Engenharia de Características para Aprendizagem Computacional /

Engenharia de Atributos

Chapter 1 – Introduction
Chapter 2 – Problems and Applications

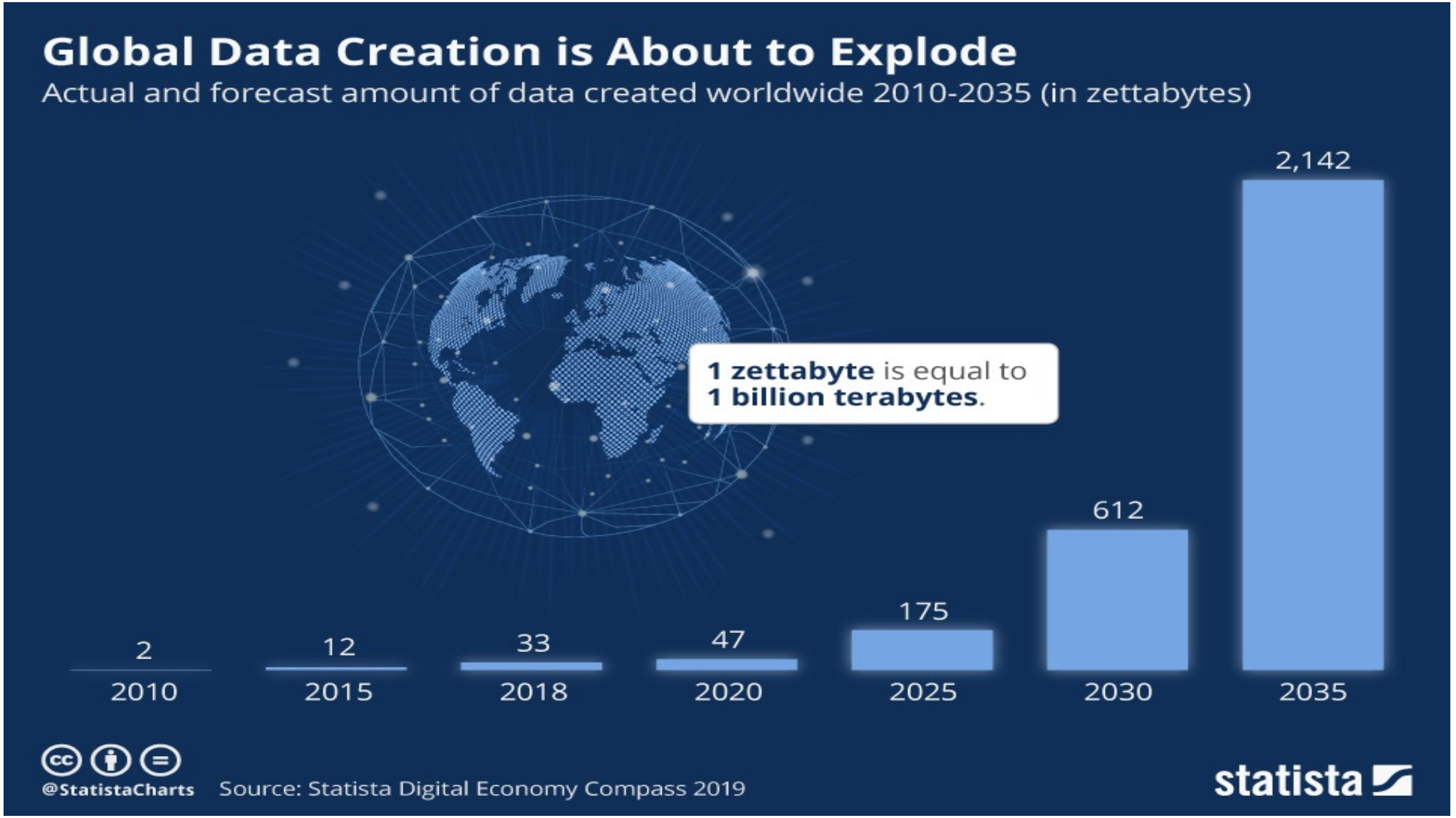
Paulo de Carvalho

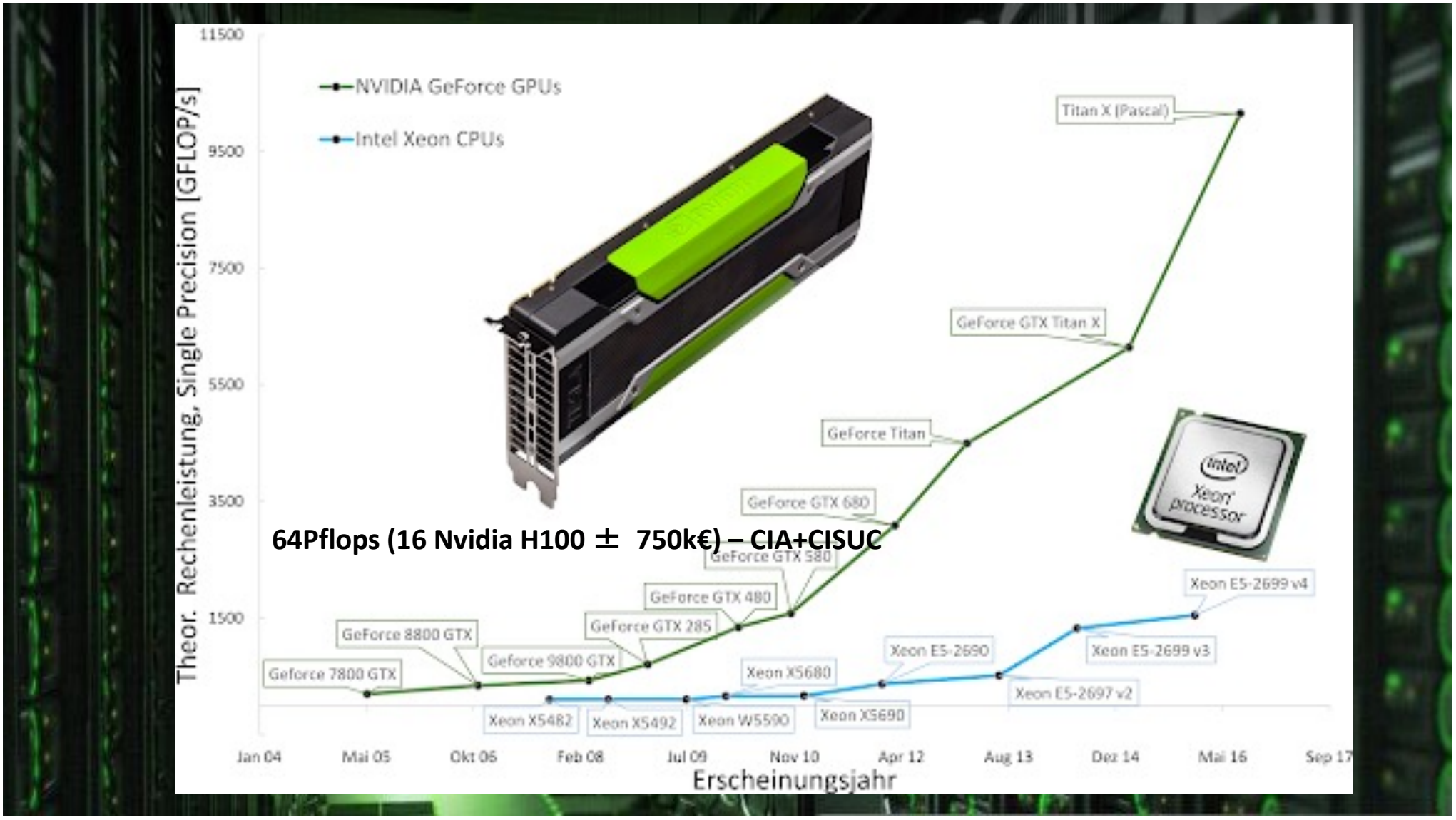
Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia da Universidade de Coimbra
Edição 2023-2024

Outline

- Introduction
  - Why data science?
  - LifeCycle
  - Problems
  - Course outline

Introduction – Data Science Why?





64Pflops (16 Nvidia H100 ± 750k€) – CIA+CISUC

Artificial Intelligence

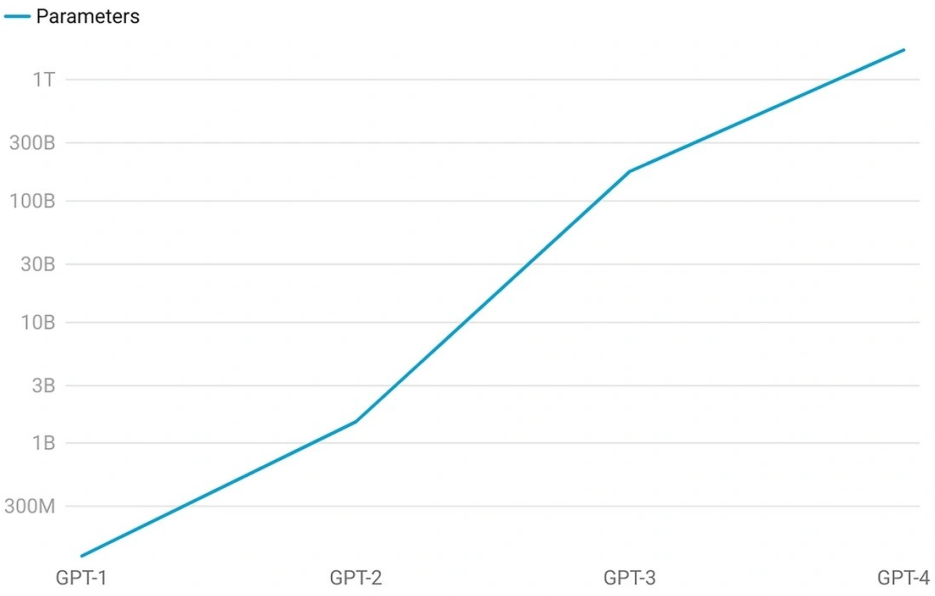
DATA

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed wh training for 300B to
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3. "Weight in training mix" refers to the fraction of exa that are drawn from a given dataset, which we intentionally do not make proportional to the si result, when we train for 300 billion tokens, some datasets are seen up to 3.4 times during trainir are seen less than once.

ChatGPT Parameters

The number of parameters in successive models of ChatGPT has increased massively

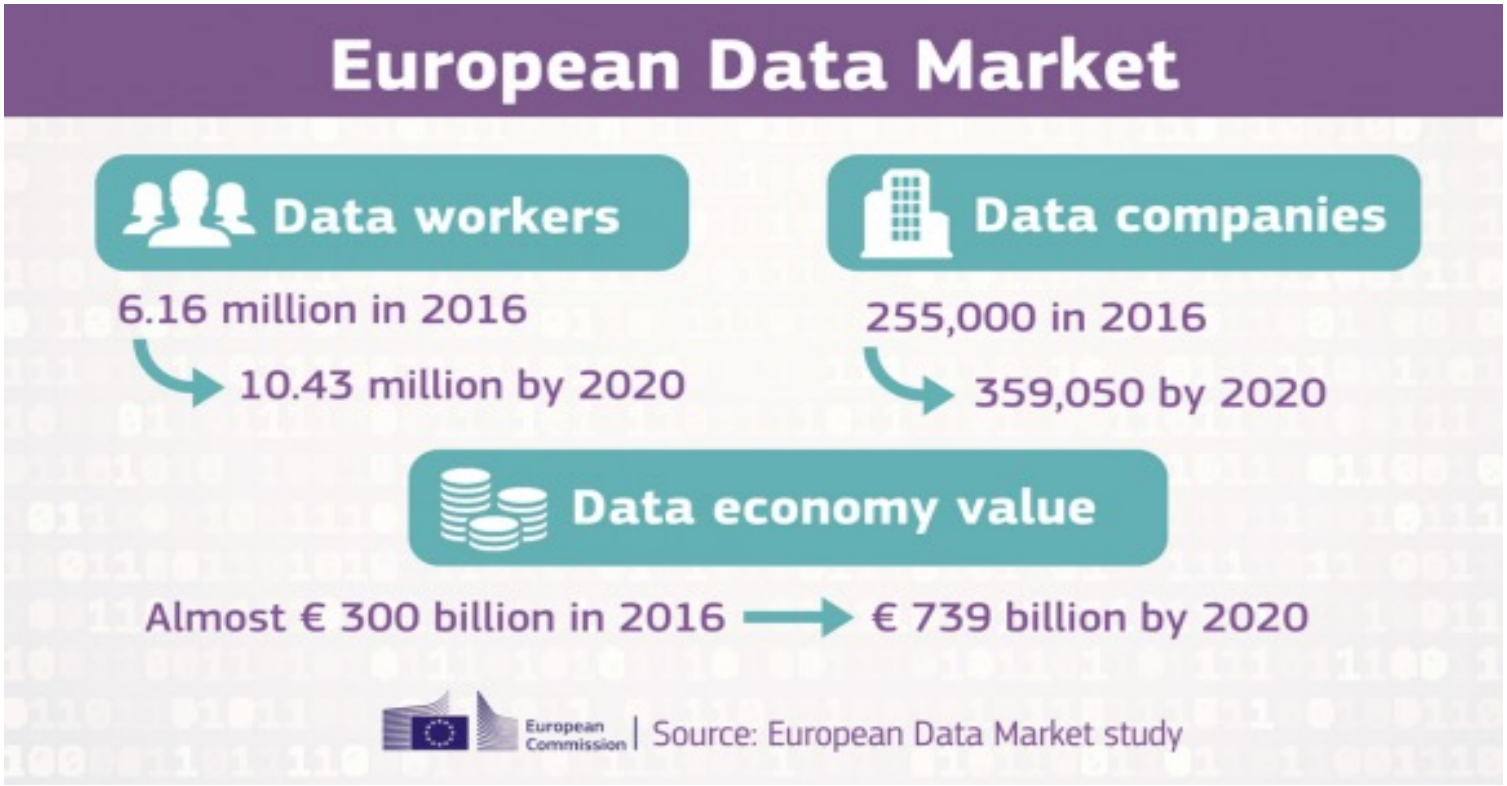


IVEN

ment curation
rial effort
nt training
parent
RDABLE

https://medium.com/@rizaberkan/non-data-driven-approach-to-machine-learning-practical-ai-df686800e288

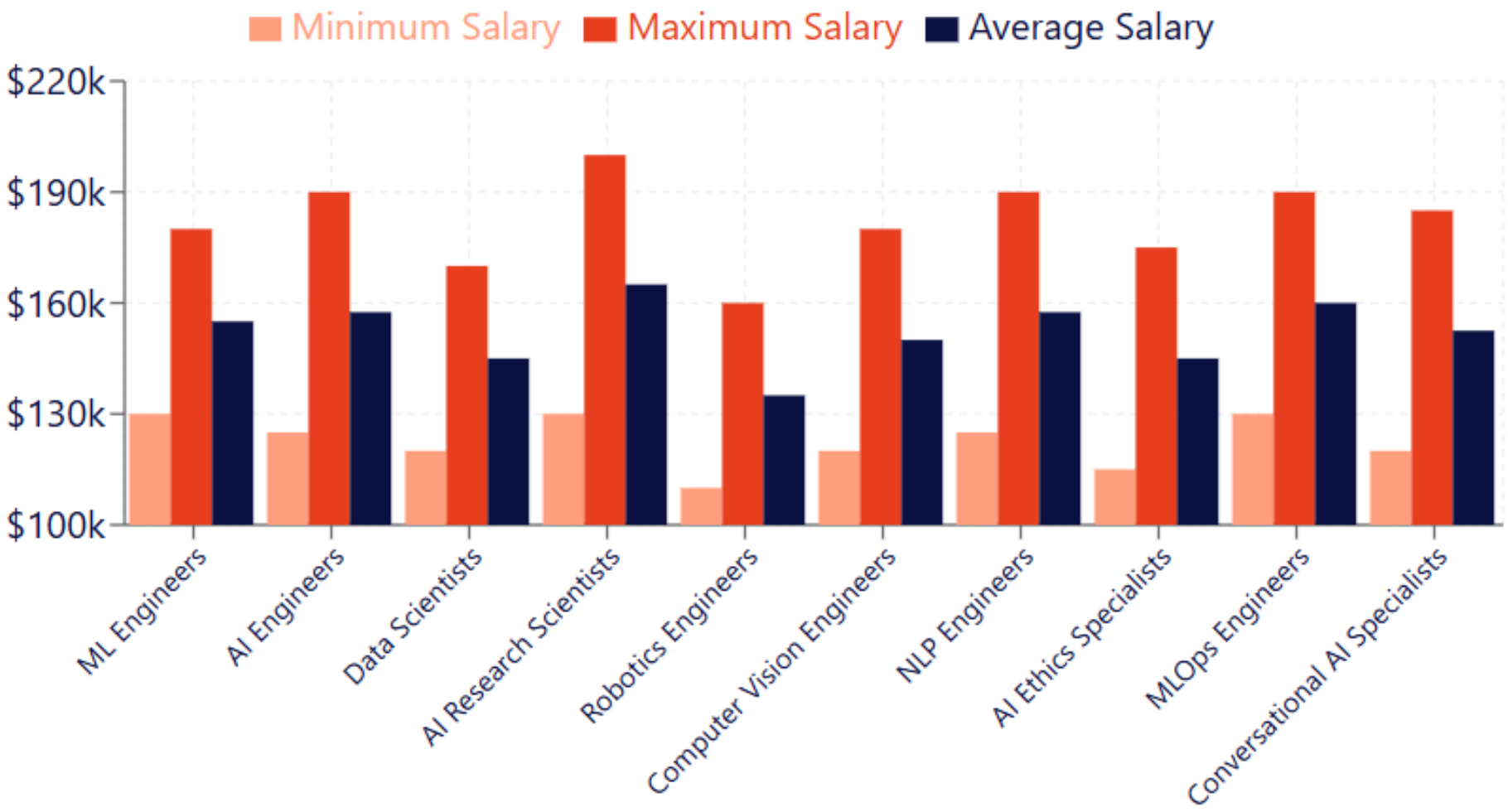
Motivação – Economia Digital



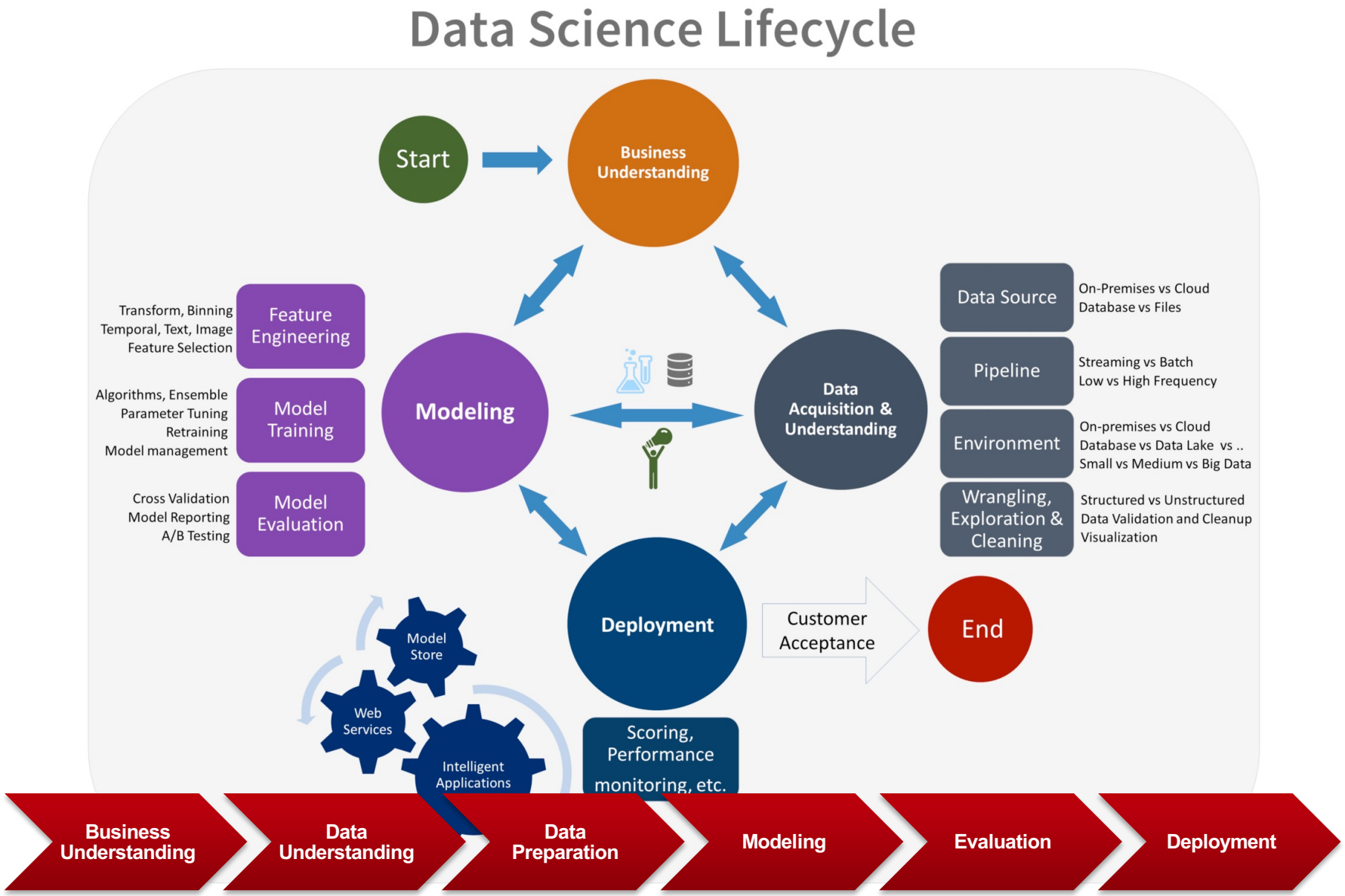
https://www.europeandataportal.eu/pt/highlights/size-and-trends-eu-data-economy

Motivação – Mercado de Emprego

Motivação – Mercado de Emprego



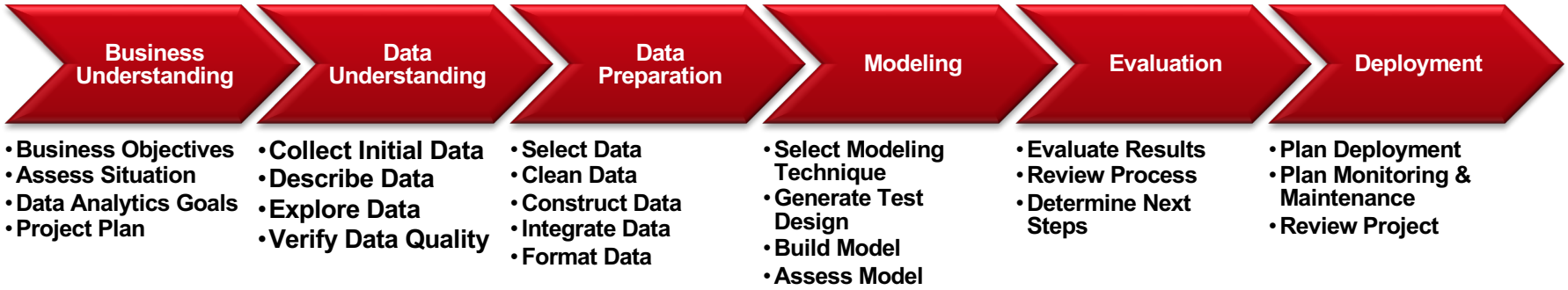
Introduction



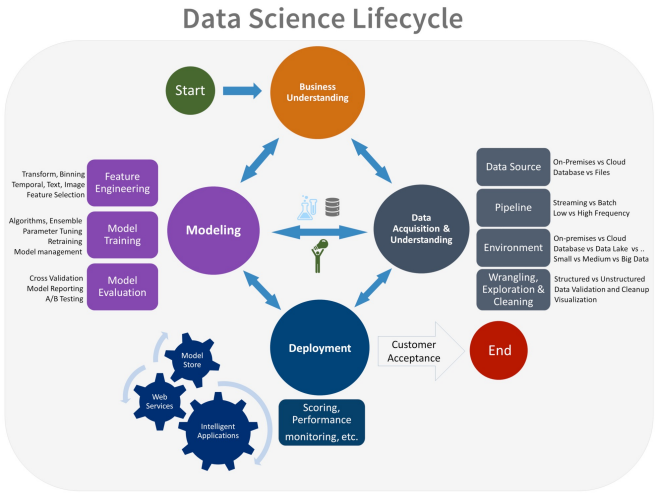
Source:docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/lifecycle

Introduction

CRISP – DM Phases and Tasks

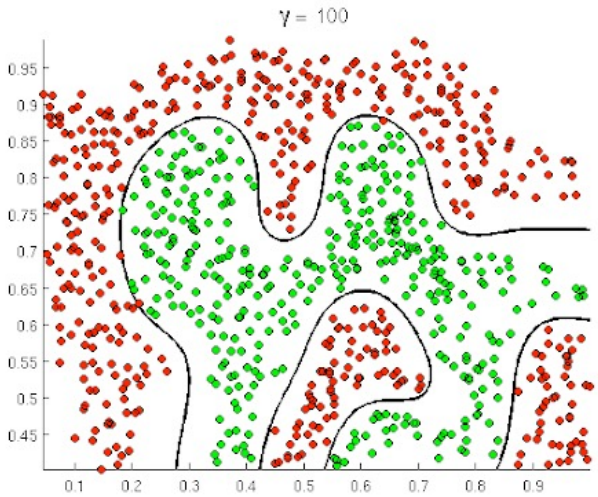


Introduction

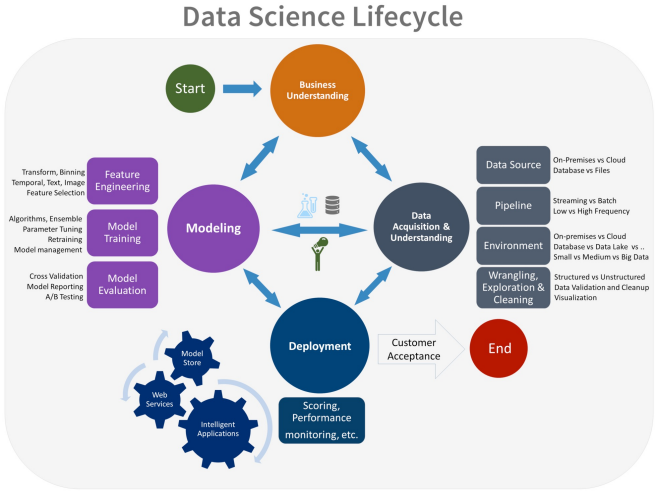


- Business Understanding
  - Why, why, why?
  - Determine the **business objectives**
  - Identify the objectives and **questions** to be answered
  - Do we have **data** do backup decisions

- Data science might answer five types of questions:
  - How much or how many? (regression)
    - Predict evolution of a variable
  - Which category? (classification)
    - Which risk group (e.g., financial group)
  - Which group? (clustering)
    - Customer profile
  - Is this weird? (anomaly detection)
    - Fraud detection
  - Which option should be taken? (recommendation)
    - Clinical Decision support system

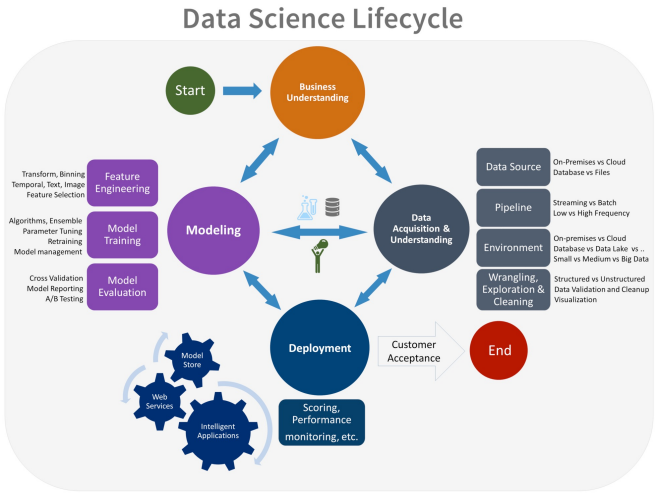


Introduction



- Gathering and understanding the data
  - **Where is the data**?
  - Is the data available/already collected?
    - List **available datasets** (locations, methods used to acquire, constraints for access, …)
    - Describe the data/Data Exploration
      - Check **data volume** and examine its gross properties
      - **Accessibility and availability** of atributes
      - **Type of atributes; ranges of atributes, correlations**
      - **Meaning of each atribute** and its value in business terms
      - Evaluate **basic statistics** (distribution, average, max, min, standard deviation, variance, mode, skewness,..)

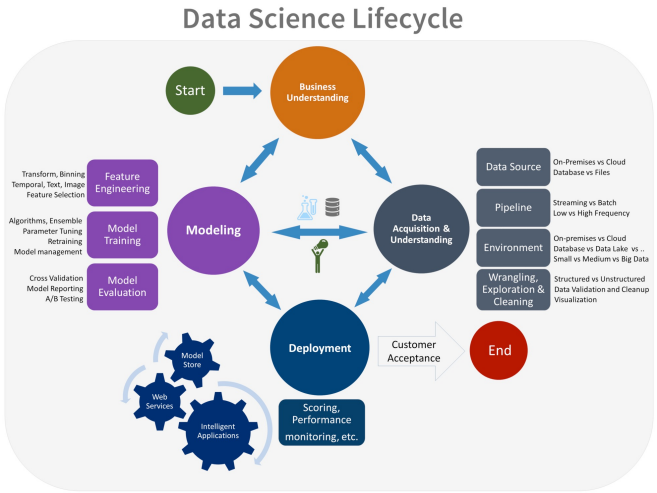
Introduction



- Gathering and understanding the data
  - How can we collect the data?
    - Web
    - NLP
    - Organization of data collection studies
    - GDPR
  - How much data can we collect?
    - Will constraint analysis techniques
  - Very time and resource consuming…

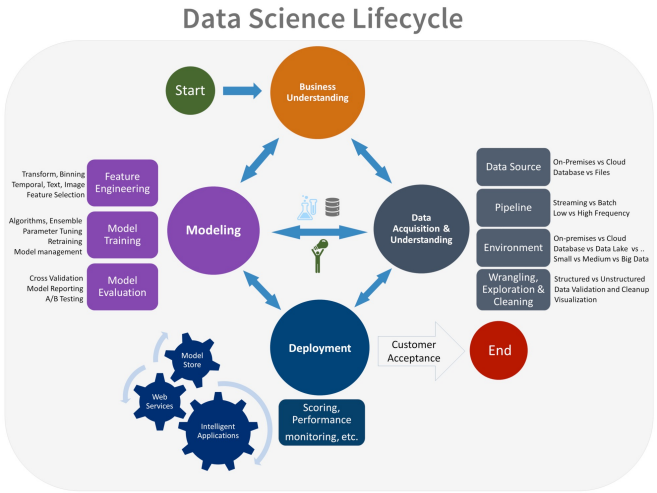
Garbage In – Garbage out

Introduction



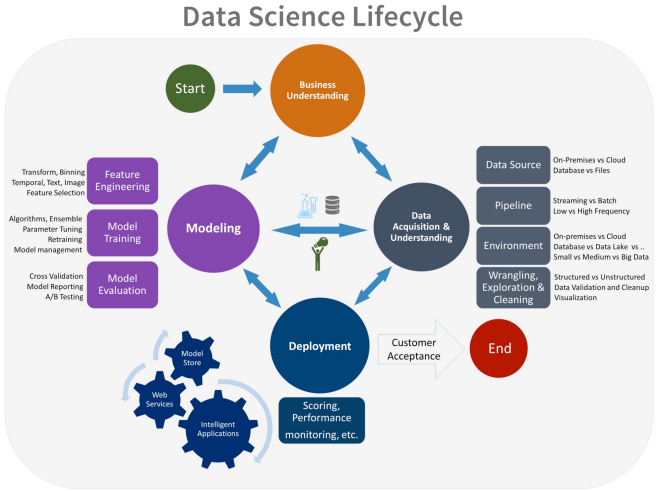
- Data exploration
  - Understand the patterns and biases in your data
    - relations between pairs or small number of attributes
    - Properties of sub-populations
    - Statistical analysis
  - Use graphical tools

Introduction



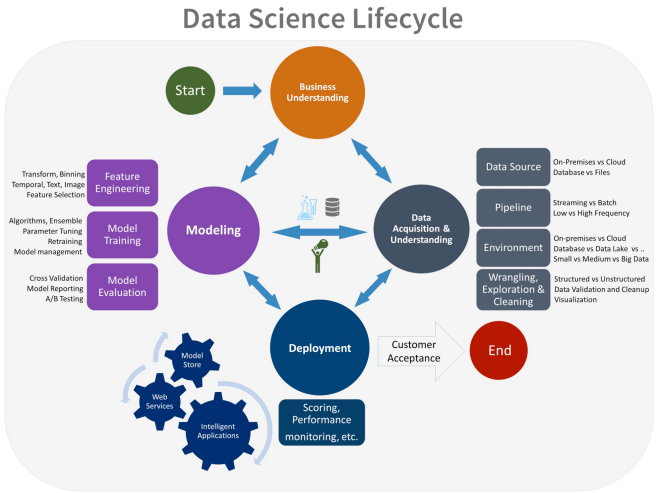
- Data preparation
  - Select the data
    - Decide which datasets will be used
    - Consider use of sampling techniques
      - E.g., unbalanced classes
    - Consider use of augmentation techniques
    - Explain why certain data was included

Introduction

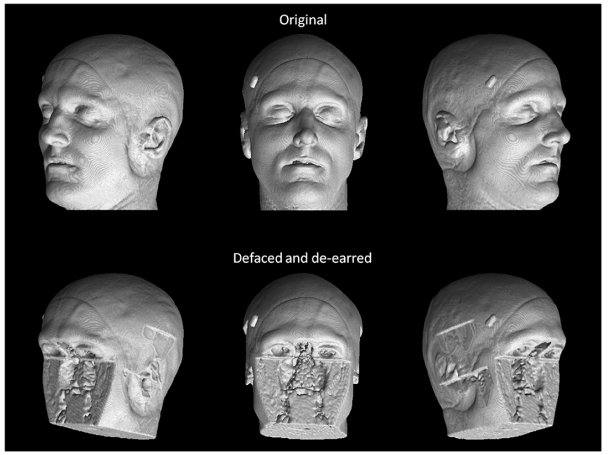


- Data preparation
  - Data Cleaning
    - Data is noisy
      - Acquisition noise
      - Occlusions
      - …
    - Data is missing
      - E.g., Clinical data collection
    - Data is redundant
      - E.g., multiple entries
    - **Data is inconsistent**
      - E.g.: yes/no and 0/1 in the same column
      - Misspelling
      - Date and time formats
  - Main cleaning operations (**also called data cleansing**)
    - Correct, remove or ignore noise
    - Missing values
    - Outliers
    - Duplicate removal
  - 50 to 80% of a data science project is related to data cleaning and preparation!

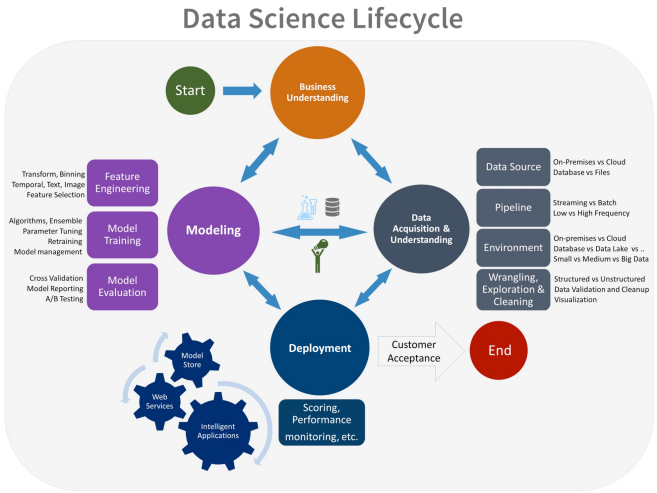
Introduction



- Data preparation – construct
  - Data Transformation
    - Process of changing data presentation; E.g. from alphanumeric to numeric
    - Anonymization
  - Data Normalization
    - Avoiding scale and bias effects
  - Data Augmentation
    - Generate “artificial” data to increase available data; e.g. transforming the data using geometrical transformations
  - Data Labeling (annotation)
    - Label the data set
    - Very expensive! Requires domain knowledge and is very time consuming. (E.g. Medical doctors labeling a medical DB)
  - Feature Engineering
    - Reduce the dimensionality of the data in order to
      - Reduce noise
      - Avoid overfitting



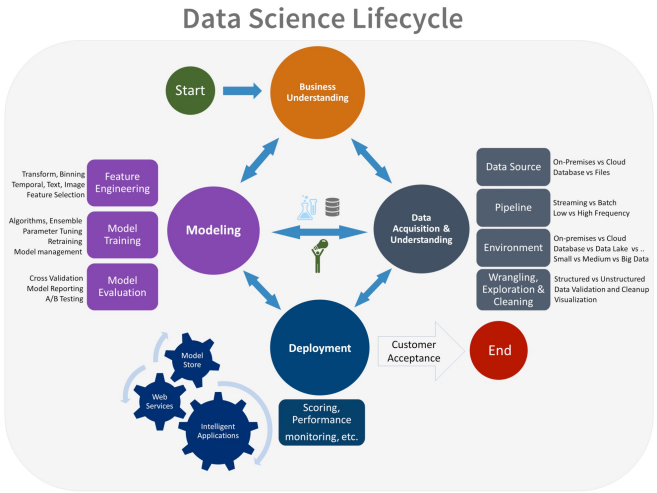
Introduction



- Feature Engineering
  - Feature: property or attribute of a phenomenon being observ
  - Approaches:
    - Knowledge-based
      - E.g., student score is correlated to the amount of sleep hours
    - Ad-hoc
      - Through in all you can think of
      - Reduce the dimension
    - Learning
      - Let the data show the relevant dimensions
      - Requires large amounts of data

Coming up with features is difficult, time-consuming, requires expert knowledge. 'Applied machine learning' is basically feature engineering." *Andrew Ng*

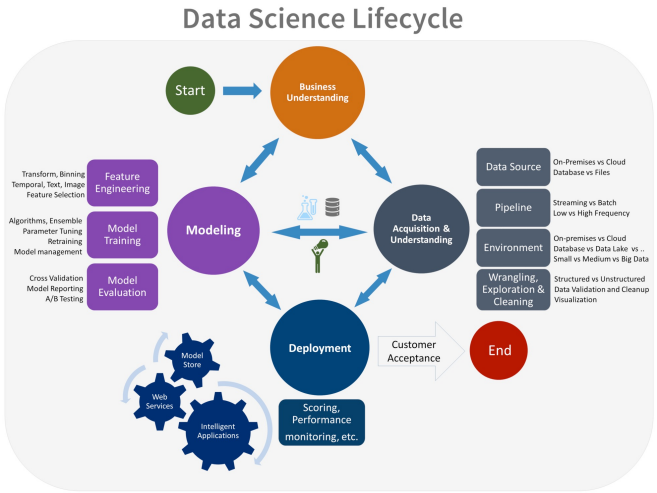
Introduction



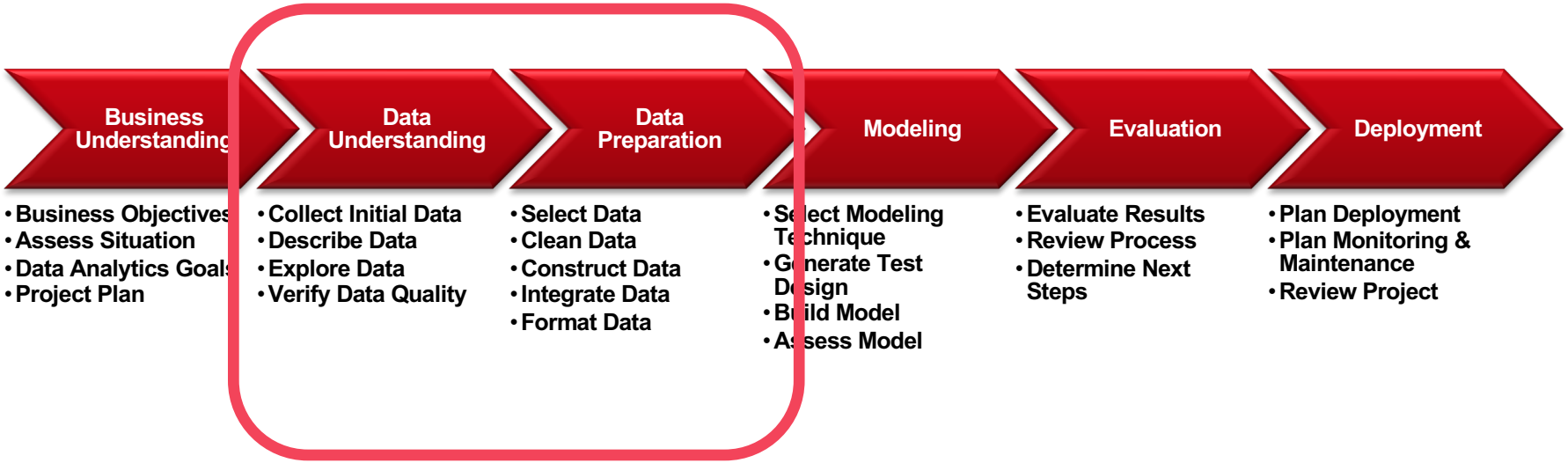
- Integrate Data
  - Integrate sources and store the results (in new tables
- Format Data
  - Format according to tool requirements
    - E.g. order of the attributes
  - Reordering records
    - E.g. some tools require that data is sorted according to output class
  - Reformatting within-value
    - Remove subset of characters
    - Lowcase, uppercase,
    - …

Introduction

- Modeling
  - Machine learning



Course


Business Understanding
• Business Objectives
• Assess Situation
• Data Analytics Goals
• Project Plan
Data Understanding
• Collect Initial Data
• Describe Data
• Explore Data
• Verify Data Quality
Data Preparation
• Select Data
• Clean Data
• Construct Data
• Integrate Data
• Format Data
Modeling
• Select Modeling Technique
• Generate Test Design
• Build Model
• Assess Model
Evaluation
• Evaluate Results
• Review Process
• Determine Next Steps
Deployment
• Plan Deployment
• Plan Monitoring & Maintenance
• Review Project

Problems and solutions: Examples

- Some Examples
  - Personalized healthcare recommendations
    - Oncora's software uses machine learning to create personalized recommendations for current cancer patients based on data from past ones.

Problems and solutions: Examples

- Some Examples
  - Optimizing shipping routes in real-time
  - UPS uses data science to optimize package transport from drop-off to delivery. Its latest platform for doing so, Network Planning Tools (NPT), incorporates machine-learning and AI to crack challenging logistics puzzles, such as how packages should be rerouted around bad weather or service bottlenecks.

Problems and solutions: Examples

- Some Examples
  - Automating digital ad placement
  - Instagram uses data science to target its sponsored posts. The company's data scientists pull data from Instagram as well as its owner, Facebook, which has exhaustive web-tracking infrastructure and detailed information on many users, including age and education. From there, the team crafts algorithms that convert users' likes and comments, their usage of other apps and their web history into predictions about the products they might buy.

Engenharia de Características para Aprendizagem Computacional / Engenharia de Atributos

Chapter 3 – Data Preparation

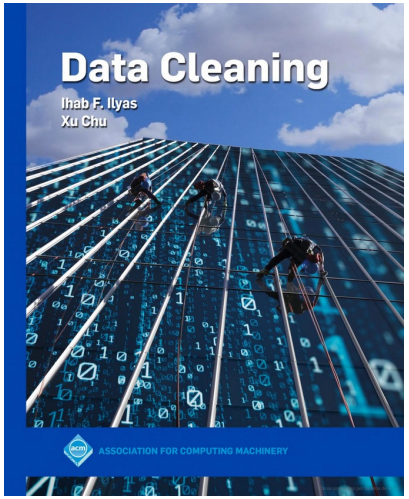
Attribute handling & Cleaning

Paulo de Carvalho/Rui Paiva

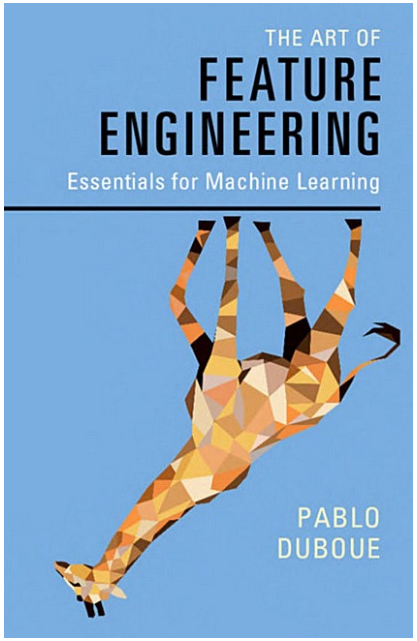
Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia da Universidade de Coimbra
Edição 2020-2021

Bibliography

- Chapters:
  - 2: Outlier
  - 3: Deduplication
  - 4: Data transformation

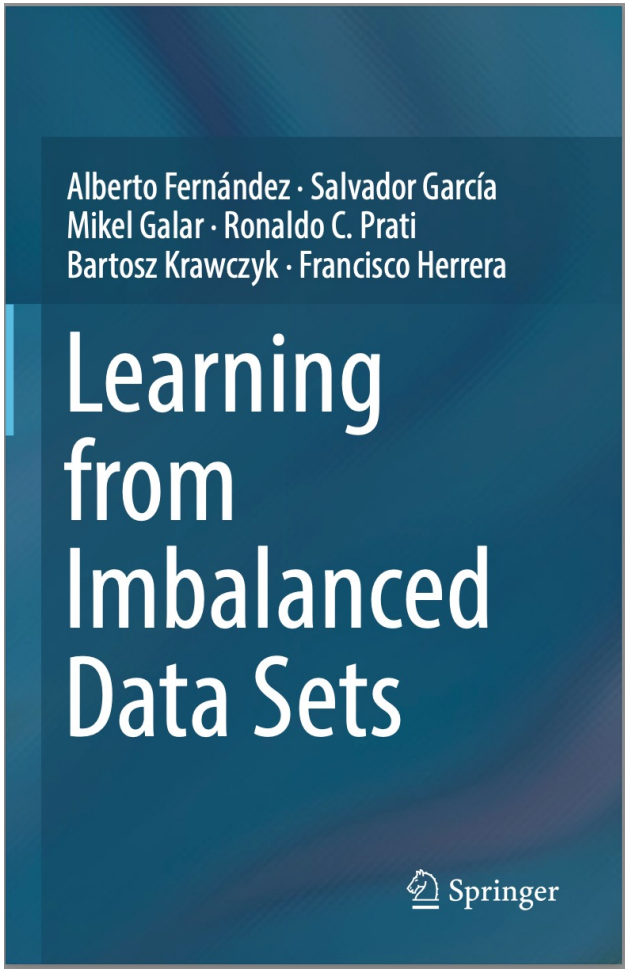


- Chapters:
  - 2: Normalization+Binning+Outliers
  - 3: Imputation

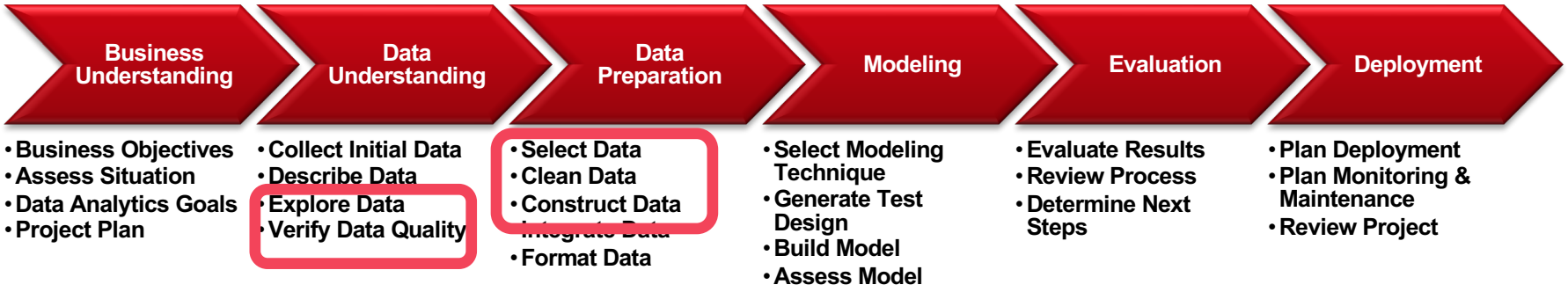


Bibliography

- Chapters:
  - 1: Introduction
  - 2: Foundations of Imbalanced Classification
  - 5: Data Level Preprocessing Methods



Course – Where are we?

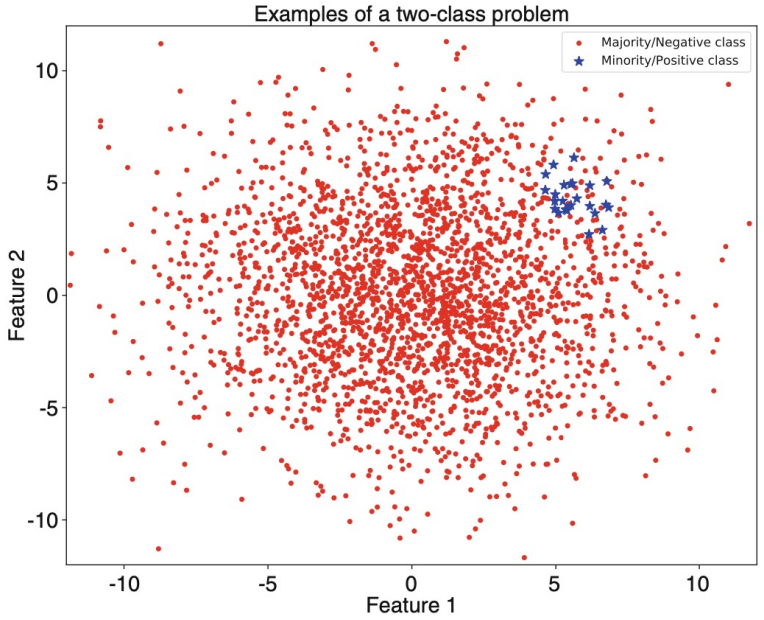


Outline

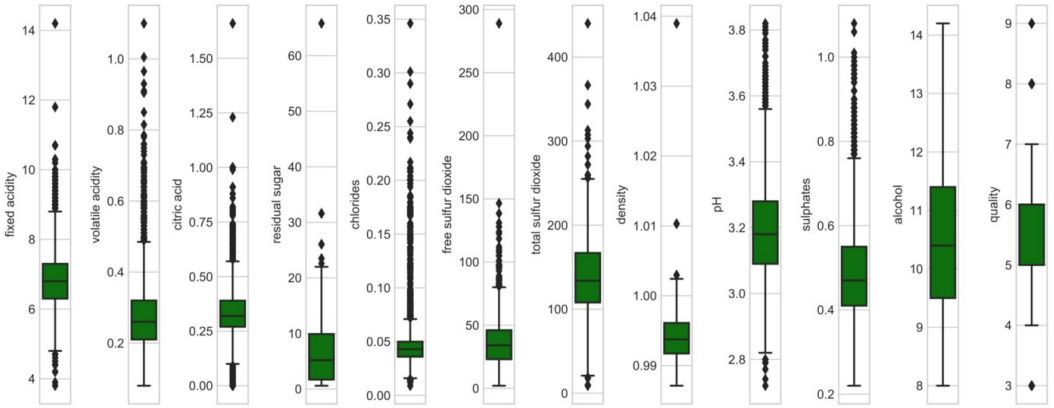
- Data Exploration
- Type of attributes
- Data conversions
- Binning
- Cleaning
  - Outlier
  - Imputation
  - Duplicate
  - Noise in continuous data (will be covered in chapter 4)
- Normalization
- Unbalanced data

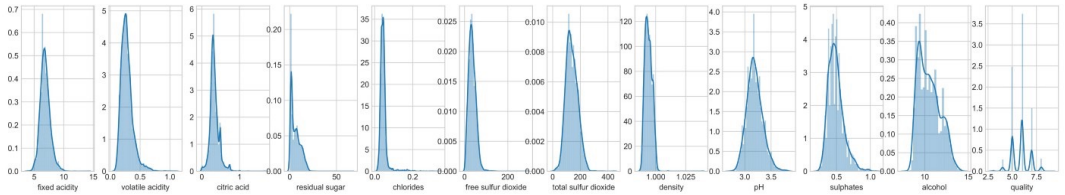
Select Data

- Goals:
  - Identification of available data sources
  - Extraction of data for preliminary analysis
- Tasks:
  - Check for quantity
    - Check for imbalanced data set
  - Check for quality
    - Domain Coverage
      - Descriptive statistics
        - » Max, min
        - » Mean, Median
        - » STD – if low -> little power
        - » Box plot, histogram
    - Discriminant Power
      - Output and inputs available? (usually need a priori information on goals and domain knowledge)
      - Correlation input-output
      - Correlation analysis input-input (correlation matrix) or scatter plots
      - Mutual Information
    - Noise / outliers (e.g. box-plots)
    - Missing values



Select Data

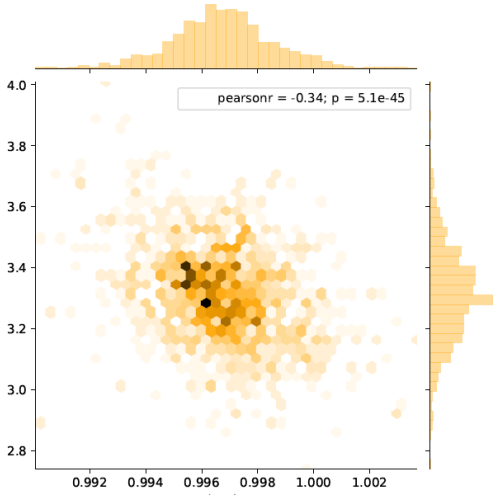




- Goals:
  - Identification of available data sources
  - Extraction of data for preliminary analysis
- Tasks:
  - Check for quantity
    - Check for imbalanced data set
  - Check for quality
    - Domain Coverage
      - Descriptive statistics
        - » Max, min
        - » Mean, Median
        - » STD – if low -> little power
        - » Skewness, pickedness (equally distributed ?)
        - » Box plot, distribution plot
    - Discriminant Power
      - Output and inputs available? (usually need a priori information on goals and domain knowledge)
      - Correlation input-output
      - Correlation analysis input-input (correlation matrix) or scatter plots
      - Mutual Information
    - Noise / outliers (e.g. box-plots)
    - Missing values

Select Data

- Goals:
  - Identification of available data sources
  - Extraction of data for preliminary analysis
- Tasks:
  - Check for quantity
    - Check for imbalanced data set
  - Check for quality
    - Domain Coverage
      - Descriptive statistics
        - » Max, min
        - » Mean, Median
        - » STD – if low -> little power
        - » Box plot, histogram
    - Discriminant Power
      - Output and inputs available? (usually need a priori information on goals and domain knowledge)
      - Correlation input-output
      - Correlation analysis input-input (correlation matrix) or scatter plots
      - Mutual Information
    - Noise / outliers (e.g. box-plots)
    - Missing values



Select Data

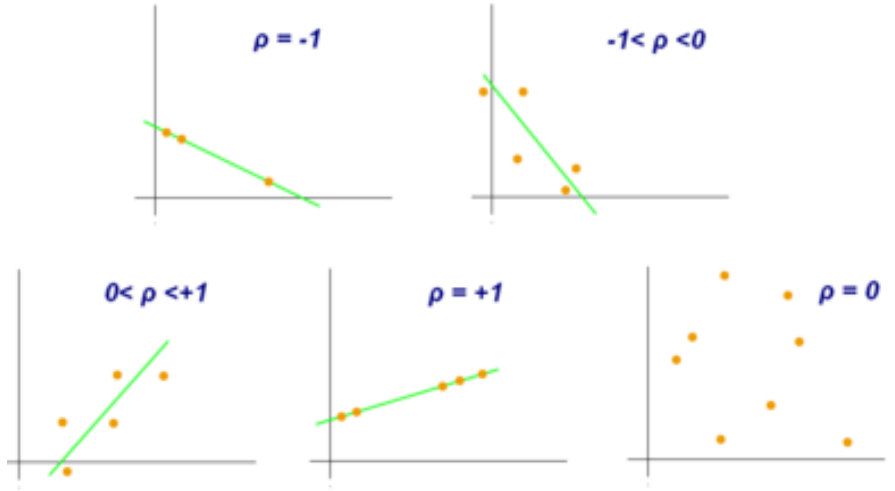
- What Correlation?
- What type of variables?

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 1: Continous-Continous
  - Variables normal distributes? (e.g. Kolmogorov-Smirnov test)
  - Pearson correlation:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$



Select Data

- What Correlation?
- What type of variables?

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 1: Continous-Continous
  - Variables NOT normal distributes? (e.g. Kolmogorov-Smirnov test)
  - Spearman correlation:
    - X and Y are converted into ranks R(X) and R(Y)

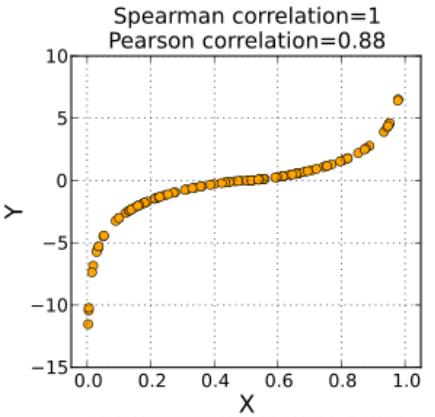
$$r_s = \rho_{\mathrm{R}(X),\mathrm{R}(Y)} = \frac{\operatorname{cov}(\mathrm{R}(X), \mathrm{R}(Y))}{\sigma_{\mathrm{R}(X)} \sigma_{\mathrm{R}(Y)}},$$

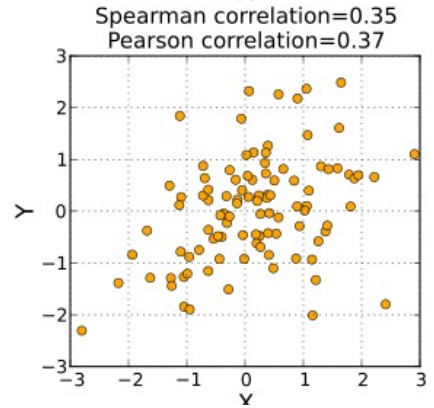
$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

If all ranks are distinct

IQ, X_i	Hours of TV per week, Y_i	rank x_i	rank y_i	d_i	d_i^2
86	2	1	1	0	0
97	20	2	6	–4	16
99	28	3	8	–5	25
100	27	4	7	–3	9
101	50	5	10	–5	25
103	29	6	9	–3	9
106	7	7	3	4	16
110	17	8	5	3	9
112	6	9	2	7	49
113	12	10	4	6	36

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$





is the sum of the non-modal frequencies for each value of the independent v

Select Data

- What Correlation?
- What type of variables?

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 2: Categorical-Categorical
  - Goodman Kruskal Lambda – assesses association between two variables;
    - How much predictive error is reduced of one variable by knowing another?
    - Output range: [0,1]; 0 – no association; 1 – perfect association
    - Assymetric property: $\lambda_{A|B} \neq \lambda_{B|A}$

    - $\lambda_{A|B} = \frac{P_E - P_{E|B}}{P_E}$
    - P_E is the error in predicting A **without** knowing B
    - $P_{E|B}$ is the error in predicting A **with** knowledge of B

    - $\lambda_{B|A} = \frac{P_E - P_{E|A}}{P_E}$
    - P_E is the error in predicting B **without** knowing A
    - $P_{E|A}$ is the error in predicting B **with** knowledge of A

Select Data

- $\lambda_{B|A} = \frac{P_E - P_{E|A}}{P_E}$
- P_E is the error in predicting B **without** knowing A
- $P_{E|A}$ is the error in predicting B **with** knowledge of A

ID	A	B
1	OFF	ON
2	ON	OFF
3	ON	ON
…	…	…
1000	OFF	ON

A

B	Off	On	Total
Off	118	182	300
On	72	628	700
Total	190	810	

B

A	Off	On	Total
Off	118	72	190
On	182	628	810
Total	300	700	

is the sum of the non-modal frequencies for each value of the independent v

Select Data

- *Question: How is B explained by A?*
- *B – Dependent variable*
- *A - Independent variable*
- $\lambda_{B|A} = \frac{P_E - P_{E|A}}{P_E}$
- P_E is the error in predicting B **without** knowing A
- $P_{E|A}$ is the error in predicting B **with** knowledge of A

B ← dependent variable

	Off	On	Total
Off	118	72	190
On	182	628	700
Total	300	700	

A ← independent variable

- P_E is the error in predicting B without knowing A
- What is the best guess we can do about B without knowing A?
  - Best guess is B=ON => Pcorrect B = 700/1000; P_E **= 1-Pcorrect = 0.3**
- $P_E = 1 - \frac{\max_C N_{+C}}{N_{++}} = 1 - \frac{700}{300+700}$

Select Data

- $Question: How\ is\ B\ explained\ by\ A?$
- B – *Dependent variable*
- A - *Independent variable*
- $\lambda_{B|A} = \frac{P_E - P_{E|A}}{P_E}$
- P_E is the error in predicting B **without** knowing A
- $P_{E|A}$ is the error in predicting B **with** knowledge of A

B ← dependent variable

A ← independent variable

	Off	On	Total
Off	118	72	190
On	182	628	700
Total	300	700	

- $P_{E|A}$ is the error in predicting B with knowledge of A
- What is the best guess we can do about B with knowledge A?
  - If A = OFF
    - Best guess is B=OFF => Pcorrect B|A=OFF = 118/1000;
  - If A = ON
    - Best guess is B=ON => Pcorrect B|A=ON = 628/1000;
  - Pcorrect B|A = Pcorrect B|A=ON+ Pcorrect B|A=OFF

- $P_{E|A} = 1 - \frac{\sum_r \max_C N_{rC}}{N_{++}} = 1 - \frac{118+628}{300+700}$

is the sum of the non-modal frequencies for each value of the independent v

Select Data

		Relationship Status		Total
		Unmarried	Married	
Blood Pressure	Normal	80% (120)	51% (102)	63.4% (222)
	High	20% (30)	49% (98)	36.6% (128)
Total		42.9% (150)	57.1% (200)	100% (350)

Question: "can the relationship status be predicted better if the blood pressure is known?"

