Chapter 3: Data Preprocessing

Contents

Getting to know your data

Before jumping into mining, we need to familiarize ourselves with the data.

- What are the types of attributes that make up your data?
- What kind of values does each attribute have?
- Which attributes are discrete, and which are continuous-values?
- What the data look like?
- How are the values distributed?
- Are there ways we can visualize the data to get a better sense of it all?
- Can we spot any outliers?
- Can we measure the similarity of some data objects with respect to others? etc.

Exploratory Data Analysis (EDA)

An approach for data analysis that employs a variety of techniques for

- Better understanding the data
- Detection of mistakes
- Checking of assumptions
- Preliminary selection of appropriate models
- Determining relationships among the explanatory variables, and
- Assessing the direction and rough size of relationships between explanatory and outcome variables

Exploratory data analysis (EDA)

EDA can be **graphical** or **non-graphica**l.

Non-graphical methods generally involve calculation of summary statistics, while graphical methods summarize the data in a diagrammatic or pictorial way.

Non-graphical methods: Univariate data

- **Categorical data:**
  A simple tabulation of the frequency of each category

Statistic/College	H&SS	MCS	SCS	other	Total
Count	5	6	4	5	20
Proportion	0.25	0.30	0.20	0.25	1.00
Percent	25%	30%	20%	25%	100%

Source: http://www.stat.cmu.edu/~hseltman/309/Book/chapter4.pdf

Non-graphical methods: Univariate data

- **Quantitative data:** Population distribution of the variable using the data of the observed sample
  - *Central tendency*: mean, median, mode etc.
  - *Spread* (an indicator of how far away from the center we are still likely to find data values): variance, standard deviation, interquartile range (IQR)
  - *Modality* (number of peaks in the pdf)
  - *Shape*: Skewness, Kurtosis
  - *Outliers* (an observation that lies "far away" from other values)

Non-graphical methods: Bivariate data

- **Categorical data:**
  - *Cross-tabulation*

Age Group / Sex	Female	Male	Total
young	2	3	5
middle	2	1	3
old	3	0	3
Total	7	4	11

  - *Correlation*
    Chi-squared test, $$\chi^2 = \sum_{i,j} \frac{(n_{ij} - \frac{n_i . n_j}{n})^2}{\frac{n_i . n_j}{n}}$$

    Cramer's V, etc. $$V = \sqrt{\frac{\chi^2/n}{min(k-1, r-1)}}$$

Non-graphical methods: Bivariate data

- **Quantitative data**:
  - *Covariance*:
    A measure of how much two variables "co-vary"
    - +ve covariance = when one measurement is above the mean the other will probably also be above the mean, and vice versa
    - -ve covariance = when one variable is above its mean, the other is below its mean
    - Covariances near zero = the two variables vary independently of each other.
  - *Correlation*:
    Statistical relationship between two variables
    Commonly used correlation coefficient: Pearson's correlation coefficient
    - -1 = data lie on a perfect straight line with a negative slope
    - 0 = no linear relationship between the variables
    - 1 = data lie on a perfect straight line with a positive slope

Non-graphical methods: Multivariate data

- **Covariance and correlation matrices:**
  Pairwise covariances and/or correlations assembled into a matrix

	mpg	disp	hp	drat	wt	qsec
mpg	1.0000000	-0.8475514	-0.7761684	0.68117191	-0.8676594	0.41868403
disp	-0.8475514	1.0000000	0.7909486	-0.71021393	0.8879799	-0.43369788
hp	-0.7761684	0.7909486	1.0000000	-0.44875912	0.6587479	-0.70822339
drat	0.6811719	-0.7102139	-0.4487591	1.00000000	-0.7124406	0.09120476
wt	-0.8676594	0.8879799	0.6587479	-0.71244065	1.0000000	-0.17471588
qsec	0.4186840	-0.4336979	-0.7082234	0.09120476	-0.1747159	1.00000000

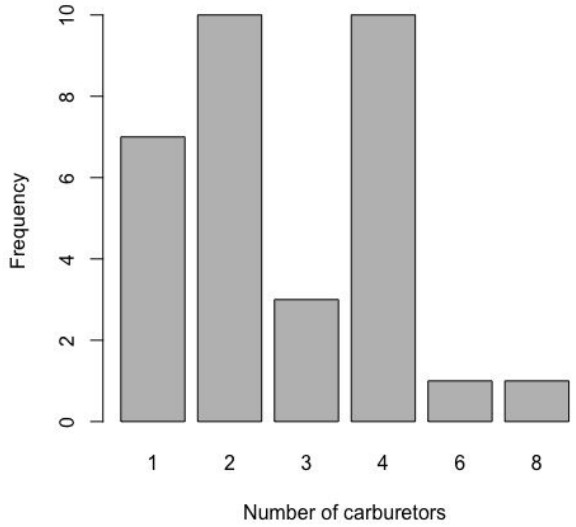
Graphical methods

Univariate data

- Bar chart
- Histogram
- Density plot
- Box and whiskers plot
- Time-series plot

Bar plot

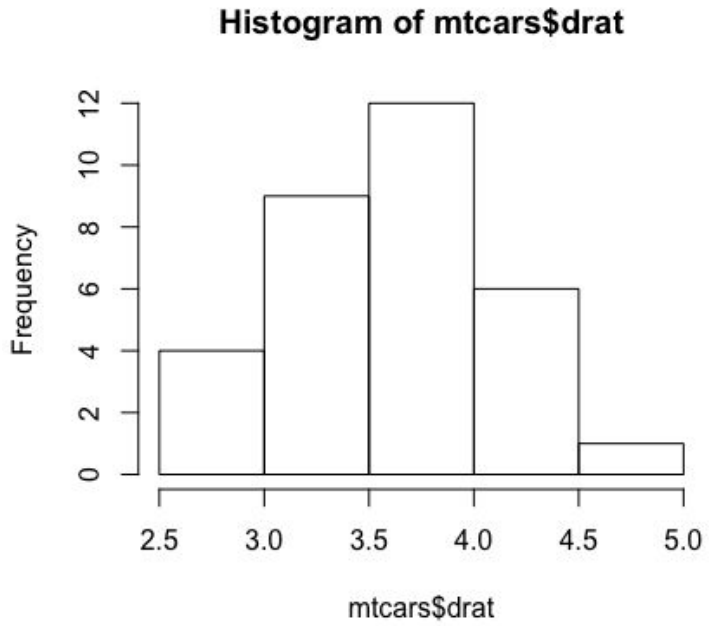
Bar plots display the distribution (frequencies) of a categorical variable through vertical or horizontal bars.



