Data in Data Mining

Ishwar Sethi

What is Data?

- Data means a collection of objects and their attributes
- An attribute is a property or characteristic of an object
  - Examples: eye color of a person, age, marital status etc.
  - Attribute is also known as variable, or feature
- A collection of attributes describe an object
  - Object is also known as pattern, record, case, sample, entity, or instance

Attributes

Id#	Marital Status	Annual Income	Owns House	Credit Card Balance
101	M	58,000	Y	900
102	M	67,700	Y	1200
103	S	43,200	N	0
104	D	54,000	N	1200
105	D	48,000	Y	2300
106	S	38,000	N	300
107	M	78,000	Y	0
108	M	87,000	Y	0
109	S	41,000	N	0
110	M	62,000	N	600

Objects

Feature Vectors in DM

- A data mining system builds models using properties of objects being modeled. These properties are called *features* or *attributes.* It is common to represent the properties of objects as feature vectors.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Sepal width

Sepal length

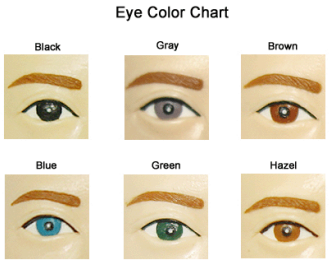
Petal width

Petal length

Size (feet²) x_1	Number of bedrooms x_2	Number of floors x_3	Age of home (years) x_4
2104	5	1	45
1416	3	2	40
1534	3	2	30
852	2	1	36

Types of Attributes

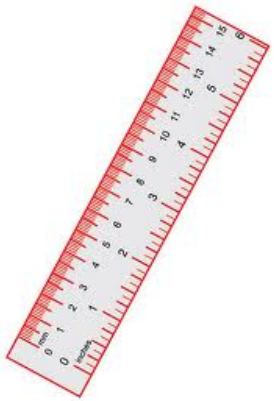
- Categorical or Qualitative Attributes
  - Nominal: Such an attribute has different possible values, each of which is represented by a distinct name or symbol. No ordering is implied by numbers or symbols designating different attribute values. Examples: Eye Color, Zip code, Customer type for a utility company: Residential, commercial or industrial
  - Ordinal: An ordering or gradation is implied on different attribute values. Examples: taste rankings, grades, credit risk of a loan: high, medium, and low, Instructor rating of excellent, very good, good, and poor



Types of Attributes

- Numeric or Quantitative Attributes

  - Interval: The measurement scale used to measure the attribute value has an arbitrary zero value. Example: Temperature in Celsius or Fahrenheit.
  - Ratio: An absolute zero exists on the measurement scale and ratios of two values are meaningful. Examples are mass, length, speed



Properties of Attribute Values

- The type of an attribute depends on which of the following properties it possesses:
  - Distinctness: = ≠
  - Order: < >
  - Addition: + -
  - Multiplication: * /

  - Nominal attribute: distinctness
  - Ordinal attribute: distinctness & order
  - Interval attribute: distinctness, order & addition
  - Ratio attribute: all 4 properties

Attribute Transformations

Attribute Type	Transformation	Comment
Nominal	Any one-to-one mapping	Such a transformation does not change distinctness
Ordinal	An order-preserving change of values via a monotonic function	For example, long, medium and short can be also represented as 3,2 and 1.
Interval	*new_value =a * old_value + b* where a and b are constants	The origin or zero value can be shifted
Ratio	New_value = a * old_value	For example, Length can be measured in meters or feet.

Discrete and Continuous Attributes

- Discrete Attribute
  - Has only a finite set of values
  - Often represented as integer or binary variables.
- Continuous Attribute
  - Has real numbers as attribute values
  - Practically, real values can only be measured and represented using a finite number of digits.
  - Continuous attributes are typically represented as floating-point variables.

Characterizing a Data Set

- Dimensionality

  It refers to the number of attributes possessed by objects in the data set. As the number of attributes grows, the Curse of Dimensionality begins to appear. Thus, dimensionality reduction is often attempted.

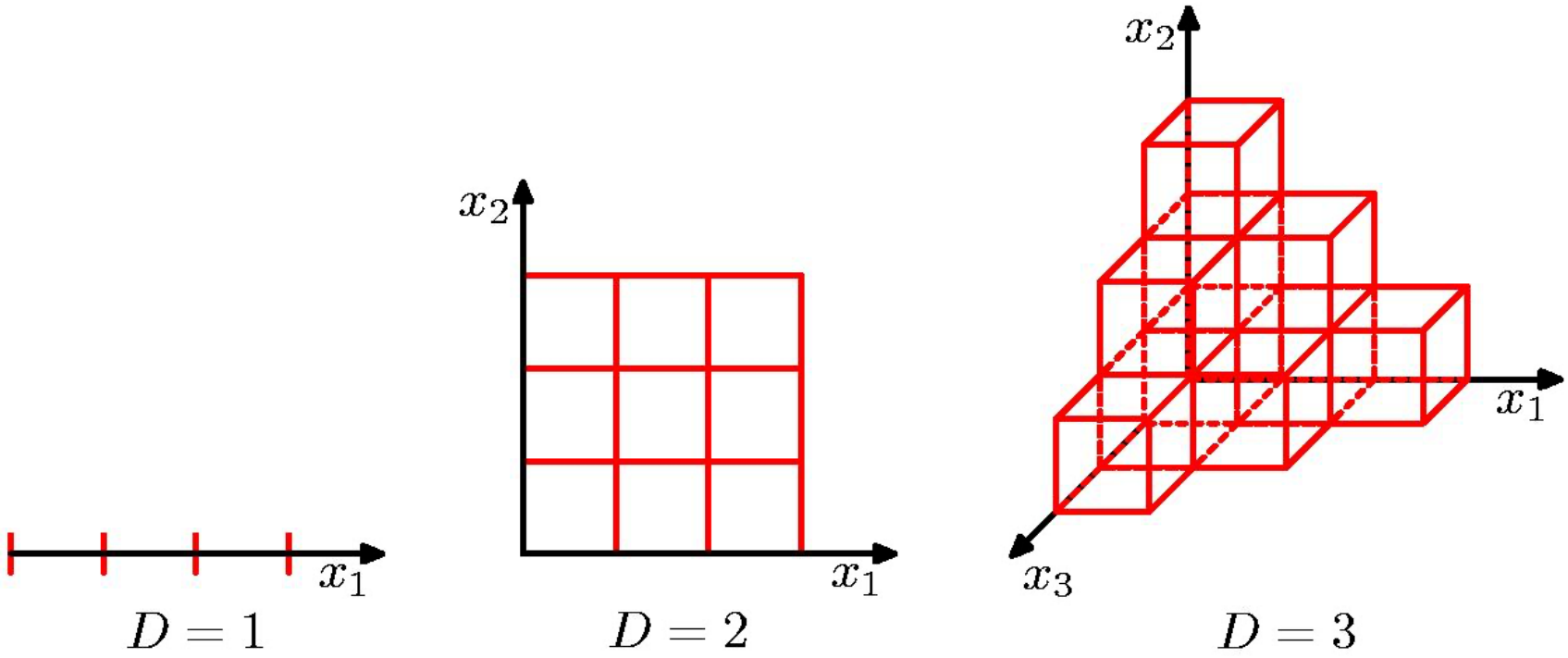
- Sparsity

  Although the number of attributes may be large, each object in the data set may possess only a small subset of attributes. In such cases, we might end up with sparse data. This is common in information retrieval.

- Resolution

  It refers to the level of resolution at which the data is collected.

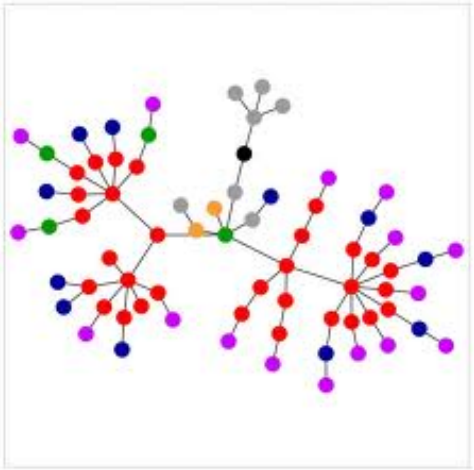
Curse of Dimensionality Illustration



Suppose you want to place a single object in each cell. You can quickly check how exponentially the number of objects needed grows with D.

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
- Ordered Unstructured
  - Image Collection (Spatial ordering)
  - Videos (Spatio-temporal ordering)

Record Data

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

Id#	Marital Status	Annual Income	Owns House	Credit Card Balance
101	M	58,000	Y	900
102	M	67,700	Y	1200
103	S	43,200	N	0
104	D	54,000	N	1200
105	D	48,000	Y	2300
106	S	38,000	N	300
107	M	78,000	Y	0
108	M	87,000	Y	0
109	S	41,000	N	0
110	M	62,000	N	600

Data Matrix

- When 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, and the data 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.

Attribute-1	Attribute-2	Attribute-3	Attribute-4	Attribute-5
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

Term-Document Matrix

- Often used to represent textual/multimedia data.
  - each term is a component (attribute) of the document,
  - the value of each component is the number of times the corresponding term occurs in the document.

Term	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

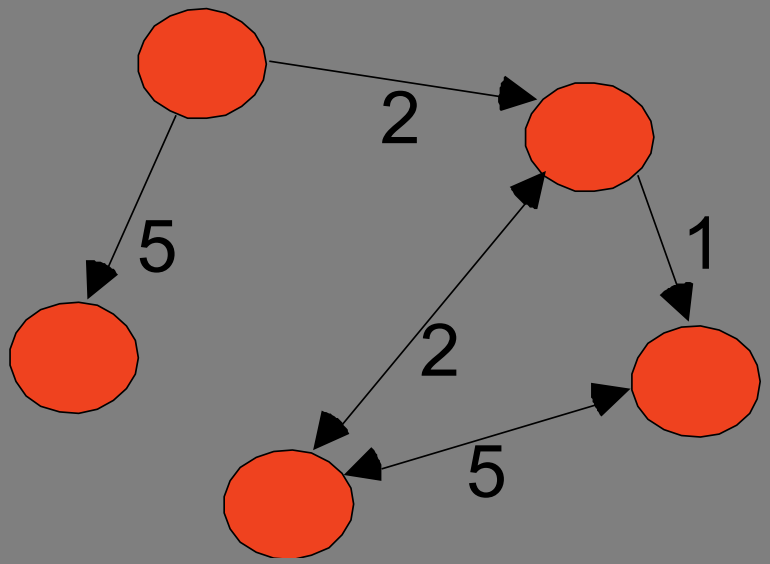
Transaction Data

- A special type of record data, where
  - each record (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.