Relationship status dependent variable,
Blood pressure is the independent variable

$$\lambda_{Relationship\ status|Blood\ Pressure} = \frac{P_E - P_{E|Blood\ Pressure}}{P_E}$$

- P_E is the error in predicting RELATIONSHIP STATUS without knowing BLOOD PRESSURE
- What is the best guess we can do about RELATIONSHIP STATUS without knowing BLOOD PRESSURE?
- Best guess is RS=MARRIED => Pcorrect RS = 200/350; P_E **= 1-Pcorrect = 1-2/3.5**
- $P_E = 1 - \frac{\max_C N_{+C}}{N_{++}} = 1 - \frac{200}{150+200} = \frac{150}{350}$

Select Data

		Relationship Status		Total
		Unmarried	Married	
Blood Pressure	Normal	80% (120)	51% (102)	63.4% (222)
	High	20% (30)	49% (98)	36.6% (128)
Total		42.9% (150)	57.1% (200)	100% (350)

Question: "can the relationship status be predicted better if the blood pressure is known?"

Relationship status dependent variable,
Blood pressure is the independent variable

$$\lambda_{Relationship\ status|Blood\ Pressure} = \frac{P_E - P_{E|Blood\ Pressure}}{P_E}$$

- $P_{E|BP}$ is the error in predicting RS with knowledge of BP
- What is the best guess we can do about RS with knowledge BP?
  - If BP = NORMAL
    - Best guess is RS=UNMARIED => Pcorrect RS|BP = NORMAL = 120/350;
  - If BP = HIGH
    - Best guess is RS=MARIED => Pcorrect RS|BP = HIGH = 98/350;
  - Pcorrect RS|BP = Pcorrect RS|BP=NORMAL+ Pcorrect RS|BP=HIGH

- $P_{E|BP} = 1 - \frac{\sum_r \max_C N_{rC}}{N_{++}} = 1 - \frac{120+98}{150+200} = \frac{132}{350}$
- $\lambda_{Relationship\ status|Blood\ Pressure} = \frac{P_E - P_{E|BP}}{P_E} = \frac{150-132}{150}$=0.12

Select Data

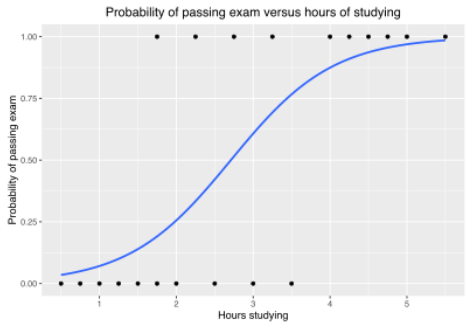
- What Correlation?
- What type of variables?

- Situation 3: Continuous-Categorical

	Categorical	**Continuous**
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

  - Use logistic regression (see classification using logistic regression in MECD):
    - If there is a relationship between the categorical and continuous variable, we should be able to construct an accurate predictor of the categorical variable from the continuous variable.
    - If the resulting classifier has a high degree of fit, is accurate, sensitive, and specific we can conclude the two variables share a relationship and are indeed correlated.

  - Advantage:
    - Does not require continuous variable to be normally distributed
    - Does not require continuous variable to have equal variance in each group
    - Does not imply linear relationship between categorical and continuous variable
  - Disadvantage:
    - Is sensitive to class imbalance
    -



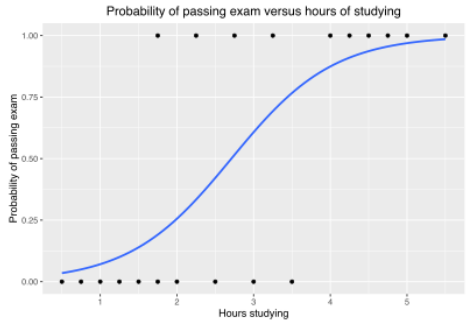
Select Data

- What Correlation?
- What type of variables?

- Situation 3: Continuous-Categorical

	Categorical	**Continuous**
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Lets model our probability using a logistic function (for ouput equal to 1)
- $p(\boldsymbol{x}) = \frac{1}{1+e^{-\beta_0 - \sum_{i=1}^{n} \beta_i x_i}}$
- Where n is the number of independent variables $\boldsymbol{x} = [x_1 \quad \dots \quad x_n]$.
- Let y be the dependent variable
- $E[y|\boldsymbol{x}] = odds(\boldsymbol{x}) \equiv \left(\frac{p(\boldsymbol{x})}{1-p(\boldsymbol{x})}\right) = e^{\beta_0 + \sum_{i=1}^{n} \beta_i x_i}$
- Furthermore, the odds ratio is defined by
- $OR(x_i) \equiv \left(\frac{odd(x_i+1)}{odd(x_i)}\right) = e^{\beta_i}$
  - the odd multiply by e^{β_i} for each increase of one unit in x_i, assuming all other independent variables are kept equal



Select Data

- What Correlation?
- What type of variables?

	Categorical	**Continuous**
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 3: Continuous-Categorical

- How to compute:
  - Use the cross-entropy between the {y, 1-y} distribution and the estimated {p(x), 1-p(x)}

$$\ell_k = -y_k \ln p_k - (1 - y_k) \ln(1 - p_k).$$

  - Solution is numeric

Select Data

- What Correlation?
- What type of variables?

- Situation 3: Continuous-Categorical

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Example: Flight Carrier Survey Data
- x1 – Seat Confort (scale: 1-5)
- x2 - Cabin Staff (scale: 1-5)
- x3 - Food & Beverage (scale: 1-5)
- x4 - Ground Service (scale: 1-5)
- x5 - Inflight Entertainment (scale: 1-5)
- X6 - WiFi & Connectivity (scale: 1-5)
- Y – Recommendation (Yes/No)

	$ln\left(\frac{p(recommend = yes)}{p(recommend = no)}\right)$
β_0	-9.496271
β_1	0.717416
β_2	0.511033
β_3	0.430392
β_4	0.945390
β_5	0.033929 (p=0.683820)
β_6	0.259780

Select Data

- What Correlation?
- What type of variables?

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 3: Continuous-Categorical

- What if our independente variable is categorical:

- One hot encoding
  - Convert each category into one input variable
  - Do not use sequential numbering -> leads to diferente relevances

Color
Red
Green
Blue

→

Color	Red	Green	Blue
Red	1	0	0
Green	0	1	0
Blue	0	0	1

Select Data

	Categorical	Continuous
Categorical	Lambda, Corrected Cramer's V	Point Biserial, Logistic Regression
Continuous	Point Biserial, Logistic Regression	Spearman, Kendall, Pearson

- Situation 3: Continuous-Categorical
- What if our dependente variable is categorical with more than two categories:
  - Analyse $ln\left(\frac{p(y \leq cati)}{p(y > cati)}\right), i = 1,2, \ldots n$
  - E.g. “Value for Money” categorical variable = 1,2,3,4,5

	$ln\left(\frac{p(y \leq 1)}{p(y > 1)}\right)$	$ln\left(\frac{p(y \leq 2)}{p(y > 2)}\right)$	$ln\left(\frac{p(y \leq 3)}{p(y > 3)}\right)$	$ln\left(\frac{p(y \leq 4)}{p(y > 4)}\right)$
β_0	14.9464 (p<0.05)	10.9636 (p<0.05)	8.7419 (p<0.05)	4.8272 (p<0.05)
β_1	-1.2773 (p<0.05)	-0.9118 (p<0.05)	-0.7672 (p<0.05)	-0.2711 (p<0.05)
β_2	-0.7900 (p<0.05)	-0.5855 (p<0.05)	-0.4865 (p<0.05)	-0.4513 (p<0.05)
β_3	-0.7434 (p<0.05)	-0.5286 (p<0.05)	-0.3010 (p=0.051)	-0.0825 (p<0.391)
β_4	-1.5056 (p<0.05)	-0.9763 (p<0.05)	-0.7056 (p<0.05)	-0.3349 (p<0.05)
β_5	-0.0920 (p=0.393)	0.0107 (p=0.920)	-0.0018 (p=0.986)	0.0851 (p=0.335)
β_6	-0.4293 (p<0.05)	-0.2684 (p0.056)	-0.2521(p=0.058)	-0.2357 (p<0.05)

Types of Measurements

- Nominal/Categorical scale
- Ordinal scale

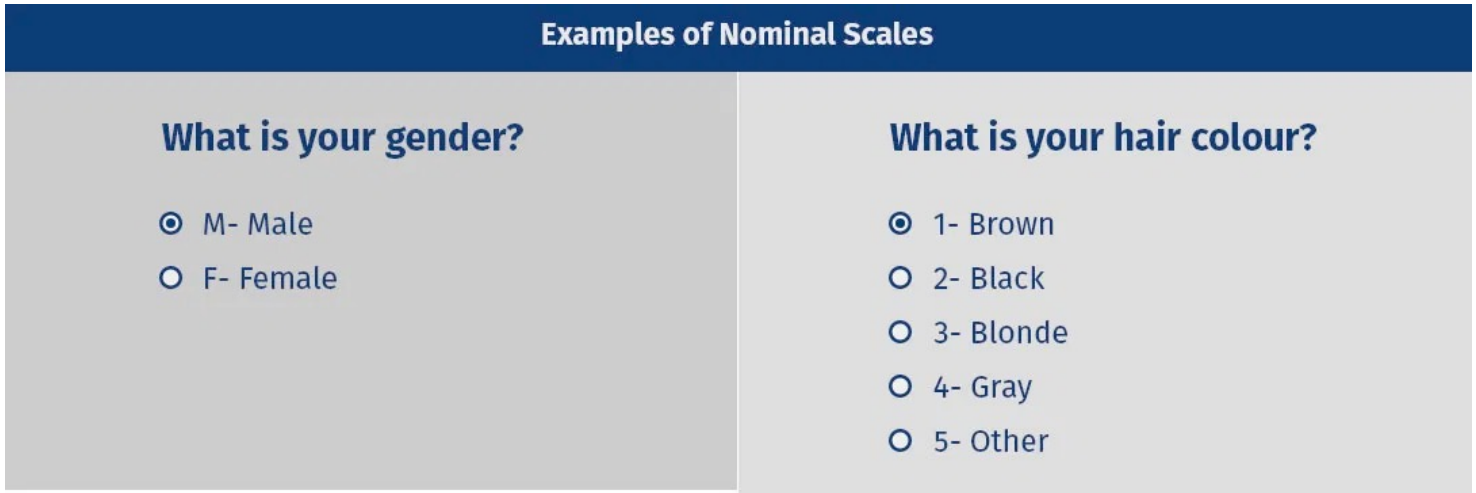
Qualitative

- Interval scale
- Ratio Scale

Quantitative

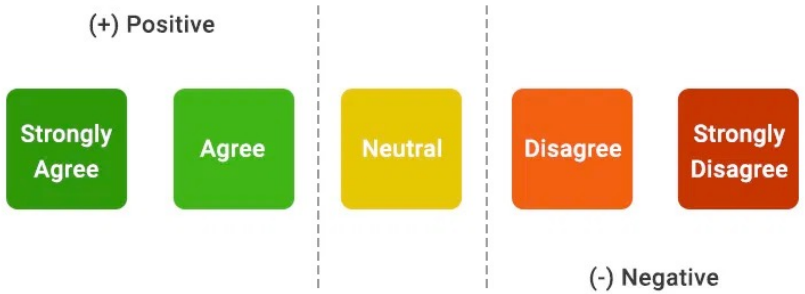
Types of Measurements

- Nominal scale
  - Variable is divided into two or more categories
  - Numbers or letters serve as TAGS
  - Nominal scale is qualitative in nature



Types of Measurements

- Ordinal scale
  - **Reports the ranking** and ordering of the data without actually establishing the **degree of variation between** them.
  - Along with identifying and describing the magnitude, the ordinal scale shows the relative rank of variables.
  - **Measurement of non-numeric attributes** such as frequency, satisfaction, happiness etc.
  - In addition to the information provided by nominal scale, ordinal scale identifies the rank of variables.



Types of Measurements

- Interval scale
  - Is **quantitative** in nature
  - It measures variables that exist along a **common scale at equal intervals**
  - Presence of **zero is arbitrary**
  - You can **subtract values between two variables** that help understand the difference between two variables. Interval measurement allows you to calculate the mean and median of variables.
  - Calendar dates, temperature in Celsius, GRE (Graduate Record Examinat

Types of Measurements

- Ratio scale
  - Is quantitative in nature
  - Allows to **compare** the intervals or differences
  - **Has an absolute zero** characteristic (our reference)
  - **All statistical analysis** including mean, mode, the median can be calculated using ratio scale.
  - Temperature in Kelvin, Length, time, counts,…

Data Conversion

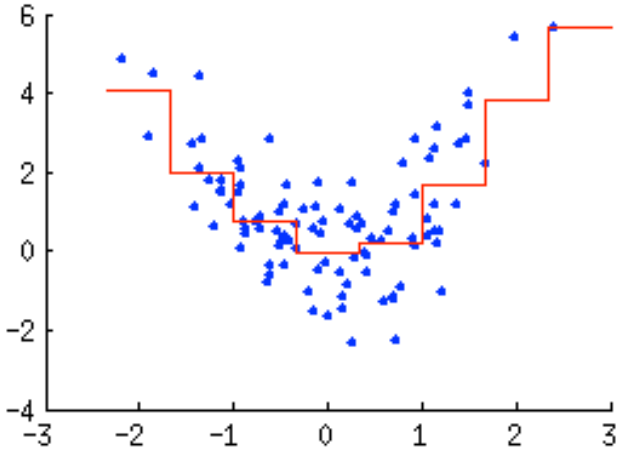
- Most tools require **Nominal** values to be **numeric**

- Group nominal fields “naturally”:
  - E.g. 50 US states -> 3 or 5 regions
  - Professions – select the most frequent ones, group the rest

- Convert ordinal fields to numeric to be able to use “>” and “<“ comparisons

Binning – Discretization of continuous variables

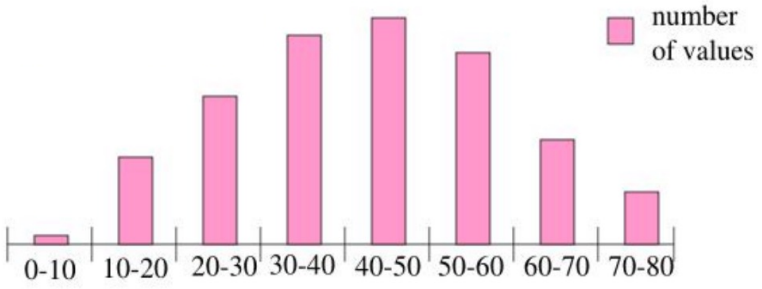
- Divide the range of continuous attributes into intervals
  - Some methods require discrete values (e.g. many versions of Naïve Bayes)
  - Reduce data size (Float->Byte)
  - Prepare for further analysis (e.g. statistical visualization)



Binning (unsupervised)

- Equal-width Binning
  - Uniform grid: Divide the range into N intervals of equal size
  - Let A and B be the lowest and the highest values in the attributes

$$width = \frac{B - A}{N}$$

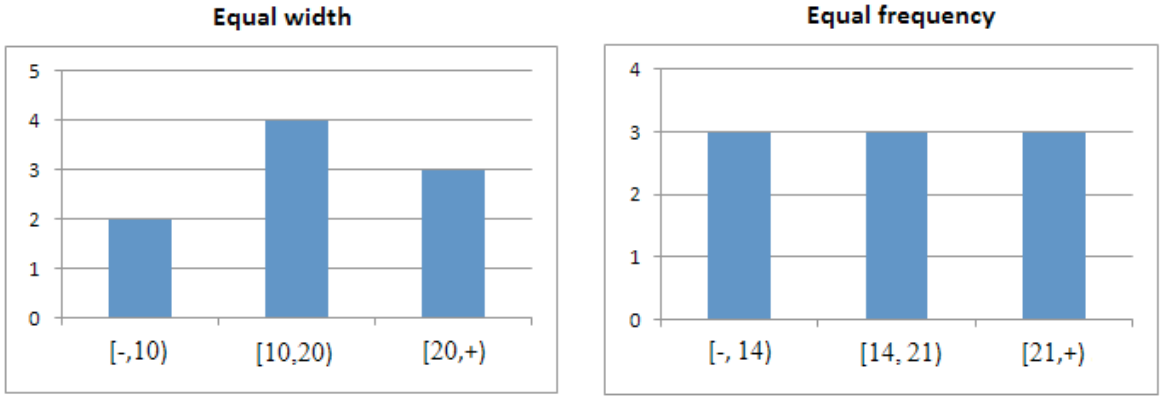


  - Advantages: simple, produces reasonable abstraction of values
  - **Disadvantages**: N = ?, Sensitive to outliers

Binning

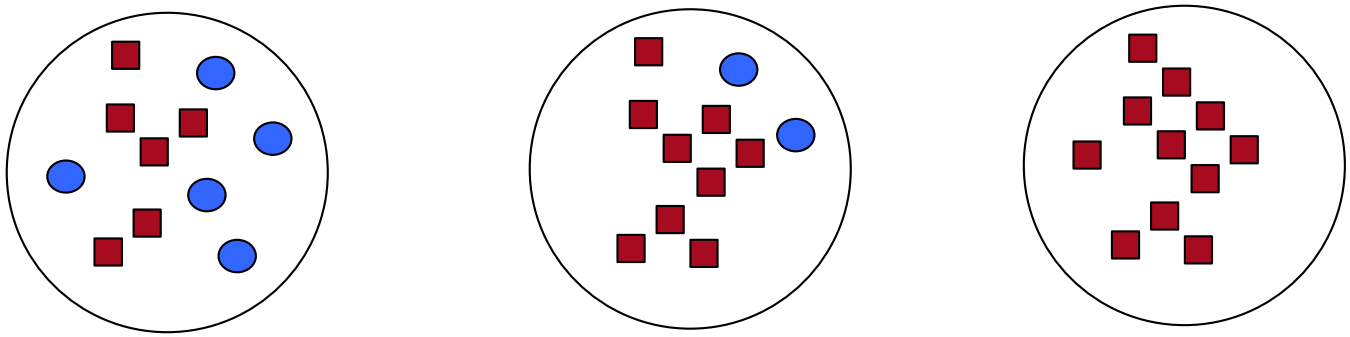
- Equal-depth (or height or frequency) Binning
  - Data is divided into N intervals, such that each contains approx. the same number of samples
  - In practice “almost-equal” height binning is used to give more intuitive breakpoints

- Additional considerations
  - Create separate bins for special values
  - Readable breakpoints (e.g., round breakpoints)



Binning - Supervised

- Classes are known for each point
- How to select boundaries?

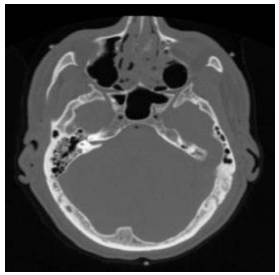


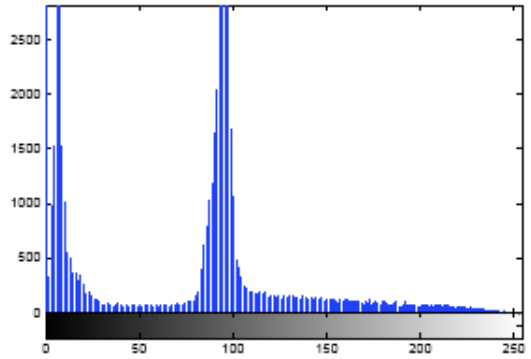
Binning (supervised)

- Entropy-based Binning
  - Entropy: measure of randomness in a set

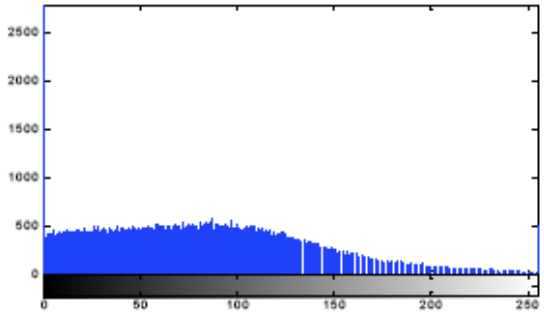
$$A \equiv \{a_1, a_2, ..., a_n\}$$

$$H(A) = -\sum_{i=1}^{n} P(a_i) \log_2 P(a_i)$$





H=5.94 bits/p



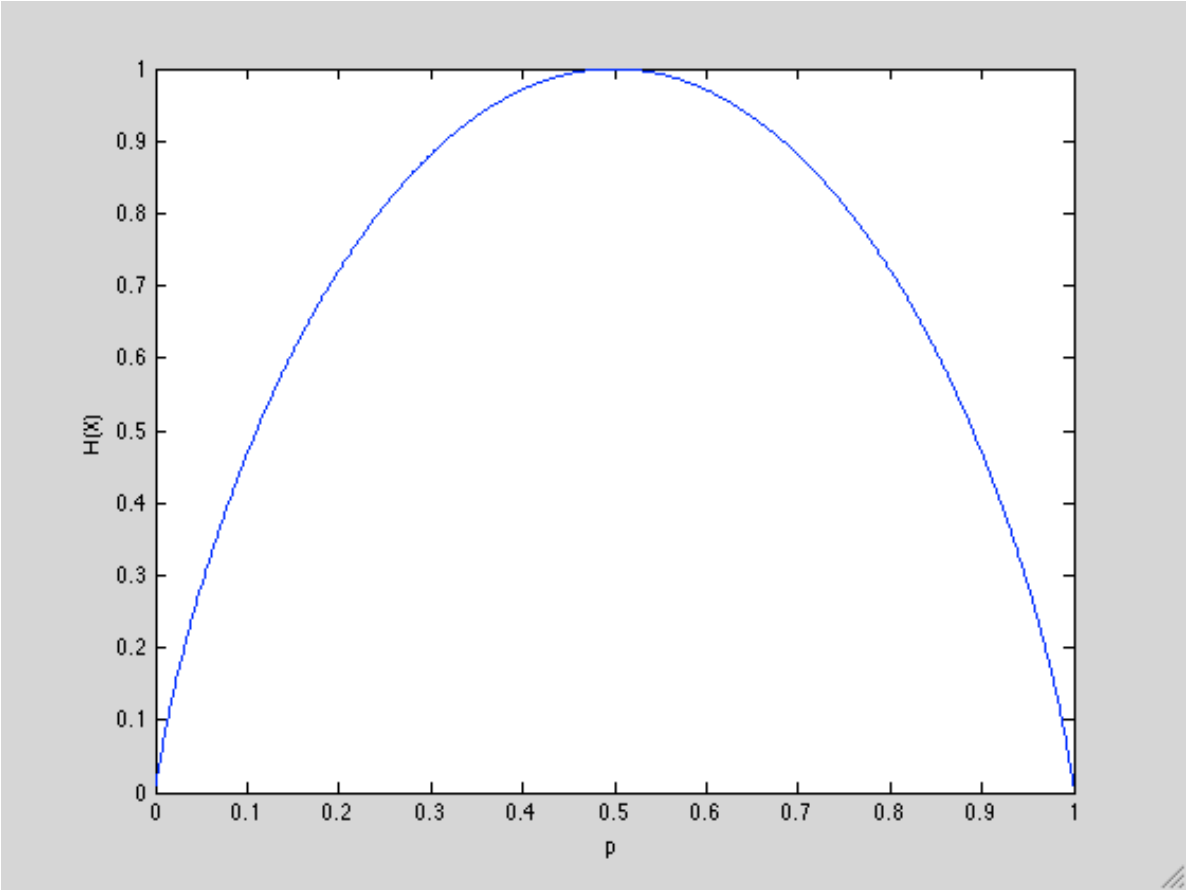
H=6.96 bits/p

Binning

- Entropy & Uncertainty

$$X = \begin{cases} 1 \Leftarrow P(X=1) = p \\ 0 \Leftarrow P(X=0) = 1-p \end{cases}, p \in [0,1]$$

$$H(X) = -p\log_2 p + (1-p)\log_2(1-p)$$

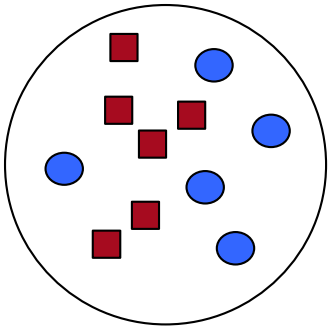


Binning (Supervised)

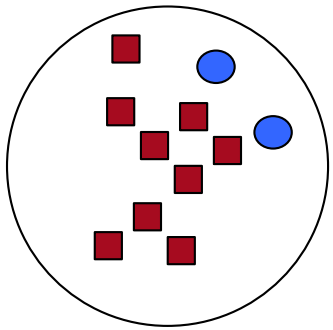
- Entropy / Impurity
  - Entropy measures de impurity or the uncertainty in a group
  - S – Training set with $C_1, \ldots, C_N$ classes
  - m_i – number of values in the *ith* interval
  - m_{ij} – number of values of class j in the *ith* interval

$$p_{ij} = \frac{m_{ij}}{m_i}$$

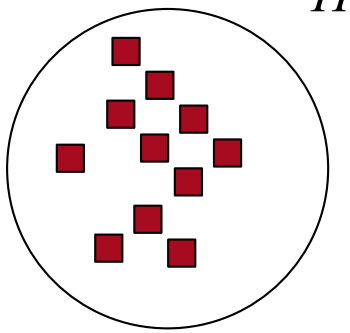
$$H\left(S_i\right) = -\sum_{i=1}^{N} p_{ij} \log_2 p_{ij}$$



High impurity: H(S)=0.99



Less impurity: H(S)=0.61



Minimum impurity: H(S)=0

Binning

- Entropy-based binning
  - Minimize the impurity of the discretization
  - S – Training set with $C_1, \ldots, C_N$ classes
  - m_i – number of values in the *ith* interval
  - m_{ij} – number of values of class j ($j=1,...,N$) in the *ith* interval

1. Sort examples in increasing order to obtain set S

2. Calculate the Entropy of this discretization (set S)
3. Find the binary split boundary that minimizes the entropy function over all possible boundaries.

$$H(S,T) = \sum_{i=1}^{K} \frac{|S_i|}{|S|} H(S_i)$$

4. Apply the process recurseviley and select T that maximizes the information gain

$$Gain = H(S) - H(S,T)$$

Binning

- Entropy-based binning
- Set: $S \equiv \{(16,N),(26,N),(0,Y),(4,Y),(12,Y),(16,N),(18,Y),(24,N),(28,N)\}$
  1. Sort examples in increasing order to obtain set S

     $$S \equiv \{(0,Y),(4,Y),(12,Y),(16,N),(16,N),(18,Y),(24,N),(26,N),(28,N)\}$$

  2. Calculate the Entropy of this discretization (set S)

     $$p_Y = \frac{4}{9}, p_N = \frac{5}{9}, H(S) = -p_Y \log_2 p_Y - p_N \log_2 p_N = 0.99$$

  3. Find the binary split boundary that minimizes the entropy function over all possible boundaries. ***Let us start with T=14***

     $$S_1 \equiv \{(0,Y),(4,Y),(12,Y)\} \qquad S_2 \equiv \{(16,N),(16,N),(18,Y),(24,N),(26,N),(28,N)\}$$

     $$p_{1,Y} = \frac{3}{3} = 1, p_{1,N} = \frac{0}{3} = 0, H(S_1) = 0 \qquad p_{2,Y} = \frac{1}{6}, p_N = \frac{5}{6}, H(S_2) = 0.650$$

     $$H(S,T) = \sum_{i=1}^{2} \frac{|S_i|}{|S|} H(S_i) = \frac{3}{9} H(S_1) + \frac{6}{9} H(S_2) = \frac{3}{9} \times 0 + \frac{6}{9} \times 0.650 = 0.433$$

     $$Gain = H(S) - H(S,T) = 0.557$$

Binning

- Entropy-based binning (Example 2)

Steps	Healthy
640	Yes
1650	No
1700	Yes
1800	Yes
2000	Yes
2500	No
3000	No
3000	Yes
4050	Yes
4050	Yes
5000	No
6000	Yes
6500	Yes
7000	No

Sixth Split

Fifth Split

Fourth Split

Third Split

Second Split

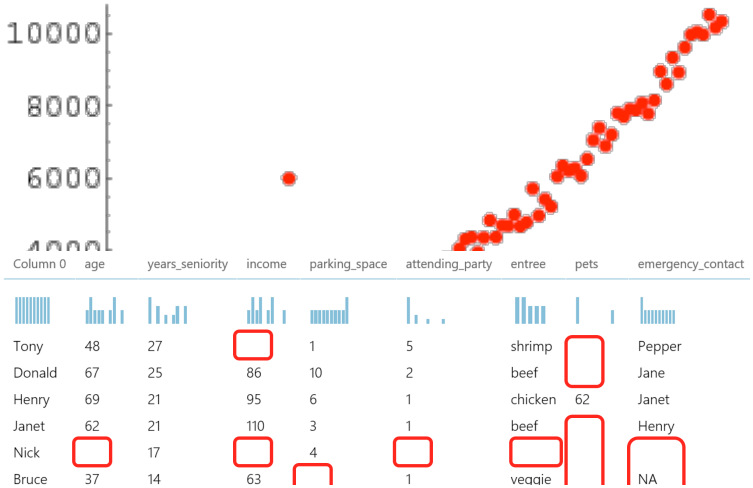
First Split

In each iteration the two resulting branches are processed recursively

In this example, recursion only occurs in one of the intervals. This is an artifact.

Cleaning of Data

- Main cleaning operations
  - Outliers
  - Missing values
  - Correct, remove or ignore noise
    - We shall detail this after Fourier Analysis
  - Duplicate removal



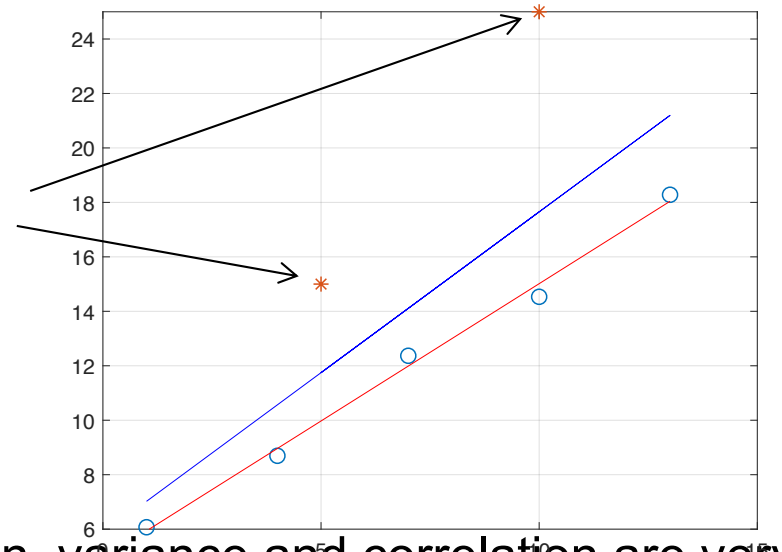
id	first_name	last_name	email
1	Carine	Schmitt	carine.schmitt@verizon.net
4	Janine	Labrune	janine.labrune@aol.com
6	Janine	Labrune	janine.labrune@aol.com
2	Jean	King	jean.king@me.com
12	Jean	King	jean.king@me.com
5	Jonas	Bergulfsen	jonas.bergulfsen@mac.com
10	Julie	Murphy	julie.murphy@yahoo.com
11	Kwai	Lee	kwai.lee@google.com
3	Peter	Ferguson	peter.ferguson@google.com
9	Roland	Keitel	roland.keitel@yahoo.com
14	Roland	Keitel	roland.keitel@yahoo.com
7	Susan	Nelson	susan.nelson@comcast.net
13	Susan	Nelson	susan.nelson@comcast.net
8	Zbyszek	Piestrzeniewicz	zbyszek.piestrzeniewicz@att.net

Outliers

- What are outliers?
- "*An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism*"

$$\arg\min\left(\left\|y-\hat{f}\left(x\right)\right\|^2\right)$$

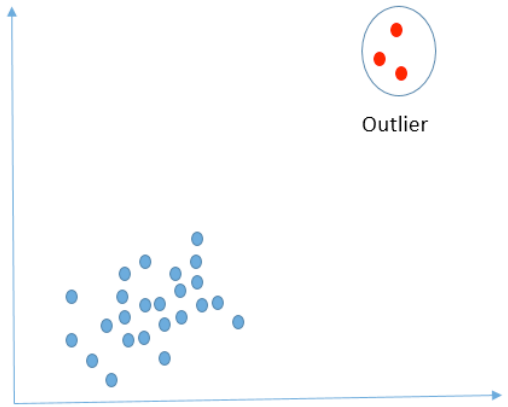
Outlier



- Statistical measures such as mean, variance and correlation are very susceptible to outliers
- ML algorithms are very susceptible to outliers

Outliers

- Origins of outliers
  - **Genuine extreme** high and low values in the dataset
  - Introduced due to **human or mechanical error**
  - Introduced by **replacing missing values**
  - **Rare** “highly relevant” cases
    - E.g., fraud detection

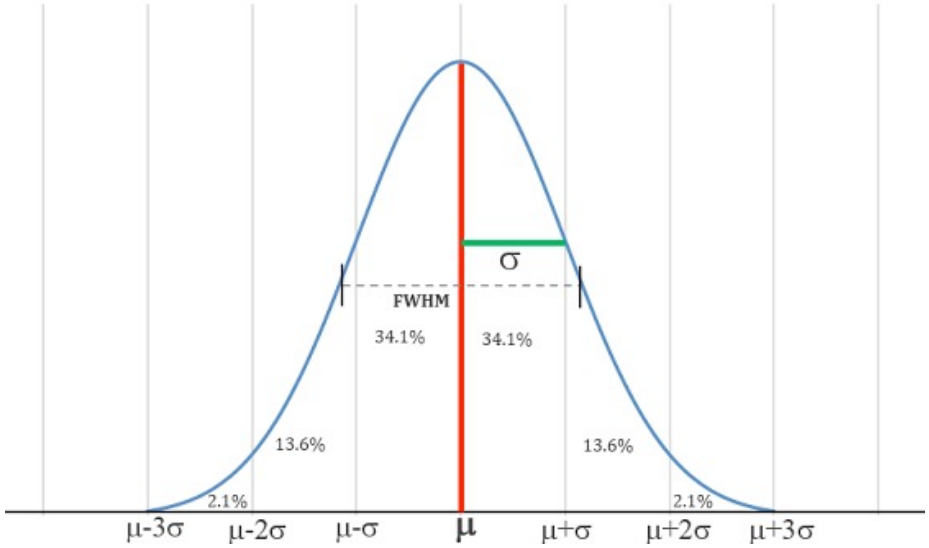


- Algorithms for detection
  - Extreme Value Analysis
  - Z-score method
  - Clustering-based approach
  - Visualizing the data
- Algorithms for treatment
  - Do nothing
  - Imputation
  - Top, Bottom and Zero Coding

Outliers

- Univariate
- Extreme value analysis
  - Determine the statistical tails of the underlying distribution of the variable and find the values at the extreme end of the tails.
  - If a Gaussian Distribution (e.g., use Kolmogorov-Smirnov):
    - Compute the mean and the std. deviation

$$[\mu - k\sigma, \mu + k\sigma]$$

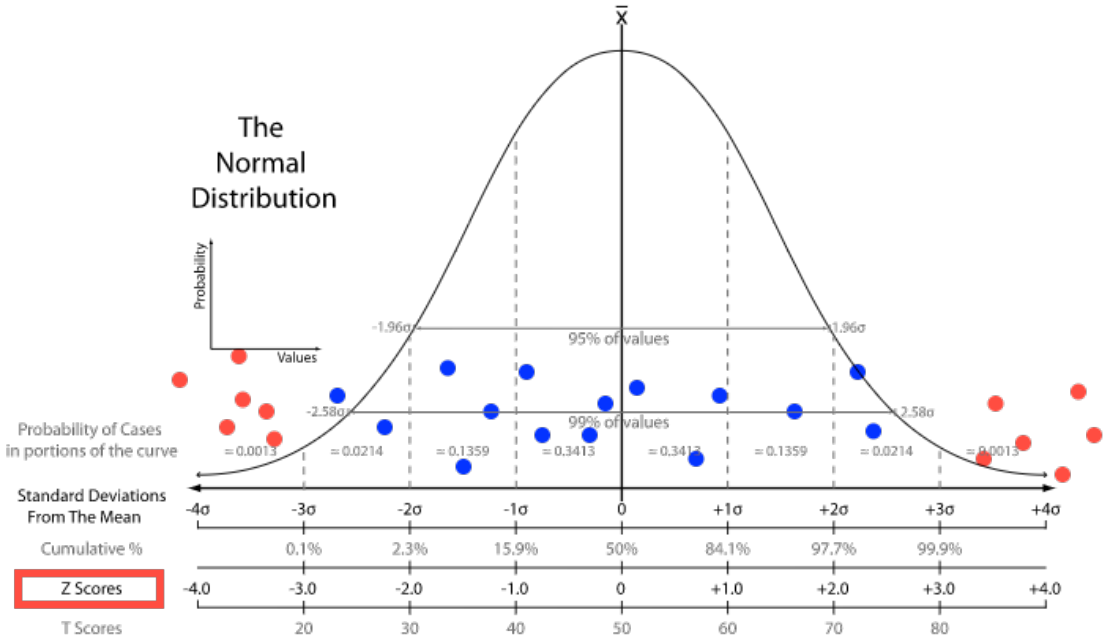


Outliers

- Univariate
- Z-Score: represents how many standard deviations a given measurement deviates from the mean

$$Z = \frac{X - \mu}{\sigma}$$

$$|Z| < k$$



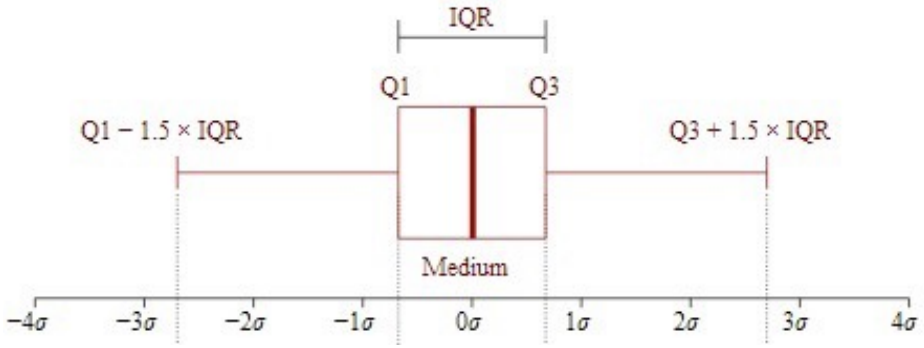
- K=3

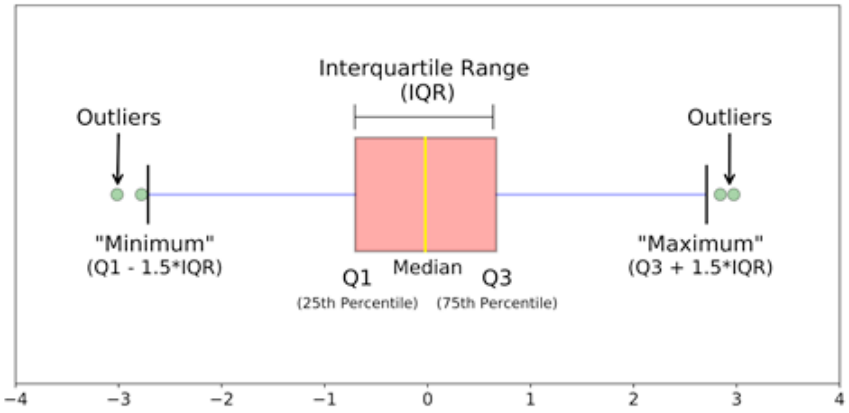
Outliers

- Univariate
- Extreme value analysis
  - Determine the statistical tails of the underlying distribution of the variable and find the values at the extreme end of the tails.
  - If not Gaussian Distribution:

$$[Q_1 - k \times IQR, Q_3 + k \times IQR]$$

  - Mild outlier:
    - outside k=1.5
    - inside: k=3



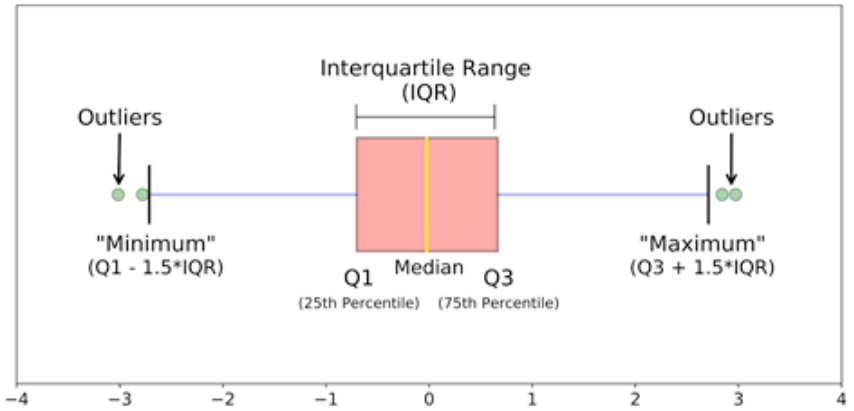


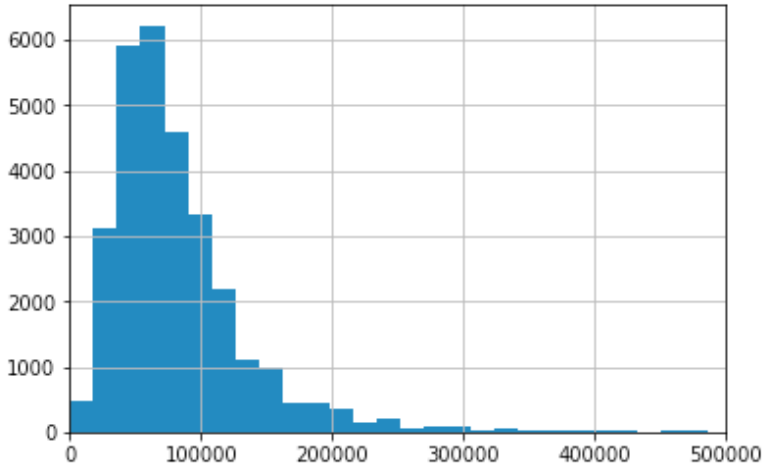
Outliers

- Univariate – Visual Analysis
- Box Plots or Whisker's plots

$$[Q_1 - k \times IQR, Q_3 + k \times IQR]$$

- Histograms

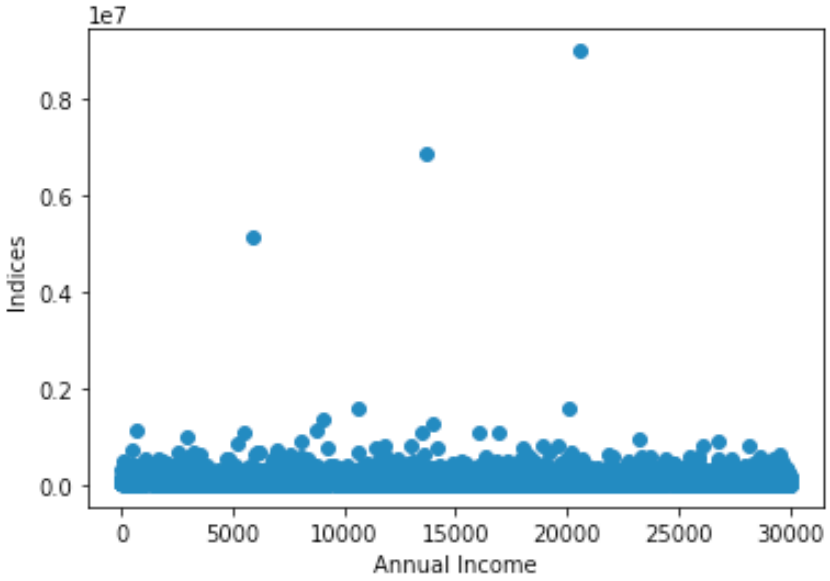




Outliers

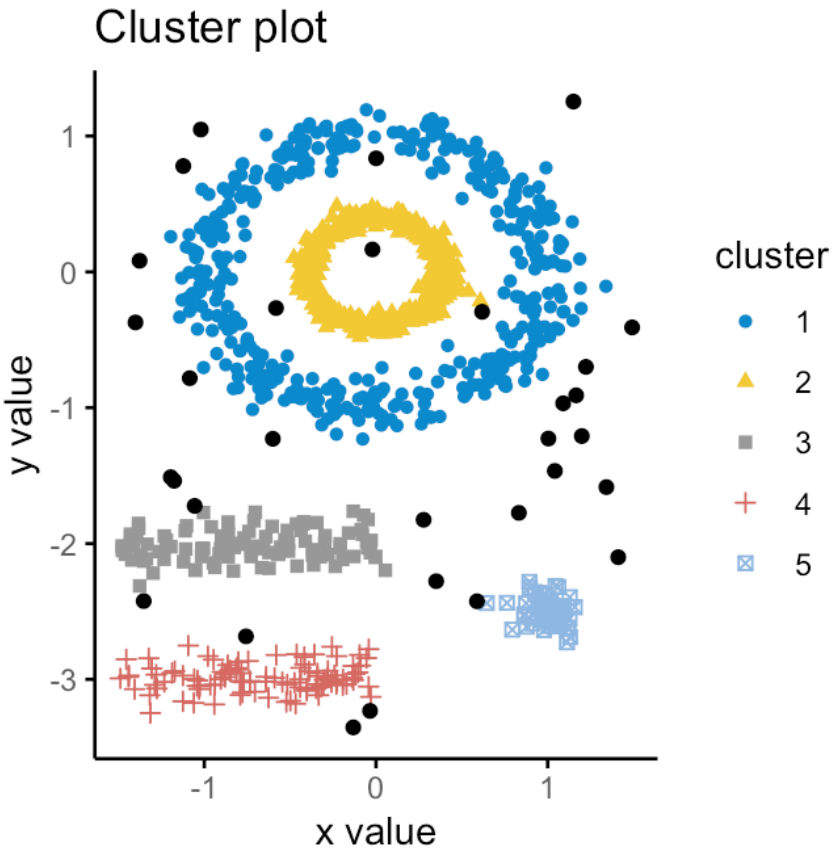
- Bivariate – Visual Analysis

- Scatter plots
  - Outliers do not fit the “pattern”



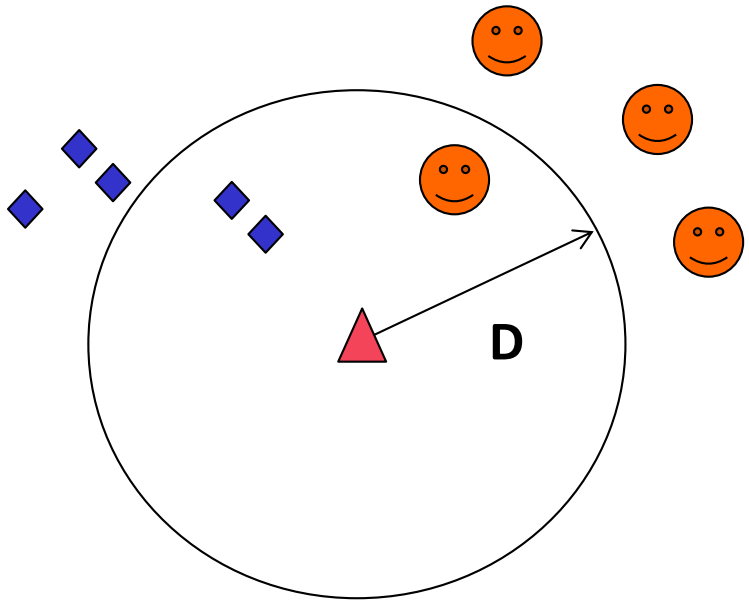
Outliers

- Multivariate
- Clustering: very small clusters are outliers



Outliers

- Multivariate
- Criteria:
  - Distance-based: an instance with very few neighbors within D is regarded as an outlier



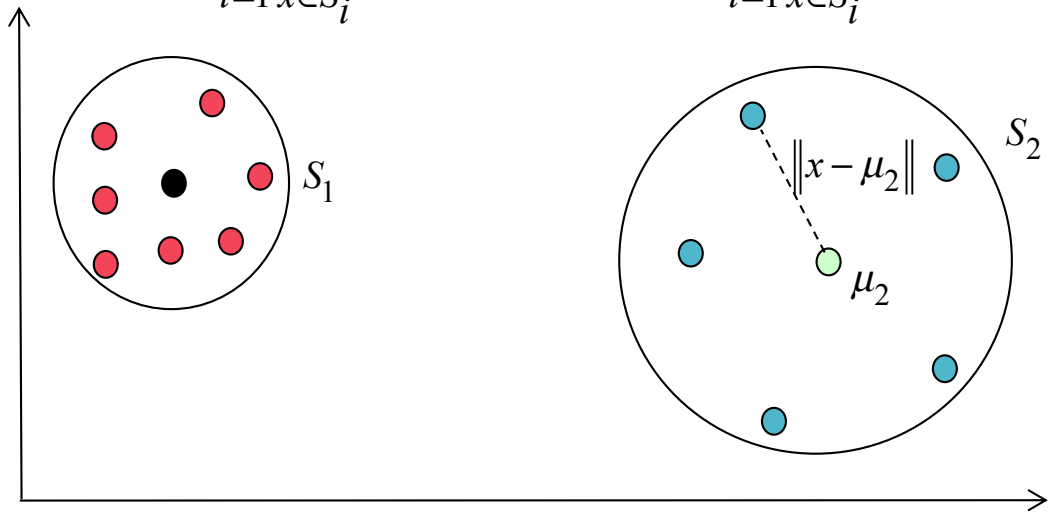
Outliers

- What clustering algorithms?
  - K –means
    - Simple
    - Clusters have regular shapes
    - Number of clusters has to be known
  - DBSCAN
    - More complex
    - Number of clusters is not required a priori

Outliers

- K Means:
  - Unsupervised
  - Cluster *N* observations points into *K* clusters
  - Each observation is allocated to the cluster with the nearest mean
  - *k*-means clustering minimizes within-cluster variances

$$\underset{S}{\operatorname{argmin}} \sum_{i=1}^{k} \sum_{x \in S_i} \left\| x - \mu_i \right\|^2 = \underset{S}{\operatorname{argmin}} \sum_{i=1}^{k} \sum_{x \in S_i} \left| S_i \right| \operatorname{var}\left(S_i \right)$$



Outliers

- K Means:

  Given a set of observations ($\mathbf{x_1}$, $\mathbf{x_2}$, $\mathbf{x_3}$,…, x_N), partition the N observations into $k(\leq N)$

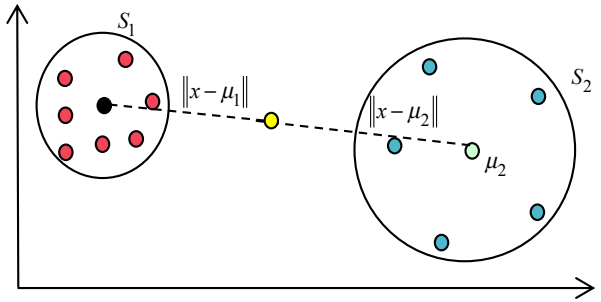
  1.Step 1: select *k* initial cluster centers $\left(\mu_1^{(1)}, \mu_2^{(1)}, \ldots, \mu_k^{(1)}\right)$

  2.Step 2: (Assignment) Each observation $\boldsymbol{x_j}$ is allocated to the cluster S_i with the closest center

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - \mu_i^{(t)} \right\|^2 \leq \left\| x_p - \mu_j^{(t)} \right\|, \forall j, 1 \leq j \leq k \right\}$$

  3.Step 3: (Update)

$$\mu_i^{(t+1)} = \frac{1}{\left| S_i^{(t)} \right|} \sum_{x_j \in S_i^{(t)}} x_j$$



  4.Step 4: Goto step 2, unless the assignments have not changed

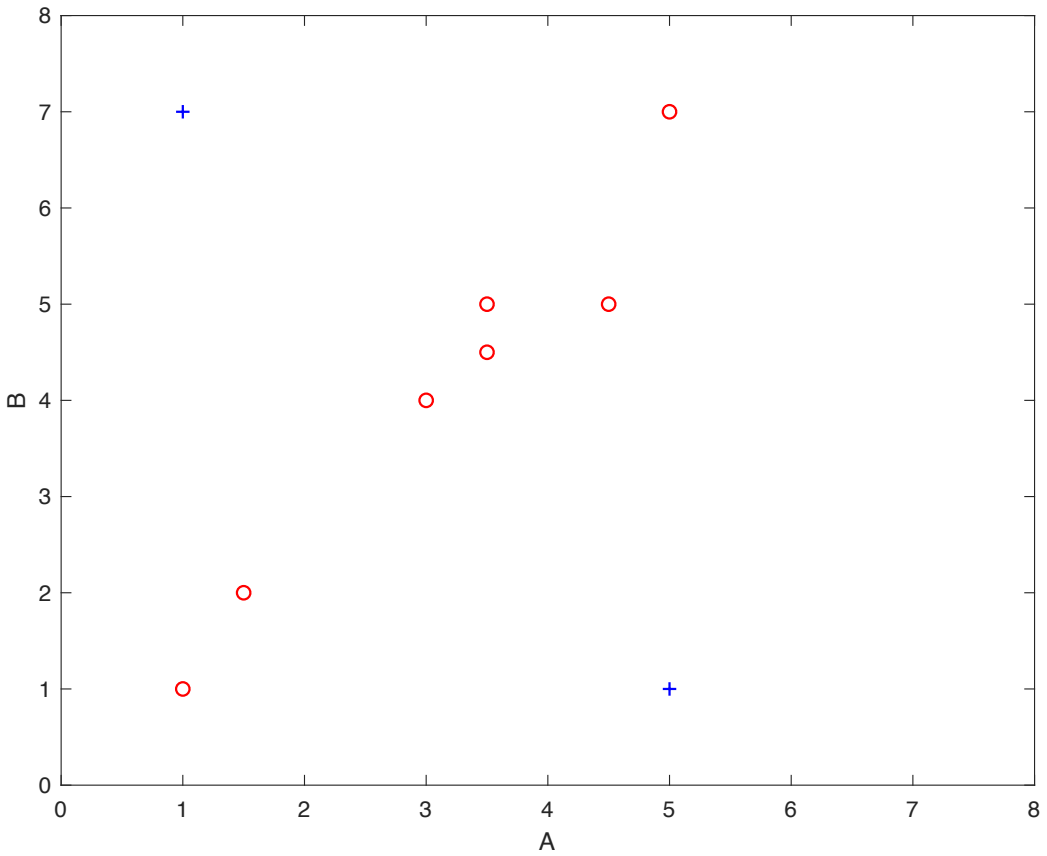
Outliers

- K Means:
  - Initialization
    - Forgy method: **randomly choose k observations** from the dataset and use these as the initial mean
      - Tends to spread the means
    - Random Partition: **randomly assign each observation to a cluster**; compute the means
      - Tends to concentrate the means close to the center of the dataset
- Distance
  - Euclidean distance
  - L1 (more robust to outliers)
  - …

K-means

/Users/Carvalho/Personal/Aulas/2020-2021/TCD/matlab/Mykmeans

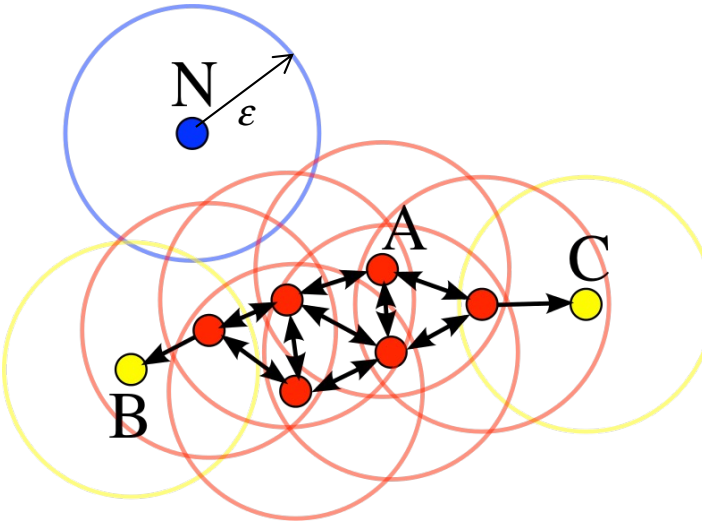
- Example:
  - **A** **B**
- p=[1, 1;...
  1.5, 2;...
  3, 4;...
  5, 7;...
  3.5, 5;...
  4.5, 5;...
  3.5 4.5]
- m1= [1,7]
- m2= [5,1]



Outliers

- DBSCAN: Density-based spatial clustering of applications with noise
  1. Find the points in the **ε (eps)** neighborhood of every point, and identify the **core points with more than minPts neighbors**.
  1. Find the connected components of core points on the neighbor graph, ignoring all non-core points.
  2. Assign each non-core point to a nearby cluster if the cluster is an ε (eps) neighbor, otherwise assign it to noise.

