3 – Data in CI for the IoT

Computational Intelligence for the Internet of Things (2019-20)

João Paulo Carvalho
joao.carvalho@inesc-id.pt

INESC-ID
Instituto Superior Técnico, Universidade de Lisboa

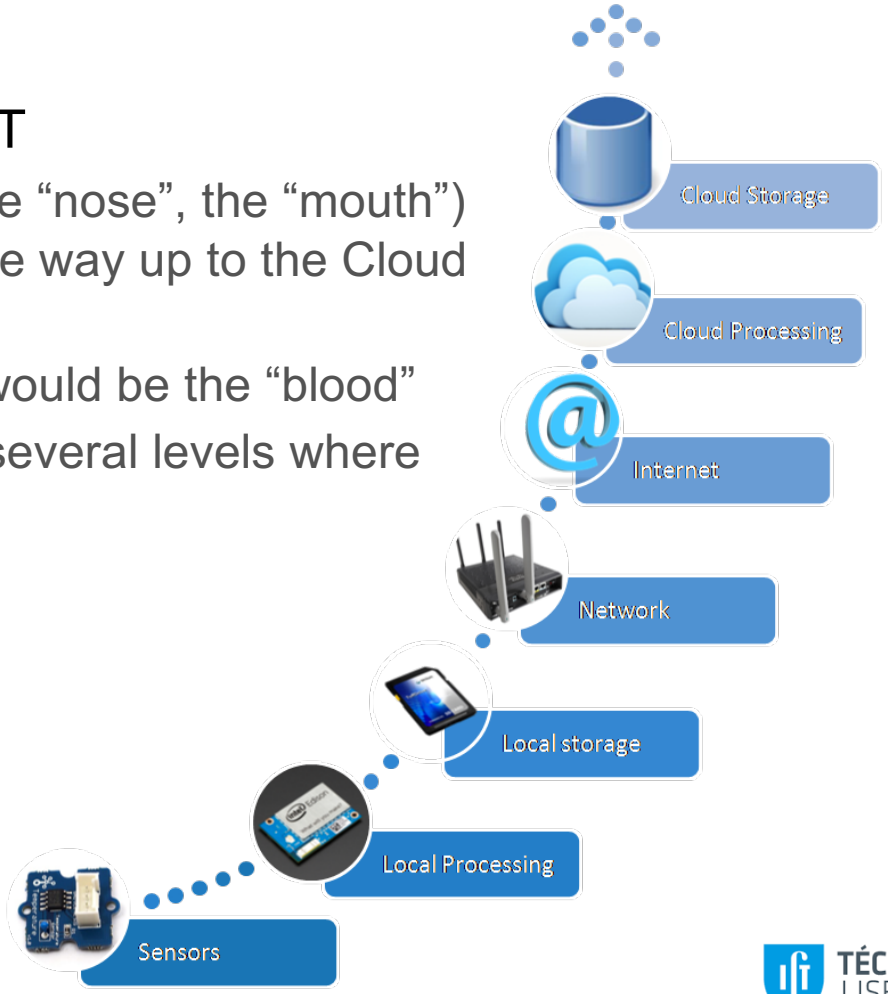
Data in Computational Intelligence for the IoT

- Data in the IoT
- What to do with Data
- Getting to Know your Data
  - Data and Attribute Types
  - Basic statistical descriptions of data
- Data Preprocessing
- Measuring Data Similarity and Dissimilarity
- Outlier Detection
- Typical Classes of Problems Involving Data

Data in the IoT

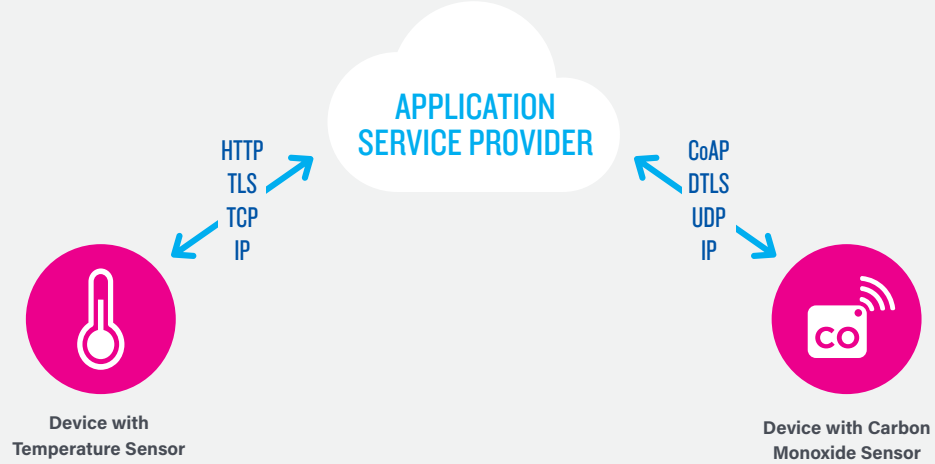
Data in the IoT

- Data is the “oxygen” of the IoT
  - It is collected in the sensors (the “nose”, the “mouth”) and ultimately transmitted all the way up to the Cloud (the “brain”)
  - The communication protocols would be the “blood”
  - Note that in the IoT “body” we several levels where the data might be processed



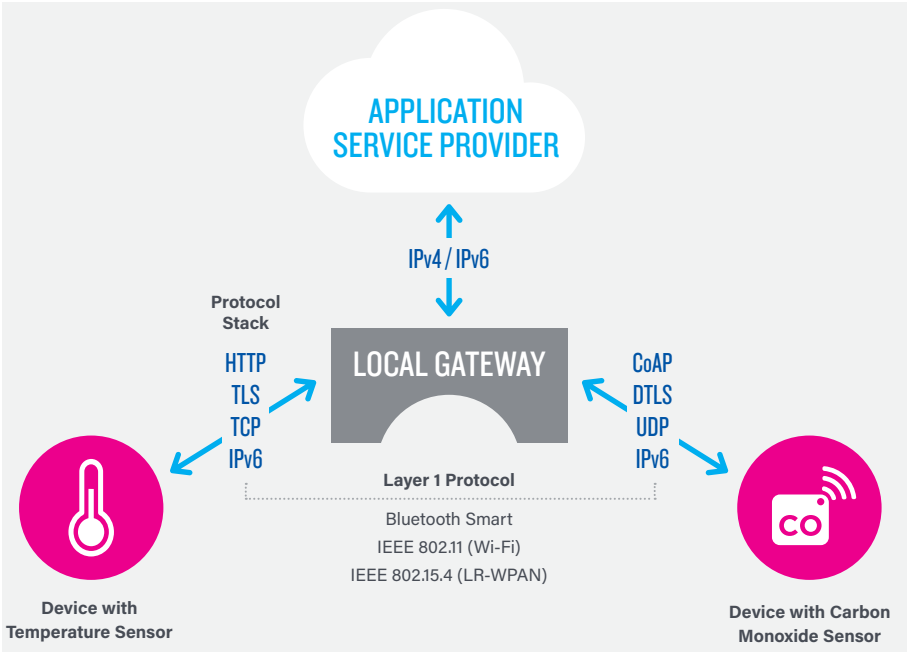
IoT Communication models

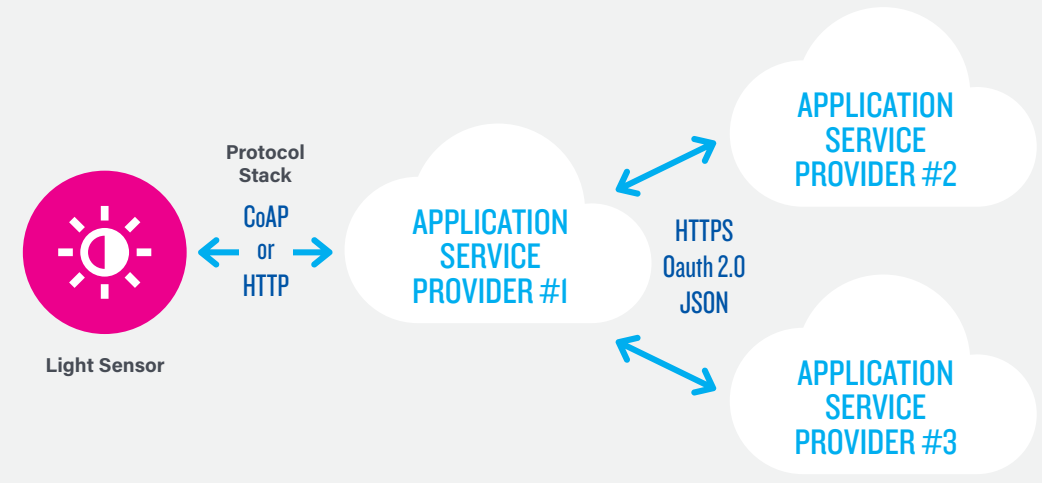
- The IoT considers 4 communication models:
  - Device-To-Device
    - Devices exchange data between them (e.g.: sensors in sensor networks; home automation devices; etc.)
    - Small data packets containing information
    - Protocols: IP; Bluetooth; ZigBee; Z-Wave
  - Device-To-Cloud
    - Devices that use Cloud services (e.g.: smartphone using Google maps for traffic information; Alexa speaker; etc.)
    - Data is transmitted to be analysed on a server
    - Protocols: IP



IoT Communication models

- The IoT considers 4 communication models (cont.):
  - Device-To-Gateway
    - The device connects to the cloud through an Application Layer Gateway (ALG) service (e.g.: fitness tracker connecting via a smartphone; hubs with ability to connect with Zigbee or Z-Wave devices; etc.)
  - Back-End Data-Sharing
    - Grant access of collected sensors' data to third parties
    - Allows data to be stored and aggregated in data silos, and then analysed by others



