
Lecture Notes for Machine Learning in Python

Professor Eric Larson
Visualization

Class Logistics and Agenda

- Participation for Distance
- Look at **Lab One**! Due at end of week!
- Dataset Selection Questions?
- Agenda
 - Pandas Demo with Imputation
 - Data Exploration
 - Data Preprocessing
 - Data Visualization

What did we talk about last time?

Start Pandas demo

DataFrames

Loading

Indexing

Imputing



03.Data Visualization.ipynb



Data Exploration

What is data exploration?

A preliminary exploration of the data to better understand its characteristics.

- Help **select** the **right tool** for preprocessing or analysis
- Exploratory Data Analysis, EDA by Dr. John Tukey:
 - The focus was visualization
 - Clustering and anomaly detection were viewed as exploratory techniques
- In our discussion,
 - Summary statistics, aggregations
 - Visualizing summaries



Summary Statistics

- frequency, location, and spread
 - Examples: location by **mean**
spread by **standard deviation**
- Most summary statistics can be calculated in a single pass through the data

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

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

- For nominal data, mode or frequency is most common

Measures of Spread

- **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.

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

- However, this is also sensitive to outliers, so that other measures are often used.

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

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

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

Higher order statistics

- A comparison of the tails of a distribution

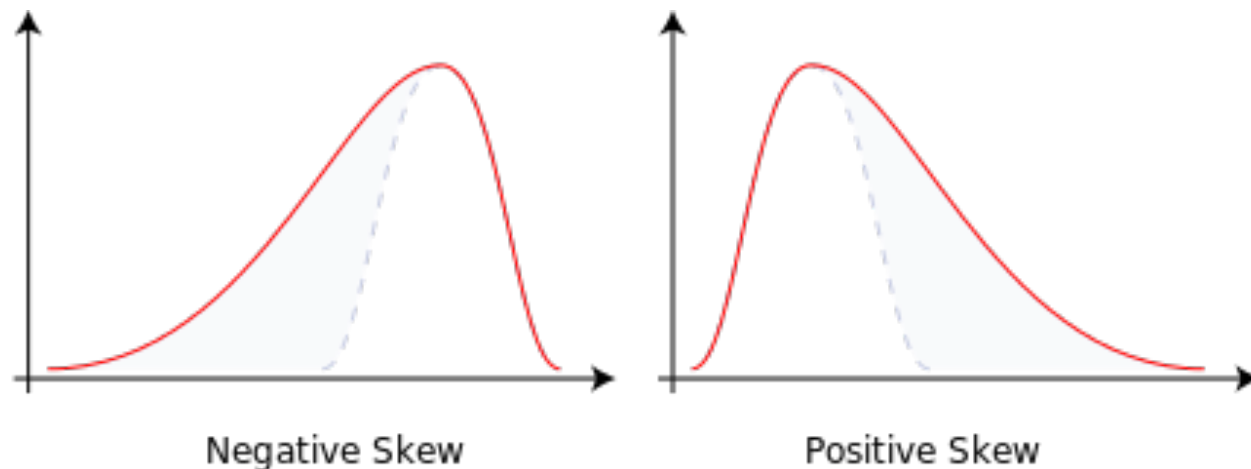


image: wikipedia

$$skewness(x) = \frac{1}{N} \sum_i \left(\frac{x_i - \bar{x}}{\sigma} \right)^3$$

$$kurtosis(x) = \frac{1}{N} \sum_i \left(\frac{x_i - \bar{x}}{\sigma} \right)^4$$

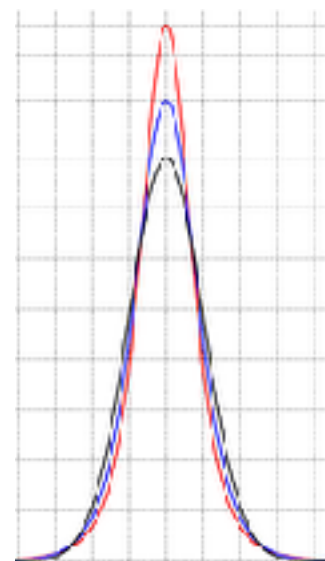
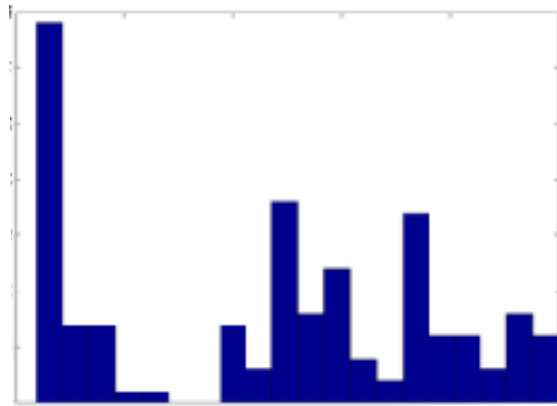


image: wikipedia

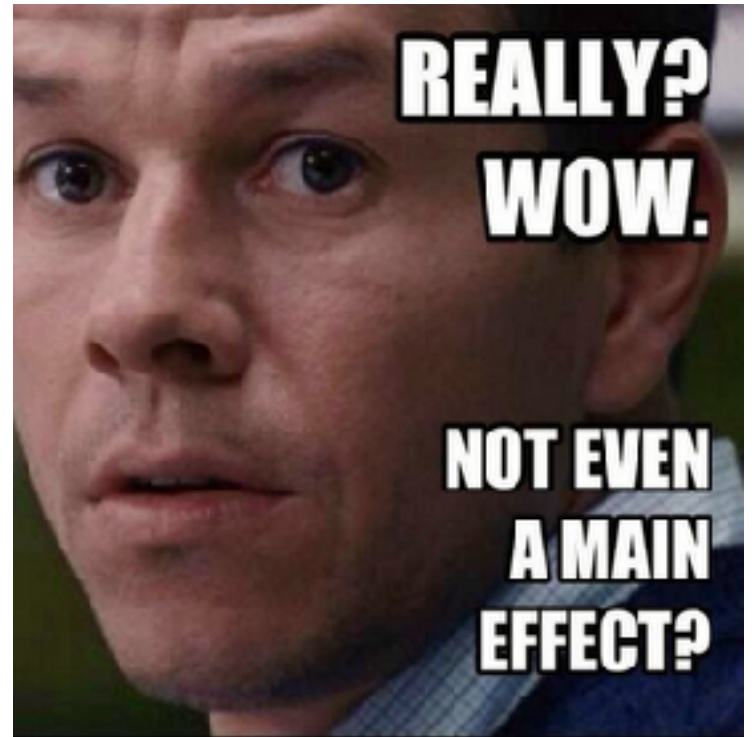
Self Test 2a.1

What measure of spread is most appropriate for the data in the histogram below?



- A) Standard Deviation
- B) Interquartile Range
- C) Median Absolute Difference
- D) None of these

Data Preprocessing



Data Preprocessing

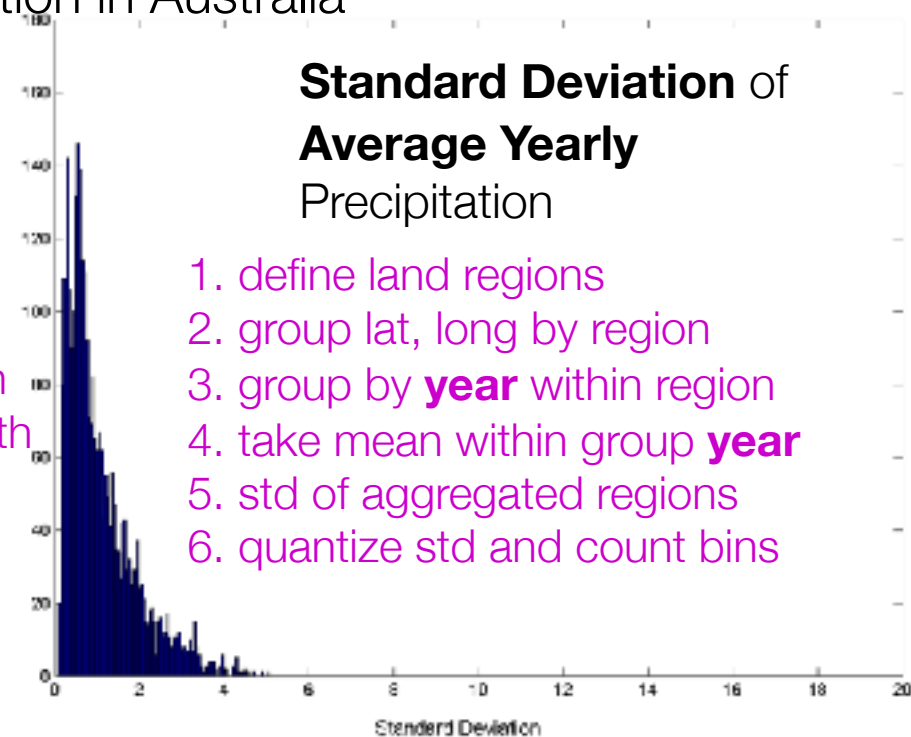
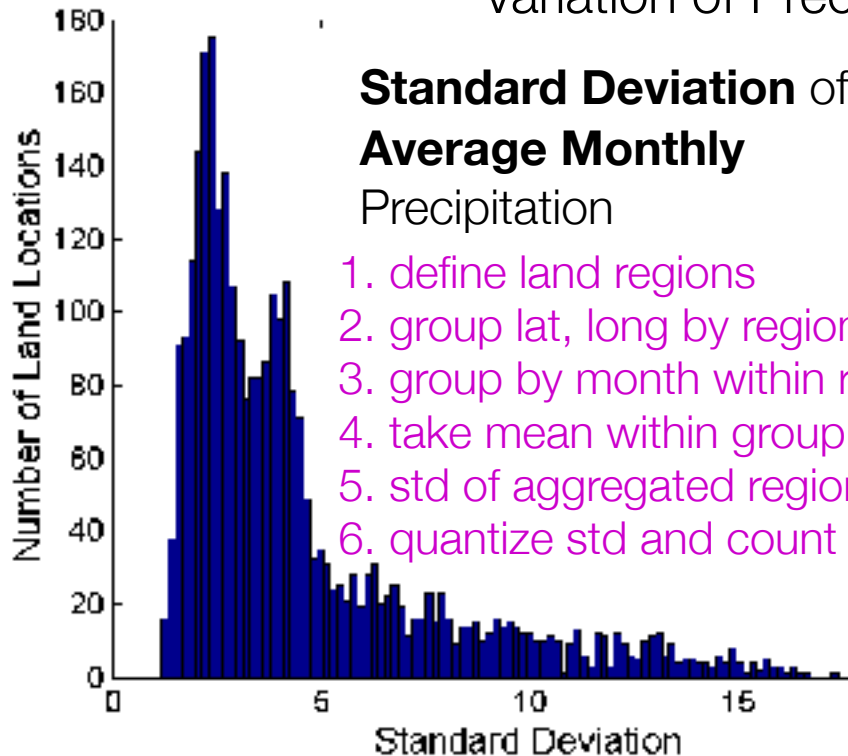
- Aggregation
- Quantization: Making Discrete or Binary
- Attribute Transformation
- *Dimensionality Reduction*
 - *PCA and LDA (look at separately, next time)*

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
 - More “stable” data
 - ◆ Aggregated data tends to have less variability

