Data Preparation

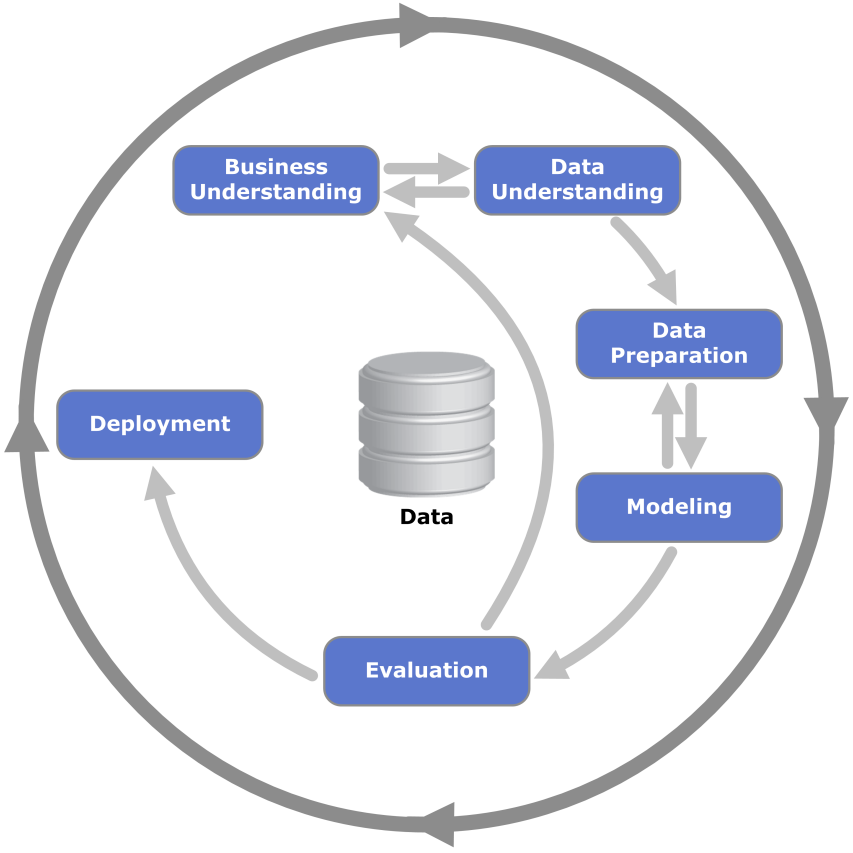
Rita P. Ribeiro

Machine Learning - 2022/2023

U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

[dcc] DEPARTAMENTO DE CIÊNCIA DE COMPUTADORES
FACULDADE DE CIÊNCIAS DA UNIVERSIDADE DO PORTO

From previous class . . .



Shearer C.: The CRISP-DM model: the new blueprint for data mining, J Data Warehousing (2000)