Histogram

Histograms are constructed by binning the data and counting the number of observations in each bin

To visualize the shape of the distribution



Histogram

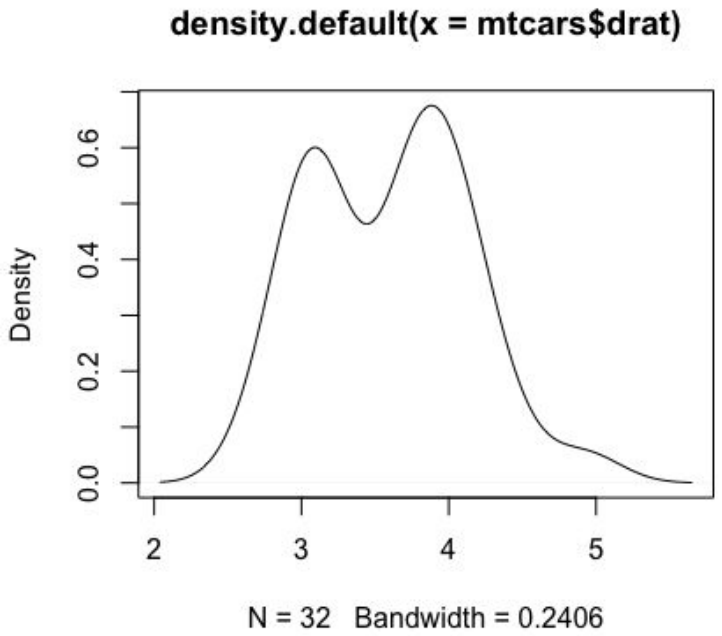
Bar plot vs histogram

- Bar plots use categorical data
- Histograms use continuous data grouped or categorized (binning) in such a way that they are considered to be ranges

- Normally, bars in bar plots do not touch each other
- Bars in histograms touch each other to indicate that the items are non-discrete

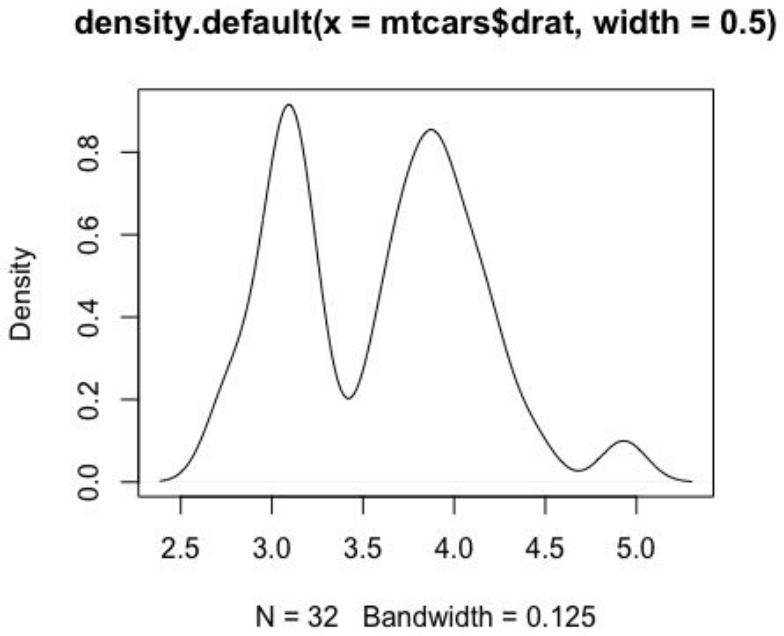
Density plot

Density plots can be thought of as plots of smoothed histograms



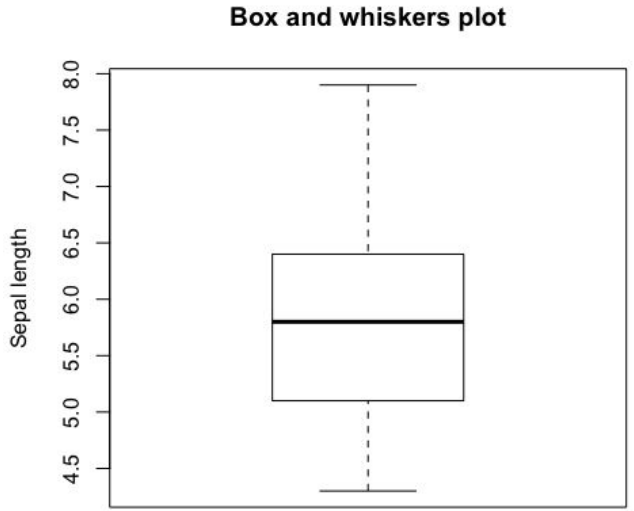
Density plot

The smoothness is controlled by a bandwidth parameter that is analogous to the histogram binwidth

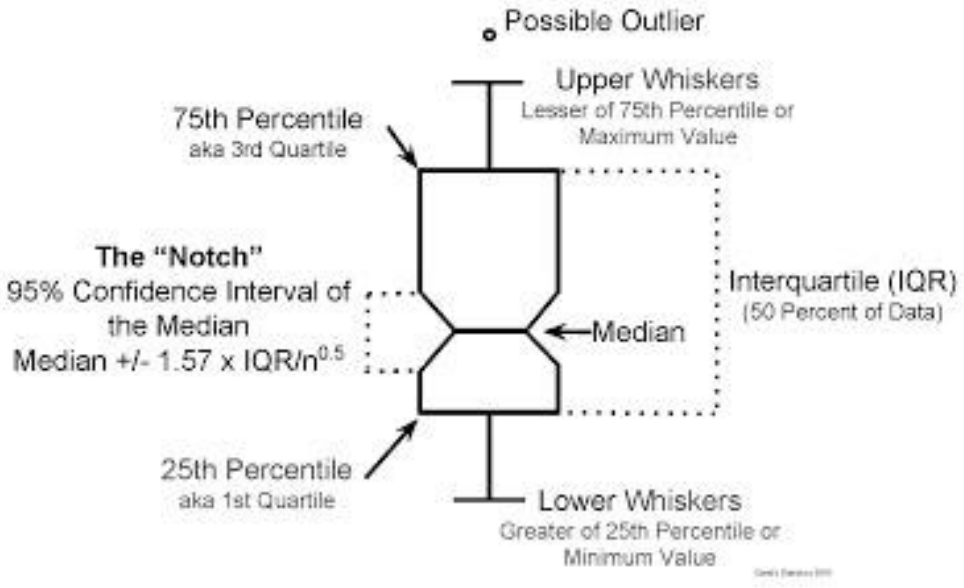


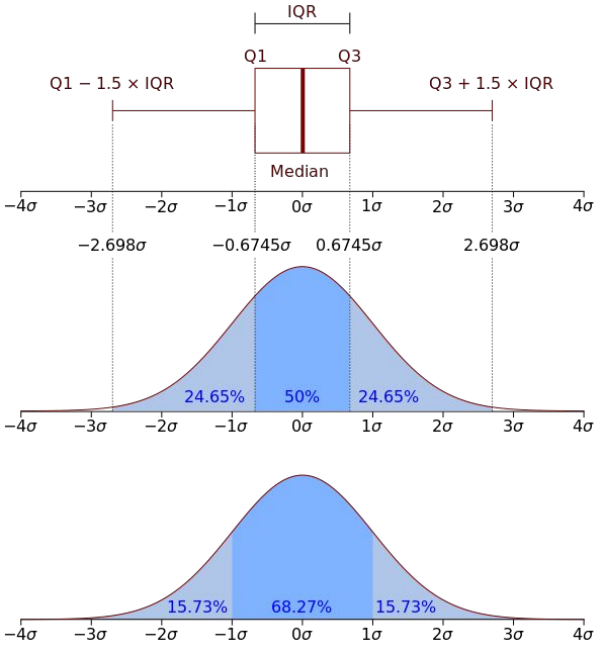
Box and whiskers plot

Presents information about the central tendency, symmetry, and skew as well as outliers