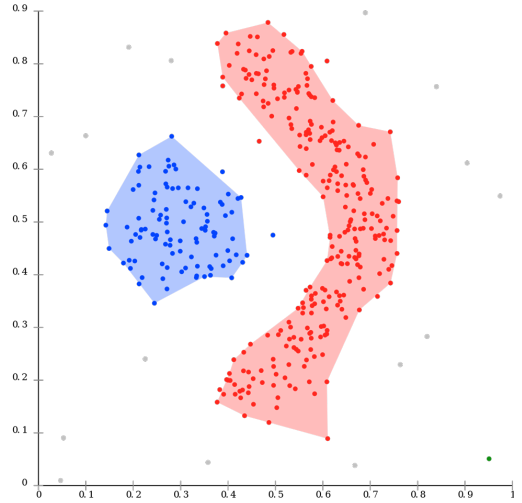
minPts = 4
Red – Core Points
Yellow – Not Core Points
Blue – Noisy point

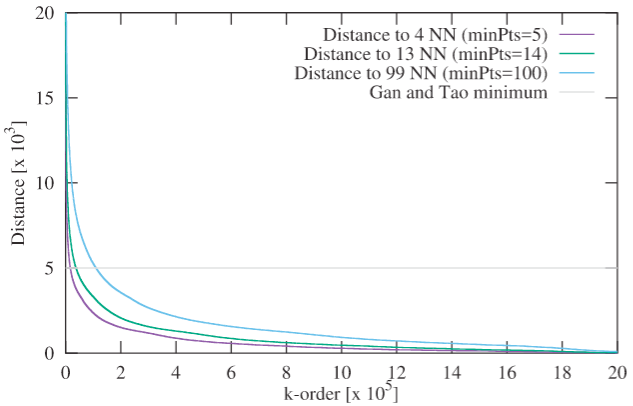


Outliers

https://en.wikipedia.org/wiki/DBSCAN#/media/File:DBSCAN-density-data.svg

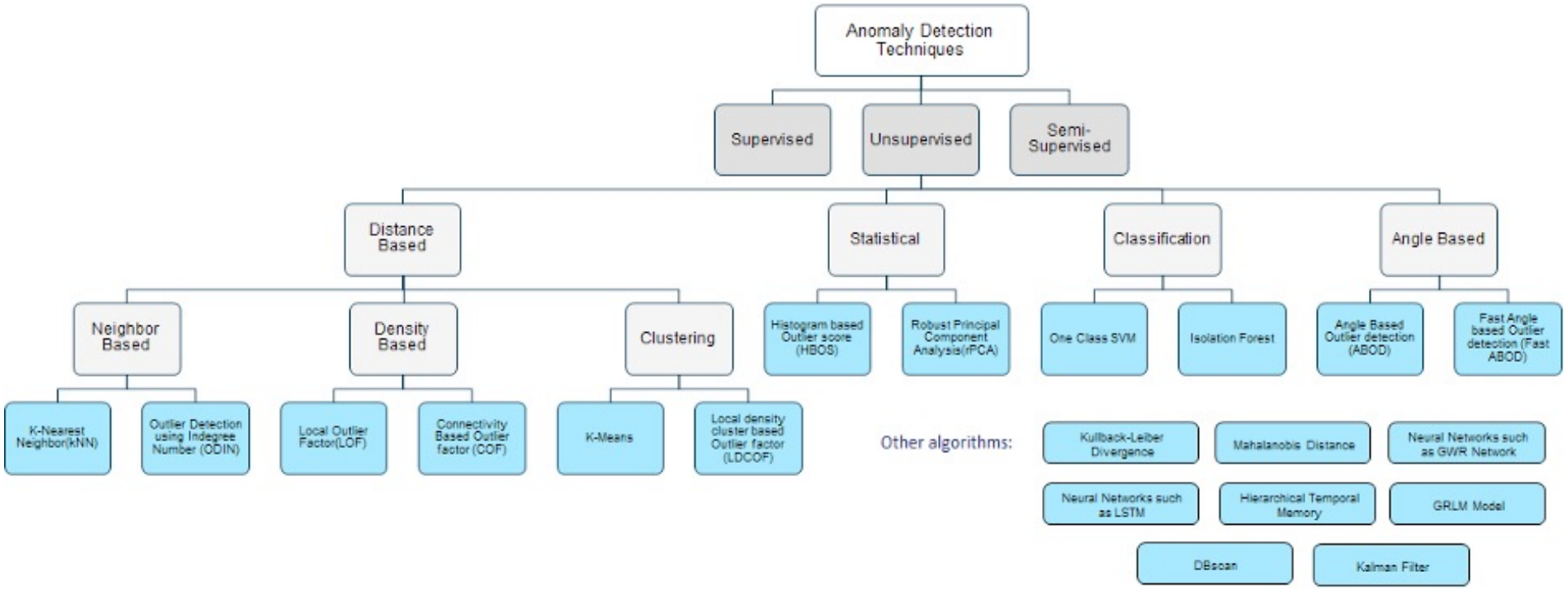
- DBSCAN: Density-based spatial clustering of applications with noise
- Advantages
  - Does **not require the specification of the number** of clusters
  - Can find **arbitrarily shaped** clusters. It can even find a cluster completely surrounded by (but not connected to) a different cluster.
  - Has a **notion of noise**, and is robust to outliers.
  - Requires **just two parameters** and is mostly insensitive to the ordering of the points in the database.
  - Parameters **minPts and ε can be set by a domain expert**, if the data is well understood.
- Parameters
  - *MinPts*: As a rule of thumb, a minimum *minPts* can be derived from the number of dimensions *D* in the data set, **as *minPts* ≥ *D* + 1 and *minPts* ≥ *3***
  - ε: k-distance graph, k smallest minPts-1 **nearest neighbor**. Good values of ε are where this plot shows an "elbow"



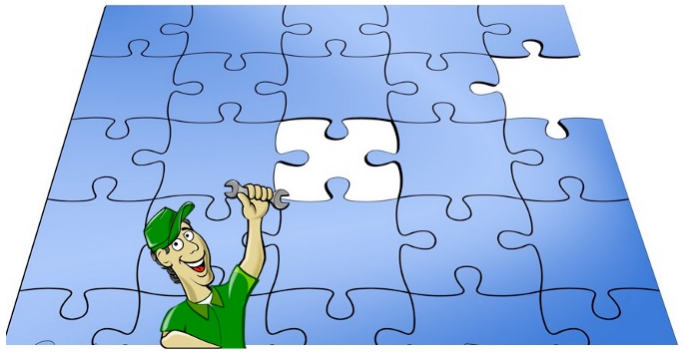


Outliers

- SoA



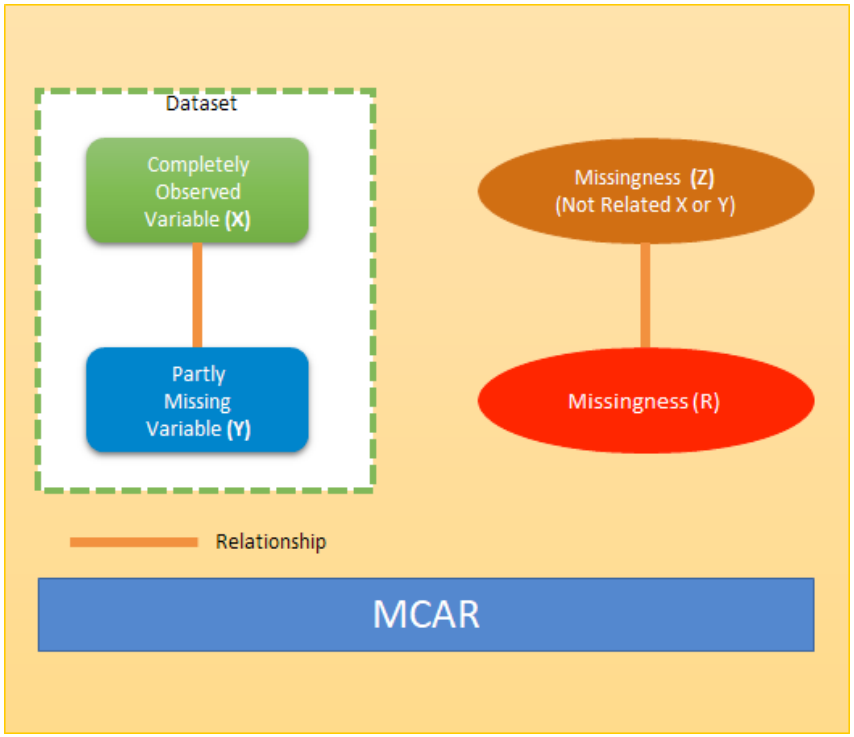
Missing Values



- Data is not always available
  - E.g. many values in clinical records are missing due to time constraints or even unavailable results
- Missing data may be due to
  - **Equipment malfunction** (e.g. sensor malfunction)
  - **Inconsistent** with other recorded data and thus deleted
  - **Data not entered** due to misunderstanding, unavailability,…
  - Certain data may not be **considered important** at the time of entry
  - ..
- Missing data needs to be infered!
  - Otherwise the available data for modeling might be insuficient

Missing Values

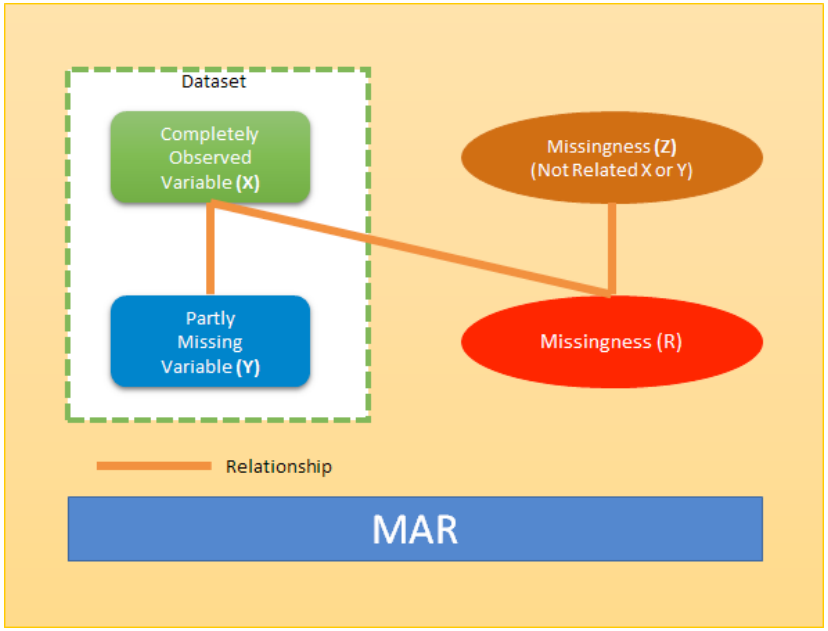
- Types of missing data
  - Missing completely at random (MCAR)
    - has nothing to do with the observation being studied
    - E.g.
      - weighing scale that ran out of batteries
      - a questionnaire might be lost in the post
      - a blood sample might be damaged in the lab
    - Statistically: MCAR analysis remains **unbiased**



https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

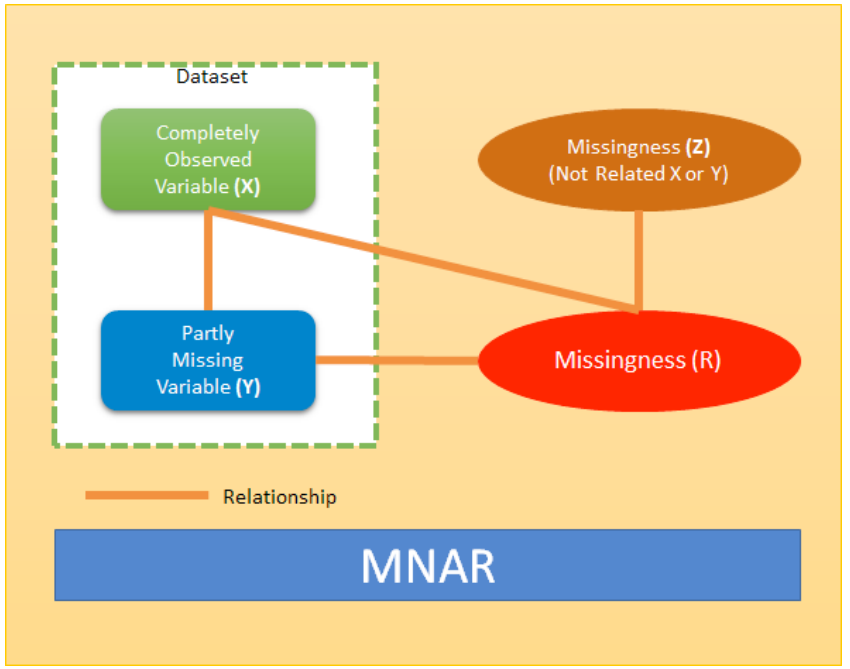
- Types of missing data
  - Missing at random (MAR)
    - missing data on a partly missing variable (Y) is related to some other completely observed variables(X) in the analysis model but not to the values of Y itself
    - E.g.
      - if a child does not attend an examination because the child is ill, this might be predictable from other data we have about the child's health, but it would not be related to what we would have examined had the child not been ill.



https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

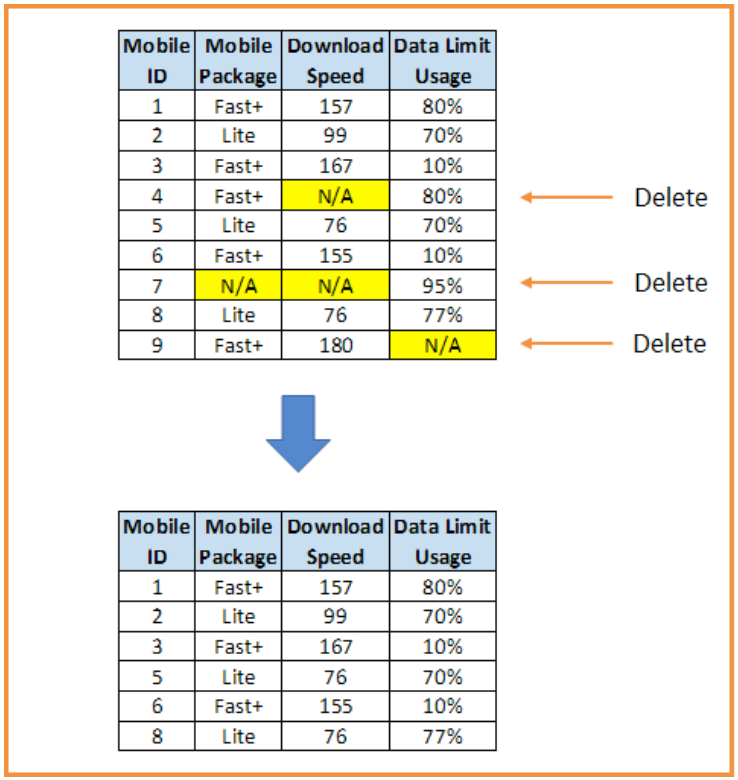
- Types of missing data
  - Missing not at random (MNAR)
    - Not MCAR or MAR
    - Might induce bias unless imputed using domain knowledge
    - E.g.
      - Persons with high BP might be more prone to miss appointment due to headache



https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

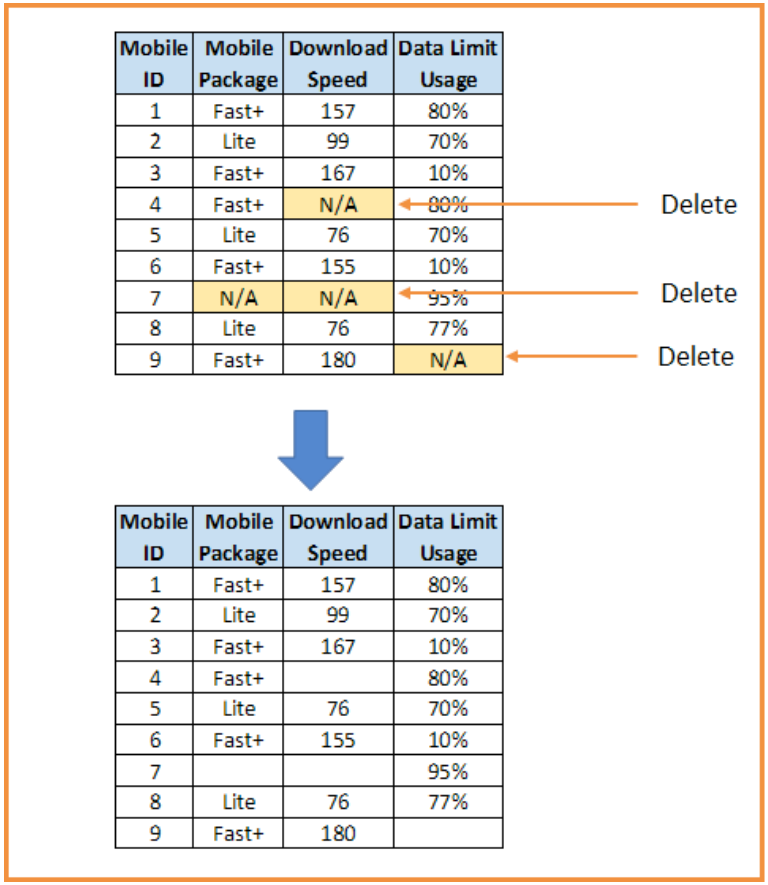
- How to handle
  - Discard Data
    - Complete-case Analysis
      - Omit cases with missing data
      - Use when:
        - » Large enough data set is available
        - » MCAR is satisfied; otherwise will induce bias



https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

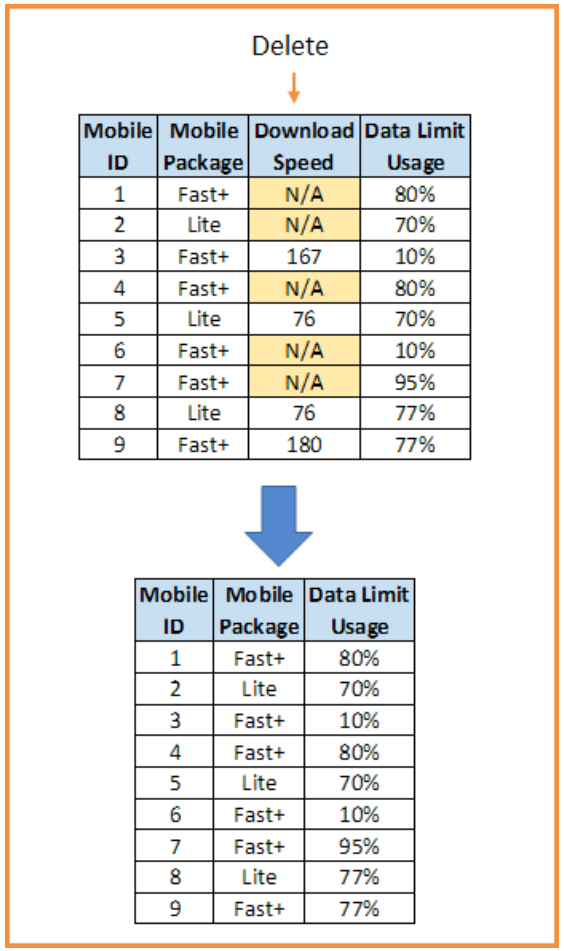
- How to handle
  - Discard Data
    - Pairwise deletion (available case analysis)
      - only the missing observations are ignored and analysis is done on variables present
      - Use when:
        - » MCAR or MAR is satisfied; otherwise will induce bias



https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

- How to handle
  - Discard Data
    - Dropping variables
      - Drop the entire variable
      - Use when:
        - » Too many values are missing for that variable
        - » Last resort-> check if model improves when deleting
      - Problems:
        - » Might loose a valuable feature



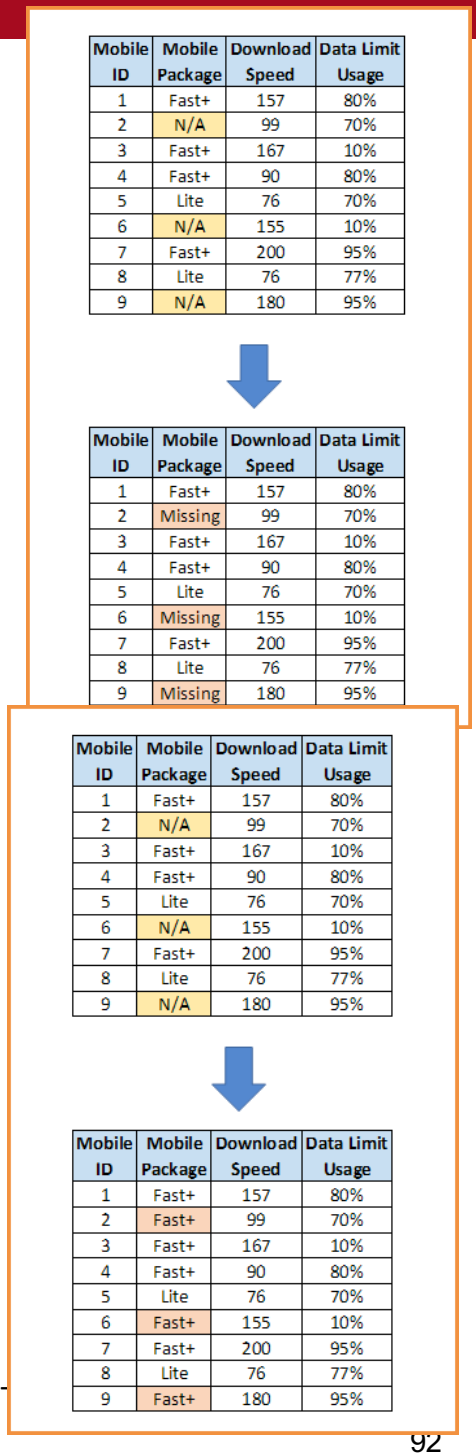
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	N/A	80%
2	Lite	N/A	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	N/A	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	77%

Mobile ID	Mobile Package	Data Limit Usage
1	Fast+	80%
2	Lite	70%
3	Fast+	10%
4	Fast+	80%
5	Lite	70%
6	Fast+	10%
7	Fast+	95%
8	Lite	77%
9	Fast+	77%

https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

- How to handle
  - Retain Data (ordinal and nominal variables)
    - **Common-point imputation**
      - For rating scales (ordinal variables) -> use the middle value
      - Similar to mean
    - **Frequent category imputation**
      - Impute with the most frequent category (equivalent to mean/median) for categorical variables
    - **Add a category** to capture missing
      - Most used approach for categorical nominal variables
    - **Random sampling imputation**
      - Take a random observation from the pool of available observations of the variable
      - Keeps the statistical parameters of the original variable
      - Assumes MCAR



Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	N/A	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	N/A	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	N/A	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Missing	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	Missing	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	Missing	180	95%

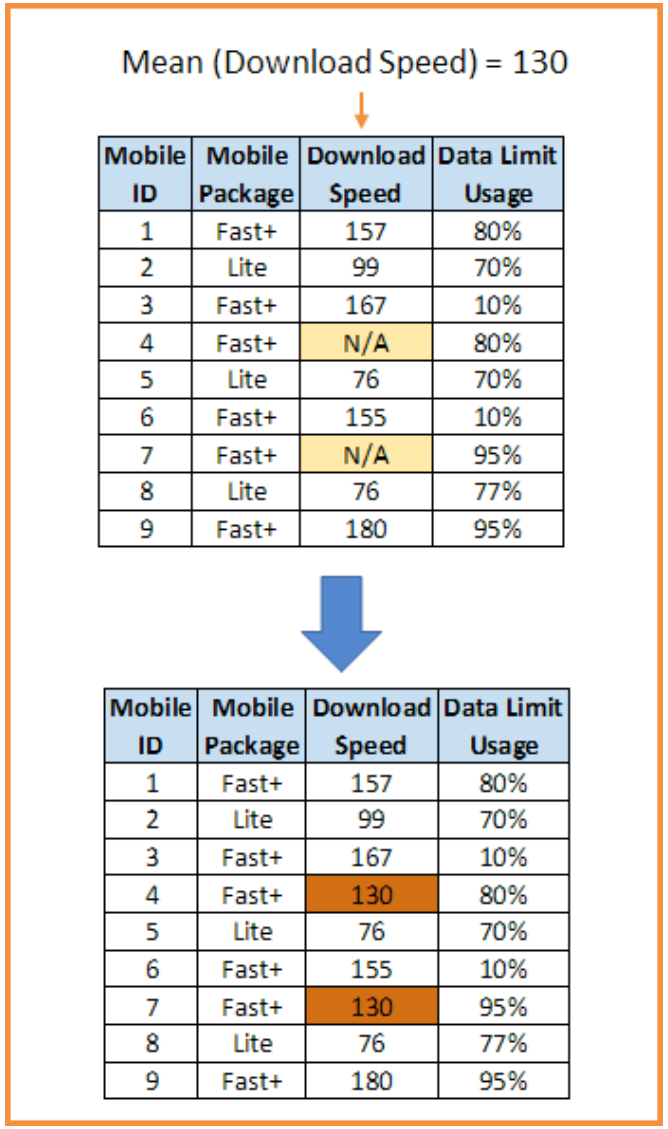
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	N/A	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	N/A	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	N/A	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Fast+	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	Fast+	180	95%

https://towardsdatascience.com/all-about-missing-

Missing Values

- How to handle
  - Retain Data
    - Mean, median, mode
      - replace missing data with statistical estimates of the missing values
    - Mean:
      - mean is a reasonable estimate for a randomly selected observation from a normal distribution
    - Median:
      - Use for skewed distribution (see 3rd statistical moment)
    - Mode
      - Substitute with the most frequent one (most likely)
    - Problems:
      - Distortion of original variance
      - Distortion of co-variance with remaining variables within the dataset



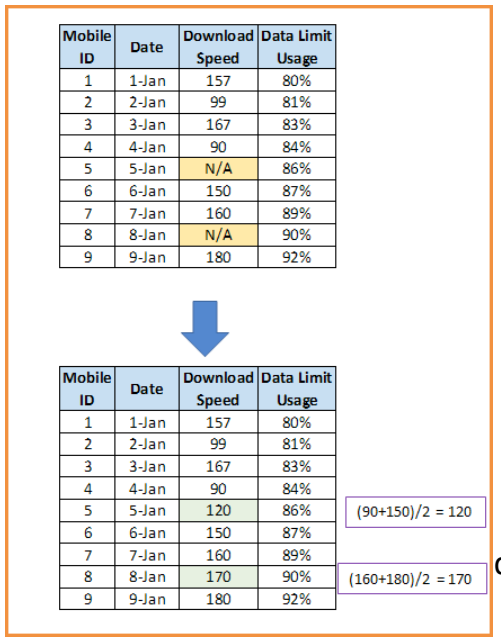
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

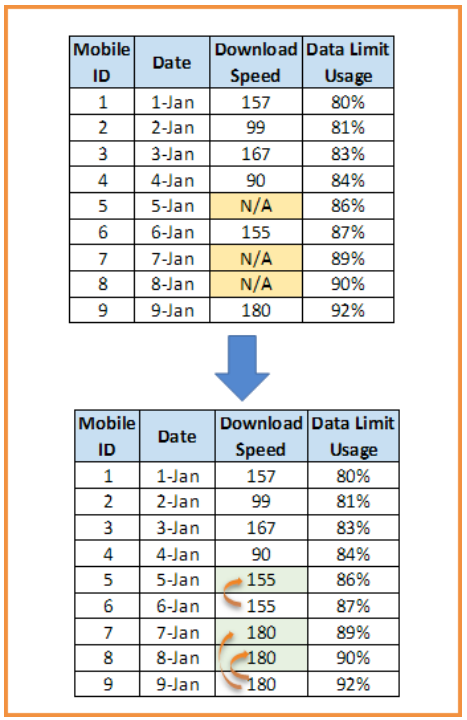
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	130	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	130	95%
8	Lite	76	77%
9	Fast+	180	95%

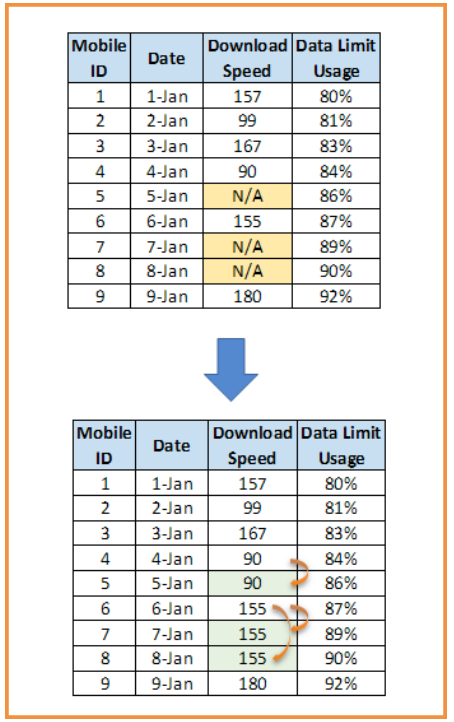
https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

- How to handle (interpolation)
  - Last observation carried forward
    - If data is time-series data, one of the most widely used imputation methods is the last observation carried forward
    - Nearest neighbor is a good interpolation method
  - Next observation carried backward
    - Similar to previous
  - Linear interpolation
    - Use a linear model



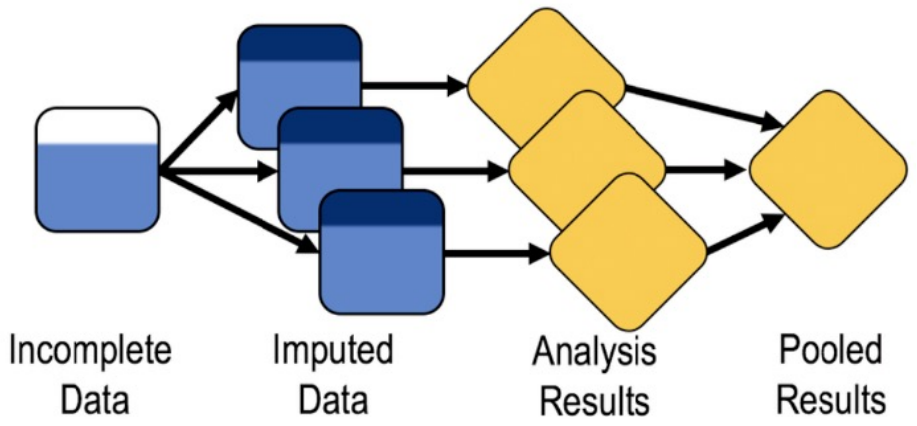




datascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values

- How to handle
  - Multiple imputation
    - Idea: use the distribution of the observed data to estimate a set of plausible values for the missing data
    - Multiple datasets are created and then analyzed individually but identically to obtain a set of parameter estimates. Estimates are combined to obtain a set of parameter estimates.

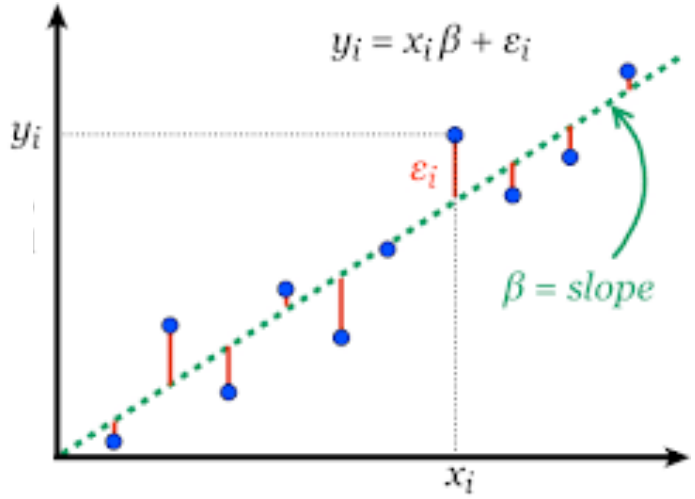


https://towardsdatascience.com/all-about-missing-data-handling-b94b8b5d2184

Missing Values – Predictive models

- Linear Regression
  - Let the known (complete) observations be $\left\{y_i, x_{i,1}, x_{i,2}, \ldots, x_{i,p}\right\}, i = 1, \ldots, n$
  - Model:

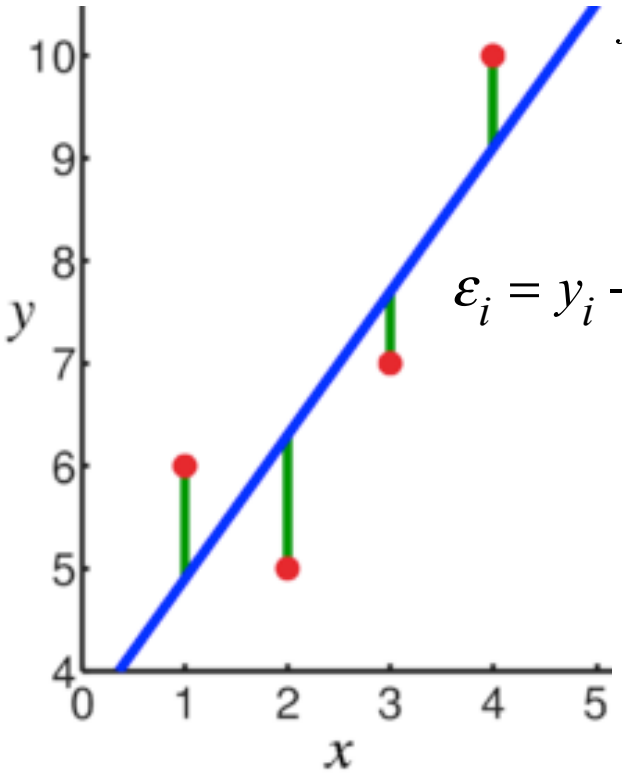
$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_p x_{i,p} + \varepsilon_i, i = 1, \ldots, n$$



Missing Values

- Linear Regression

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_p x_{i,p} + \varepsilon_i, i = 1,\ldots,n$$



$$f(x_i) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_p x_{i,p}$$

$$\varepsilon_i = y_i - \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_p x_{i,p}$$

$$\underset{\beta}{\operatorname{argmin}} \left(\sum_{i=1}^{n} (\varepsilon_i)^2 \right)$$

Missing Values

- Linear Regression

$$X = \begin{bmatrix} 1 & x_{1,1} & \cdots & x_{1,p} \\ 1 & x_{2,1} & \cdots & x_{2,p} \\ & & \vdots & \\ 1 & x_{n,1} & \cdots & x_{n,p} \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ \\ y_n \end{bmatrix}$$

$$\underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{p} \left(y_i - \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_p x_{i,p} \right)^2 = \underset{\beta}{\operatorname{argmin}} \left\| Y - X\beta \right\|^2$$

$$\frac{\partial}{\partial \beta} \left\| Y - X\beta \right\|^2 = 0 \Leftrightarrow \beta = \underbrace{\left(X^T X \right)^{-1} X^T}_{X^+} Y$$

Pseudo-inverse

- Low condition number-> use SVD

$$X = U \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} V^T \qquad X^+ = V \begin{pmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T$$

Missing Values

- Linear Regression (maximum likelihood interp.)

- Maximum likelihood
  - Independence of observations

$$P\left(\left(y_1,\mathbf{x}_1\right),\left(y_2,\mathbf{x}_2\right),....,\left(y_n,\mathbf{x}_n\right)|\,\beta,\sigma\right)=\prod_{i=1}^{n}P\left(y_i,\mathbf{x}_i\,|\,\beta,\sigma\right)$$

  - Likelihood

$$L\left(\beta,\sigma\,|\left(y_1,\mathbf{x}_1\right),\left(y_2,\mathbf{x}_2\right),....,\left(y_n,\mathbf{x}_n\right)\right)=P\left(\left(y_1,\mathbf{x}_1\right),\left(y_2,\mathbf{x}_2\right),....,\left(y_n,\mathbf{x}_n\right)|\,\beta,\sigma\right)$$

  - Maximum likelihood

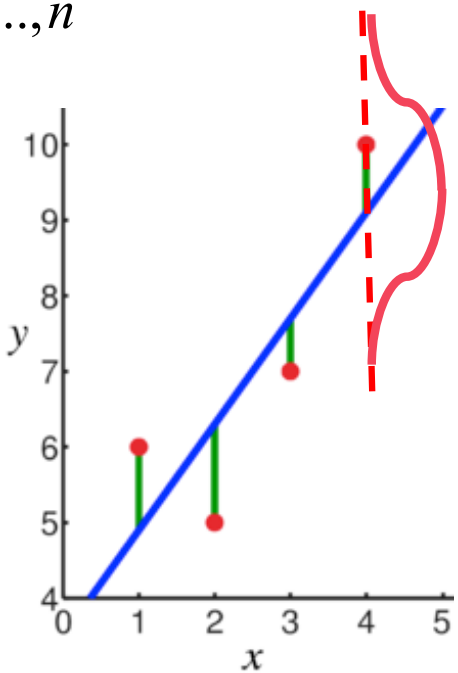
$$\underset{\beta}{\operatorname{argmax}}\,L\left(\beta,\sigma\,|\left(y_1,\mathbf{x}_1\right),\left(y_2,\mathbf{x}_2\right),....,\left(y_n,\mathbf{x}_n\right)\right)$$

Missing Values

- Linear Regression (maximum likelihood interp.)

$$\varepsilon_i = y_i - \beta_0 - \beta_1 x_{i,1} - \beta_2 x_{i,2} - \ldots - \beta_p x_{i,p}, i = 1,\ldots,n$$

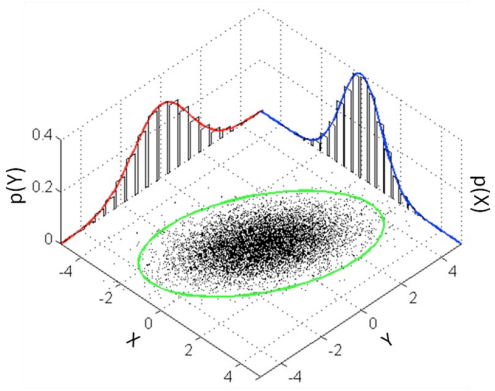
$$\varepsilon_i \sim \aleph\left(0,\sigma^2\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$$



$$P\left(y,x \mid \beta,\sigma\right) = \prod_{i=1}^{n} P\left(y_i,x_i \mid \beta,\sigma\right), i.i.d$$

$$P\left(y,x \mid \beta,\sigma\right) = P\left(\varepsilon\right)$$

Missing Values



- Linear Regression

$$P\left(y,x \mid \beta,\sigma\right) = P\left(\varepsilon\right)$$

$$P\left(y,x \mid \beta,\sigma\right) = \prod_{i=1}^{n} P\left(y_i,x_i \mid \beta,\sigma\right), i.i.d$$

$$\Omega = diag\left(\sigma_1,...,\sigma_n\right) \qquad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \\ \varepsilon_n \end{bmatrix}$$

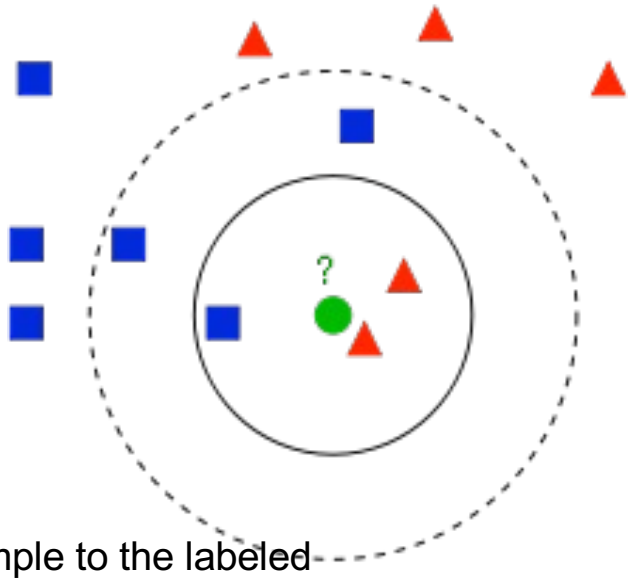
$$\underset{\beta}{\operatorname{argmax}}\, P\left(y,x \mid \beta,\sigma\right) = \underset{\beta}{\operatorname{argmax}} \frac{1}{\left(2\pi\right)^{\frac{n}{2}} \left|\Omega\right|^{\frac{1}{2}}} \exp\left(-\frac{\left(Y - X\beta\right)\Omega^{-1}\left(Y - X\beta\right)^T}{2} \right)$$

$$\underset{\beta}{\arg\min} -2\ln\left(P\left(y,x \mid \beta,\sigma\right)\right) = \underset{\beta}{\arg\min} \underbrace{n\ln\left(2\pi\right) + \ln\left|\Omega\right|}_{cte} + \left(Y - X\beta\right)\Omega^{-1}\left(Y - X\beta\right)^T$$

$$\underset{\beta}{\operatorname{argmin}} \left(Y - X\beta\right)\Omega^{-1}\left(Y - X\beta\right)^T \qquad \underset{\beta}{\operatorname{argmin}} \left(Y - X\beta\right)\left(Y - X\beta\right)^T, \Omega = \sigma I$$

Missing Values

- K-nearest neighbors
  - Can be used for classification (ordinal variables), or regression
  - Algorithm
    - Let *D( )* be a distance metric (typically Euclidean distance)
    - Let *k* be a predefined constant

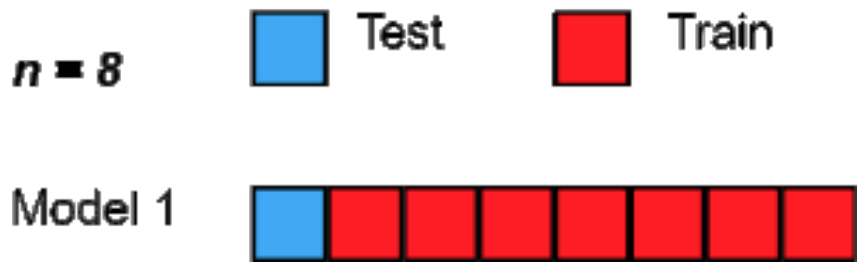


    - Get the k nearest neighbors of point x
    - **Classification:**
      - majority voting
      - Weighted voting (inverse of distance)
    - **Regression**
      - Weighted average (inverse of distance)

      1. Compute the Euclidean distance from the query example to the labeled examples.
      2. Order the labeled examples by increasing distance.
      3. Find a heuristically optimal number k of nearest neighbors, based on RMSE. This is done using **cross validation (see next)**.
      4. Calculate an inverse distance weighted average with the k-nearest multivariate neighbors.

Missing Values

- K = ?
- Cross-validation (more on this in the validation chapter)
  - Iterate for several K
  - Choose the one with the lowest RMSE



Missing values

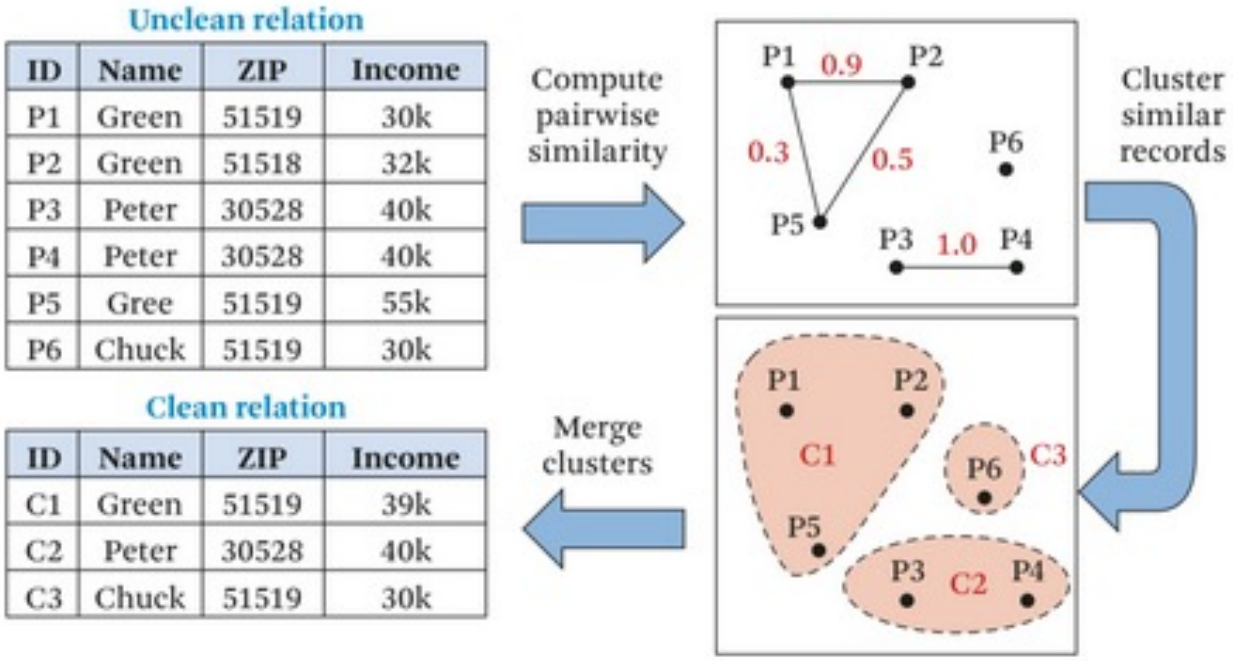
- If domain knowledge is available
  - Use known domain models
- If domain knowledge is not available
  - Use basic regression models
  - Use more advanced regression/Clustering models
    - Logistic
    - NN
    - …

Duplicates

- Duplicates
  - Detection: Identif. of multiple records that refer to the same entity
    - Have the same key-> easy
    - **Do not share the same key or contain errors-> difficult**
  - Source:
    - Lack of standard formats
    - Transcription errors
    - Incomplete information
  - Types of heterogeneity
    - ***Structural heterogeneity***: different organization of fields (e.g. DB1: address; DB2: Street, zip, City…)
    - ***Lexical heterogeneity***: identical structure of fields in DBs; data use different representations (e.g. DB1: R. Carlos Seixas, 141-2D; DB2: Rua C. Seixas, 141-2Dir)
  - Algorithms:
    - Fingerprinting of records

Duplicates

- Duplicates
  - Algorithms:
    - Pre-processing
    - Fingerprinting of records (similarity of fields, similarity of records)
    - Detection



Duplicates

- Pre-processing
  - Goal: Solve structural heterogeneity
  - Steps:
    - Parsing
      - Locate, Identify and isolate individual data elements
    - Data Transformation
      - Manipulation of each field to the same type representation (e.g. same units)
      - Renaming of fields
      - Range checking
    - Standartization
      - Information representation to a specific content format (e.g. Time hh:mm:ss; Date: dd:mm:yyyy)
    - Store pre-processed data in tables with comparable fields
    - Identify which fields should be compared to assess the fingerprint

Duplicates

- Field Matching Techniques
  - Character-based similarity
    - **Edit distance**: distance between two strings S_1 and S_2 is the minimum number of edit operations of single characters needed to transform S_1 into S_2
      - Type of operations:
        - » **Insert character**
        - » **Delete character**
        - » **Replace character**
      - Each operation might have different cost
      - Cost = 1 for all operations -> Levenshtein distance
        - » E.g. S1=Hello, S2=hello - > distance = 1 (1 replacement)
        - » E.g. S1=John R. Smith, S2=Jonathan Richard Smith -> Johnathan R.ichard Smith distance = 13 (1 delete+5 inserts+1delete+ 6inserts)
      - Works well for **typographical errors**
      - Does **not work well for trunctated or shortened strings**

Duplicates

- Field Matching Techniques
  - Character-based similarity
    - **Affine distance**: tries to overcome problem with **truncation and shortening**
      - Type of operations (2 additional operations):
        - » Insert character
        - » Delete character
        - » Replace character
        - » Open gap – start a new substring
        - » Extend gap – add a new char to the new substring
        - » Open gap has usually a cost much higher than extend gap
      - E.g. S1="Chris R Lang", S2="Christopher Richard Lang"
        - » Edit distance = 12 (6 inserts "topher" + 6 inserts "ichard")
        - » Affine distance using Open gap weight =1 and Extend Gap Weight = 0.1-> distance = 7 (1 Open gap + 5 extend for "topher" and 1 Open gap + 5 extend for "ichard")

Duplicates

- Field Matching Techniques
  - Character-based similarity
    - **Smith-Waterman distance**:
      - Similar to Affine Distance
      - **Mismatches at the beginning and the end** have smaller weights or are ignored
      - Allows for better local alignment (substring alignment)
      - E.g. S1="Prof. John R. Smith", S2="John R. Smith, Prof."
        - » Edit distance = 13 (6 deletes "Prof. " + 7 inserts ", Prof.")
        - » Affine distance using Open gap weight =1 and Extend Gap Weight = 0.1-> distance = 7.6 (6 deletes "Prof. " +1 open gap "," 6 extend gap" Prof.")
        - » Smith-Waterman: ignore prefix and suffix: distance = 0

Duplicates

- Field Matching Techniques
  - Character-based
    - Good for Typographical errors
    - **Bad** for capturing similarity of strings with **same tokens, but different orderings**
    - E.g.: “Paulo Carvalho” vs “Carvalho Paulo”
  - Token-based similarity
    - Token of S is a sequence of alphanumeric chars delimited by separator (space, punctuation)
    - Distance
      - Overlap: $$Overlap(S_1, S_2) = \frac{|tok(S_1) \cap tok(S_2)|}{\min(|tok(S_1)|, |tok(S_2)|)}$$
      - Jaccard $$Jaccard(S_1, S_2) = \frac{|tok(S_1) \cap tok(S_2)|}{|tok(S_1) \cup tok(S_2)|}$$
      - Dice: $$Dice(S_1, S_2) = \frac{2|tok(S_1) \cap tok(S_2)|}{|tok(S_1)| + |tok(S_2)|}$$

More sensitive to difference in token number

Duplicates

$$Overlap(S_1, S_2) = \frac{\left|tok(S_1) \cap tok(S_2)\right|}{\min(tok(S_1), tok(S_2))}$$

- Field Matching Techniques"
  - Token-based similarity
    - E.g.: S1="iphone 6s", S2="iphone 6s plus"
      - token(S1) = {"iphone", "6s"}
      - token(S2) = {"iphone", "6s", "plus"}
    - Distance
      - Overlap:

$$\left|tok(S_1) \cap tok(S_2)\right| = \underbrace{\left|\underbrace{\{"iphone","6s"\} \cap \{"iphone","6s","plus"\}}_{=\{"iphone","6s"\}}\right|}_{=2}$$

$$\min(\left|tok(S_1)\right|, \left|tok(S_2)\right|) = \min(2,3) = 2$$

$$Overlap(S_1, S_2) = \frac{2}{2} = 1$$

Duplicates

- Field Matching Techniques"

$$Jaccard(S_1, S_2) = \frac{|tok(S_1) \cap tok(S_2)|}{|tok(S_1) \cup tok(S_2)|}$$

  - Token-based similarity
    - E.g.: S1="iphone 6s", S2="iphone 6s plus"
      - token(S1) = {"iphone", "6s"}
      - token(S2) = {"iphone", "6s", "plus"}
    - Distance
      - Jaccard:

$$|tok(S_1) \cap tok(S_2)| = \underbrace{|\underbrace{\{"iphone","6s"\} \cap \{"iphone","6s","plus"\}}_{=\{"iphone","6s"\}}|}_{=2}$$

$$|tok(S_1) \cup tok(S_2)| = \underbrace{|\underbrace{\{"iphone","6s"\} \cup \{"iphone","6s","plus"\}}_{=\{"iphone","6s","plus"\}}|}_{=3}$$

$$Jaccard(S_1, S_2) = \frac{2}{3}$$

Duplicates

- Field Matching Techniques"
  - Token-based similarity
    - E.g.: S1="iphone 6s", S2="iphone 6s plus"
      - token(S1) = {"iphone", "6s"}
      - token(S2) = {"iphone", "6s", "plus"}
    - Distance
      - Dice:

$$Dice(S_1, S_2) = \frac{2\left|tok(S_1) \cap tok(S_2)\right|}{\left|tok(S_1)\right| + \left|tok(S_2)\right|}$$

$$\left|tok(S_1) \cap tok(S_2)\right| = \underbrace{\left|\underbrace{\{"iphone","6s"\} \cap \{"iphone","6s","plus"\}}_{=\{"iphone","6s"\}}\right|}_{=2}$$

$$\left|tok(S_1)\right| + \left|tok(S_2)\right| = \underbrace{\left|\{"iphone","6s"\}\right|}_{=2} + \underbrace{\left|\{"iphone","6s","plus"\}\right|}_{=3}$$

$$Dice(S_1, S_2) = \frac{4}{5}$$

Duplicates

- Field Matching Techniques”
  - Phonetic similarity: assesses phonetic similarity (e.g. Clair vs Clare)
  - Soundex: assign identical code digits t phonetically similar groups; mainly used to match surnames
    - 1: Keep the first letter of the surname and ignore all occurrences of W and H
    - 2. Assign following codes
      - B, F, P, V-> 1
      - C, G, J, K, Q, S, X, Z-> 2
      - D,T->3
      - L->4
      - M,N->5
      - R->6
    - 3. A,E,I,O,U and Y are nor coded and serve as separators
    - 4. Consolidate **sequences** of identical codes by keeping only the first occurrence
    - 5. Drop separators
    - 6. Keep the letter prefix and the three first codes, padding with zeros if there are fewer than 3 codes

Duplicates

- Soundex: E.g.: “Peter”
  - 1: Keep the first letter of the surname and ignore all occurrences of W and H
    - “Peter”->“P”
  - 2. Assign following codes
    - B, F, P, V-> 1
    - C, G, J, K, Q, S, X, Z->2
    - D,T->3
    - L->4
    - M,N->5
    - R->6
    - ->Pe3e6
  - 3. A,E,I,O,U and Y are nor coded and serve as separators
    - -> P0306
  - 4. Consolidate sequences of identical codes by keeping only the first occurrence
    - -> P0306
  - 5. Drop separators
    - -> P36
  - 6. Keep the letter prefix and the three first codes, padding with zeros if there are fewer than 3 codes
    - -> P360

Duplicates

- Field Matching Techniques”
  - Phonetic similarity: assesses phonetic similarity (e.g. Clair vs Clare)
  - Other algos:
    - **New York State Identification and Intelligence System**: replaces consonants with similar phonetic letters
    - …
- Detection?
  - Requires matching records composed by multiple fields
    - Unsupervised methods
      - Thresholding: Distance > Threshold
      - Domain specific rules: e.g. if the the first and the last name of a person are similar then records are duplicated
      - Clustering of similarities (e.g. Hierarchical clustering-> will be covered latter)
    - Supervised
      - Use a training dataset to develop a supervised classifier (any o the classifiers that will be covered latter)

Duplicates

- Detection?
  - Naïve Bayes
    - Use a training dataset to develop a classifier
    - Let A and B be the two tables we would like to match (could be the same table)
    - Let $\langle \alpha, \beta \rangle : (\alpha \in A, \beta \in B)$ be the two records we would like to assess if they match (M) or if the unmatch (U)
    - Let $\gamma = [\gamma_1, \ldots, \gamma_k]$ be the similarity of the k fields of records $\langle \alpha, \beta \rangle : (\alpha \in A, \beta \in B)$
    - Bayes classification

$$\langle \alpha, \beta \rangle = \begin{cases} M \Leftarrow P(M \mid \gamma) \geq P(U \mid \gamma) \\ U \Leftarrow otherwise \end{cases}$$

Likelihood ratio

$$\langle \alpha, \beta \rangle = \begin{cases} M \Leftarrow \dfrac{P(\gamma \mid M)}{P(\gamma \mid U)} \geq \dfrac{P(U)}{P(M)} \\ U \Leftarrow otherwise \end{cases}$$

Threshold

Duplicates

- Detection?
  - Naïve Bayes
    - Use a training dataset estimate

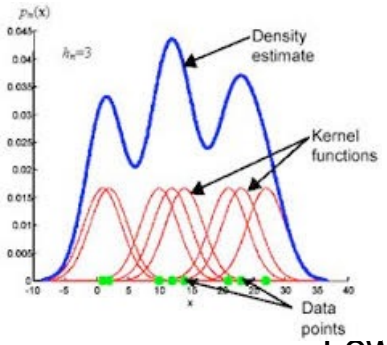
$$P(U)$$
$$P(M)$$
$$P(\gamma \mid U)$$
$$P(\gamma \mid M)$$

    - Naïve Bayes: assume independence in γ

$$P(\gamma \mid U) = \prod_{i=1}^{k} P(\gamma_i \mid U)$$

$$P(\gamma \mid M) = \prod_{i=1}^{k} P(\gamma_i \mid M)$$

    - Large number of known training data
      - » Assume Gaussinanity (use class mean and std of similarity)



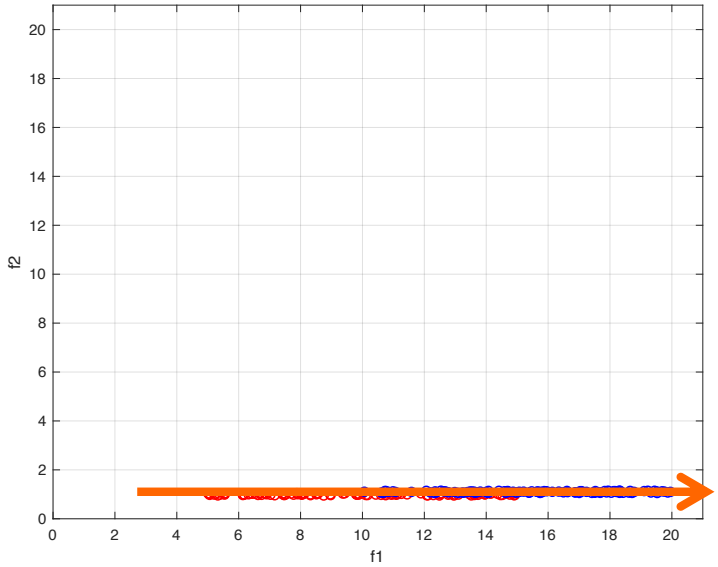
$$P(\gamma \mid U) = \prod_{i=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{Ui}} e^{-\frac{(\gamma_i - \mu_{Ui})^2}{2\sigma_{Ui}^2}}$$

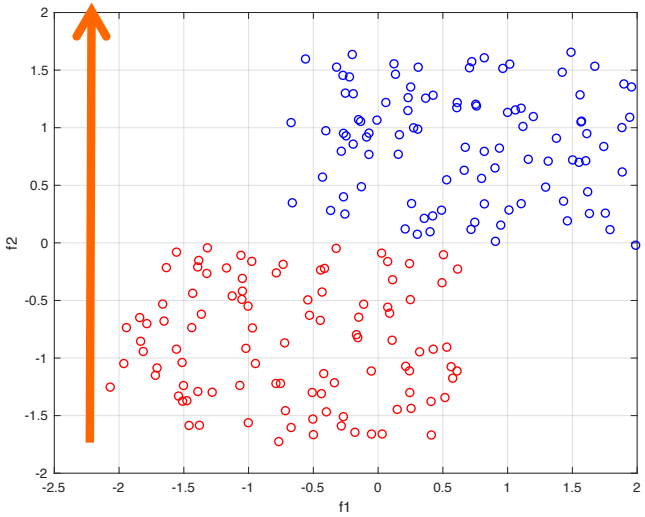
$$P(\gamma \mid M) = \prod_{i=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{Mi}} e^{-\frac{(\gamma_i - \mu_{Mi})^2}{2\sigma_{Mi}^2}}$$

- Low number of point Parzen Window using EM

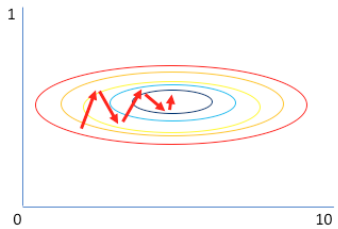
Normalization/standartization

- Problem: different scales

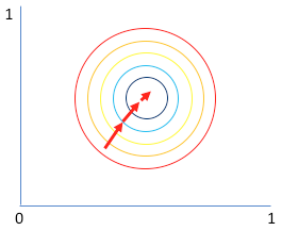




Why normalize?



Gradient of larger parameter dominates the update



Both parameters can be updated in equal proportions

Normalization/standartization

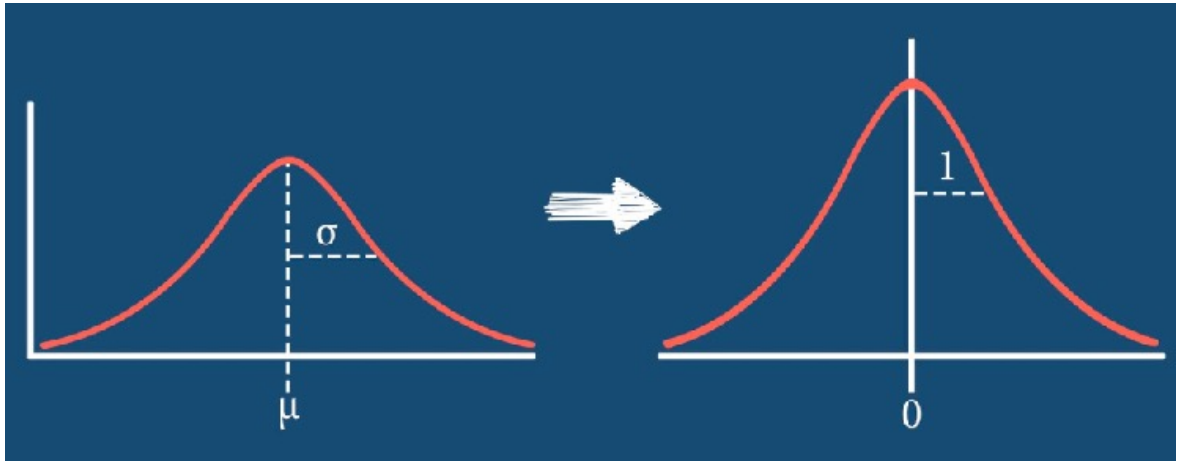
- Why, when?
  - **“Normalizing”** associated to rescaling by the minimum and range of the vector, to make all the elements lie between 0 and 1
    - Variables that are measured at different scales
    - Distribution is unknown
    - Distribution is non-Gaussian
    - Should be used when ML alg. does not make assumption about data distribution (e.g. ANN, K Nearest Neighbors, etc.)