Today

References

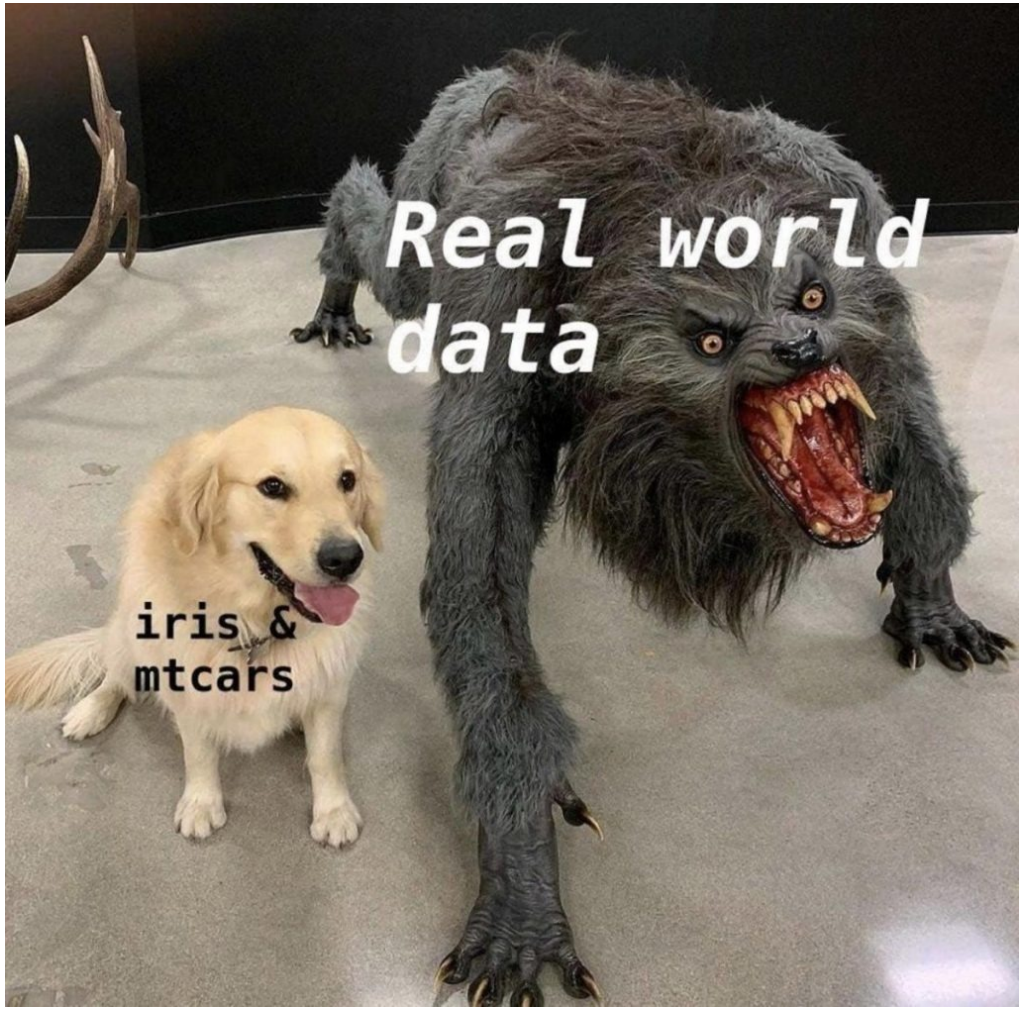
- Moreira, João, et al. 2018. Data Analytics: A General Introduction. Ch 4
- Gama, João, et al. 2015. Data Mining 3rd Ed. Ch 3.
- Aggarwal, Charu C. 2015.Data Mining, the Textbook. Ch 2.1-2.4.3.1.

Today

- Data Quality Issues
- Data Pre-processing
  - Feature Extraction
  - Data Cleaning
  - Data Transformation
  - Feature Engineering
  - Sampling
  - Dimensionality Reduction

Data Quality Issues

Why?



Data Quality

- In its raw format, real world data may be
  - missing
  - insconsistent across different data sources
  - erroneous
- Poor data quality challenges effective data analysis

Example:

- A classification model for predicting a client's loan risks is built using poor data
  - credit-worthy candidates are denied loans
  - loans are given to individuals that default