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

Graph/Link Data

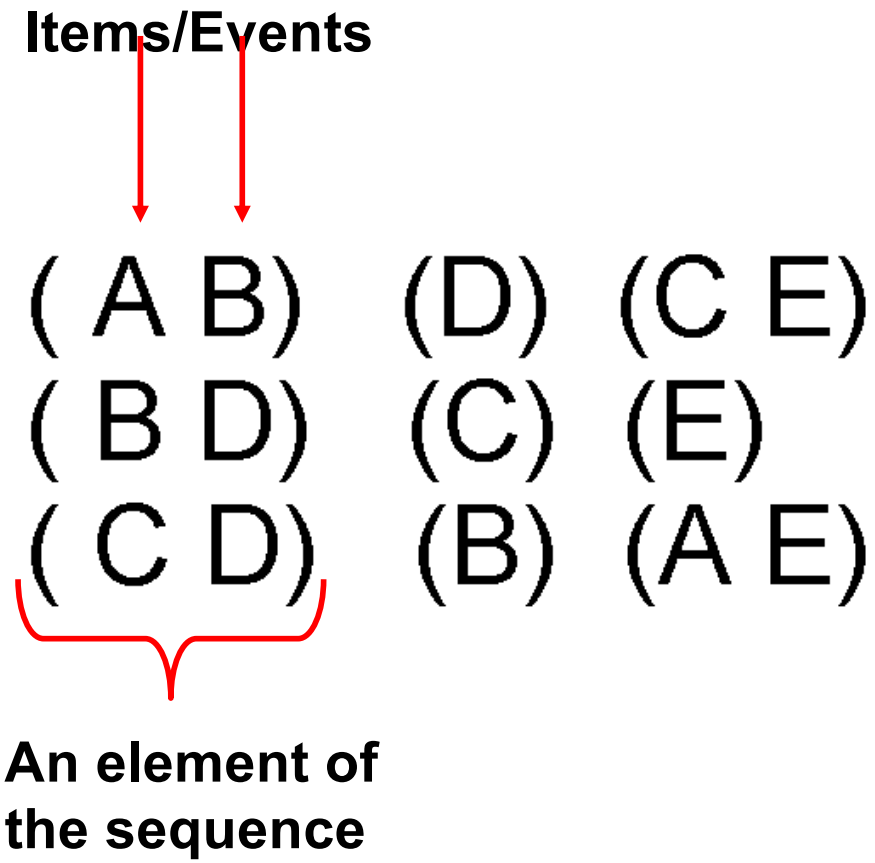
- Such data represents relationships between different objects, for example the web pages via HTML Links.
- The link information can be converted into a link/association matrix.




```
<a href="papers/papers.html#bbbb">
Data Mining </a>
<li>
<a href="papers/papers.html#aaaa">
Graph Partitioning </a>
<li>
<a href="papers/papers.html#aaaa">
Parallel Solution of Sparse Linear System of Equations </a>
<li>
<a href="papers/papers.html#ffff">
N-Body Computation and Dense Linear System Solvers
```

Ordered Data

- Sequences of transactions



Ordered Data

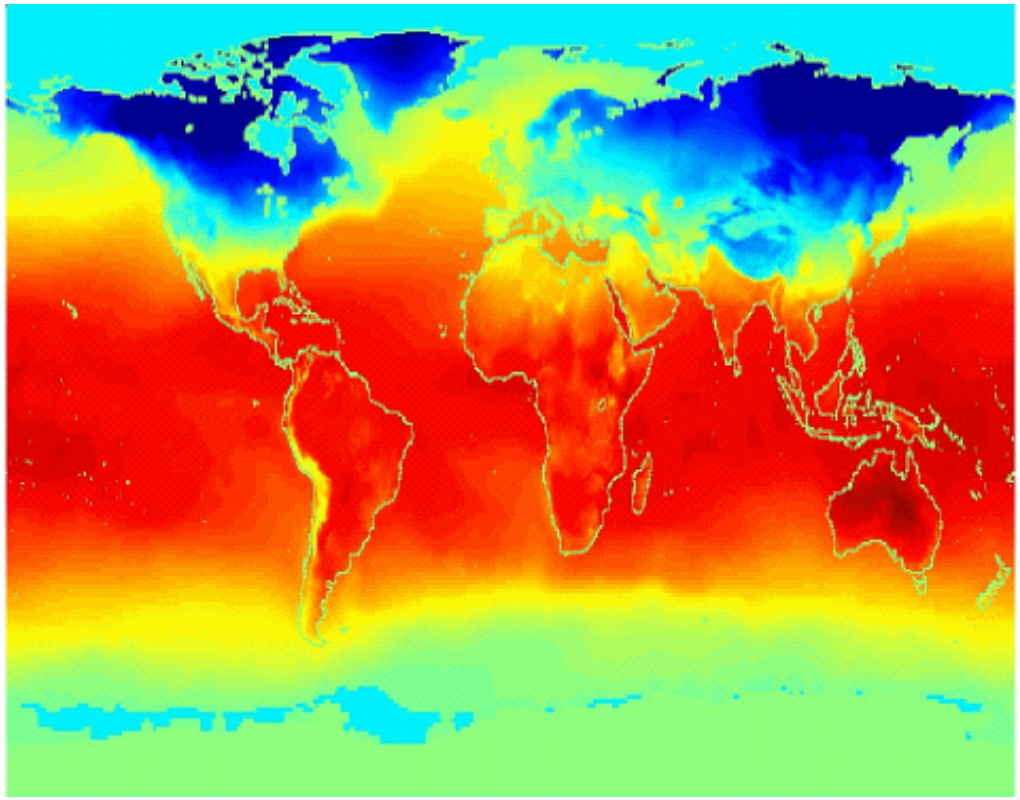
- Genomic sequence data
- Another example will be daily stock price data.

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

Ordered Data

- Spatio-Temporal Data

Average Monthly Temperature of land and ocean



Ordered Data

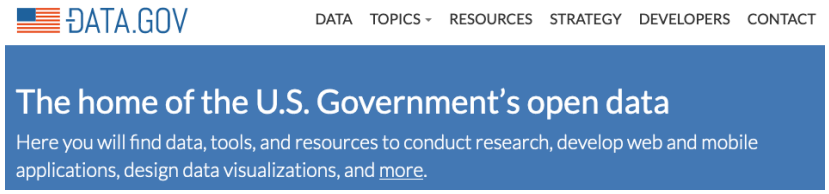
- Video is also an example of spatio-temporal data. Each pixel in a video frame represents spatial information which changes over time as successive frames are viewed. However, it is an unstructured data.

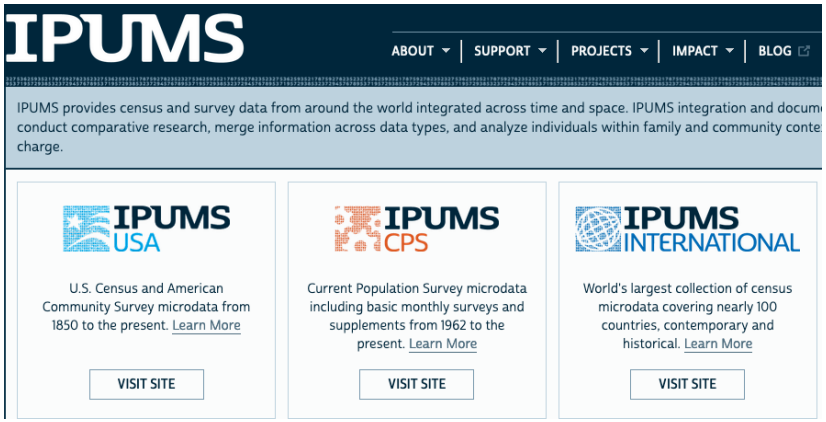
How to Find Data for Mining

- Existing data libraries
- Gathering your own data
- Extracting/scrapping from internet

Existing Data Libraries

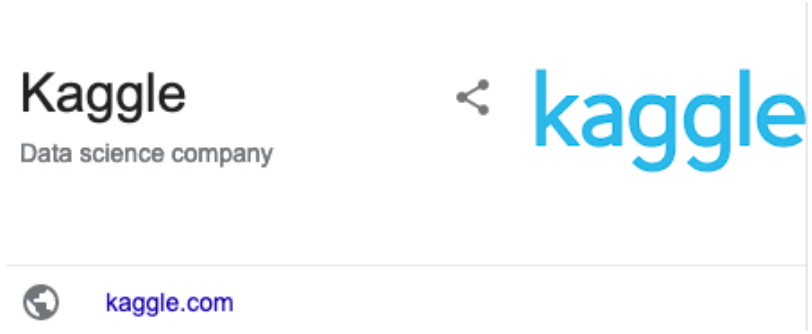
- Some examples of popular government data sources





Existing Data Libraries

- Some examples of websites with data for direct use



Gathering Your Own Data

- This situation applies when you are working on a specific project where you need to collect your own data.
- For example, you may be doing a project on quality control of glass manufacturing. In that case, you will need to gather your own data.

M. Li, S. Feng, I. K. Sethi, J. Luciow and K. Wagner, "Mining production data with neural network & CART," *Third IEEE International Conference on Data Mining*, Melbourne, FL, USA, 2003, pp. 731-734, doi: 10.1109/ICDM.2003.1251019.

Scrapping Data from Web

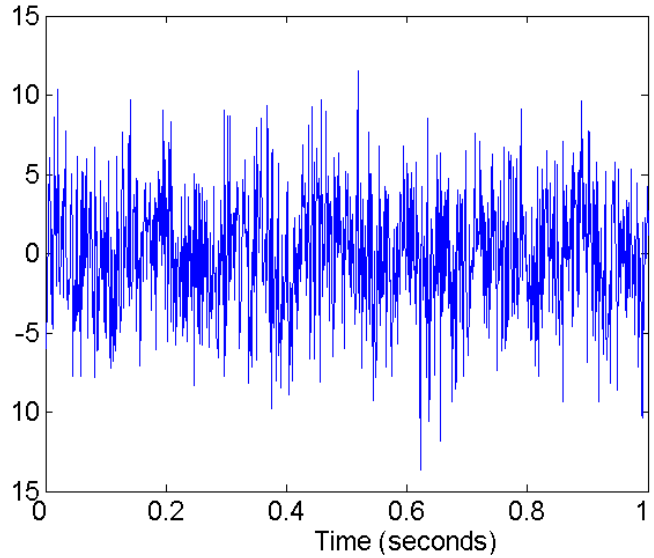
- Web scraping is an automated method used to extract large amounts of data from websites. See the links below for examples of web scrapping.

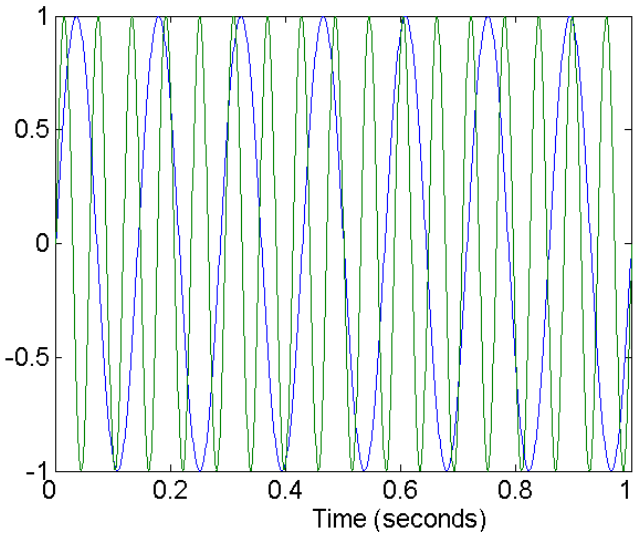
https://www.edureka.co/blog/web-scraping-with-python/

https://realpython.com/python-web-scraping-practical-introduction/

Data Quality Problems

- Duplication and incorrect names/spellings
- Missing values
- Noise and outliers

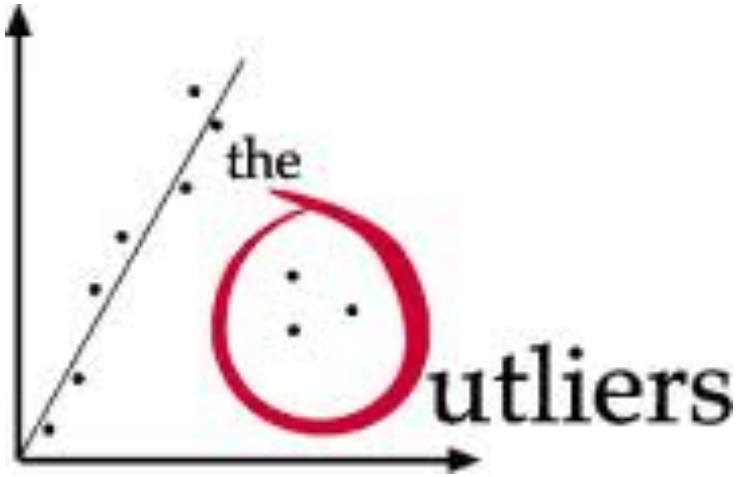




Clean and noisy data

Outliers

- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set. The outliers must be removed prior to model building.



