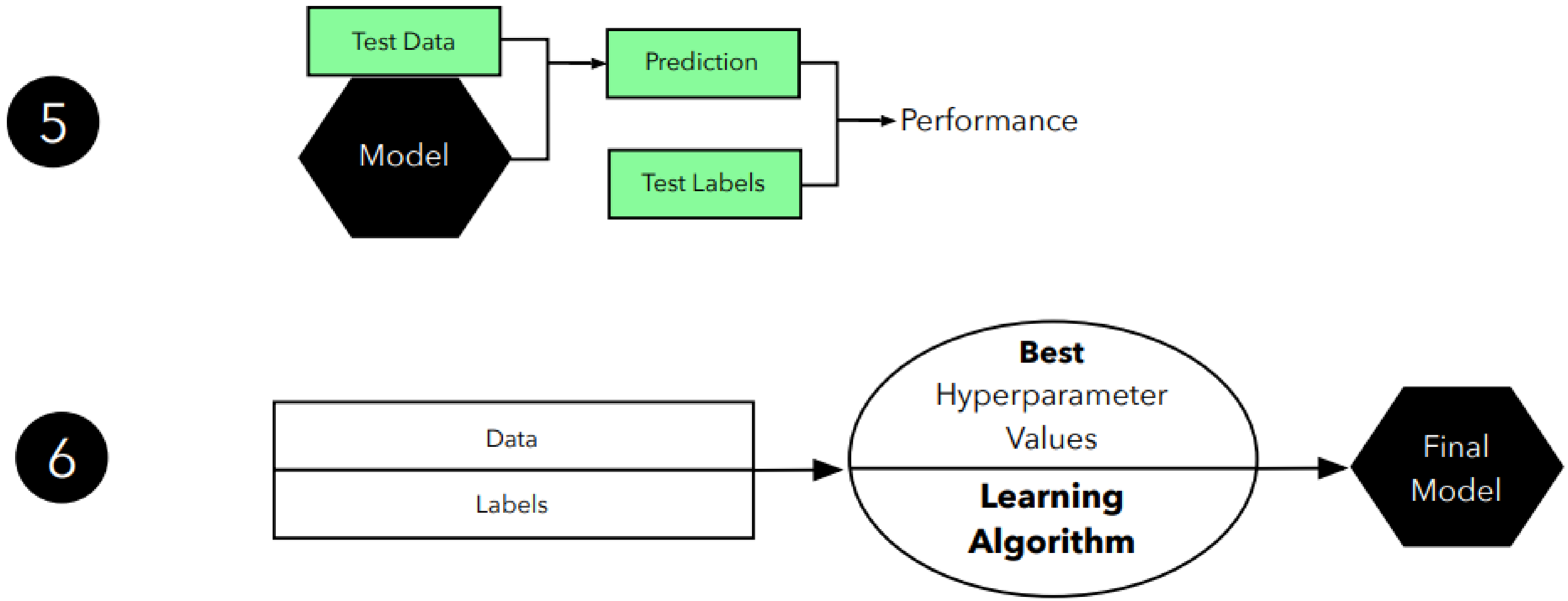
3



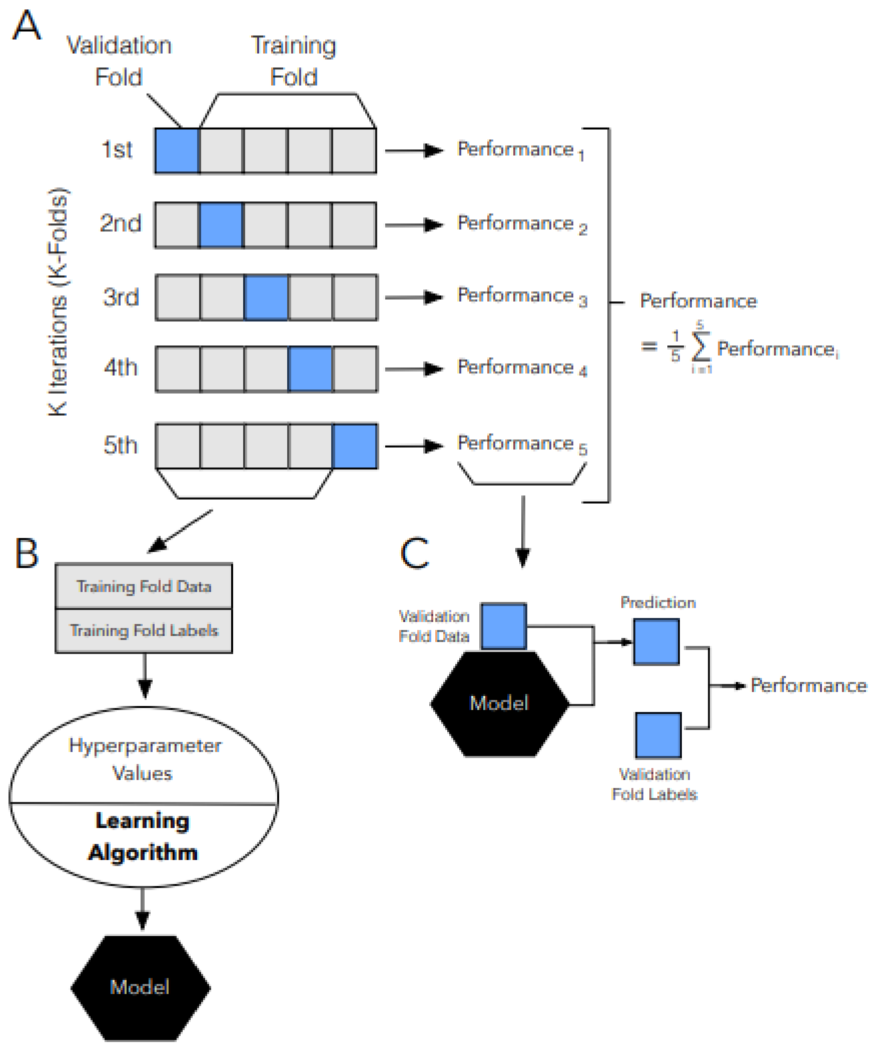