Data Quality

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

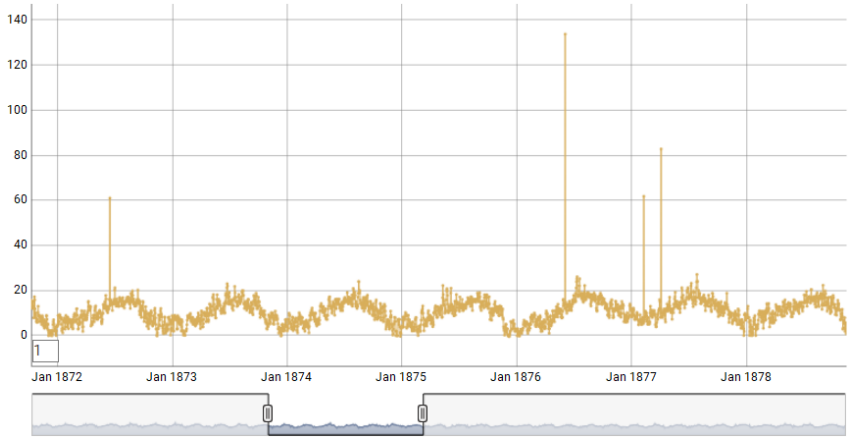
- Examples of data quality problems:
  - Noise and outliers
  - Missing values
  - Duplicate data
  - Incosistent or incorrect data

Noise

- Noise may refer to modification of original values
- Examples: distortion of a person's voice when talking on a poor phone and "snow" on television screen.

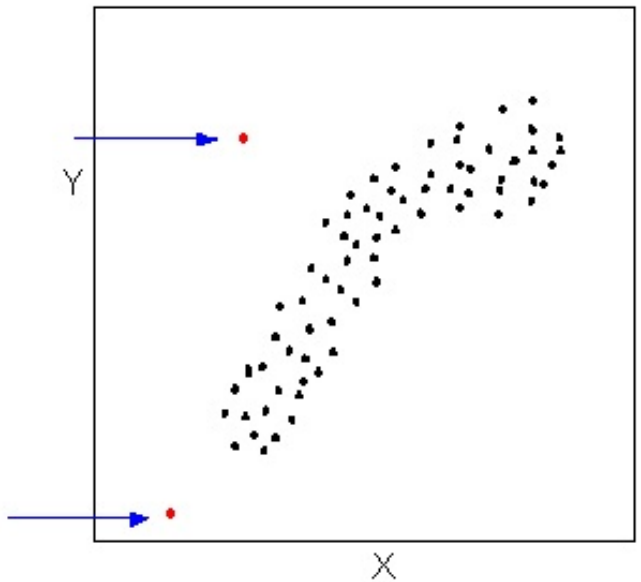
Outliers

- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set
- Case 1: outliers are noise that interferes with data analysis
  - 130$^{\circ}C$ value for air temperature



Outliers

- Case 2: outliers are the goal of our analysis
  - credit card fraud, intrusion detection



- What are the causes?

Missing Values

- Missing Completely at Random (MCAR)
  - missing value is independent of observed and unobserved data
  - there is nothing systematic about it
  - e.g. a lab value because a lab sample was processed improperly
- Missing at Random (MAR)
  - missing value is related to unobserved data of the variable itself
  - informative / non-ignorable missingness
  - e.g. a person did not entered his/her weight in a survey
- Missing Not at Random (MNAR)
  - missing value is related to observed data, not to unobserved data.
  - there may be something systematic about it
  - e.g. missing income value may depend on the age

Missing Values (cont.)

Solutions:

- remove observations with missing values, i.e. consider only complete cases
  - critical if there are many observations with missing values
- ignore missining values in the analytical phase
  - use methods that are inherently designed to work robustly with missing values
- make estimates to fill the missing values - **imputation**
  - most common value of the attribute (e.g. mean, mode);
  - based on other(s) attribute(s);
  - more sophisticated methods
  - it might introduce bias in data and affect the results

Duplicates

- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeneous sources
- Examples
  - Same person with multiple email addresses
- It is necessary a process of dealing with duplicate data issues
  - When should duplicate data not be removed?