Box and whiskers plot

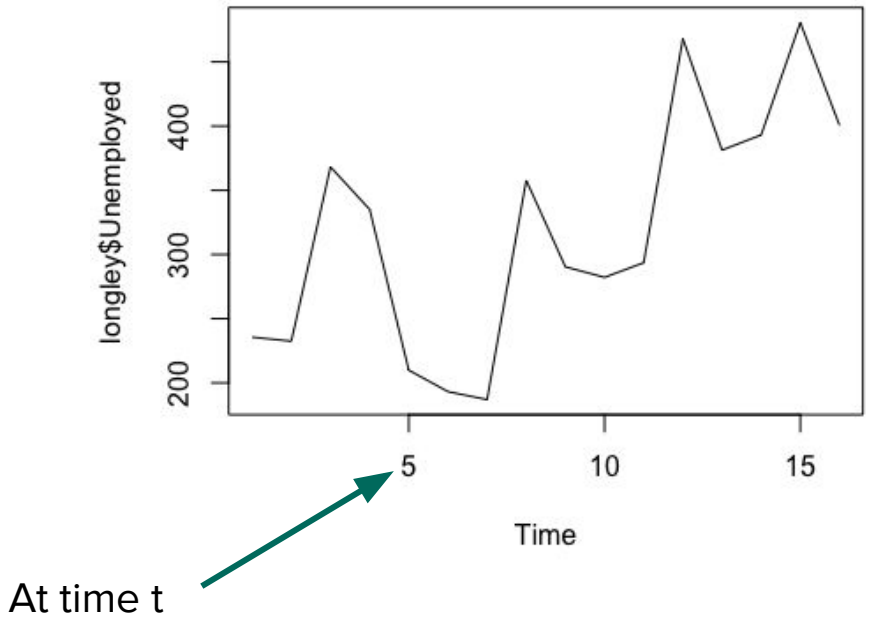




Source: https://sites.google.com/site/davidsstatistics/home/notched-box-plots

https://en.wikipedia.org/wiki/Box_plot

Time series plot



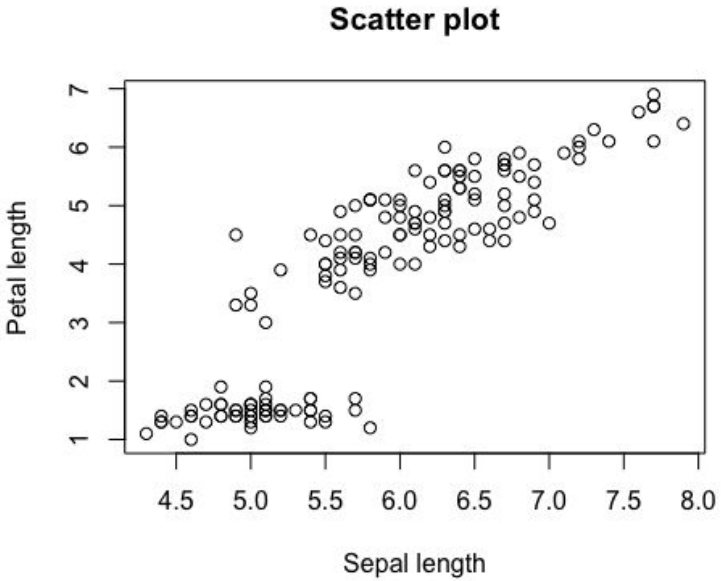
Graphical methods

Bivariate data

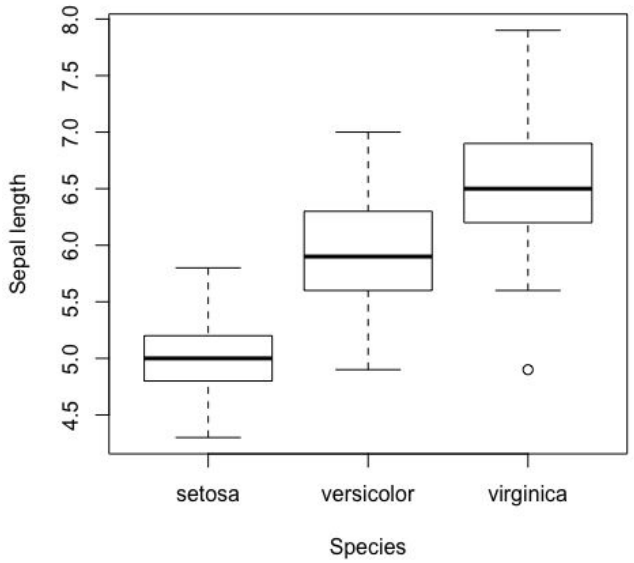
- Scatter plot
- Regression
- Box and whiskers plot
- Bar chart

Scatter plot

Uses Cartesian coordinates to show the relationship between two variables of a set of data