4



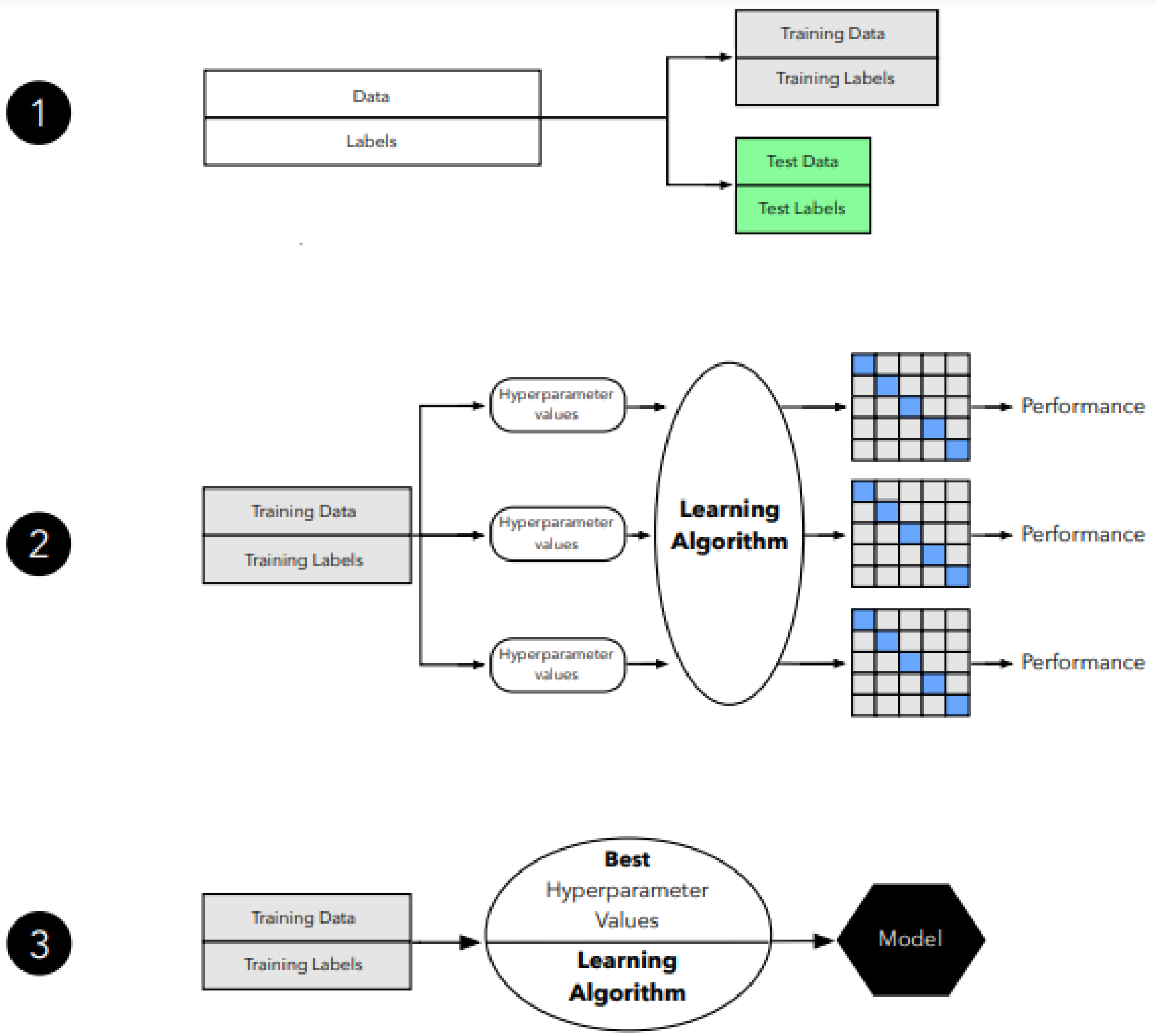
Holdout for Model Selection (3)