Aggregation

Variation of Precipitation in Australia

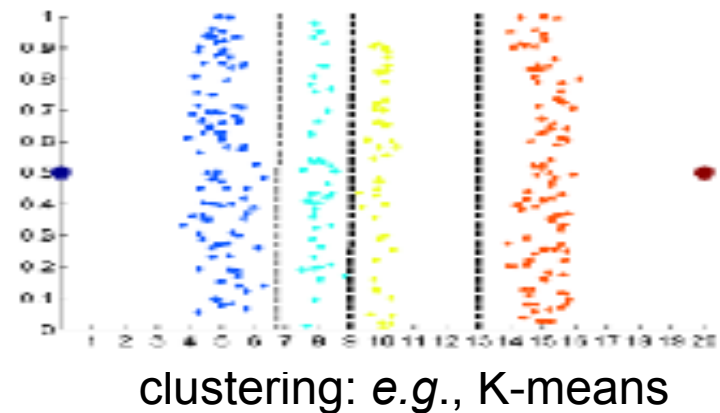
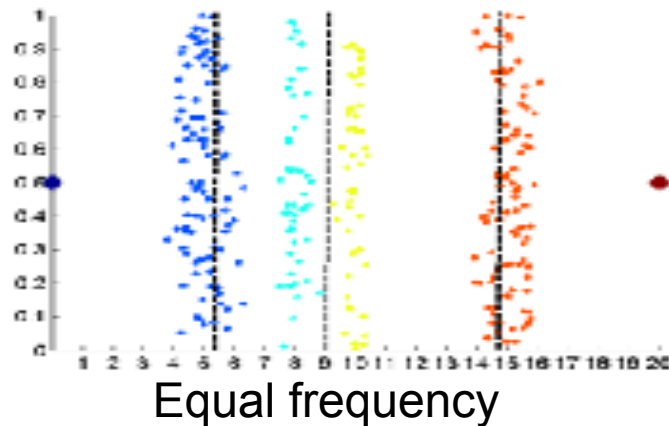
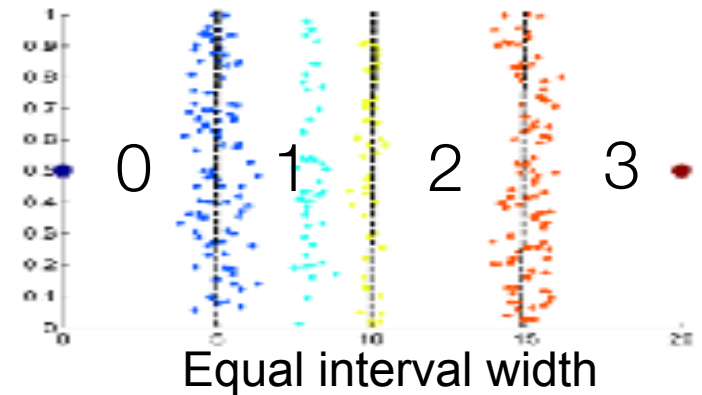
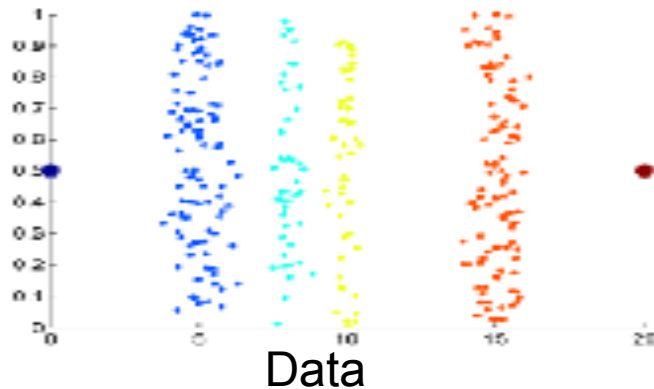


How has aggregation has been used to create these plots?

<i>TID</i>	<i>Location</i>	<i>time</i>	<i>measured rainfall</i>
<i>1</i>	<i>lat, long</i>	<i>measured daily</i>	<i>X.XX cm</i>

Feature quantization: make ordinal

```
pandas.cut(dataframe.var, [5,10,15])
```



```
num_quantiles = 4
```

```
pandas.qcut(dataframe.var, num_quantiles)
```

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 monotonic functions: x^k , $\log(x)$, e^x , $|x|$
 - Standardization and Normalization
 - min/max, z-scores
 - Polynomial and Interaction Variables
 - $x[:,1]**2$
 - $x[:,1]*x[:,2]$

Attribute Transformation in Python

```
>>> from sklearn import preprocessing
>>> import numpy as np
>>> X = np.array([[ 1., -1.,  2.],
...               [ 2.,  0.,  0.],
...               [ 0.,  1., -1.]])
>>> X_scaled = preprocessing.scale(X)
>>> X_scaled
array([[ 0. ..., -1.22...,  1.33...],
       [ 1.22...,  0. ..., -0.26...],
       [-1.22...,  1.22..., -1.06...]])
```

```
>>> scaler = preprocessing.StandardScaler().fit(X)
>>> scaler
StandardScaler(copy=True, with_mean=True, with_std=True)
```

```
>>> scaler.mean_
array([ 1. ...,  0. ...,  0.33...])
```

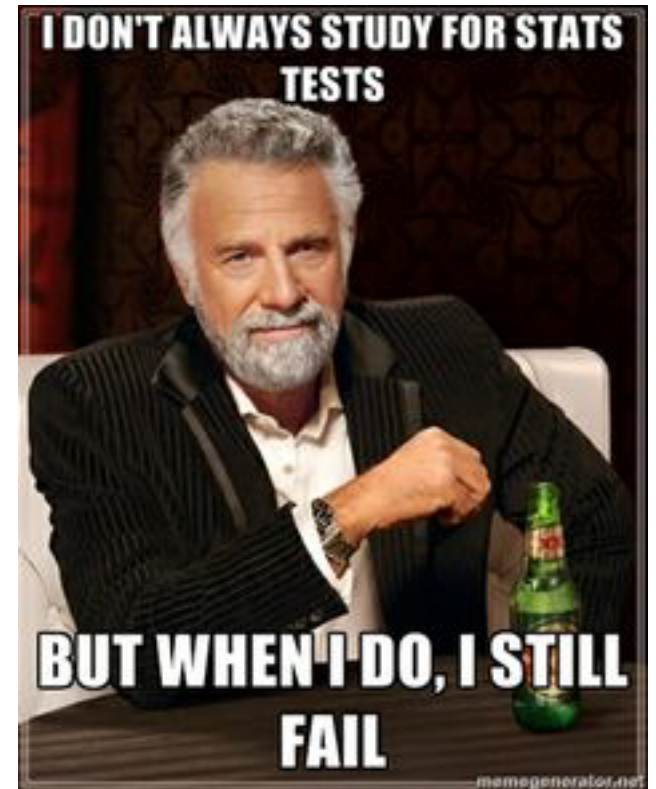
```
>>> scaler.std_
array([ 0.81...,  0.81...,  1.24...])
```

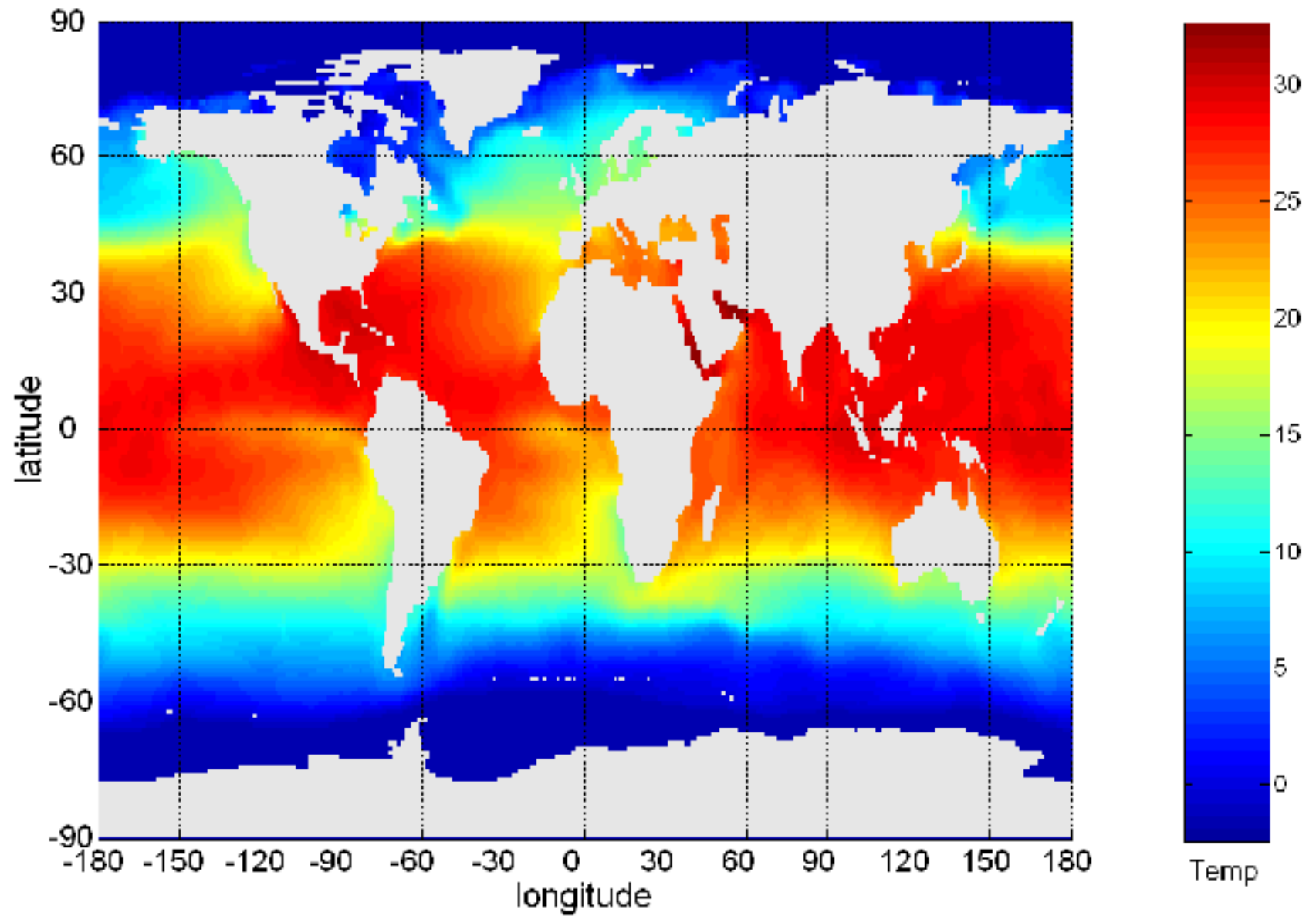
```
>>> scaler.transform(X)
array([[ 0. ..., -1.22...,  1.33...],
       [ 1.22...,  0. ..., -0.26...],
       [-1.22...,  1.22..., -1.06...]])
```

Standardization and Normalization

```
>>> import pandas
>>> df_normalized = (df-df.mean())/(df.std())
```