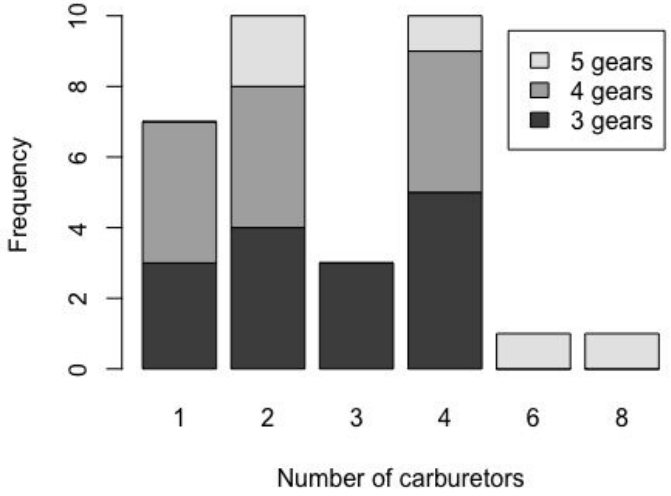


Box and whiskers plot



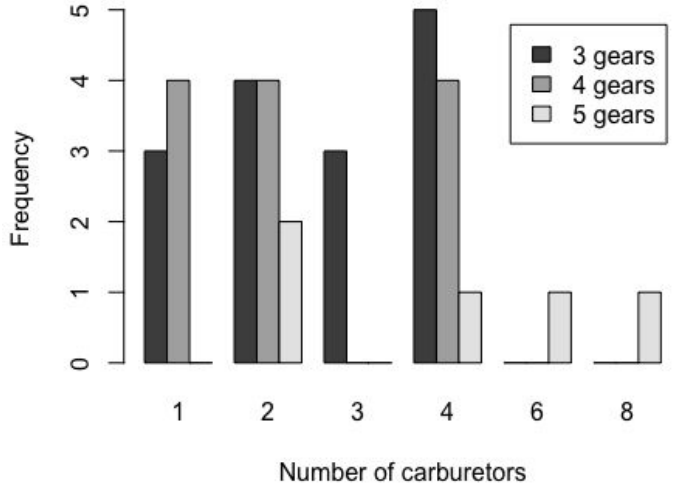
Bar plot

Stacked bar plot



Bar plot

Grouped bar plot



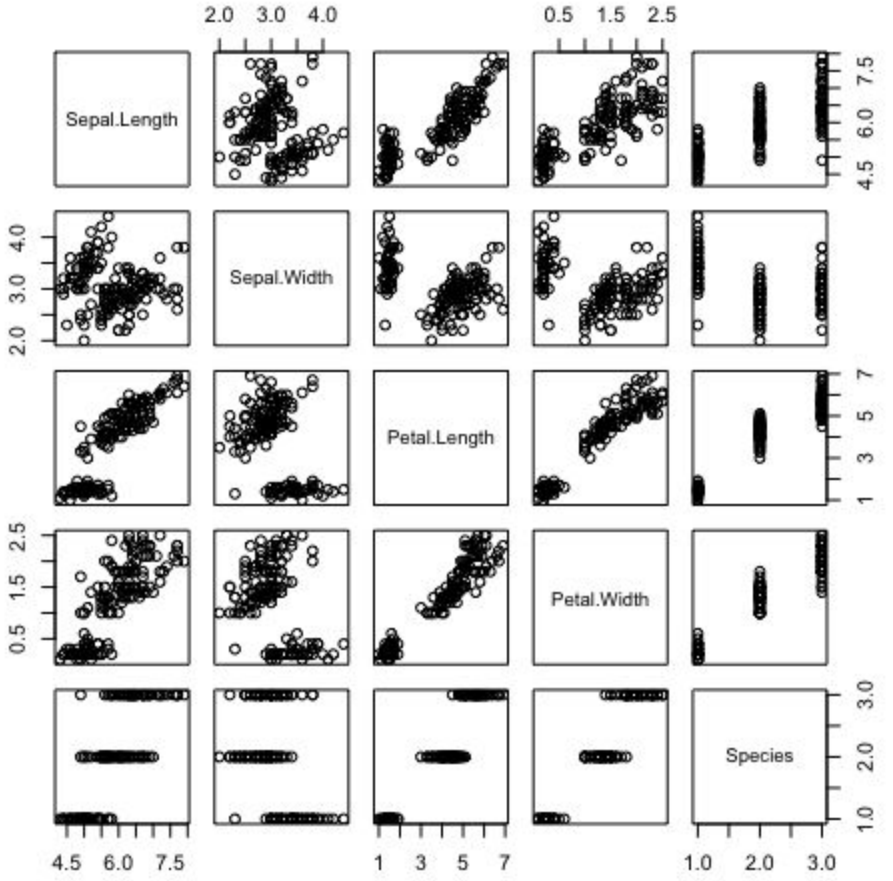
Graphical methods

Multivariate data

- Scatter plot matrix
- Bubble plot
- Line chart

Scatter plot matrix

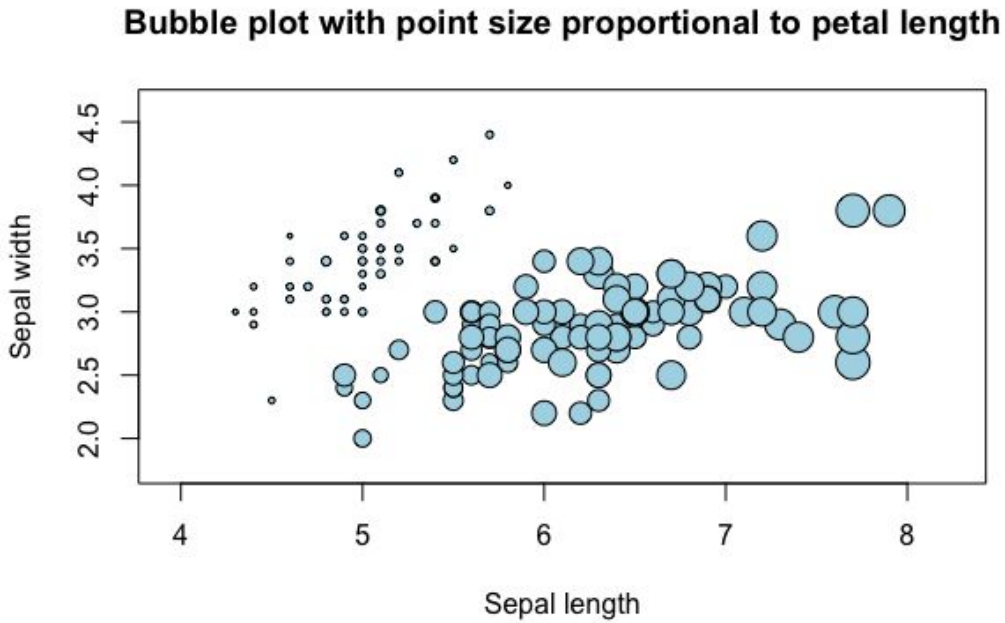
Can be used to roughly determine if there is a linear correlation between multiple variables



Bubble plot

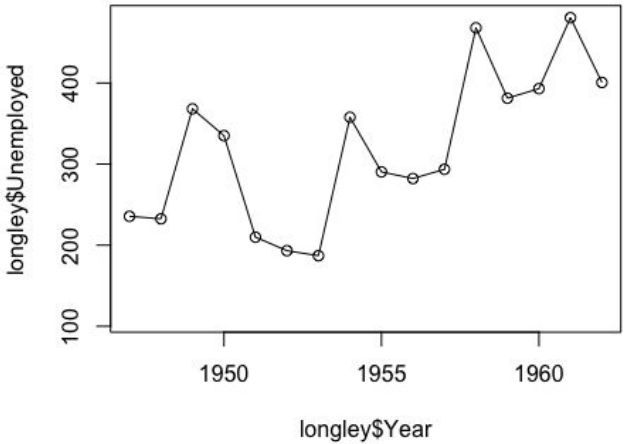
Bubble plots can display the relationship between three quantitative variables