K Fold Cross Validation (CV) for Model Evaluation

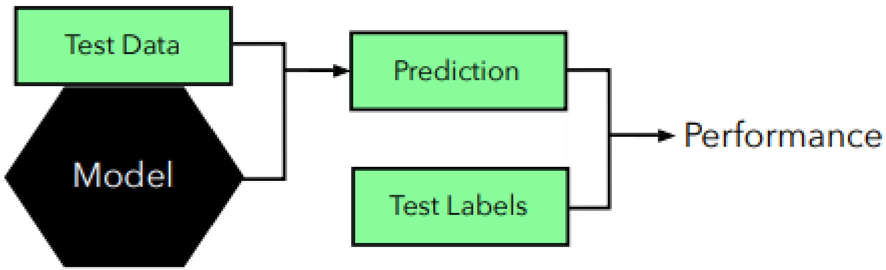


CV for Model Selection (1)

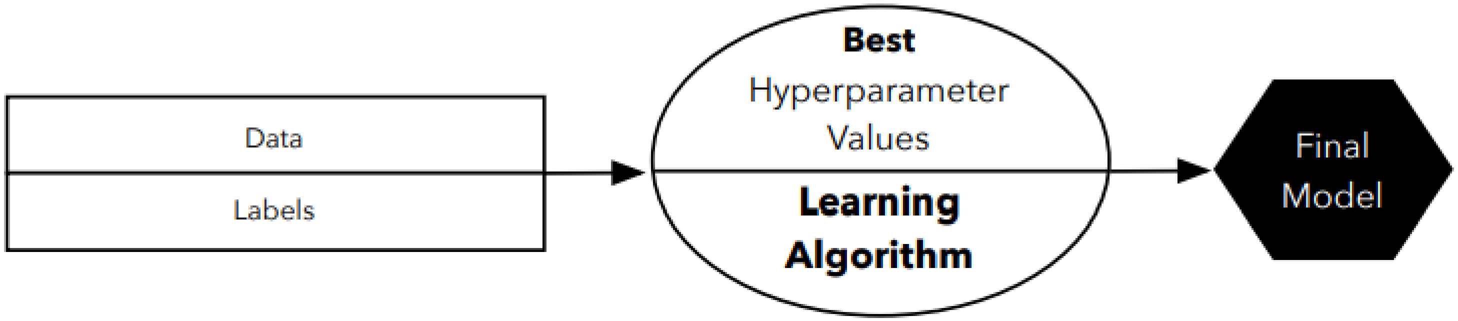


CV for Model Selection (2)

4



5



END
of
Session 2