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
- Handling missing values
  - Eliminate data objects with missing values
  - Estimate missing values
  - Replace with all possible values (weighted by their probabilities)
  - Use a surrogate variable

Data Cleaning

- Data cleaning requires a major effort when data comes from multiple heterogeneous sources.
- It involves detecting duplicate values, outliers, missing values, correcting entry errors, and reconciling field names and representations from different sources.

Data Preprocessing

- Aggregation
- Sampling
- Feature extraction
- Discretization and Binarization
- Attribute Transformation
- Dimensionality Reduction/Visualization
- Feature subset selection

Aggregation

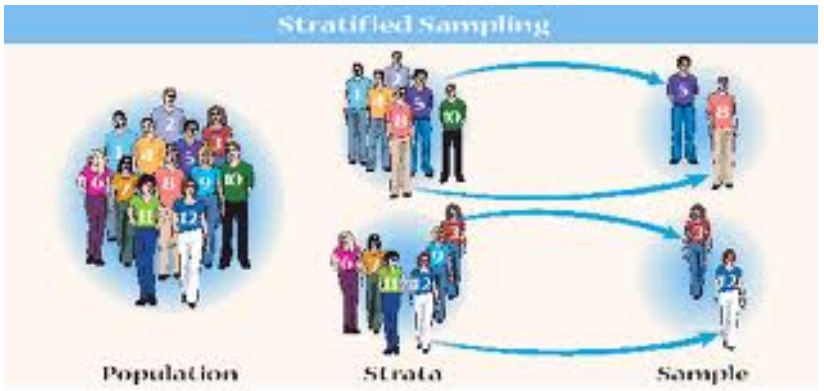
- Combining two or more attributes (or objects) into a single attribute (or object)
- Aggregations helps in
  - Data reduction by reducing the number of attributes or objects
  - Changing the granularity of the data. For example, we can aggregate cities into regions, states etc.
  - Reducing noise or variability in data

Sampling

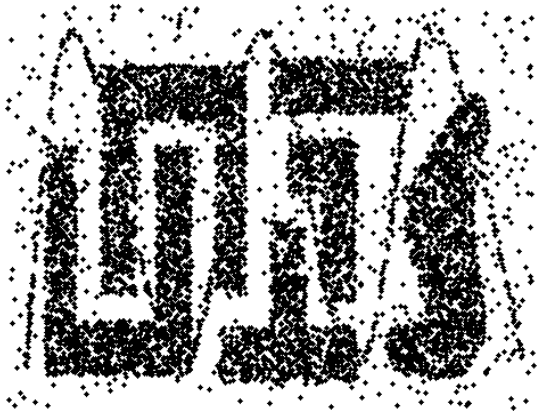
- Sampling involves selecting a subset of data which is representative of the entire data set.
- A sample is representative if it has approximately the same property (of interest) as the original set of data
- Sampling is used in data mining because processing the entire set of data of interest may be too expensive or time consuming.

Types of Sampling

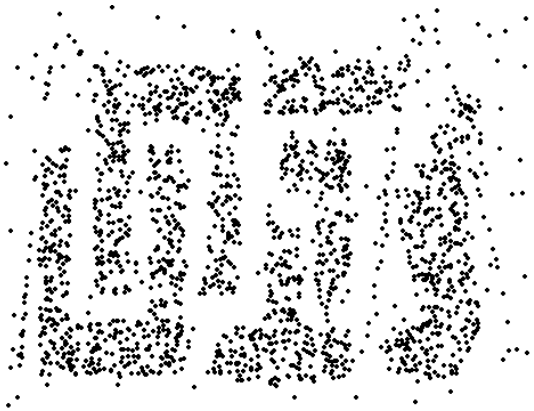
- Simple Random Sampling
  - Equal probability of selecting any particular item
- Sampling without replacement
  - As each item is selected, it is removed from the population
- Sampling with replacement
  - The selected items are not removed from the population. It means the same item can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition



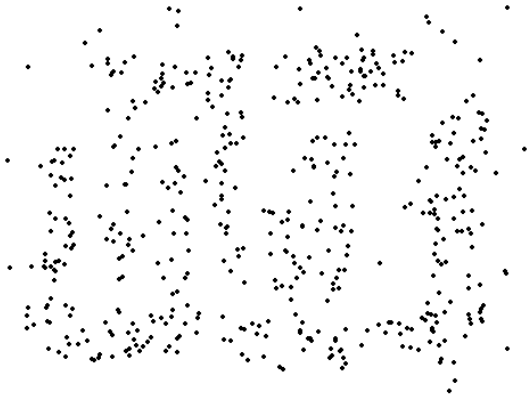
Sample Size



8000 points



2000 Points



500 Points

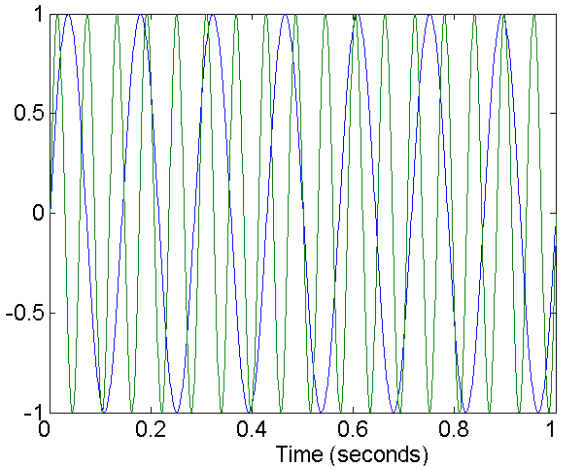
Feature Creation

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

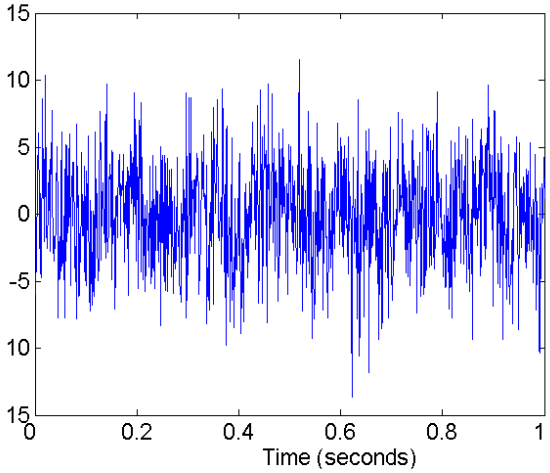
- Three general methodologies:
  - Feature Extraction
    - domain-specific
  - Mapping Data to New Space
  - Feature Construction
    - combining features

Deep learning learns features from raw data. This is one of the reasons for deep learning success.

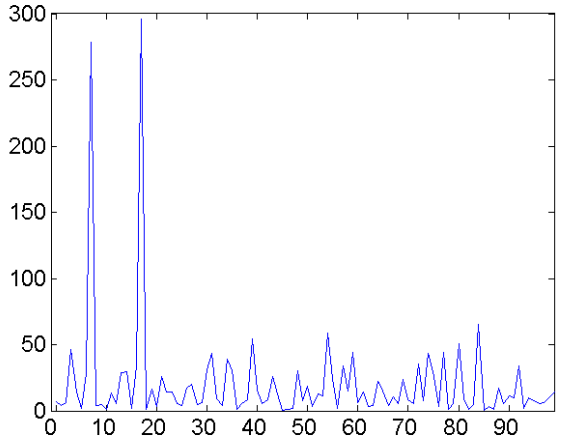
An Example of Mapping Data to a New Space



Two Sine Waves



Two Sine Waves + Noise



Frequency Domain

Binarization

- Binarization of categorical attributes
  - Done using 1-out of- m representation
    - Example: Grade attribute {A, B, C, D, F}

A	B	C	D	E
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

- Binarization of continuous attributes
  - Done by thresholding (attribute_Value > Threshold -> 1; otherwise 0)

Discretization

- We want to discretize age
  - Two possible options
    - Equal interval width
      - For example 0-25, 26-50, 51-75, 76-100
    - Equal frequency
      - Select intervals such that each interval has equal number of data points

Feature Construction Example

- CUPID database containing records of 194046 customers who bought or leased vehicles from Ford. Each record contains 32 attributes as shown on right.

Table1:

Title code:	title	1-3
First name:	first	4-33
Middle init:	middle	34-34
Last name:	lastname	35-69
Adress:	address	70-109
City:	city	110-149
State:	state	150-151
Zip:	zip	152-157
Zip4:	zip4	158-161
Country:	country	162-164
Language:	language	165-166
Home phone:	homepho	167-176
Selling date:	selling	177-184
Vehicle identification number:	vin	185-201
Model year:	model	202-205
Model type:	modelt	206-213
Delivery code:	delivery	214-214
Selling dealer:	selling2	215-220
Assigned dealer:	adealer	221-226
Assigned Ford:	aford	227-232
Assigned LM:	alm	233-238
Vehicle disposal indicator:	vdi	239-239
Global delivery type:	gdt	240-241
Retail lease indicator:	releind	242-243
Customer type:	custype	244-244
New Used:	newused	245-245
Customer ID:	custrid	246-256
FinCon Status:	fincon	257-257
Contract end date:	condate	258-265
Cupid ID:	cupidid	266-273
Vehicle paint code:	vpc	274-275
RM flag:	rm	276-276