A bubble plot is basically a 2D scatter plot that uses the size of the plotted point to represent the value of the third variable

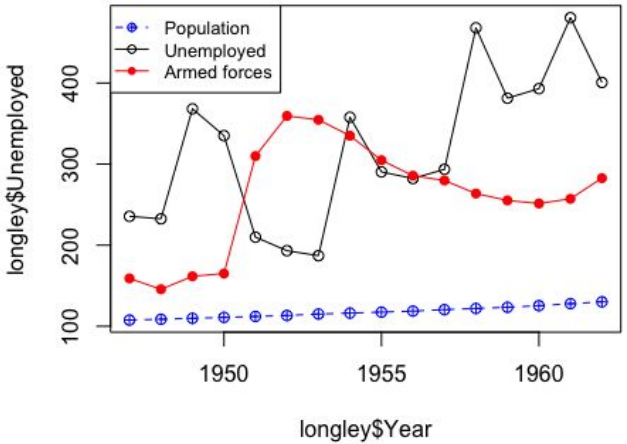


Line chart

Connecting the points in a scatter plot moving from left to right gives a line plot



Multiple lines can be drawn in the same plot



Graphical methods

Quantitative	**Categorical**	**Quantitative & Categorical**
Time-series plot Univariate: *Histograms*, *Density plots*, *Box and whiskers plots* Bivariate: *Scatterplots* Multivariate: *Scatterplot matrix*, *Bubble plots* (3 variables)	Univariate: *Pie charts*, *Bar graphs* Bi- or multi-variate: *Bar graphs*	*Box and whiskers plots*

Why preprocess the data?

In many practical situations

- Data contains too many attributes, and some of them are clearly irrelevant and redundant
- Data is incomplete (some values are missing)
- Data is inaccurate or noisy (contains errors, or values that deviate from the expected)
- Data is inconsistent (e.g., containing discrepancies in the department codes)

Garbage-In-Garbage-Out (GIGO): *Low-quality data will lead to low-quality mining results*

Data quality

- Accuracy
- Completeness
- Consistency
- Timeliness
- Believability
- Interpretability

Why is the real world data dirty?

- Inaccurate (or noisy) data may come from
  - Faulty data collection instruments or methods
  - Human or computer errors while entering data
  - Incorrect data purposely submitted by users (aka disguised missing data)
  - Technological limitations
  - Inconsistencies in naming conventions or inconsistent formats for input fields
  - etc.
- Inconsistent data may come from
  - Integration of data from various sources
  - Modification of linked data
  - etc.

- Incomplete data may come from
  - Unavailability of attributes of interest
  - Recording missed due to equipment malfunctions
  - Deletion of inconsistent data
  - Data not submitted in timely fashion
  - etc.

Major Tasks in Data Preprocessing

- Data cleaning
  - Fill in missing values, smooth noisy data, identify or remove outliers, resolve inconsistencies, detect and remove redundancies
- Data integration
  - Include data from multiple sources (e.g., multiple databases, data cubes, files etc.)
- Data reduction
  - Obtain a reduced representation of the data set that is much smaller in volume, yet produces the same (or almost the same) analytical results
- Data transformation and data discretization
  - Normalize data values, replace raw data values by ranges or higher conceptual levels (e.g., replacing age by higher-level concepts, such as youth, adult or senior) etc.

Data cleaning

- Tasks
  - Handle missing values,
  - Smooth noisy data,
  - Identify or remove outliers,
  - Resolve inconsistencies,
  - Detect and remove redundancies

Missing data

- Data may not always be available
- Incomplete data may come from
  - Unavailability of attributes of interest
  - Recording missed due to equipment malfunctions
  - Deletion of inconsistent data
  - Data not submitted in timely fashion
  - etc.
- Missing data may need to be inferred

Handling missing values

- Ignore the tuple
  - Usually done when the class level is missing
  - Poor when the percentage of missing values per attribute varies considerably
- Use a global constant (e.g., NA, Unknown, -∞ etc.) to fill in the missing value
  - This method is simple but not foolproof as the mining program may mistakenly think that they form an interesting concept
- Fill in the missing values manually
  - Time consuming
  - May not be feasible for a large data set
- Fill in the missing values automatically
  - Using random values
  - Using a measure of central tendency for the attribute (e.g., mean, median)
  - Using the attribute mean or median for all samples belonging to the same class
  - Using the most probable value (e.g., with regression, decision tree, Bayesian inference etc.)

Noisy data

- Noise is a random error or a variance in measured variable.
- Inaccurate (or noisy) data may come from
  - Faulty data collection instruments or methods
  - Human or computer errors while entering data
  - Incorrect data purposely submitted by users (aka disguised missing data)
  - Technological limitations
  - Inconsistencies in naming conventions or inconsistent formats for input fields
  - etc.

Handling noisy data

- Binning
  - First sort the data, and distribute the sorted values into a number of buckets or bins (equal-frequency bins or equal-width bins)
  - Then replace each value in a bin by the mean of the bin (smoothing by bin means), or by the median of the bin (smoothing by bin medians), or by the closest boundary value (smoothing by bin boundaries)
- Regression
  - Fitting the data into regression functions
  - In (simple) linear regression, the data are modeled to fit a straight line.

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

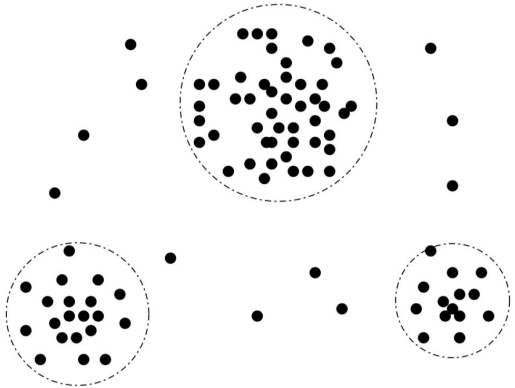
Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