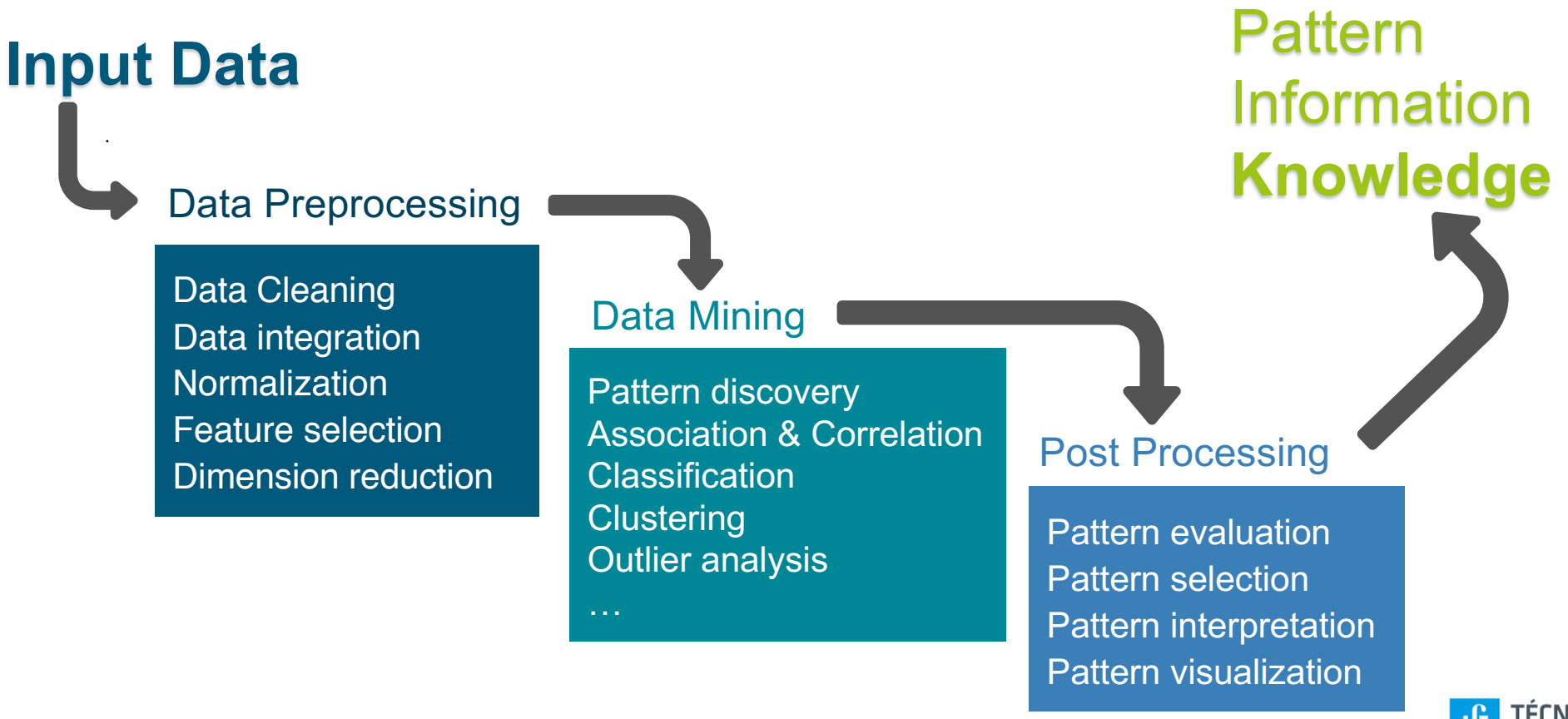


Data and what to do with it

"We are drowning in data… but starving for knowledge!"

Knowledge Discovery Process: "From data deluge to data delight"*

*João Xavier, ISR/IST



What to do with Data?

- Generalization
  - Information integration and Data Warehouse construction
    - Data cleaning, transformation, integration and multidimensional data model
  - Characterization and Discrimination
    - Generalize, summarize, and contrast data characteristics (e.g.: light network load vs. heavy network load
- Aggregation
  - Merge the information of different data points

What to do with Data? (II)

- Association and Correlation analysis
  - Frequent patterns or frequent itemsets
    - e.g., what items are frequently bought together in a supermarket
  - Association and correlation vs. causality
    - Correlation does not imply causation

What to do with Data? (III)

- Classification (and label prediction)
  - Construct models (functions) based on some training examples
  - Describe and distinguish classes for future prediction
  - Predict unknown class labels
  - Typical methods: Neural Networks, (Fuzzy) Rule based systems, Decision trees, naïve Bayesian, support vector machines, pattern-based classification, logistic regression,…
  - Typical applications: Credit card fraud detection, direct marketing, classifying stars, diseases, web-pages,…

What to do with Data? (IV)

- Clustering
  - Unsupervised learning (i.e., Class label is unknown)
  - Group data to form new categories (i.e., clusters), e.g., cluster houses to find distribution patterns
  - Principle: Maximizing intra-class similarity & minimizing interclass similarity
  - Many methods: K-Means, Fuzzy C-Means, Hierarchical, DBScan, etc.
  - Unaccountable applications

What to do with Data? (V)

- Outlier Analysis
  - Outlier: A data object that does not comply with the general behavior of the data
  - Noise or exception? — One person's garbage could be another person's treasure
  - Methods: by product of clustering or regression analysis, …
  - Useful in fraud detection, rare events analysis
- Optimization
  - e.g., find the parameters that optimize a given output using multivariate data

What to do with Data? (VI) – Time and Ordering

- Sequence, trend and evolution analysis
  - Trend, time-series, and deviation analysis
    - e.g., regression, forecasting, etc.
  - Sequential pattern mining
    - e.g., first buy digital camera, then buy large SD memory cards
  - Periodicity analysis
  - Motifs and biological sequence analysis
    - Approximate and consecutive motifs
  - Similarity-based analysis
- Mining data streams
  - Ordered, time-varying, potentially infinite, data streams

Getting to know your data

Data and Attribute Types

Data Types

- Raw data streams and sensor data
- Time series data, temporal data, sequence data (including bio-sequences)
- Spatial data and spatiotemporal data
- Multimedia
- Text
- The World-Wide Web
- Structure data, graphs, social networks and multi-linked data

Attribute (Feature) Types

- Nominal: categories, states, or “names of things”
  - e.g., *Hair_color = {auburn, black, blond, brown, grey, red, white}*
  - e.g., marital status, occupation, ID numbers, zip code
- Binary / Boolean
  - Nominal attribute with only 2 states (0 and 1)
  - Symmetric binary: both outcomes equally important (e.g., gender)
  - Asymmetric binary: outcomes not equally important (e.g., medical test positive vs. negative)
    - Convention: assign 1 to most important outcome (e.g., HIV positive)
- Ordinal
  - Values have a meaningful order (ranking) but magnitude between successive values is not known (e.g., Size = {small, medium, large})

Attribute Types (II)

- Numerical
  - Quantity (integer or real-valued)
  - Interval
    - Measured on a scale of equal-sized units
    - Values have order
      - E.g., temperature in C˚or F˚, calendar dates
    - No true zero-point
  - Ratio
    - Inherent zero-point
    - We can speak of values as being an order of magnitude larger than the unit of measurement (10 K˚ is twice as high as 5 K˚).
    - e.g., temperature in Kelvin, length, counts, monetary quantities

Attribute Types (III) – Continuous vs. Discrete

- Discrete Attributes
  - Have a finite or countably infinite set of values
    - e.g., zip codes, profession, or the set of words in a collection of documents
  - Typically represented as integer variables
  - Note: Binary attributes are a special case of discrete attributes
- Continuous Attributes
  - Have real numbers as attribute value
    - e.g., temperature, height, or weight
  - Practically, real values can only be measured and represented using a finite number of digits
  - Continuous attributes are typically represented as floating-point variables

Getting to know your data

Basic statistical descriptions of data

Basic Statistical Descriptions of Data

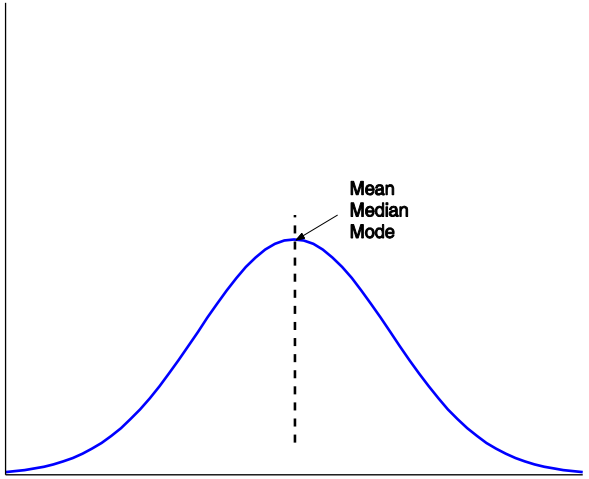
- Motivation
  - To better understand the data: central tendency, variation and spread
- Data dispersion characteristics
  - median, max, min, quantiles, outliers, variance, etc.

Measuring the Central Tendency

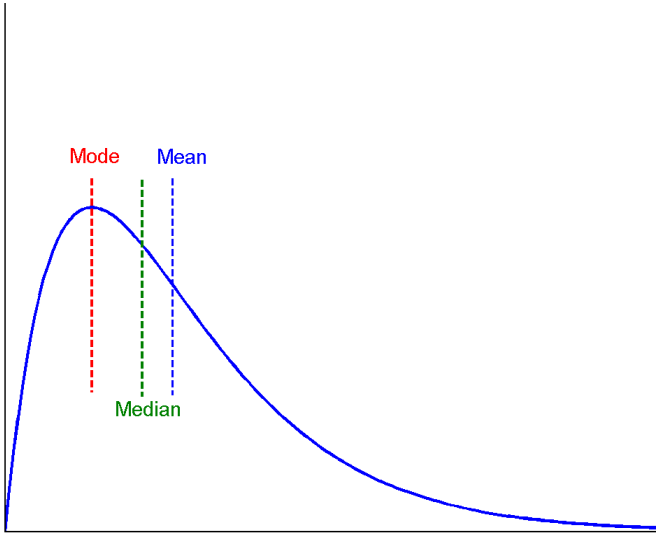
- Mean (algebraic measure): $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
  - Weighted arithmetic mean: $\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$
  - Trimmed mean: remove extreme values
- Median:
  - Middle value if odd number of values, or average of the middle two values otherwise
  - Estimated by interpolation (for *grouped data*): $median = L_1 + (\frac{n/2 - (\sum freq)l}{freq_{median}})width$
- Mode:
  - Value that occurs most frequently in the data
- Empirical formula: $mean - mode = 3 \times (mean - median)$