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Normalization/standartization

- Why, when?
  - **“Standardizing”** associated to subtracting a measure of location and dividing by a measure of scale.
    - Variables that are measured at different scales
    - Distribution is Gaussian
    - Should be used when ML alg. assumes data is Gaussian distributes (e.g. linear regression, logistic regression, LDA, etc.)

$$Z = \frac{X - \mu}{\sigma}$$



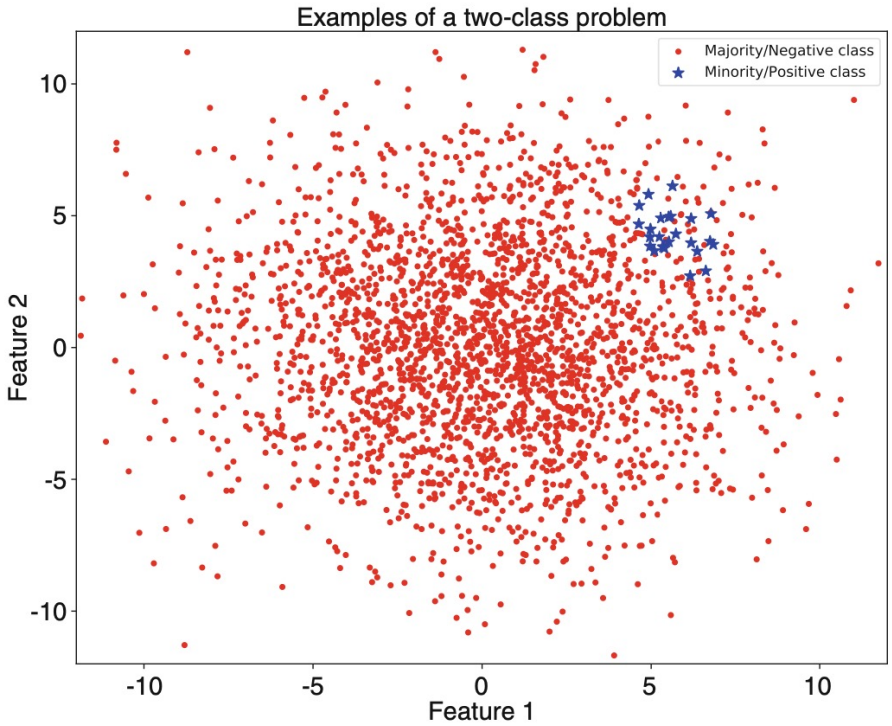
Unbalanced datasets

- Sometimes, classes have very unequal frequencies
  - Attrition prediction: 97% stay, 4% attrite
  - Medical diagnosis: 90% Healthy, 10% disease
  - eCommerce: 99% do not buy, 1% buy
  - Security: > 99.99% of Portuguese are not terrorists
- Similar situation with multiple classes

- Majority class classifier can be 99% correct, but useless
  - E.g.: eCommerce: 99% do not buy, 1% buy -> always output “does not buy”

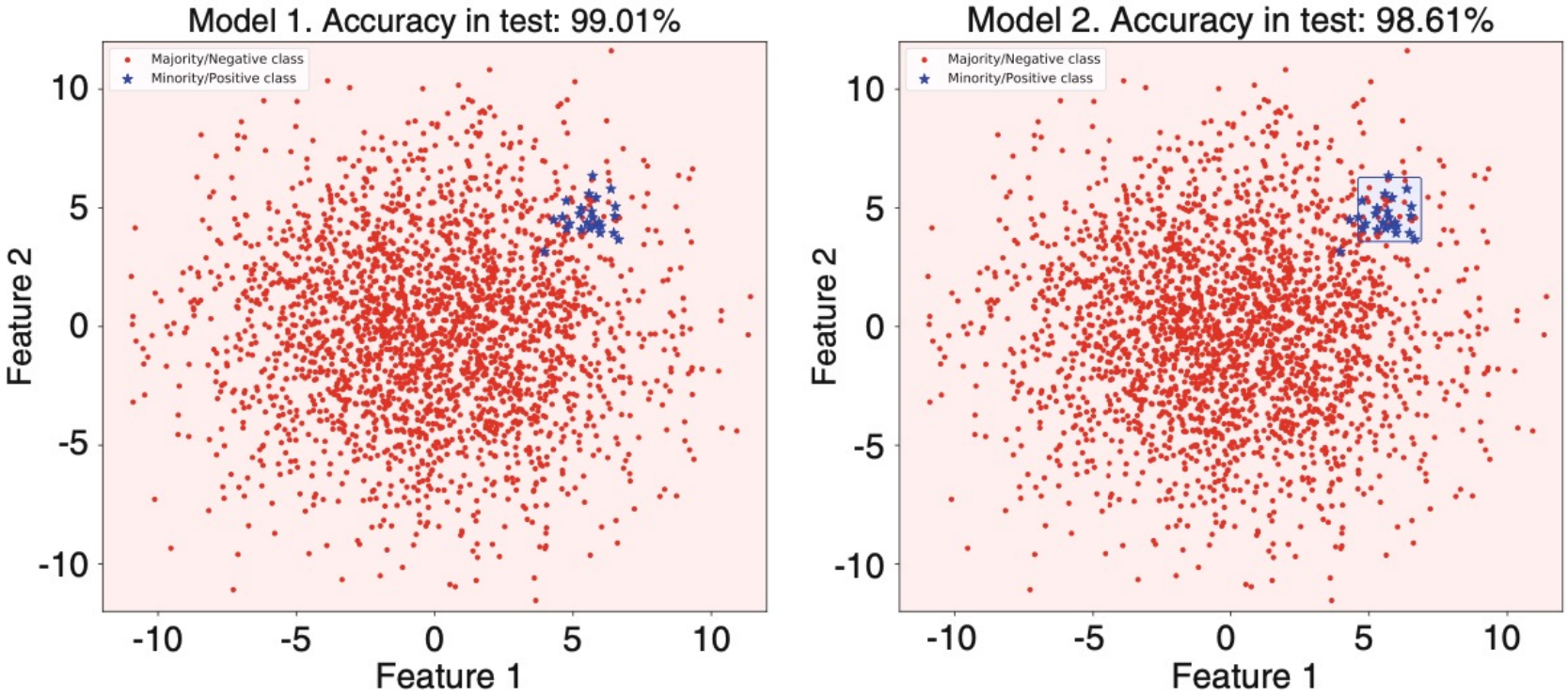
Unbalanced data sets

- Data imbalance:
  - Any data set with unequal class distribution
- Sometimes it is intrinsic to the problem:
  - Fault detection (ex: Feedzai)
  - Medical diagnosis
  - …
- Assessment:
  - Imbalance Ratio:
    - IR = #Positive Class/#Negative Class



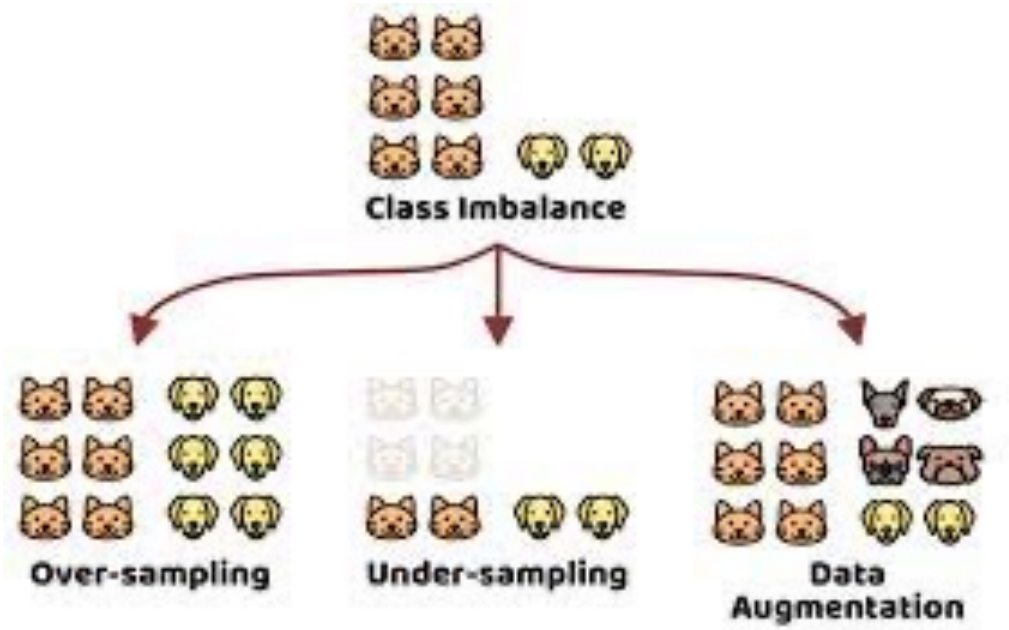
Unbalanced data sets

- Consequences of using usual performance metrics (accuracy)



Unbalanced data sets

- Imbalance Scenarios:
  - Need more informative performance measures
  - Need algorithms that are able to handle unbalanced data
  - Need data rebalance



Unbalanced data sets

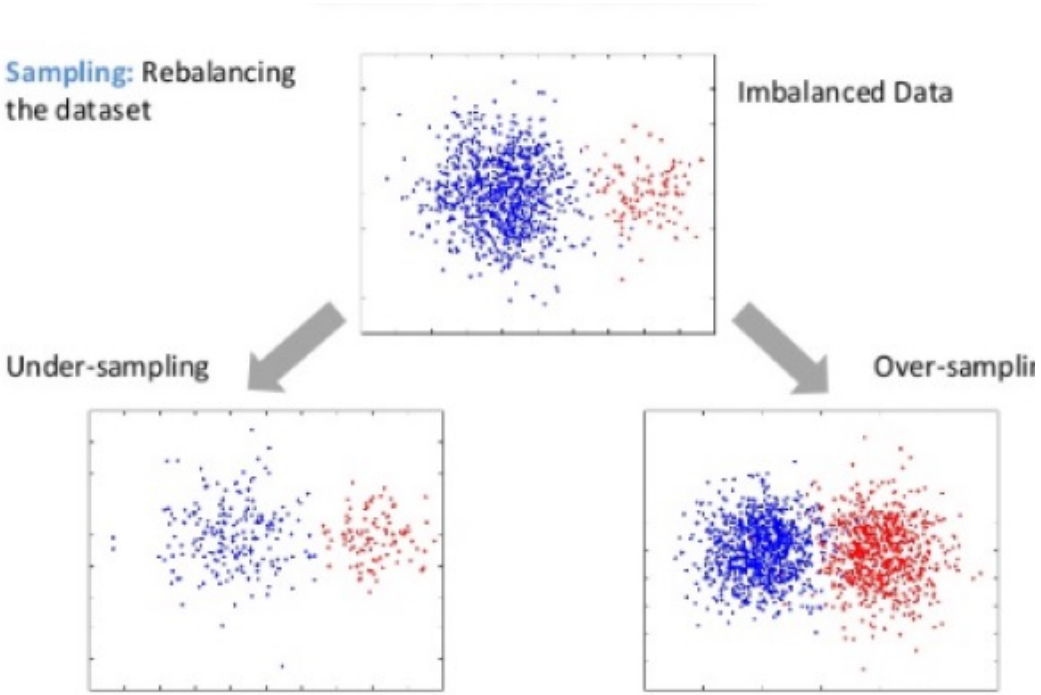
- Approaches:
  - Algorithmic level:adapt existing algorithms to bias towards minority class (master level)
  - Data level: rebalance class distribution
  - Cost-sensitive level: Algorithmic+Data level (master level)

$$O_i^*(x) = \sum_{j=1}^{M} O_i(x)\, C(i,j)$$

  - Ensemble-based: múltiple classifiers + combination and balance (master level)

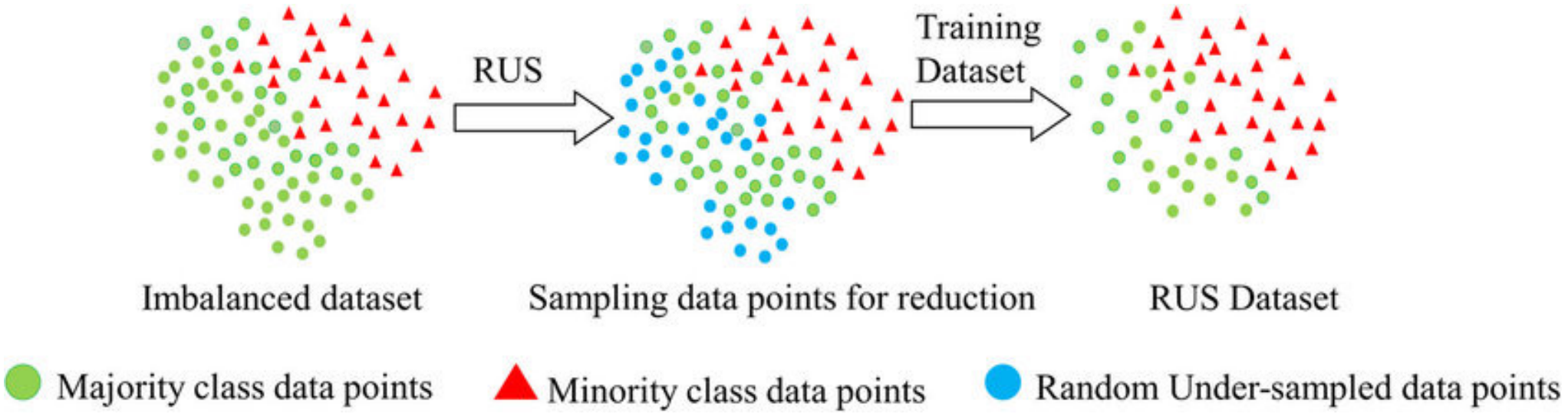
Unbalanced data sets

- Data level approaches:
  - Data sampling: sample to propduce a more balanced distribution
    - Undersampling: subset of original data set
      - Applicable when we have large data set
      - Can discard potential useful data
    - Oversampling: superset of dataset (duplication or generation)
      - Overfitting
    - Hybrid methods



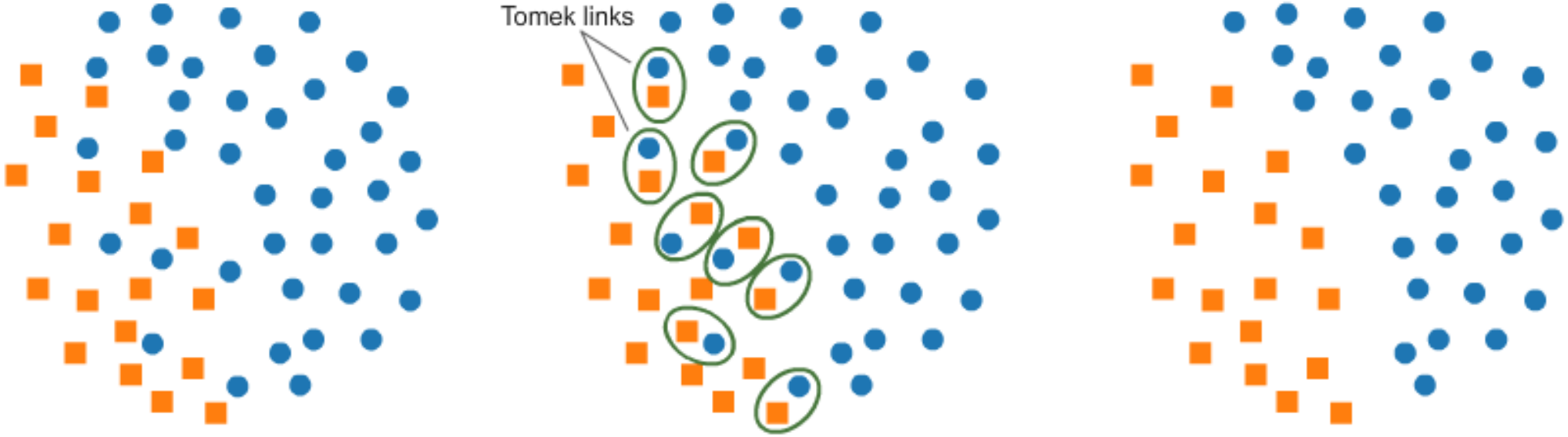
Unbalanced data sets

- Undersampling
  - RUS: Random undersampling
    - Non heuristic
    - Random elimination of majority class samples
    - Potential problem: Can discard potential useful data => use heuristics



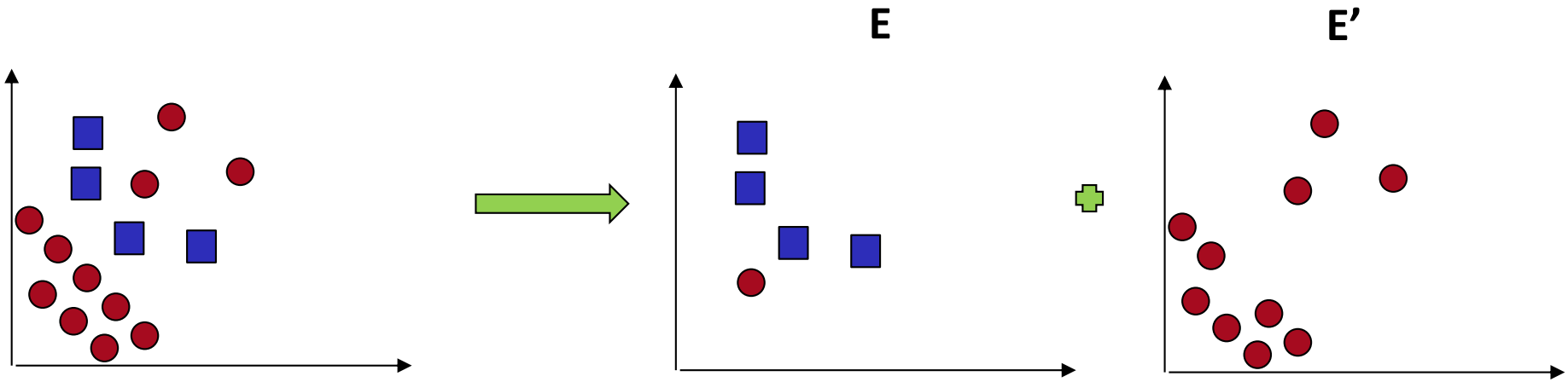
Unbalanced data sets

- Undersampling – heuristic-based methods
  - TL: Tomek Links
    - $E_i = (x_i, y_i)$ – x_i feature vector, y_i class label
    - Heuristic: Eliminate TLs of pairs (E_i, E_j); $i \neq j$; where distance is smaller than (E_i, E_i) or (E_j, E_j) (i.e. the two points, one of the majority and the other of the minority classes are nearest to each other)



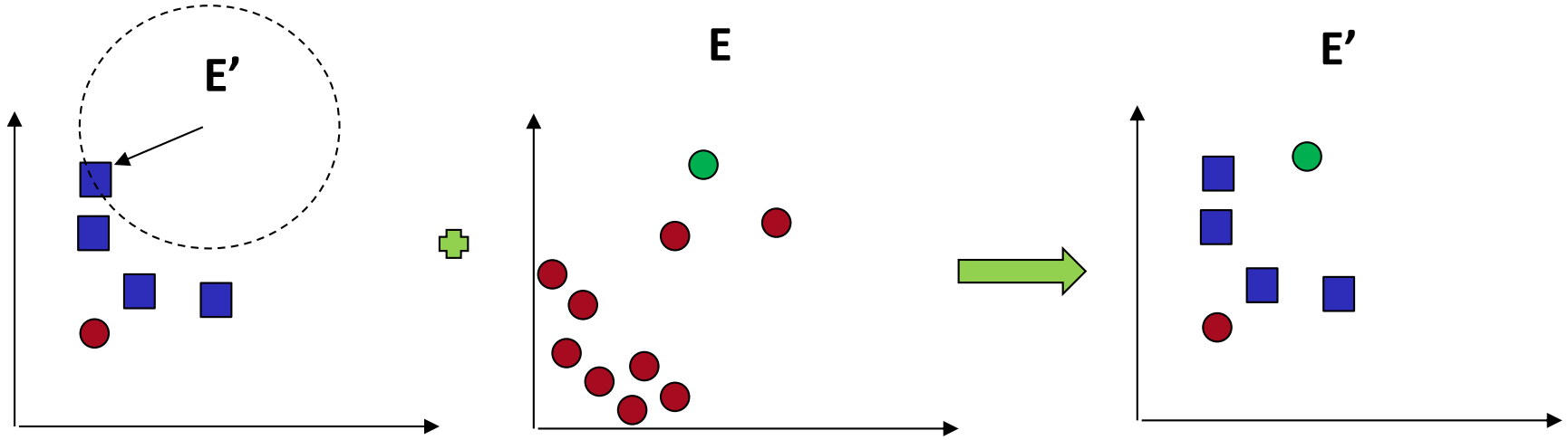
Unbalanced data sets

- Undersampling
  - US-CNN: Condensed Nearest Neighbor Rule
    - Heuristic: Eliminate the samples of the majority class that are far away from the decision border (less relevant)
    - 1st: one sample of majority class + all sampes of minority class



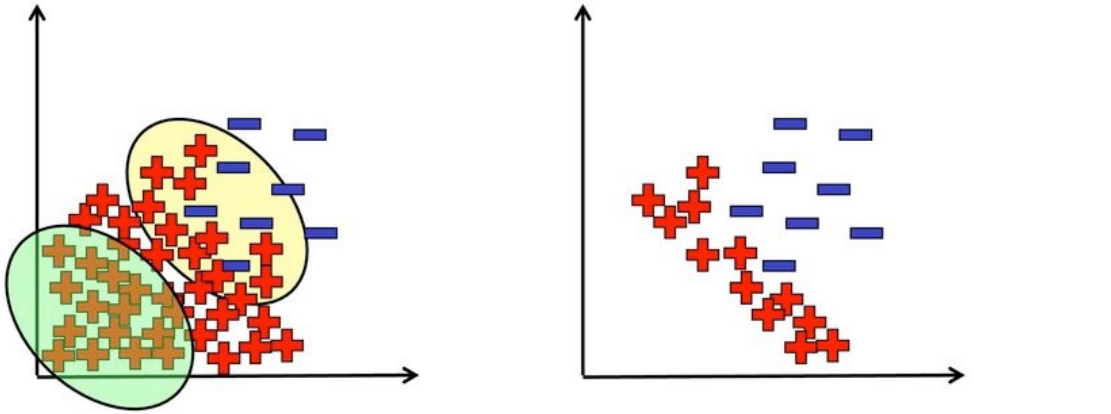
Unbalanced data sets

- Undersampling
  - US-CNN: Condensed Nearest Neighbor Rule
    - Heuristic: Eliminate the samples of the majority class that are far away from the decision border (less relevant)
    - 2nd: use 1-NN to classify each point in E'; add all that are wrongly classified



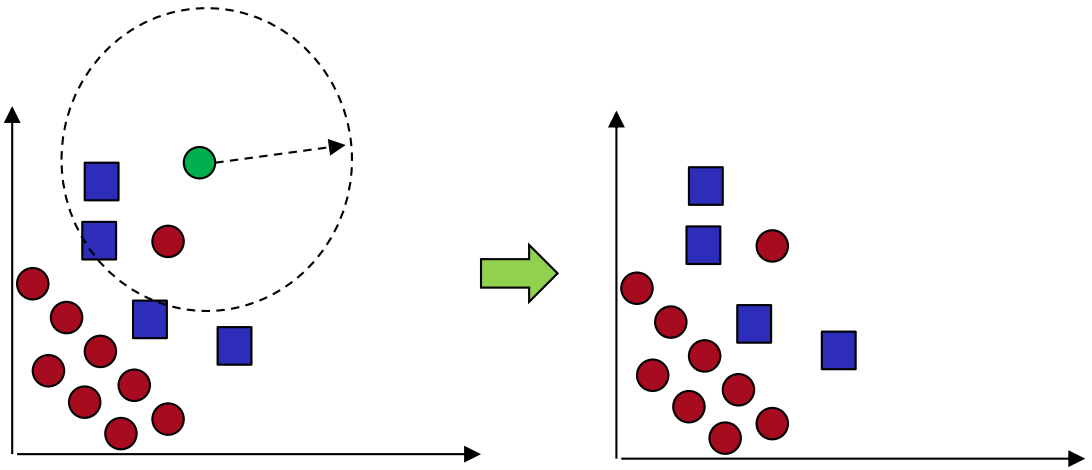
Unbalanced data sets

- Undersampling
  - OSS: One-sided Selection
    - Heuristic:
      - 1st use Tomek-Links to eliminate noisy borderline majority class data points
      - 2nd use Condensed Nearest Neighbor Rule that are far away from the decision border



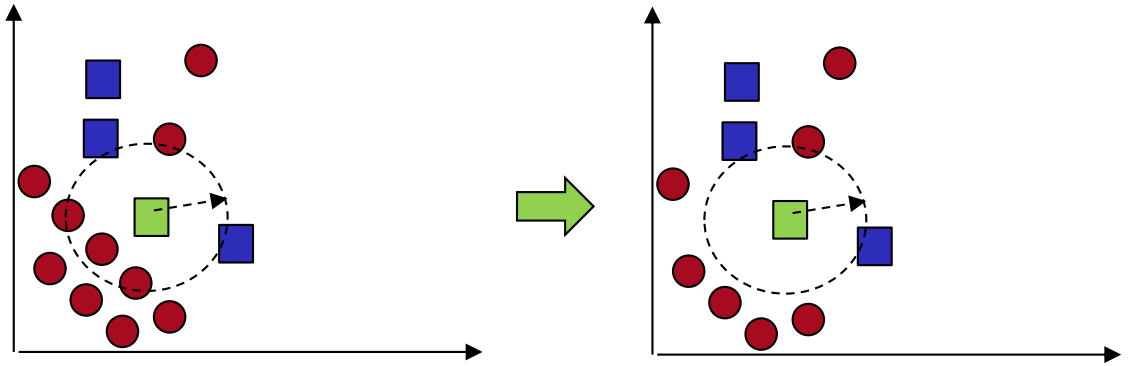
Unbalanced data sets

- Undersampling
  - NCL: Neighborhood Cleaning Rule
    - Heuristic:
      - Clean the data while reducing it:
      - Get the 3-NN of a data sample
        - » If data sample of majority class and 3-NN contradict classification, ELIMINATE DATA SAMPLE
        - » If data sample of minority class and 3-NN contradict classification, ELIMINATE DATA SAMPLES of 3-NN that belong to majority class



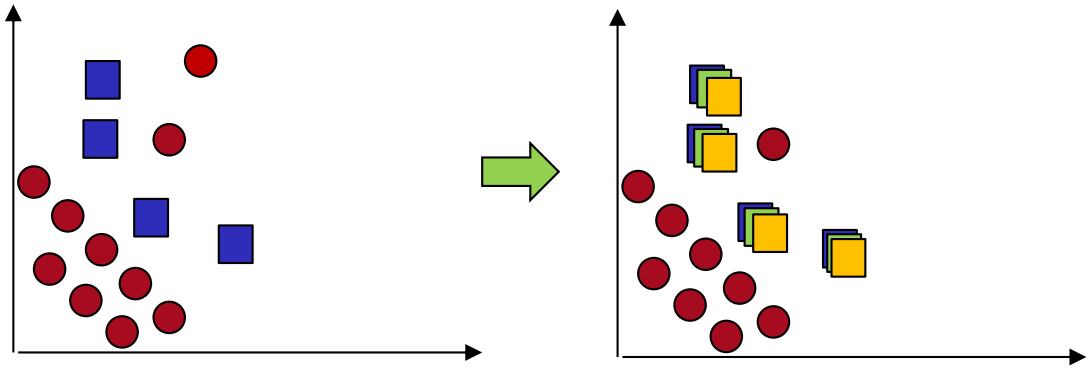
Unbalanced data sets

- Undersampling
  - NCL: Neighborhood Cleaning Rule
    - Heuristic:
      - Clean the data while reducing it:
      - Get the 3-NN of a data sample
        - » If data sample of majority class and 3-NN contradict classification, ELIMINATE DATA SAMPLE
        - » If data sample of minority class and 3-NN contradict classification, ELIMINATE DATA SAMPLES of 3-NN that belong to majority class



Unbalanced data sets

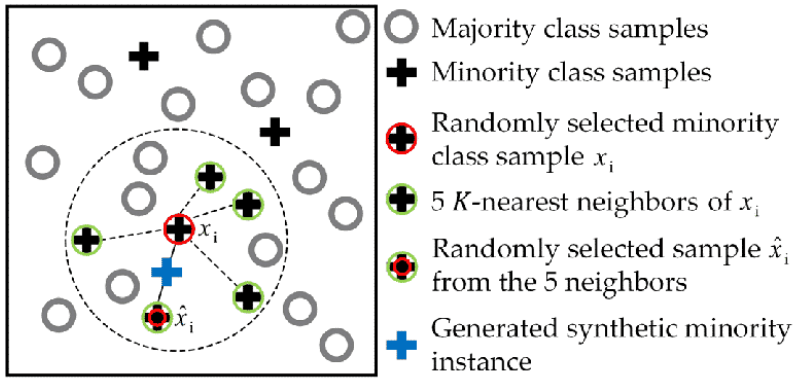
- What if there is a small number of data points? Oversampling
  - Naive Random Oversampling:
    - *Randomly duplicate (with replacement) minority data points*
    - *Disadvantage: possible overfitting of several models/regions*



Unbalanced data sets

- What if there is a small number of data points? Oversampling
  - SMOTE: Synthetic Minority Oversampling Technique
    - Idea: create synthetic data samples using interpolation of minority class data points that are near eachother
    - Select a minority class data point (x_i) at random
    - Select the k (usually k=5) NNs of x_i
    - Use randomised interpolation to generate synthetic data points
      - *Interpolation distance is selected at random*

- $x = x_i + rand\ (0-1)(\hat{x}_i - x_i)$
- $\hat{x}_i$ - randomly selected NN

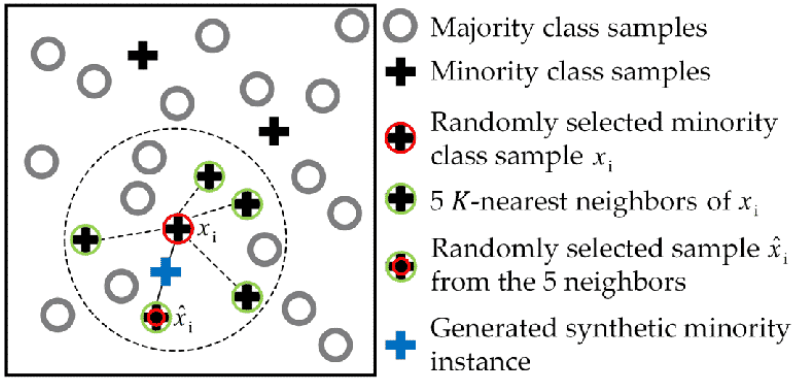


Unbalanced data sets

- Where to interpolate?

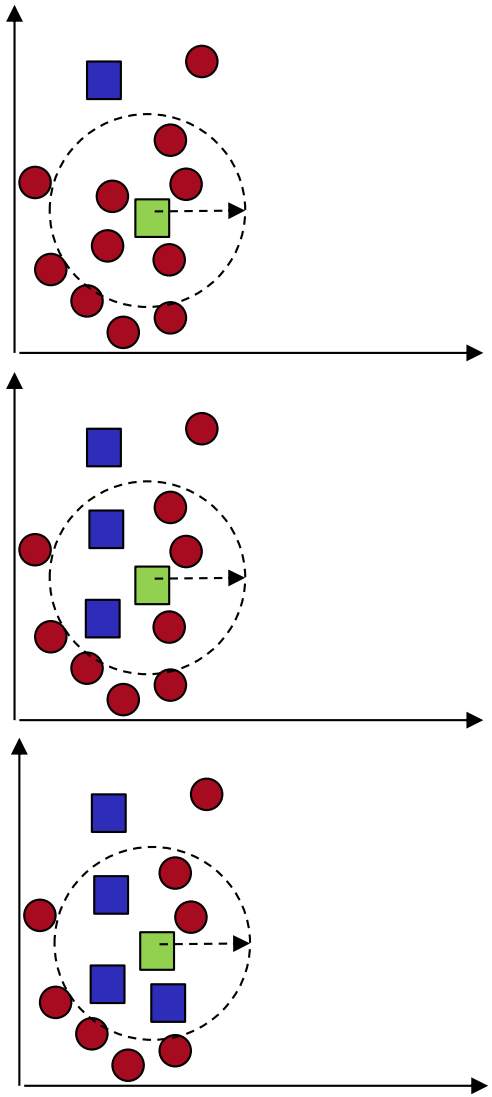
- N – number of data points to generate
- T – number of minority class points
- For each minority point:
  - Step 1: Select compute its KNN
  - Step 2: Generate randomly N/T synthetic data points

- $x = x_i + rand\ (0-1)(\hat{x}_i - x_i)$



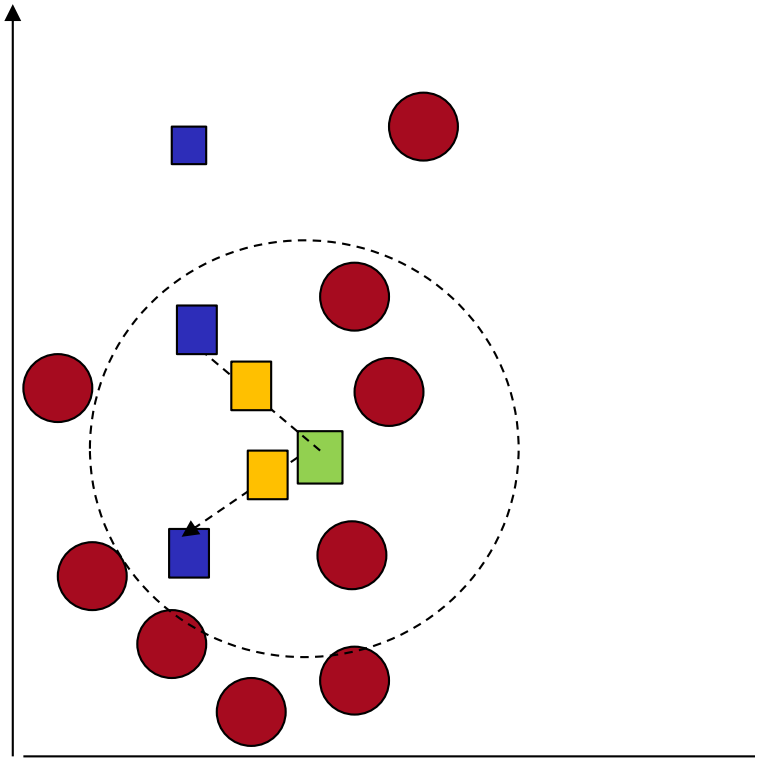
Unbalanced data sets

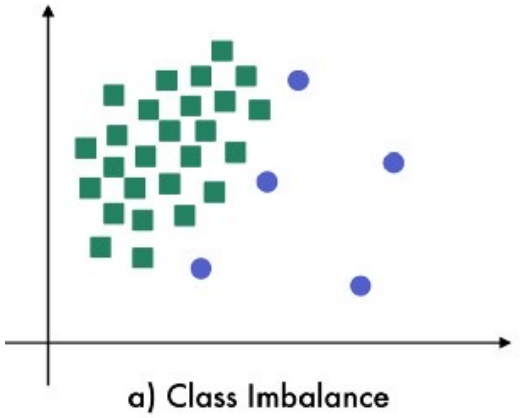
- Borderline SMOTE
  - To learn adequately: algorithms should learn the limits of each class
  - First identify the borderline minority samples
  - Generate synthetic samples
- Algorithm:
  - For each minority class point pi:
    - Calculate the KNN
      - If the all KNNs are majority class=> pi is considered noisy
      - If the number of majority points in KNNs is greater than minority points in KNN => Danger of being missclassified
      - If the number of majority points in KNNs is lower than minority points in KNN => NO Danger of being missclassified



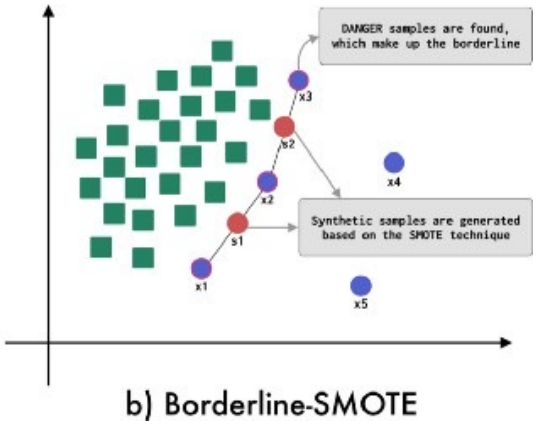
Unbalanced data sets

- Borderline SMOTE
  - To learn adequately: algorithms should learn the limits of each class
  - First identify the borderline minority samples
  - Generate synthetic samples
- Algorithm:
  - For each minority class point pi in DANGER:
    - Calculate the KNN of pi
      - Use SMOTE to generate synthetic data points





a) Class Imbalance



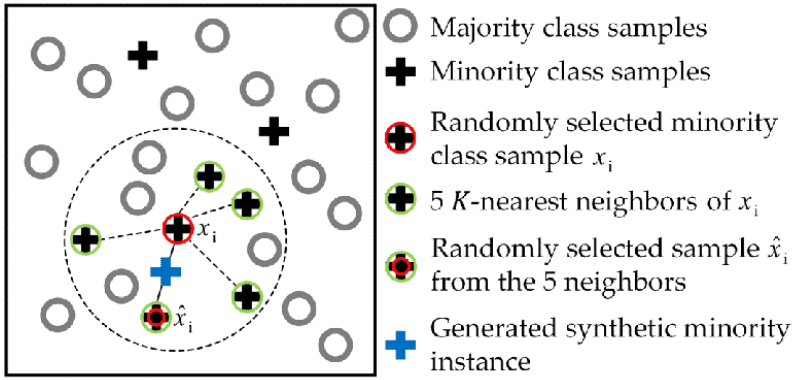
b) Borderline-SMOTE

Unbalanced data sets

- Where to interpolate?

- D – number of minority class points in DANGER
- For each minority point in DANGER:
  - Step 1: Select compute its KNN
  - s – number of minority class data points in KNN
  - Step 2: Generate randomly s synthetic data points

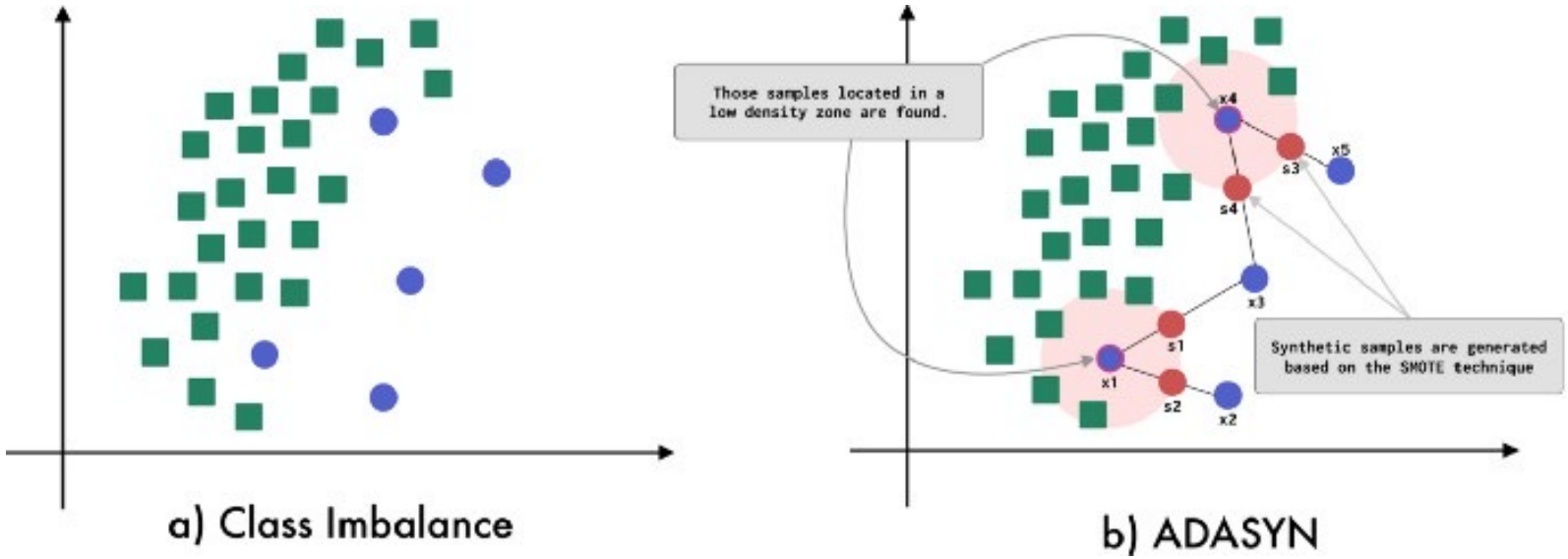
- $x = x_i + rand\ (0 - 1)(\hat{x}_i - x_i)$



Select the data

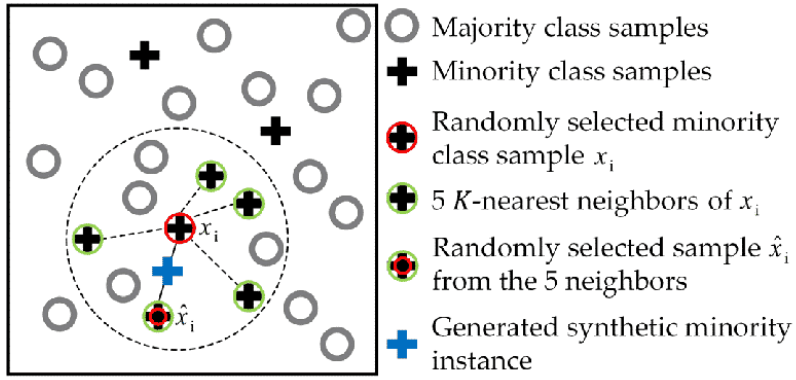
- Borderline SMOTE

- ADASYN Adaptive Synthetic Sampling Approach
  - there are variants where the density is taken into consideration to generate more or less data points for each pi



Unbalanced data sets

Where to interpolate?

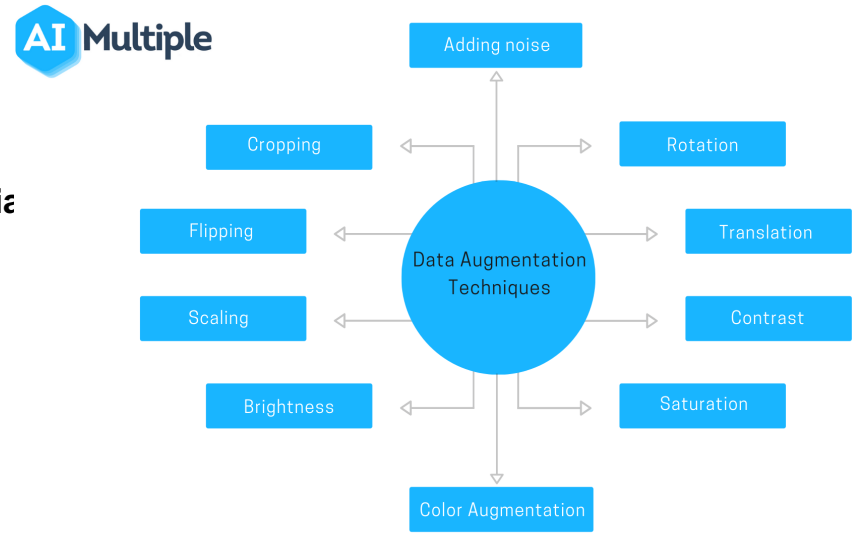


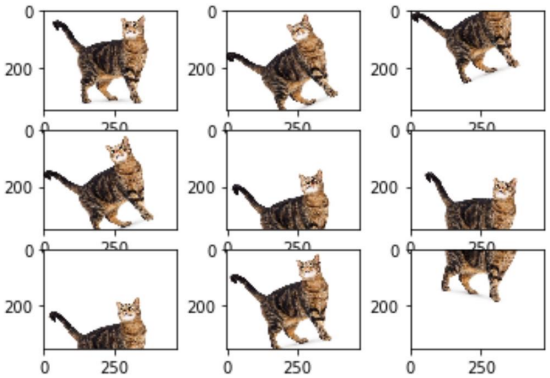
- N^-– number of minority class points
- N^+ – number of majority class points
- $\boldsymbol{G = (N^+ - N^-)\beta}, \beta \in]0,1]$, number of synthetic data points to be generated (NOTE that: if β=1, then $N^+ = N^-$)

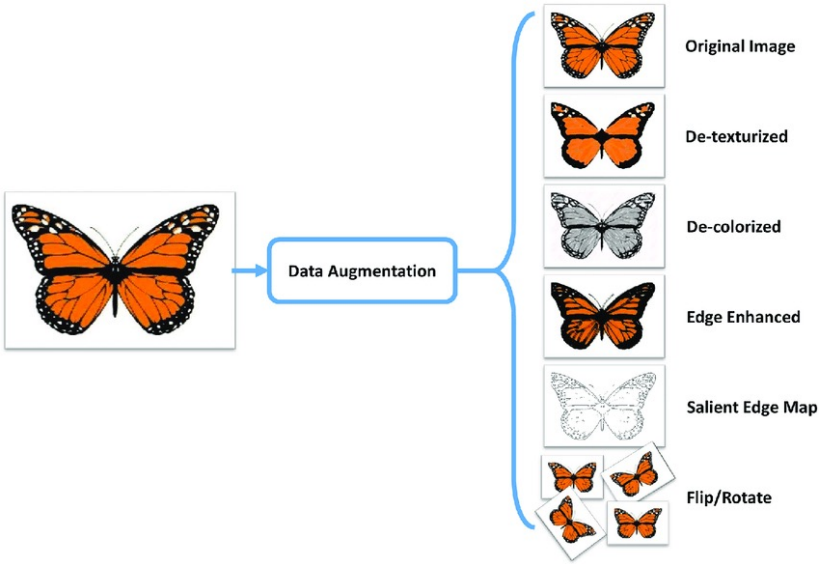
- For each minority point p_i: // compute the density
  - Compute its KNN
  - Let *ki* – number of minority points and let d_i – number of majority points in KNN
  - $r_i = d_i/k_i$
- $\widehat{r_i} = r_i / \sum_{j=1}^{N^-} r_i$ // compute the density
- For each minority point :
  - Generate randomly $s = G * \widehat{r_i}$ synthetic data points

Unbalanced data sets

- Data Augmentation
  - Transform the data
  - Generate the data (e.g. GANs **generative adversaria network**) -> MECD







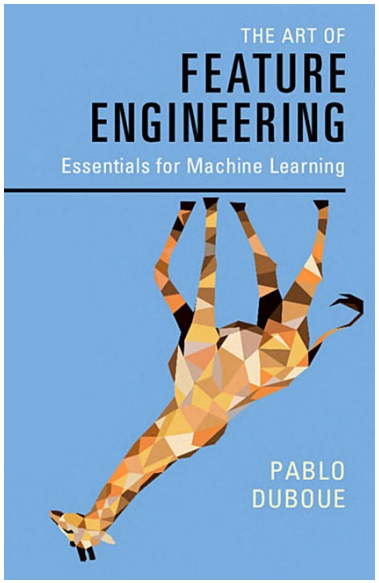
Engenharia de Características para Aprendizagem Computacional / Engenharia de Atributos

Chapter 5 – Data Preparation and Feature Engineering

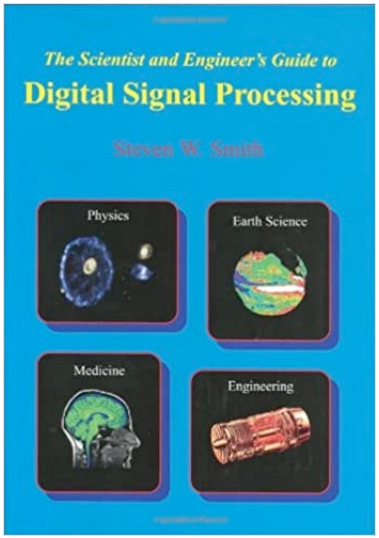
Paulo de Carvalho/Rui Paiva

Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia da Universidade de Coimbra
Edição 2020-2021

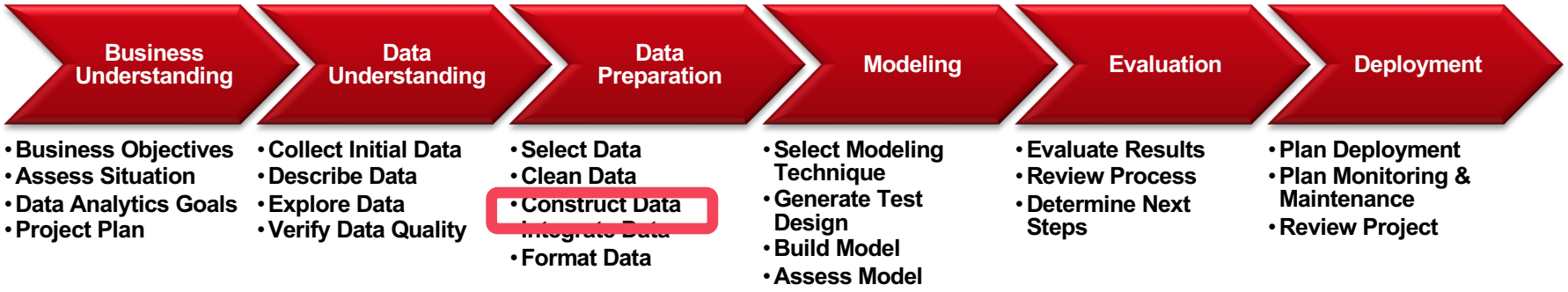
Bibliography



- Chapters:
  - 1: Types of features
  - 4: Dimensionality reduction, feature selection
  - 6-12, 14-20: Fourier transform and analysis



Course – Where are we?

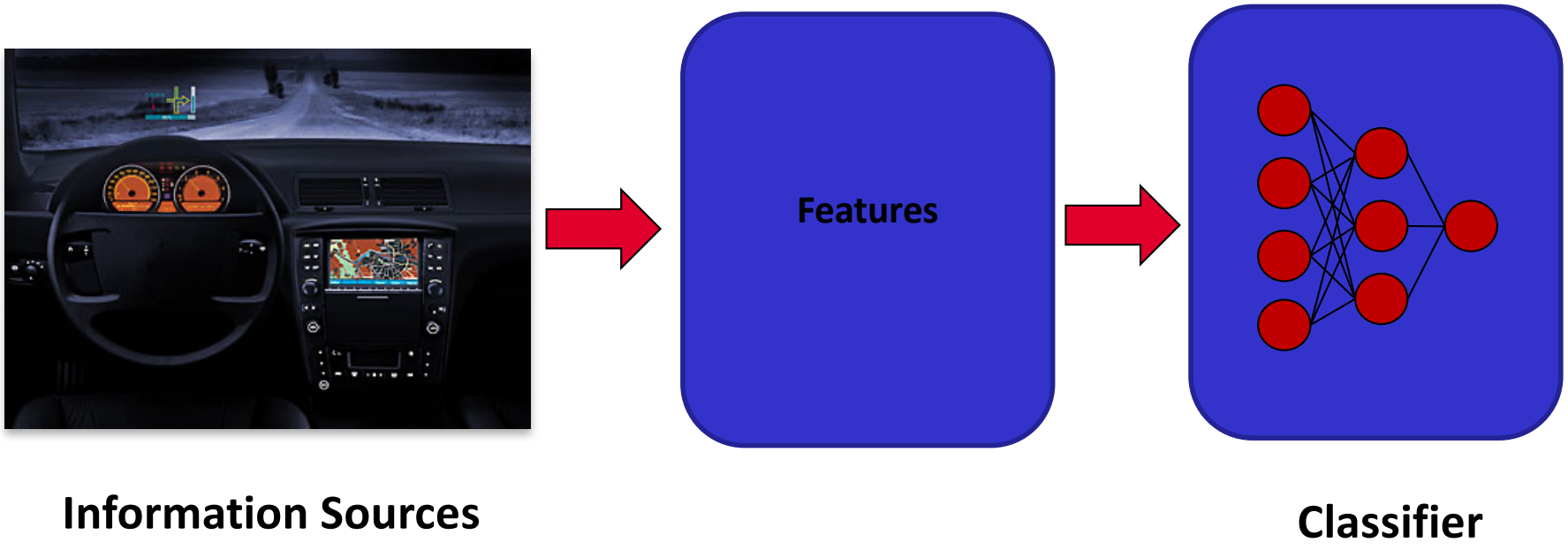


Outline

- Features
- Why feature Engineering?
- Approaches in Feature Engineering
- Feature Engineering: Dimensionality Reduction
- Feature Engineering: Feature Selection
- Feature Engineering: Frequency-based feature Engineering

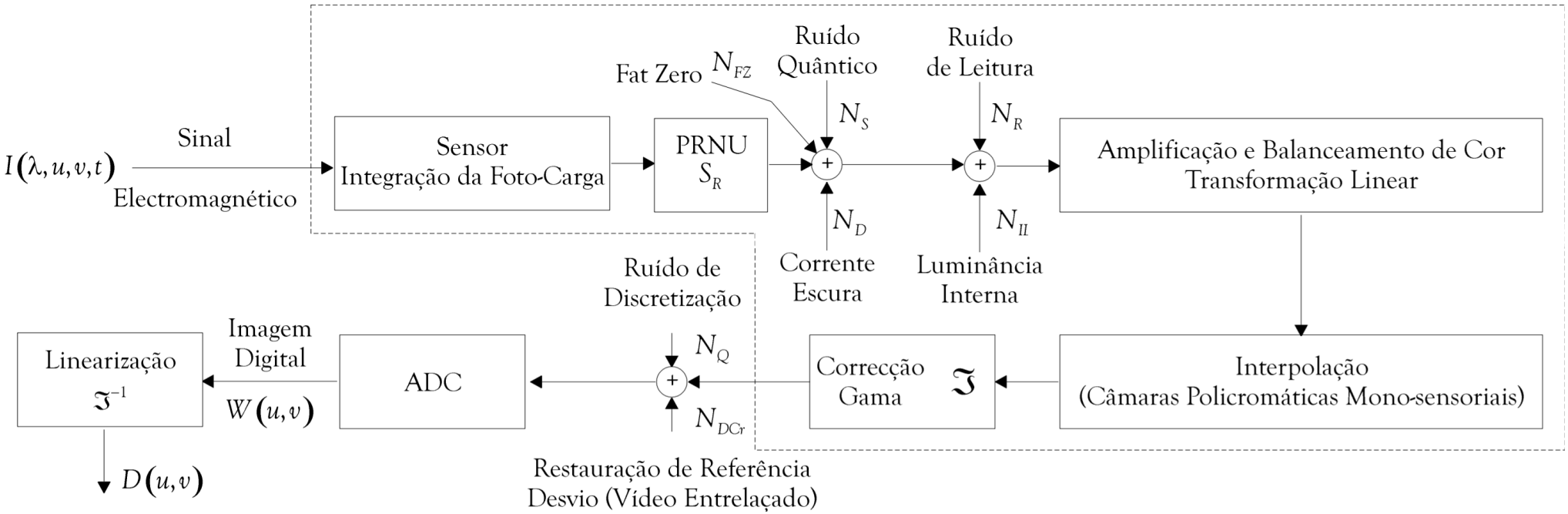
Feature Engineering - Intro

- Intelligent Systems



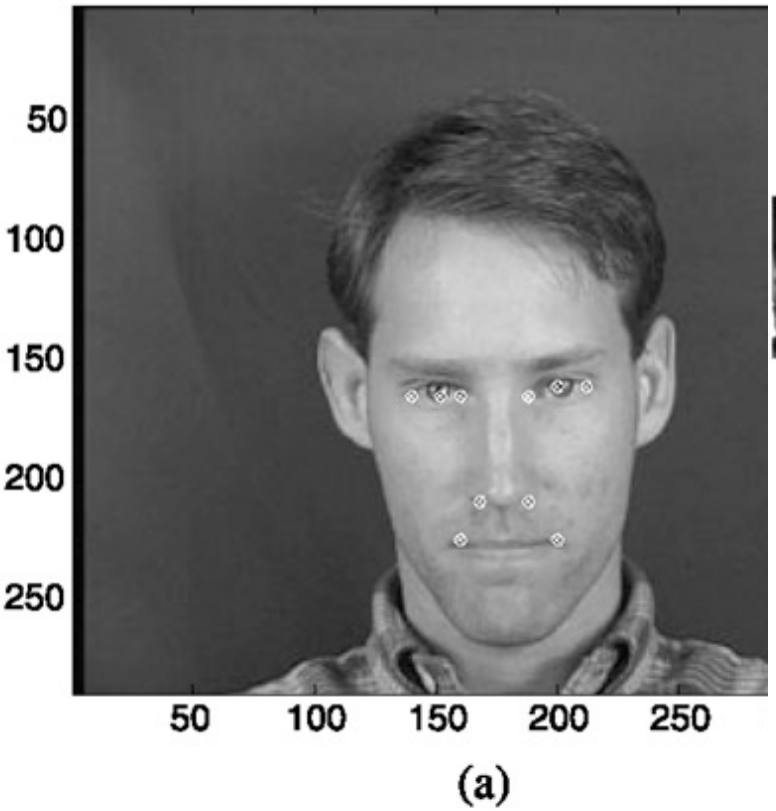
Feature Engineering - Intro

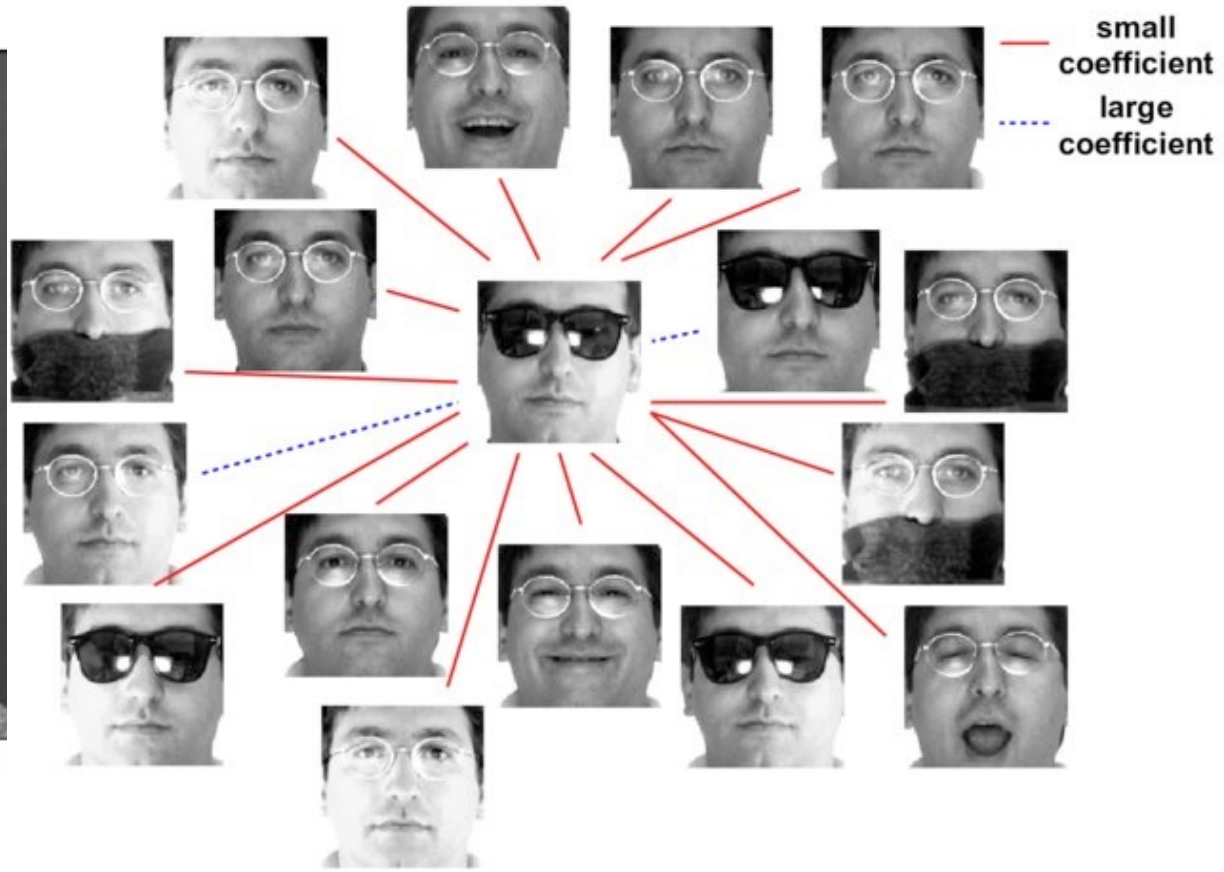
- Why feature engineering?
  - Noise



Feature Engineering - Intro

- Why feature engineering?
  - Noise

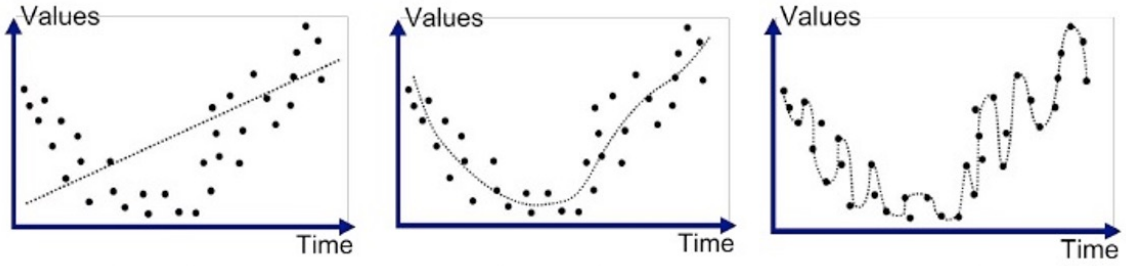


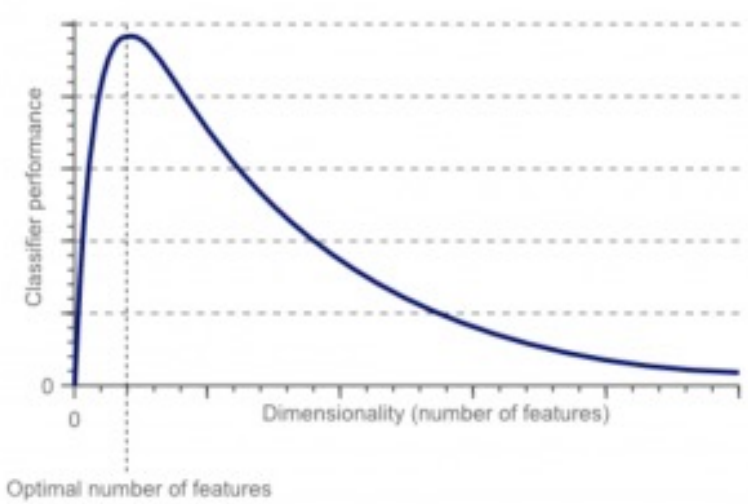


Feature Engineering - Intro

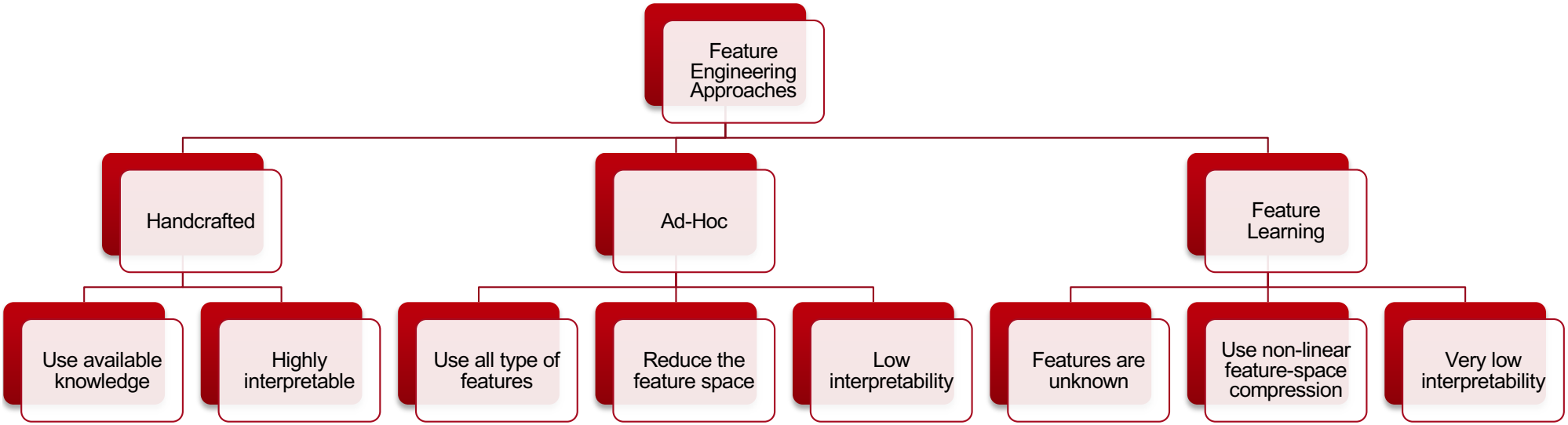
- Why feature engineering?
- Curse of dimensionality

The common theme of these problems is that when the dimensionality increases, the volume of the space increases so fast that the available data become sparse. This sparsity is problematic for any method that requires statistical significance. In order to obtain a statistically sound and reliable result, the amount of data needed to support the result often grows exponentially with the dimensionality.