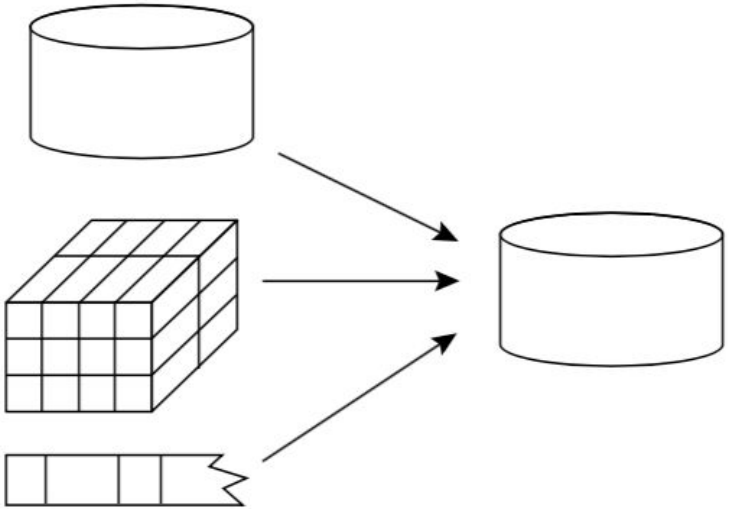
Handling noisy data

- Clustering / Outlier analysis
  - Grouping similar values into groups or clusters. Values that fall outside of the set of clusters may be considered outliers
- Concept hierarchy
  - Replacing data with higher-level concepts, e.g., a concept hierarchy of price may map real values into inexpensive, moderately_priced, and expensive.
- Combined human and computer inspection
  - Detect suspicious values and check by human
  - Correct the data using external references



Data Integration

- Combine data from multiple sources



Data Integration

- Challenges
  - Entity identification problem
  - Redundancies
  - Tuple duplication
  - Data value conflict

Data Integration: Challenges

- Entity identification problem
  - How can equivalent real-world entities from multiple sources be matched up?
  - Schema integration
    - How to be sure that customer_id in one database and cust_number in another refer to the same attribute?
    - Data codes for for pay_type in one database are “H” and “S” but 1 and 2 in another.
  - Object matching
    - Bill Clinton = William Clinton
  - The metadata of the attributes (name, meaning, data type, range of values permitted for the attributes, and null rules for handling blank, zero, and null values etc.) can be used to match attributes.

Data Integration: Challenges

Redundancy

- Data integration may result in duplicate attributes or duplicate tuples.
- Redundant data may occur
  - Due to inconsistencies in attribute naming across multiple sources
  - Due to the use of denormalized tables
  - When an attribute can be derived from another attribute or a set of attributes

Handling redundant data

- Correlation analysis can detect attribute redundancy
  - Given two attributes, correlation analysis can measure how strongly one attribute implies the other, based on the available data
  - Numerical data: Pearson's correlation coefficient
  - Categorical data: Chi-square test
- Different probabilistic approaches and machine learning techniques can be used to detect duplicate tuples
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

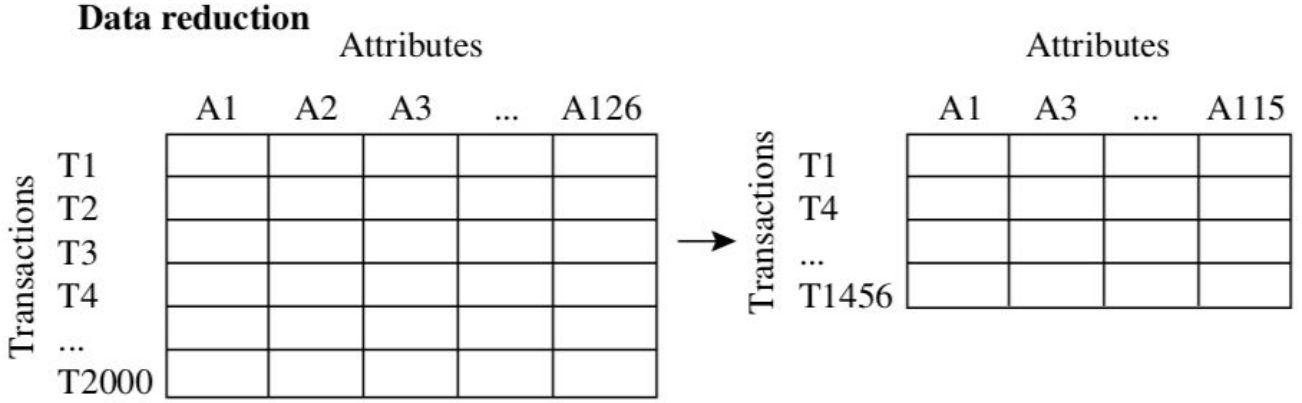
Data Integration: Challenges

Data value conflict

- For the same real world entity, attribute values from different sources are different
- Possible reasons: different representations, different scales, e.g., metric vs. British units

Data reduction

- Complex data analysis and mining on huge amounts of data can take a long time
- Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data



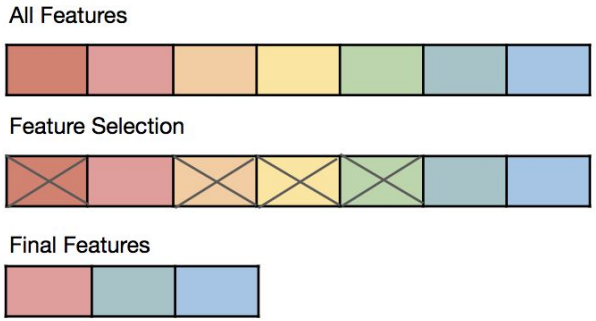
Data reduction

- Data reduction strategies
  - Dimensionality reduction
    Reducing the number of random variables or attributes under consideration
  - Numerosity reduction
    Replacing the original data volume by alternative, smaller forms of data representation
  - Data compression
    Applying transformations to obtain a reduced or “compressed” representation of the original data

Dimensionality reduction

- Reducing the number of random variables or attributes under consideration
- Some methods:
  - Attribute subset selection / feature selection
  - Principal Component Analysis
  - Wavelet Transforms

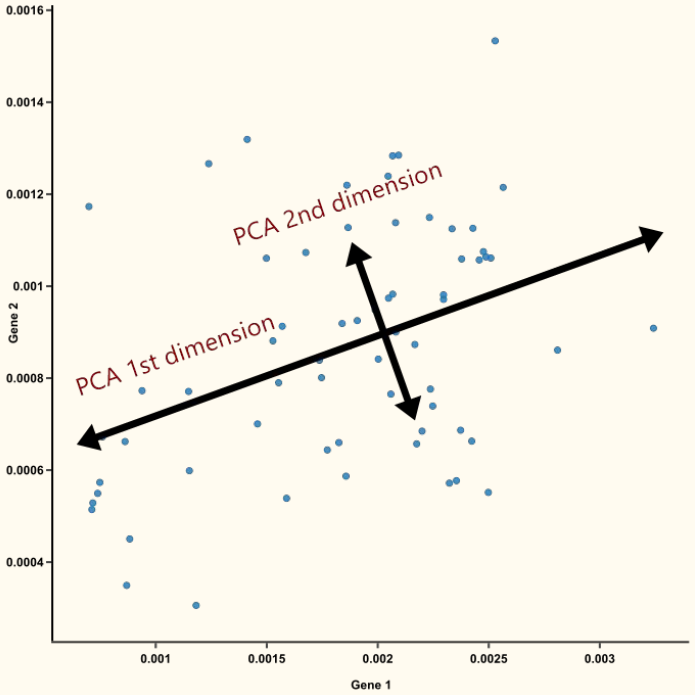
Attribute subset selection



- Reduces the data set size by removing irrelevant or redundant attributes (or dimensions)
- The goal is to find a minimum set of attributes such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all attributes.
- For n attributes, there are 2^n possible subsets.
  - An exhaustive search for the optimal subset of attributes (brute-force approach) may not be feasible
  - Heuristic/greedy methods that explore a reduced search space are commonly used for attribute subset selection.
  - Stepwise forward selection, stepwise backward selection, decision tree induction etc.