Symmetric vs. Skewed Data

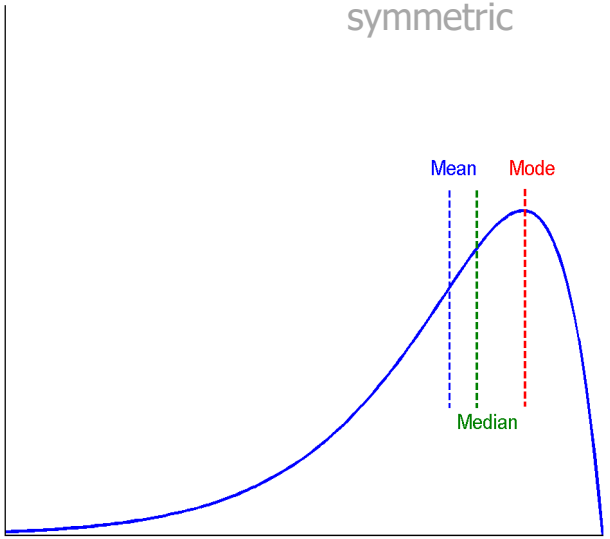
- Median, mean and mode of symmetric, positively and negatively skewed data:



symmetric



positively skewed



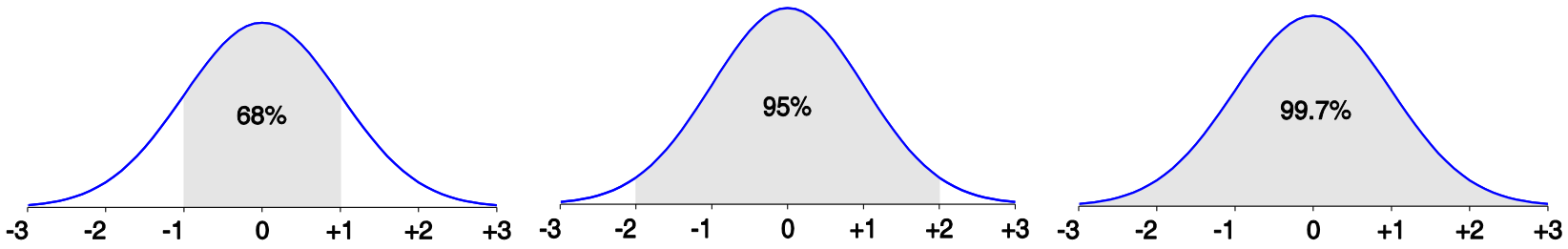
negatively skewed

Measuring the Dispersion of Data

- Quartiles, outliers and boxplots
  - Quartiles: Q_1 (25th percentile), Q_3 (75th percentile)
  - Inter-quartile range: IQR = $Q_3 - Q_1$
  - Five number summary: min, Q_1, median, Q_3, max
  - Boxplot: ends of the box are the quartiles; median is marked; add whiskers, and plot outliers individually
  - Outlier: usually, a value higher/lower than 1.5 x IQR
- Variance and standard deviation (*sample: s, population: σ)*
  - Variance: $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}[\sum_{i=1}^{n} x_i^2 - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2]$ $\sigma^2 = \frac{1}{N}\sum_{i=1}^{n}(x_i - \mu)^2 = \frac{1}{N}\sum_{i=1}^{n} x_i^2 - \mu^2$
  - Standard deviation: *s (or σ)* is the square root of variance s^2 *(or* σ^2*)*

Properties of Normal Distribution Curve

- The normal (distribution) curve
  - From μ–σ to μ+σ: contains about 68% of the measurements (μ: mean, σ: standard deviation)
  - From μ–2σ to μ+2σ: contains about 95% of it
  - From μ–3σ to μ+3σ: contains about 99.7% of it

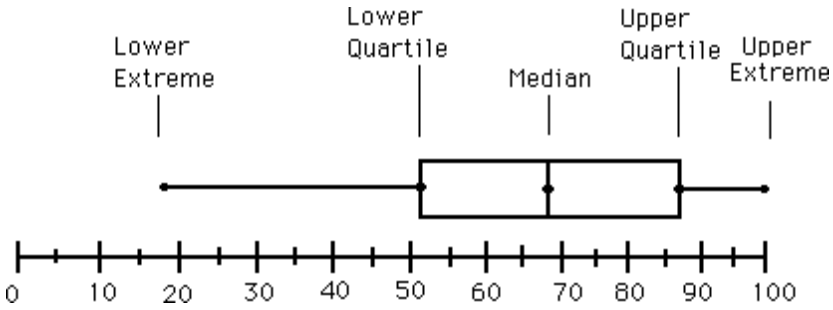


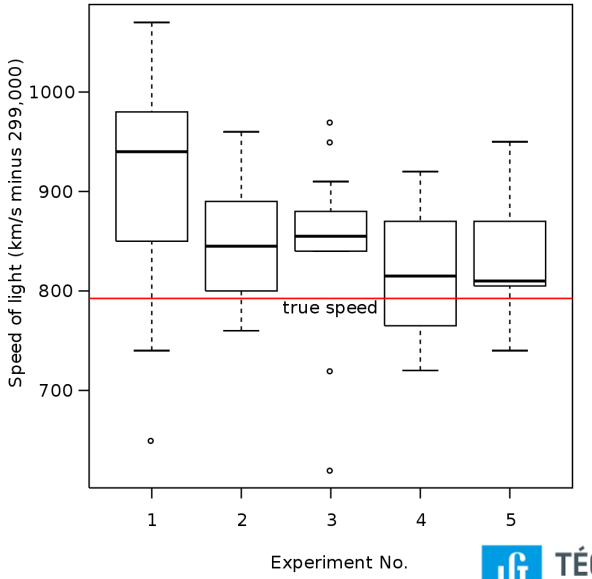
Graphic Displays of Basic Statistical Descriptions

- Boxplot: graphic display of five-number summary
- Histogram: *x*-axis are values, *y*-axis repres. frequencies
- Scatter plot: each pair of values is a pair of coordinates and plotted as points in the plane
- Quantile-quantile (q-q) plot: graphs the quantiles of one univariant distribution against the corresponding quantiles of another

Boxplot Analysis

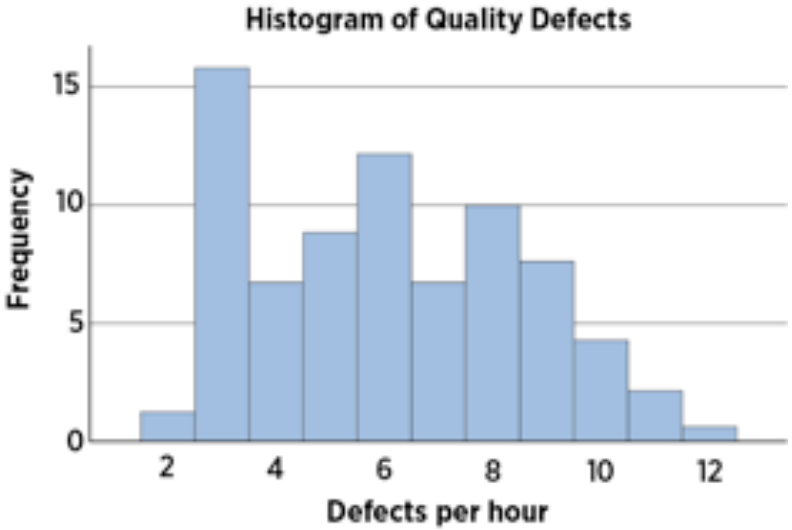
- **Five-number summary** of a distribution
  - Minimum, Q1, Median, Q3, Maximum
- **Boxplot**
  - Data is represented with a box
  - The ends of the box are at the first and third quartiles, i.e., the height of the box is IQR
  - The median is marked by a line within the box
  - Whiskers: two lines outside the box extended to Minimum and Maximum
  - Outliers: points beyond a specified outlier threshold, plotted individually





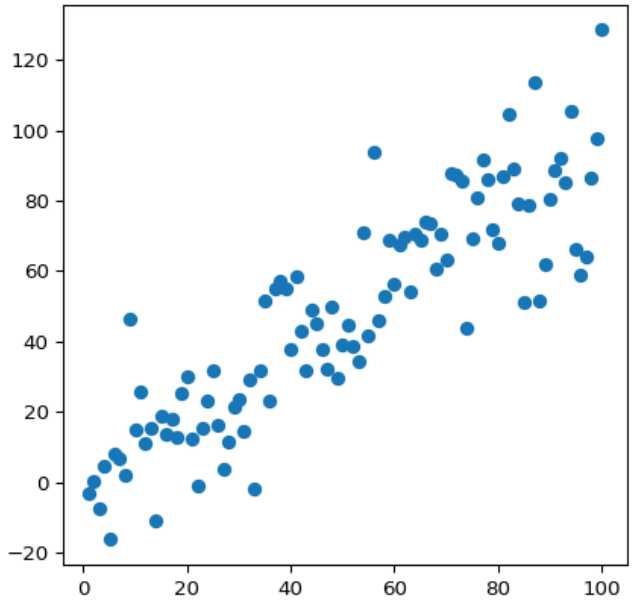
Histogram Analysis

- Histogram: Graph display of tabulated frequencies, shown as bars
- It shows what proportion of cases fall into each of several categories
- The categories are usually specified as non-overlapping intervals of some variable. The categories (bars) must be adjacent

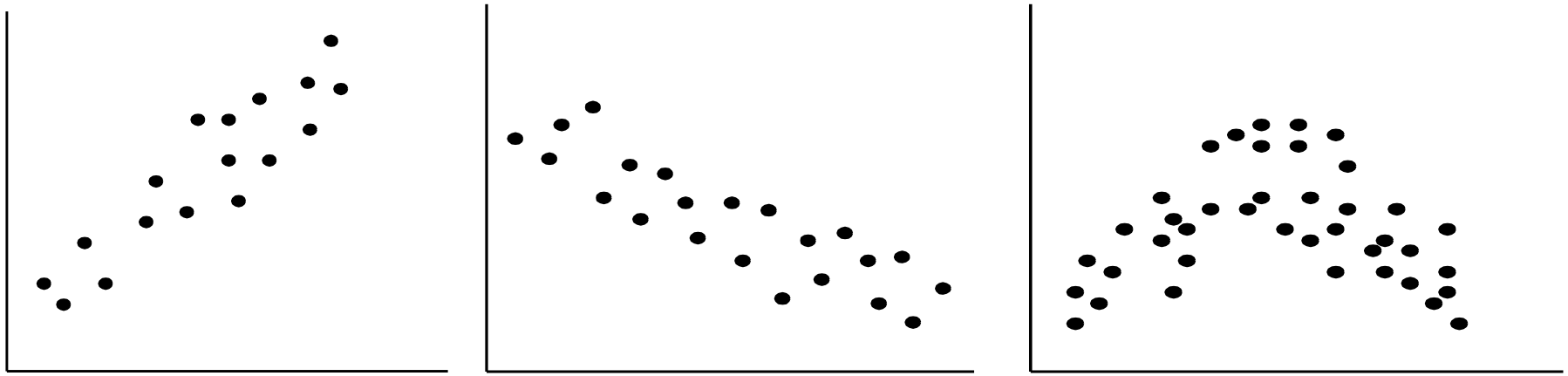


Scatter plot

- Provides a first look at bivariate data to see clusters of points, outliers, etc.
- Each pair of values is treated as a pair of coordinates and plotted as points in the plane