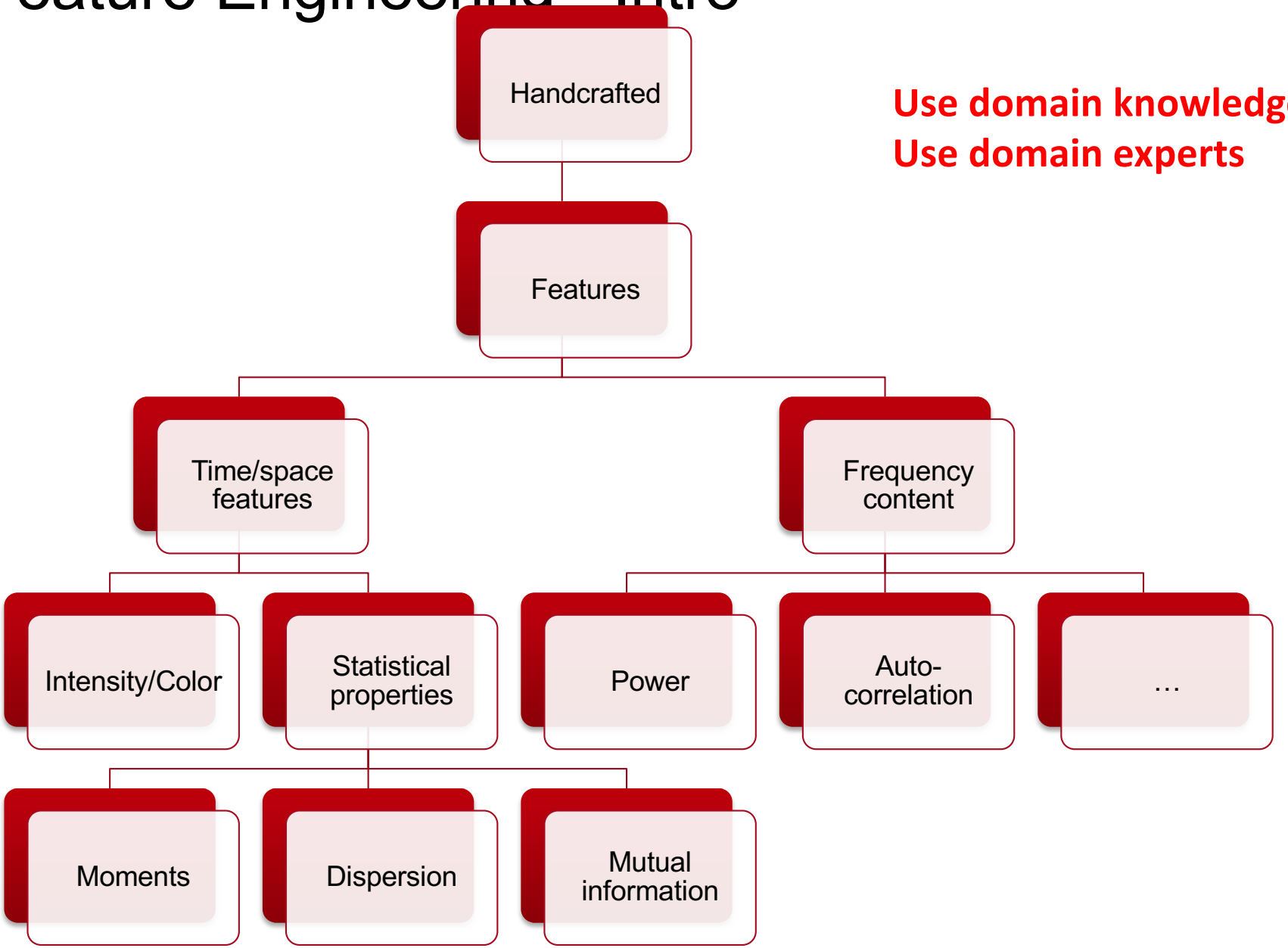




Feature Engineering - Intro



Feature Engineering - Intro



Use domain knowledge
Use domain experts

Feature Engineering - Intro

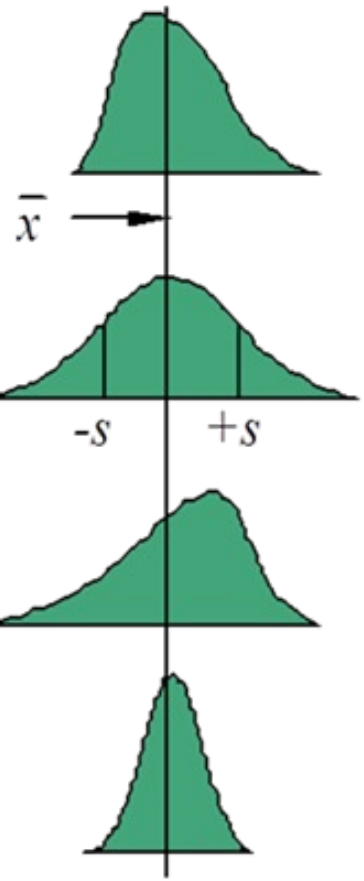
- Statistical properties

First Moment:
mean - measure of location

Second Moment:
Standard deviation - measure of spread

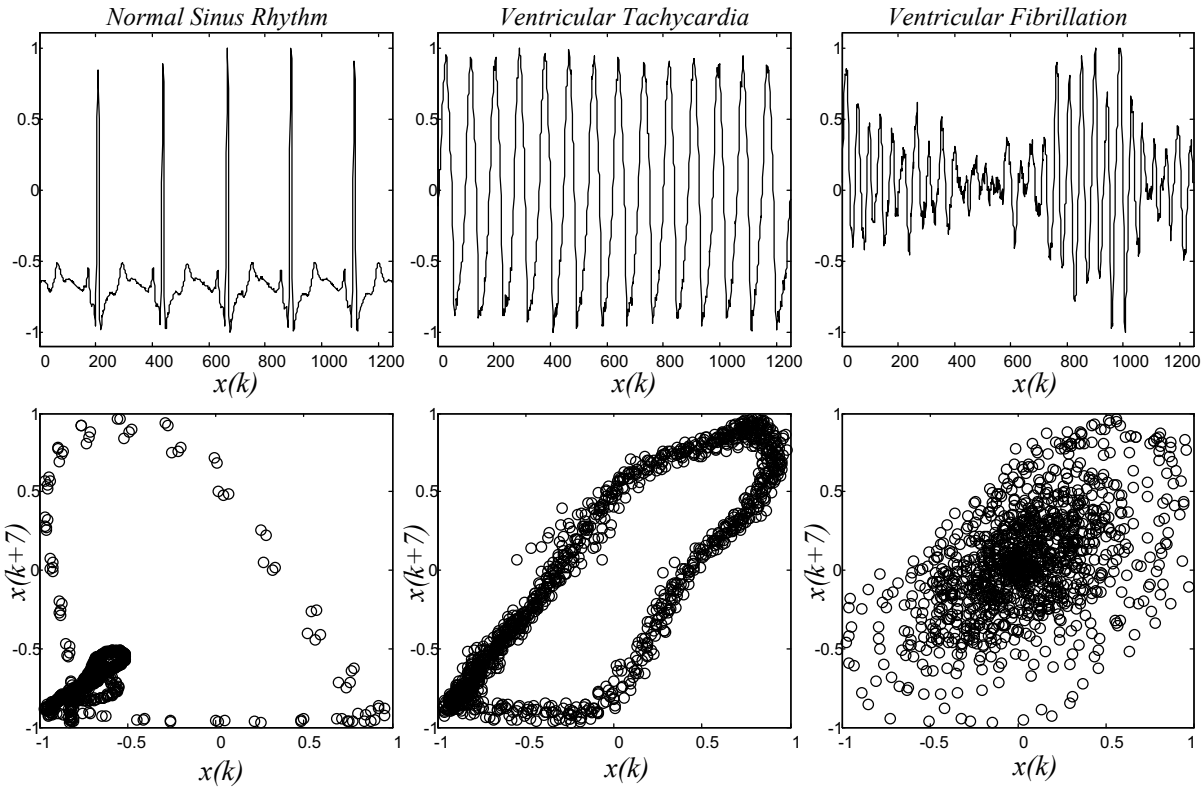
Third Moment:
skewness - measure of symmetry

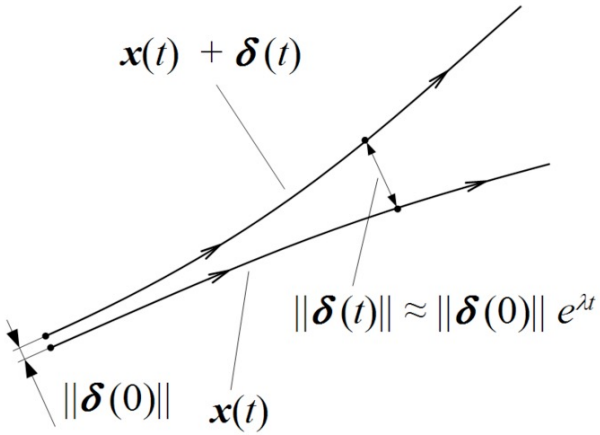
Fourth Moment:
kurtosis - measure of peakedness



Feature Engineering - Intro

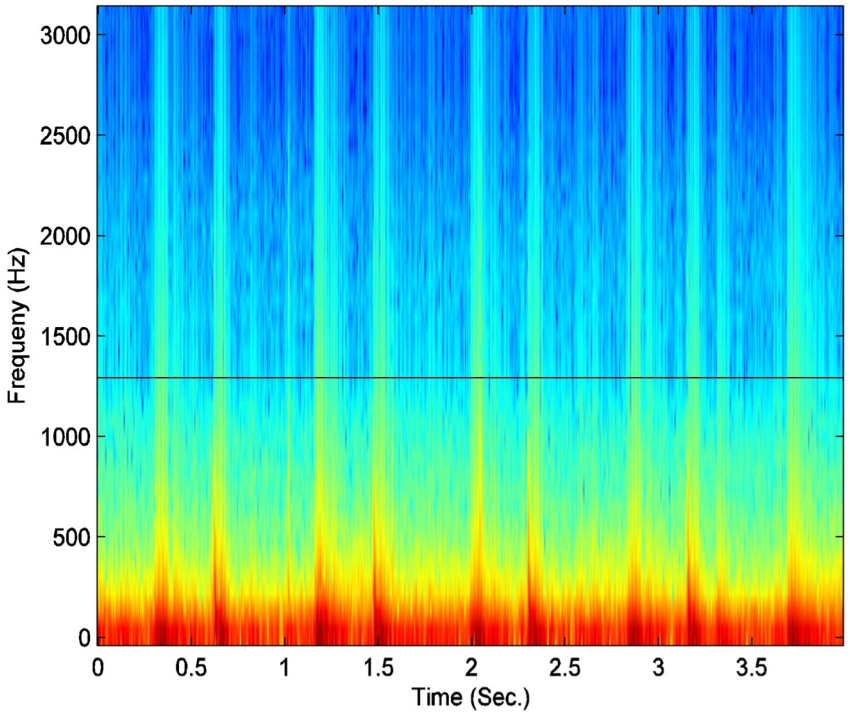
- Complexity

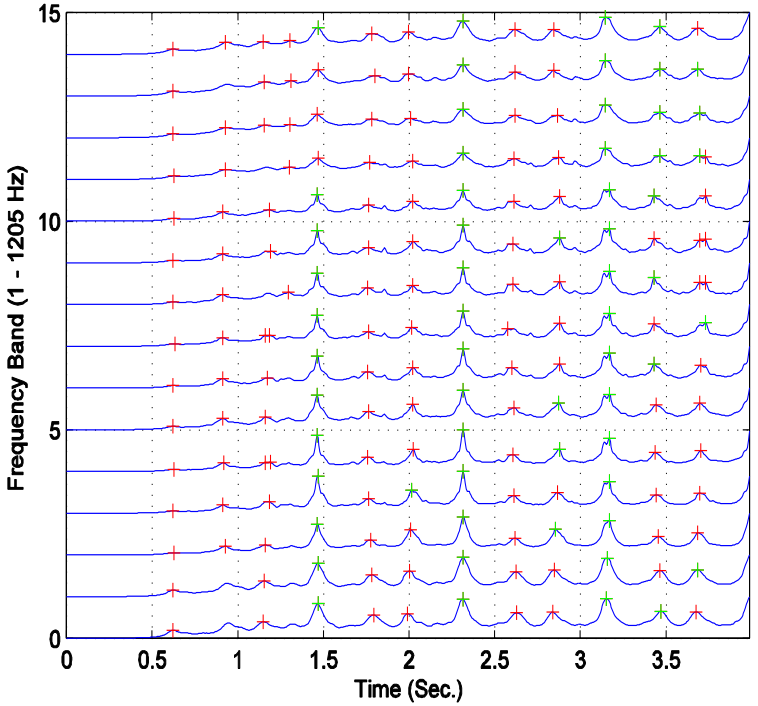




Feature Engineering - Intro

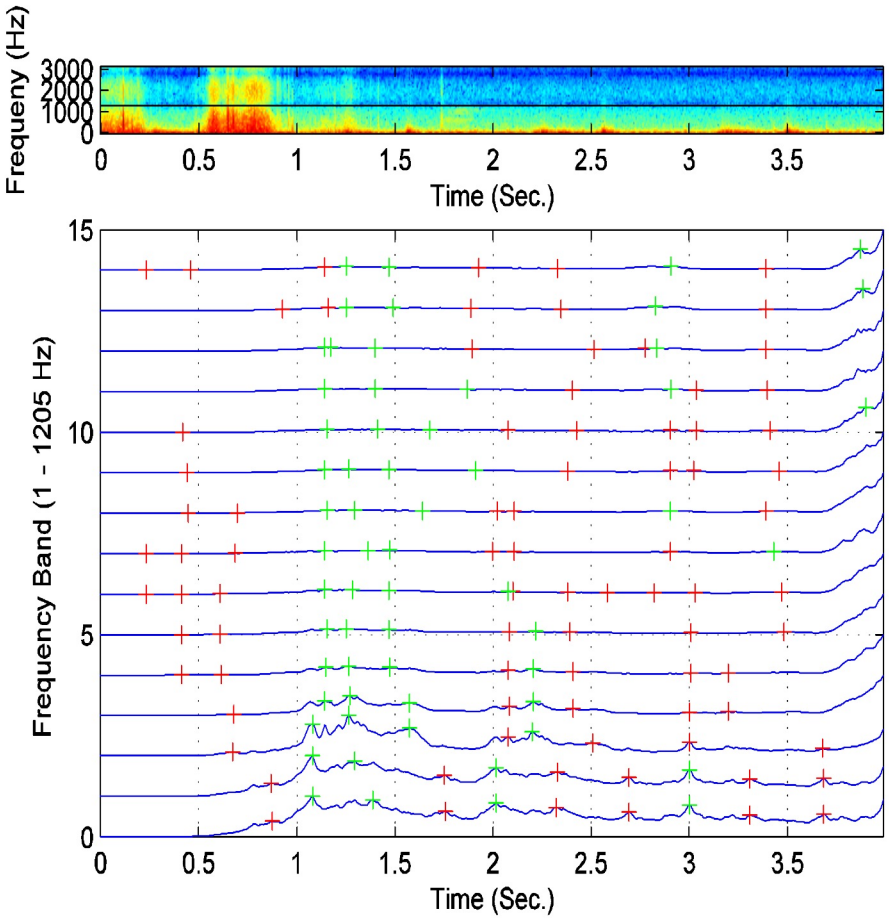
- Frequency content





Feature Engineering - Intro

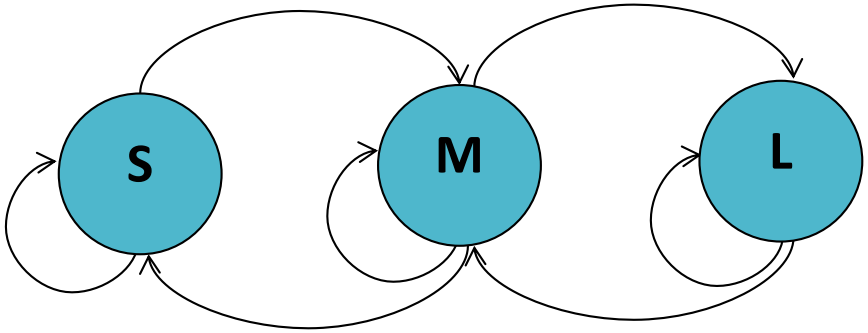
- Frequency content



Feature Engineering - Intro

- Example based on a priori knowledge: Identification of AF

- Regularity measure



$$P(x_i \mid x_{i-1}) = \begin{bmatrix} P(x_S \mid x_S) & P(x_S \mid x_M) & P(x_S \mid x_L) \\ P(x_M \mid x_S) & P(x_M \mid x_M) & P(x_M \mid x_L) \\ P(x_L \mid x_S) & P(x_L \mid x_M) & P(x_L \mid x_L) \end{bmatrix}$$

$$H(P(x_i, x_{i-1}))$$

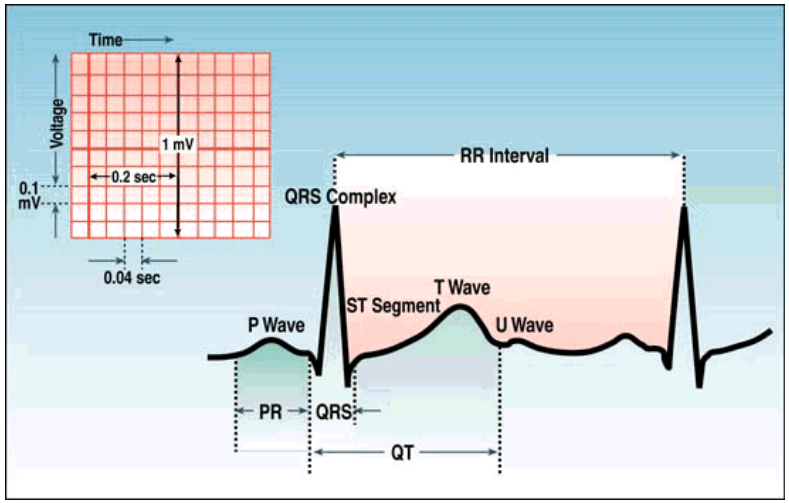
$$I(P(x_i, x_{i-1}), P_{Modelo}(x_i, x_{i-1}))$$

$$\sigma_t$$

Feature Engineering - Intro

- Example: Identification of AF

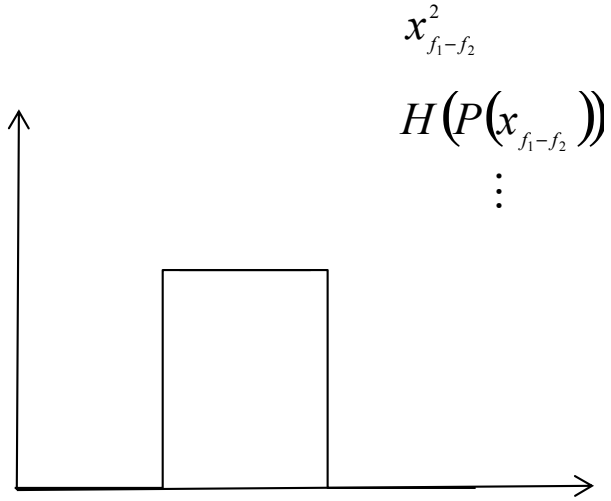
- Absence of P wave



Feature Engineering - Intro

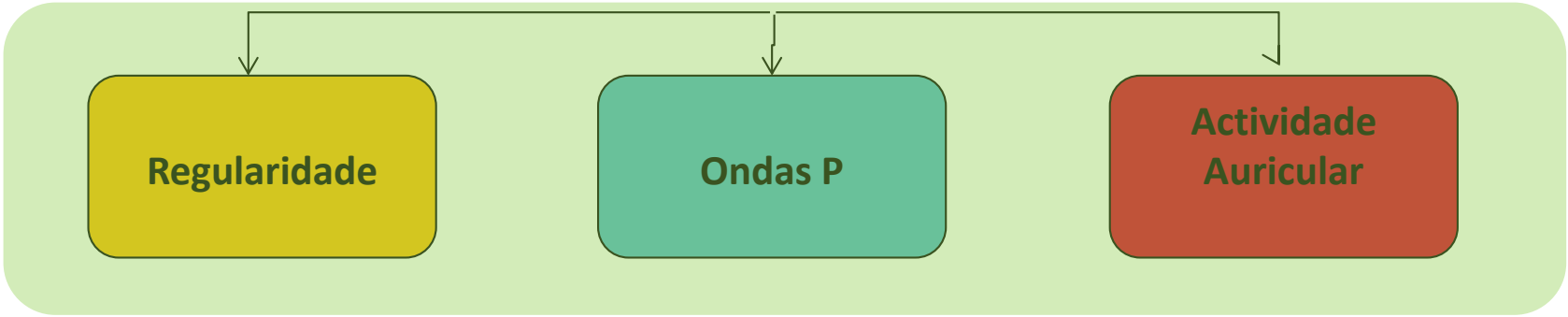
- Exemplo: Identificação de AF

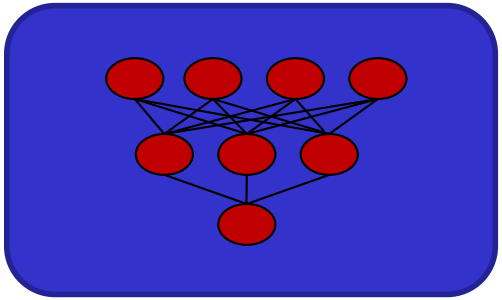
- Actividade de Fibrilhação



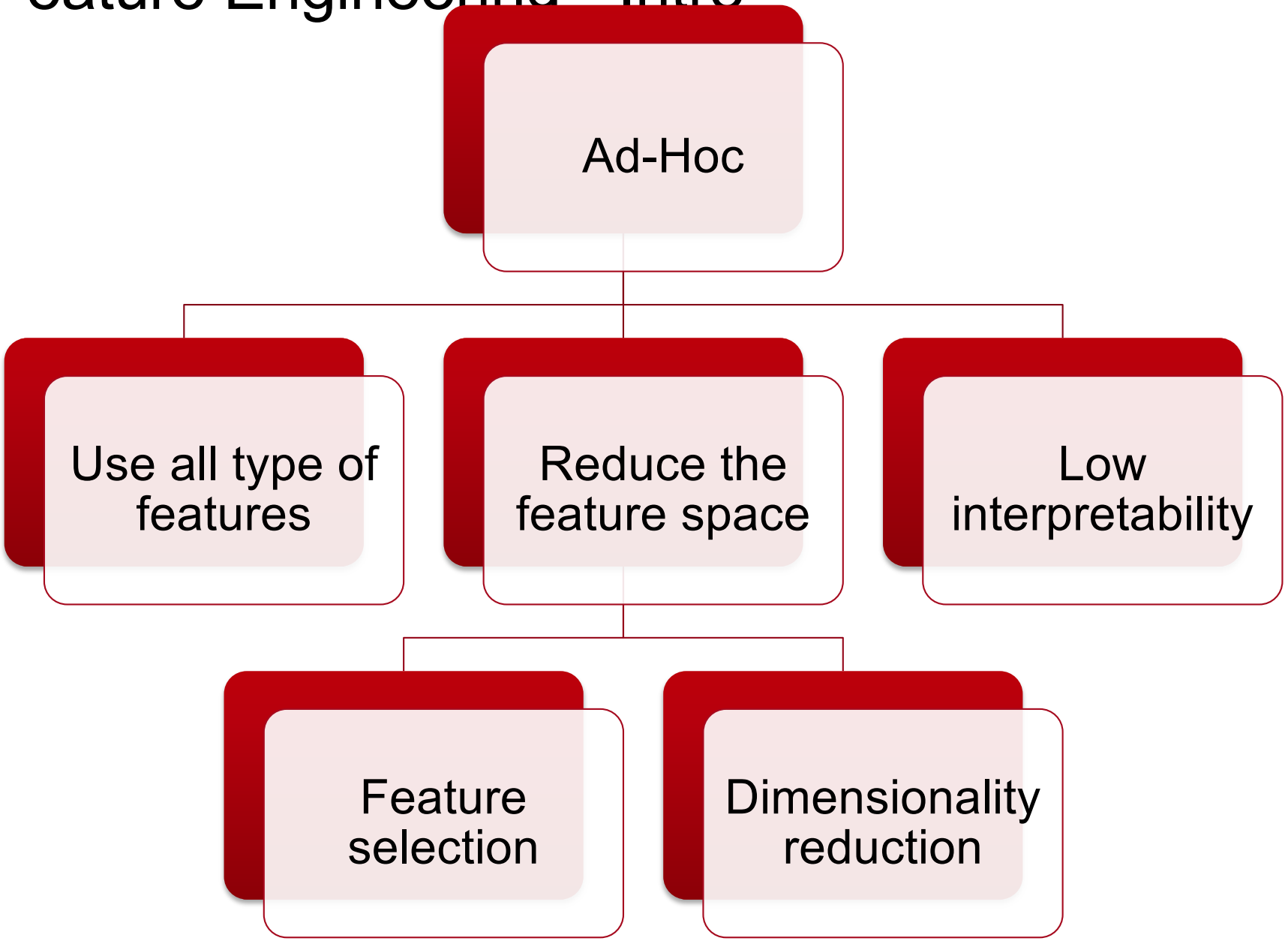
Feature Engineering - Intro

- Example: Identification of AF





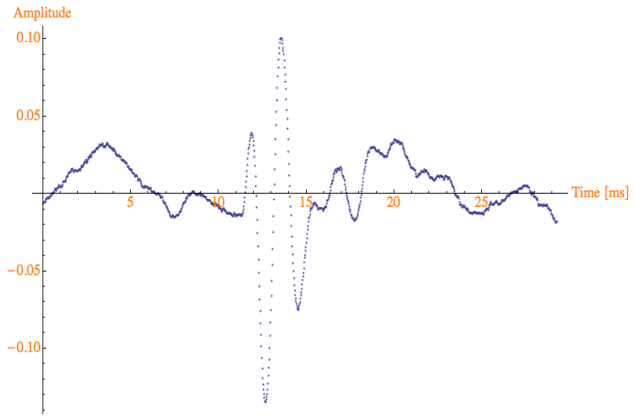
Feature Engineering - Intro



Ad-Hoc Example : Identification of Adventicious Sounds

- **Crackles**

  Crackles are short explosive sounds



- **Wheezes**

  Whistling sounds
  Musical in nature

- **Association:**

  Fluid in lower airways,

  Pulmonary edema, heart failu

- **Classification**
  - **Fine Crackles**

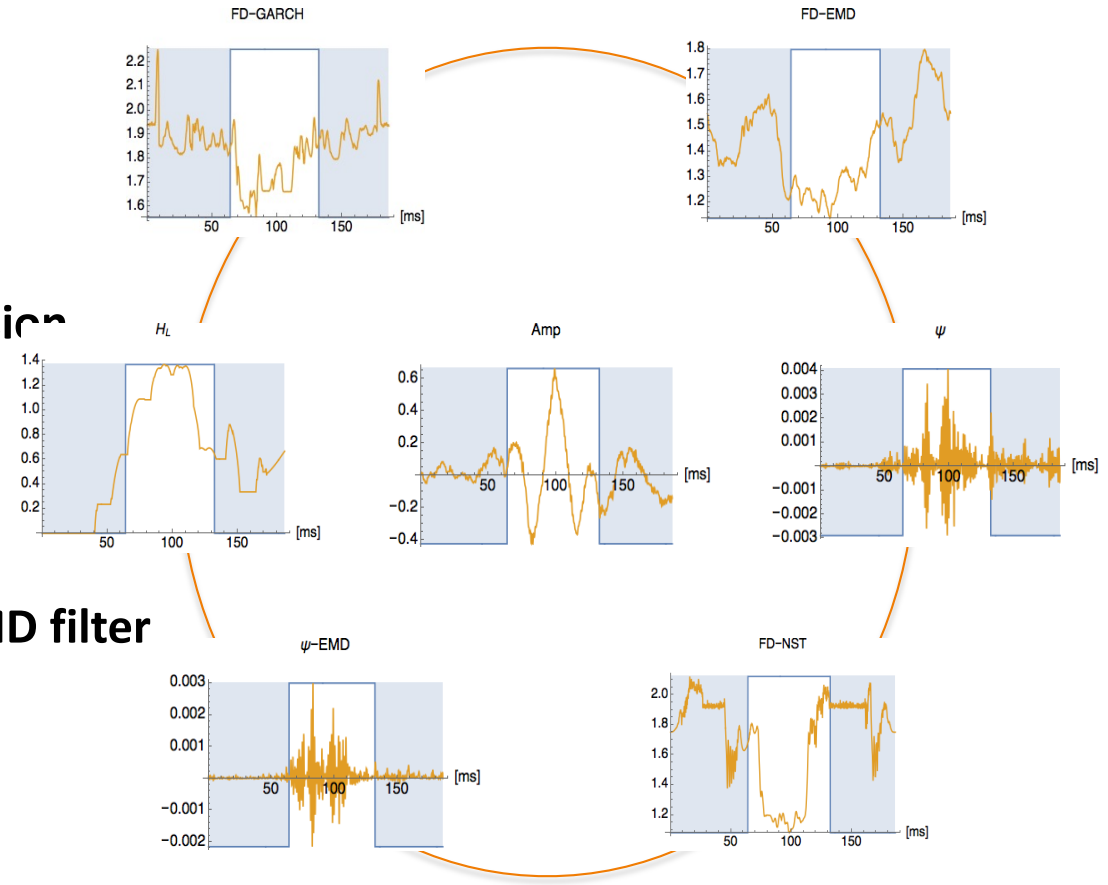
    High-pitched exclusively inspiratory;
  - **Coarse Crackles**

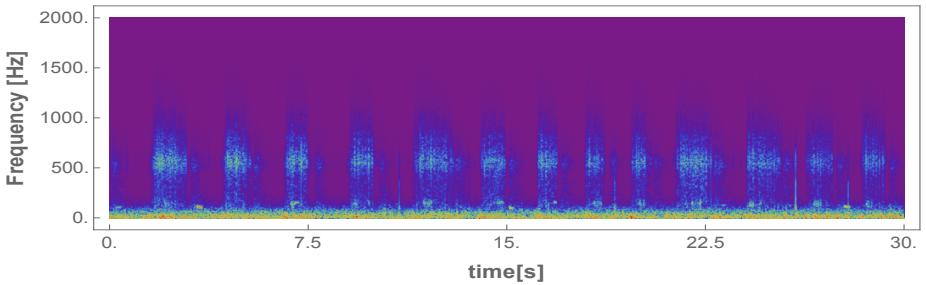
    Low-pitched sound events with a high amplitude and long duration

- **Association:**

  Air obstractions

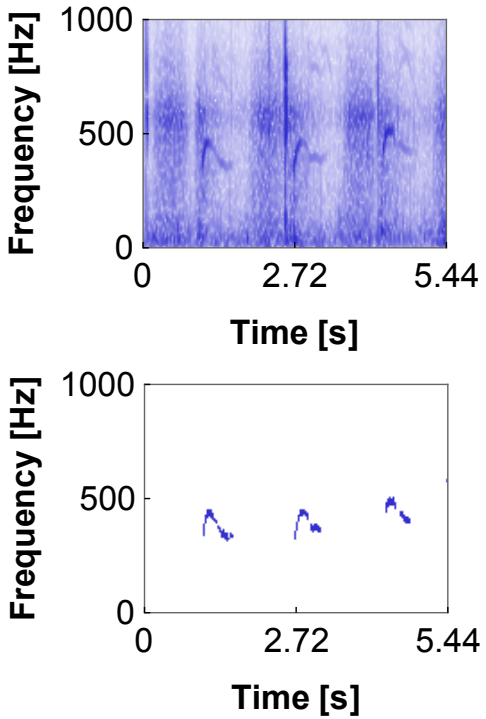
- **Processing algorithms:**
  - **H_L -> Local entropy (H_L)**
  - **FD-EMD ->Higutich Fractal dimension of a EMD filter**
  - **Ψ-EMD ->Teager energy of the EMD filter**





- RMS
- Spectral Centroid
- Spectral Brightness
- Spectral Spread
- Skewness
- Spectral Kurtosis
- Spectral Rolloff 85
- Spectral Rolloff 75
- Spectral. Flatness
- Spectra Irregularity
- Chromagram centroid
- Chromagram peak
- Zerocross
- Keyclarity
- 13 MFCC
- Pitch
- Inharmonicity
- Roughness

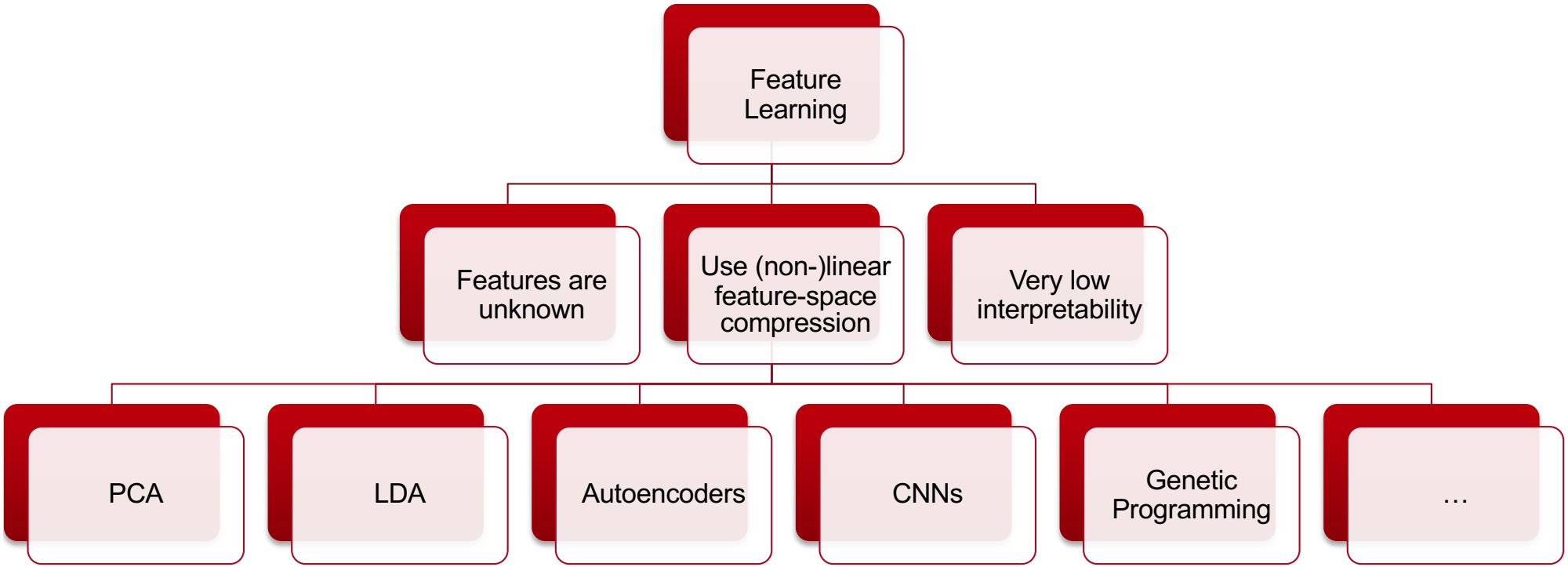
- **Detection of the signature of wheezes in the spectrogram space (WS-SS)**



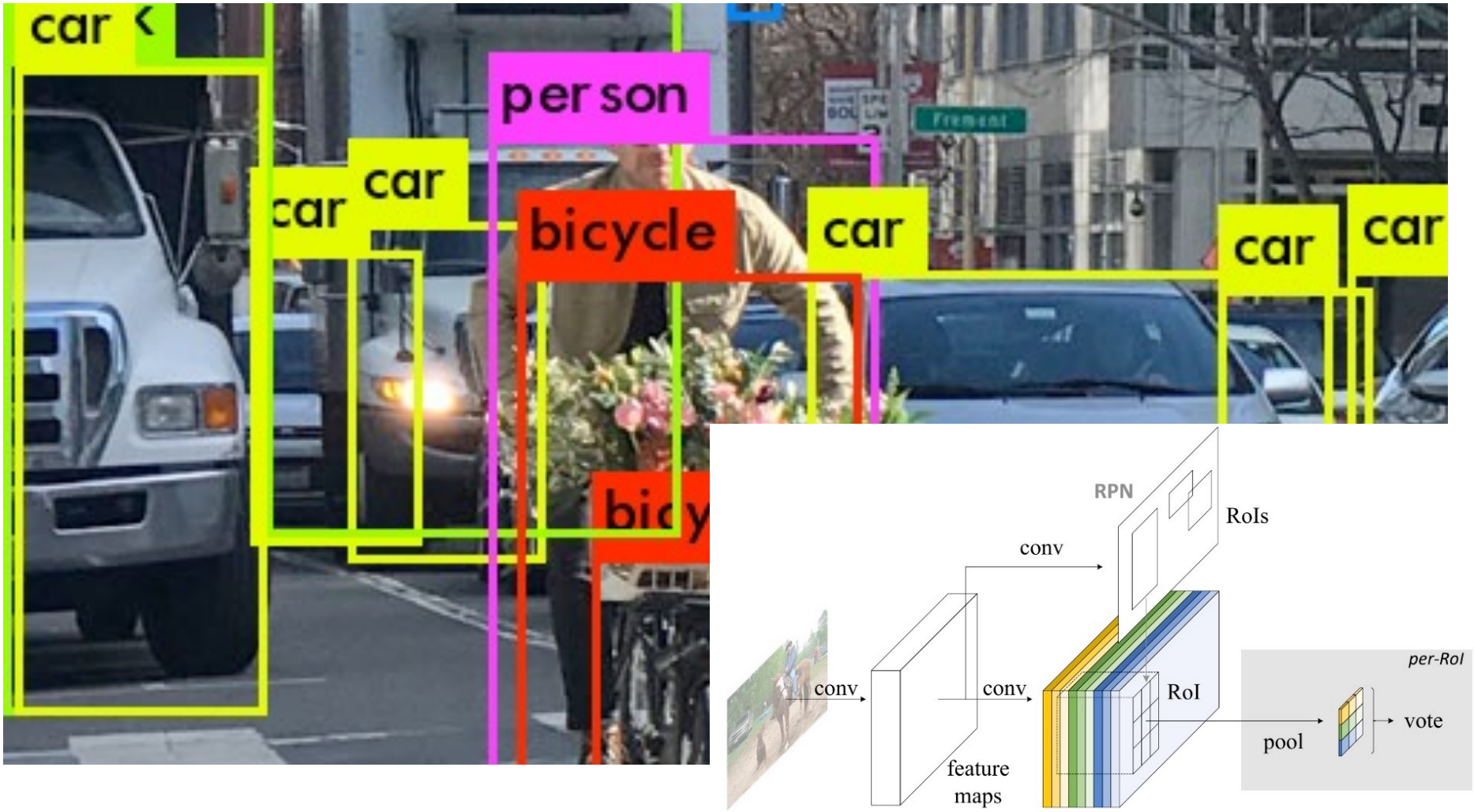
signature of the wheezes in the spectrogram space

Features	Crackles	Wheezes
Teager energy	x	
Fractal dimension	x	
Entropy	x	
WS-SS	x	x
29 Musical features	x	x

Feature Engineering - Intro



Feature Engineering – Intro
Example for feature learning



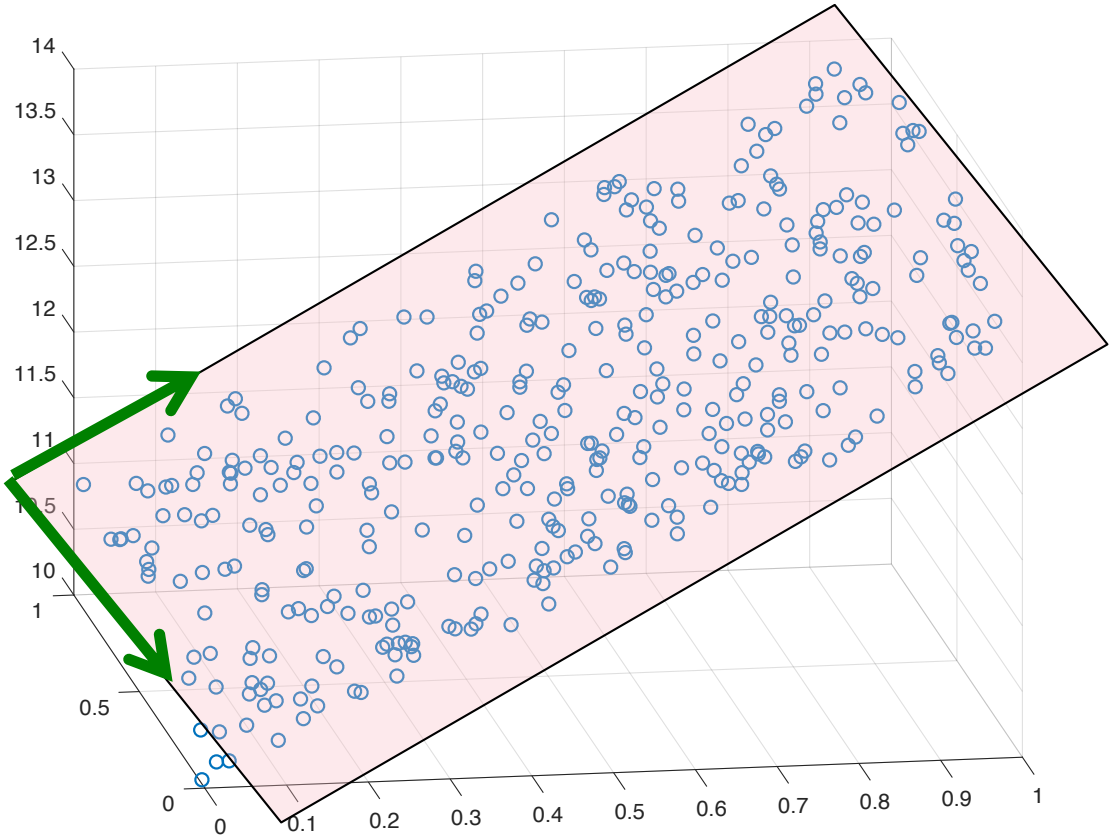
Engenharia de Características para Aprendizagem Computacional /

Engenharia de Atributos

Chapter 5: Feature Engineering: Dimensionality Reduction

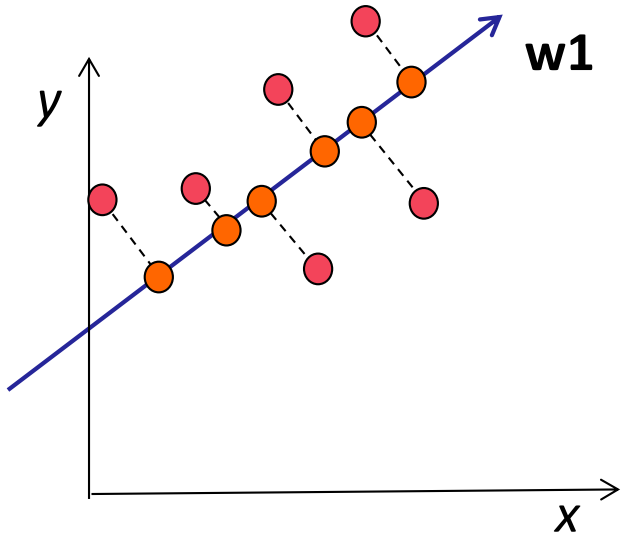
PCA-Principal Component Analysis

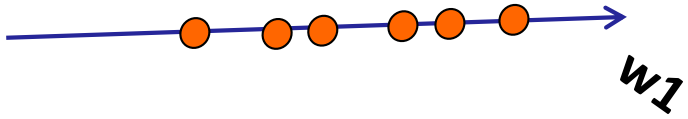
- Idea: Compute set of alternative orthogonal directions that “explain the data”



PCA-Principal Component Analysis

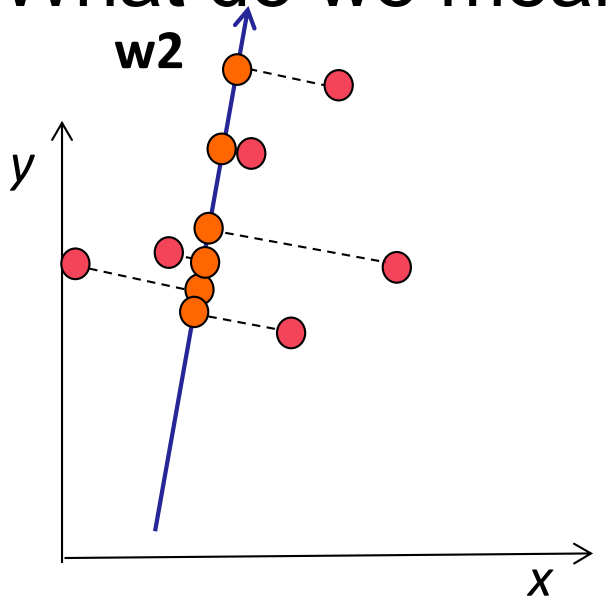
- What do we mean by “explain the data”?

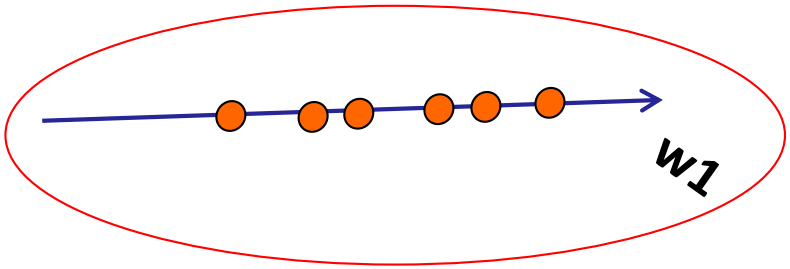


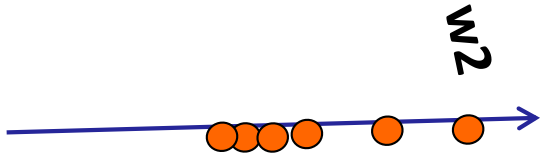


PCA-Principal Component Analysis

- What do we mean by “explain the data”?



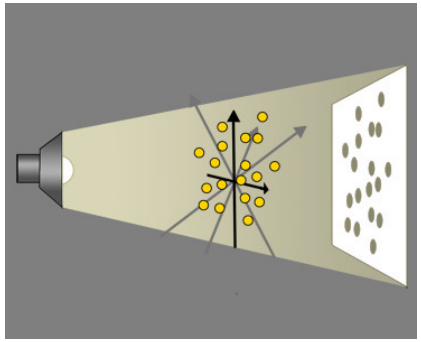


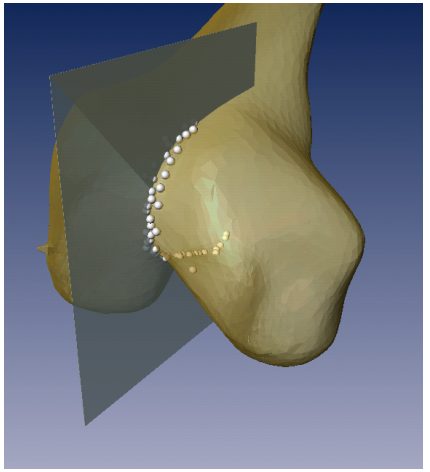


- Maximal variability of Data!

PCA-Principal Component Analysis

- Problem: Compute the orthogonal directions of maximal variability of the data
  - Reduction of dimensionality
    - Visualization
    - Feature learning
    - Compression
  - Interpretabilidade



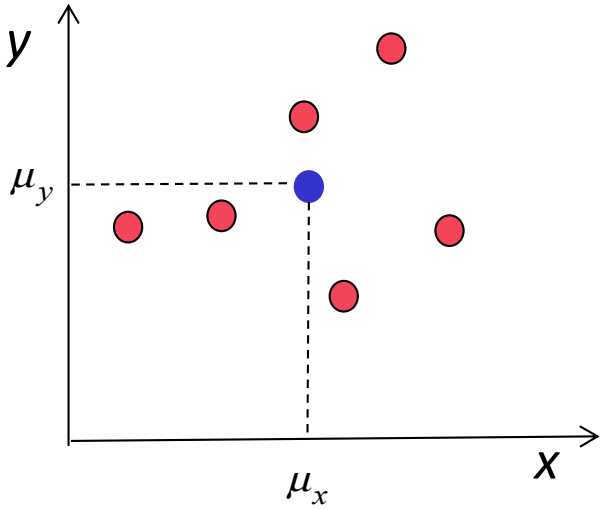


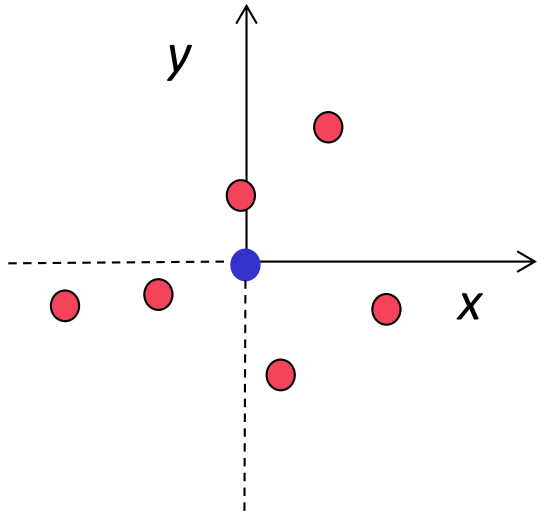
PCA – Principal Component Analysis

- Maximal variability of the data – Interpretation
- First: Centre the data

$$x \leftarrow x - \mu_x \qquad \mu_x = \frac{1}{n}\sum_{i=1}^{n} x_i$$

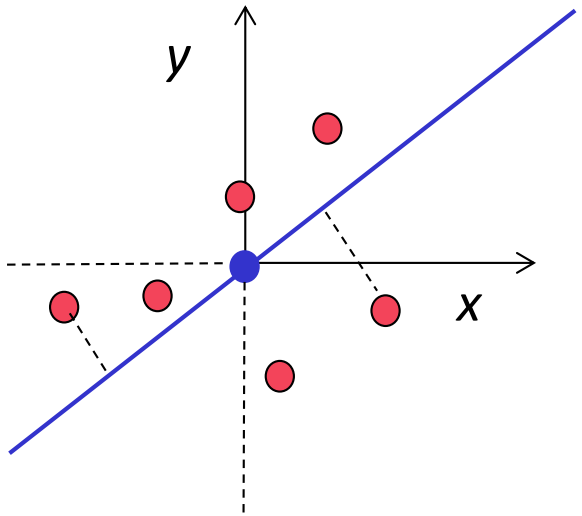
$$y \leftarrow y - \mu_y \qquad \mu_y = \frac{1}{n}\sum_{i=1}^{n} y_i$$





PCA – Principal Component Analysis

- Maximal variability of the data – Interpretation
- Second: fit best line through centre



PCA – Principal Component Analysis

- Maximal variability of the data – Interpretation
- Second: fit best line through centre

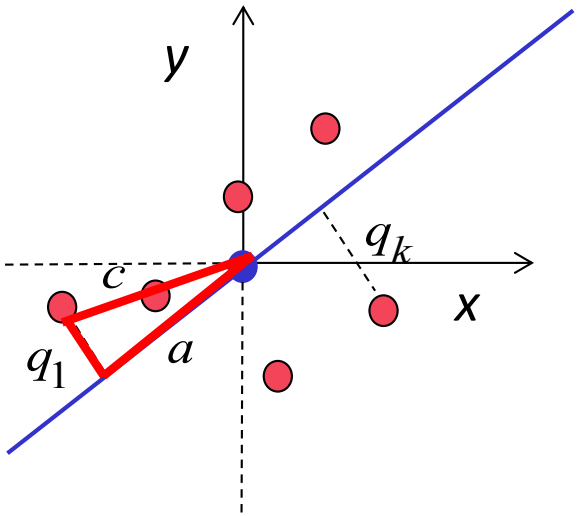
$$\text{argmin}\left\{\sum_{i=1}^{n} q_i^2\right\}$$

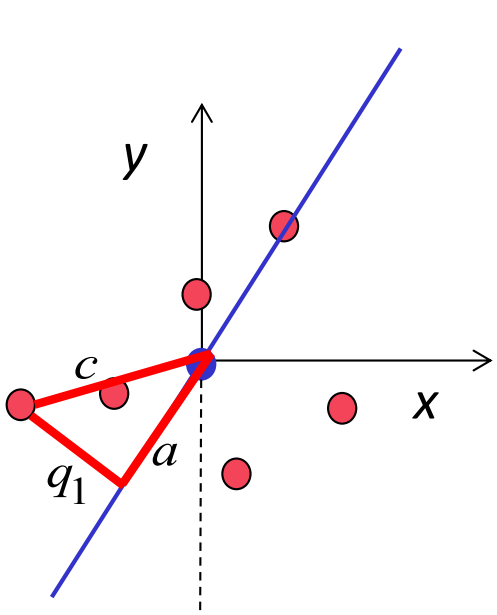
$$q_1^2 + a^2 = c^2$$

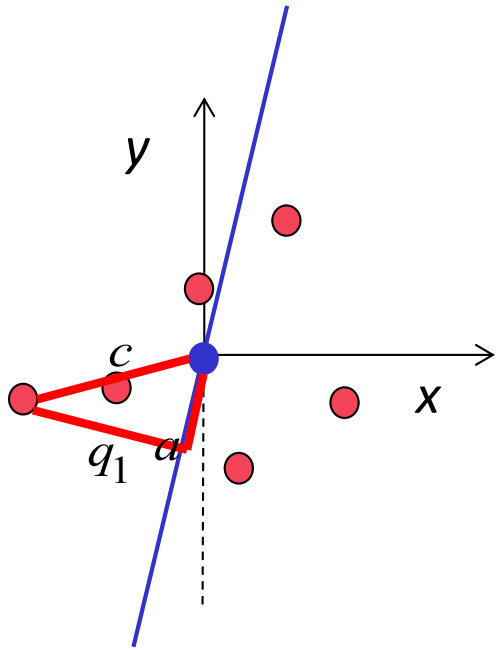
$$c^2 = \text{constant}$$

$$q_1 \downarrow \Leftrightarrow a \uparrow$$

$$q_1 \uparrow \Leftrightarrow a \downarrow$$

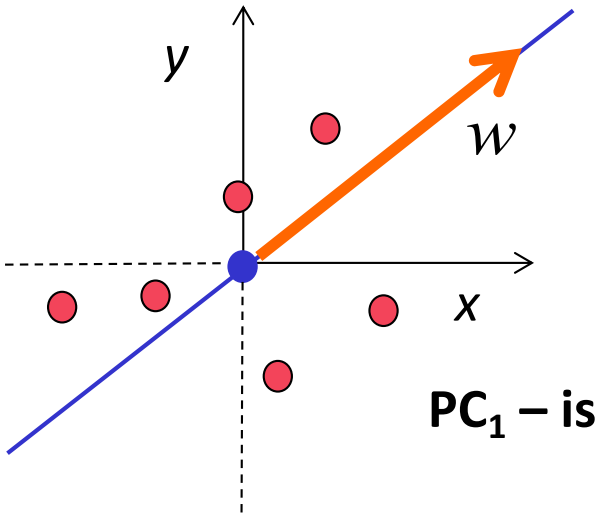






PCA – Principal Component Analysis

- Maximal variability of the data – Interpretation
- Second: fit best line through centre



$$\text{argmin}\left\{\sum_{i=1}^{n} q_i^2\right\} \Leftrightarrow \arg\max\left\{\sum_{i=1}^{n} d_i^2\right\}$$

$$PC_1 \equiv w$$

$$PC_1 = \underbrace{\cos(\alpha)}_{\gamma_1} x + \underbrace{\sin(\alpha)}_{\gamma_2} y$$

PC_1 – is a linear combination of the features

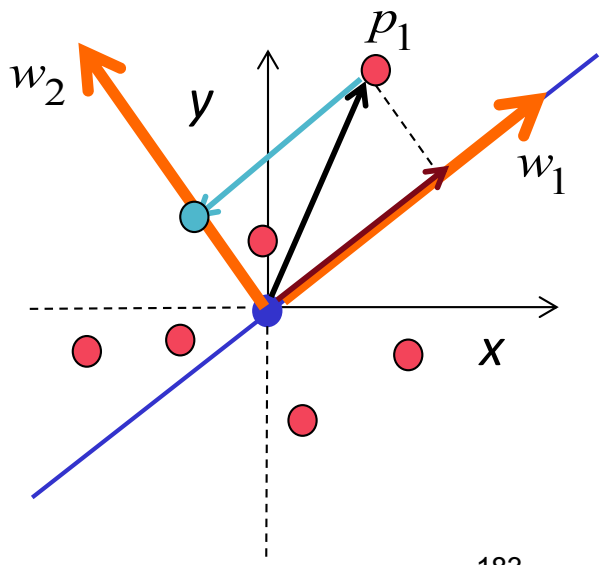
PC_1 – is a rotation of the original space