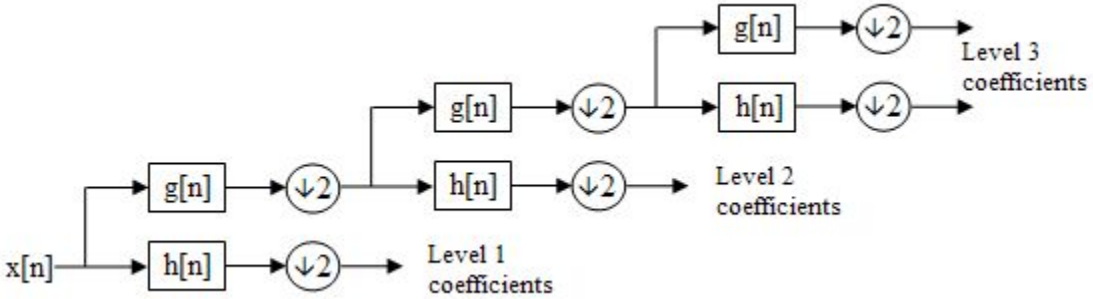
Principal component analysis

- Transforms the data from d-dimensional space into a new coordinate system of dimension p, where $p \leq d$
- Unlike attribute subset selection, which reduces the attribute set size by retaining a subset of the initial set of attributes, PCA "combines" the essence of attributes by creating an alternative, smaller set of variables. The initial data can then be projected onto this smaller set.



Wavelet transforms

- The discrete wavelet transform (DWT) is a linear signal processing technique that, when applied to a data vector X, transforms it to a numerically different vector, X′, of wavelet coefficients. The two vectors are of the same length.
- A compressed approximation of the data can be retained by storing only a small fraction of the strongest of the wavelet coefficients.



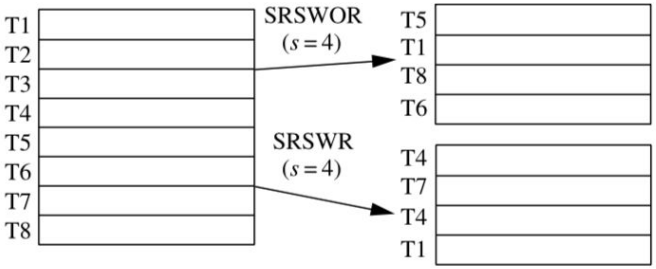
Numerosity reduction

- Replacing the original data volume by alternative, smaller forms of data representation
- These techniques may be parametric or non-parametric
  - Parametric: Regression and log-linear models
  - Non-parametric: Sampling, histograms, clustering

Sampling

- Allows a large data set to be represented by a much smaller random data sample (or subset)
- Types
  - Simple random sampling with replacement (SRSWR)
  - Simple random sampling without replacement (SRSWOR)
  - Cluster sampling
  - Stratified sampling

Sampling

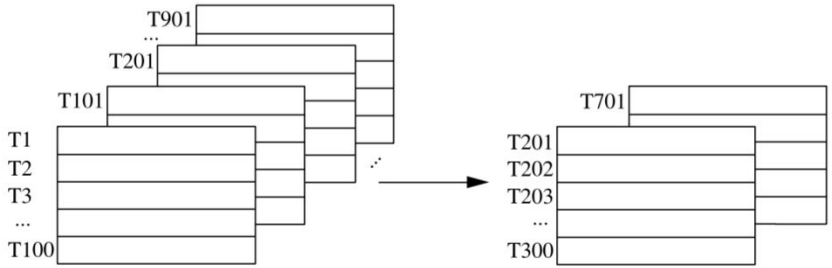


Startified sample
(according to *age*)

T38	youth
T256	youth
T307	youth
T391	youth
T96	middle_aged
T117	middle_aged
T138	middle_aged
T263	middle_aged
T290	middle_aged
T308	middle_aged
T326	middle_aged
T387	middle_aged
T69	senior
T284	senior

T38	youth
T391	youth
T117	middle_aged
T138	middle_aged
T290	middle_aged
T326	middle_aged
T69	senior

Cluster sample
(*s* = 2)



Clustering

- Partition the objects into groups, or clusters, so that objects within a cluster are “similar” to one another and “dissimilar” to objects in other clusters
- In data reduction, the cluster representations of the data are used to replace the actual data

Data compression

- Applying transformations to obtain a reduced or “compressed” representation of the original data
- If the original data can be reconstructed from the compressed data without any information loss, the data reduction is called **lossless.**
- If we can reconstruct only an approximation of the original data, then the data reduction is called **lossy**. Examples: PCA, DWT

Data transformation

- Transforming or consolidating data into forms appropriate for mining
- Strategies
  - Smoothing: removing noise from data
  - Attribute construction: constructing new attributes from the given set of attributes
  - Aggregation: summarizing or aggregating the data
  - Normalization: scaling the attribute data so that they fall within a smaller range
  - Discretization: replacing numerical attribute values with interval labels or conceptual labels
  - Concept hierarchy generation for nominal data: constructing hierarchies of concepts by generalizing concepts to higher level

Normalization

- Transforming the data to fall within a smaller or common range such as [—1,1] or [0.0, 1.0].
- Normalization is particularly useful for classification algorithms involving neural networks or distance measurements such as nearest-neighbor classification and clustering.
- For distance-based methods, normalization helps prevent attributes with initially large ranges (e.g., income) from outweighing attributes with initially smaller ranges (e.g., binary attributes).
- Some methods: min-max normalization, z-score normalization, normalization by decimal scaling etc.

Min-max normalization

Performs a linear transformation on the original data.

Suppose that $\min_A$ and $\max_A$ are the minimum and maximum values of an attribute, A. Min-max normalization maps a value, v_i, of A to v_i' in the range $[new\_min_A, new\_max_A]$ by computing

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A.$$

Min-max normalization

Suppose that the minimum and maximum values for the attribute income are $12,000 and $98,000, respectively. We would like to map income to the range [0.0,1.0]. By min-max normalization, what will be the new value for $73,600?