Inconsistent or Incorrect Data

- This the hardest type of data quality issues to detect
- It may depend on expert domain knowlege
- Examples:
  - author name in a publication (e.g. John Smith, J. Smith, Smith J.)
  - a city called Shanghai in the United States

Data Pre-processing

Data Pre-processing

- Steps carried out before any further analysis of the available data.
- Data can come from a multitude of different sources
- Frequently, we have data sets with unknown variable values
- Many data mining methods are sensitive to the scale and/or the type of variables
  - Different variables may have different scales
  - Some methods are unable to handle either nominal or numerical variables

Data Pre-processing

- The need to "create"new variables to achieve our objectives
  - Sometimes we are more interested in relative values (variations) than absolute values
  - We may be aware of some domain-specific mathematical relationship among two or more variables that is important for the task
- The need to select representative subsets of data, as our data set may be too large for some methods to be applicable

Data Pre-processing

- Feature Extraction
  - extract features from raw data on which analysis can be performed.
- Data Cleaning
  - data may be hard to read or require extra parsing efforts.
- Data Transformation
  - it may be necessary to change some of the values of the data.
- Feature Engineering
  - to incorporate some domain knowledge.
- Data and Dimensionality Reduction
  - to make modeling possible.

Feature Extraction

- It is very application specific and a very crucial step.
  - **sensor data**: large volume of low-level signals associated with date/time attributes
  - **image data**: very high-dimensional data that cane be represented by pixels, color histograms, etc.
  - **web logs**: text in a prespecified format with both categorical and numerical attributes
  - **network traffic**: network packets information
  - **document data**: raw and unstructured data

Data Cleaning: Handling Missing Values

Ultimate Goal

- Making our data set tidy
  - each value belongs to a variable and an observation
  - each variable contains all values of a certain property measured across all observations
  - each observation contains all values of the variables measured for the respective case
- These properties lead to data tables where:
  - each row represents an observation
  - each column represents an attribute measured for each observation

Data Cleaning: Handling Missing Values

Main Strategies

- Remove all cases in a data set with some unknown value
- Fill-in the unknowns with the imputation of the most common value (a statistic of centrality)
- Fill-in with the most common value on the cases that are more "similar" to the one with unknowns.
- Fill-in with linear interpolation of nearby values in time and/or space.
- Explore eventual correlations between variables
- Do nothing: many data mining methods are designated to work robustly with missing values

Data Cleaning: Handling Incorrect Values

- Inconsistency detection
  - data integration techniques within the database field
- Domain knowledge
  - data auditing that use domain knowledhe and constraints
- Data-centric methods
  - statistical-based methods to detect outliers

Data Transformation

- Map entire set of values of an attribute to a new set of values such that each old value can be identified with one of the new values
- Why it may be useful?
  - Imagine two attributes (e.g. age, salary) with a very different scale
  - Any aggregation function (e.g. euclidean distance) computed on the set of cases, will be dominated by the attribute of larger magnitude.
- Some common strategies:
  - Normalization
  - Binarization / One-Hot Enconding
  - Discretization

Data Transformation: Normalization

- Min-Max Scaling (Range-based Normalization)

$$y_i = \frac{x_i - \min_x}{\max_x - \min_x}$$

  - $\min_x$ and $\max_x$ are the minimum and maximum values of attribute x
  - values will lie in the range $[0, 1]$
- It is not robust for scenarios where there are outliers
  - if an erroneous age value of 800 is registered instead of 80, most of the values will be in the range $[0, 0.1]$

Data Transformation: Normalization

- Standarization (z-score Normalization):

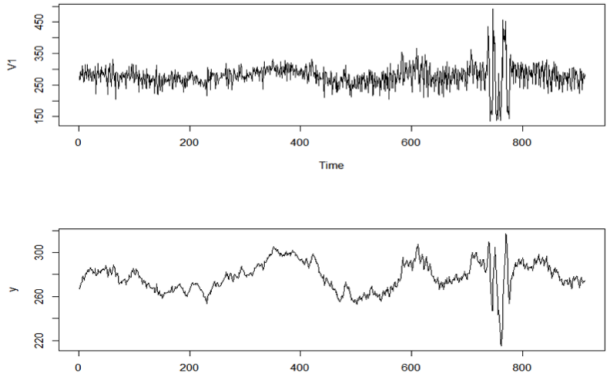
$$y_i = \frac{x_i - \mu_x}{\sigma_x}$$

  - μ_x and σ_x are the mean and the standard deviation of attribute x
  - values are rescaled so that they have $\mu_x = 0$ and $\sigma_x = 1$
  - values will, typically, lie in the range $[-3, 3]$ under a normal distribution assumption