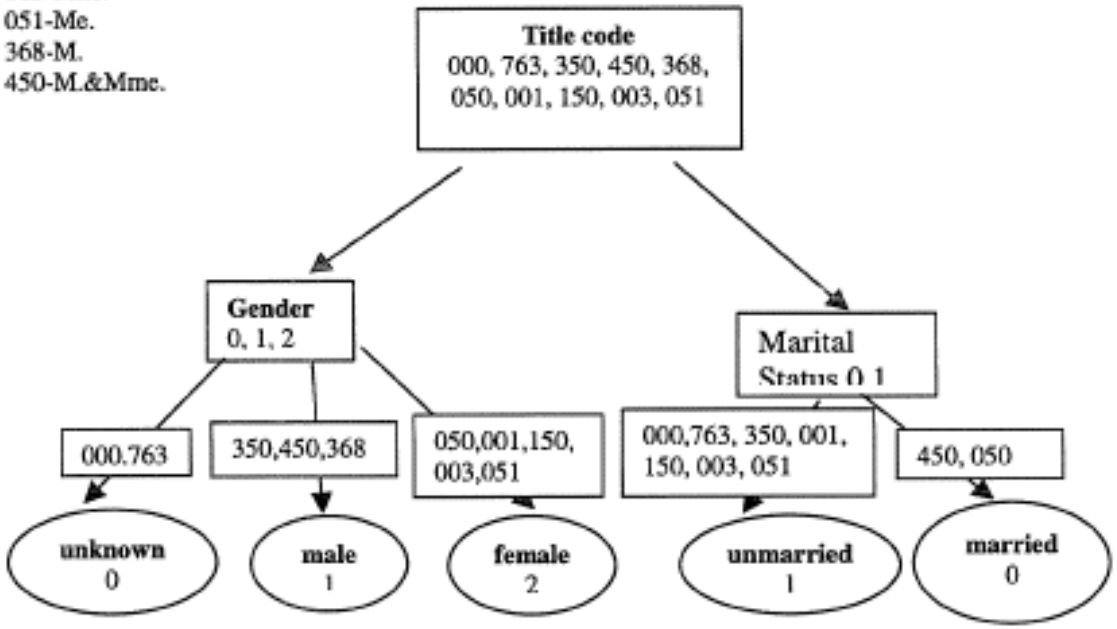
Creating Gender and Marital Status Attributes from Title Code

Title Code: (title) ⇒ **Gender:** **Marital Status:**
(mar_status)

000-Blank or Company
350-Mr.
050-Mrs.
001-Miss.
450-Mr.&Mrs.
763-Dr.
150-Ms.
003-Mlle.
051-Me.
368-M.
450-M.&Mme.

0-Unknown
1-Male
2-Female

0-Married
1-Unmarried

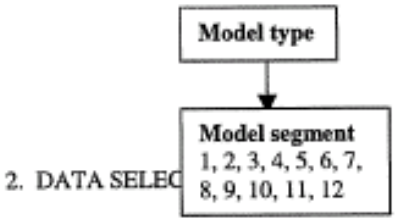


Grouping Attribute Values to Create a New Feature

Model Type: (modelt)
ESCORT
EXPLORER
AEROSTAR
ASPIRE
BRONCO
CLUBWAGN
CONTINEN
MOUNTAIN, NAVIGATO
CONTOUR
COUGAR
CROWNVIC
ECONOLIN
EXPEDITI
F-SERIES
GRANDMAR
MARK
MOUNTAIN
MUSTANG
MYSTIQUE
NAVIGATO
PROBE
RANGER
SABLE
TAURUS
TBIRD
TEMPO
TOWNCAR
TRACER
VILLAGER
WINDSTAR

Model Segmentation: (mod_segm)
1- ASPIRE, CONTINEN
2- TEMPO
3- ESCORT, TRACER
4- COUGAR, MARK, MUSTANG, PROBE, TBIRD
5- CONTOUR, MYSTIQUE, SABLE, TAURUS
6- AEROSTAR, VILLAGER, WINDSTAR
7- EXPLORER, BRONCO, EXPEDITI,

8- F-SERIES
9- CLUBWAGN, ECONOLIN
10- CROWNVIC, GRANDMAR
11- TOWNCAR
12- RANGER



Attribute Transformation

- 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|
- Standardization and Normalization
  - Typically used to ensure that all instances of all attributes lie in some standard range. This way no particular attribute is capable of dominating calculations or learning.

Similarity and Dissimilarity

- Similarity
  - Numerical measure of how alike two data objects are.
  - Is higher when objects are more alike.
  - Often normalized to lie in the range [0,1]
- Dissimilarity
  - Numerical measure of how different are two data objects
  - Lower when objects are more alike
  - Minimum dissimilarity is often 0
  - Upper limit varies
- Proximity refers to a similarity or dissimilarity

Similarity/Dissimilarity for Simple Attributes

p and *q* are the attribute values for two data objects.

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert p-q \rvert}{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{\lvert p-q \rvert}{n-1}$
Interval or Ratio	$d = \lvert p - q \rvert$	$s = -d$, $s = \frac{1}{1+d}$ or $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

Table 5.1. Similarity and dissimilarity for simple attributes

Euclidean Distance

- Euclidean Distance

$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

Where n is the number of dimensions (attributes) and p_k and q_k are, respectively, the k^{th} attributes (components) or data objects p and q.

- Standardization is necessary, if scales differ.

Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

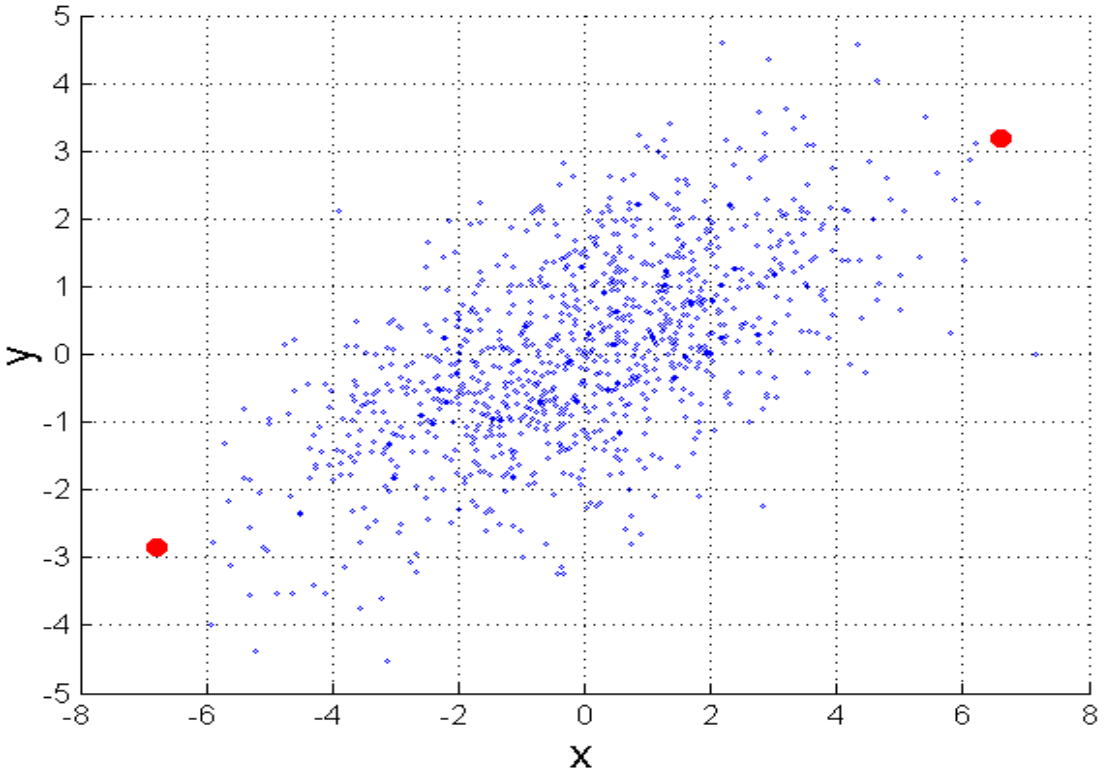
Where r is a parameter, n is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects p and q.

Minkowski Distance: Examples

- $r = 1$. City block (Manhattan, taxicab, L_1 norm) distance.
  - A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors
- $r = 2$. Euclidean distance
- $r \rightarrow \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component of the vectors

Mahalanobis Distance

$$mahalanobis(p,q) = (p-q)\Sigma^{-1}(p-q)^T$$



Σ is the covariance matrix of the input data *X*

$$\Sigma_{j,k} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{ij} - \overline{X}_j)(X_{ik} - \overline{X}_k)$$

This distance function considers how much a feature value varies over the entire dataset.

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.

Mahalanobis Distance in 2-dimensions with diagonal covariance matrix

Assuming no correlation, our covariance matrix is:

$$S = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

The inverse of a 2x2 matrix can be found using the following:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \qquad A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Applying this to get the inverse of the covariance matrix:

$$S^{-1} = \frac{1}{\sigma_1^2 \sigma_2^2} \begin{bmatrix} \sigma_2^2 & 0 \\ 0 & \sigma_1^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{bmatrix}$$

Now we can work through the Mahalanobis equation to see how we arrive at our earlier variance-normalized distance equation.

$$d_M(x, y) = \sqrt{(x - y)^T S^{-1} (x - y)}$$

$$= \sqrt{\begin{bmatrix} x_1 - y_1 & x_2 - y_2 \end{bmatrix} \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{bmatrix} \begin{bmatrix} x_1 - y_1 \\ x_2 - y_2 \end{bmatrix}}$$

$$= \sqrt{\begin{bmatrix} \frac{x_1 - y_1}{\sigma_1^2} & \frac{x_2 - y_2}{\sigma_2^2} \end{bmatrix} \begin{bmatrix} x_1 - y_1 \\ x_2 - y_2 \end{bmatrix}}$$

$$= \sqrt{\frac{(x_1 - y_1)^2}{\sigma_1^2} + \frac{(x_2 - y_2)^2}{\sigma_2^2}}$$

The final equation tells that features showing less variability are weighted more in the distance calculation.

Distance Calculation Example

- Point P = $[3.9\ 4.3]^T$, Point Q = $[6.2\ 5.2]^T$
- Covariance matrix = [3.05 0.27; 0.27 1.93]
- Euclidean distance between P & Q = $SQRT[(3.9 – 6.2)^2 + (4.3 – 5.2)^2] = 2.47$
- City Block distance between P & Q = [abs(3.9 – 6.2) + abs(4.3-5.2)] = 3.2
- Chess Board distance ($L_{infinity}$ norm) = max{abs(3.9 – 6.2), abs(4.3-5.2)} = 2.3
- We need inverse of covariance matrix -> [0.332 -0.047; -0.047 0.526]
- Mahalanobis distance = [-2.3 -0.9] [0.332 -0.047; -0.047 0.526] $[-2.3\ -0.9]^T$ → ? 1.41

Common Properties of a Distance

- Distances, such as the Euclidean distance, have some well-known properties.

  1. $d(p, q) \geq 0$ for all p and q and $d(p, q) = 0$ only if $p = q$. (Positive definiteness)
  2. $d(p, q) = d(q, p)$ for all p and q. (Symmetry)
  3. $d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q, and r. (Triangle Inequality)

  where $d(p, q)$ is the distance (dissimilarity) between points (data objects), p and q.

- A distance that satisfies these properties is a metric

Common Properties of a Similarity Measure

- Similarities, also have some well-known properties.

  1. *s(p, q) = 1* (or maximum similarity) only if *p = q*.

  2. *s(p, q) = s(q, p)* for all *p* and *q*. (Symmetry)

  where *s(p, q)* is the similarity between points (data objects), *p* and *q*.