Z-score normalization

The values for an attribute, A, are normalized based on the mean (i.e., average) and standard deviation of A.

A value, v_i, of A is normalized to v_i' by computing

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A},$$

Where $\bar{A}$ and σ_A are the mean and standard deviation, respectively, of attribute A.

Z-score normalization is useful when the actual minimum and maximum of attribute A are unknown, or when there are outliers that dominate the min-max normalization.

Z-score normalization

Suppose that the mean and standard deviation of the values for the attribute income are $54,000 and $16,000, respectively.

With z-score normalization, what will be the new value for $73,600?

Normalizing by decimal scaling

Normalizes by moving the decimal point of values of attribute A.

A value, v_i, of A is normalized to v_i' by computing

$$v_i' = \frac{v_i}{10^j},$$

where j is the smallest integer such that $\max(|v_i'|) < 1$.

Example: Suppose that the recorded values of A range from —986 to 917. The maximum absolute value of A is 986. To normalize by decimal scaling, we therefore divide each value by 1000 (i.e., j = 3) so that —986 normalizes to —0.986 and 917 normalizes to 0.917.

Discretization

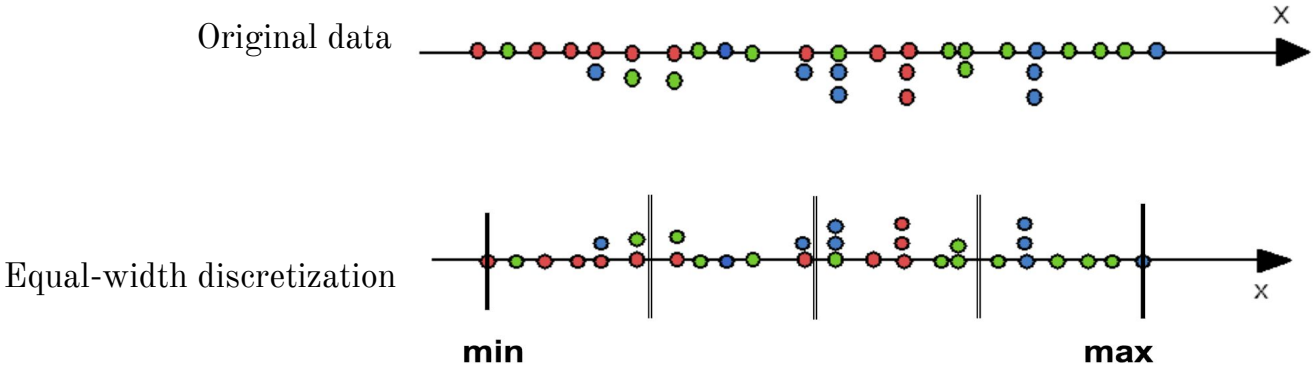
- Dividing the range of a continuous attribute into intervals
- Interval labels can then be used to replace actual data values
- Some classification algorithms only accept categorical attributes
- Types:
  - Supervised (using class information) vs unsupervised (without using class information)
  - Top-down / splitting or bottom-up / merging

Discretization

- Unsupervised discretization
  - Equal-width or equal-frequency
- Supervised discretization
  - Clustering, decision tree
  - Entropy-based discretization
  - Chi square discretization
  - etc.

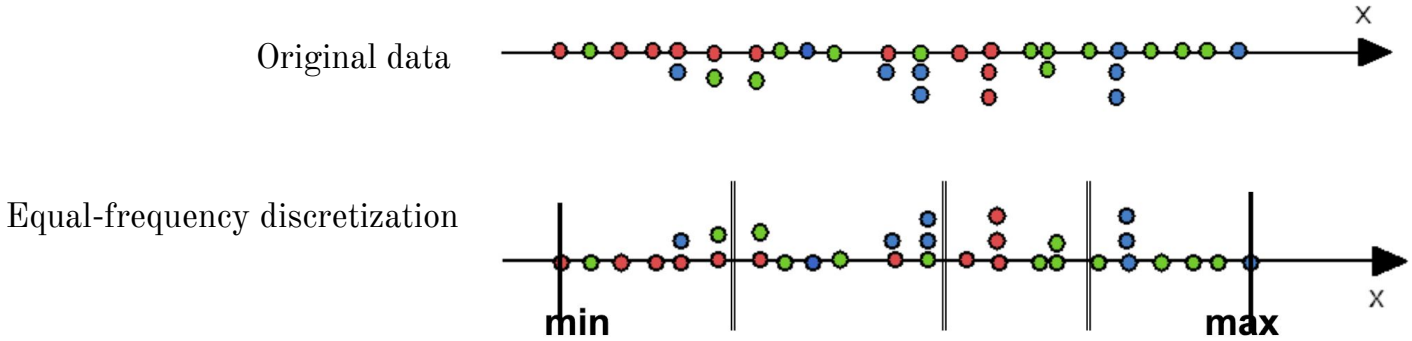
Unsupervised discretization

- Does not used class information
- Equal-width discretization
  - First, find the minimum and maximum values for the continuous attribute
  - Then, divide the range of the attribute values into the user-specified equal-width discrete intervals



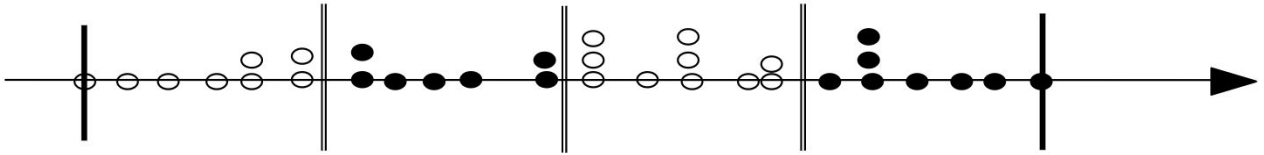
Unsupervised discretization

- Equal-frequency discretization
  - Sort the values of the attribute in ascending order
  - Find the number of all possible values for the attribute
  - Then, divide the attribute values into the user-specified number of intervals such that each interval contains the same number of sorted sequential values



Supervised discretization

Uses class information



Concept hierarchy generation for nominal data

- Nominal attributes have a finite (but possibly large) number of distinct values, with no ordering among the values. Examples: geographic_location, job_category etc.
- The concept hierarchies can be used to transform the data into multiple levels of granularity.

Concept hierarchy generation for nominal data

- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
  - street $<$ city $<$ state $<$ country
- Specification of a hierarchy for a set of values by explicit data grouping
  - {Urbana, Champaign, Chicago} $<$ Illinois
- Specification of only a partial set of attributes
  - e.g., only street $<$ city, not others
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
  - e.g., for a set of attributes: {street, city, state, country}

Automatic concept hierarchy generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
- The attribute with the most distinct values is placed at the lowest level of the hierarchy