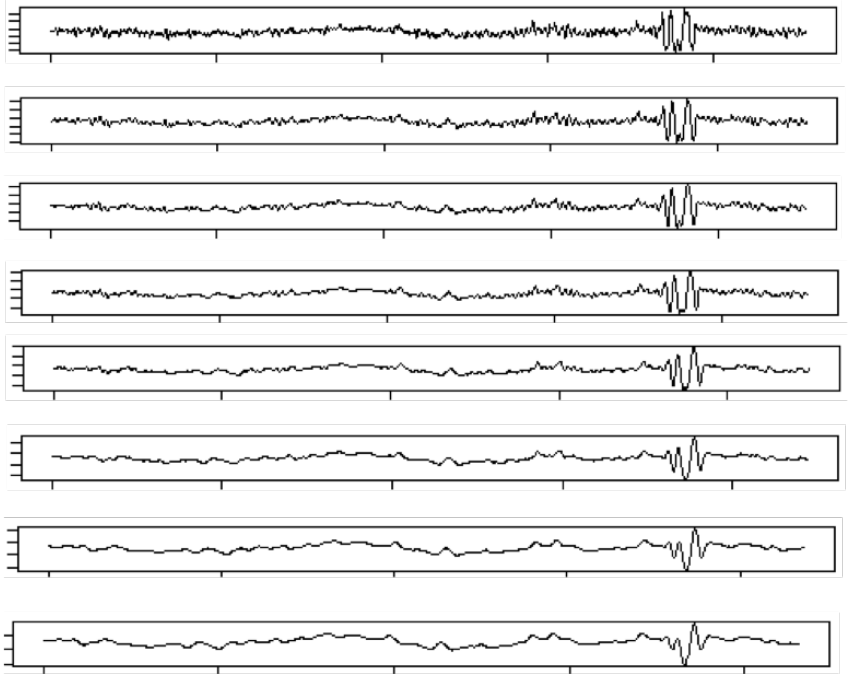
Data Transformation: Normalization

- In time series it is common to use different techniques.
  - to adjust mean, variance, range
  - to remove unwanted, common signal

Low-pass filter



Minimum Description Length (MDL)



Data Transformation: Binarization / One-Hot Enconding

- Some data mining methods only handle numeric attributes.
- If the categorical attribute is not ordinal, it is necessary to convert it into a numerical attribute.
- Binarization: if the atribute has only 2 possible nominal values, it can be transformed into 1 binary attribute
  - fever: yes/no $\rightarrow$ fever: 1/0
- One-Hot Enconding: if the atribute has k possible nominal values, it can be transformed into k binary attributes
  - eye_color: brown/blue/green $\rightarrow$
    eye_brown: 1/0, eye_blue: 1/0, eye_green: 1/0

Data Transformation: Discretization

- Convert a continuous attribute into an ordinal attribute of numeric variables.
- Some unsupervised discretization: find breaks in the data values
  - Equal-width
    - it divides the original values into equal-width range of values
    - it may be affected by the presence of outliers
  - Equal-frequency
    - it divides the original values so that the same number of values are assigned to each range
    - it can generate ranges with very different amplitudes
- Supervised discretization: use class labels to find breaks

Feature Engineering

Fundamental to the application of machine learning.

'(...) some machine learning projects succeed and some fail. What makes the difference? Easily the most important factor is the features used.' - Pedro Domingos, 2012

- The process of using domain knowledge of the data to create features that might help when solving the problem.
- New features that can capture the important information in a data set much more efficiently than the original features.