Similarity Between Binary Vectors

- Common situation is that objects, *p* and *q*, have only binary attributes
- Compute similarities using the following quantities

  M_{01} = the number of attributes where p was 0 and q was 1
  M_{10} = the number of attributes where p was 1 and q was 0
  M_{00} = the number of attributes where p was 0 and q was 0
  M_{11} = the number of attributes where p was 1 and q was 1
- Simple Matching and Jaccard Coefficients

  SMC = number of matches / number of attributes
  $= (M_{11} + M_{00}) / (M_{01} + M_{10} + M_{11} + M_{00})$
  J = number of 11 matches / number of not-both-zero attributes values

  $= (M_{11}) / (M_{01} + M_{10} + M_{11})$

Cosine Similarity

- If d_1 and d_2 are two document vectors, then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2|| ,$$

where • indicates vector dot product and || *d* || is the length of vector *d*.

- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**
d_2 = **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2$= 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5

$||d_1||$ = $(3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)^{0.5}$ = $(42)^{0.5}$ = 6.481

$||d_2||$ = $(1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{0.5}$ = $(6)^{0.5}$ = 2.245

$\cos(d_1, d_2)$ = .3150

Extended Jaccard Coefficient (Tanimoto)

- Variation of Jaccard for continuous or count attributes
  - Reduces to Jaccard for binary attributes

$$T(p,q) = \frac{p \bullet q}{\|p\|^2 + \|q\|^2 - p \bullet q}$$

Correlation

- Correlation measures the linear relationship between objects
- Its given by the following relationship for arrays X and Y

$$Correl(X,Y) = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sqrt{\sum (x-\bar{x})^2 \sum (y-\bar{y})^2}}$$

$$\begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

Covariance matrix in 2-d

Correlation coefficient

Example

x	y
2	2
2	5
6	5
7	3
4	7
6	4
5	3
4	6
2	5
1	3

Calculate the correlation coefficient?

1. Mean of x and y → 3.9 4.3

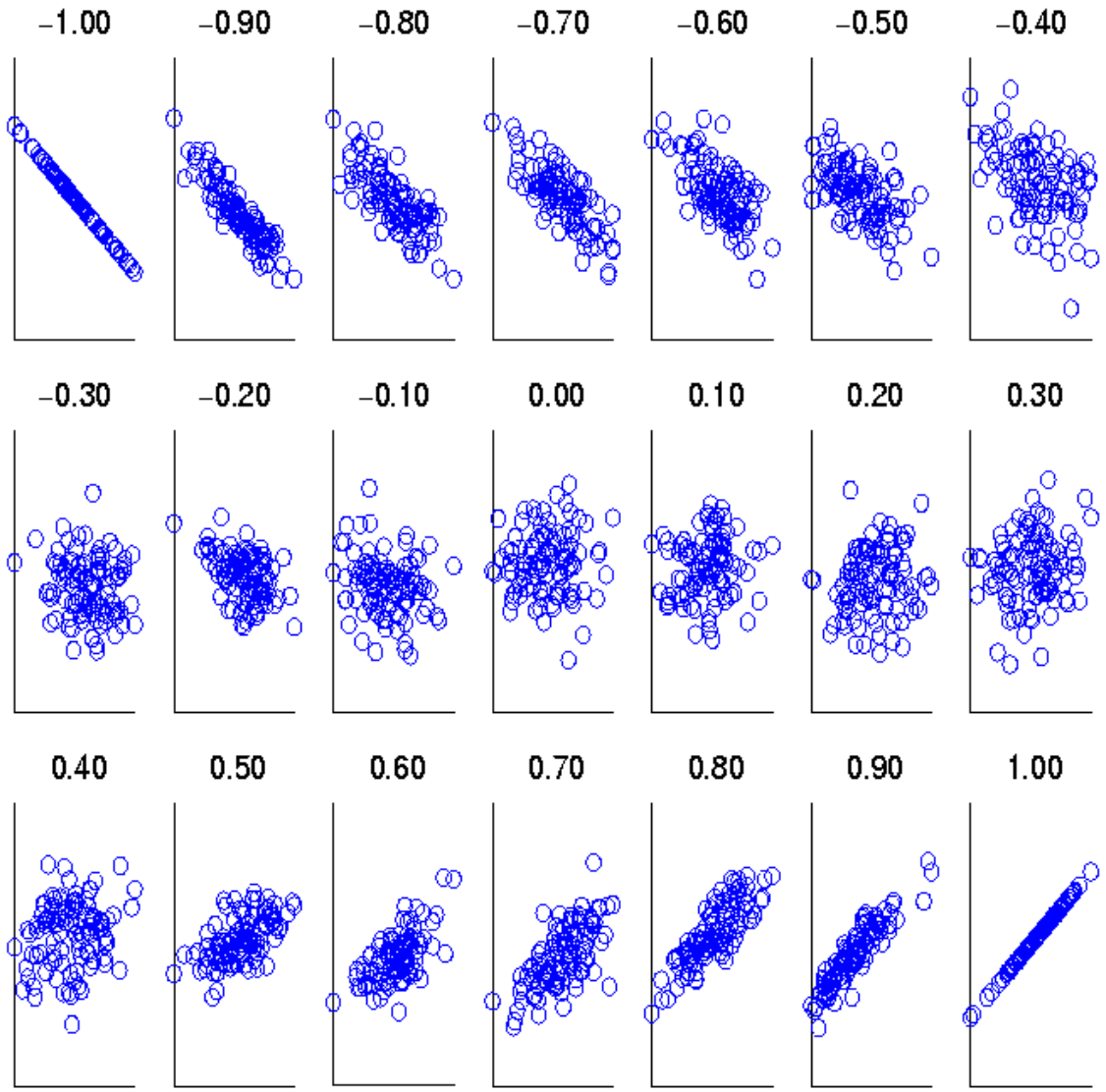
$\sum (x-\bar{x})(y-\bar{y})$ → 1.3

$\sqrt{\sum (x-\bar{x})^2 \sum (y-\bar{y})^2}$ → sqrt(38.9*22.1)

X-mean x	y-mean y
-1.9	-2.3
-1.9	0.7
2.1	0.7
3.1	-1.3
0.1	2.7
2.1	-0.3
1.1	-1.3
0.1	1.7
-1.9	0.7
-2.9	-1.3

Correlation Coefficient → 1.3/sqrt(38.9*22.1)
→0.0443

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.

General Approach for Combining Similarities

- Sometimes attributes are of many different types, but an overall similarity is needed.

1. For the k^{th} attribute, compute a similarity, s_k, in the range $[0, 1]$.
2. Define an indicator variable, δ_k, for the k_{th} attribute as follows:

$$\delta_k = \begin{cases} 0 & \text{if the } k^{th} \text{ attribute is a binary asymmetric attribute and both objects have a value of 0, or if one of the objects has a missing values for the } k^{th} \text{ attribute} \\ 1 & \text{otherwise} \end{cases}$$

3. Compute the overall similarity between the two objects using the following formula:

$$similarity(p, q) = \frac{\sum_{k=1}^{n} \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

Using Weights to Combine Similarities

- May not want to treat all attributes the same.
  - Use weights w_k which are between 0 and 1 and sum to 1.

$$similarity(p,q) = \frac{\sum_{k=1}^{n} w_k \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

$$distance(p,q) = \left(\sum_{k=1}^{n} w_k |p_k - q_k|^r \right)^{1/r}$$

Descriptive Statistics

- Descriptive statistics is used to obtain a feel for data. Some examples of descriptive statistics are:
  - Mean
  - Median
  - Mode
  - Percentiles
  - Variance
  - Standard deviation

Descriptive Statistics Example: Iris Sample Data Set

- Three flower types (classes):
  - Setosa
  - Virginica
  - Versicolour
- Four attributes
  - Sepal width and length
  - Petal width and length

Virginica. Robert H. Mohlenbrock. USDA NRCS. 1995. Northeast wetland flora: Field office guide to plant species. Northeast National Technical Center, Chester, PA. Courtesy of USDA NRCS Wetland Science Institute.

Mean and Median

- The mean is the most common measure of the location of a set of points.
- However, the mean is very sensitive to outliers.
- Thus, the median or a trimmed mean is also commonly used.

$$\text{mean}(x) = \overline{x} = \frac{1}{m}\sum_{i=1}^{m} x_i$$

$$\text{median}(x) = \begin{cases} x_{(r+1)} & \text{if } m \text{ is odd, i.e., } m = 2r+1 \\ \frac{1}{2}(x_{(r)} + x_{(r+1)}) & \text{if } m \text{ is even, i.e., } m = 2r \end{cases}$$

Measures of Spread: Range and Variance

- Range is the difference between the max and min
- The variance or standard deviation is the most common measure of the spread of a set of points.

$$\text{variance}(x) = s_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \overline{x})^2$$

- Howeve neasures are often used.

Absolute Average Deviation →

$$\text{AAD}(x) = \frac{1}{m}\sum_{i=1}^{m}|x_i - \overline{x}|$$

Median Absolute Deviation →

$$\text{MAD}(x) = median\left(\{|x_1 - \overline{x}|, \ldots, |x_m - \overline{x}|\}\right)$$

$$\text{interquartile range}(x) = x_{75\%} - x_{25\%}$$

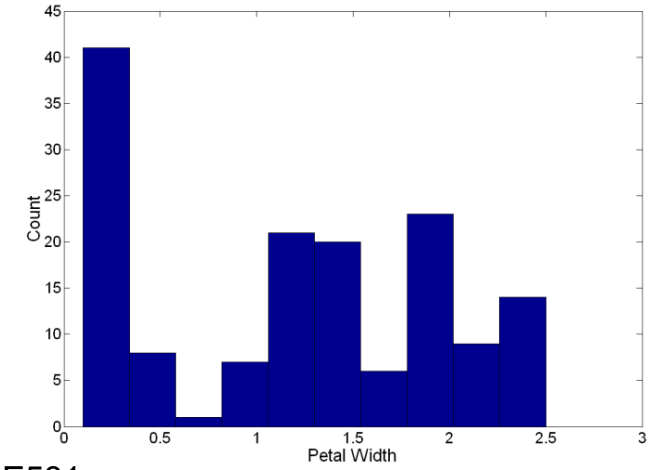
Visualization

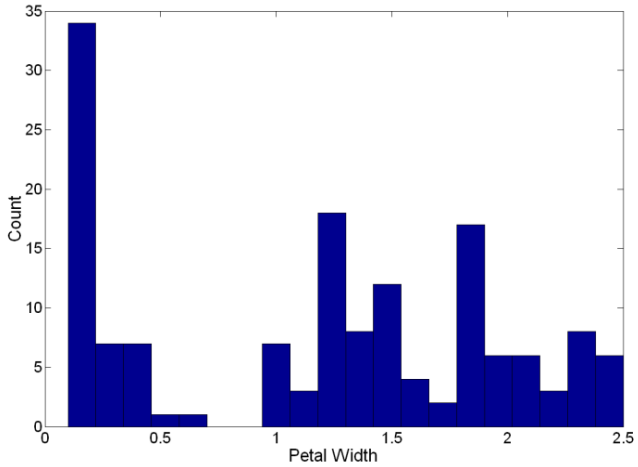
- Another way to understand data is through visualization.
- Visualization of data is one of the most powerful and appealing techniques for data exploration.
  - We are good at analyzing large amounts of information that is presented visually
  - We can detect general patterns and trends in graphs and pictures
  - We can detect outliers and unusual patterns