γ_1/γ_2 **x relevance wrt y to describe PC1**

PCA – Principal Component Analysis

- Maximal variability of the data – Interpretation
- Third: Remove PC_1 and repeat the process until you have exhausted the number samples or dimensions

$$P_k = P - \sum_{s=1}^{k-1} w_s^T P w_s$$



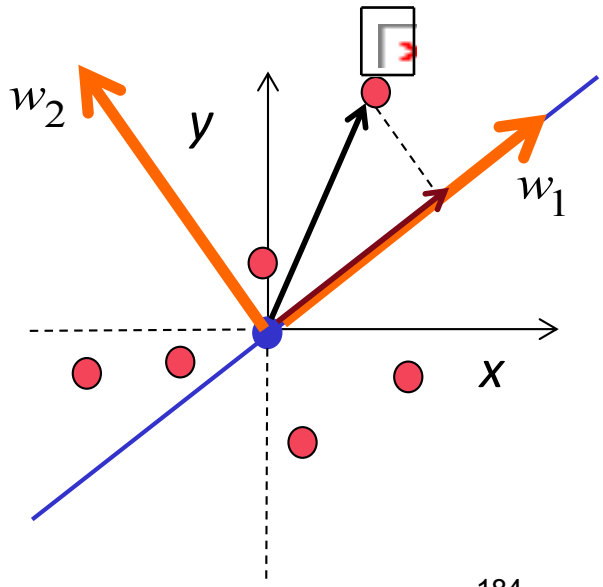
PCA – Principal Component Analysis

- Notation

$$p_i \equiv \begin{bmatrix} p_{i,x} & p_{i,y} & p_{i,z} & \cdots \end{bmatrix}^T$$ ← Point

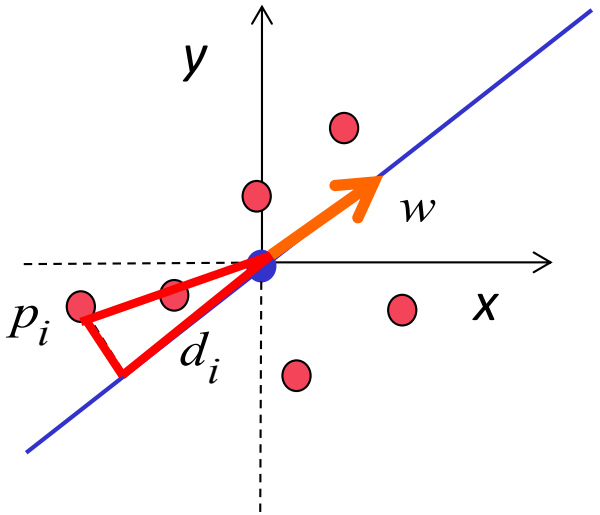
$$w_i \equiv \begin{bmatrix} w_{i,x} \\ w_{i,y} \\ w_{i,z} \\ \vdots \end{bmatrix}$$ ← Vector

$$P \equiv \begin{bmatrix} p_1^T \\ p_2^T \\ p_3^T \\ \vdots \end{bmatrix}$$ ← Set of Points



PCA – Principal Component Analysis

- Algorithm
  - Maximize variability of data
  - Subject to unity length



$$\text{argmax}\left\{\sum_{i=1}^{n}\left(p_i^T w\right)^2\right\} = \text{argmax}\left\{w^T P^T P w\right\}$$

$$subject: \|w\|^2 = 1$$

$$J = w^T P^T P w - \lambda\left(w^T w - 1\right)$$

$$\frac{\partial J}{\partial w} = 0 \Leftrightarrow w^T P^T P - \lambda w^T = 0$$

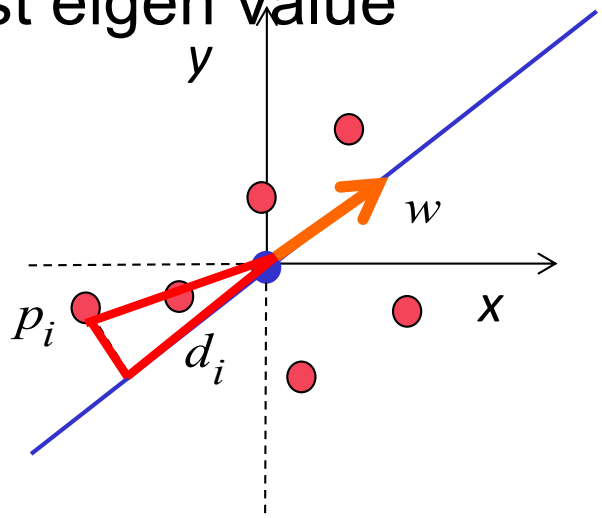
$$\frac{\partial J}{\partial w} = 0 \Leftrightarrow P^T P w = \lambda w$$

PCA – Principal Component Analysis

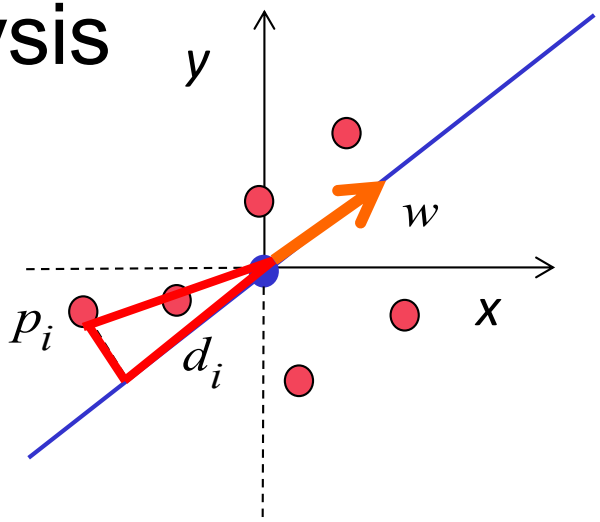
$$\frac{\partial J}{\partial w} = 0 \Leftrightarrow P^T P w = \lambda w$$

- Algorithm
  - Principal Components -> eigen vectors of P^TP
    - PC_1 – eigen vector with largest eigen value
    - PC_2 – eigen vector with second largest eigen value
    - …
    - PC_k – eigen vector with kth largest eigen value
  - Popular algorithm:
    - Singular value decomposition

$$SVD(F) = U \Sigma V^T$$



PCA – Principal Component Analysis



- Algorithm
  - Singular value decomposition

$$SVD(F) = U\Sigma V^T$$

$$F \in R^{m\times n}, U = \begin{bmatrix} u_1 & u_2 & \dots & u_m \end{bmatrix} \in R^{m\times m}, \|u_i\|^2 = 1, u_i^T u_j = \begin{cases} 1 \Leftarrow i = j \\ 0 \Leftarrow i \neq j \end{cases}$$

$$\Sigma = diag(\lambda_i) \in R^{m\times n}, i = \min(m,n)$$

$$V = \begin{bmatrix} v_1 & v_2 & \dots & v_n \end{bmatrix} \in R^{n\times n}, \|v_i\|^2 = 1, v_i^T v_j = \begin{cases} 1 \Leftarrow i = j \\ 0 \Leftarrow i \neq j \end{cases}$$

$$SVD(P^T P) = V\Sigma^T U^T U\Sigma V^T = V\Sigma^2 V^T \quad \Sigma^2 = diag(\lambda_i^2), i = 1,\dots,\min(n,m)$$

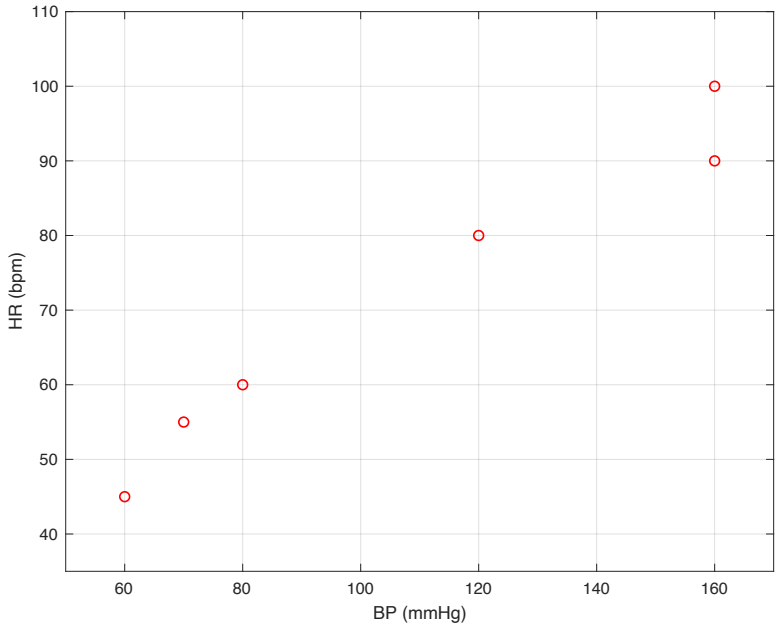
PCA – Principal Component Analysis

- Closer look at the matrixes: $P^T P$

Samples

Variables observed

	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)	160	160	120	80	70	60
HR (bpm)	100	90	80	60	55	45



PCA – Principal Component Analysis

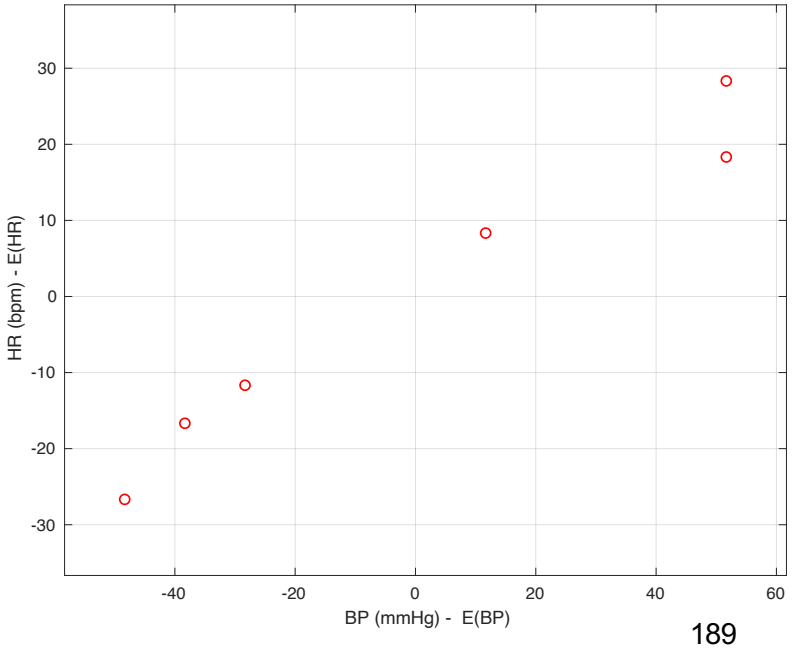
- Centre around E(P)

Samples

Variables observed

	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)-E(BP)	51,7	51,7	11,7	-28,3	-38,3	-48,3
HR (bpm)-E(HR)	28,3	18,3	8,3	-11,7	-16,7	-26,7

$$P \equiv \begin{bmatrix} P_1^T \\ P_2^T \\ \vdots \\ P_n^T \end{bmatrix}, P_i \equiv \begin{bmatrix} BP_i \\ HR_i \end{bmatrix}$$



PCA – Principal Component Analysis

- $P^T P$ – co-variance

$$P \equiv \begin{bmatrix} BP_1 - E[BP] & HR_1 - E[HR] \\ BP_2 - E[BP] & HR_2 - E[HR] \\ \vdots & \vdots \\ BP_6 - E[BP] & HR_6 - E[HR] \end{bmatrix}$$

$$P^T P = \begin{bmatrix} BP_1 - E[BP] & BP_2 - E[BP] & \cdots & BP_6 - E[BP] \\ HR_1 - E[HR] & HR_2 - E[HR] & \cdots & HR_6 - E[HR] \end{bmatrix} \begin{bmatrix} BP_1 - E[BP] & HR_1 - E[HR] \\ BP_2 - E[BP] & HR_2 - E[HR] \\ \vdots & \vdots \\ BP_6 - E[BP] & HR_6 - E[HR] \end{bmatrix}$$

$$= \begin{bmatrix} \text{cov}(BP, BP) & \text{cov}(BP, HR) \\ \text{cov}(HR, BP) & \text{cov}(HR, HR) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{k=1}^{6} \left(BP_k - E[BP]\right)^2 & \sum_{k=1}^{6} \left(BP_k - E[BP]\right)\left(HR_k - E[HR]\right) \\ \sum_{k=1}^{6} \left(HR_k - E[HR]\right)\left(BP_k - E[BP]\right) & \sum_{k=1}^{6} \left(HR_k - E[HR]\right)^2 \end{bmatrix}$$

PCA – Principal Component Analysis

- P^TP – co-variance

$$P^T P = \begin{bmatrix} \mathrm{cov}(BP,BP) & \mathrm{cov}(BP,HR) \\ \mathrm{cov}(HR,BP) & \mathrm{cov}(HR,HR) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{k=1}^{6}\left(BP_k - E[BP]\right)^2 & \sum_{k=1}^{6}\left(BP_k - E[BP]\right)\left(HR_k - E[HR]\right) \\ \sum_{k=1}^{6}\left(HR_k - E[HR]\right)\left(BP_k - E[BP]\right) & \sum_{k=1}^{6}\left(HR_k - E[HR]\right)^2 \end{bmatrix}$$

$$P^T P = \begin{bmatrix} \sigma_{BP}^2 & \rho_{BP,HR}^2\left(\sigma_{BP}\sigma_{HR}\right) \\ \rho_{HR,BP}^2\left(\sigma_{BP}\sigma_{HR}\right) & \sigma_{HR}^2 \end{bmatrix}$$

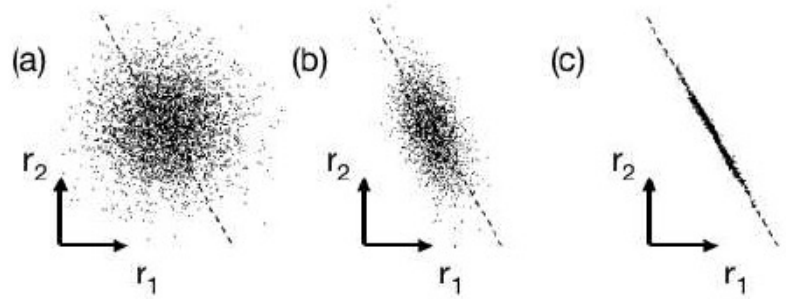
CORRELATION

PCA – Principal Component Analysis

- (Co)-variance
  - Normalized co-variance -> correlation

$$\rho_{i,j} \leftarrow \frac{\sigma_{i,j}}{\sqrt{\sigma_{i,i}\sigma_{j,j}}}$$

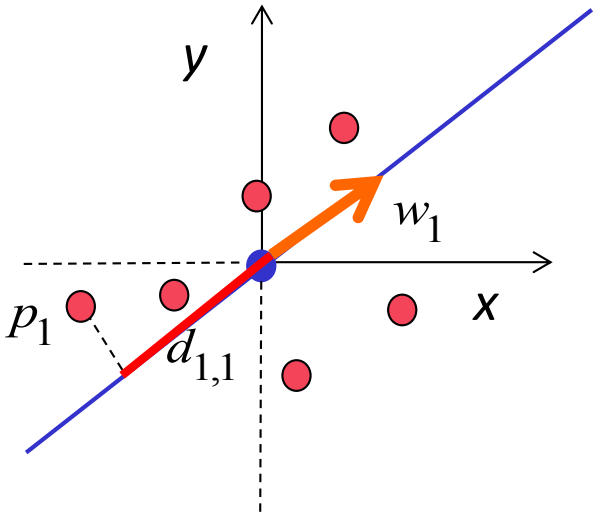
  - If $y = kx + b$ $\quad \rho = \begin{cases} 1 \Leftarrow k > 0 \\ -1 \Leftarrow k < 0 \end{cases}$
    - Same information
    - Discard



PCA – Principal Component Analysis

- $\Sigma^2 = diag(\lambda_i^2), i = 1,\ldots,\min(n,m)$

$$\lambda_i^2 = \sum_{k=1}^{n} \left(P_k w_i \right)^2 = \sum_{k=1}^{n} \left(d_{k,i} \right)^2$$



Dispersion measure around a given Principal Component

$$\lambda_i^2 = \sum_{k=1}^{n} \left(P_k w_i \right)^2 = w_i^T P^T P w_i = w_i^T V \Sigma^2 V^T w_i$$

$$SVD\left(P^T P \right) = V \Sigma^2 V^T$$

$$w_i^T V = \begin{cases} 1 \Leftarrow coluna\ i \\ 0 \Leftarrow c.c. \end{cases}$$

PCA – Principal Component Analysis

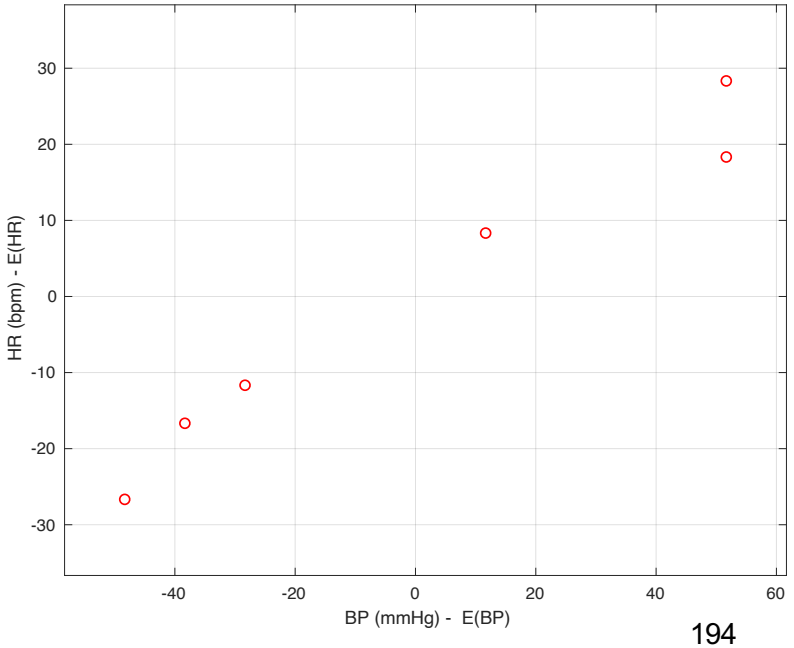
- $\Sigma^2 = diag(\lambda_i^2), i = 1,\ldots,\min(n,m)$

Samples

Variables observed

	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)- E(BP)	51,7	51,7	11,7	-28,3	-38,3	-48,3
HR (bpm)- E(HR)	28,3	18,3	8,3	-11,7	-16,7	-26,7

$$P \equiv \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{bmatrix} = \begin{bmatrix} (28.3;51.7) \\ (51.7;18.3) \\ (11.7;8.3) \\ (-28.3;-11.7) \\ (-38.3;-16.7) \\ (-48.3;-26.7) \end{bmatrix}$$



PCA – Principal Component Analysis

- $\Sigma^2 = diag(\lambda_i^2), i = 1,\ldots,\min(n,m)$

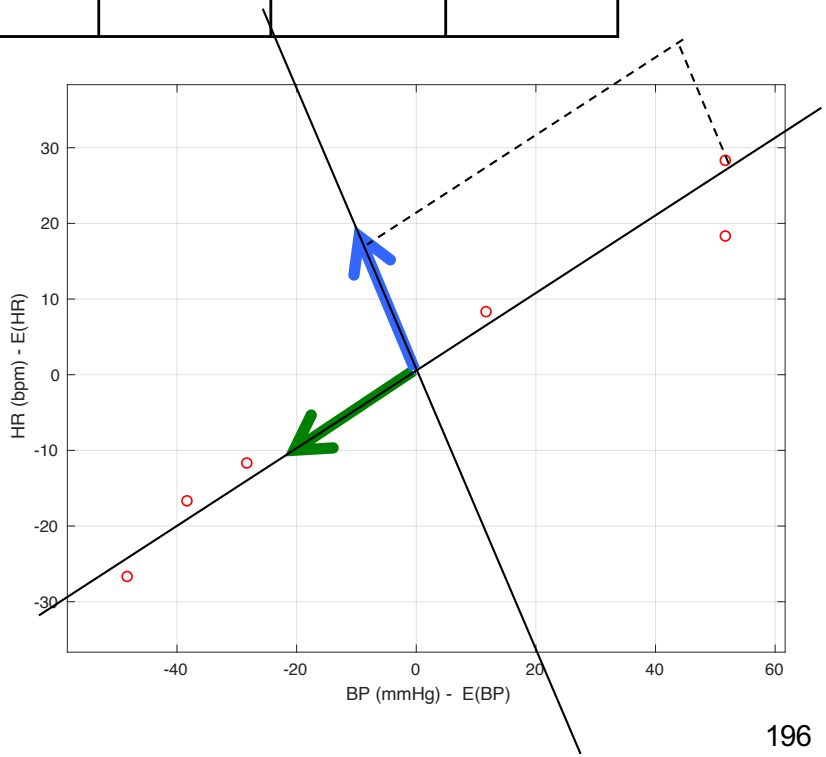
Samples

Variables observed

	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)-E(BP)	51,7	51,7	11,7	-28,3	-38,3	-48,3
HR (bpm)-E(HR)	28,3	18,3	8,3	-11,7	-16,7	-26,7

$$P^T P \equiv \begin{bmatrix} 10083 & 4767 \\ 4767 & 2333 \end{bmatrix}$$

$$SVD(P^T P) \equiv \begin{bmatrix} -0.9030 & -0.4297 \\ -0.4297 & 0.9030 \end{bmatrix} \times \begin{bmatrix} 12351 & 0 \\ 0 & 65 \end{bmatrix} \times \begin{bmatrix} -0.9030 & -0.4297 \\ -0.4297 & 0.9030 \end{bmatrix}$$



PCA – Principal Component Analysis

- $\Sigma^2 = diag(\lambda_i^2), i = 1,\ldots,\min(n,m)$

Samples

Variables observed

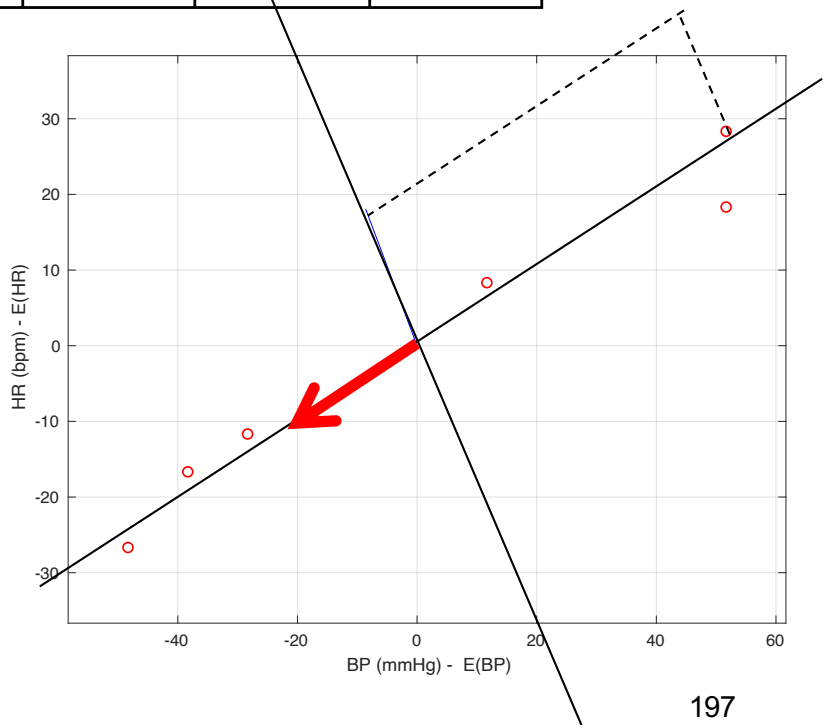
	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)-E(BP)	51,7	51,7	11,7	-28,3	-38,3	-48,3
HR (bpm)-E(HR)	28,3	18,3	8,3	-11,7	-16,7	-26,7

$$d = Pw$$

$$= \begin{bmatrix} (28.3;51.7) \\ (51.7;18.3) \\ (11.7;8.3) \\ (-28.3;-11.7) \\ (-38.3;-16.7) \\ (-48.3;-26.7) \end{bmatrix} \times \begin{bmatrix} -0.9030 \\ -0.4297 \end{bmatrix}$$

$$= \begin{bmatrix} -58.83 \\ -54.53 \\ -14.12 \\ 30.60 \\ 41.78 \\ 55.10 \end{bmatrix}$$

$$\sum_{k=1}^{6} d_k^2 = 12351 = \lambda_1$$



PCA – Principal Component Analysis

- **Interpretation of Singular Values**
  - $\lambda_1 > \lambda_2 = \lambda_3 > 0$
  - $\lambda_1 = \lambda_2 = \lambda_3 > 0$
  - $\lambda_1 = \lambda_2 > \lambda_3 = 0$
  - $\lambda_1 > \lambda_2 = \lambda_3 = 0$

The larger λ, *the larger the dispersion in that direction*

PCA – Principal Component Analysis

- $\Sigma^2 = diag(\lambda_i^2), i = 1,\ldots,\min(n,m)$

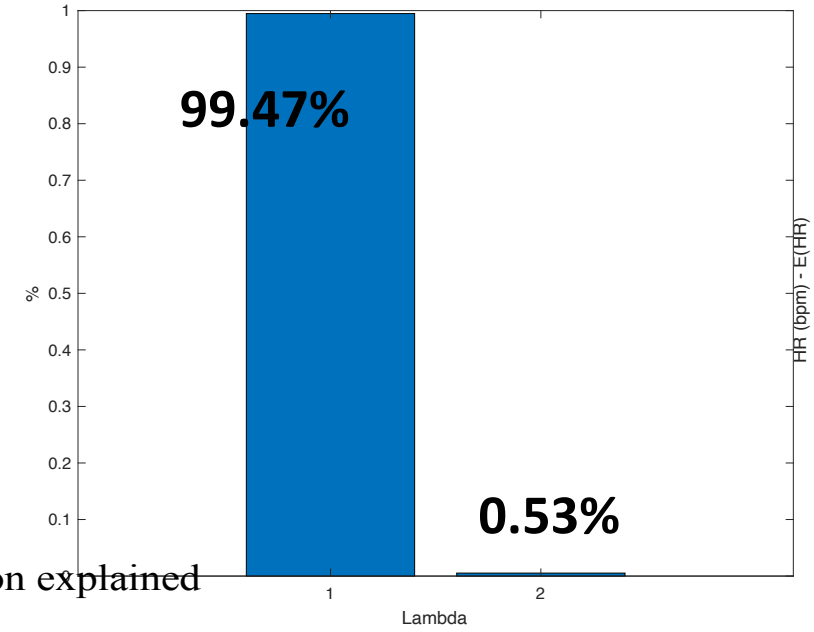
Samples

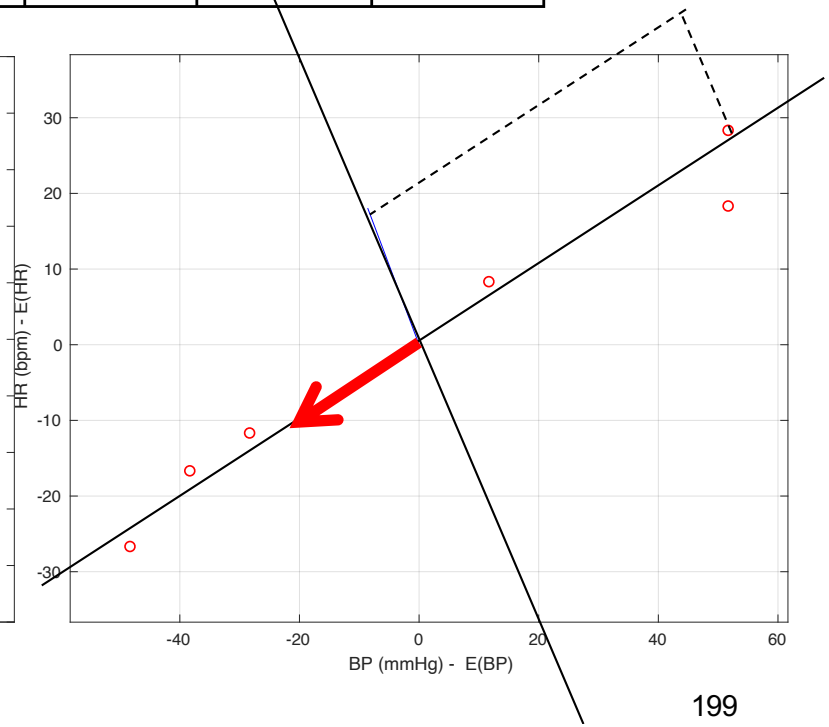
Variables observed

	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5	Patient 6
BP (mmHg)-E(BP)	51,7	51,7	11,7	-28,3	-38,3	-48,3
HR (bpm)-E(HR)	28,3	18,3	8,3	-11,7	-16,7	-26,7

$$\sum_{k=1}^{6} d_{k,i}^2 = \lambda_i$$

$$\lambda = \begin{bmatrix} 12351 \\ 65 \end{bmatrix}$$

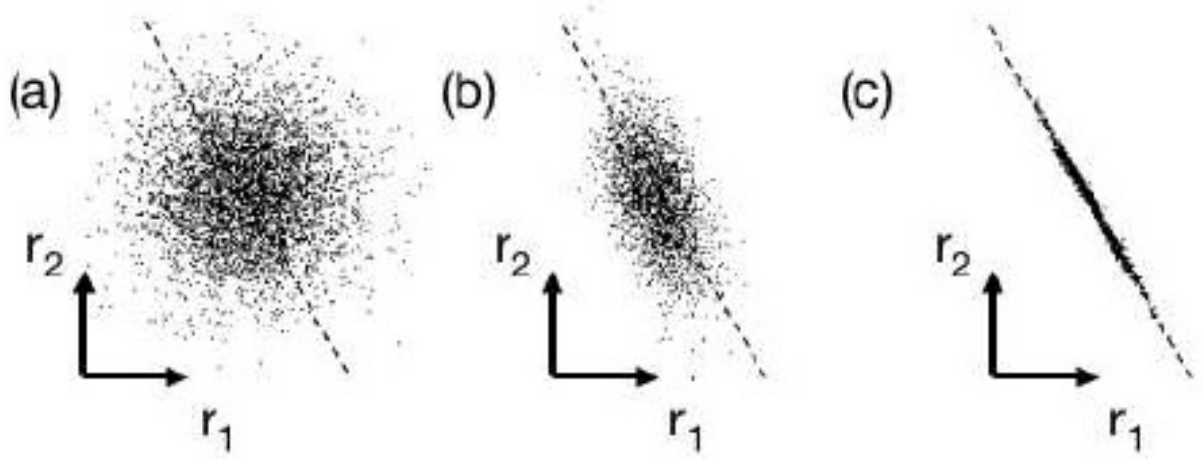




$$\beta_i = \frac{\lambda_i}{\sum_k \lambda_i} \rightarrow \text{variation explained}$$

PCA – Principal Component Analysis

- Dimensionality reduction
  - Eliminate redundant dimensions



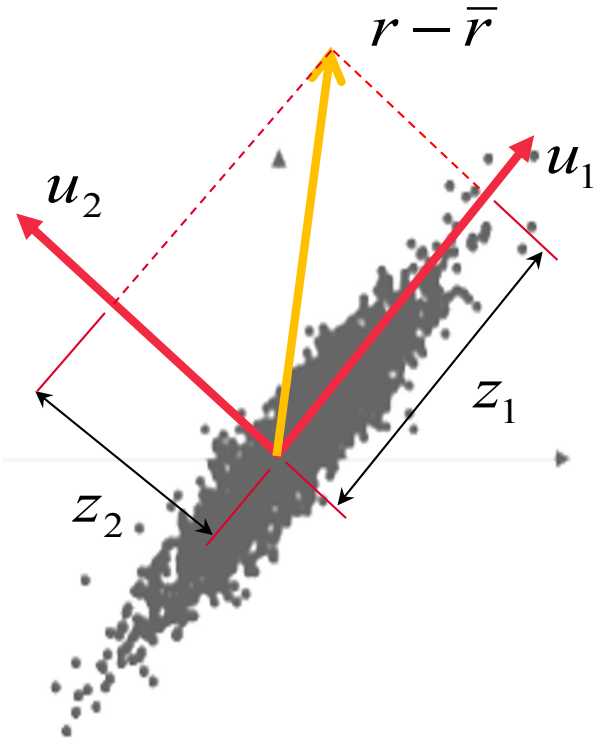
Transformada Karuhnen-Loéve

- Dimensionality reduction

$$\tilde{U} = \begin{bmatrix} u_1 & u_2 & \cdots & u_{\tilde{p}} \end{bmatrix}$$

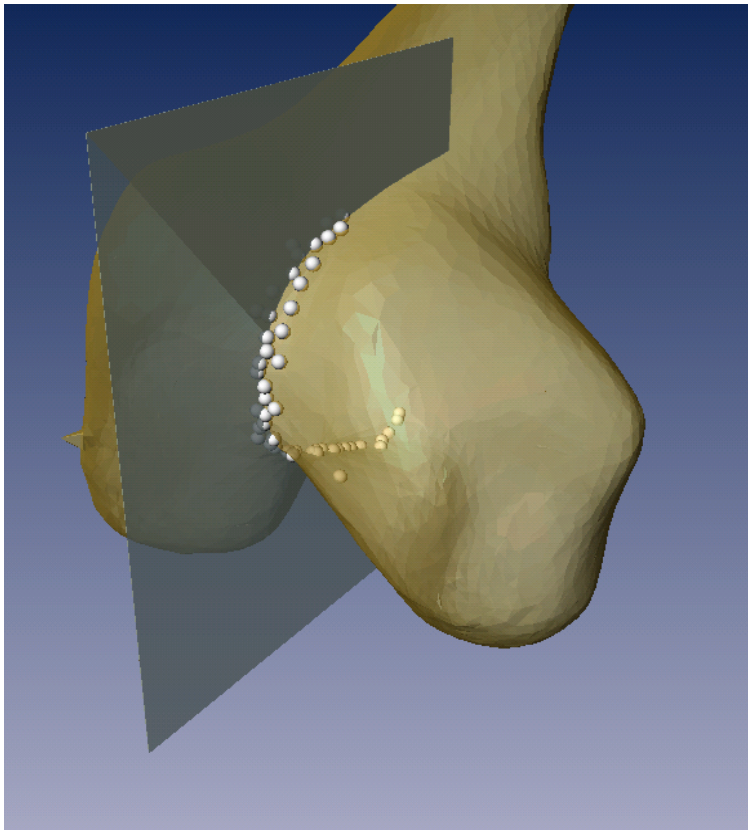
$$Z = \tilde{U}^t (r - \bar{r}) = \begin{bmatrix} u_1^T \\ \\ u_{\tilde{p}}^T \end{bmatrix} (r - \bar{r})$$

$$\tilde{x} = \sum_{i=1}^{\tilde{p}} z_i u_i$$



PCA – Principal Component Analysis

- Interpretability -> abstract



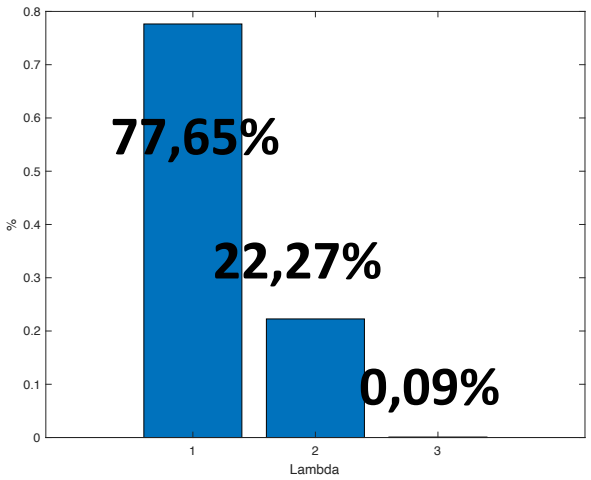
PCA – Principal Component Analysis

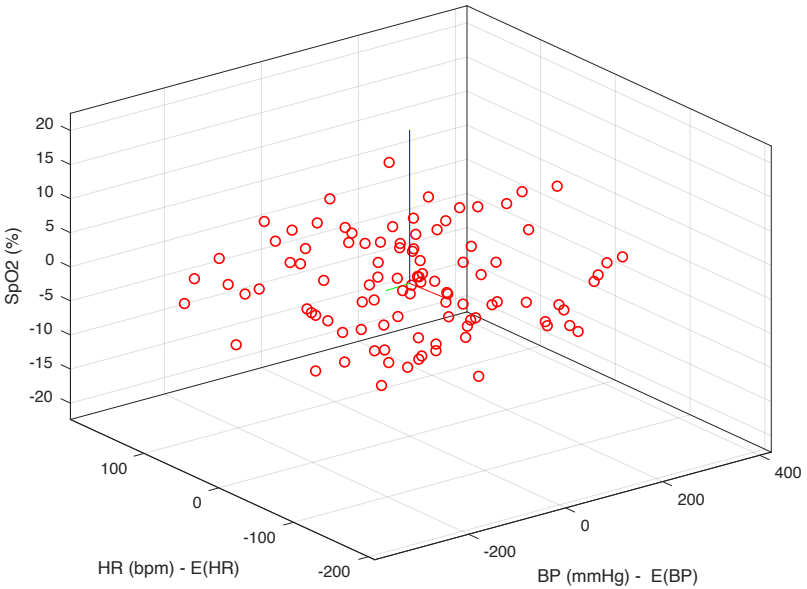
- Feature Learning
  - Reduce dimensionality until reconstruction % is achieved
  - Project input vectors onto selected PCs

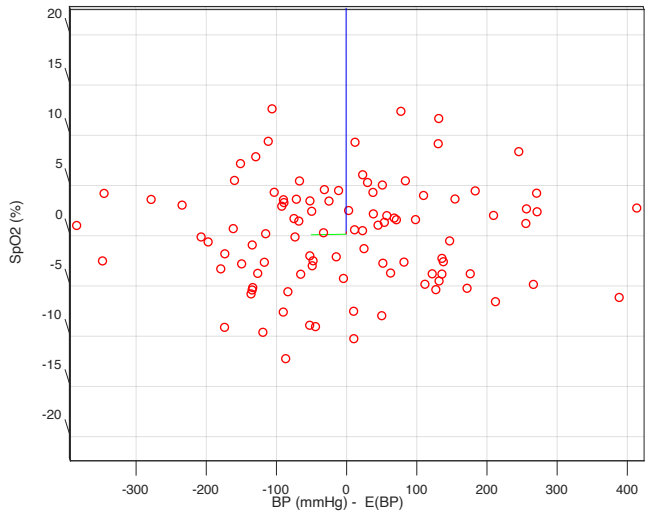
PCA – the silver bullet?

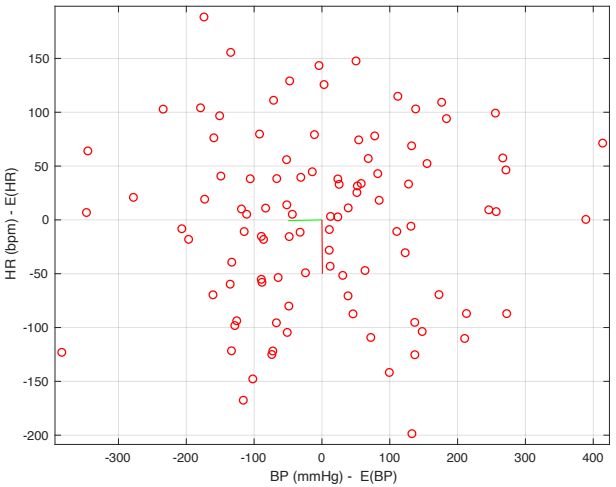
- Lack of Scale invariance
- Orthogonality
- Nonlinear Spaces
- Supervised learning

PCA – Scale Invariance

- 



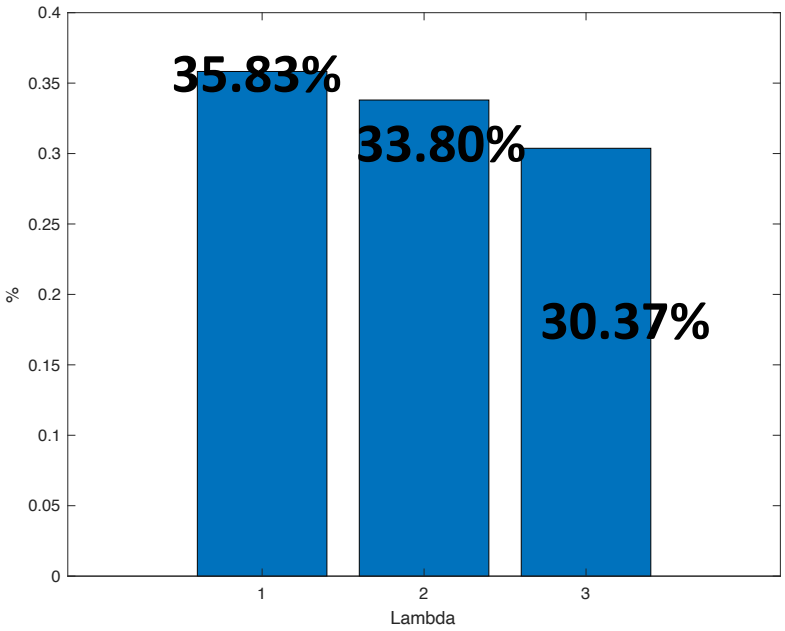


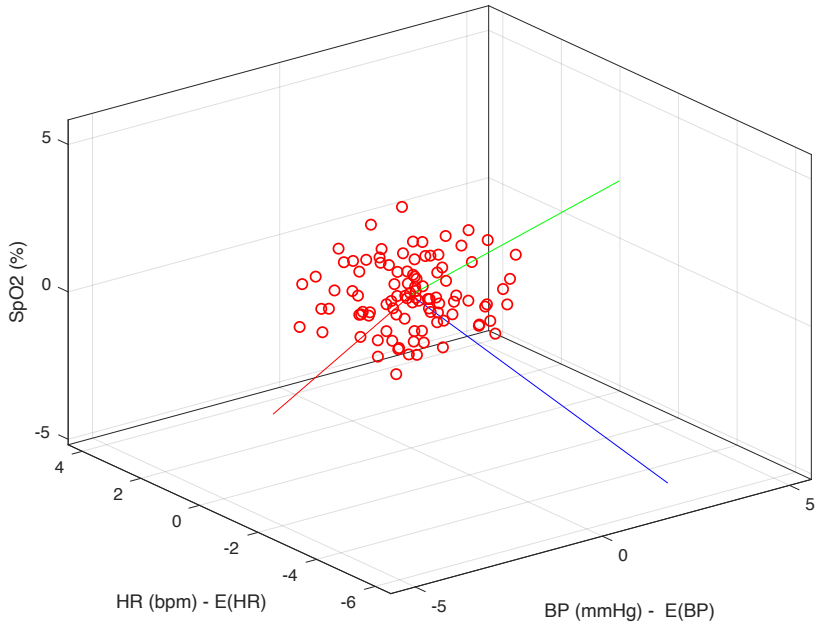


There is a lot of variance in the SP02! WHY is Lambda so low?

PCA – Scale Invariance

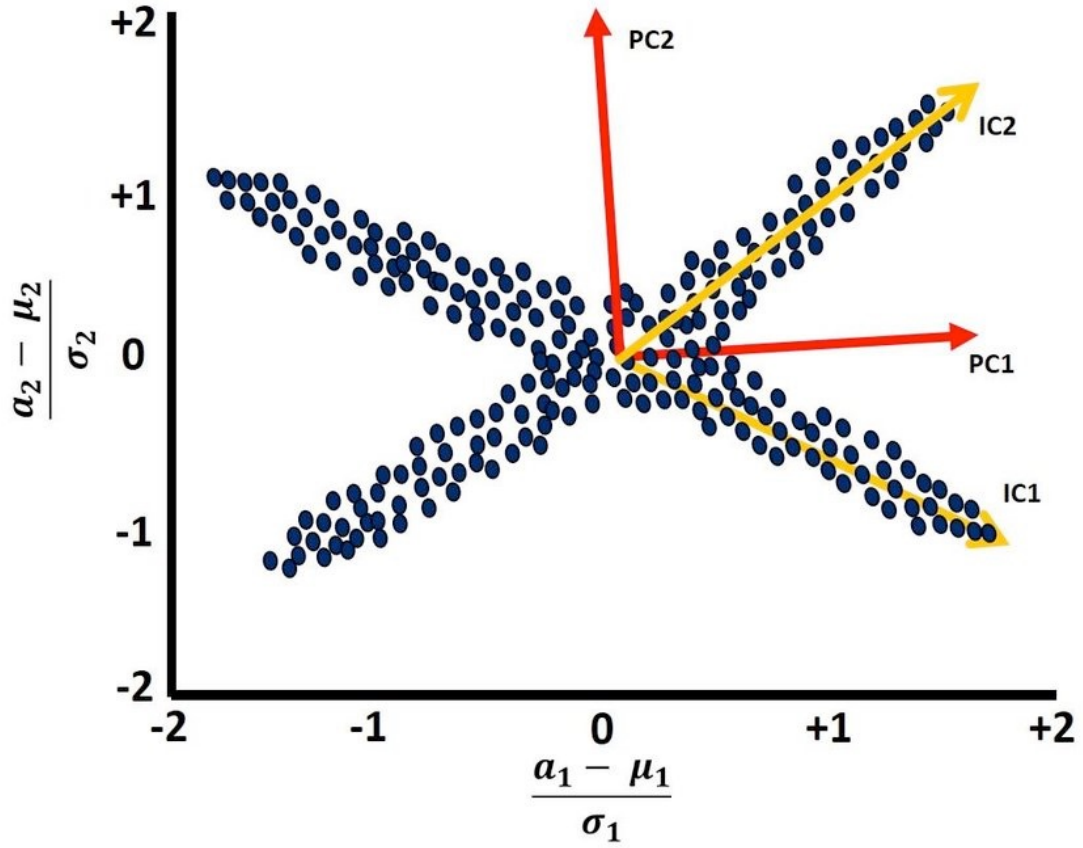
- **Scale invariance -> Normalize the data: e.g. Z-Score**





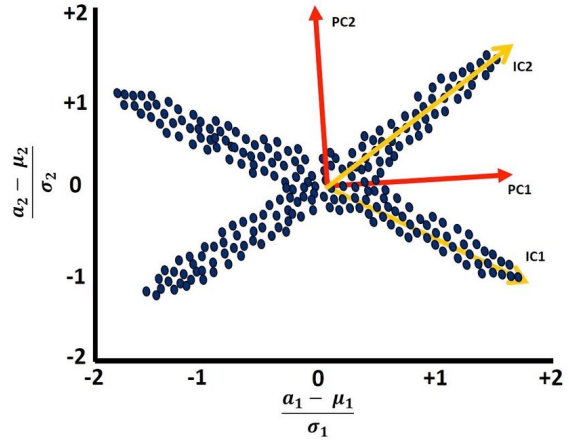
PCA - Orthogonality

- What if the main dimensions are not orthogonal?



PCA - Orthogonality

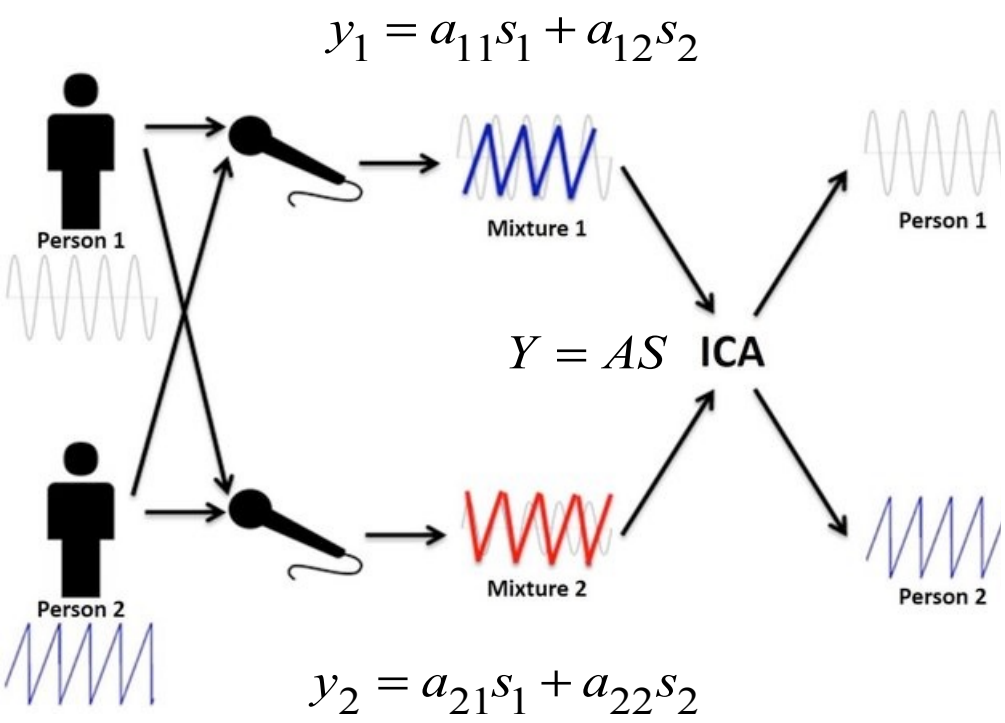
- What if the main dimensions are not orthogonal?
- Maximum variation occurs in abstract direction
- Solution ?
  - Independent Component Analysis
  - Assumption: different sources are independent



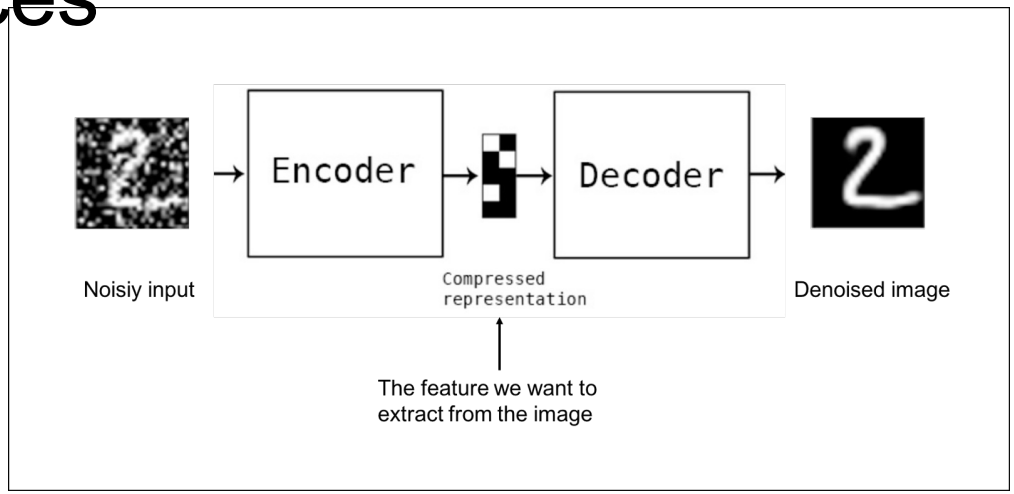
PCA - Orthogonality

Independent Component Analysis

- Let v1, v2, v3, … vd denote the projection directions of independent components
- ICA: find these directions such that data projected onto these directions have maximum statistical independence
- How to actually maximize independence?
  - Minimize the mutual information
  - Or maximize the non-Gaussianity
  - Actual formulation quite complicated !
  - Popular algo. FastICA



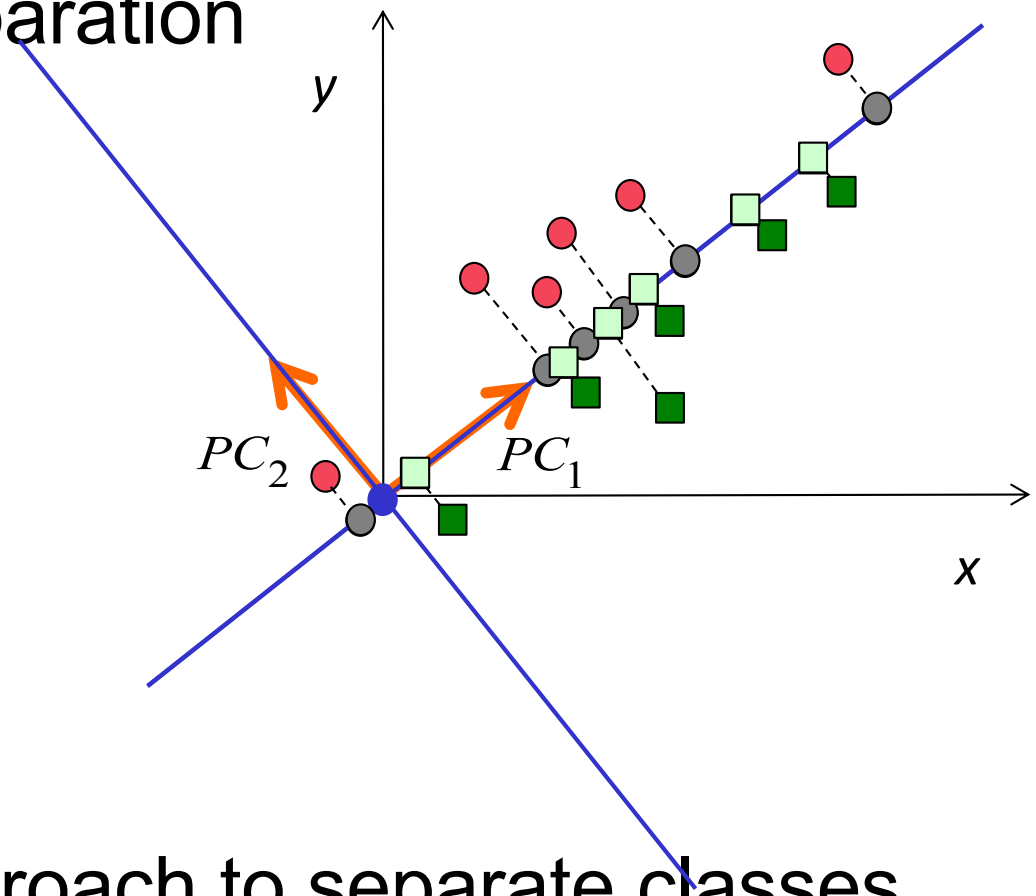
PCA – non-linear spaces



Non-linear PCA and AutoEncoders -> latter in the curriculum

PCA – supervised learning

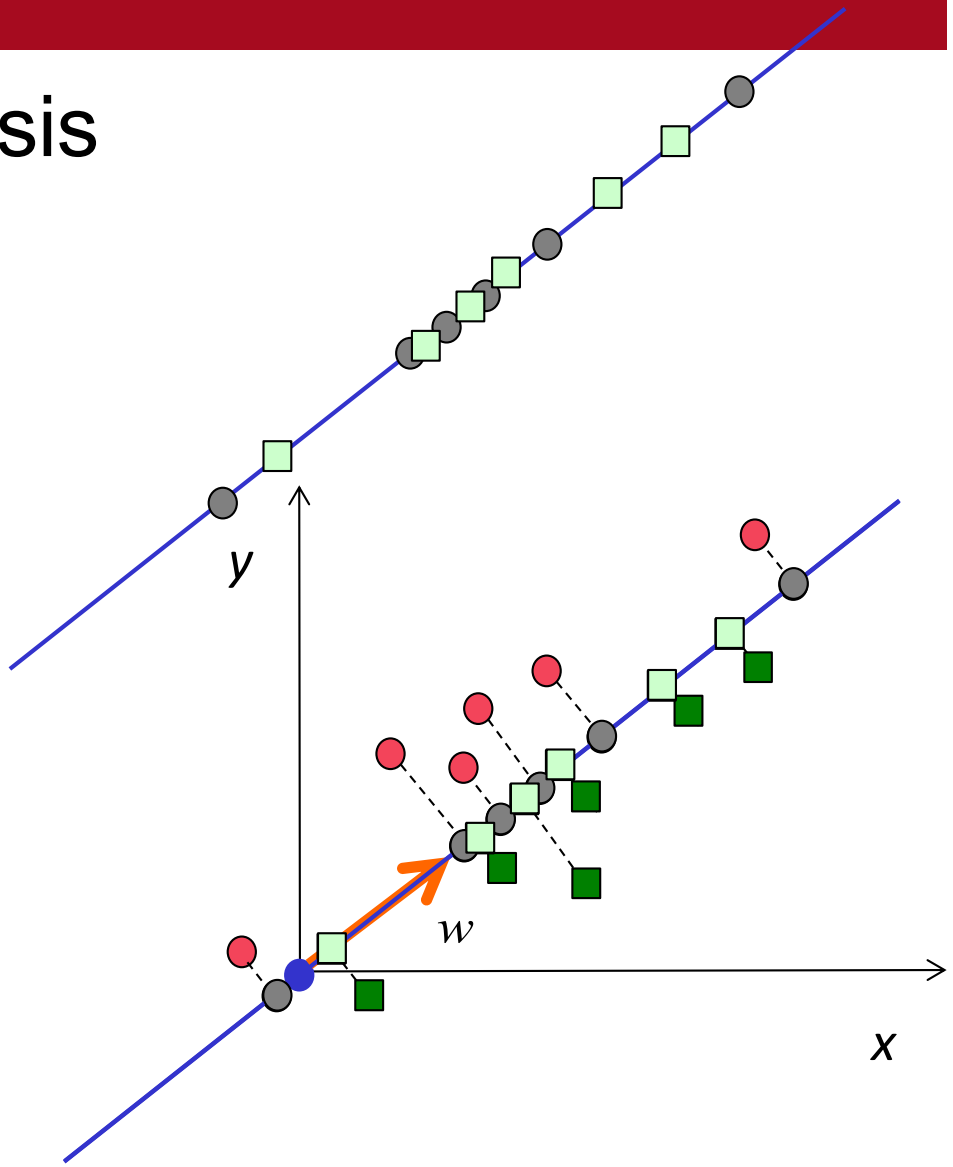
- PC1 – not a good separation



- We need a better approach to separate classes
  - Take advantage of class labels

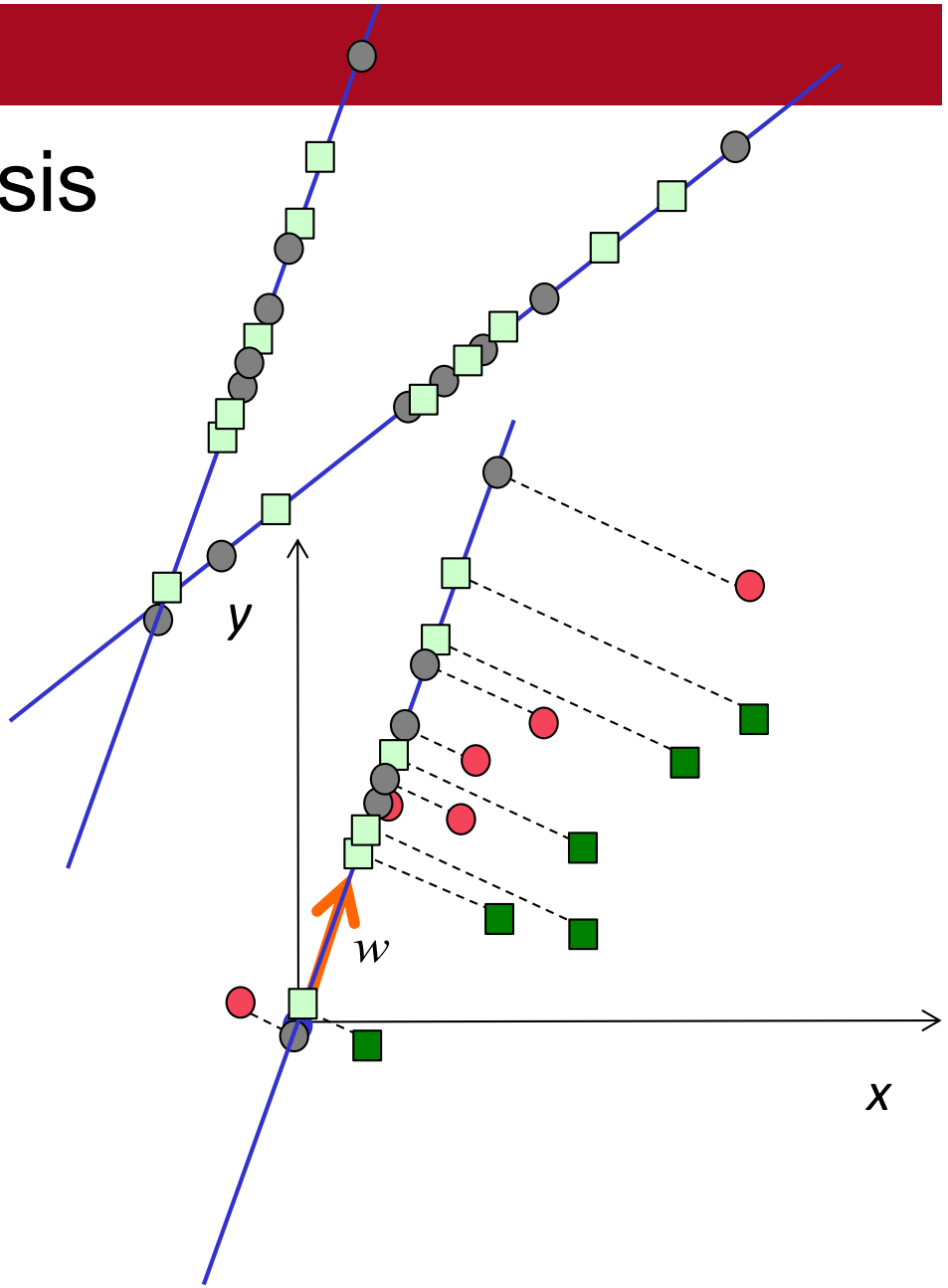
Fisher Discriminant Analysis

- Supervised technique
- Applicable for classification tasks
- **Idea: detect direction that maximizes the separation between classes**