Feature Engineering

Case 1

- express known relationships between existing variables
  - create ratios and proportions like credit card sales per person
  - the average web session duration per user, the frequency of access, etc.

Case 2

- express known case dependencies
  - some tools shuffle the cases, or are not able to use the information about their dependencies (time, space, space-time)
  - two main ways of handling this issue:
    - constrain ourselves to tools that handle these dependencies directly
    - create variables that express the dependency relationships

Feature Engineering: Cases Dependencies

- In time series is common to create features that represent relative values instead of absolute values, so to avoid trend effects.

$$y_t = \frac{x_t - x_{t-1}}{x_{t-1}}$$

- Other common technique is Time Delay Embedding

X_{t-3}	X_{t-2}	X_{t-1}	X_t
x_{t_1}	x_{t_2}	x_{t_3}	x_{t_4}
x_{t_2}	x_{t_3}	x_{t_4}	x_{t_5}
...			
$x_{t_{n-3}}$	$x_{t_{n-3}}$	$x_{t_{n-1}}$	x_{t_n}

- Create variables whose values are the value of the same variable in previous time steps
- With these embedded variables, standard tools will be able to model the time relationships

- Similar “tricks” can be done with space and space-time dependencies

Sampling Data

- Sample the cases of the original data set to obtain a much smaller data set.
- It is often used for both the preliminary investigation of the data and the final data analysis.
- Processing the entire set of data of interest is too expensive or time consuming.

Sampling Data

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

8000 points 2000 Points 500 Points

Sampling Data

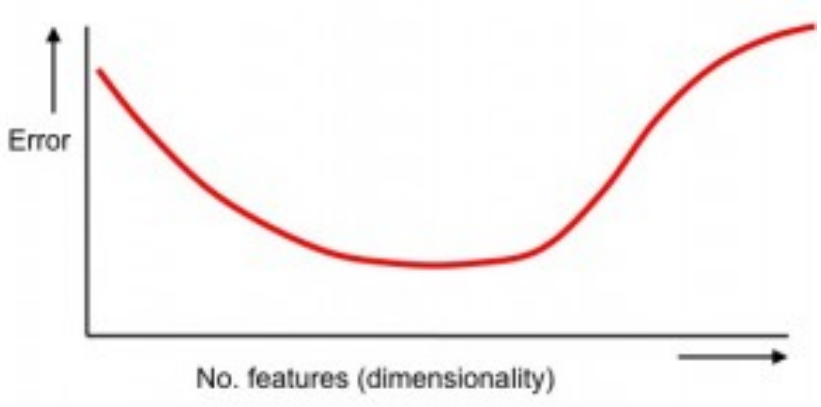
- Random Sampling
  - There is an equal probability of selecting any particular item
  - Sampling without replacement: each selected object is removed from the population
  - Sampling with replacement: selected bjects are not removed from the population, so they can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition
- Incremental Sampling
  - Start with a small sample and increment the size of it until there is no gain in the model performance.

The curse of dimensionality

- When dimensionality of feature space increases, the number of possible combinations of feature values increases exponentially.
- The data becomes increasingly sparse in the space that it occupies.
- We may assume that the more details (features) of the case we collect, the better description of the situation we have at hand.
- Counter-intuitively, it is is not valid.

Dimensionality Reduction

- There is a certain point after which adding new details becomes useless, and moreover, they may work against your model.
- In very high dimensional data many data mining algorithms do not work effectively.



- For example, distance between points, which is critical to some algorithms, becomes less meaningful.