Histograms

- Usually shows the distribution of values of a single variable
- Divide the values into bins and show a bar plot of the number of objects in each bin.
- The height of each bar indicates the number of objects
- Shape of histogram depends on the number of bins

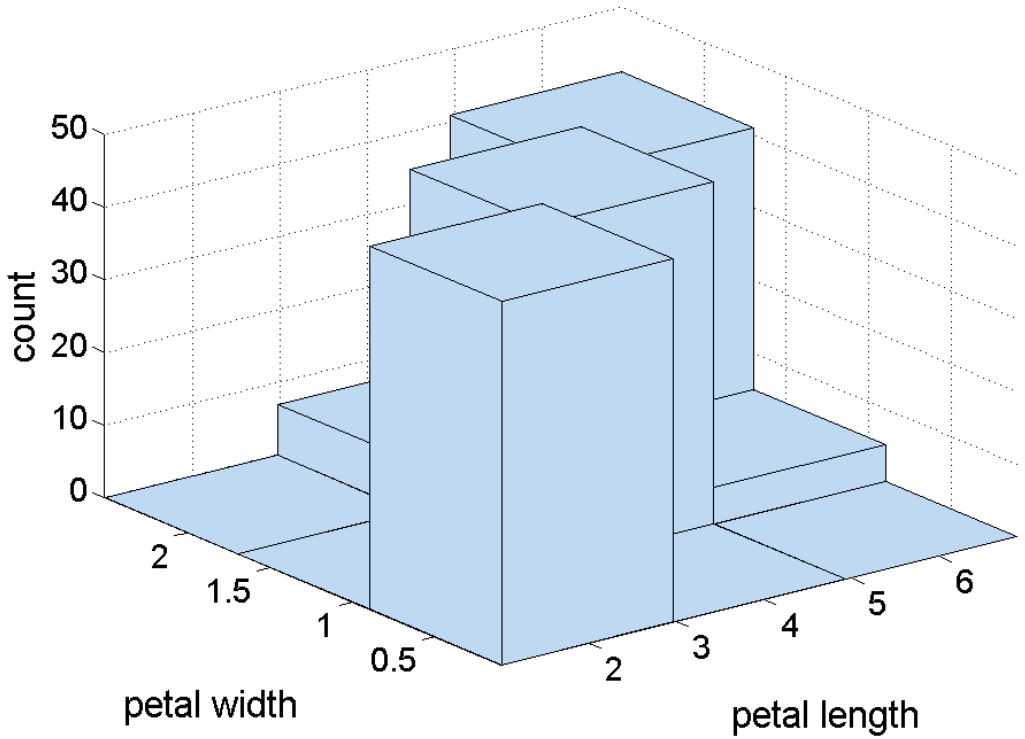
- Example: Petal Width (10 and 20 bins, respectively)



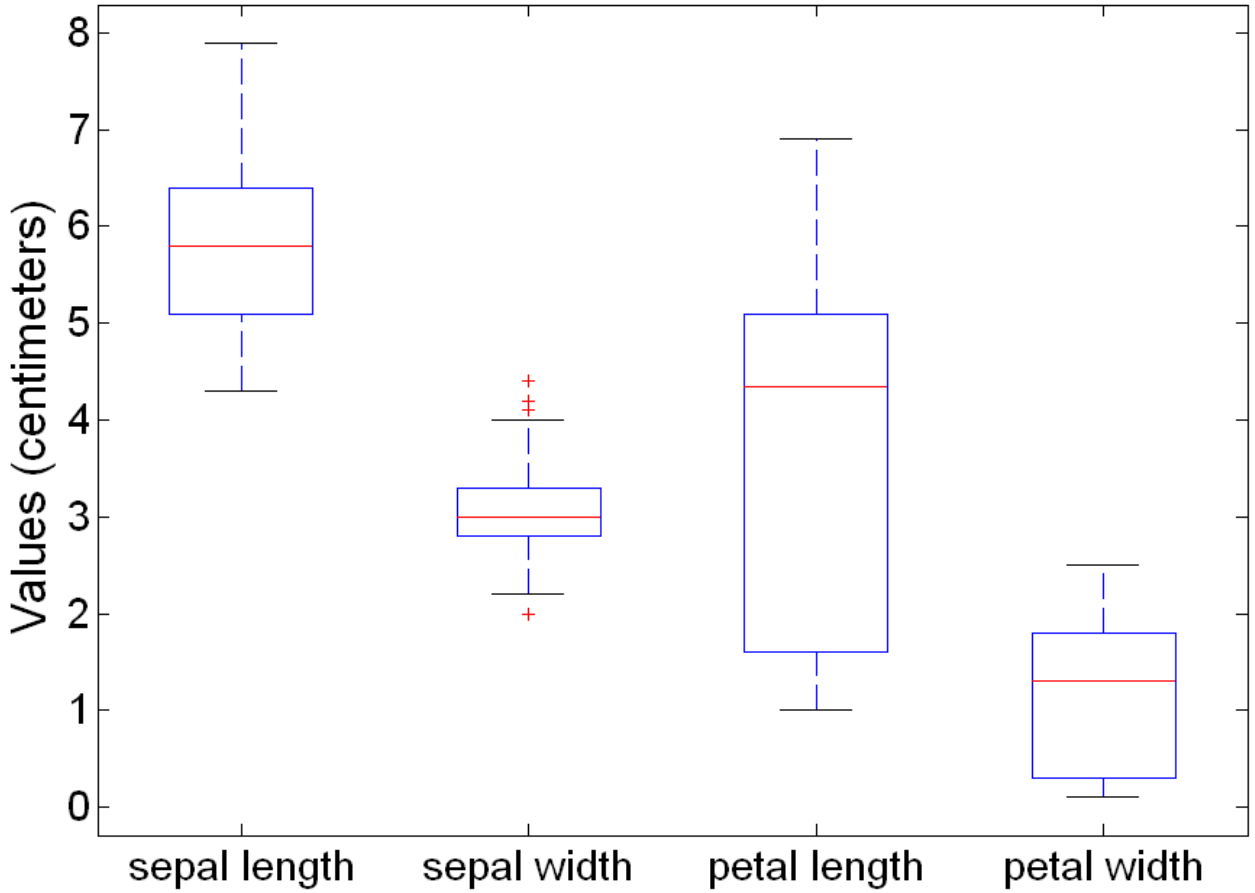


Two-Dimensional Histograms

- Show the joint distribution of the values of two attributes
- Example: petal width and petal length



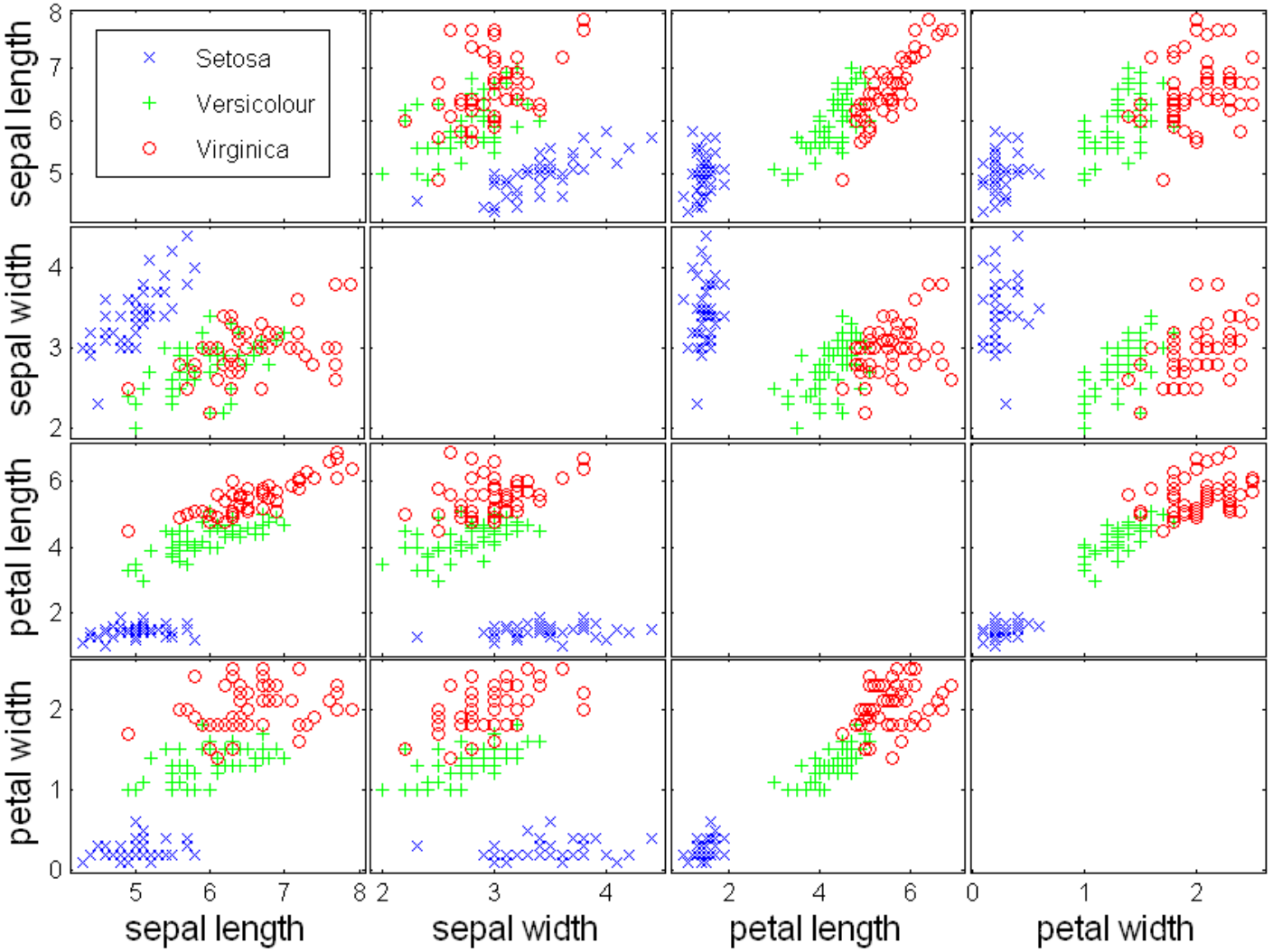
Visualization Through Box Plots



Visualization Through Scatter Plots

- Scatter plots
  - Attributes values determine the position
  - Two-dimensional scatter plots most common, but can have three-dimensional scatter plots
  - Often additional attributes can be displayed by using the size, shape, and color of the markers that represent the objects
  - It is useful to have arrays of scatter plots can compactly summarize the relationships of several pairs of attributes