Data Visualization





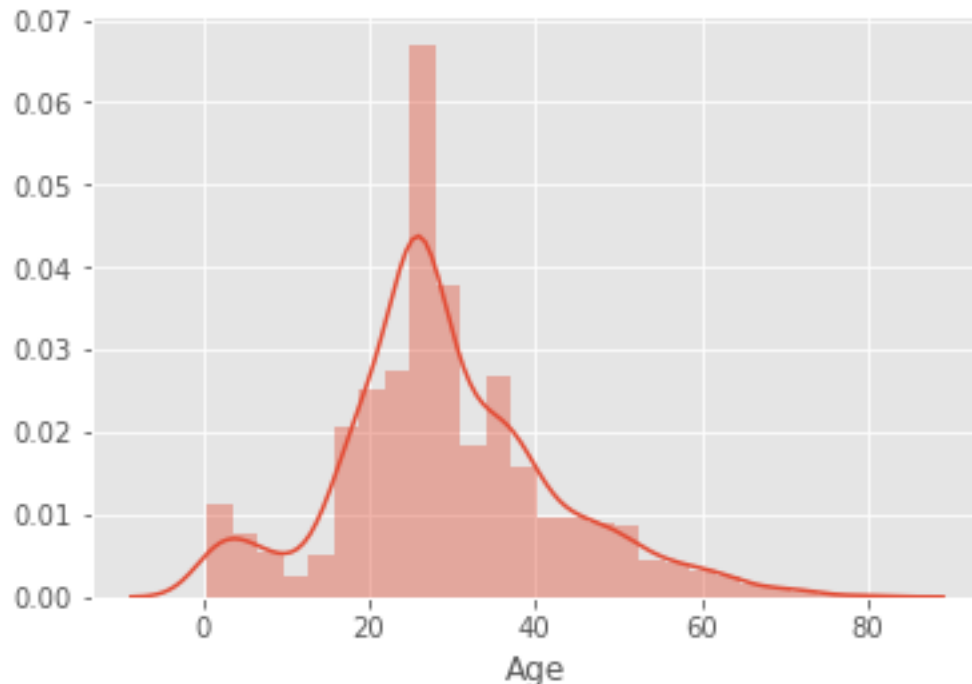
Let's look some graphs

WAKE UP (*please*)

- You tell me what conclusions we are getting from these graphs

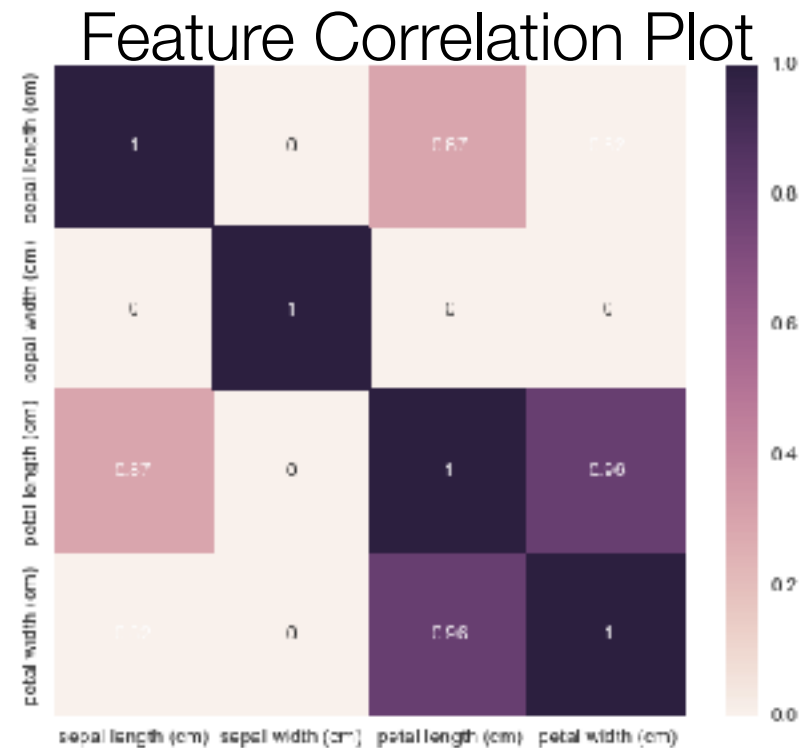
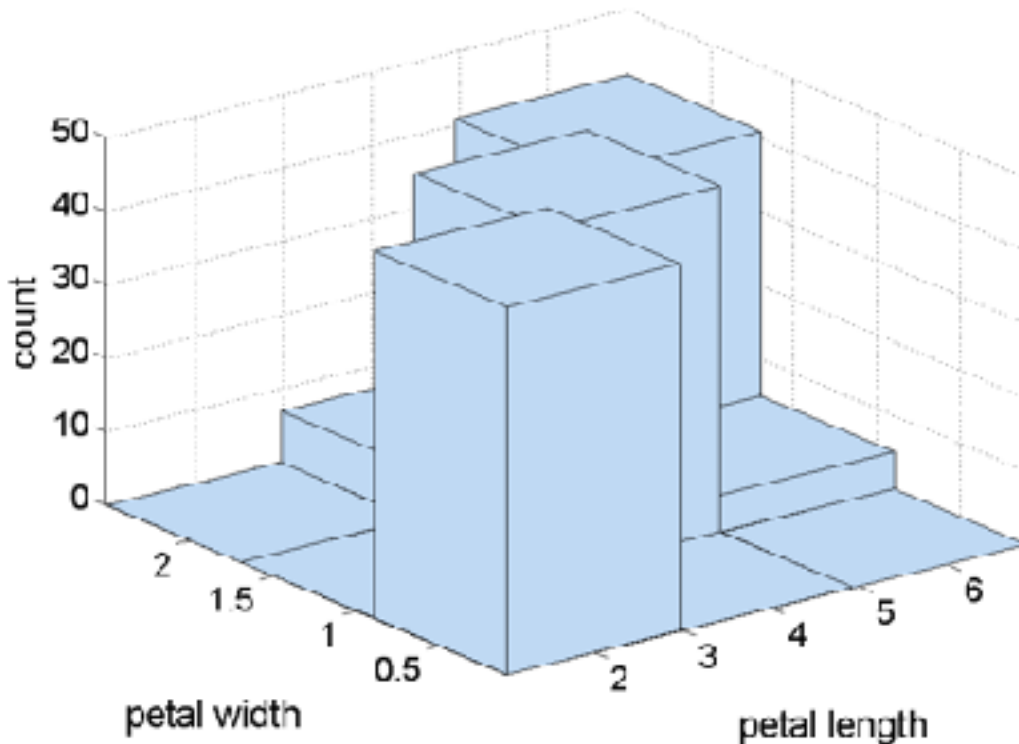
Visualization Techniques: Distributions

- Histogram
 - 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.
- KDE
 - Add up Gaussian underneath each point value
 - STD of gaussian is equivalent to number of bins in histogram



Two-Dimensional Distributions

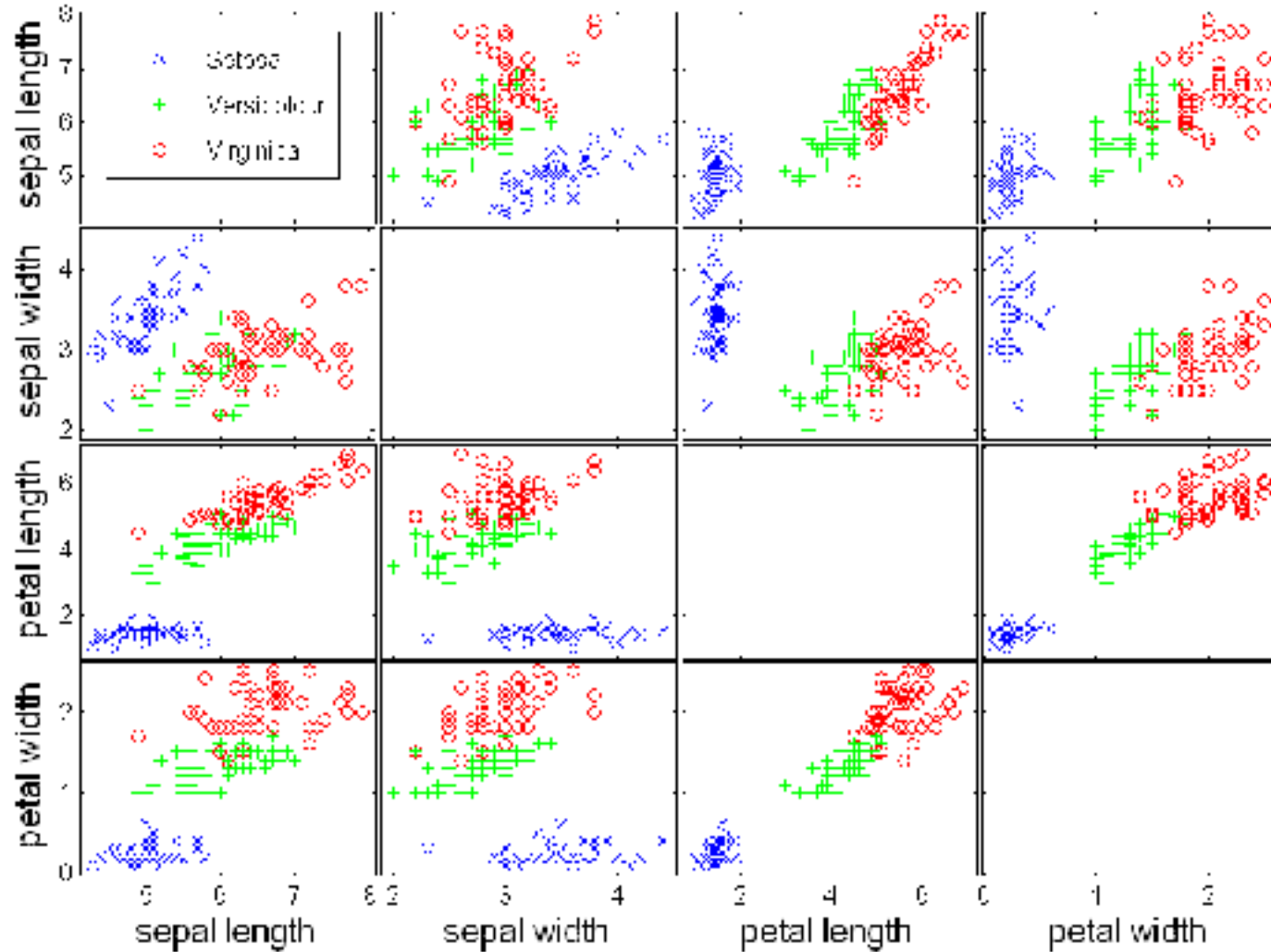
- Estimate the joint distribution of the values of two attributes
- Example: petal width and petal length
 - What does this tell us?