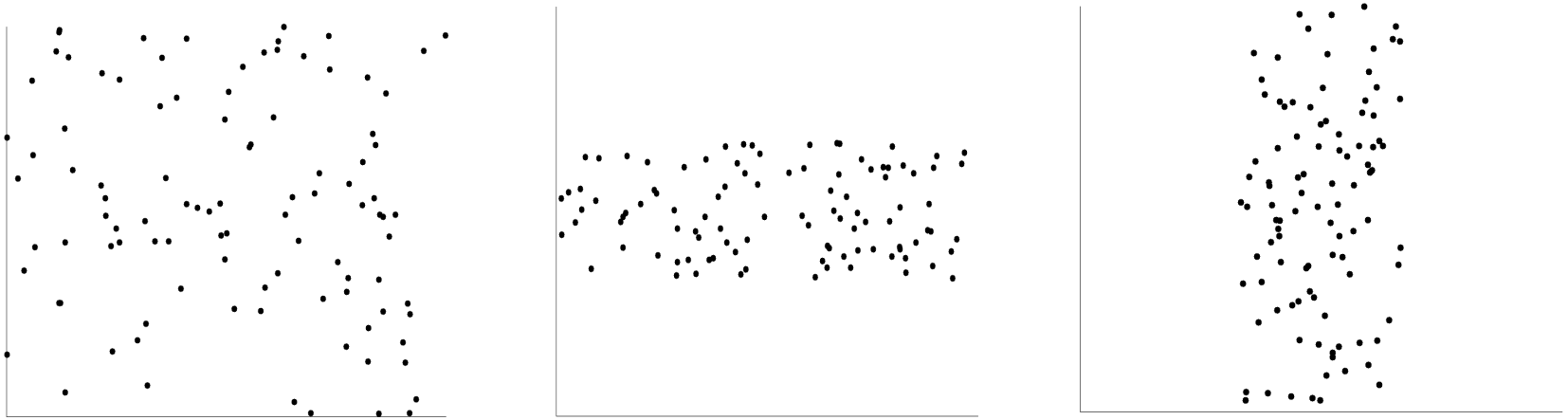


Positively and Negatively Correlated Data



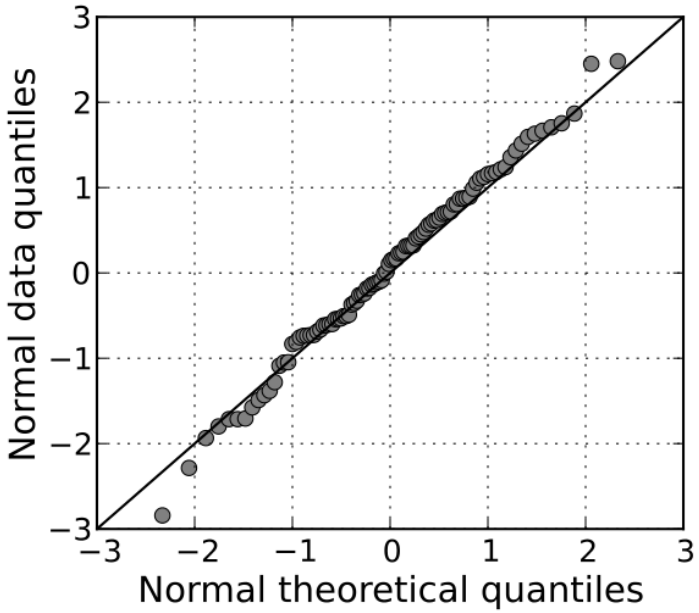
- Positively correlated data
- Negatively correlated data
- Positively correlated on left side, and negatively correlated on right side

Uncorrelated Data



Quantile-Quantile (Q-Q) Plot

- A Q–Q plot is a plot of the quantiles of two distributions against each other, or a plot based on estimates of the quantiles. The pattern of points in the plot is used to compare the two distributions.
  - Quantile information: For data x_i sorted in increasing order, f_i indicates that approximately $100f_i\%$ of the data are below or equal to the value x_i



Data Preprocessing

Data Preprocessing

- **Data cleaning**
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- **Data integration**
  - Integration of multiple datasets
- **Data reduction**
  - Dimensionality reduction; Numerosity reduction; Data compression
- **Data transformation and data discretization**
  - Normalization, Aggregation, Discretization

Data Preprocessing

Data Cleaning

Data Cleaning

- Data in the Real World is dirty (lots of potentially incorrect data due to faulty instruments, human or computer error, transmission error, etc.):
  - Incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., *Occupation*=“ ” (missing data)
  - Noisy: containing noise, errors, or outliers
    - e.g., *Salary*=“−10” (an error)
  - Inconsistent: containing discrepancies in codes or names, e.g.,
    - *Age*=“42”, *Birthday*=“03/07/2010”
    - Rating was “1, 2, 3” and now “A, B, C”
    - Discrepancy between duplicate records
  - Intentional (e.g., *disguised missing* data)
    - e.g., Jan. 1 as everyone’s birthday?

Incomplete / Missing Data

- Data is not always available
  - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - equipment malfunction
  - inconsistent with other recorded data and thus deleted
  - data not entered due to misunderstanding
  - certain data may not be considered important at the time of entry
  - not register history or changes of the data
- Missing data may need to be inferred

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (when doing classification) – not effective when the % of missing values per attribute varies considerably
- Fill in the missing value manually: tedious + infeasible?
- Fill it in automatically with:
  - a global constant : e.g., “unknown”, a new class?!
  - the attribute mean
  - the attribute mean for all samples belonging to the same class: smarter
  - the most probable value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to:
  - faulty data collection instruments;
  - data entry problems;
  - data transmission problems;
  - technology limitation;
  - inconsistency in naming convention.
- Other data problems that require data cleaning:
  - duplicate records;
  - incomplete data;
  - inconsistent data.

How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by human (e.g., deal with possible outliers)

Data Preprocessing

Data Integration

Data Integration

- Combine data from different sources
  - Need to detect and resolve data value conflicts
  - Entity identification problems
    - e.g., Bill Clinton vs. William Clinton
- Attribute values from different sources are different due to:
  - Different representations (e.g., Unix encoding vs. Windows encoding)
  - Different scales (e.g., grams vs. kg)
  - Different units (e.g., Metric units vs. Imperial units)

Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases
  - *Object identification*: The same attribute or object may have different names in different databases
  - *Derivable data:* One attribute may be a "derived" attribute in another database, e.g., price+tax vs. price and tax
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality
- Redundant attributes may be able to be detected by *correlation analysis (Chi-square, Pearson's)* and *covariance analysis*

Data Preprocessing

Data Reduction

Data Reduction

- Data reduction: Obtain a reduced representation of the data set that is much smaller in volume but produces the same (or almost the same) analytical results
- Why data reduction? Complex data analysis may take a very long time to run on very large data sets .
- Data reduction strategies:
  - Dimensionality reduction, e.g., remove unimportant attributes
  - Numerosity reduction (Data reduction)
  - Data compression

Dimensionality Reduction

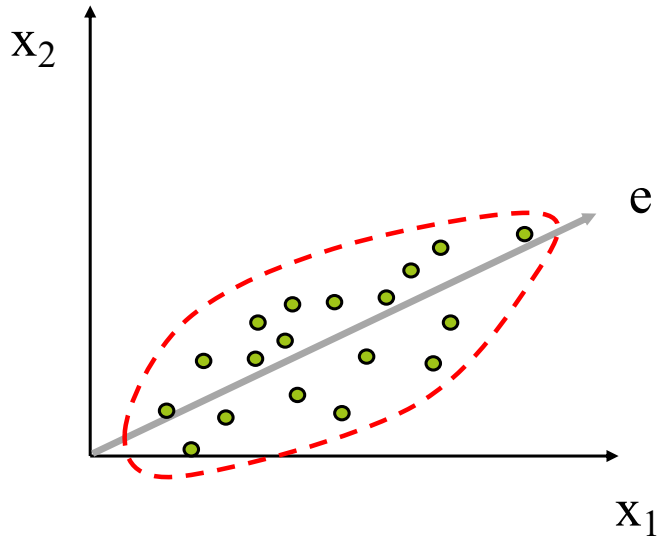
- Curse of dimensionality
  - When dimensionality increases, data becomes increasingly sparse
  - Density and distance between points, which is critical to clustering, outlier analysis, becomes less meaningful
  - The possible combinations of subspaces will grow exponentially
- Dimensionality reduction
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization

Dimensionality Reduction Techniques

- Fourier and Wavelet transforms
  - Mapping data into a new space
- Principal Component Analysis (PCA)
- Supervised and nonlinear techniques (e.g., feature selection)

Dimensionality Reduction Techniques: PCA

- Principal Component Analysis
  - Find a projection that captures the largest amount of variation in data
  - Original data are projected onto a much smaller space, resulting in dimensionality reduction. We find the eigenvectors of the covariance matrix, and these eigenvectors define the new space



Dimensionality Reduction Techniques: PCA Steps

- Given *N* data vectors from n-dimensions, find $k \leq n$ orthogonal vectors (principal components) that can be best used to represent data
  - Normalize input data (each attribute must fall within the same range)
  - Compute *k* orthonormal (unit) vectors, i.e., principal components
    - Each input data (vector) is a linear combination of the *k* principal component vectors
  - Sort the principal components in order of decreasing "significance" or strength
  - Reduce the data by eliminating the weaker components, i.e., those with lowest variance
    - It is possible to reconstruct a good approximation of the original data using the strongest principal components,
- Works for numeric data only

Dimensionality Reduction Techniques: Feature Selection

- Redundant attributes (features)
  - Attributes that duplicate much or all of the information contained in one or more other attributes
  - E.g., purchase price of a product and the amount of sales tax paid
- Irrelevant attributes (features)
  - Contain no information that is useful for the data mining task at hand
  - E.g., students' ID is often irrelevant to the task of predicting students' grades
- There are 2^d possible attribute combinations of d attributes…
  - Use heuristic approaches to find most relevant subset of attributes for a given problem (using an evaluation measure)
  - Many alternative methods