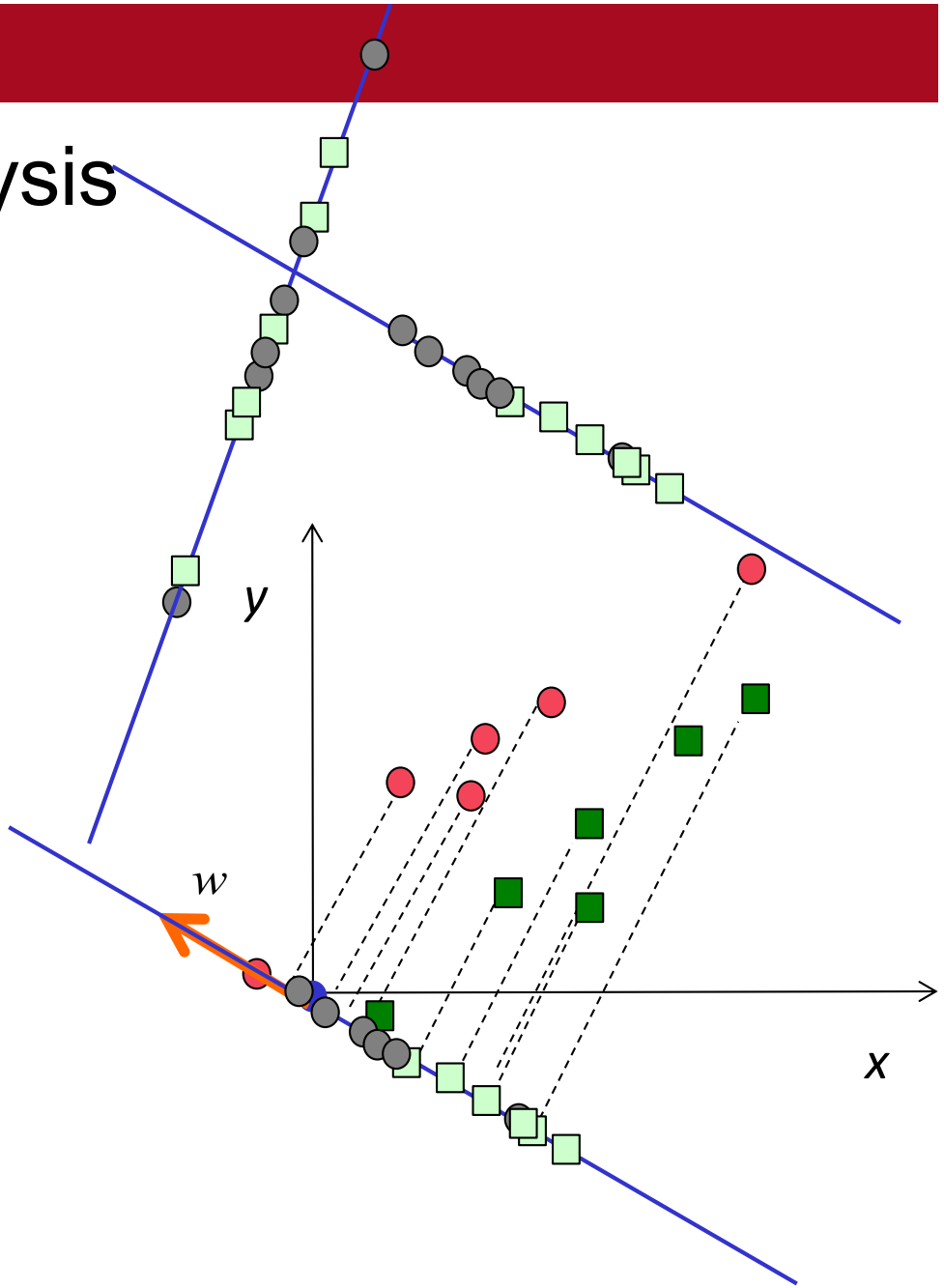
Fisher Discriminant Analysis

- Supervised technique
- Applicable for classification tasks
- Idea: detect direction that maximizes the separation between classes



Fisher Discriminant Analysis

- Supervised technique
- Applicable for classification tasks
- Idea: detect direction that maximizes the separation between classes



Fisher Discriminant Analysis

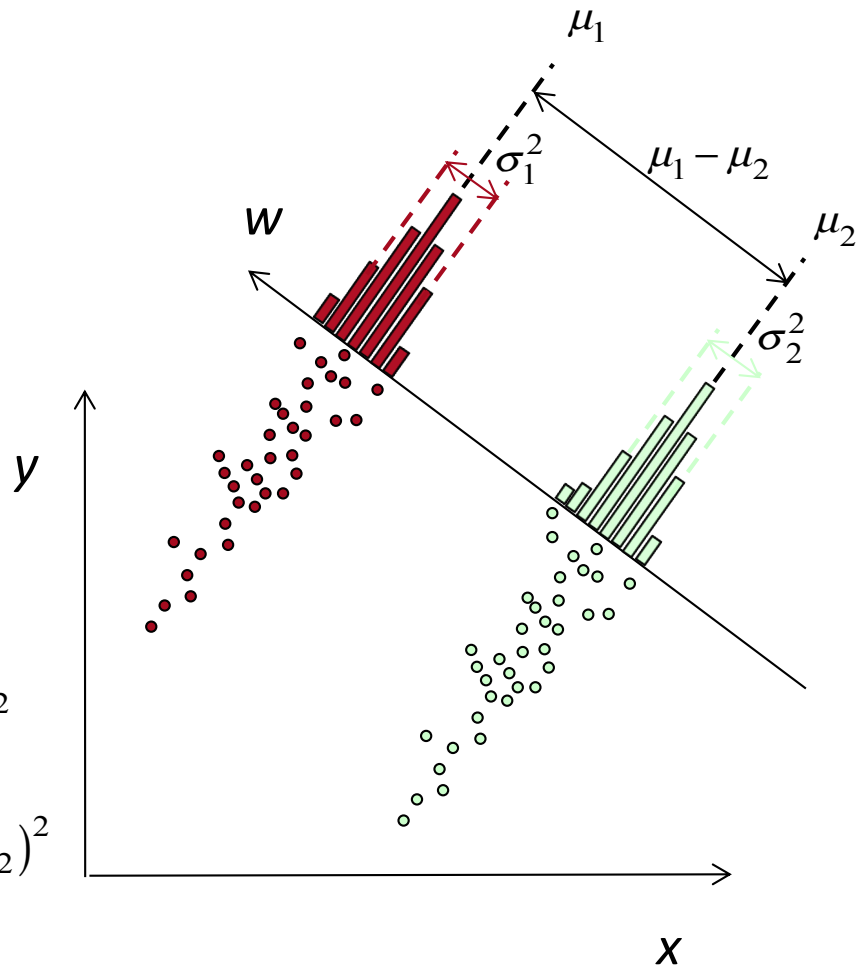
- PCA and LDA
  - rank dimensions
    - PC1 – best variation direction
    - PC2 – second best variation direction
    - …
    - LDA1 – best separation between the categories
    - LDA2 – second best separation between the categories
    - ...
  - Both let you dig in and see which original variables are driving/correlate with the new axis

Fisher Discriminant Analysis

- FDA
  - Maximum separation between means of projected classes
  - Minimum variance within each projected class

$$\mu_1 = \frac{1}{n_1}\sum_{i\in C_1} p_i; n_1 = \#C_1 \quad \sigma_1^2 = \frac{1}{n_1 - 1}\sum_{i\in C_1}\left(p_i - \mu_1\right)^2$$

$$\mu_2 = \frac{1}{n_2}\sum_{i\in C_2} p_i; n_2 = \#C_2 \quad \sigma_2^2 = \frac{1}{n_2 - 1}\sum_{i\in C_2}\left(p_i - \mu_2\right)^2$$

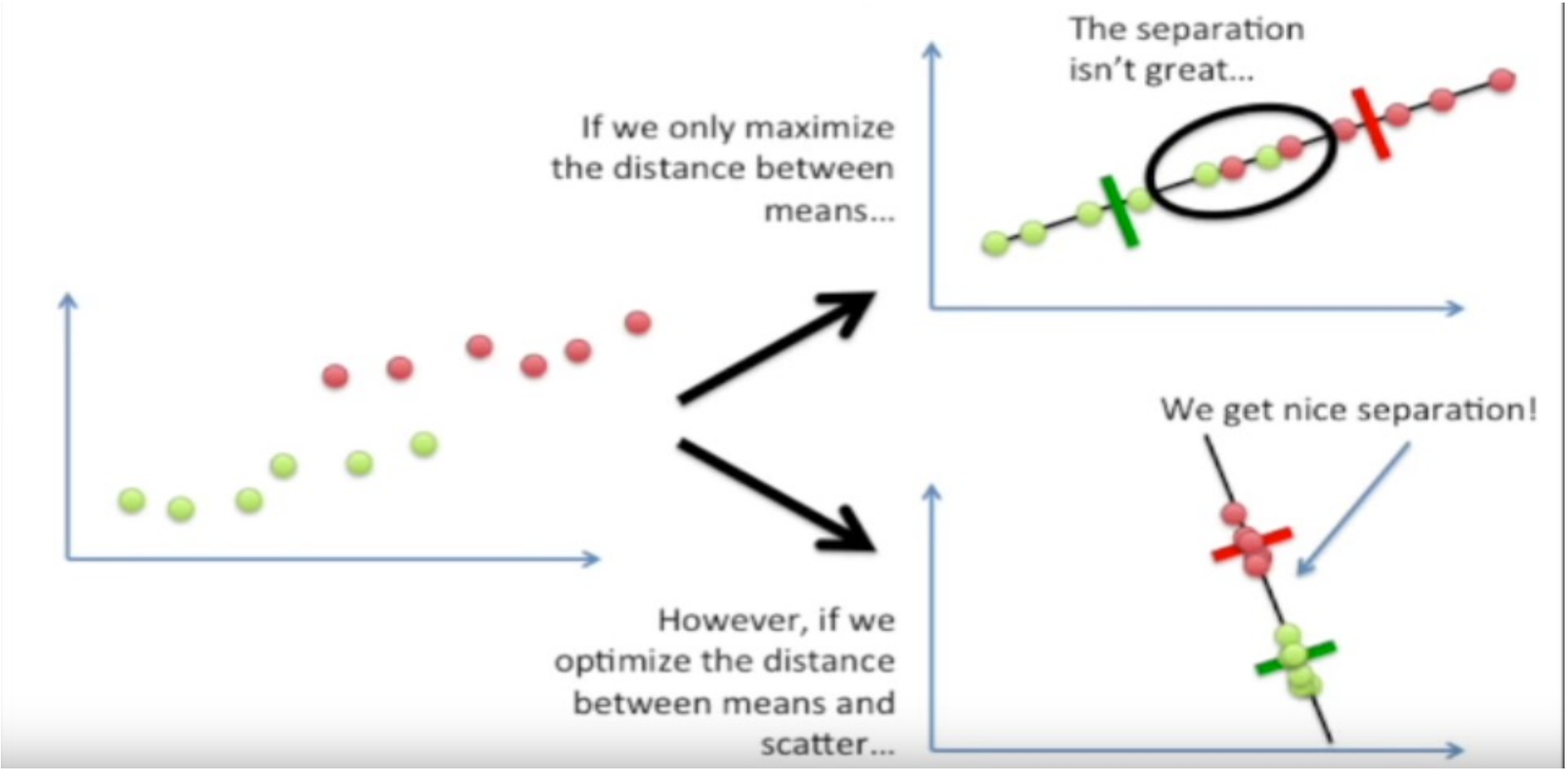


$$\max\left\{\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}\right\}$$

Invariance to "negative" separation

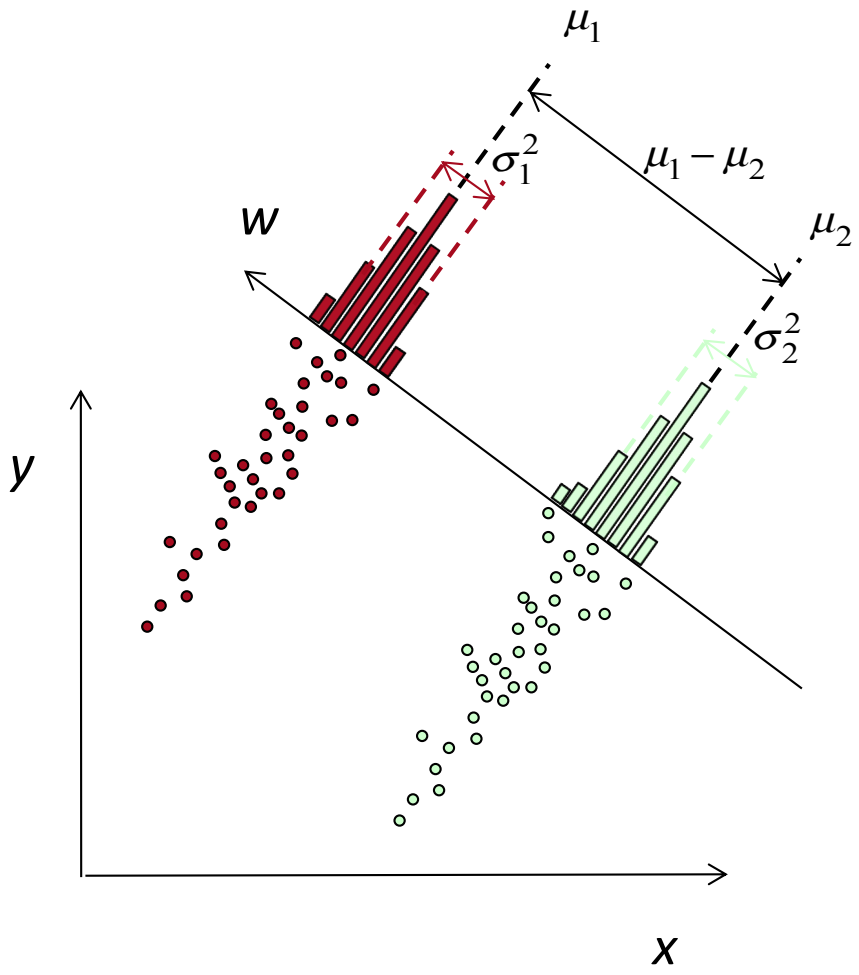
Fisher Discriminant Analysis

- Why do we need two criteria?



Fisher Discriminant Analysis

- What is the best hyper-plane that:
  - Maximum separation between means of projected classes
  - Minimum variance within each projected class



$$y = w_0 + w^T p$$

$$\mu_1 = \frac{1}{n_1}\sum_{i \in C_1} p_i; n_1 = \#C_1 \quad \Sigma_1 = \frac{1}{n_1}\sum_{i \in C_1}(p_i - \mu_1)(p_i - \mu_1)^T$$

$$\mu_2 = \frac{1}{n_2}\sum_{i \in C_2} p_i; n_2 = \#C_2 \quad \Sigma_2 = \frac{1}{n_2}\sum_{i \in C2}(p_i - \mu_2)(p_i - \mu_2)^T$$

$$E\left[\hat{y} \mid p \in C_i\right] = w_0 + w^T \mu_i, i = 1,2$$

$$\text{var}\left[\hat{y} \mid p \in C_i\right] = w^T \Sigma_i w, i = 1,2$$

Fisher Discriminant Analysis

- Best hyper-plane:

$$\underset{w}{\text{argmax}} = \left(\frac{\left(\mu_1^T w - \mu_2^T w\right)^2}{w^T \Sigma_1 w + w^T \Sigma_2 w} \right)$$

$$\mu_1 = \frac{1}{n_1} \sum_{i \in C_1} p_i ; n_1 = \#C_1 \quad \Sigma_1 = \frac{1}{n_1 - 1} \sum_{i \in C_1} \left(p_i - \mu_1\right)\left(p_i - \mu_1\right)^T$$

$$\mu_2 = \frac{1}{n_2} \sum_{i \in C_2} p_i ; n_2 = \#C_2 \quad \Sigma_2 = \frac{1}{n_2 - 1} \sum_{i \in C2} \left(p_i - \mu_2\right)\left(p_i - \mu_2\right)^T$$

- Cost function is not linear -> gradient is complex
- We shall use a trick:

$$\underset{w}{\text{argmax}} \left(\frac{\left(\mu_1^T w - \mu_2^T w\right)^2}{w^T \Sigma_1 w + w^T \Sigma_2 w} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\frac{\left(\overbrace{\left(\mu_1^T - \mu_2^T\right)}^{m^T} w \right)^2}{w^T \underbrace{\left(\Sigma_1 + \Sigma_2\right)}_{S} w} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\frac{\left(m^T w\right)^2}{w^T S w} \right)$$

Fisher Discriminant Analysis

- Use the Square-root matrices:
  - Note: S is symmetric

$$\because S = R^T R$$

$$\underset{w}{\text{argmax}} \left(\frac{\left(m^T w\right)^2}{w^T S w} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\frac{\left(m^T w\right)^2}{w^T R^T R w} \right)$$

  - Let:

Normalized vector

$$\because v \equiv Rw, \quad w = R^{-1} v$$

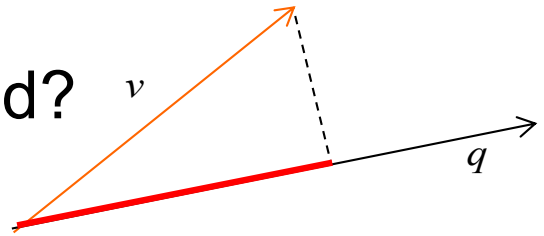
$$\underset{w}{\text{argmax}} \left(\frac{\left(m^T w\right)^2}{w^T R^T R w} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\frac{\left(m^T R^{-1} v\right)^2}{v^T \left(R^{-1}\right)^T R^T R R^{-1} v} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\frac{\left(m^T R^{-1} v\right)^2}{\underbrace{v^T v}_{\|v\|^2}} \right) \Leftrightarrow \underset{w}{\text{argmax}} \left(\left(m^T R^{-1} \frac{v}{|v|} \right)^2 \right)$$

Fisher Discriminant Analysis

- Solution:

$$\underset{w}{\text{argmax}}\left(\left(m^T R^{-1}\frac{v}{|v|}\right)^2\right) \Leftrightarrow \underset{w}{\text{argmax}}\left(\left(\underbrace{\left(\left(R^{-1}\right)^T m\right)}_{q}{}^T \frac{v}{|v|}\right)^2\right)$$

  - When is the maximum reached?



$$v = a\left(R^{-1}\right)^T m, \quad a \in \Re^+$$

Fisher Discriminant Analysis

- Solution

$$v = a\left(R^{-1}\right)^T m = a\left(R^{-1}\right)^T \left(\mu_1 - \mu_2\right)$$

$$\because w = R^{-1}v = aR^{-1}\left(R^{-1}\right)^T \left(\mu_1 - \mu_2\right) = a\left(R^T R\right)^{-1}\left(\mu_1 - \mu_2\right)$$

$$w = aS^{-1}\left(\mu_1 - \mu_2\right) = a\left(\Sigma_1 + \Sigma_2\right)^{-1}\left(\mu_1 - \mu_2\right)$$

Rotation Matrix

- w_0:

$$w_0 = E\left[y - w^T p\right] = \frac{1}{n}\sum y_k - w^T p_k$$

Linear Discriminant Analysis

- LDA assumptions:
  - homoscedasticity $\Sigma = \Sigma_1 = \Sigma_2$

$$w = a\Sigma^{-1}\left(\mu_1 - \mu_2\right) \propto \Sigma^{-1}\left(\mu_1 - \mu_2\right)$$

  - More restrictive then FDA

Linear Discriminant Analysis

- General solution:

$$\underset{w}{\operatorname{argmax}}\left(\frac{\left(\left(\mu_1^T-\mu_2^T\right)w\right)^2}{w^T S w}\right) \Leftrightarrow \underset{w}{\operatorname{argmax}}\left(\frac{w^T \overbrace{\left(\mu_1-\mu_2\right)\left(\mu_1-\mu_2\right)^T}^{S_b} w}{w^T S w}\right) \Leftrightarrow \underset{w}{\operatorname{argmax}}\left(\frac{w^T S_b w}{w^T S w}\right)$$

- Solution:
  - Let $A \equiv S^{-1} S_b$
  - w – eigen vectors of A
  - Ranking of w, based on eigen-values

Fisher Discriminant Analysis

- What if we have C>2 classes?

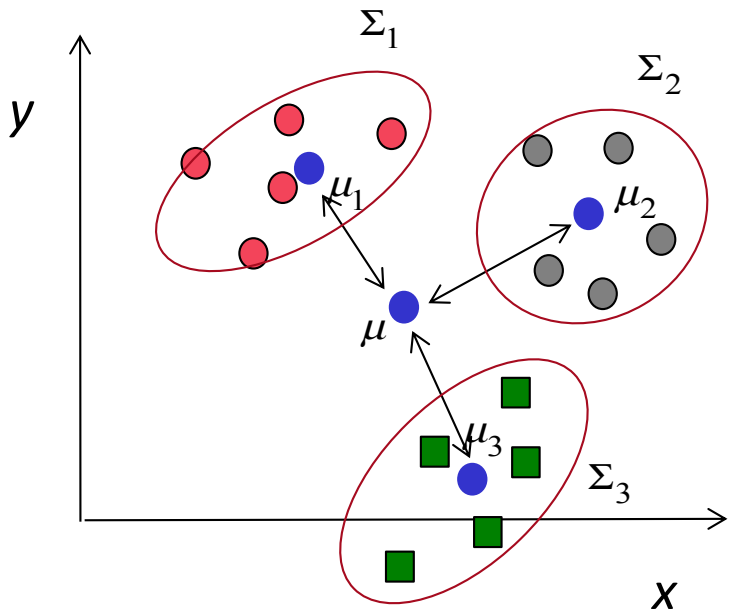
$$\underset{w}{\text{argmax}}\left(\frac{w^T S_b w}{w^T S w}\right)$$

$$\mu = \frac{1}{n}\sum_{i\in C} p_i, \quad n = \sum_{i\in C} \#C_i$$

$$\mu_k = \frac{1}{n_k}\sum_{i\in C_k} p_i; n_k = \#C_k$$

$$S_b = \frac{1}{C}\sum_{i=1}^{C}(\mu_i - \mu)(\mu_i - \mu)^T$$

Between class distance



$$\Sigma_k = \frac{1}{n_k}\sum_{i\in C_k}(p_i - \mu_k)(p_i - \mu_k)^T, \qquad k = 1,\ldots,C$$

$$S = \sum_{i=1}^{C} \Sigma_i$$

Within-class Scatter

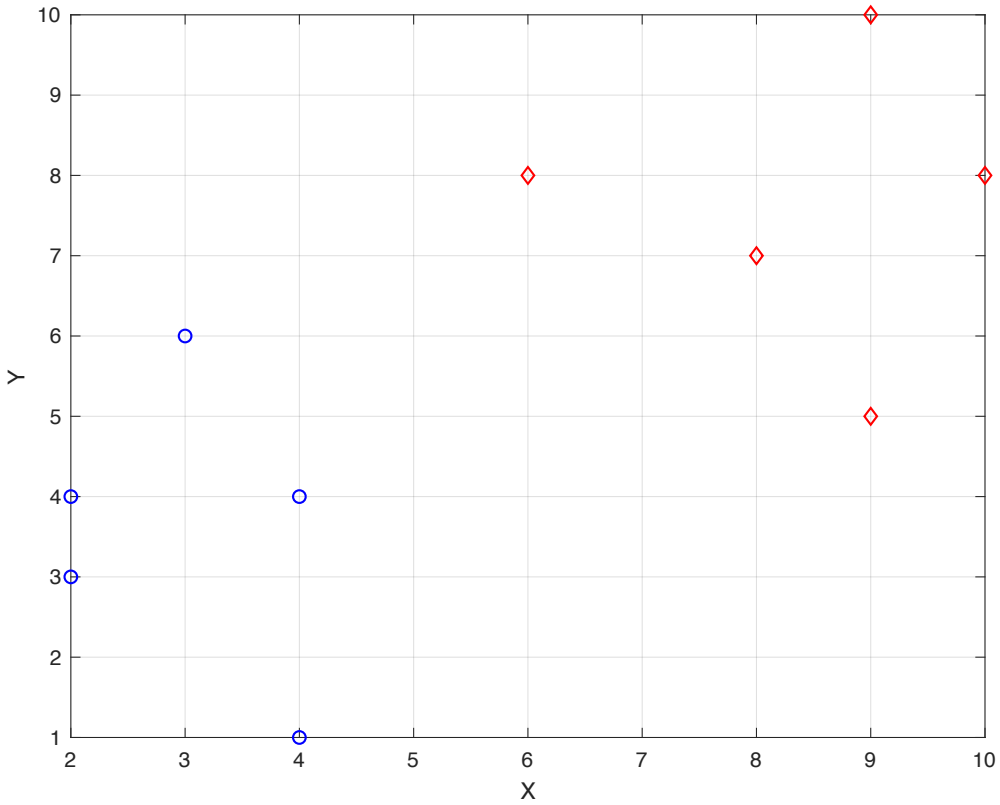
$$A \equiv S^{-1} S_b$$

LDA

- Example

$$C_1 = \begin{bmatrix} 4 & 2 & 2 & 3 & 4 \\ 1 & 4 & 3 & 6 & 4 \end{bmatrix}$$

$$C_2 = \begin{bmatrix} 9 & 6 & 9 & 8 & 10 \\ 10 & 8 & 5 & 7 & 8 \end{bmatrix}$$



LDA

- Step1: Within-Class Scatter Matrix

$$\Sigma_k = \frac{1}{n_k} \sum_{i \in C_k} (p_i - \mu_k)(p_i - \mu_k)^T, \qquad k = 1,...,C$$

$$\mu_k = \frac{1}{n_k} \sum_{i \in C_k} p_i ; n_k = \#C_k$$

$$(p_i - \mu_k)$$

$$\mu_1 = \frac{1}{5}\left(\begin{bmatrix} 4 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 \\ 4 \end{bmatrix} + \begin{bmatrix} 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 3 \\ 6 \end{bmatrix} + \begin{bmatrix} 4 \\ 4 \end{bmatrix}\right) = \begin{bmatrix} 3 \\ 3.6 \end{bmatrix}$$

$$\mu_2 = \frac{1}{5}\left(\begin{bmatrix} 9 \\ 10 \end{bmatrix} + \begin{bmatrix} 6 \\ 8 \end{bmatrix} + \begin{bmatrix} 9 \\ 5 \end{bmatrix} + \begin{bmatrix} 8 \\ 7 \end{bmatrix} + \begin{bmatrix} 10 \\ 8 \end{bmatrix}\right) = \begin{bmatrix} 8.4 \\ 7.6 \end{bmatrix}$$

$$p_i - \mu_1 = \left\{\begin{bmatrix} 1 \\ -2.6 \end{bmatrix}, \begin{bmatrix} -1 \\ 0.4 \end{bmatrix}, \begin{bmatrix} -1 \\ -0.6 \end{bmatrix}, \begin{bmatrix} 0 \\ 2.4 \end{bmatrix}, \begin{bmatrix} 1 \\ 0.4 \end{bmatrix}\right\}$$

$$p_i - \mu_2 = \left\{\begin{bmatrix} 0.6 \\ 2.4 \end{bmatrix}, \begin{bmatrix} -2.4 \\ 0.4 \end{bmatrix}, \begin{bmatrix} 0.6 \\ -2.6 \end{bmatrix}, \begin{bmatrix} -0.4 \\ -0.6 \end{bmatrix}, \begin{bmatrix} 1.6 \\ 0.4 \end{bmatrix}\right\}$$

LDA

$$p_i - \mu_1 = \left\{ \begin{bmatrix} 1 \\ -2.6 \end{bmatrix}, \begin{bmatrix} -1 \\ 0.4 \end{bmatrix}, \begin{bmatrix} -1 \\ -0.6 \end{bmatrix}, \begin{bmatrix} 0 \\ 2.4 \end{bmatrix}, \begin{bmatrix} 1 \\ 0.4 \end{bmatrix} \right\}$$

- Step1: Within-Class Scatter Matrix

$$\Sigma_k = \frac{1}{n_k} \sum_{i \in C_k} (p_i - \mu_k)(p_i - \mu_k)^T, \qquad k = 1,\ldots,C$$

$$(p_i - \mu_1)(p_i - \mu_1)^T \longrightarrow (p_1 - \mu_1)(p_1 - \mu_1)^T = \begin{bmatrix} 1 \\ -2.6 \end{bmatrix} \begin{bmatrix} 1 & -2.6 \end{bmatrix} = \begin{bmatrix} 1 & -2.6 \\ -2.6 & 6.76 \end{bmatrix}$$

$$(p_2 - \mu_1)(p_2 - \mu_1)^T = \begin{bmatrix} -1 \\ 0.4 \end{bmatrix} \begin{bmatrix} -1 & 0.4 \end{bmatrix} = \begin{bmatrix} 1 & -0.4 \\ -0.4 & 1.16 \end{bmatrix}$$

$$(p_3 - \mu_1)(p_3 - \mu_1)^T = \begin{bmatrix} -1 \\ -0.6 \end{bmatrix} \begin{bmatrix} -1 & -0.6 \end{bmatrix} = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 0.36 \end{bmatrix}$$

$$\Sigma_1 = \begin{bmatrix} 0.8 & -0.4 \\ -0.4 & 2.64 \end{bmatrix}$$

$$(p_4 - \mu_1)(p_4 - \mu_1)^T = \begin{bmatrix} 0 \\ 2.4 \end{bmatrix} \begin{bmatrix} 0 & 2.4 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 5.76 \end{bmatrix}$$

$$(p_5 - \mu_1)(p_5 - \mu_1)^T = \begin{bmatrix} 1 \\ 0.4 \end{bmatrix} \begin{bmatrix} 1 & 0.4 \end{bmatrix} = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 0.16 \end{bmatrix}$$

LDA

- Step1: Within-Class Scatter Matrix

$$\Sigma_k = \frac{1}{n_k - 1} \sum_{i \in C_k} (p_i - \mu_k)(p_i - \mu_k)^T, \qquad k = 1,...,C$$

$$\Sigma_1 = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 3.3 \end{bmatrix} \qquad \Sigma_2 = \begin{bmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{bmatrix}$$

$$\Sigma = \Sigma_1 + \Sigma_2 = \begin{bmatrix} 3.3 & -0.55 \\ -0.55 & 6.6 \end{bmatrix}$$

LDA

- Step2: Between-Class Matrix

$$\underset{w}{\text{argmax}}\left(\frac{w^T\overbrace{\left(\mu_1-\mu_2\right)\left(\mu_1-\mu_2\right)^T}^{S_b}w}{w^TSw}\right)\Leftrightarrow\underset{w}{\text{argmax}}\left(\frac{w^TS_bw}{w^TSw}\right)$$

$$S_b=\left(\mu_1-\mu_2\right)\left(\mu_1-\mu_2\right)^T=\begin{bmatrix}-5.4\\-4\end{bmatrix}\begin{bmatrix}-5.4 & -4\end{bmatrix}=\begin{bmatrix}29.16 & 21.6\\21.6 & 16.00\end{bmatrix}$$

LDA

- Step3: find best FDA directions
  - Solution 1: Eigen-values/eigen-vectors

$$S_b S^{-1} w = \lambda w$$

$$\left| S_b S^{-1} - \lambda I \right| = 0 \Leftrightarrow \begin{vmatrix} 9.5140 - \lambda & 7.0474 \\ 4.0656 & 3.0115 - \lambda \end{vmatrix} = 0$$

$$\lambda = 12.5255$$

$$w = \begin{bmatrix} 0.9196 \\ 0.3930 \end{bmatrix}$$

LDA

- Step3: find best FDA directions
  - Solution 2:

$$w = aS^{-1}\left(\mu_1 - \mu_2\right) \propto S^{-1}\left(\mu_1 - \mu_2\right) = \begin{bmatrix} 0.3841 & 0.0320 \\ 0.0320 & 0.1921 \end{bmatrix} \begin{bmatrix} -5.4 \\ -4 \end{bmatrix} = \begin{bmatrix} -2.2023 \\ -0.9411 \end{bmatrix}$$

  - Same direction as

$$w = \begin{bmatrix} 0.9196 \\ 0.3930 \end{bmatrix}$$

Engenharia de Características para Aprendizagem Computacional /

Engenharia de Atributos

Feature Engineering: Feature Selection

Feature Selection

- Feature Selection:
  - select a subset of features without any transformation.

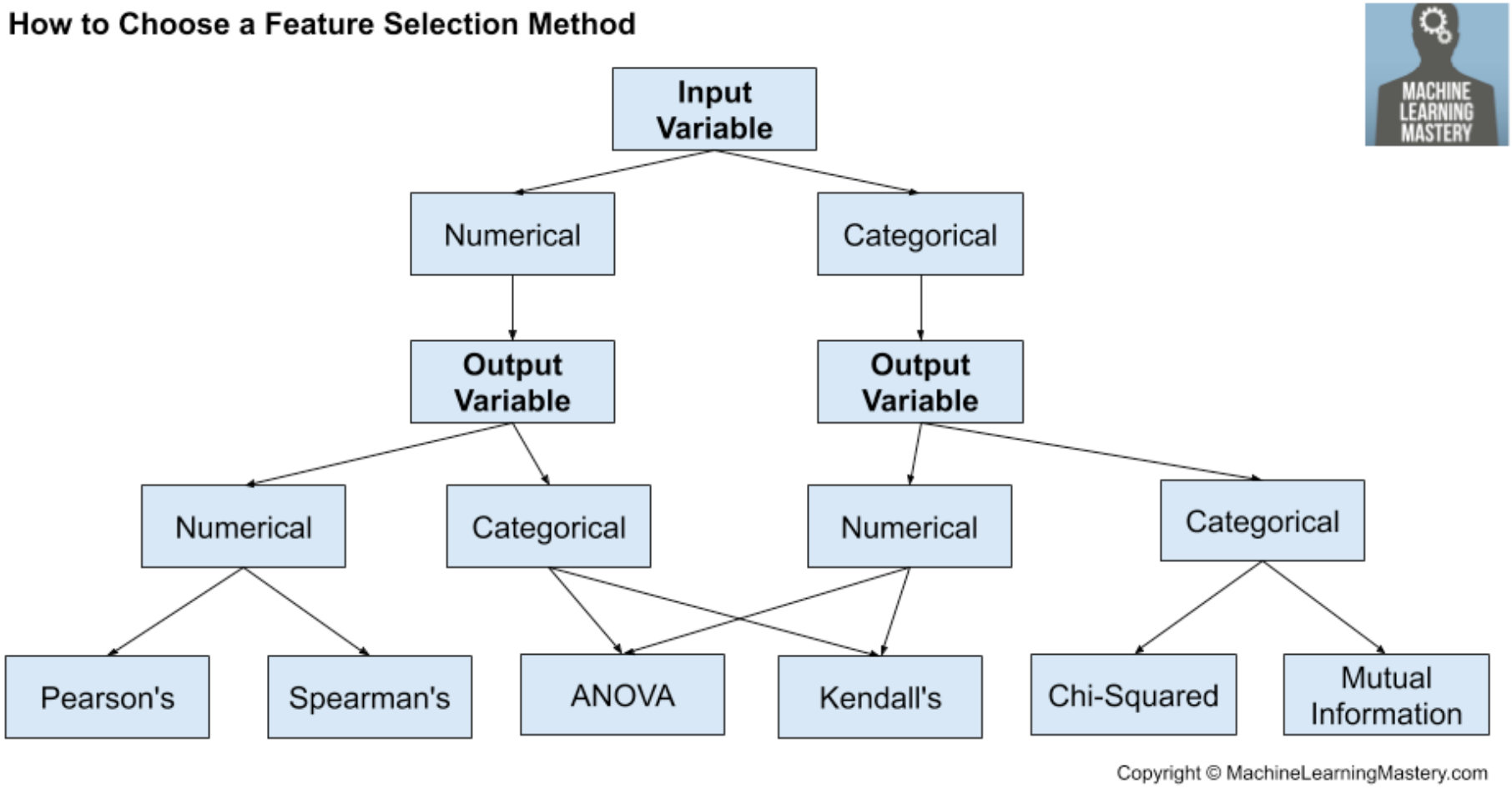
- Feature selection Why?
  - Big Data:
    - It enables the machine learning algorithm **to train faster**.
    - Memory constraints
  - Keeps meaningful features which enables **interpretability**
  - It **reduces the complexity of a model** and makes it easier to interpret
  - It **improves the accura**cy of a model if the right subset is chosen.
  - It **reduces overfitting**

Feature Selection

- Approaches
  - Filter method: rank variables using a scoring function
  - Scoring function
    - Univariate
    - Multivariate
  - Advantage: High computational and statistical scalability
  - Disadvantage:
    - Low ranked variables might provide good information when taken together -> multivariate
    - Low ranked variables might provide useful information for a give ML algorithm

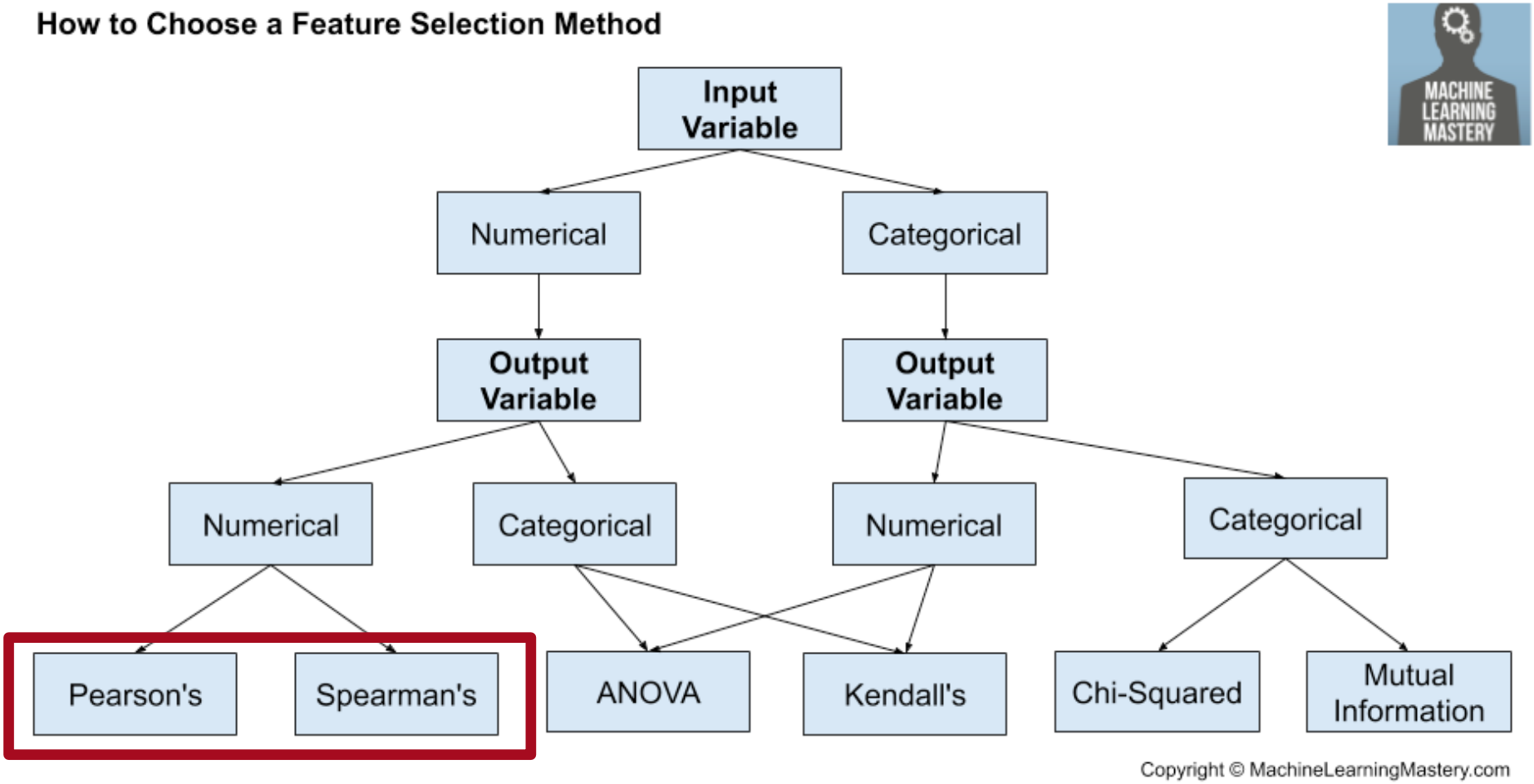
Feature Selection

- Approaches



Feature Selection

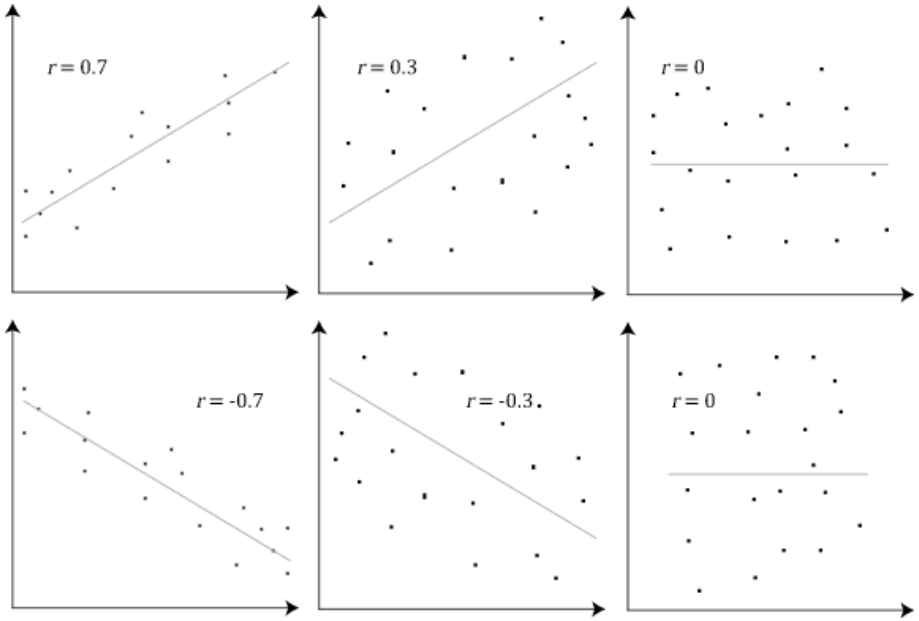
- Approaches



Feature Selection

- Approaches
  - Person's Correlation: assesses the linear relationship

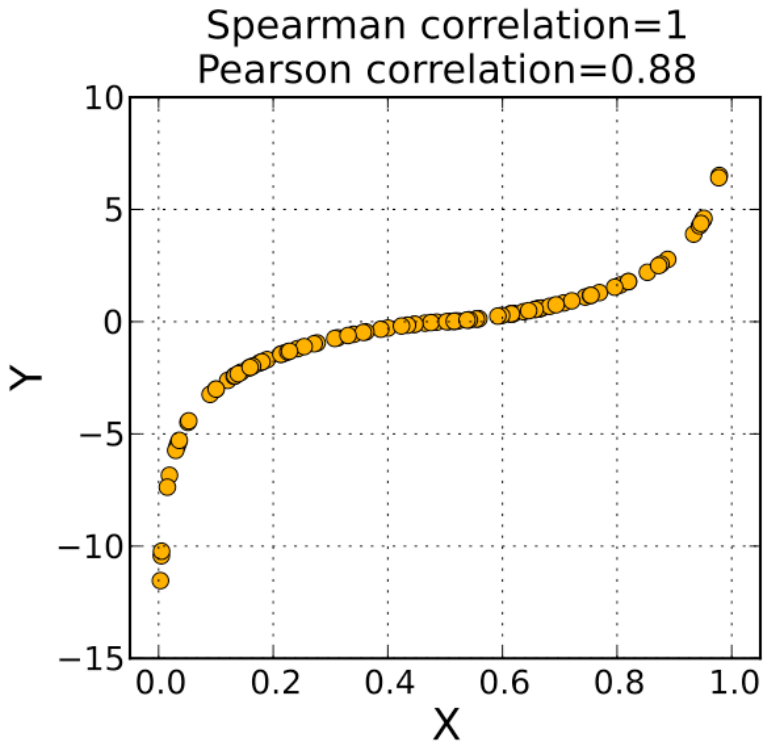
$$\rho(x,y) = \frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{var}(x)\,\mathrm{var}(y)}} \in [-1,1]$$



Feature Selection

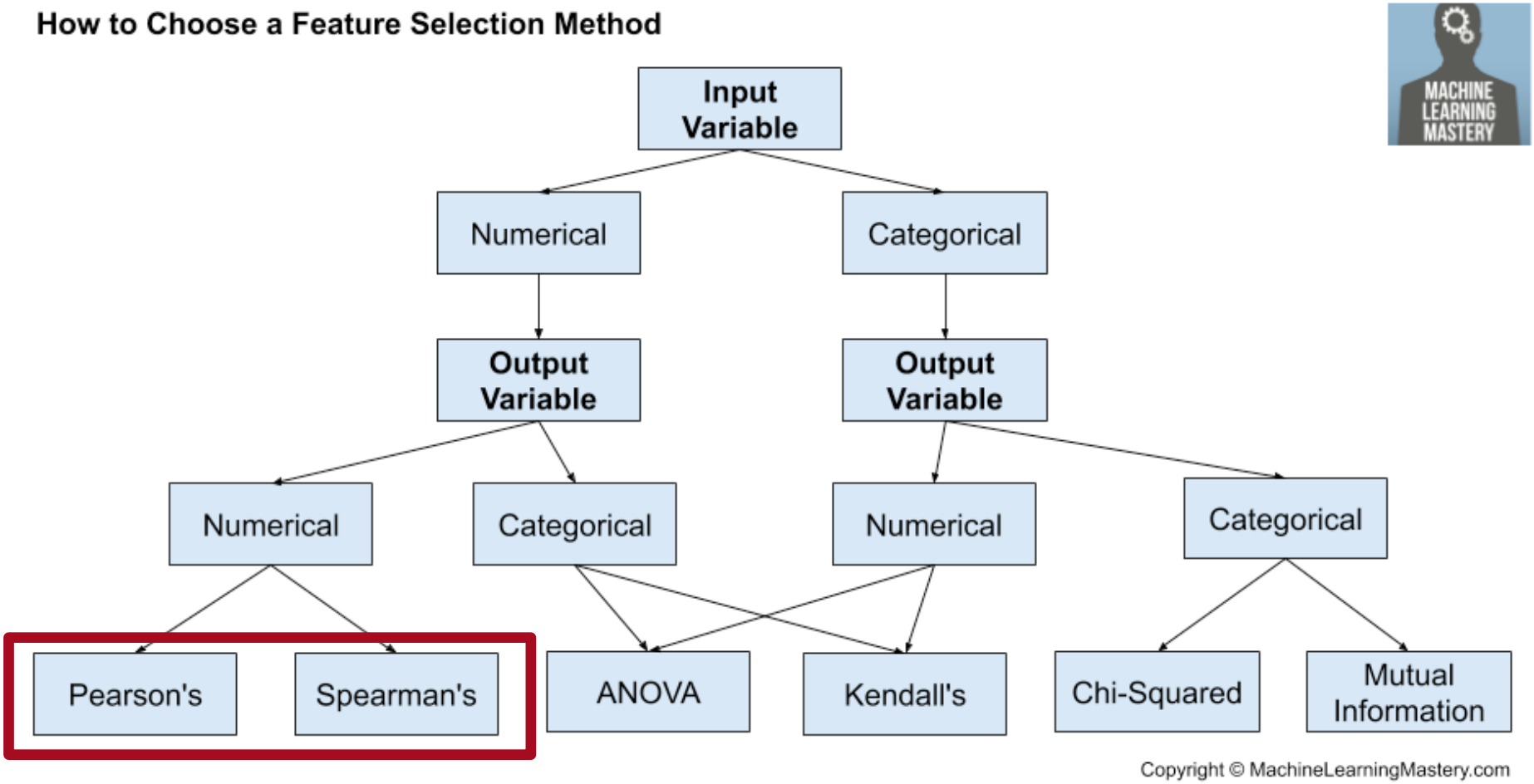
- Approaches
  - Spearman's Correlation: assesses the the monotonic relationship
  - Use when variables are not normally distributed or the relationship between the variables is not linear
  - Spearman's Correlation: Pearson's correlation between the ranked variables

$$\rho(x,y) = \frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{var}(x)\,\mathrm{var}(y)}} \in [-1,1]$$



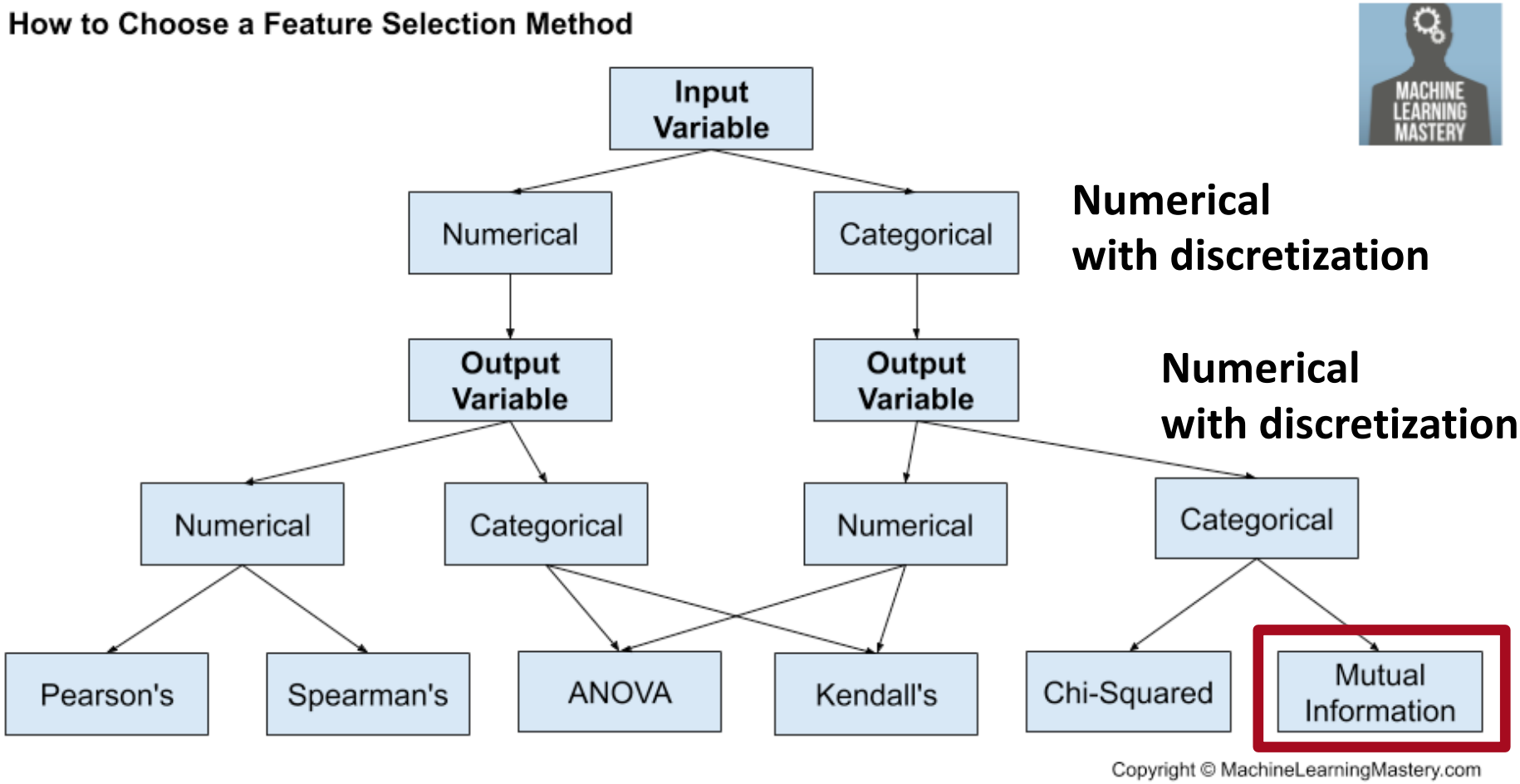
Feature Selection

- Approaches



Feature Selection

- Approaches



Kullback-Leibler

- Kullback-Leibler Divergence
  - Divergence between two distributions $P(x)$ e $Q(x)$ over the same alphabet A_x

$$D_{KL}(P,Q) = \sum_{x \in A_x} P(x) \log_2 \frac{P(x)}{Q(x)}$$

  - Gibbs inequality

$$D_{KL}(P,Q) \geq 0$$

  - Usually

$$D_{KL}(P,Q) \neq D_{KL}(Q,P)$$

Kullback-Leibler

- Interpretation

$$D_{KL}(P,Q) = \sum_{x \in A_x} P(x) \log_2 \frac{P(x)}{Q(x)}$$

  - if $P(x)=Q(x)$

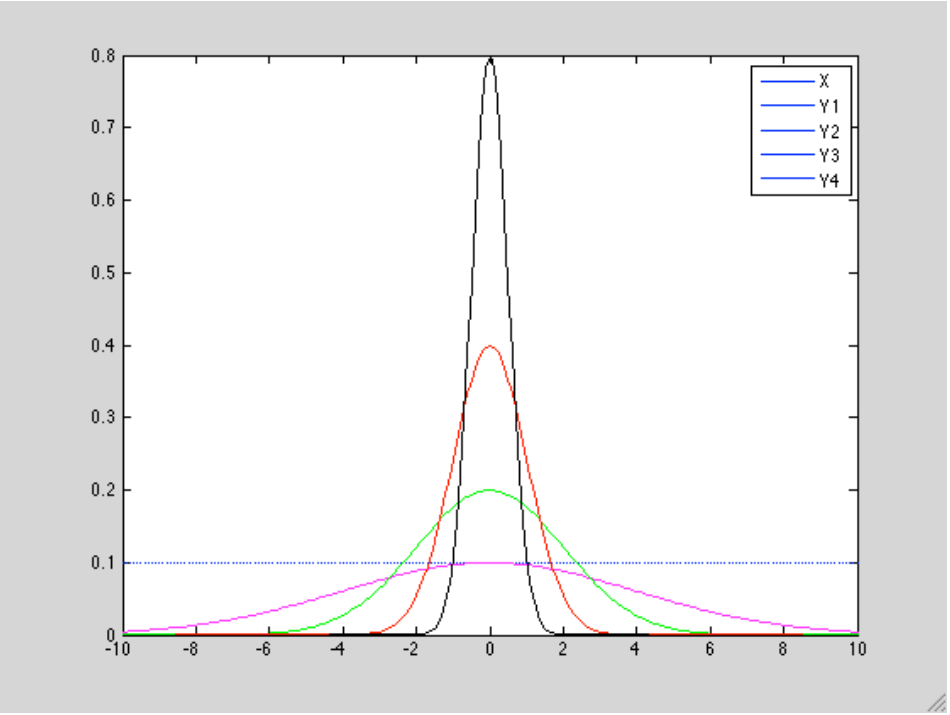
$$\log_2 \frac{P(x)}{Q(x)} = \log_2 1 = 0 \Rightarrow D(P,Q) = 0$$

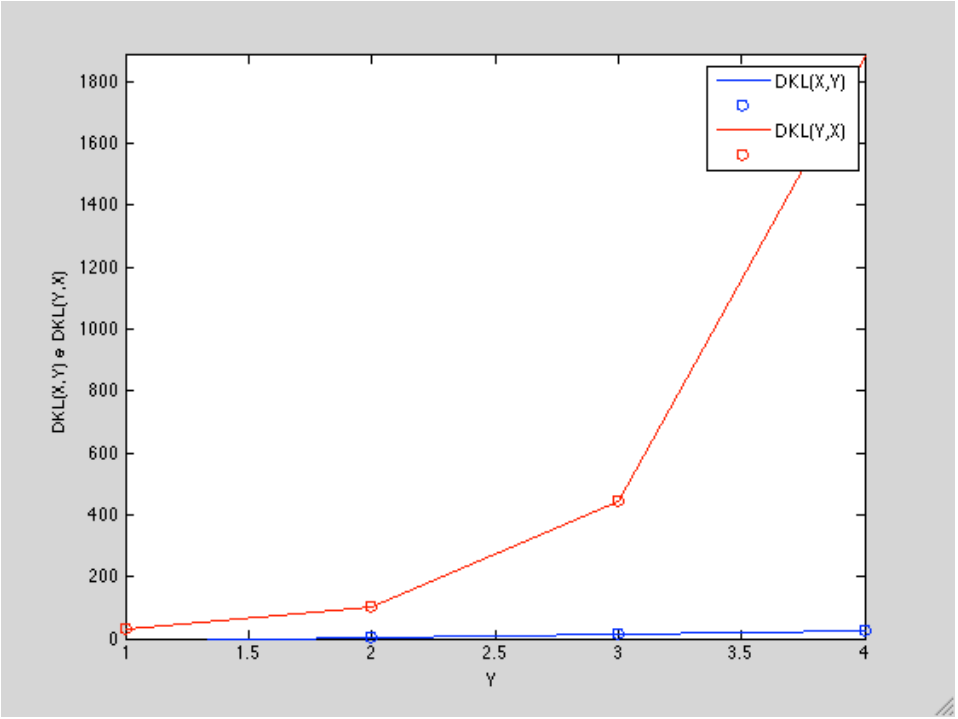
  - if $P(x) \neq Q(x)$

$$\left| \log_2 \frac{P(x)}{Q(x)} \right| \uparrow \Rightarrow D(P,Q) \uparrow$$

Desigualdade de Gibbs

- Interpretação (DKL entre dist. Uniforme e Gaussiana)





Feature Selection

- Mutual Information
  Assesses dependency

$$I(X;Y) = D_{KL}\left(P(X,Y), P(X)P(Y)\right)$$
$$= \sum_{x\in A_x}\sum_{y\in A_y} P(x,y)\log_2 \frac{P(x,y)}{P(x)P(y)}$$
$$= \sum_{x\in A_x}\sum_{y\in A_y} P(x,y)\log_2 \frac{P(y\mid x)}{P(y)}$$

  - If independent
  - If completely dependent

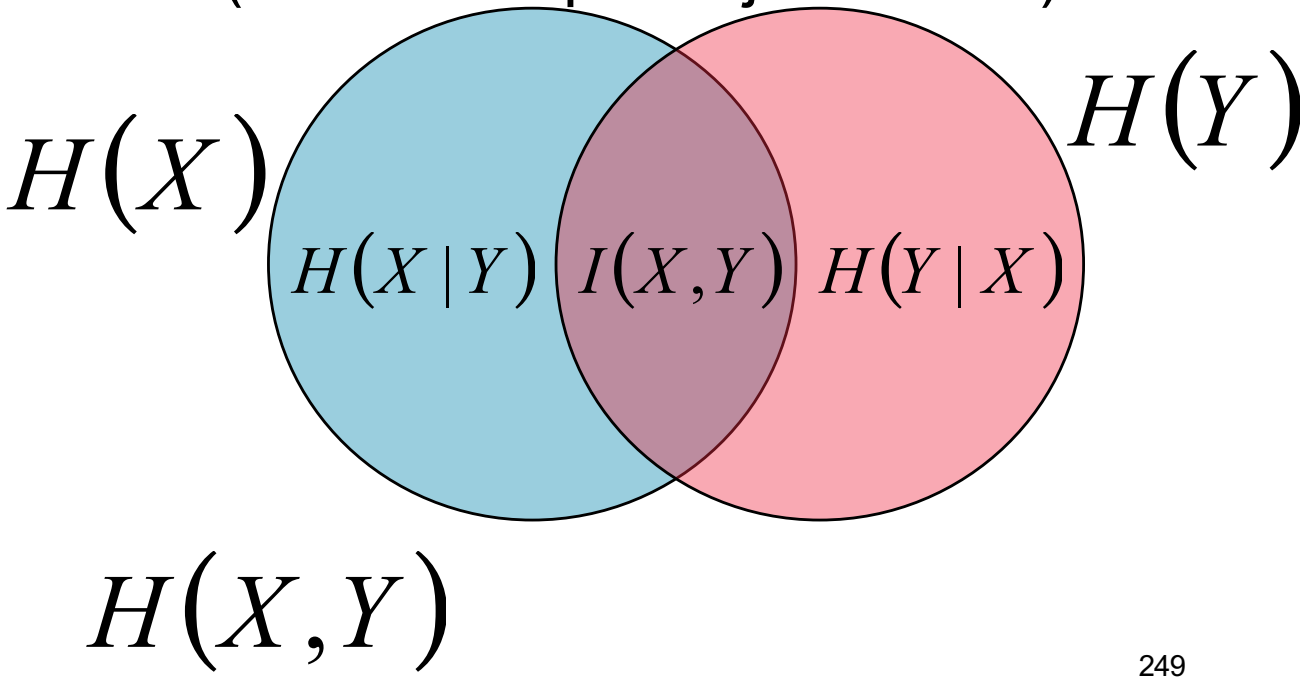
$$\log_2 \frac{P(y\mid x)}{P(y)} = \log_2 1 = 0 \Leftrightarrow I(X,Y) = 0$$

$$P(Y = y_i \mid X = x_j) = \begin{cases} 1 \Rightarrow x_j = y_i \\ 0 \Rightarrow x_j \neq y_i \end{cases}$$

$$I(X;Y) = D_{KL}\left(P(X,Y), P(X)P(Y)\right)$$
$$= \sum_{x\in A_x}\sum_{y\in A_y} P(x,y)\log_2 \frac{P(y\mid x)}{P(y)}$$
$$= \sum_{y\in A_y} P(y)\log_2 \frac{1}{P(y)}$$
$$= H(Y)$$