Dimensionality Reduction

Purpose

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

Some Strategies

- Feature Selection
- Principal Components Analysis (PCA)
- Singular Value Decomposition (SVD)

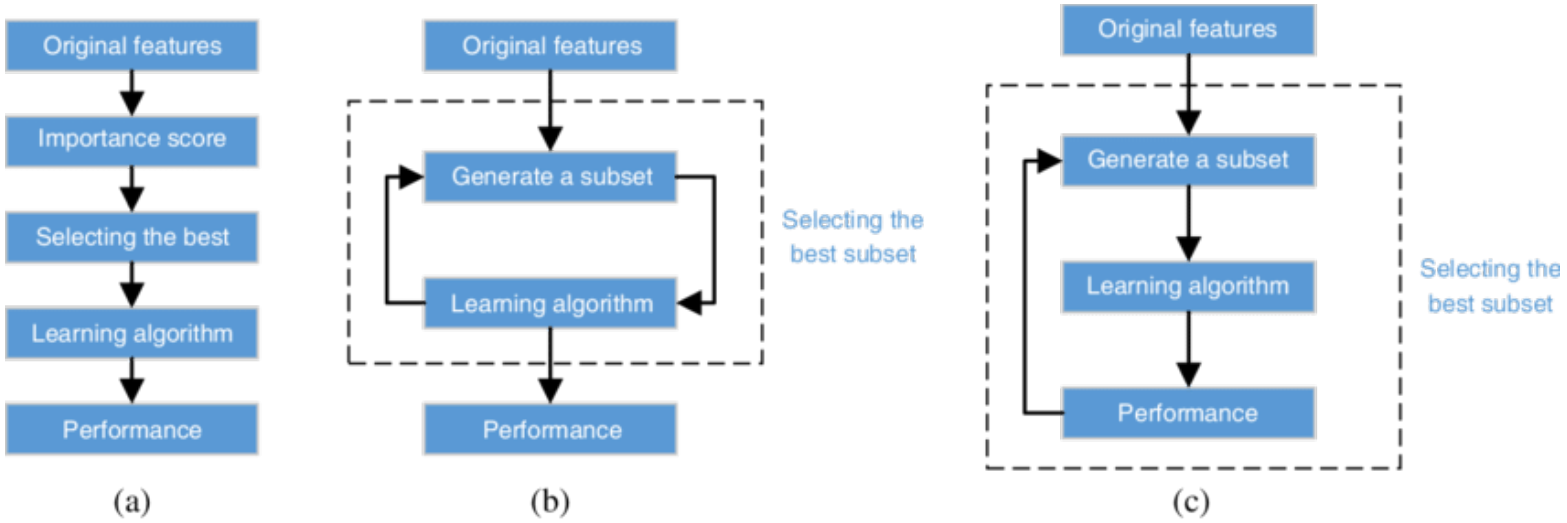
Dimensionality Reduction: Feature Selection

- Discard Irrelevant Features
  - contain information that is not useful for the data mining task at hand
  - e.g. students' ID is irrelevant for predicting students' grades
  - Identified based on domain knowledge
- Discard Redundant Features:
  - 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
  - Identified by feature selection methods

Dimensionality Reduction: Feature Selection

Main types of Feature Selection Methods:

- (a) Filter Methods, (b) Wrapper Methods, (c) Embedded Methods



Source: Wang, Shuihua et al. (2016). Pathological Brain Detection by Artificial Intelligence in Magnetic Resonance Imaging Scanning. Progress In Electromagnetics Research. 156. 105-133.

Dimensionality Reduction: Feature Selection

Filter Selection Methods

- Selects features independently of the data mining task
  - Removes features with low variance (rank by cut-off)
  - Removes features with high correlation (rank by cut-off)
  - Ranking features by relevancy measure, depending on relationship with the output variable (target)

Embedded Selection Methods

- Selection is built-in in the algorithm that produces the model for the data mining task.