Dimensionality Reduction Techniques: Attribute Creation (Feature Generation)

- Create new attributes (features) that can capture the important information in a data set more effectively than the original ones
- Attribute extraction
  - Domain-specific
- Mapping data to new space (see data reduction)
  - E.g., Fourier transformation, wavelet transformation, manifold approaches
- Attribute construction
  - Combining features
  - Data discretization

Numerosity Reduction (Data Reduction)

- Reduce data volume by choosing alternative, *smaller forms* of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - E.g.: Linear regression; Multiple Regression; Log-Linear model, etc.
- Non-parametric methods
  - Do not assume models
  - Major families: histograms, clustering, sampling,…

Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms such as Lempel-Ziv, Huffman, etc. (zip, rar, …)
  - Typically lossless, but only limited manipulation is possible without expansion
- Audio/video compression
  - Lossy (Dolby Digital, DTS, mp3, AAC, etc.) vs. non-lossy (FLAC, DTS-HD, H-264, Quicktime, etc.)
  - It is often possible to reconstruct small fragments of without reconstructing the whole
- Time sequence (non-audio)
  - Typically short and vary slowly with time

Note: Dimensionality and numerosity reduction may also be considered as forms of data compression

Data Preprocessing

Data Transformation

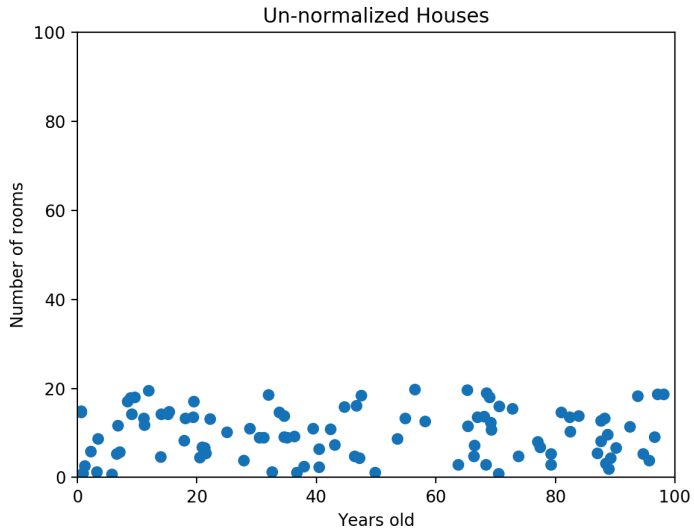
Data Transformation

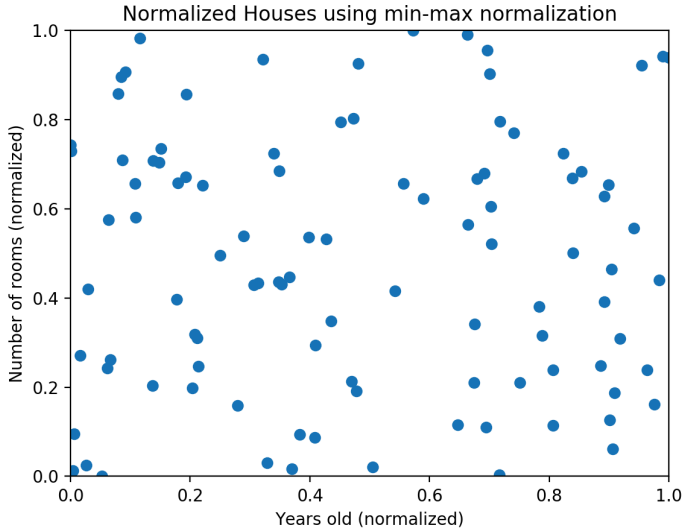
Mapping the entire set of values of a given attribute to a new set of replacement values

- Smoothing: Remove noise from data
- Attribute/feature generation
  - New attributes constructed from the given ones
- Normalization: Scale values to fall within a smaller, specified range
  - min-max normalization
  - z-score normalization
- Discretization
- Aggregation: Summarization

Normalizing and Standardizing Numeric Data

- Many algorithms attempt to find trends in the data by comparing features of data points. However, issues might arise when the features are on different scales.
- Normalization helps avoiding this issue:



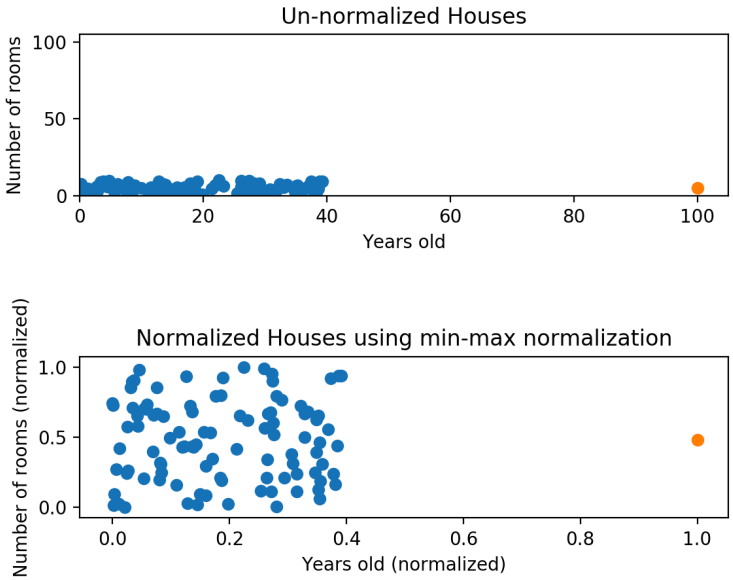


Min-Max normalization

- Min-Max normalization:

$$new_x = \frac{x - min}{max - min}$$

- Min-Max does not handle outliers well. Example:



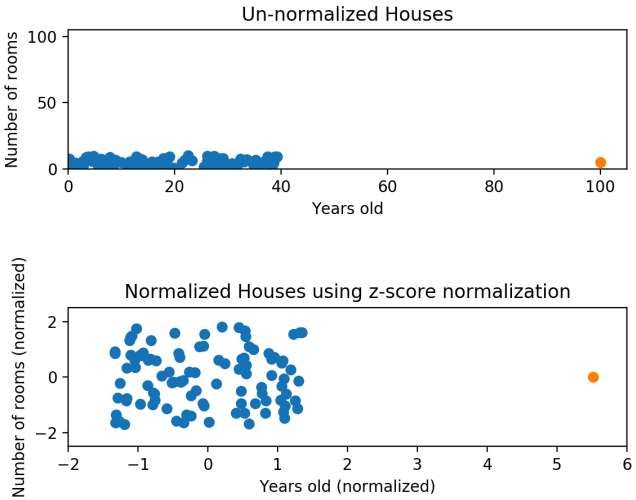
Z-Score standardization

- Z-score: $$z = \frac{x - \mu}{\sigma}$$
  - μ: mean of the population, σ: standard deviation
  - Z-score is the distance between the raw score and the population mean in units of the standard deviation
  - The Z-score is negative when the raw score is below the mean, positive when above
  - E.g. x = 73600, μ = 54000, σ = 16000:

$$Z_{score} = \frac{73600 - 54000}{16000} = 1.225$$

Z-Score standardization (II)

- The Z-score handles outliers better:



- Note that despite the “squished” look, all data points are now on a similar feature scale: all axis have most data points in similar ranges

Discretization

Divide the range of a continuous attribute into intervals

- Interval labels can then be used to replace actual data values
- Side benefit: reduce data size

- Supervised vs. unsupervised
- Split (top-down) vs. merge (bottom-up)
- Can be performed recursively
- Prepares data for further tasks, e.g., classification

Data Discretization Methods

- Binning
  - Top-down split, unsupervised
- Histogram analysis
  - Top-down split, unsupervised
- Clustering analysis
  - Top-down split or bottom-up merge, unsupervised
- Decision-tree / classification analysis
  - Top-down split, supervised
- Correlation (e.g., χ^2) analysis
  - Bottom-up merge, unsupervised

Note: All methods can be applied recursively

Simple Discretization: Binning

- Equal-width (distance) partitioning
  - Divides the range into N intervals of equal size: uniform grid
  - if A and B are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$
  - The most straightforward, but outliers may dominate presentation
  - Skewed data is not handled well
- Equal-depth (frequency) partitioning
  - Divides the range into N intervals, each containing approximately same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky

Aggregation

- An aggregation function is a function where the values of multiple objects are grouped together to form a single summary value
  - Useful to group interval data
- Examples:
  - Average (i.e., arithmetic mean), Count, Maximum, Median, Minimum, Mode, Range, Sum, STDev, etc.
  - Ordered Weighted Averaging (OWA): a parameterized class of mean type aggregation operators that generalizes all the above
  - Choquet Integral

Data Similarity and Dissimilarity

Similarity and Dissimilarity

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

Data Matrix and Dissimilarity Matrix

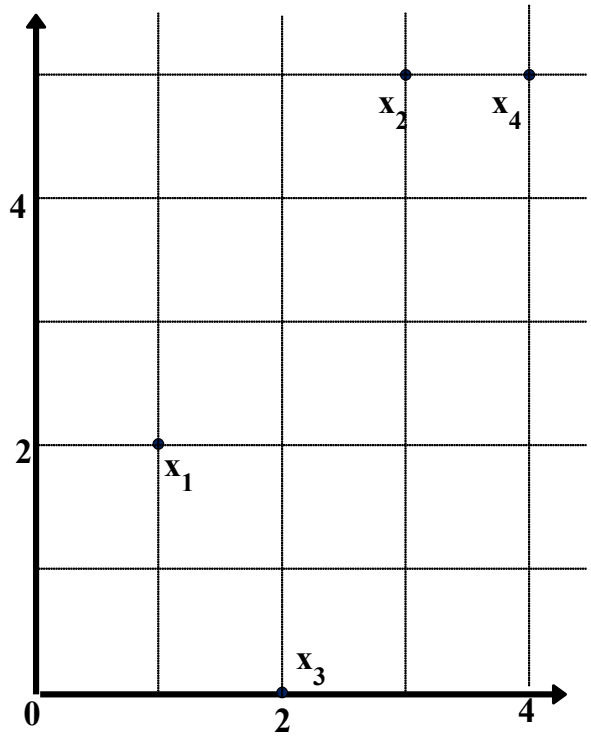
- Data matrix
  - n data points with p dimensions