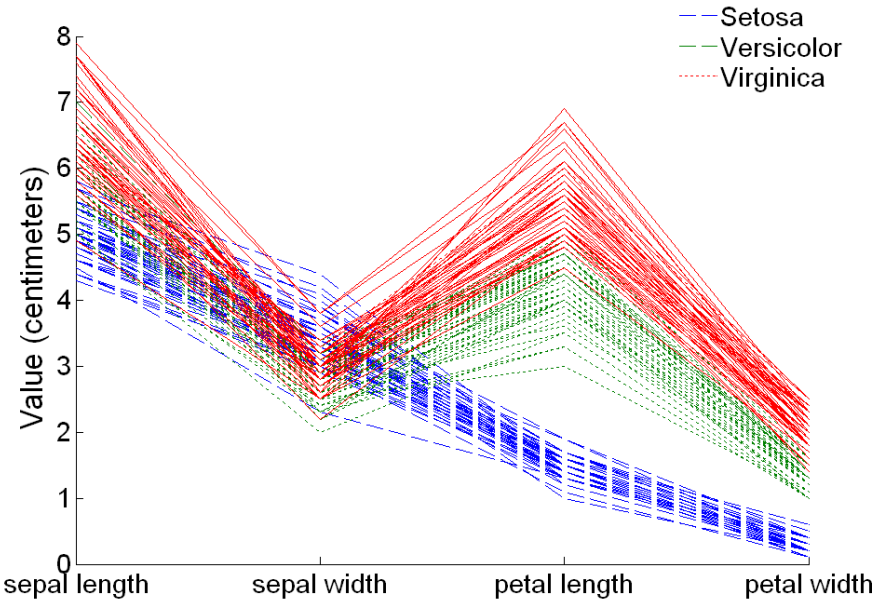
Scatter Plot Array of Iris Attributes

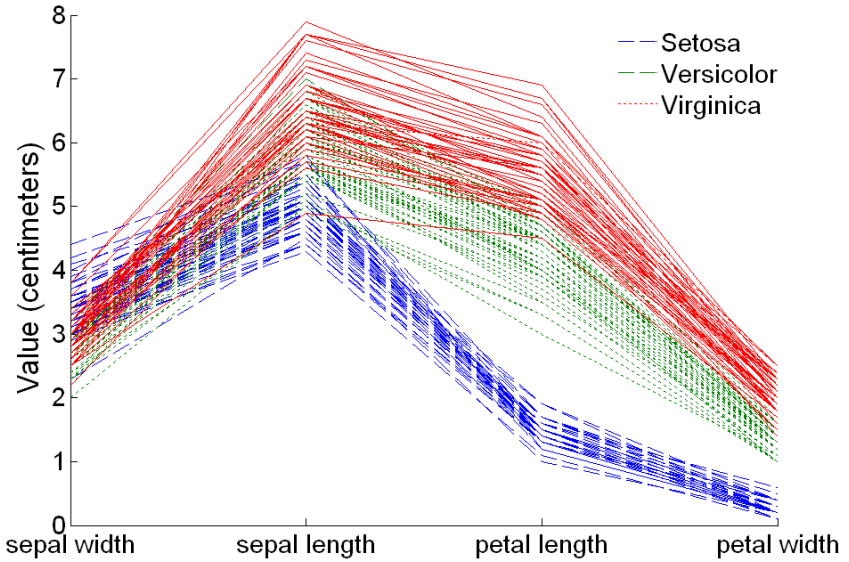


Visualization Through Parallel Coordinates

- Parallel Coordinates
  - Used to plot the attribute values of high-dimensional data
  - Instead of using perpendicular axes, use a set of parallel axes
  - The attribute values of each object are plotted as a point on each corresponding coordinate axis and the points are connected by a line
  - Thus, each object is represented as a line
  - Often, the lines representing a distinct class of objects group together, at least for some attributes
  - Ordering of attributes is important in seeing such groupings

Parallel Coordinates Plots for Iris Data

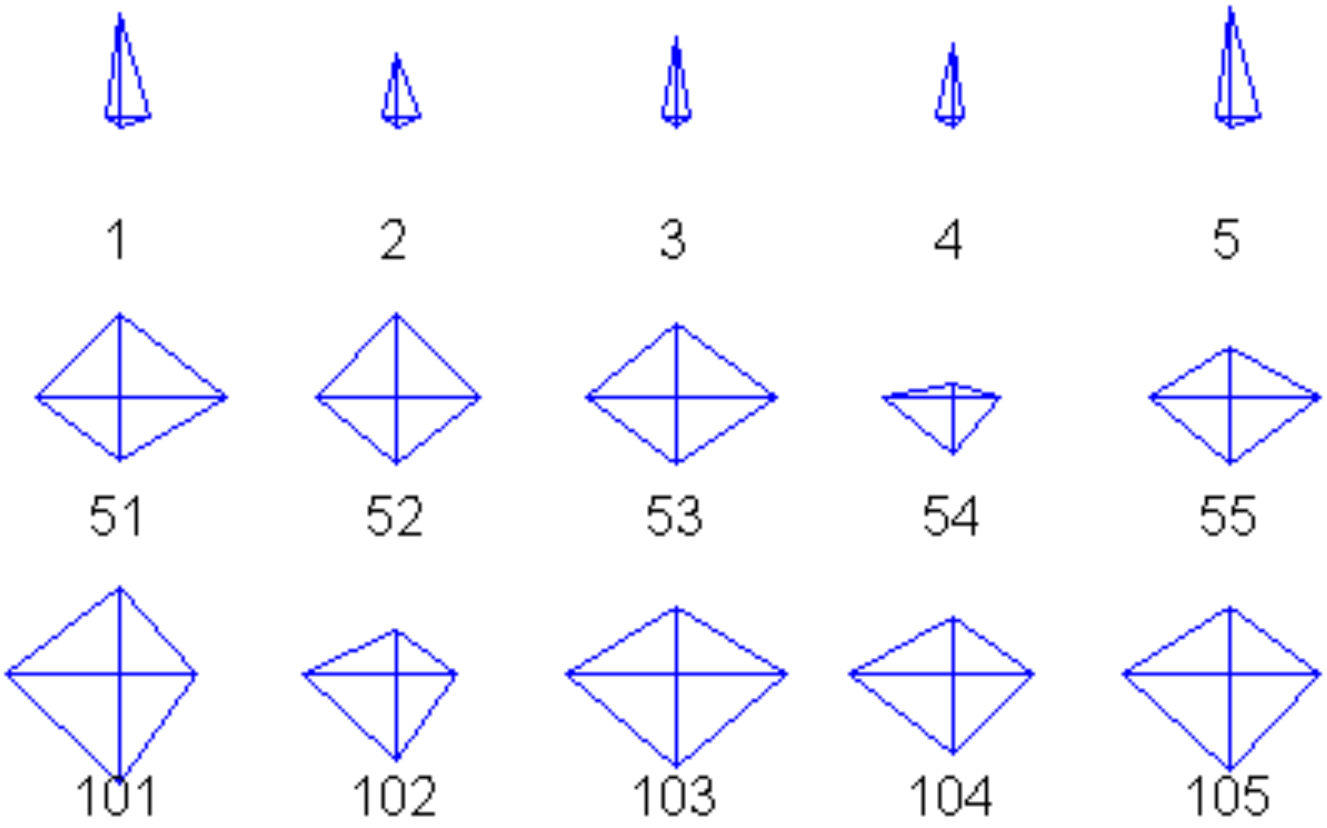




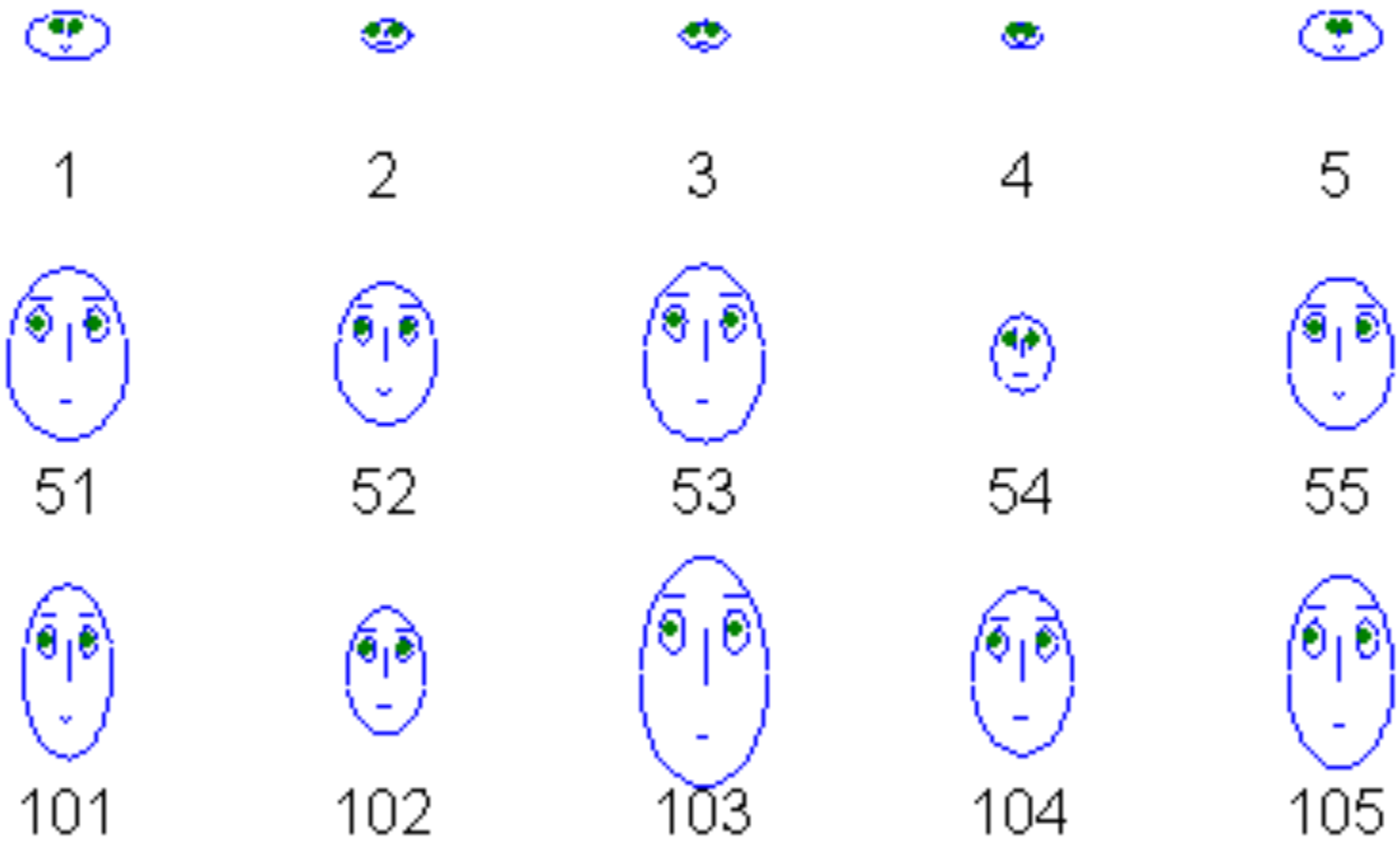
Star Plots and Chernoff Faces

- Star Plots
  - Similar approach to parallel coordinates, but axes radiate from a central point
  - The line connecting the values of an object is a polygon
- Chernoff Faces
  - Approach created by Herman Chernoff
  - This approach associates each attribute with a characteristic of a face
  - The values of each attribute determine the appearance of the corresponding facial characteristic
  - Each object becomes a separate face
  - Relies on human's ability to distinguish faces

Star Plots for Iris Data



Chernoff Faces for Iris Data



Summary

- Make sure to visualize data via histograms and scatter plots to get a feel for its complexity
- Also consider visualization/dimensionality reduction via mapping from high dimensional to a low dimensional space (Next topic)