Visualization Techniques: Scatter Plots

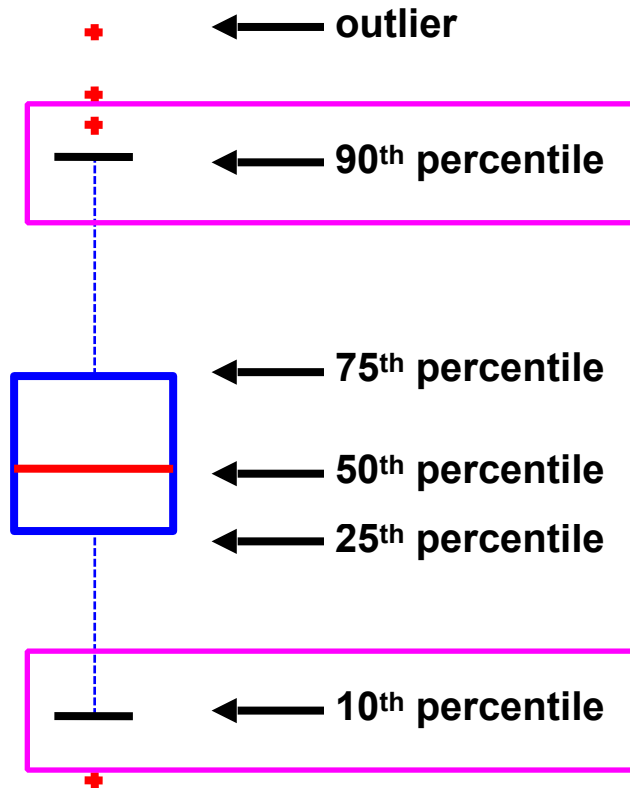
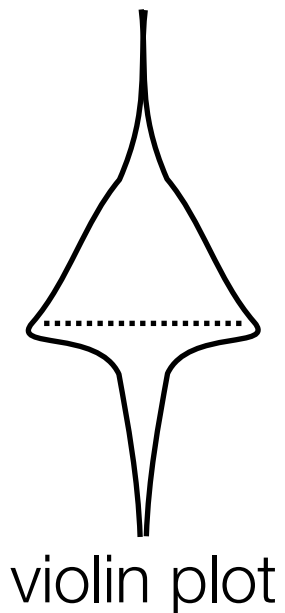
- Scatter plots
 - Two-dimensional scatter plots most common
 - **Additional attributes** can be displayed by using the **size, shape, and color** of the markers that represent the objects
 - **Interactivity** can add **insight**
 - It is useful to have **matrices of scatter plots** to compactly summarize the relationships of several pairs of attributes
 - Good for numeric data, but needs **jitter** for categorical data

Scatter Plot Matrix Colored by Class



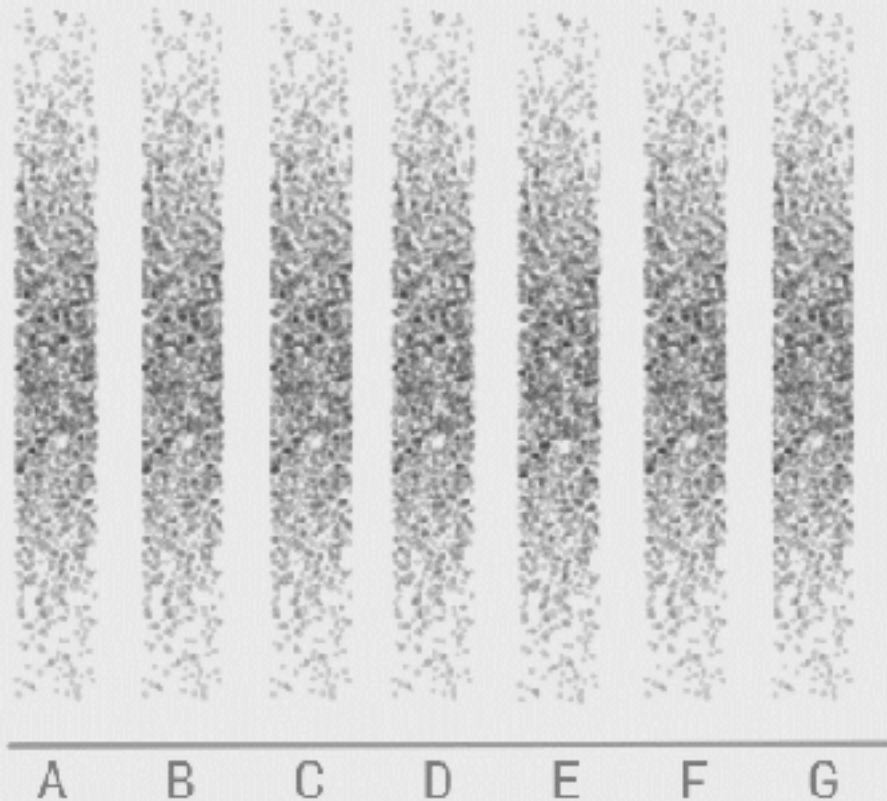
Visualization Techniques: Box Plots

- Box Plots
 - Invented by J. Tukey
 - Another way of displaying the distribution of data
 - Following figure shows the basic part of a box plot

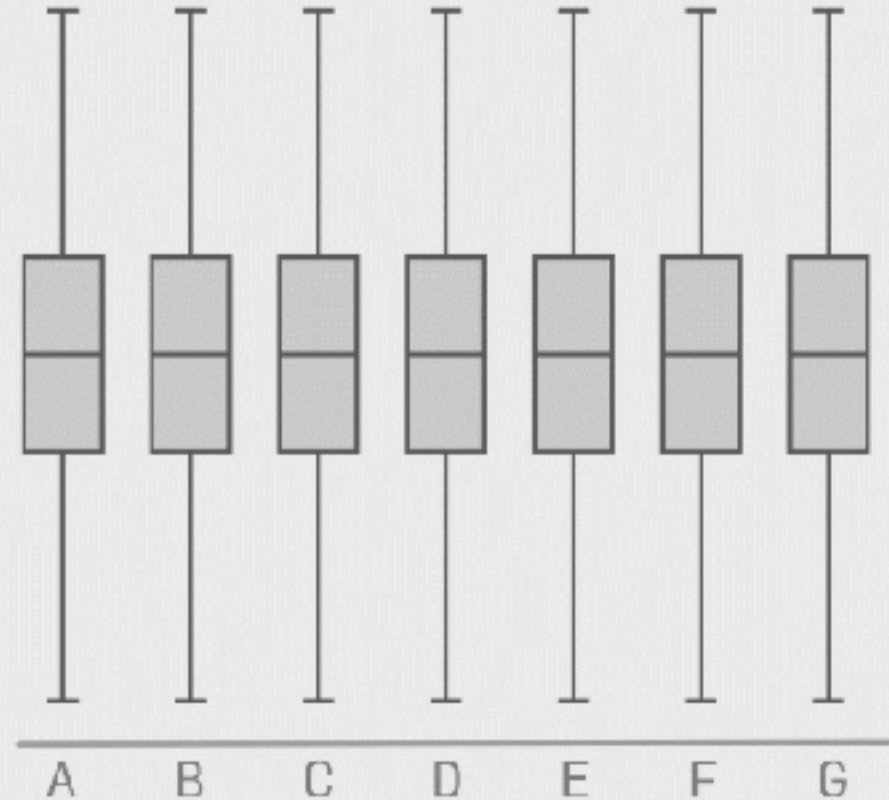


Visualization Techniques: Box Plots

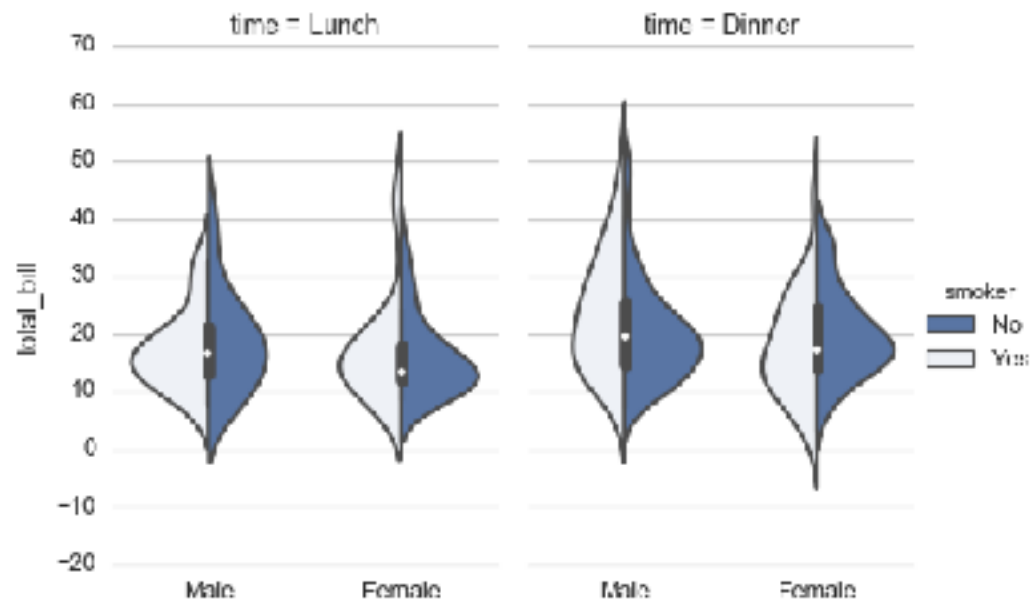
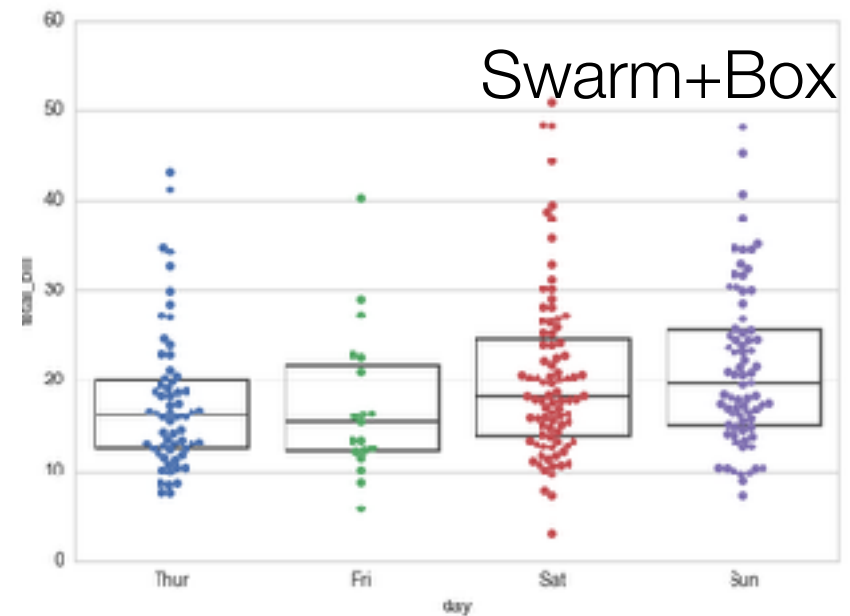
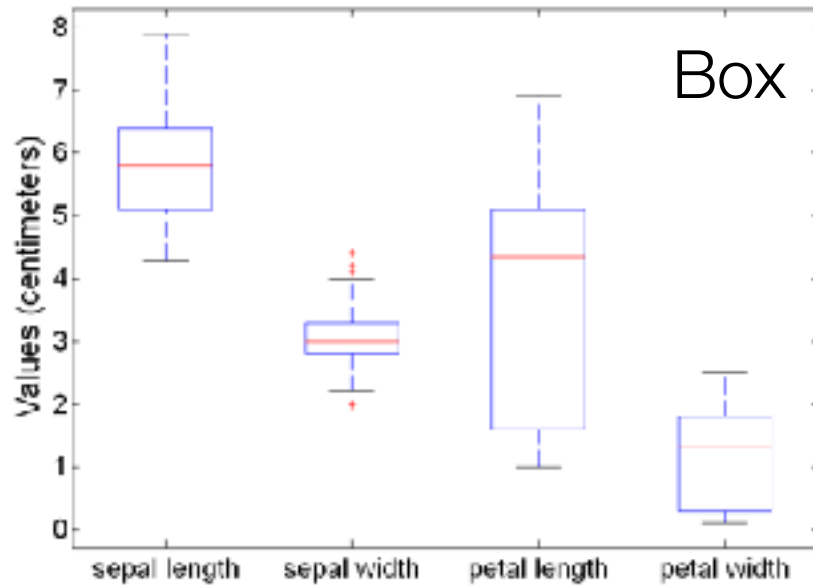
Raw Data



Box-plot of the Data



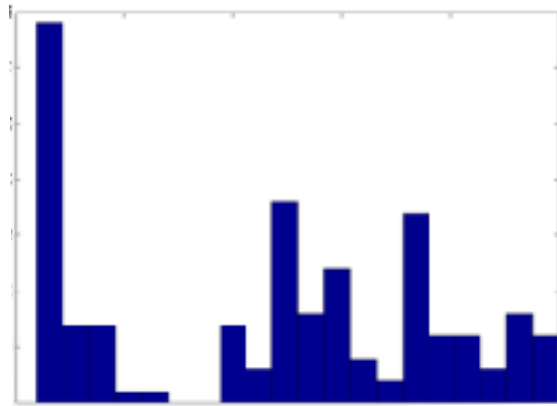
Example: Comparing Attributes



Mixed Violin + Box

Self Test 2a.2

What compact visualization is most appropriate for the data in the histogram below?