Wrapper Selection Methods

- Selects features taking into account the data mining task
  - It uses a ML algorithm to assess the performance of models based on different subset of features
  - Search for optimal subset of features
  - Many techniques developed, especially for classification
- Iterative:
  - **Forward Selection**: select one attribute, add, repeat
  - **Backward Elimination**: select one attribute, remove, repeat
- Recursive:
  - attributes are recursively removed from current set

Filter vs Wrapper Methods

- Filter Methods:
  - faster, as they do not involve training the models
  - use statistical methods for evaluation of a subset of features
  - fail to recognize importance of combined features
  - select features that carry similar information about the target
- Wrapper Methods:
  - computationally more expensive
  - use model performance estimation strategies
  - provide the best subset of features
  - ML algorithm dependent

Dimensionality Reduction

- Instead of selecting features, we can replace by “new” features.
- A new (smaller) set of features where most of the "information"on the problem is still expressed.
- Sometimes, the correlation among the features is not perfect (redundant) but there may exist significant dependencies.
- Main methods:
  - Principal Component Analysis (PCA)
  - Singular Value Decomposition (SVD)
  - Others: supervised and non-linear techniques

Dimensionality Reduction: PCA

Principal Component Analysis (PCA)

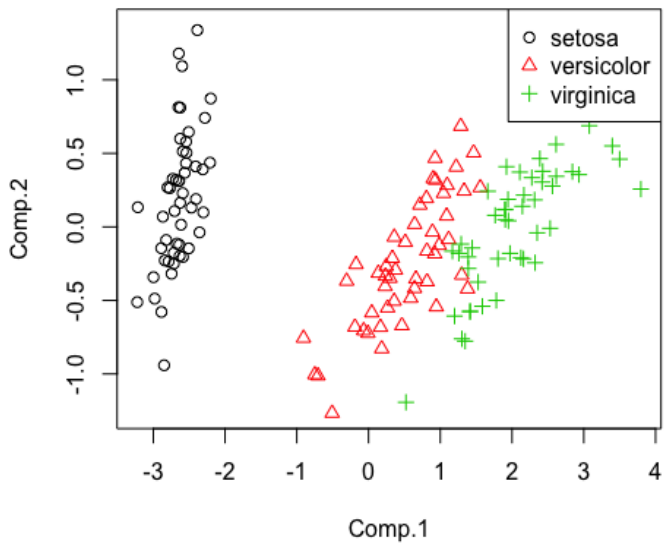
- Find a projection on a new set of axes that captures the largest amount of variability in data
- The new set of axes are formed by linear combinations of the original variables
- We search for the linear combinations that “explain” most of the variability on the original axes
- If we are “lucky” with a few of these new axes (ideally two for easy data visualization), we are able to explain most of the variability on the original data
- Each original observation is then “projected” into these new axes

Dimensionality Reduction: PCA

- Find a first linear combination which better captures the variability in the data
- Move to the second linear combination to try to capture the variability not explained by the first one
- Continue until the set of new variables explains most of the variability (frequently 90% is considered enough)

	Comp.1	Comp.2	Comp.3	Comp.4
Sepal.Length	0.361	0.657	0.582	0.315
Sepal.Width		0.730	-0.598	-0.320
Petal.Length	0.857	-0.173		-0.480
Petal.Width	0.358		-0.546	0.754

$$\begin{aligned} Comp.1 = & \ 0.361 \times Sepal.Length \\ & + 0.857 \times Petal.Length \\ & + 0.358 \times Petal.Width \end{aligned} \quad (1)$$



References

References

Aggarwal, Charu C. 2015. *Data Mining, the Texbook*. Springer.

Gama, João, André Carlos Ponce de Leon Ferreira de Carvalho, Katti Faceli, Ana Carolina Lorena, and Márcia Oliveira. 2015. *Extração de Conhecimento de Dados: Data Mining -3rd Edition*. Edições Sílabo.

Han, Jiawei, Micheline Kamber, and Jian Pei. 2011. *Data Mining: Concepts and Techniques*. 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Moreira, João, Andre Carvalho, and Tomás Horvath. 2018. *Data Analytics: A General Introduction*. Wiley.

Tan, Pang-Ning, Michael Steinbach, Anuj Karpatne, and Vipin Kumar. 2018. *Introduction to Data Mining*. 2nd ed. Pearson.