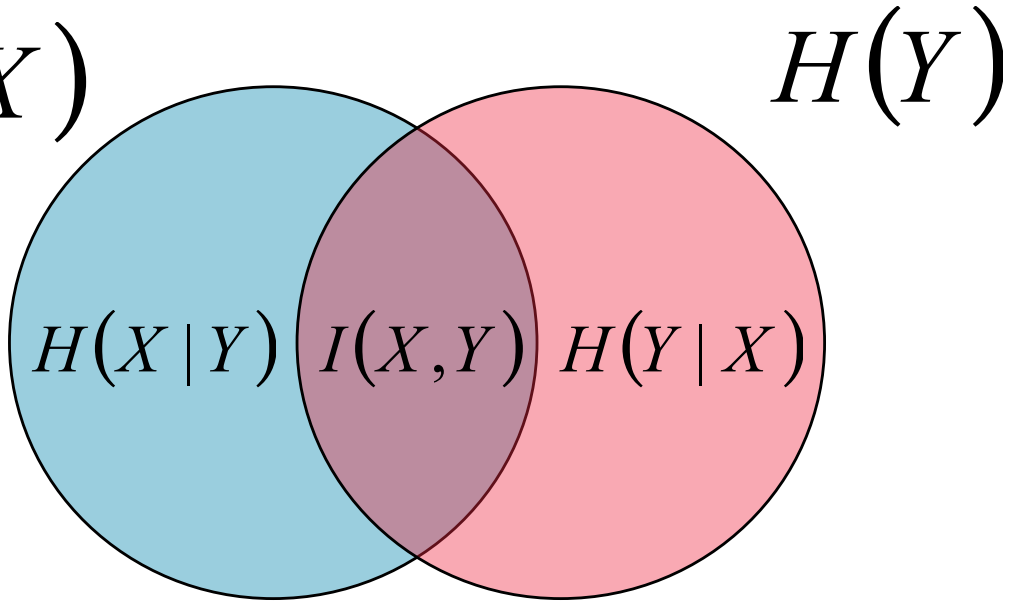
Tópicos Adicionais de Entropia

- Informação Mútua
  - H(Y|X) – incerteza remanescente após a observação de Y e conhecendo X (Interessa que seja mínima)
  - H(X|Y) – incerteza remanescente após a observação de Y e conhecendo X (Interessa que seja mínima)



Tópicos Adicionais de Entropia

- Informação Mútua



$$I(X,Y) = \sum_{x \in A_x} \sum_{y \in A_y} P(x,y) \log_2 \frac{P(y \mid x)}{P(y)}$$

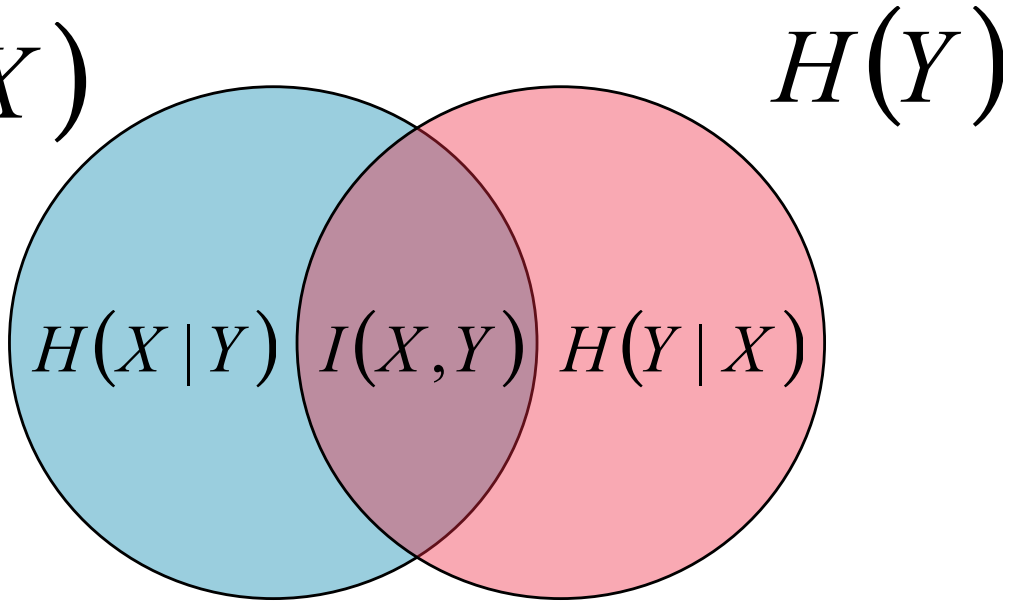
$$= \sum_{y \in A_y} P(y) \log_2 \frac{1}{P(y)} - \sum_{x \in A_x} \sum_{y \in A_y} P(x,y) \log_2 \frac{1}{P(y \mid x)}$$

$$= H(Y) - H(Y \mid X)$$

Interessa que a incerteza após tx seja mínima

Tópicos Adicionais de Entropia

- Information Gain

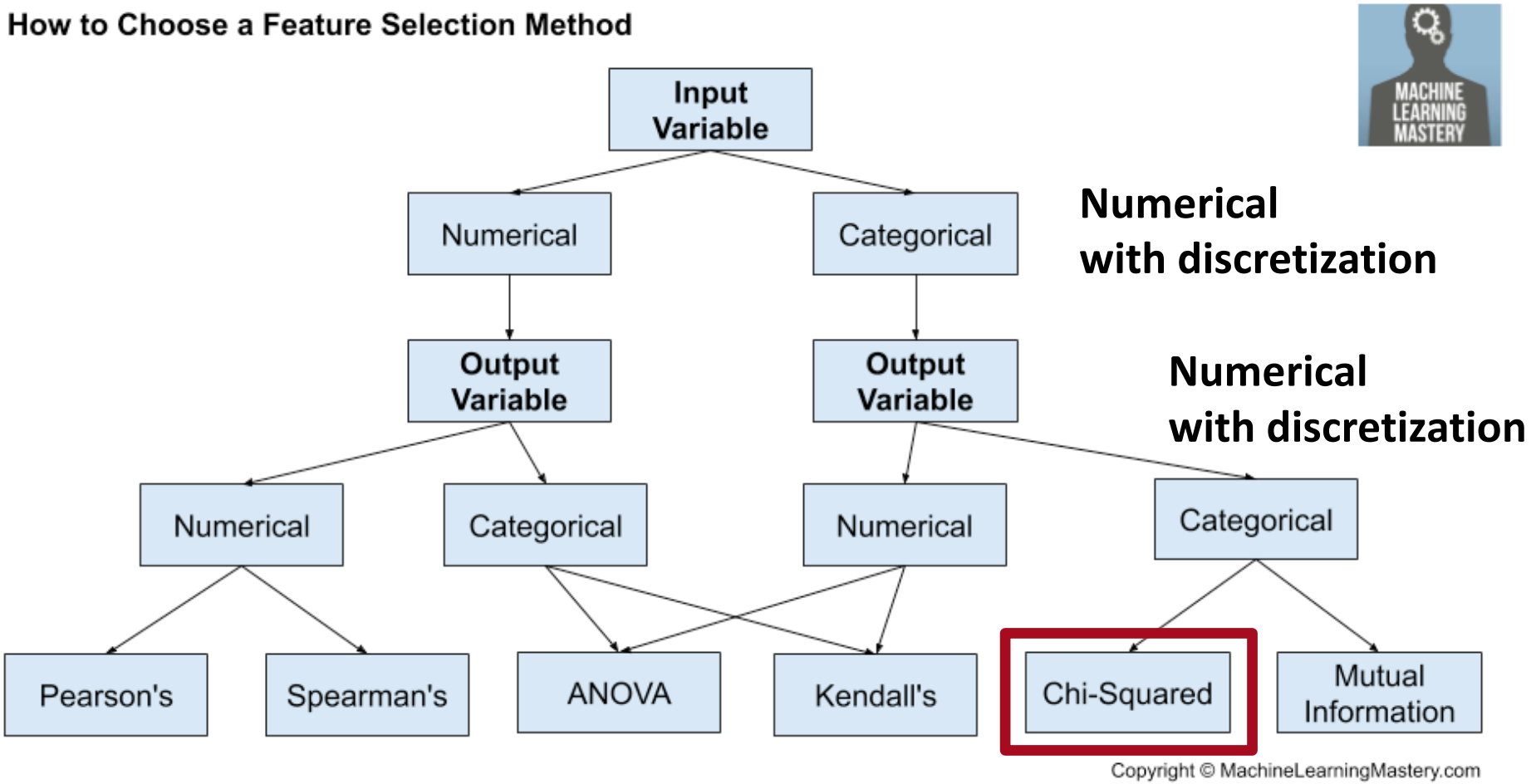


$$I(X,Y) = I(Y,X)$$
$$= H(Y) - H(Y|X)$$
$$= H(X) - H(X|Y)$$

$$\frac{I(X,Y)}{\min(H(X),H(Y))} \in [0,1]$$

Feature Selection

- Approaches



Feature Selection

- Chi-Square
  - A random variable X follows a Chi-Square distribution if we can write it as sum of **squared normal** variables.
  - $N \geq 30\ samples$
  - Samples should be Normally distributed

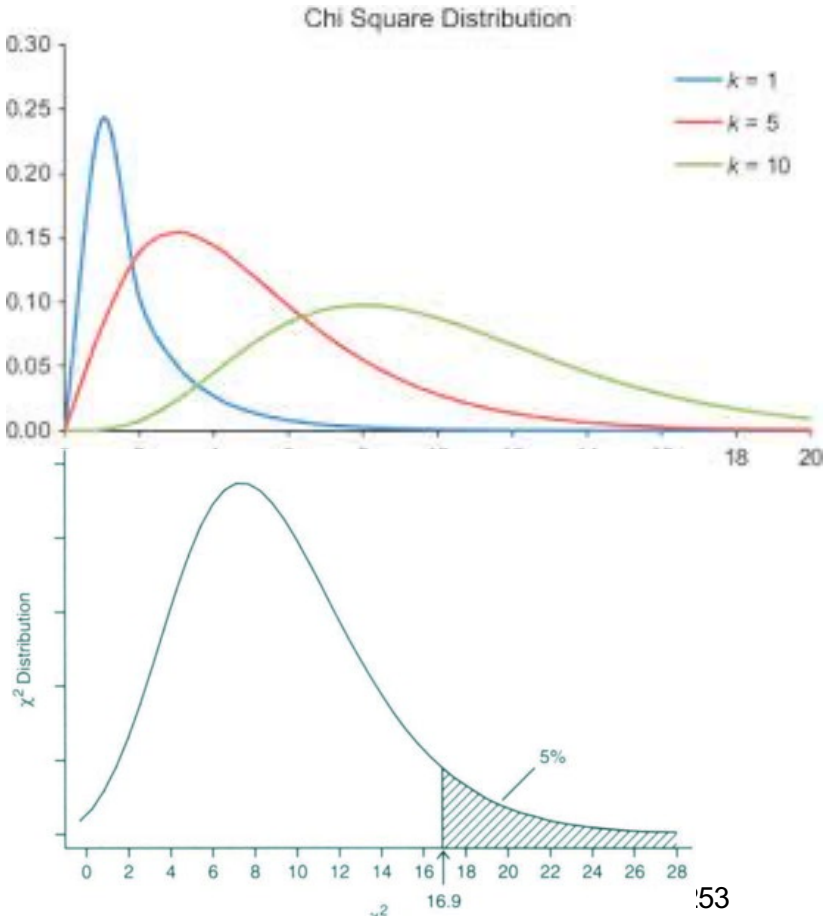
$$\chi_c^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

where:

$c =$ degrees of freedom

$O =$ observed value(s)

$E =$ expected value(s)



Feature Selection

- Chi-Square
  - Define Hypothesis.
  - Build a Contingency table.
  - Find the expected values.
  - Calculate the Chi-Square statistic.
  - Accept or Reject the Null Hypothesis

- **Define Hypothesis**

  Null Hypothesis (H0): Two variables are independent.
  Alternate Hypothesis (H1): Two variables are not independent.

Feature Selection

- **Contingency table (example adopted from https://towardsdatascience.com)**
  - Variable Gender = {Male, Female}, variable Excited={Yes, No}.
  - **df = (rows–1) * (columns–1) = 1 – degrees of Freedom**

Exited\ Gender	Yes	No	Total
Male	38	178	216
Female	44	140	184
Total	82	318	400

Feature Selection

- **Find the Expected Value**
  - Null Hypothesis (H0): Two variables are independent.
  - To be independent: $P(Gender \cap Exited) = P(Gender) * P(Exited)$
  - $P(Gender = Male) = \frac{216}{400}$
  - $P(Gender = Female) = \frac{184}{400}$
  - $P(Excited = Yes) = \frac{82}{400}$
  - $P(Excited = No) = \frac{318}{400}$

Observed Values

Exited\ Gender	Yes	No	Total
Male	38	178	216
Female	44	140	184
Total	82	318	400

  - Expected Values:
    - E$(Gender = Male \cap Exited = Yes) = P(Gender = Male \cap Exited = Yes) * N$= $P(Gender = Male) * P(Exited = Yes) * N = N * P(Gender = Male \cap Exited = Yes) =$ 82/400∗216/400∗400 = 44

Expected Values

Exited\Gender	Yes	No
Male	44	172
Female	38	146

Feature Selection

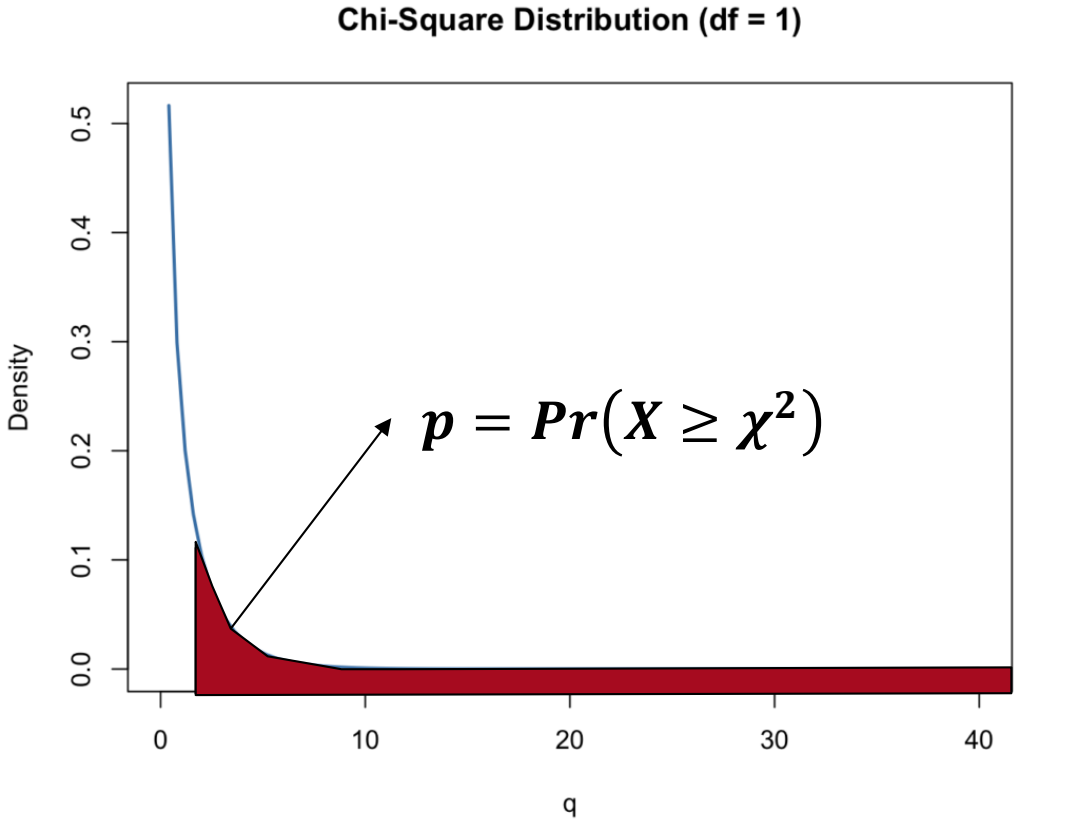
- **Calculate Chi-Square value**

Gender,Exited	O	E	O-E	Square of O-E	(Square of O-E) / E
Male,Yes	38	44	-6	36	0.818181818
Male,No	178	172	6	36	0.209302326
Female,Yes	44	38	6	36	0.947368421
Femal,No	140	146	-6	36	0.246575342
Chi Square Value					2.221427907

Feature Selection

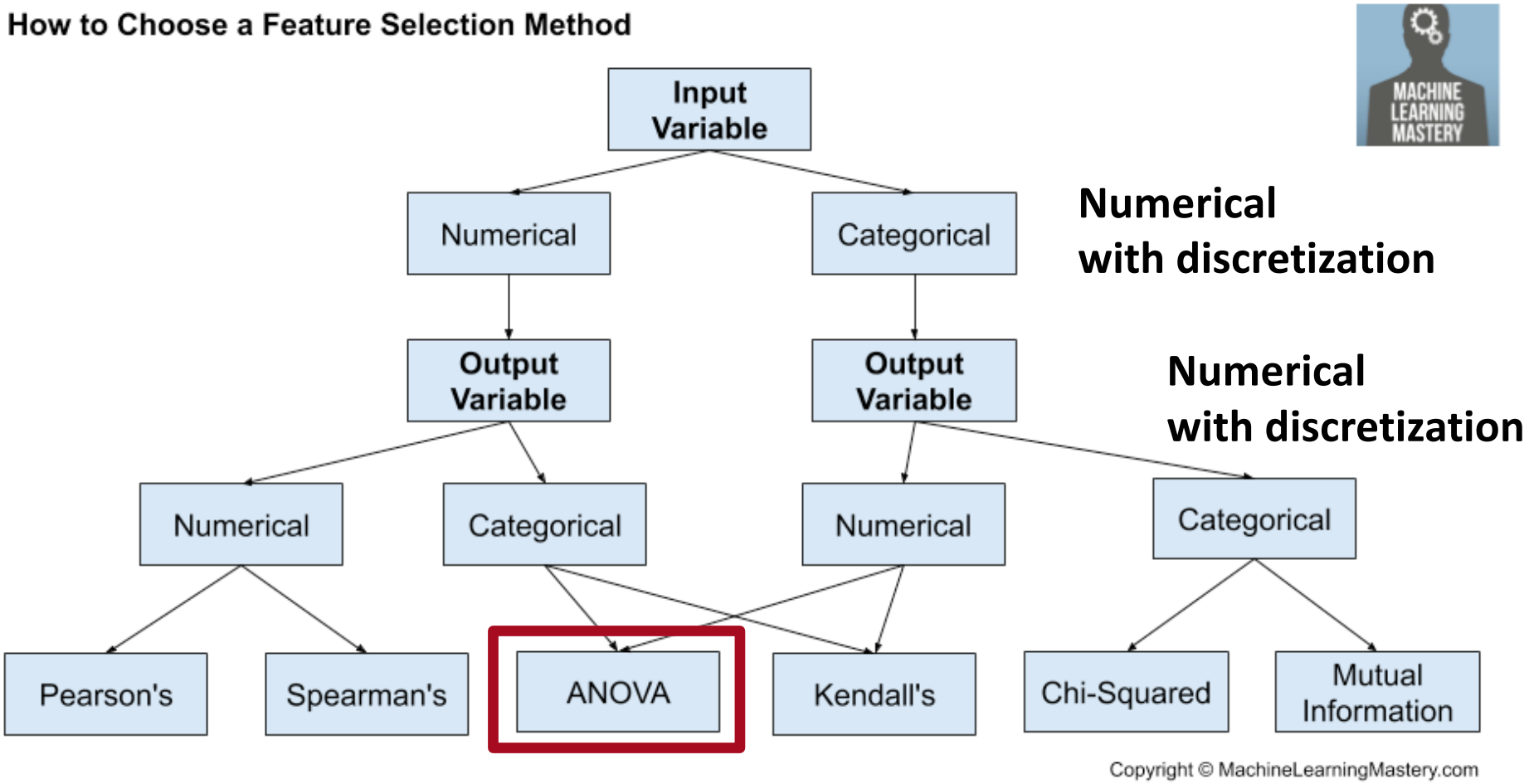
- **Calculate Chi-Square value**
- χ^2 **=2.22**
- **For** $\alpha = 5\%, \chi^2 = 3.841$ **(**$0.05 = Pr(X \geq 3.841)$**)**
- **H0 is accepted: variables are independent**

	P										
DF	0.995	0.975	0.2	0.1	0.05	0.025	0.02	0.01	0.005	0.002	0.001
1	.0004	.00016	1.642	2.706	3.841	5.024	5.412	6.635	7.879	9.55	10.828
2	0.01	0.0506	3.219	4.605	5.991	7.378	7.824	9.21	10.597	12.429	13.816
3	0.0717	0.216	4.642	6.251	7.815	9.348	9.837	11.345	12.838	14.796	16.266
4	0.207	0.484	5.989	7.779	9.488	11.143	11.668	13.277	14.86	16.924	18.467
5	0.412	0.831	7.289	9.236	11.07	12.833	13.388	15.086	16.75	18.907	20.515
6	0.676	1.237	8.558	10.645	12.592	14.449	15.033	16.812	18.548	20.791	22.458
7	0.989	1.69	9.803	12.017	14.067	16.013	16.622	18.475	20.278	22.601	24.322
8	1.344	2.18	11.03	13.362	15.507	17.535	18.168	20.09	21.955	24.352	26.124
9	1.735	2.7	12.242	14.684	16.919	19.023	19.679	21.666	23.589	26.056	27.877
10	2.156	3.247	13.442	15.987	18.307	20.483	21.161	23.209	25.188	27.722	29.588
11	2.603	3.816	14.631	17.275	19.675	21.92	22.618	24.725	26.757	29.354	31.264
12	3.074	4.404	15.812	18.549	21.026	23.337	24.054	26.217	28.3	30.957	32.909



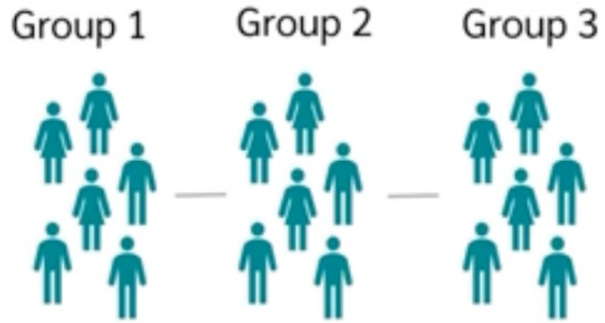
Feature Selection

- Approaches



Feature Selection

Hypotheses in one-factor analysis of variance



Null hypothesis H0:

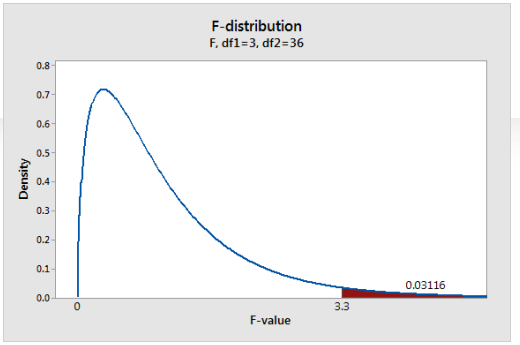
There are no differences in the population between the means of the individual groups.

Alternative hypothesis H1:

At least two group means differ from each other in the population.

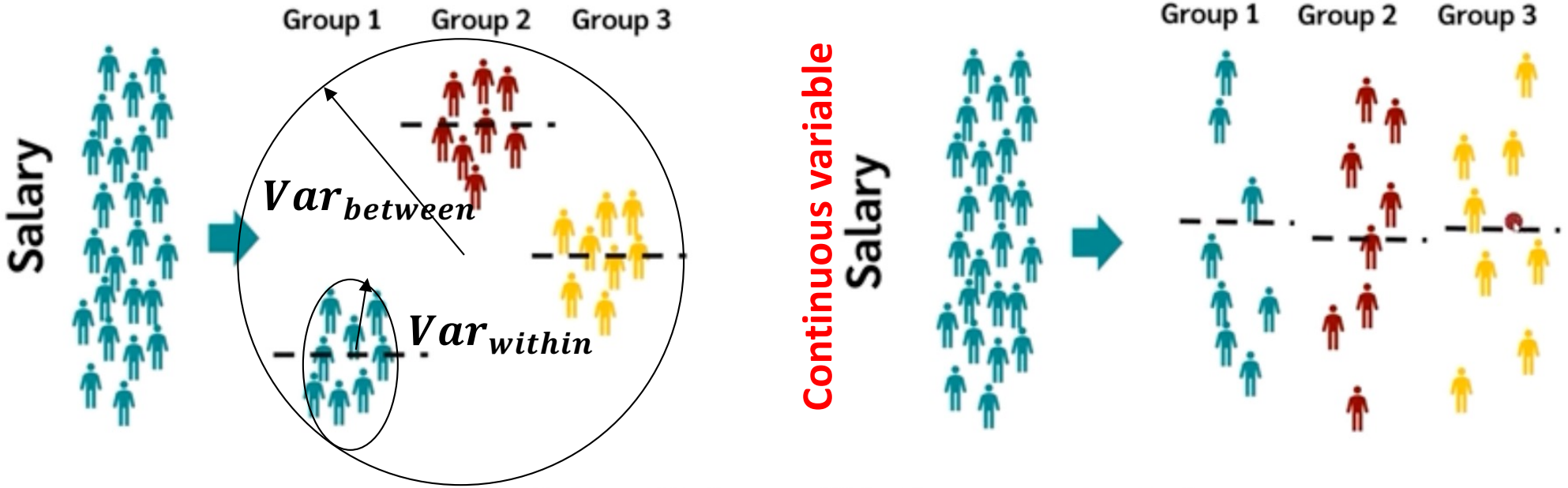
Feature Selection

$$F_{(k-1,n-k)}$$



Explained Variance

$$F = \frac{Var_{between}}{Var_{within}}$$



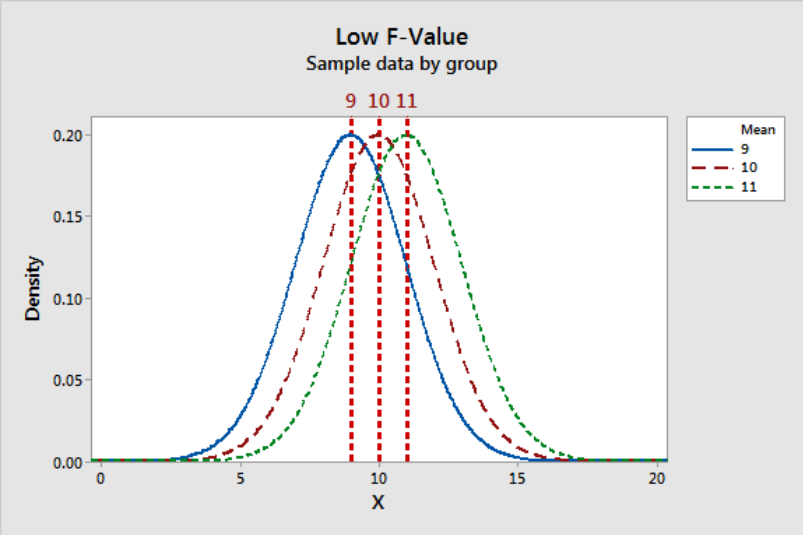
Feature Selection

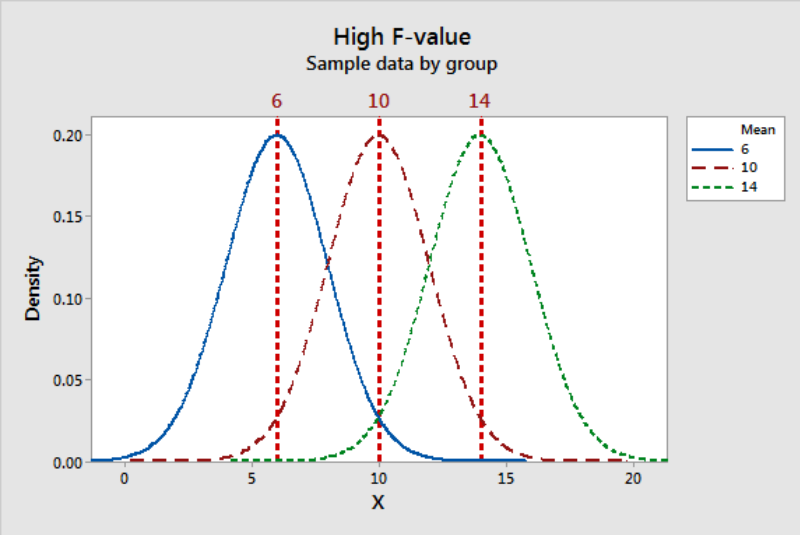
$$F = \frac{Var_{between}}{Var_{within}}$$

→ **Should be small**

→ **The higher the within variance the higher the between varance we should expect**

Quotient solves this!





Feature Selection

- ANOVA
  - Define Hypothesis
  - Determine degrees of freedom
  - Calculate the Sum of Squares
  - F-value
  - Accept or Reject the Null Hypothesis

Feature Selection

- Example:

Continuous variable

Categorical variable

Group 1	Group 2	Group 3
1	2	2
2	4	3
5	2	4

- **Define the Hypothesis**
  - H0: $\mu_1 = \mu_2 = \mu_3$
  - H1: there is at least one difference among the means

  - $\alpha = 0.05$

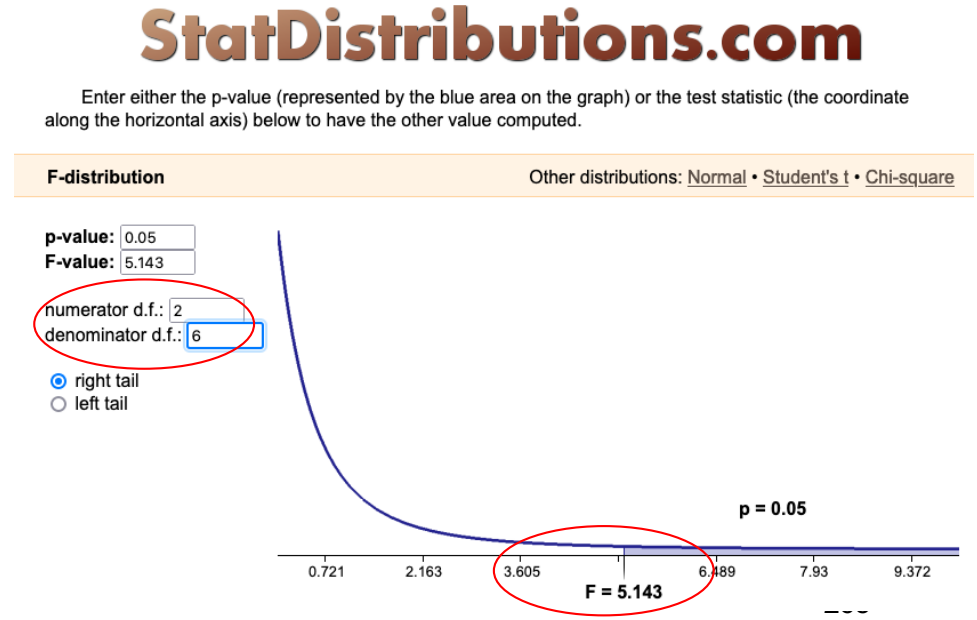
Feature Selection

Continuous variable

Categorical variable

Group 1	Group 2	Group 3
1	2	2
2	4	3
5	2	4

- Determine degrees of freedom:
  - Between groups $df_{between} = k - 1 = 3 - 1 = 2$, k = number of groups
  - Within groups $df_{within} = n - k = 9 - 3 = 6$, n = number of points
  - $F_{critical} = 5.14$



Feature Selection

Continuous variable

Categorical variable

Group 1	Group 2	Group 3
1	2	2
2	4	3
5	2	4

- Calculate the Sum of Squares
  - Group means:
    - $\mu_1 = \frac{1+2+5}{3} = 2.67$
    - $\mu_2 = \frac{2+4+2}{3} = 2.67$
    - $\mu_3 = \frac{2+3+4}{3} = 3.00$
  - $SS_{within} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left(x_{ij} - \mu_i\right)^2 =$
  - $SS_{within} = (1-2.67)^2 + (2-2.67)^2 + (5-2.67)^2 + (2-2.67)^2 + (4-2.67)^2 + (2-2.67)^2 + (2-3.00)^2 + (3-3.00)^2 + (4-3.00)^2 = 13.34$
  - $Var_{within} = \frac{SS_{within}}{df_{within}} = \frac{13.34}{6} = 2.22$

Feature Selection

Continuous variable

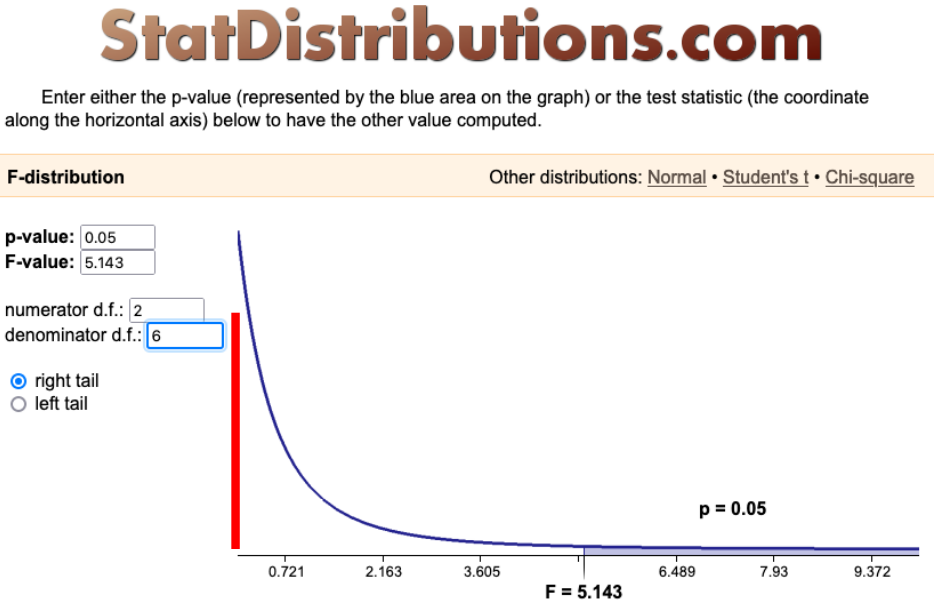
Categorical variable

Group 1	Group 2	Group 3
1	2	2
2	4	3
5	2	4

- Calculate the Sum of Squares
  - Grand mean:
    - $\bar{\bar{\mu}} = \frac{1+2+5+2+4+2+2+3+4}{9} = 2.78$
  - $SS_{Total} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left(x_{ij} - \bar{\bar{\mu}}\right)^2 =$
  - $SS_{Total} = (1-2.78)^2 + (2-2.78)^2 + (5-2.78)^2 + (2-2.78)^2 + (4-2.78)^2 + (2-2.78)^2 + (2-2.78)^2 + (3-2.78)^2 + (4-2.78)^2 = 13.60$
  - $SS_{Total} = SS_{between} + SS_{within}, SS_{Total} = SS_{between} = -SS_{Total} - SS_{within} = 0.23$
  - $Var_{between} = \frac{SS_{between}}{df_{between}} = \frac{0.23}{2} = 0.12$

Feature Selection

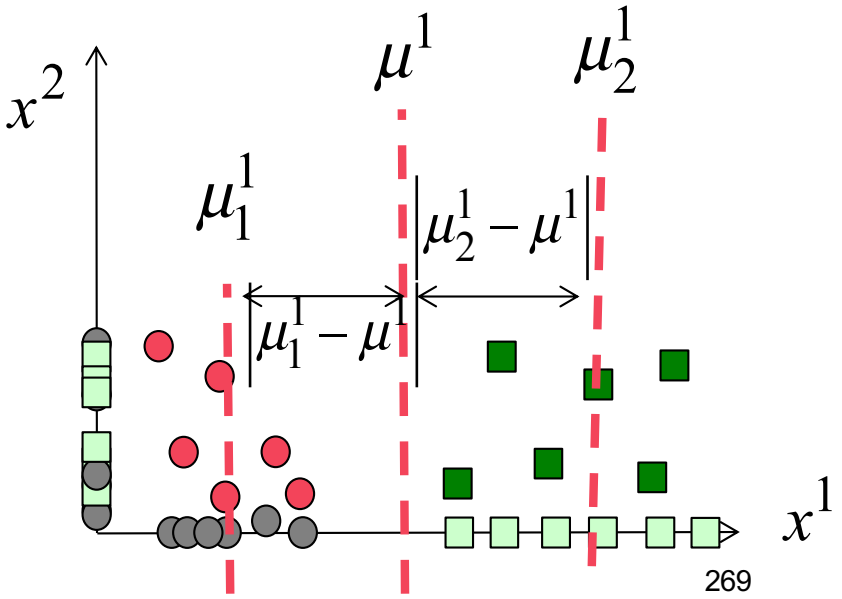
- F-value
  - $F = \frac{Var_{between}}{Var_{within}} = \frac{0.12}{2.22} = 0.05$
- Accept or Reject the Null Hypothesis
  - $F < Fcritical$
  - We cannot reject H0



Feature Selection

- Multivariate
  - Fisher score:
    - Irrelevant features (Y) will make instances with different classes seam closer as they might overlap
    - A good feature set (X) will be one where the instances for a given **target class are very close to each other** and **very far from the instances of other classes**
    - **Maximize inter-class variability**
    - **Minimize intra-class variability**

$$F\left(x^j\right) = \frac{\sum_{k=1}^{c} n_k \left(\mu_k^j - \mu^j\right)^2}{\sum_{k=1}^{c} n_k \underbrace{\left(\sigma_k^j\right)^2}_{\text{intra-class scatter}}} = \frac{\sum_{k=1}^{c} n_k \left(\mu_k^j - \mu^j\right)^2}{\underbrace{\left(\sigma^j\right)^2}_{\text{total scatter}}}$$



Feature Selection

$$F\left(x^j\right)=\frac{\sum_{k=1}^{c} n_k\left(\mu_k^j-\mu^j\right)^2}{\sum_{k=1}^{c} n_k \underbrace{\left(\sigma_k^j\right)^2}_{\text{intra-class scatter}}}=\frac{\sum_{k=1}^{c} n_k\left(\mu_k^j-\mu^j\right)^2}{\underbrace{\left(\sigma^j\right)^2}_{\text{total scatter}}}$$

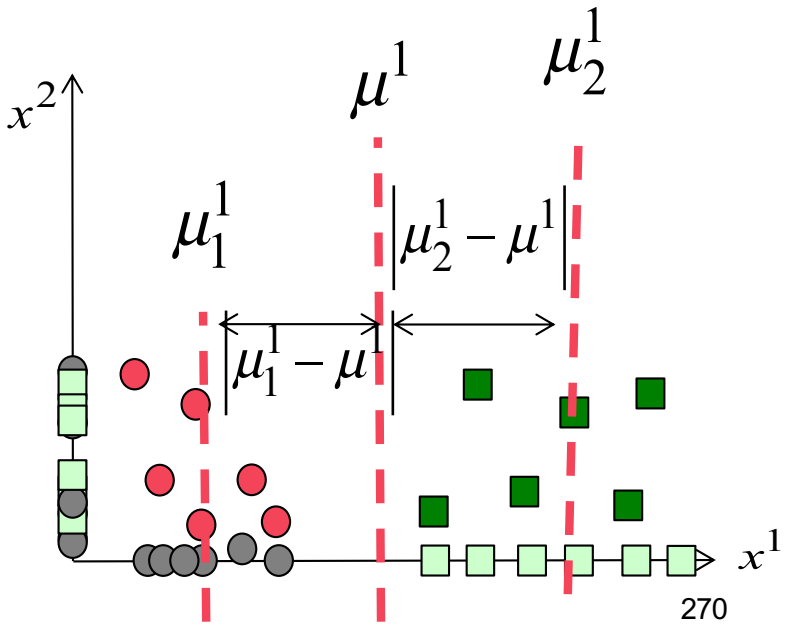
- Multivariate
  - Fisher score:
    - Let $Z=\left[x^1,x^2,...,x^w\right]$ be a feature vector
    - Let the output have *c* classes with n_i elements each
    - The Fisher-score

$$F\left(Z\right)=tr\left\{S_b\left(S_t+\gamma I\right)^{-1}\right\},\gamma>0$$

$$S_b=\sum_{k=1}^{c} n_k\left(\mu_k-\mu\right)\left(\mu_k-\mu\right)^T$$

$$S_t=\sum_{k=1}^{n}\left(z_k-\mu\right)\left(z_k-\mu\right)^T$$

$$\mu=\sum_{k=1}^{c} n_k\mu_k$$



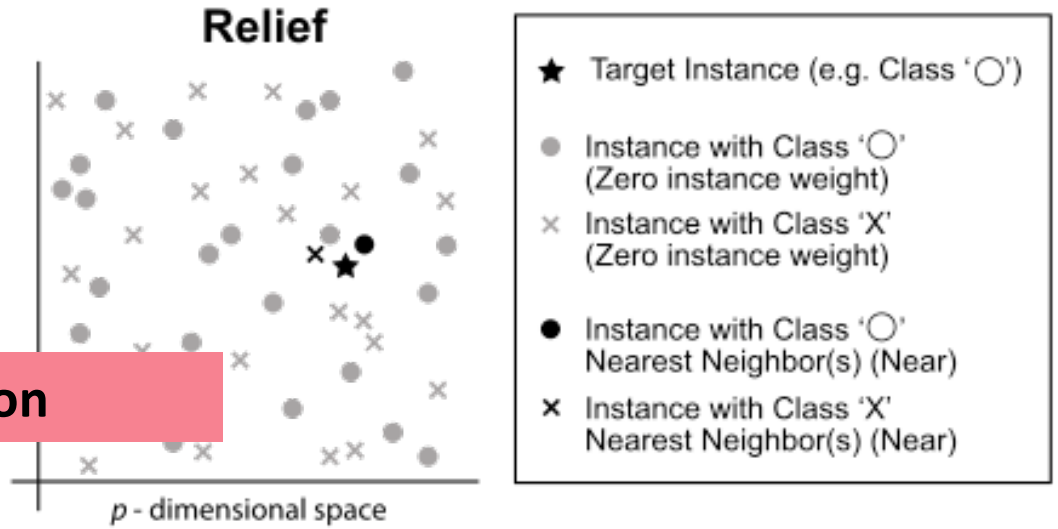
Feature Selection

- ReliefF
  - For each sample (or randomly sampled)
  - Score w_i for feature x_i

$$w_i \leftarrow w_i - \left|x_i - nearHit_i\right| + \left|x_i - nearMiss_i\right|$$

  - After iterating over the *m* available data points

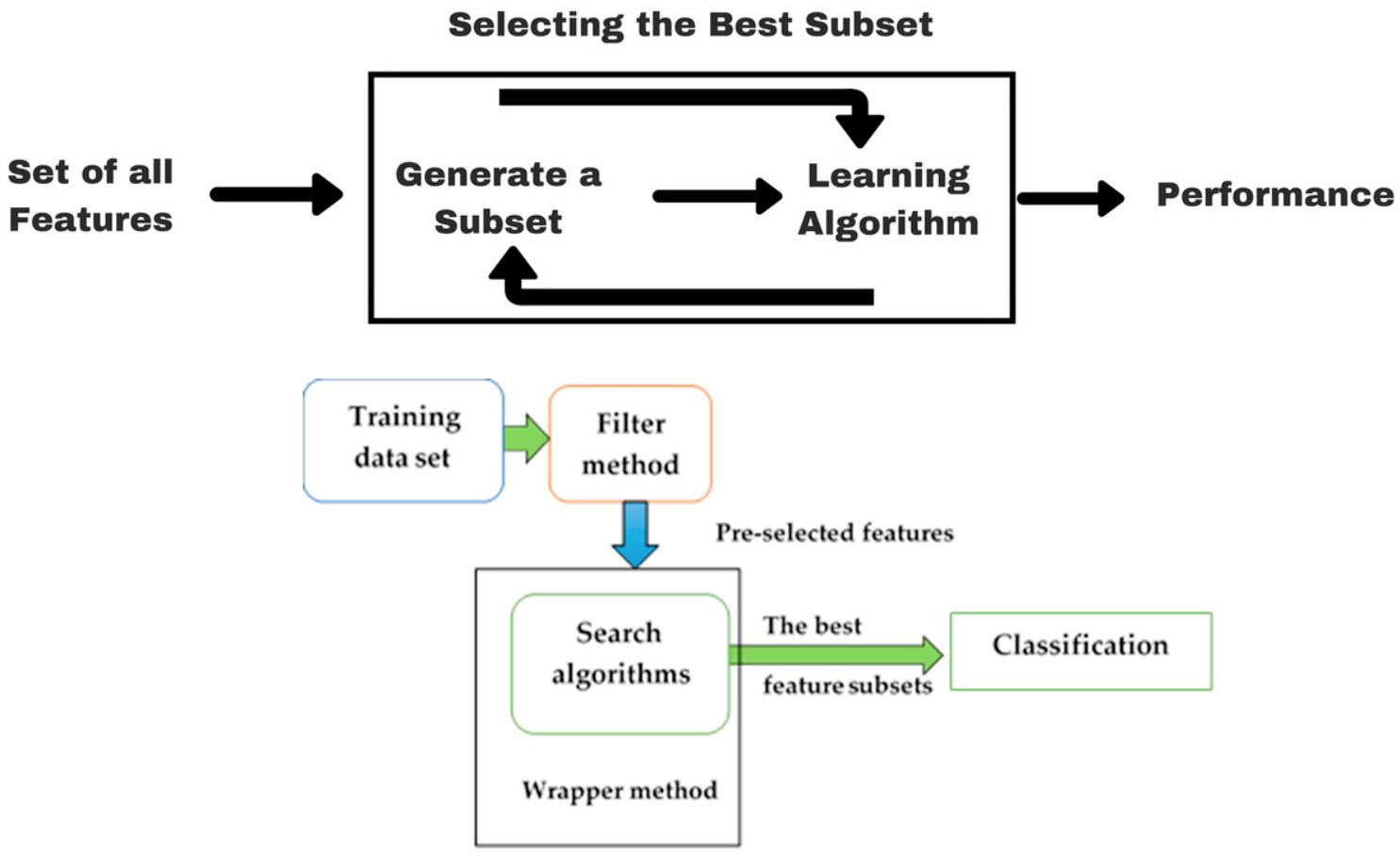
$$w_i \leftarrow \frac{w_i}{m}$$



Extensions do exist for regression

Feature Selection

- Wrapper method

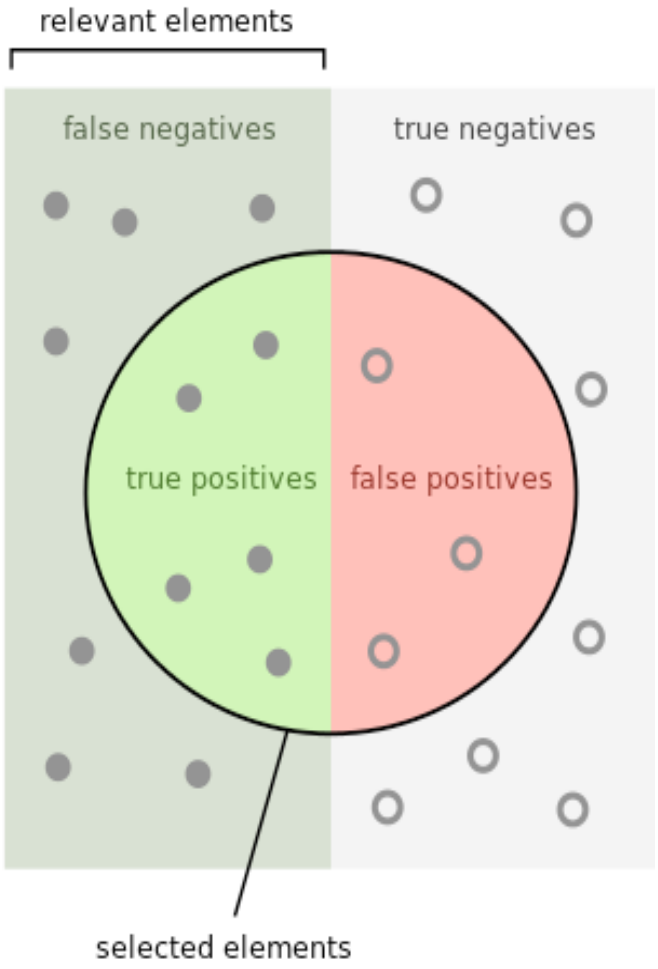


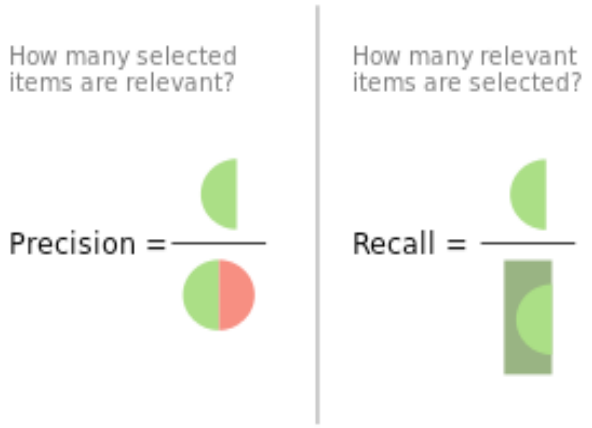
Feature Selection

- Wrapper method
  - F-Measure or F-score or F1-measure
  - Harmonic Mean of
    - Precision and
    - Recall
  - Precision = TP/(classified positives)=TP/(TP+FP)
  - Recall=TP/(real positives)=TP/(TP+FN)

$$F = \left(\frac{\text{Precision}^{-1} + \text{Recall}^{-1}}{2} \right)^{-1} =$$

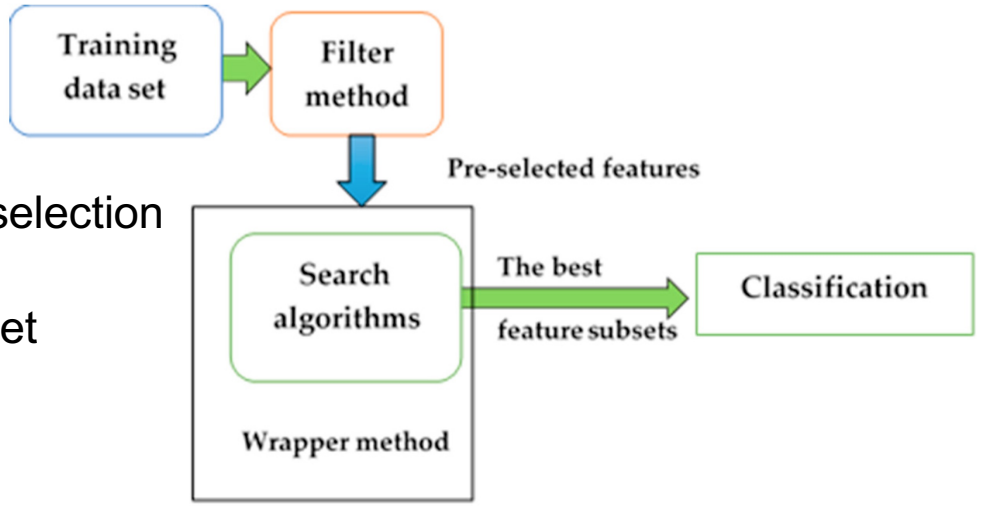
$$= \frac{Tp}{Tp + \frac{1}{2}\left(Fp + FN\right)}$$





Feature Selection

- Wrapper methods are computationally expensive
  - Complexity is $O\left(2^{\#Z}\right)$
  - Possible solutions
    - Alt1: Use filter methods for initial selection to reduce #Z
      - Apply Wrapper on reduced set
      - Forward feature selection
      - Backward feature selection
      - Termination criterion
        - » Measure does not increase
        - » Measure achieves a threshold
        - » Use a random feature (generated randomly or permutation of actual features) to define the threshold
        - » Real features which lead to lower improvement than the random feature can be dropped.



Feature Selection

- Wrapper methods are computationally expensive
  - Complexity is
  - Possible solutions $O\left(2^{\#Z}\right)$

    - Alt 2: Use an Embedded algorithm: selection is naturally performed inside the learning algorithm
    - Lasso (Least Absolute Shrinkage and Selection Operator) algorithm
      - Performs regularization
      - Performs feature selection
      - Formulation (linear function)

$$y_i = \beta_0 + x_{i,1}\beta_1 + x_{i,2}\beta_2 + ... + x_{i,k}\beta_k + \varepsilon_i, i = 1,...,n$$

$$\Leftrightarrow Y = X\beta + \varepsilon$$

$$\varepsilon_i \sim \aleph\left(0, \sigma^2\right), i = 1,...,n$$

$$\underset{\beta}{\operatorname{argmin}}\left(\frac{\left\|Y - X\beta\right\|_2^2}{n}\right), \text{ subject to } \sum_{j=1}^{k}\left\|\beta_j\right\|_1 < t \Leftrightarrow \beta(\lambda) = \underset{\beta}{\operatorname{argmin}}\left(\frac{\left\|Y - X\beta\right\|_2^2}{n} + \lambda\left\|\beta_j\right\|\right)$$

Feature Selection

- Wrapper methods are computationally expensive
  - As lambda increases, some parameters will tend to 0

  - Example: Motor Trend US magazine
  - *n*=32 observations
  - *k*=10 features (β)+ fuel consumption (y)

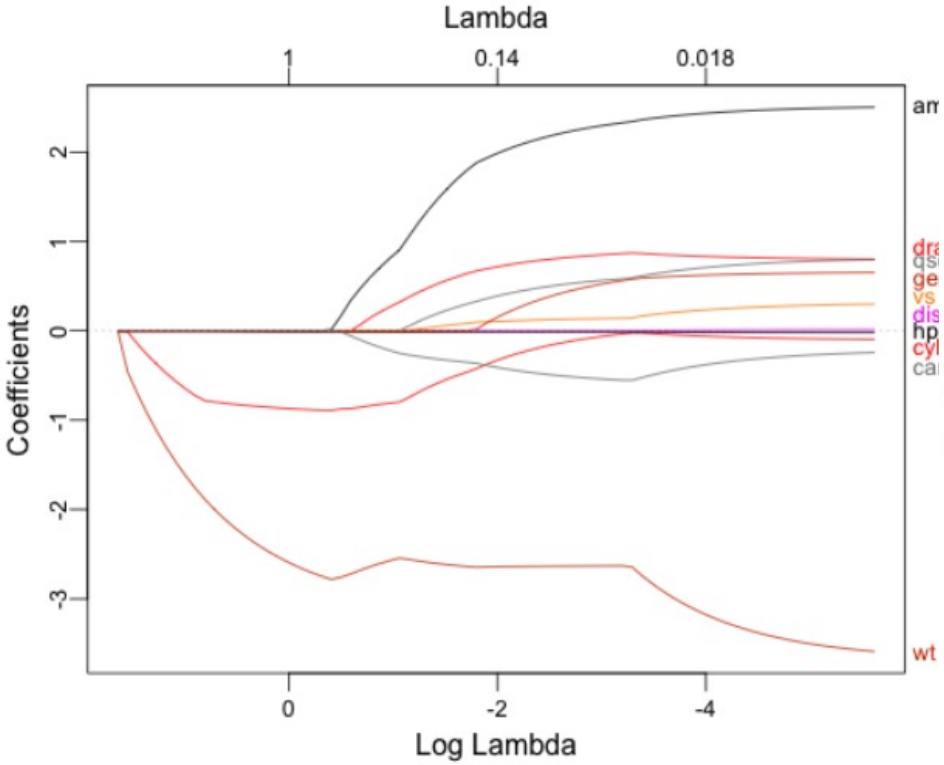
  - Features
    - cyl: number of cylinders
    - Disp: displacement
    - Hp: hourse power
    - Drat: rear axel ratio
    - Wt: weight (1000 lbs)
    - Qsec: ¼ mile time
    - Vs: V/S engine
    - Am: transmission autom. / manual
    - Gear: number of gears
    - Carb: number of carborators

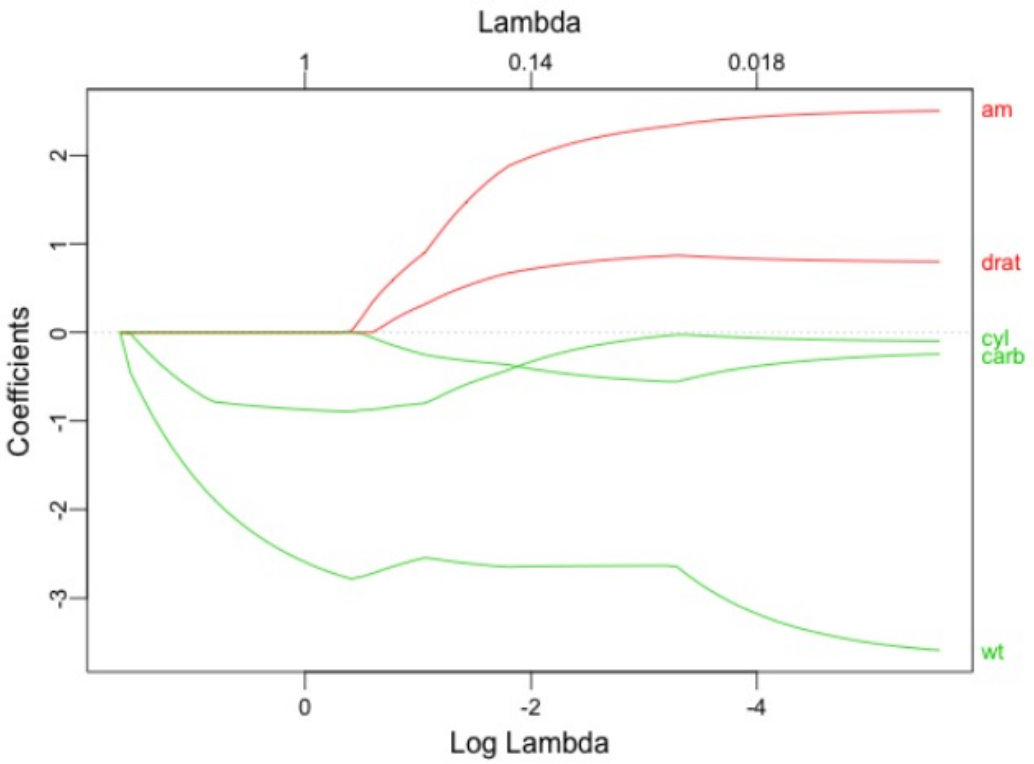
$$\beta(\lambda) = \underset{\beta}{\operatorname{argmin}} \left(\frac{\|Y - X\beta\|_2^2}{n} + \lambda \|\beta_j\| \right)$$

Feature Selection

- Wrapper methods are computationally expensive
  - As lambda increases, some parameters will tend to 0
  - Example: Motor Trend US magazine

$$\beta(\lambda) = \underset{\beta}{\operatorname{argmin}} \left(\frac{\|Y - X\beta\|_2^2}{n} + \lambda \|\beta_j\| \right)$$





Engenharia de Características para Aprendizagem Computacional /

Engenharia de Atributos

Feature Engineering– frequency-based