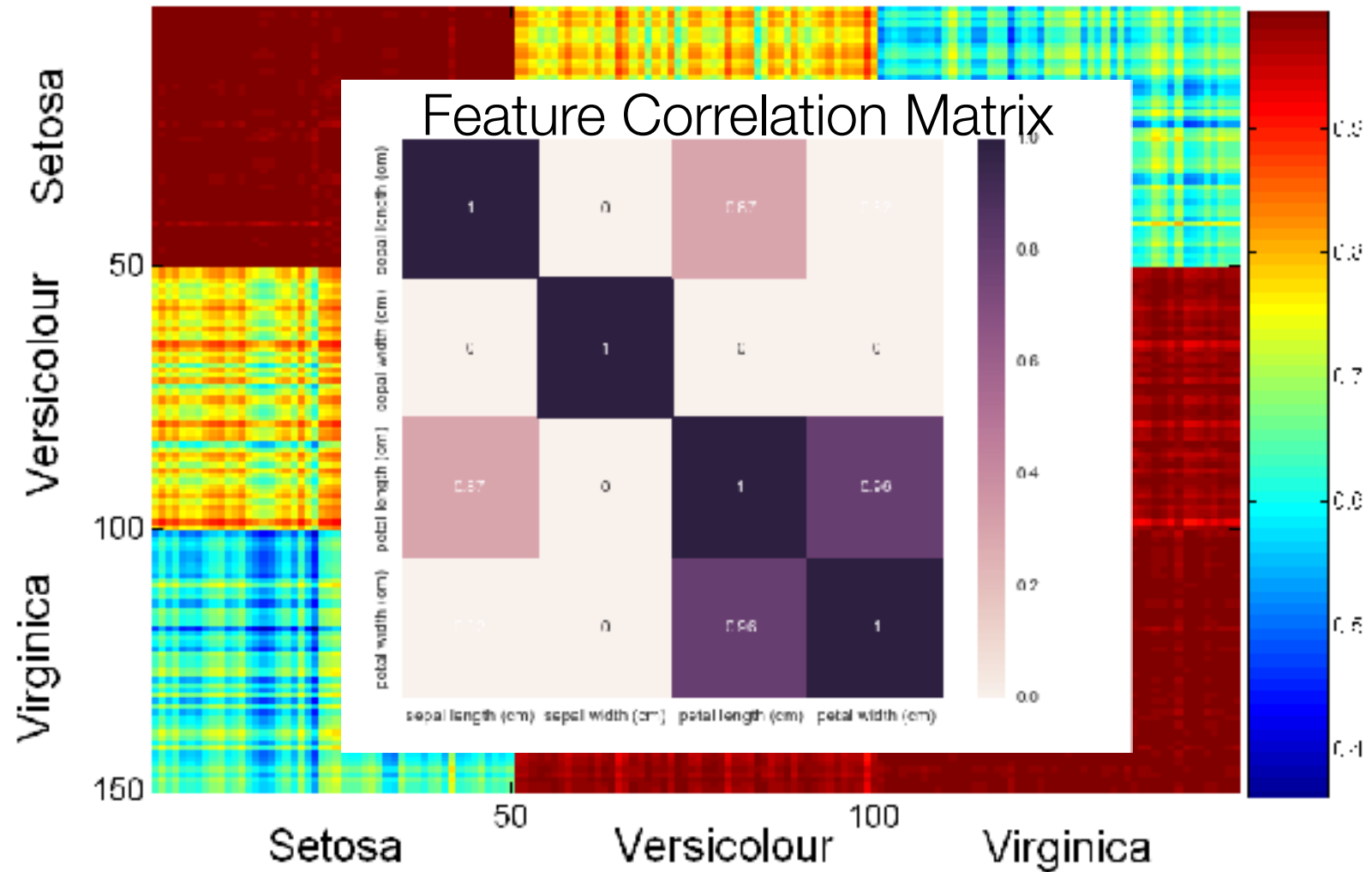


- A) Box Plot
- B) Violin Plot
- C) Swarm Plot
- D) None of these

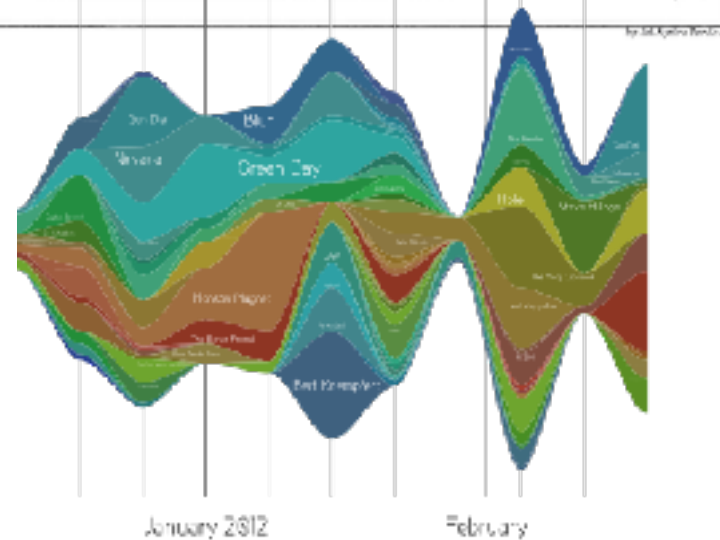
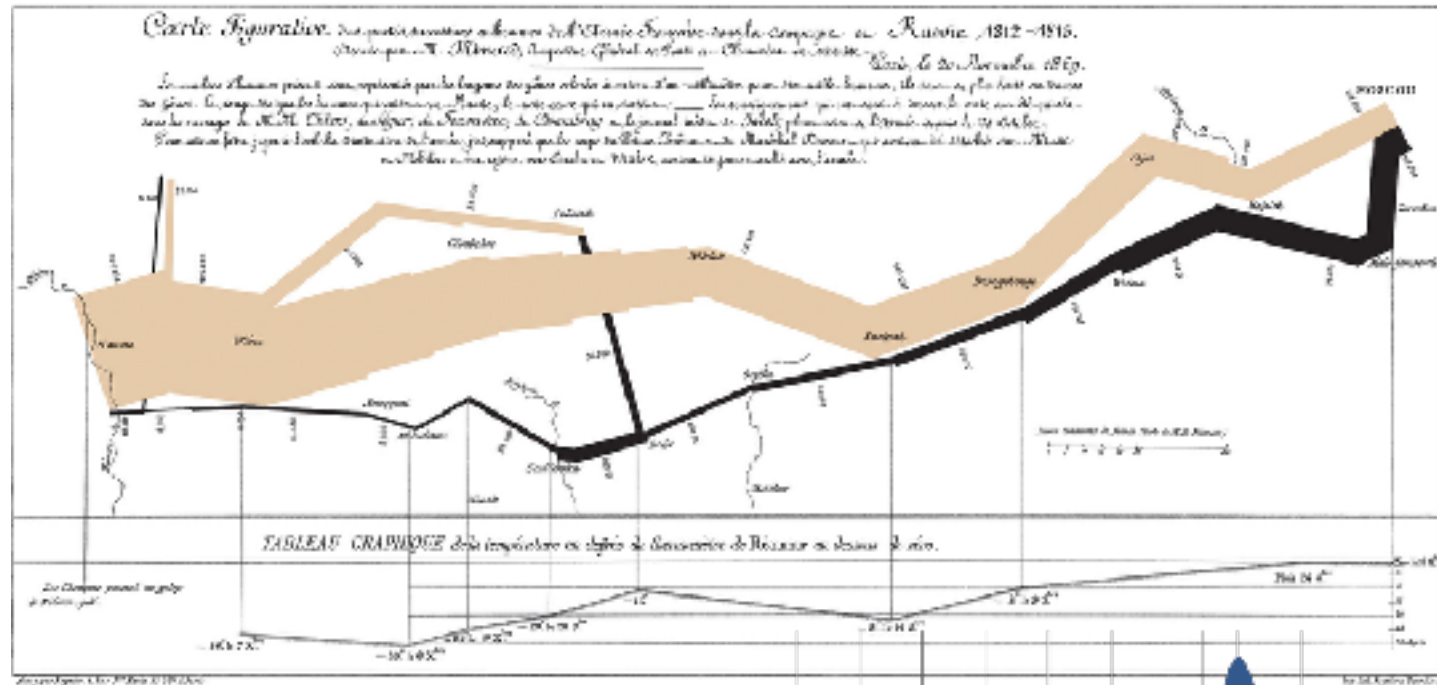
Visualization Techniques: Matrix Plots

- Matrix plots (typically heatmaps)
 - Plot some data matrix
 - This can be useful when objects are sorted well
 - Typically, the attributes are normalized to prevent one attribute from dominating the plot
 - Plots of similarity or distance matrices can also be useful for visualizing the relationships between objects

Instance Correlation Matrix



There are lots of plots out there!



Matplotlib

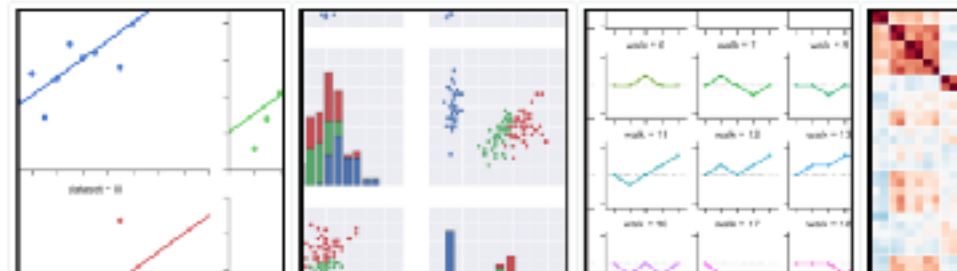
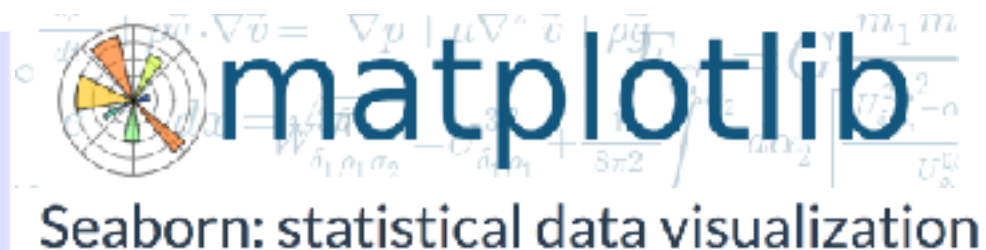
- Python plotting utility
 - Has **low level plotting** functionality
 - Highly **similar to Matlab and R** for plotting
- Extended for visually be more beautiful by
 - **seaborn**: stanford data visualization group

John Hunter (1968-2012)



On August 28 2012, John D. Hunter, the creator of matplotlib, died from complications arising from cancer treatment, after a brief but intense battle with this terrible illness. John is survived by his wife Miriam, his three daughters Rahel, Ava and Clara, his sisters Layne and Mary, and his mother Sarah.