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

- Dissimilarity matrix
  - n data points, but registers only the distance
  - A triangular matrix

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Data Matrix and Dissimilarity Matrix: Example



Data matrix

point	attribute1	attribute2
x1	1	2
x2	3	5
x3	2	0
x4	4	5

Dissimilarity matrix
(with Euclidean Distance)

	x1	*x2*	*x3*	*x4*
x1	0			
x2	3.61	0		
x3	5.1	5.1	0	
x4	4.24	1	5.39	0

Proximity Measure for Nominal Attributes

- Can take 2 or more states, e.g. “red, yellow, blue, green” (generalization of a binary attribute)
- Method 1: Simple matching
  - *m*: # of matches, *p*: total # of variables

$$d(i,j)=\frac{p-m}{p}$$

- Method 2: Use a large number of binary attributes
  - Create a new binary attribute for each of the *M* nominal states (see next slide)

Proximity Measure for Binary Attributes

- Contingency table (for binary data):

Object *i* \ Object *j*	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

- Distance measure for symmetric binary variables:

$$d(i,j) = \frac{r+s}{q+r+s+t}$$

- Distance measure for asymmetric binary variables:

$$d(i,j) = \frac{r+s}{q+r+s}$$

- Jaccard coefficient for *asymmetric* binary variables (*similarity* measure):

$$sim_{Jaccard}(i,j) = \frac{q}{q+r+s}$$

  - For binary attributes Jaccard = coherence

Dissimilarity between Binary Variables: Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- Gender is a symmetric attribute
- The remaining attributes are asymmetric binary (ignored if both are 0)
- Let the values Y and P be 1, and the value N be 0

Object *i* \ Object *j*	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

$$d(i,j) = \frac{r+s}{q+r+s}$$

$$d(Jack, Mary) = \frac{1+1}{2+1+1} = 0.5$$

$$d(Jack, Jim) = \frac{1+1}{2+1+1} = 0.5$$

$$d(Jim, Mary) = \frac{2+2}{1+2+2} = 0.8$$

Distance on Numeric Data: Minkowski Distance

- Minkowski distance: A popular distance measure

$$d(i, j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \cdots + |x_{ip} - x_{jp}|^h}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects, and h is the order (the distance so defined is also called L-h norm)

- Properties
  - $d(i, j) > 0$ if $i \neq j$, and $d(i, i) = 0$ (Positive definiteness)
  - $d(i, j) = d(j, i)$ (Symmetry)
  - $d(i, j) \leq d(i, k) + d(k, j)$ (Triangle Inequality)
- A distance that satisfies these properties is a metric

Special Cases of Minkowski Distance

- $h = 1$: Manhattan (city block, L_1 norm) distance
  - E.g., the Hamming distance: the number of bits that are different between two binary vectors

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$

- $h = 2$: (L_2 norm) Euclidean distance

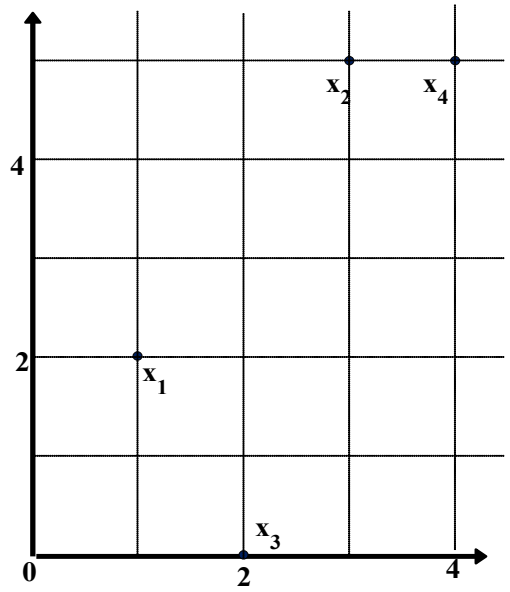
$$d(i,j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \ldots + |x_{ip} - x_{jp}|^2)}$$

- $h \to \infty$: "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component (attribute) of the vectors

$$d(i,j) = \lim_{h \to \infty} \left(\sum_{f=1}^{p} |x_{if} - x_{jf}|^h \right)^{\frac{1}{h}} = \max_{f}^{p} |x_{if} - x_{jf}|$$

Minkovski Distance: Example

point	attribute1	attribute2
x1	1	2
x2	3	5
x3	2	0
x4	4	5



Dissimilarity matrices

Manhattan (L1)

L	x1	x2	x3	x4
x1	0			
x2	5	0		
x3	3	6	0	
x4	6	1	7	0

Euclidean (L2)

L2	x1	x2	x3	x4
x1	0			
x2	3.61	0		
x3	2.24	5.1	0	
x4	4.24	1	5.39	0

Supremum

L_∞	x1	x2	x3	x4
x1	0			
x2	3	0		
x3	2	5	0	
x4	3	1	5	0

Similarity between Ordinal Variables

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled
  - replace x_{if} by their rank $r_{if} \in \{1,...,M_f\}$
  - normalize the range of each variable onto [0, 1] by replacing i-th object in the f-th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - compute the dissimilarity using methods for interval-scaled variables

Attributes of Mixed Type

- A database may contain all attribute types
  - Nominal, symmetric binary, asymmetric binary, numeric, ordinal
- One may use a weighted formula to combine their effects

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

  - f is binary or nominal: $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$, or $d_{ij}^{(f)} = 1$ otherwise
  - f is numeric: use the normalized distance
  - f is ordinal: compute ranks r_{if} and treat z_{if} as interval-scaled

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

Vector Similarity

- Many data objects can be represented as vectors
  - A **document** can be represented by thousands of attributes, each recording the *frequency* of a particular word (such as keywords) or

Document	*team*	*coach*	*hockey*	*baseball*	*soccer*	*penalty*	*score*	*win*	*loss*	*season*
Document1	5	0	3	0	2	0	0	2	0	0
Document2	3	0	2	0	1	1	0	1	0	1
Document3	0	7	0	2	1	0	0	3	0	0
Document4	0	1	0	0	1	2	2	0	3	0

  - Other vector objects: gene features in micro-arrays; list of destinations in a user phone calls; accessed webpages; etc.
- Applications: information retrieval, biologic taxonomy, gene feature mapping, user identification, etc.

Vector Similarity (II)

- Cosine Similarity:
  - If A and B are two vectors of real and greater than zero attributes, e.g., term frequency vectors, then:

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

  - Ex: Find the **similarity** between documents 1 and 2:

    A = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)

    B = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)

    A•B = 5*3+0*0+3*2+0*0+2*1+0*1+0*1+2*1+0*0+0*1 = 25

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

    $||B|| = (3*3+0*0+2*2+0*0+1*1+1*1+0*0+1*1+0*0+1*1)^{0.5} = (17)^{0.5} = 4.12$

    cos(A, B) = 0.94

Vector Similarity (III)

- Jaccard Similarity and Distance:
  - If X and Y are two vectors of real attributes greater or equal than zero, the Jaccard similarity coefficient is given by:

$$J_W(A,B) = \frac{\sum_{i=1}^{n} \min(A_i, B_i)}{\sum_{i=1}^{n} \max(A_i, B_i)}$$

  - and the Jaccard distance is:

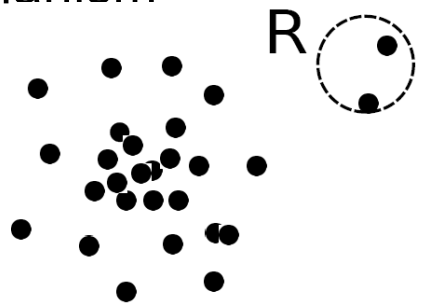
$$d_{J_W}(A,B) = 1 - J_W(A,B)$$

- Many other methods:
  - E.g., Dice, Fuzzy Fingerprints, etc.

Outlier Detection

What Are Outliers?

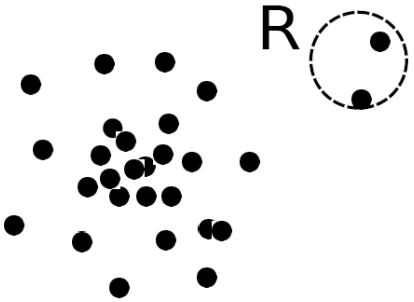
- Outlier: A data object that deviates significantly from the normal objects as if it were generated by a different mechanism
  - Is it noise?
  - Is it an error?
  - Is it an extreme point?
- What should I do? Remove it? Use it?
  - Reviews of a hotel: Probably remove extreme positive and negative points
  - Stock market trading, day value equal to zero: Remove it, probably no trade that day
  - Ebola detection, maybe the extreme value is what you are looking for: Do not remove



What Are Outliers (II)

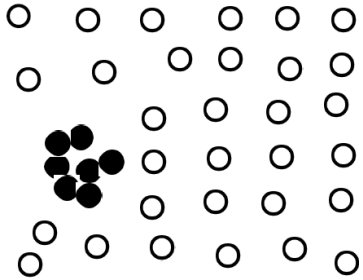
- Outliers are interesting: They violate the mechanism that generates the normal data
  - Examples: Unusual credit card purchase; Cristiano Ronaldo; Abnormally large data traffic due to some popular event; Anomalies in computer networks, etc.
- Outliers are different from the noise data:
  - Noise is random error or variance in a measured variable
  - Noise should be removed before outlier detection
- Applications: Fraud detection; Customer Segmentation; Medical Analysis; etc.
  - Novelty detection vs. outlier detection: early stage, outlier; but later merged into the model

Types of Outliers (I)



- Global outlier (or point anomaly)
  - Object is O_g if it significantly deviates from the rest of the data set
    - E.g. Intrusion detection in computer networks
  - Issue: Find an appropriate measurement of deviation
- Contextual outlier (or *conditional outlier*)
  - Object is O_c if it deviates significantly based on a selected context
    - E.g. 35^{o}C in Lisboa: Outlier? A: Depends on if it's Summer or Winter
  - Attributes of data objects should be divided into two groups
    - Contextual attributes: defines the context. E.g., time & location
    - Behavioral attributes: characteristics of the object used in outlier evaluation. E.g., temperature
  - Can be viewed as a generalization of *local outliers* whose density significantly deviates from its local area
  - Issue: How to define or formulate meaningful context?