Data Visualization Example

CSI 5810

Ishwar K Sethi

Boston Housing Data

The Boston housing data was collected in 1978 and each of the 506 entries represent 13 independent features and one dependent feature for homes from various suburbs in Boston, Massachusetts. It is available at UCI Machine Learning Repository. It is also included in sklearn library for machine learning. The features and their meanings are:

- CRIM per capita crime rate by town
- ZN proportion of residential land zoned for lots over 25,000 sq.ft.
- INDUS proportion of non-retail business acres per town
- CHAS Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- NOX nitric oxides concentration (parts per 10 million)
- RM average number of rooms per dwelling
- AGE proportion of owner-occupied units built prior to 1940
- DIS weighted distances to five Boston employment centres
- RAD index of accessibility to radial highways
- TAX full-value property-tax rate per $10,000
- PTRATIO pupil-teacher ratio by town
- B 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
- LSTAT Percentage lower status of the population
- MEDV Median value of owner-occupied homes in $1000's [Output/Target]

```
#import necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style
```

```
#Load data
from sklearn import datasets
boston = datasets.load_boston()
# Lets separate features and the dependent values
X = boston.data
Y = boston.target
print(X.shape,Y.shape)
print(boston.feature_names)
```

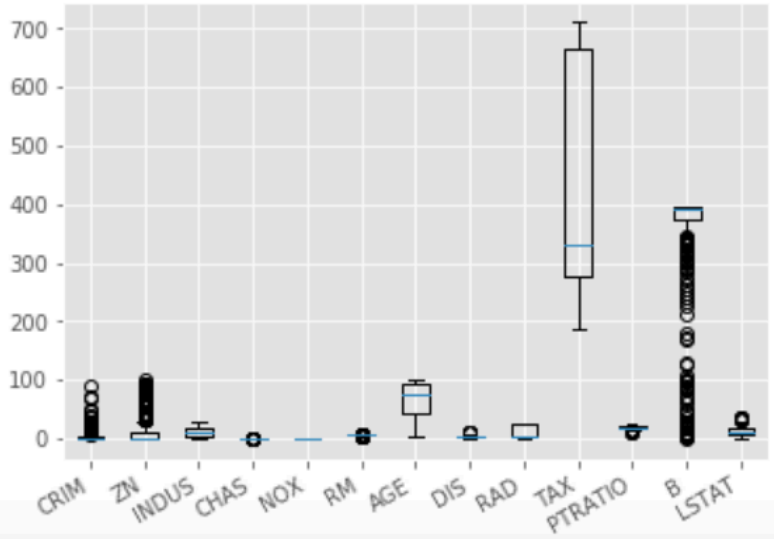
```
(506, 13) (506,)
['CRIM' 'ZN' 'INDUS' 'CHAS' 'NOX' 'RM' 'AGE' 'DIS' 'RAD' 'TAX' 'PTRATIO'
 'B' 'LSTAT']
```

```python
#Lets first look at descriptive statistics of the data.
#Will do it by using pandas because it produces a nicer output
df = pd.DataFrame(X, columns=boston.feature_names)
print(df.describe())
print((pd.DataFrame(Y,columns=['MEDV'])).describe())
```

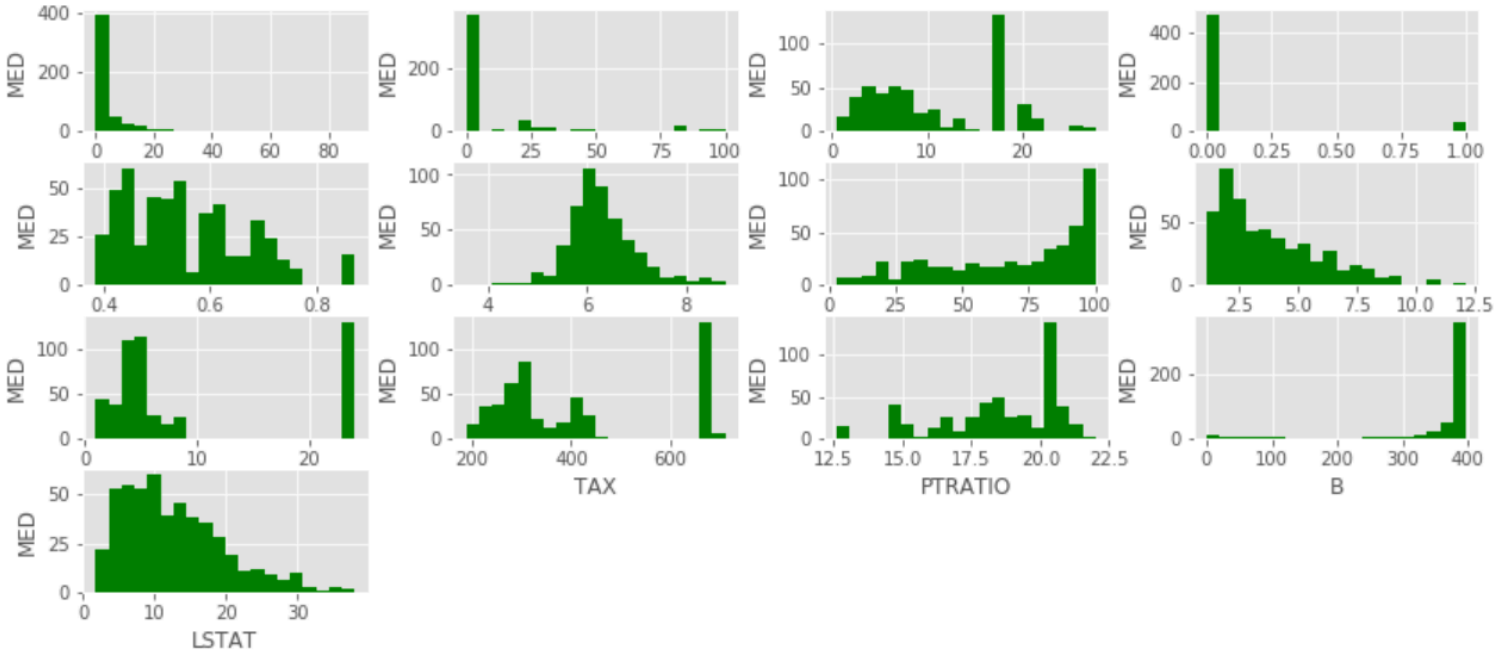
```
              CRIM          ZN       INDUS        CHAS         NOX          RM  \
count  506.000000  506.000000  506.000000  506.000000  506.000000  506.000000
mean     3.593761   11.363636   11.136779    0.069170    0.554695    6.284634
std      8.596783   23.322453    6.860353    0.253994    0.115878    0.702617
min      0.006320    0.000000    0.460000    0.000000    0.385000    3.561000
25%      0.082045    0.000000    5.190000    0.000000    0.449000    5.885500
50%      0.256510    0.000000    9.690000    0.000000    0.538000    6.208500
75%      3.647423   12.500000   18.100000    0.000000    0.624000    6.623500
max     88.976200  100.000000   27.740000    1.000000    0.871000    8.780000

              AGE         DIS         RAD         TAX     PTRATIO           B  \
count  506.000000  506.000000  506.000000  506.000000  506.000000  506.000000
mean    68.574901    3.795043    9.549407  408.237154   18.455534  356.674032
std     28.148861    2.105710    8.707259  168.537116    2.164946   91.294864
min      2.900000    1.129600    1.000000  187.000000   12.600000    0.320000
25%     45.025000    2.100175    4.000000  279.000000   17.400000  375.377500
50%     77.500000    3.207450    5.000000  330.000000   19.050000  391.440000
75%     94.075000    5.188425   24.000000  666.000000   20.200000  396.225000
max    100.000000   12.126500   24.000000  711.000000   22.000000  396.900000

            LSTAT
count  506.000000
mean    12.653063
std      7.141062
min      1.730000
25%      6.950000
50%     11.360000
75%     16.955000
max     37.970000
```

```
             MEDV
count  506.000000
mean    22.532806
std      9.197104
min      5.000000
25%     17.025000
50%     21.200000
75%     25.000000
max     50.000000
```

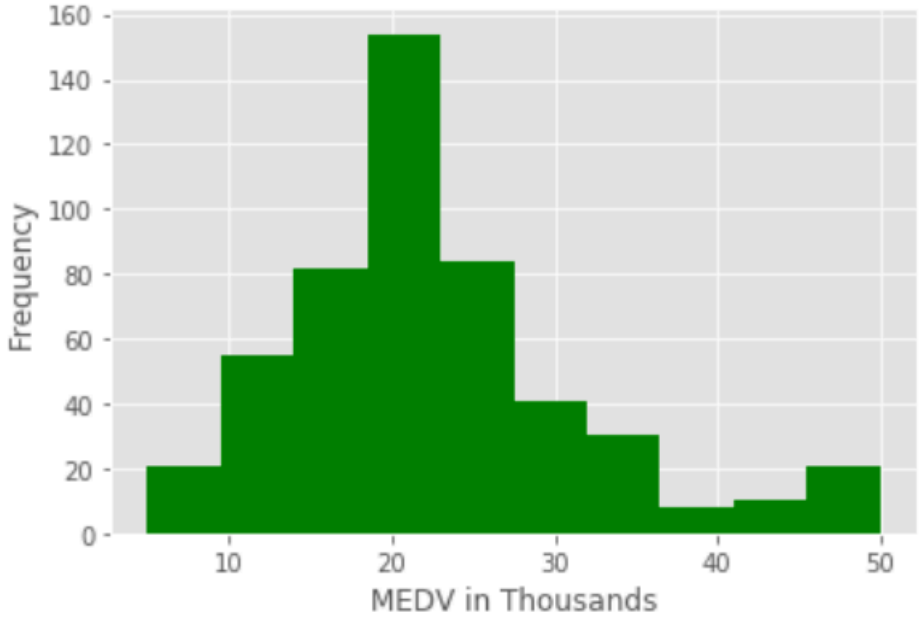
```
#Lets do boxplot of the features
plt.boxplot(X)
plt.xticks(np.arange(1, X.shape[1]+1),headings,rotation=30,ha="right")
plt.show()
```



```
# Lets look at each of the 13 independent features
plt.style.use('ggplot')
fig, axs = plt.subplots(4, 4, figsize=(14,6))
headings = boston.feature_names
for i in range(1,14):
    plt.subplot(4,4,i)
    plt.hist(X[:,i-1], bins=20, color='g')
    plt.xlabel(headings[i-1])
    plt.ylabel('MED')
plt.subplots_adjust(wspace=0.30, hspace=0.25)
fig.delaxes(axs[3][1])
fig.delaxes(axs[3][2])
fig.delaxes(axs[3][3])
plt.show()
```



```
# Lets look at the price distribution (MED) also
plt.hist(Y, color='g')
plt.title('MEDV Histogram')
plt.xlabel('MEDV in Thousands')
plt.ylabel('Frequency')
plt.show()
```



```
# Lets look at now the relationship between the target value and each independent feature
fig, axs = plt.subplots(3, 5, figsize=(16,6))
for i in range(1,14):
    plt.subplot(3,5,i)
    plt.scatter(X[:,i-1],Y,marker ='.', color='g')
    plt.xlabel(headings[i-1])
    plt.ylabel('MED')
plt.subplots_adjust(wspace=0.25, hspace=0.50)
fig.delaxes(axs[2][3])
fig.delaxes(axs[2][4])
plt.show()
```