If you have benefited from John's many contributions, please say thanks in the way that would matter most to him. Please consider making a donation to the [John Hunter Memorial Fund](#).



What did we talk about last time?

Visualization

Matplotlib

Seaborn

Plotly

03.Data Visualization.ipynb

Other Tutorials:

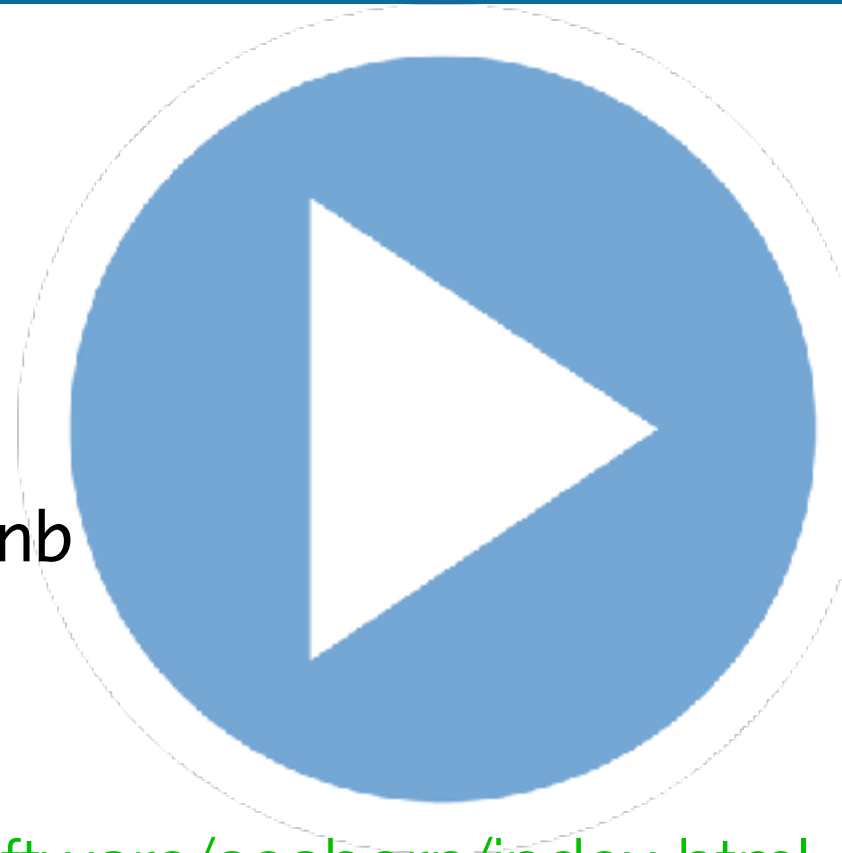
<https://t.co/zNzD8Q8w5E>

<http://stanford.edu/~mwaskom/software/seaborn/index.html>

<http://pandas.pydata.org/pandas-docs/stable/visualization.html>

<http://matplotlib.org/examples/index.html>

http://nbviewer.ipynb.org/github/mwaskom/seaborn/blob/master/examples/plotting_distributions.ipynb



For Next Lecture

- Next Time: Visualization Demo and Data Dimensionality Reduction
- Look at chapter 5 of Python Machine Learning

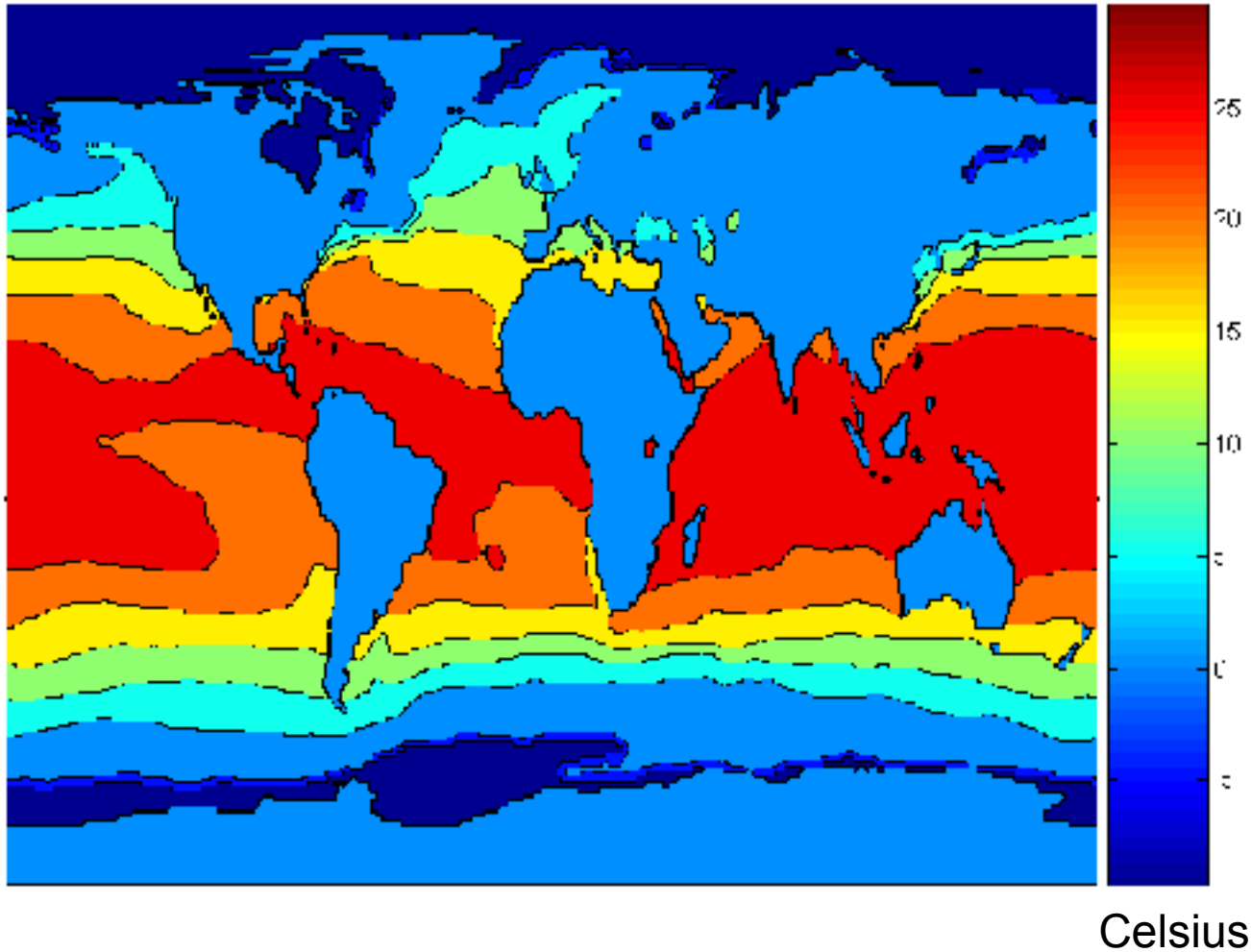
Supplemental Slides

- Peruse these at your own leisure!
- These slides might assist you as additional visual aides
- **Slides courtesy of Tan, Steinbach, Kumar**
 - **Introduction to Data Mining**

Visualization Techniques: Contour Plots

- Contour plots
 - Useful when a continuous attribute is measured on a spatial grid
 - They partition the plane into regions of similar values
 - The contour lines that form the boundaries of these regions connect points with equal values
 - The most common example is contour maps of elevation
 - Can also display temperature, rainfall, air pressure, etc.
 - ◆ An example for Sea Surface Temperature (SST) is provided on the next slide

Contour Plot Example: SST Dec, 1998



Other Visualization Techniques

- 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

Challenges of Data Mining

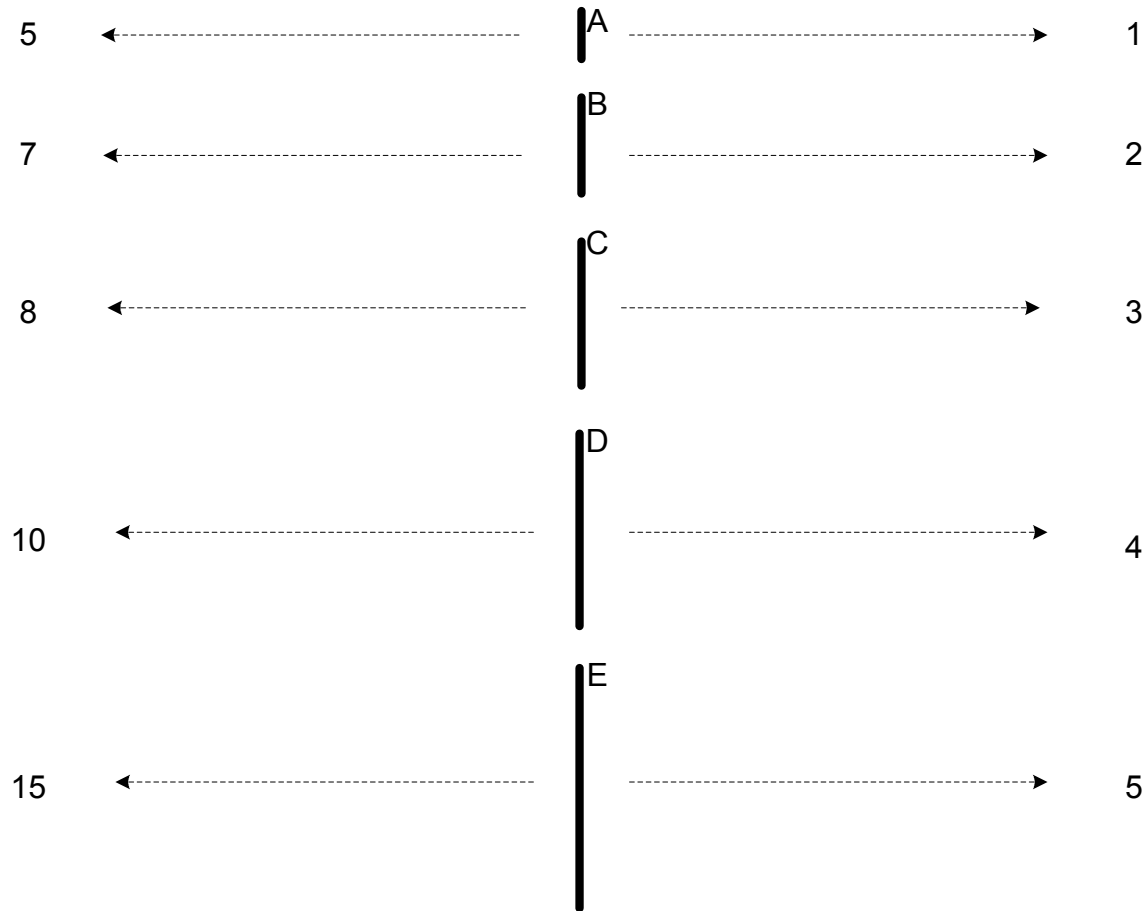
- Scalability
- Dimensionality
- Complex and Heterogeneous Data
- Data Quality
- Data Ownership and Distribution
- Privacy Preservation
- Streaming Data

Important Characteristics of Structured Data

- Dimensionality
 - ◆ Curse of Dimensionality
- Sparsity
 - ◆ Only presence counts
- Resolution
 - ◆ Patterns depend on the scale

Measurement of Length

- The way you measure an attribute is somewhat may not match the attributes properties.