Types of Outliers (II)



- Collective Outliers
  - A subset of data objects *collectively* deviate significantly from the whole data set, even if the individual data objects may not be outliers
    - E.g., *intrusion detection*: When a number of computers keep sending denial-of-service packages to each other
  - Detection of collective outliers:
    - Consider not only behavior of individual objects, but also of groups of objects
    - Need to have the background knowledge on the relationship among data objects, such as a distance or similarity measure on objects
- A data set may have multiple types of outlier
- One object may belong to more than one type of outlier

Challenges of Outlier Detection

- Modeling normal objects and outliers properly
  - Hard to enumerate all possible normal behaviors in an application
  - The border between normal and outlier objects is often a gray area
- Application-specific outlier detection
  - Choice of distance measure among objects and the model of relationship among objects are often application-dependent
    - E.g., clinic data: a small deviation could be an outlier; marketing analysis: larger fluctuations
- Handling noise
  - Noise may distort the normal objects and blur the distinction between normal objects and outliers. It may help hide outliers and reduce the effectiveness of outlier detection

Challenges of Outlier Detection (II)

- Errors
  - How to distinguish an error from an outlier?
- High Dimensional Outliers
  - Avoid distance measures
  - Use heuristics that do not deteriorate in high dimensional data
- Understandability
  - Understand why these are outliers: justification of the detection
  - Specify the degree of an outlier: the unlikelihood of the object being generated by a normal mechanism

Outlier Detection

- Based on *assumptions about normal data*:
  - Statistical (model-based) methods
    - Assume that the normal data follow some statistical (stochastic) model
      - Data not following the model are outliers
    - Many alternatives: Parametrical, Non Parametrical…
    - E.g. Use quartiles and st-dev units of distance
  - Proximity-based
    - An object is an outlier if the nearest neighbors of the object are far away, i.e., the proximity of the object significantly deviates from the proximity of most of the other objects in the same data set
    - Distance based vs. Density based
  - Clustering-based methods
- Based on user-*labeled* examples of outliers (supervised)
  - Approach Outlier Detection as a Classification problem

Outlier Detection: Supervised Methods

- Modeling outlier detection as a classification problem
  - Samples examined by domain experts used for training & testing
- Methods for Learning a classifier for outlier detection effectively:
  1. Model normal objects & report those not matching the model as outliers;
  2. Model outliers and treat those not matching the model as normal.
- Challenges
  - Imbalanced classes, i.e., outliers are rare
    - Boost the outlier class and make up some artificial outliers
  - Catch as many outliers as possible
    - Recall is more important than accuracy (i.e., don't mind mislabeling normal objects as outliers)

Outlier Detection: Unsupervised Methods

- Assume the normal objects are somewhat “clustered" into multiple groups, each having some distinct features
  - An outlier is expected to be far away from any groups of normal objects
  - Weakness: Cannot detect collective outliers effectively
  - Normal objects may not share any strong patterns, but the collective outliers may share high similarity in a small area
    - E.g. In some intrusion or virus detection, normal activities are diverse
- Many clustering methods can be adapted for unsupervised methods
  - Find clusters, then outliers (points that do not belong to any clusters)
  - Problem 1: Hard to distinguish noise from outliers
  - Problem 2: Cost after the clustering (but far less outliers than normal objects)
- Unsupervised methods may have a high false positive rate but still miss many real outliers!

Typical Classes of Problems Involving Data

Classification

- Classification is the problem of identifying to which of a set of classes (categories, labels) a new observation (input) belongs
- Classes:
  - Discrete
    - Binary vs. Multiclass
  - Examples:
    - {“Spam”, “Not Spam”}, binary;
    - {“Cancer”, “Not Cancer”}, binary;
    - {“News”, “Entertainment”, “Sports”, “Economy”, …}, multi-class;
    - {“0”, “1”, “2”, “3”, ”4”, ”5”, “6”, “7”, “8”, “9”} (digit recognition), multi-class;
    - {“Resting”, “Walking”, “Running”, “Cycling”, …}, multi-class;
    - {“Stop”, “Speed Limit”, “No U-Turn”, “No Entry”, …}, multi-class;

Classification (II)

- The inputs are often called features:
  - Binary, Categorical, Numerical, Continuous, Discrete, etc. (see section Data and Attribute Types)
  - Examples:
    - Spam: #capitalized words in subject; #URLs; #words containing letters and numbers; etc.
    - Image: Edges; Corners; Blobs; Ridges; etc.
    - Human Activity Detection: 3D accelerometer; 3D gyroscope; magnetoscope; HR; etc.

Classification (III)

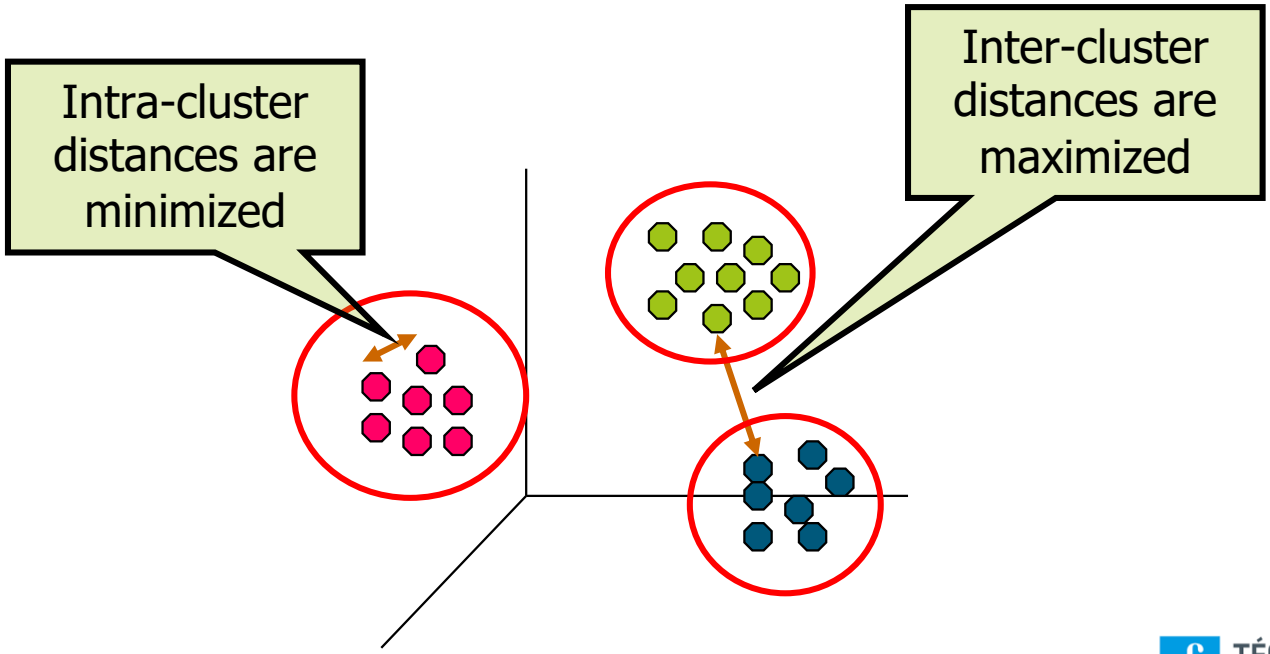
- Supervised learning
  - Models (functions) are constructed based on training examples
  - Need of a (very large) annotated dataset
  - Q: How many examples are needed?
    - A: Depends on the complexity of the problem and of the classification method, and there is no straight answer
    - Complex non-linear methods usually need much more data (e.g. deep learning)

Classification (IV)

- Algorithms:
  - Linear classifiers (Logistic Regression, Naïve Bayes, Perceptron, etc.)
  - Support Vector Machines
  - KNN
  - Decision Trees (Random Forests)
  - Neural Networks
  - (Fuzzy) Rule based systems
- Q: Which is the best algorithm?
  - A: Depends on the problem and the available data!
  - … and no one can easily give you an answer

Clustering

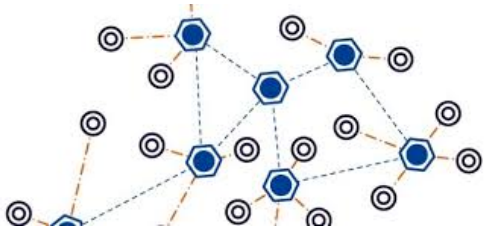
- Categorization in the absence of labels
  - Find groups in the data that share similar characteristics (clusters)
  - Main principle: Maximize intra-class similarity & minimize interclass similarity



Clustering (II)

- Clustering is unsupervised classification (class labels are unknown)
  - But note that clusters are NOT classes
  - ☺ dismiss the need for annotated datasets
- "Discovery" of new knowledge from data
- Can be very useful for summarizing large data sets
  - For large *n* and/or high dimensionality
- Main issues:
  - Number of clusters (there are no predefined classes, remember?)
  - Validation (there are no predefined classes, remember?)

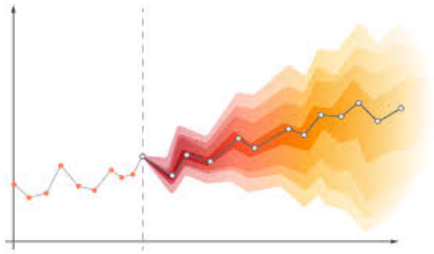
Clustering (III)



- Algorithms:
  - K-means
  - Fuzzy C-Means
  - Hierarchical Clustering (Agglomerative, Divisive)
  - Density Based
- Main application in IoT:
  - Wireless Sensor Networks (sensor clusters)
  - Aggregation of sensor data