Sampling

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

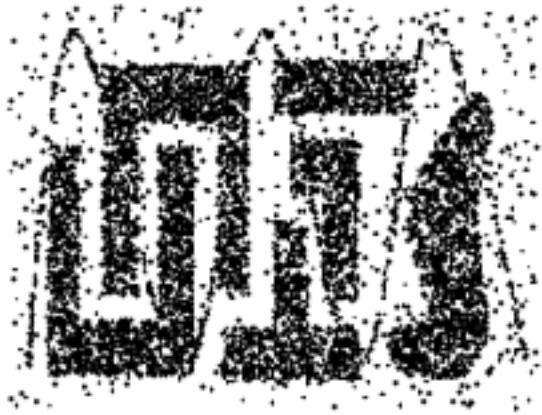
Sampling ...

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

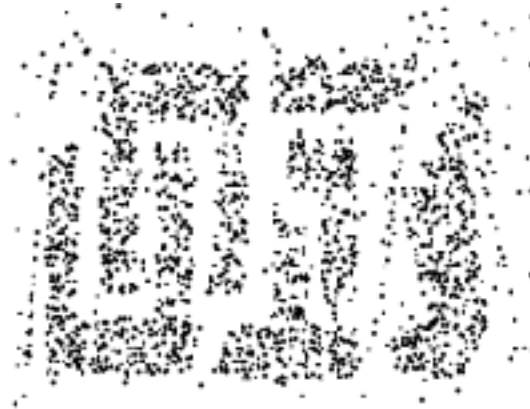
Types of Sampling

- 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

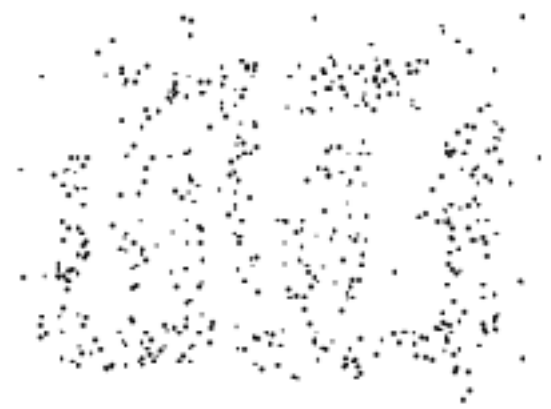
Sample Size



8000 points



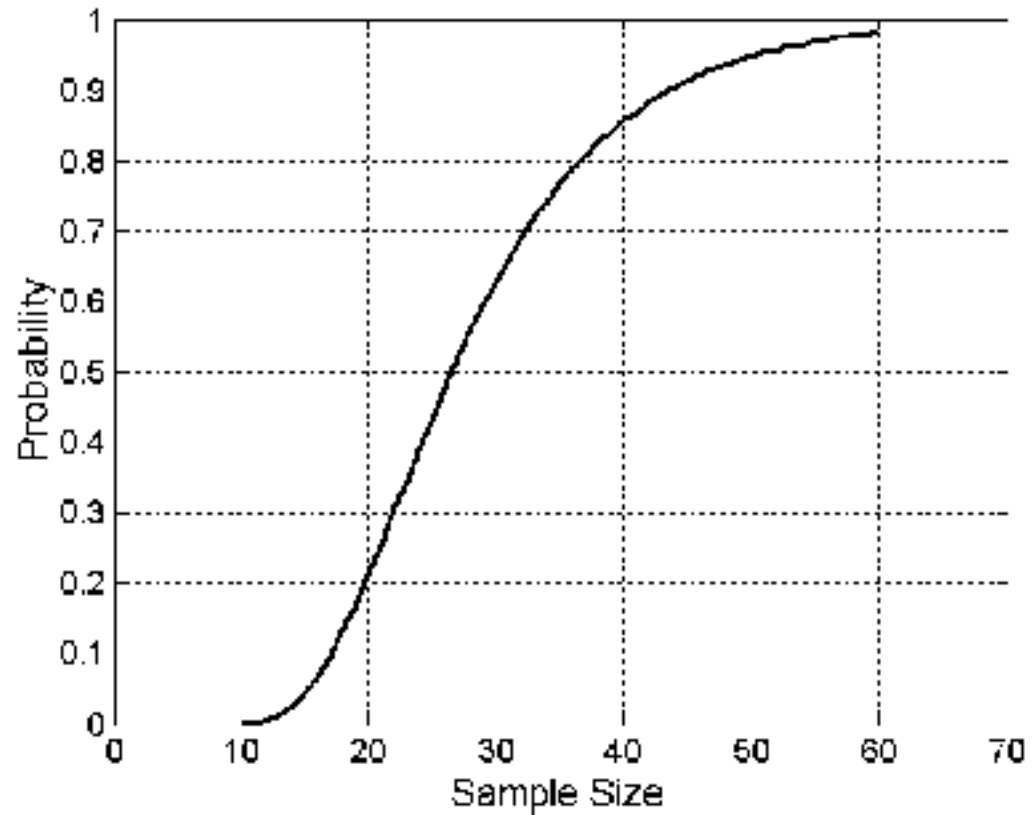
2000 Points



500 Points

Sample Size

- What sample size is necessary to get at least one object from each of 10 groups.



Similarity and Dissimilarity

- Similarity
 - Numerical measure of how alike two data objects are.
 - Is higher when objects are more alike.
 - Often falls 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	$d = \frac{ p-q }{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{ p-q }{n-1}$
Interval or Ratio	$d = p - q $	$s = -d, s = \frac{1}{1+d} \text{ or } s = 1 - \frac{d - \min_d}{\max_d - \min_d}$

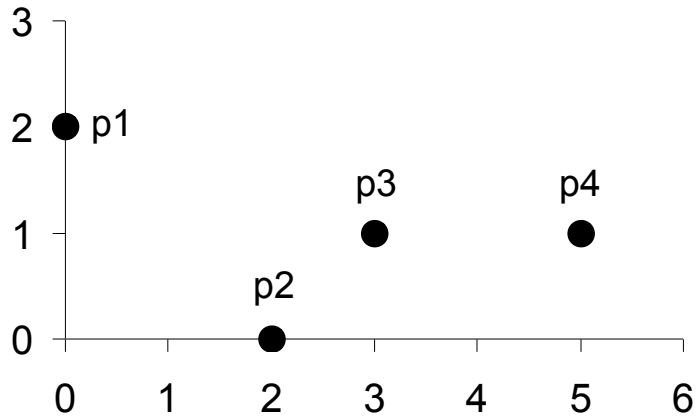
Table 5.1. Similarity and dissimilarity for simple attributes

Euclidean Distance

- Euclidean Distance

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

Euclidean Distance



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

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

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
- Do not confuse r with n , i.e., all these distances are defined for all numbers of dimensions.

Minkowski Distance

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

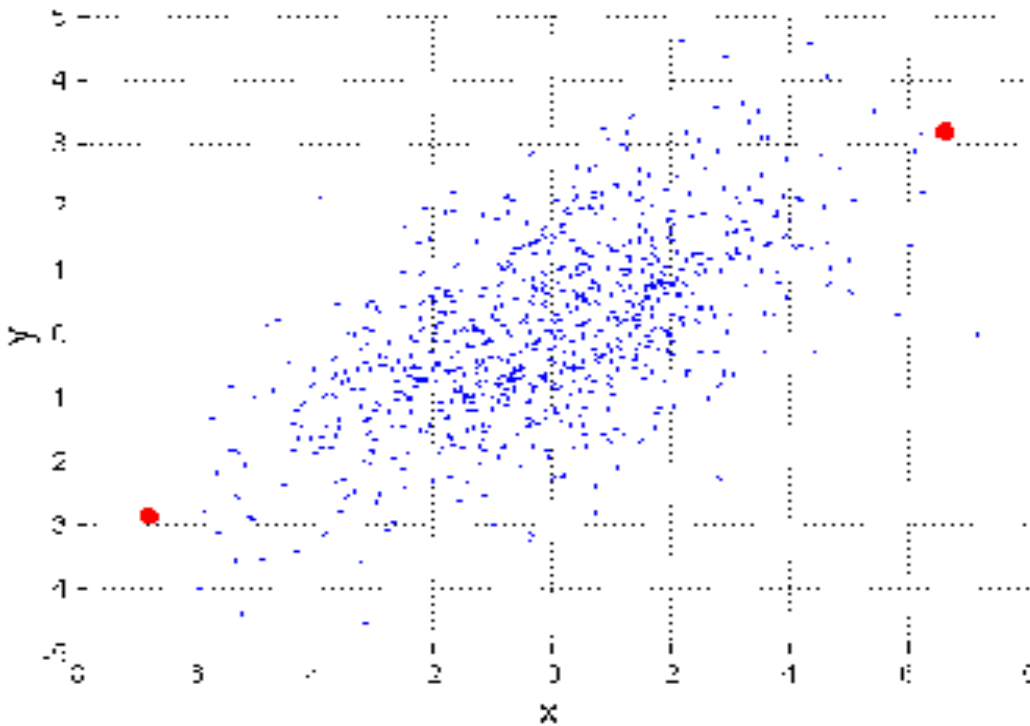
Distance Matrix

Mahalanobis Distance

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

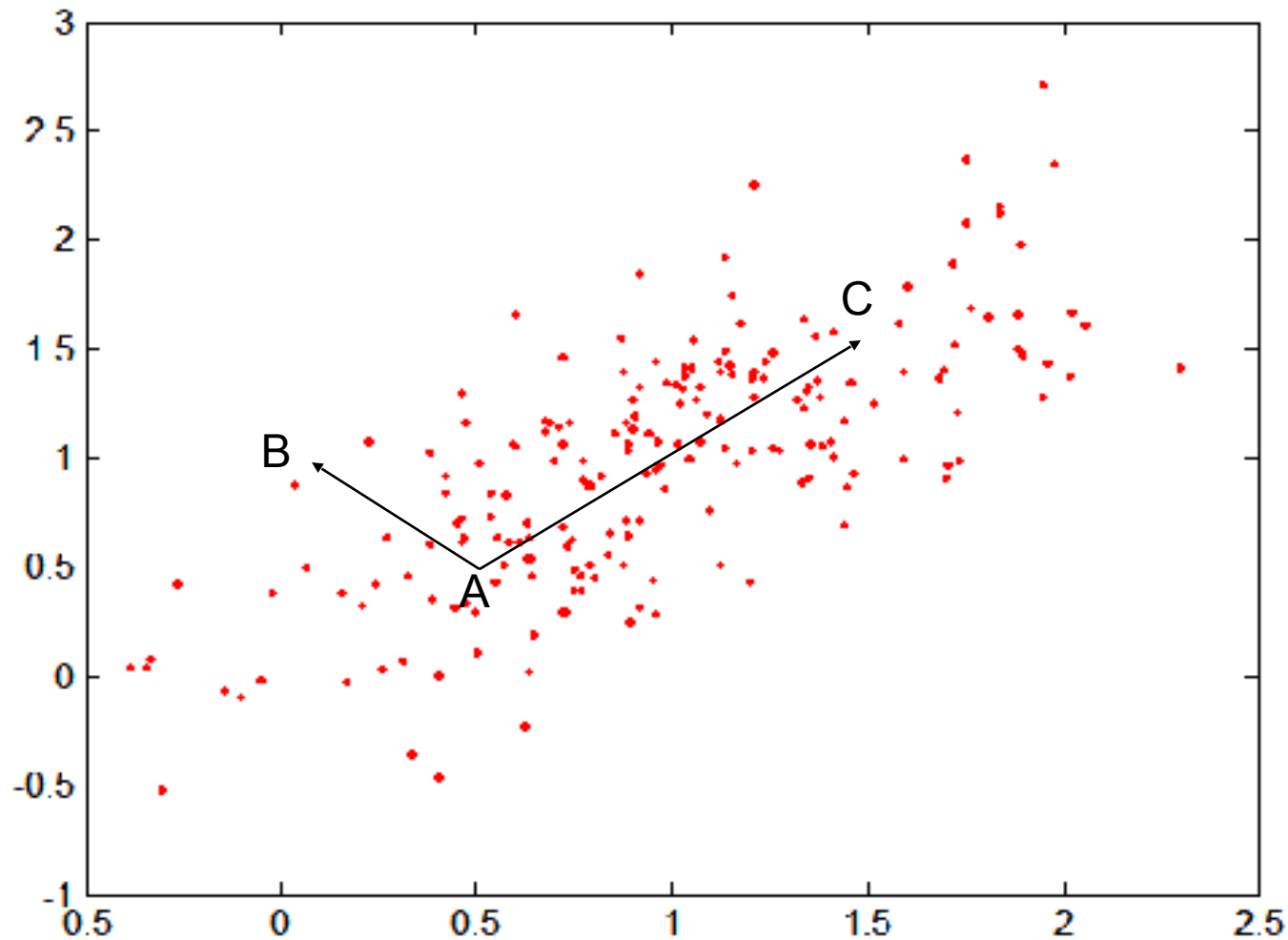
Σ is the covariance matrix of the input data X

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



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

Mahalanobis Distance



Covariance Matrix:

$$\Sigma = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

C: (1.5, 1.5)

Mahal(A,B) = 5

Mahal(A,C) = 4

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)
- i $d(p, q) = d(q, p)$ for all p and q . (Symmetry)
1. $d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q , and r . (Triangle Inequality)

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

1. A distance that satisfies these properties is a **metric**

Common Properties of a Similarity

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

SMC versus Jaccard: Example

$$p = 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0$$

$$q = 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1$$

$M_{01} = 2$ (the number of attributes where p was 0 and q was 1)

$M_{10} = 1$ (the number of attributes where p was 1 and q was 0)

$M_{00} = 7$ (the number of attributes where p was 0 and q was 0)

$M_{11} = 0$ (the number of attributes where p was 1 and q was 1)

$$SMC = (M_{11} + M_{00}) / (M_{01} + M_{10} + M_{11} + M_{00}) = (0+7) / (2+1+0+7) = 0.7$$

$$J = (M_{11}) / (M_{01} + M_{10} + M_{11}) = 0 / (2 + 1 + 0) = 0$$

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 \bullet 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

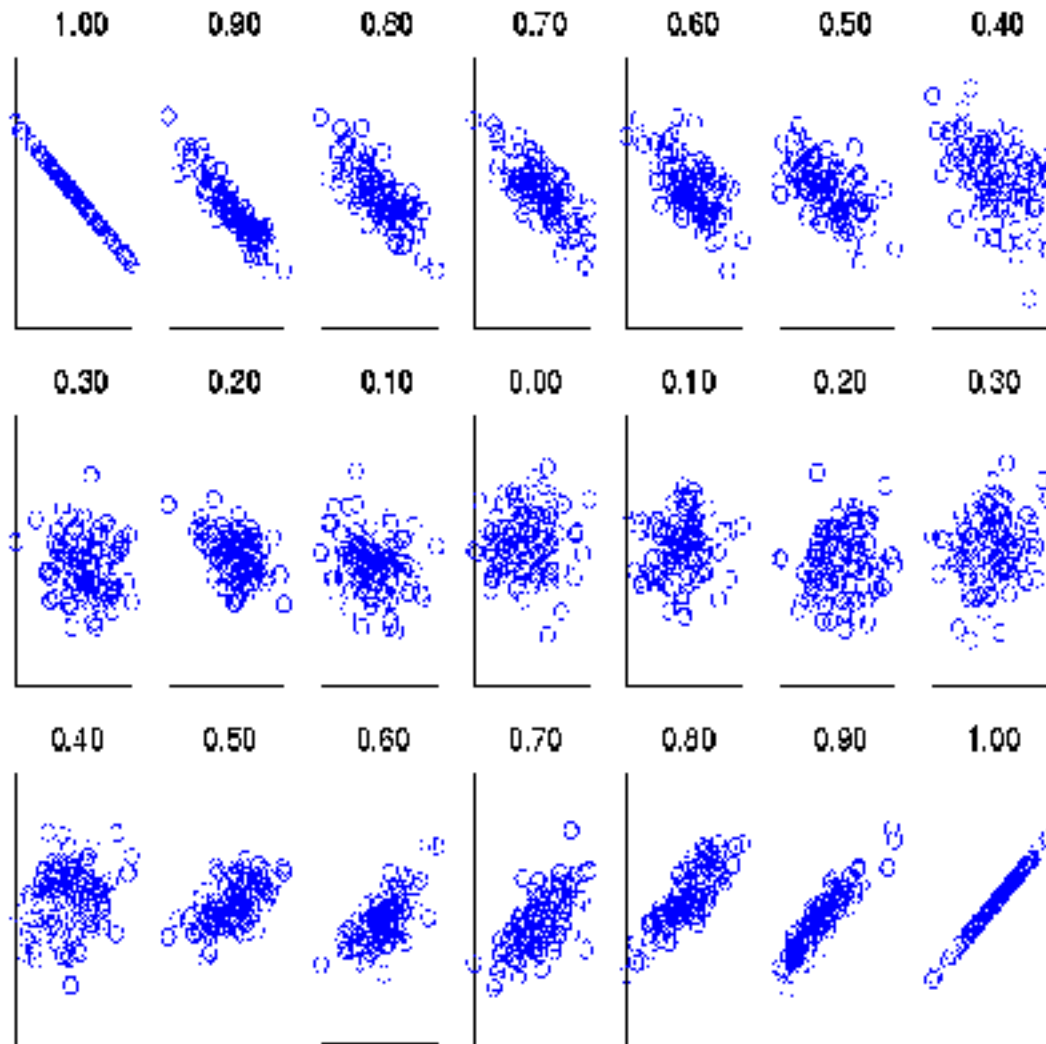
- Correlation measures the linear relationship between objects
- To compute correlation, we standardize data objects, p and q , and then take their dot product

$$p'_k = (p_k - \textit{mean}(p)) / \textit{std}(p)$$

$$q'_k = (q_k - \textit{mean}(q)) / \textit{std}(q)$$

$$\textit{correlation}(p, q) = p' \bullet q'$$

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:

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

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.

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

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

Density

- Density-based clustering require a notion of density
- Examples:
 - Euclidean density
 - ◆ Euclidean density = number of points per unit volume
 - Probability density
 - Graph-based density

Euclidean Density – Cell-based

- Simplest approach is to divide region into a number of rectangular cells of equal volume and define density as # of points the cell contains

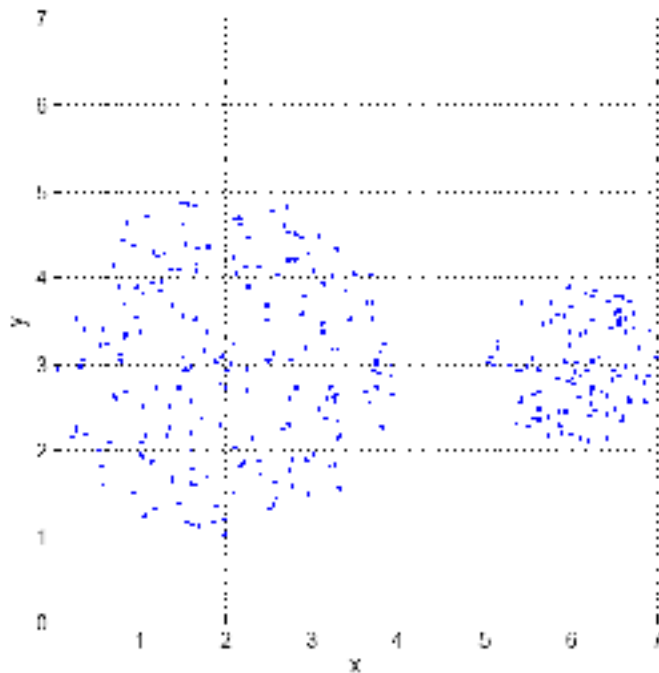


Figure 7.13. Cell-based density.

0	0	0	0	0	0	0
0	0	0	0	0	0	0
4	17	18	6	0	0	0
14	14	13	13	0	18	27
11	18	10	21	0	24	31
3	20	14	4	0	0	0
0	0	0	0	0	0	0

Table 7.6. Point counts for each grid cell.

Euclidean Density – Center-based

- Euclidean density is the number of points within a specified radius of the point

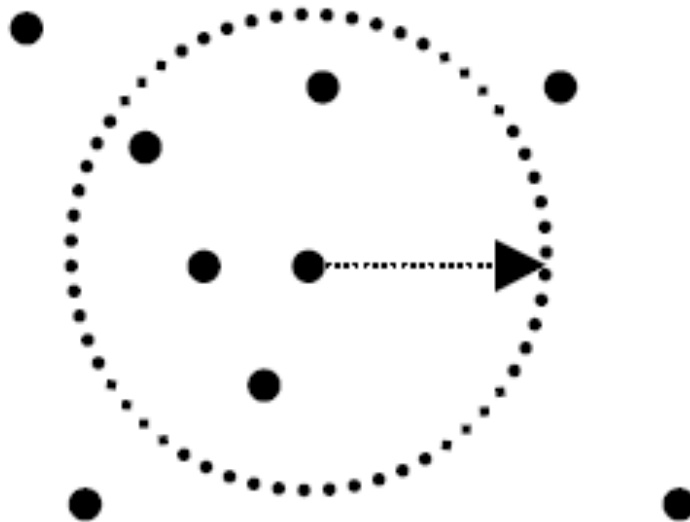


Figure 7.14. Illustration of center-based density.

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

Feature Subset Selection

- Techniques:
 - Brute-force approach:
 - ◆ Try all possible feature subsets as input to data mining algorithm
 - Embedded approaches:
 - ◆ Feature selection occurs naturally as part of the data mining algorithm
 - Filter approaches:
 - ◆ Features are selected before data mining algorithm is run
 - Wrapper approaches:
 - ◆ Use the data mining algorithm as a black box to find best subset of attributes

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