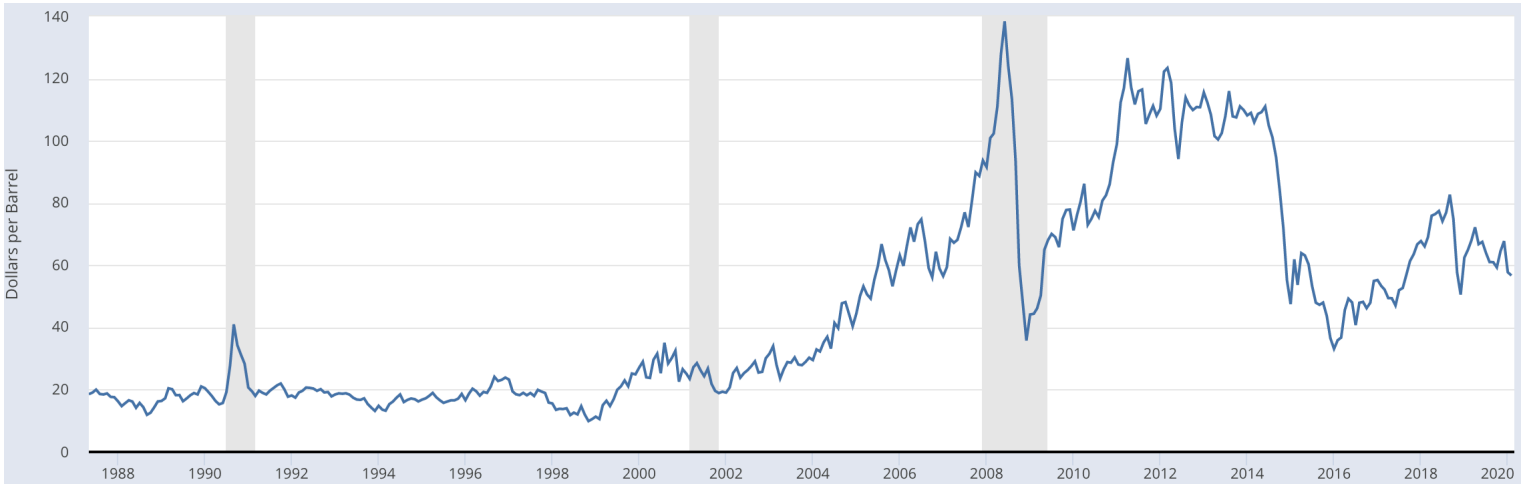
Forecasting and Prediction

- Forecasting: the process of making predictions of the future based on past and present data and most commonly by analysis of trends
- Forecasting vs. Prediction: Prediction is usually used as a more general term. E.g.:
  - Estimate next day network traffic (forecasting); Estimate number of times network traffic will be above 80% capacity in the next month (prediction)
- Uncertainty is inherent to forecasting
  - A degree of uncertainty should be indicated with each forecasted value



Forecasting and Prediction (II)

- Inputs: Time-series / Longitudinal data (or features extracted from such data)



- Features: Date Time features, Lag features (past step values), Window features (aggregation of values in past intervals)

Forecasting and Prediction (III)

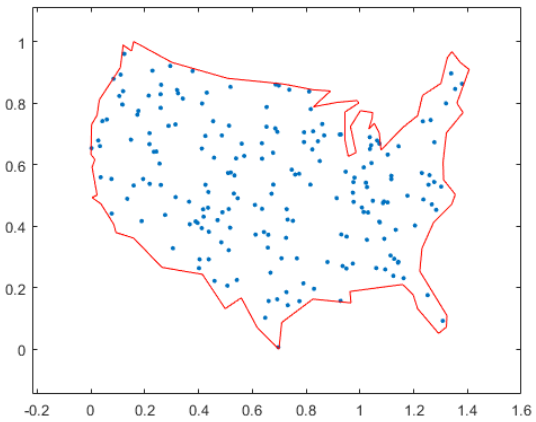
- One-step ahead vs. Multiple-steps ahead
  - Forecast the next point in the data-series vs. short/medium/long term forecasts
  - Medium/long term predictions are naturally more difficult (exception: cyclic / seasonal data)
- Applications in IoT (examples):
  - Energy consumption planning based on smart-meters data
  - Network load forecasting for resource optimization
  - Indoor temperature prediction
  - Failure prediction, Predictive maintenance
  - Useful for all time-series sensor data

Forecasting and Prediction (IV)

- Methods:
  - Naïve approach: assume next point is equal to the last one (baseline for time-series data)
  - Drift Method
  - Seasonal naïve
  - Moving Average, Weighted Moving Average, ARMA, ARIMA, Extrapolation, etc.
  - Regression Analysis
  - ANN (multi-layer perceptron and recurrent)
  - ANFIS (Neural Fuzzy model)

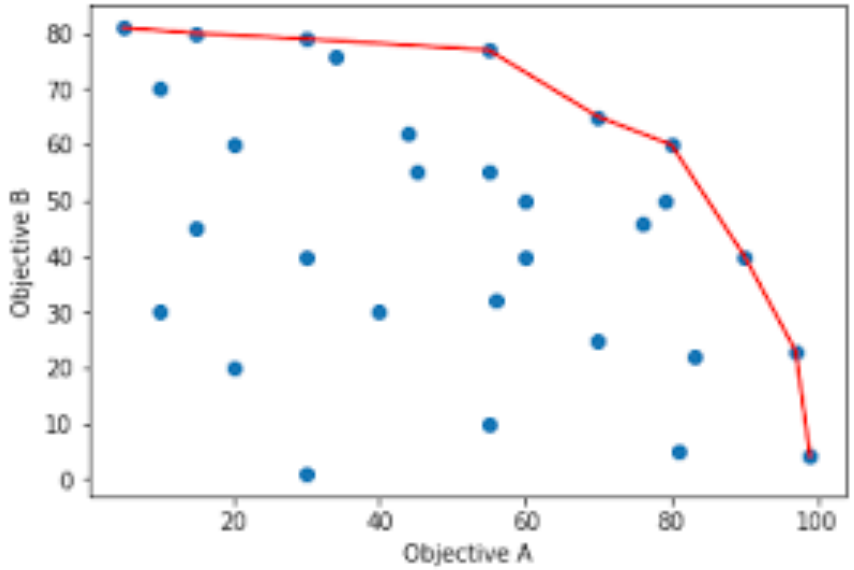
(Heuristic) Optimization

- Optimization: find the best solution among all possible solutions
  - Needs an objective function (a way to measure how good a solution is)
- Heuristic optimization:
  - Find a near optimal solution in a reasonable time frame
  - Trades optimality, completeness, accuracy, or precision for speed
- Typical example: Traveling salesman



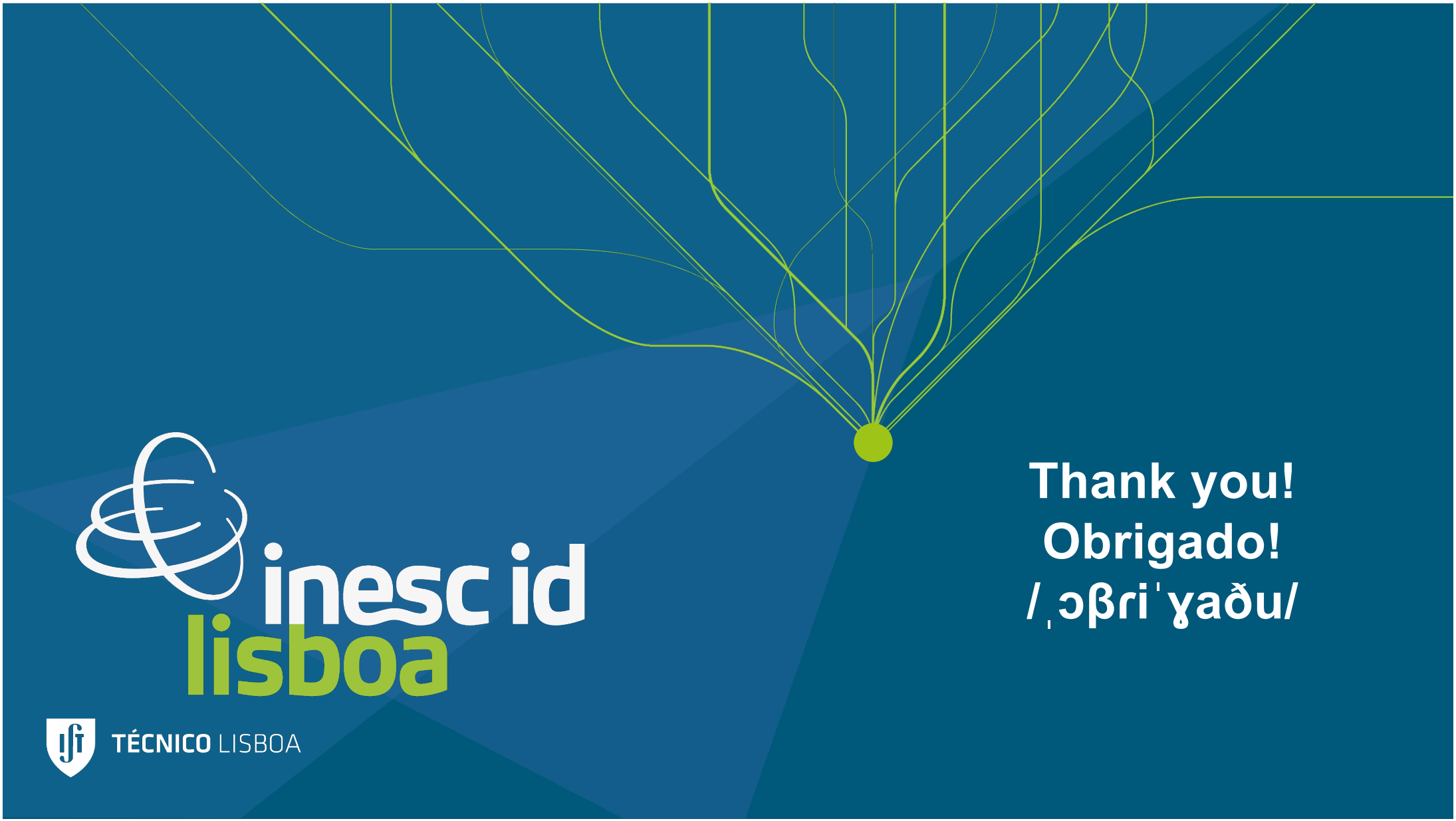
(Heuristic) Optimization (II)

- Single Objective vs. Multi-Objective
  - E.g.: minimize energy consumption while maximizing network throughput
  - Usually no single solution optimizes all objectives. A Pareto front defines a possibly infinite number of Pareto optimal solutions



(Heuristic) Optimization (III)

- Main Applications in the IoT:
  - Network Routing with different objectives: shortest path; energy conservation; network lifetime; memory footprint; congestion control; etc.
  - Parameter optimization in systems, devices, applications, and algorithms
    - E.g.: optimize parameters on a forecasting problem
- Algorithms:
  - Genetic Algorithms
  - Particle Swarm Optimization
  - Ant Colony Optimization
  - Fish Shoal Optimization
  - Bees
  - Etc.


inesc id
lisboa
TÉCNICO LISBOA
Thank you!
Obrigado!
/ˌɔβriˈɣaðu/