Unit 1
Introduction Statistical Machine Learning

Department of Analytics in the Digital Era
College of Business and Economics
United Arab Emirates University

Outlook of the course

Outlook of the unit

1 **Introduction**

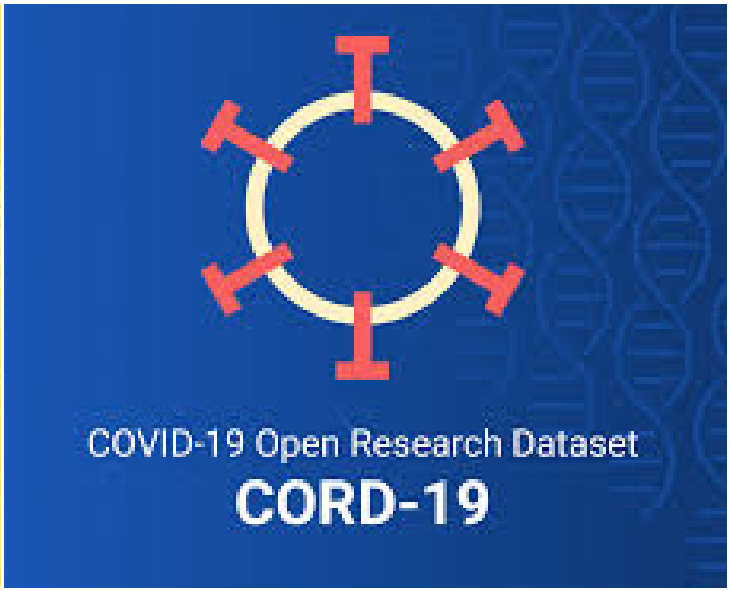
- Introduction to the big data concept
- Real applications of big data
- Introduction to big data analyses
- R-based course
- Data sets in this course:
  - WHO- life expectancy
  - Hitters
  - Boston Housing
  - Decathlon
  - Airline
  - USArrests
  - Minute-Weather

Introduction to the big data concept

What is data?

- **Data** is a set of values from individuals respect to quantitative or qualitative information.
- Data sets are collection of data maintained in an **organized form**.
- Gathering data sets can be accomplished through a primary or a secondary **source**.
- **Data content** might be also audio files, video, geospatial or digital, among others.
- (**Traditional**) data can come from a government census or organization surveys or research studies.

Primary vs. secondary sources



What is Big data?

- In contrast, **big data** sources are not commonly produced as a result of a specific research question.
- There is not an established **minimum** to be categorized as Big data.
- Definition of Big data is not in a certain number of **terabytes** because of the assumption that datasets will increase continuously:
  $\Longrightarrow$ what is called 'big' today might be small in the future.
- A **unique** definition does not exist, instead different definitions may be applicable.

Big data sets (some sources)

- https://www.kaggle.com/
- https://datasetsearch.research.google.com/
- R packages

Big data usage

- Big data sets tend to reveal **patterns** and trends according to a certain aspect.
- Sets that are 'too large' or with 'higher complexity' to be dealt with by **traditional** statistical methods.
- They encompass structured, semi-structured and unstructured **formats**.
- We refer to the Big data **tasks** as the processes of using more sophisticated techniques for data capturing and storage (e.g. Apache Hadoop), data visualization (e.g. SOM maps) and data analysis (e.g. Lasso regression).

Data types

Student ID	Name	Grade
1	Till	77
2	Juan	90
3	Paul	36
4	Hans	65

Structured:

- Properly organized
- Well-defined structure
- Tabular format
- Relational tables

Unstructured:

- No pre-defined structure
- Text, image, sound, video, etc. formats

Semi-structured:

- Combination
-"Apparent pattern"

Characteristics of Big data: The three V's

Volume:
More data is collected and data is less often discarded - Every minute...

Velocity:
Higher speed at which data is generated, stored and processed - Time needed for decoding a human genome.

- in 2003: a decade.
- in 2013: one day.

Variety:
Different types of data, often unstructured and inconsistent - Variety within one source type: E-mail.

- well-structured header: receiver, subject, ...
- text data as body of the mail.
- multi-media data in case of attachments.

New big data V's

Variability:
Constant change

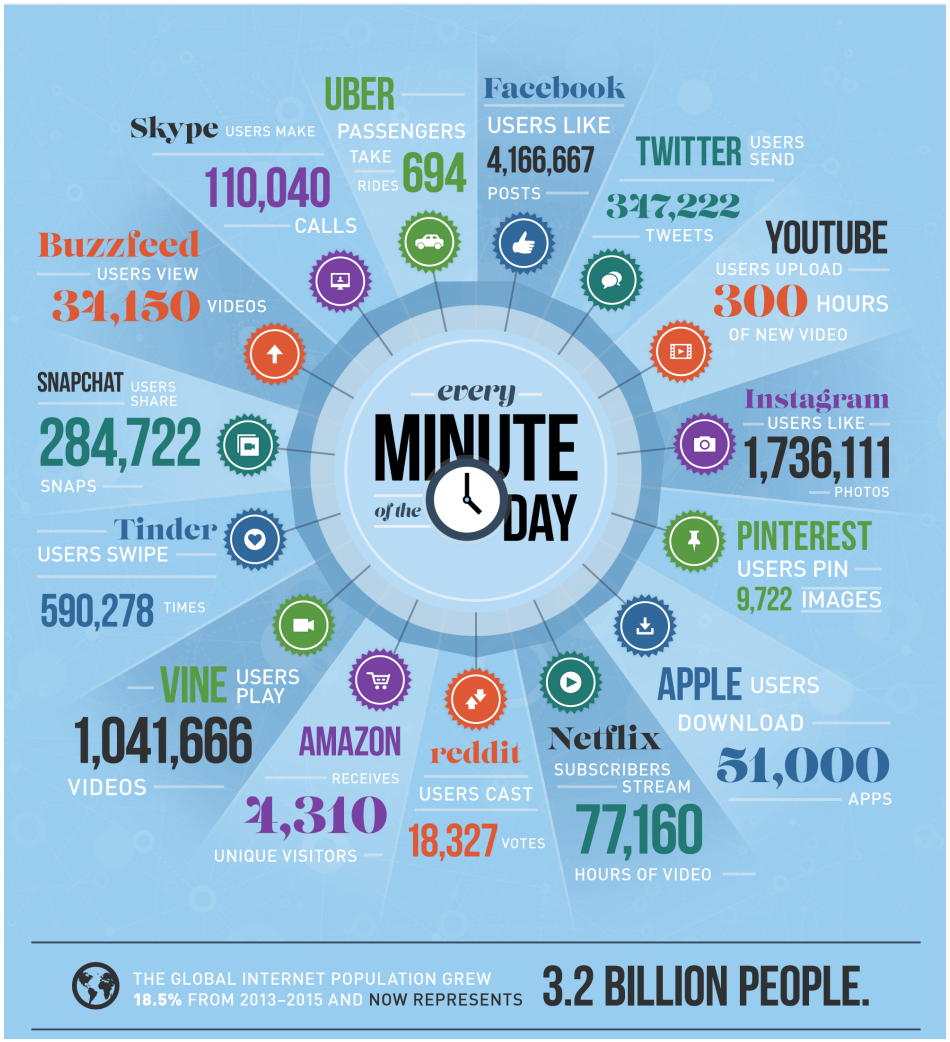
Veracity:
Availability and accuracy

Visualization:
Readability

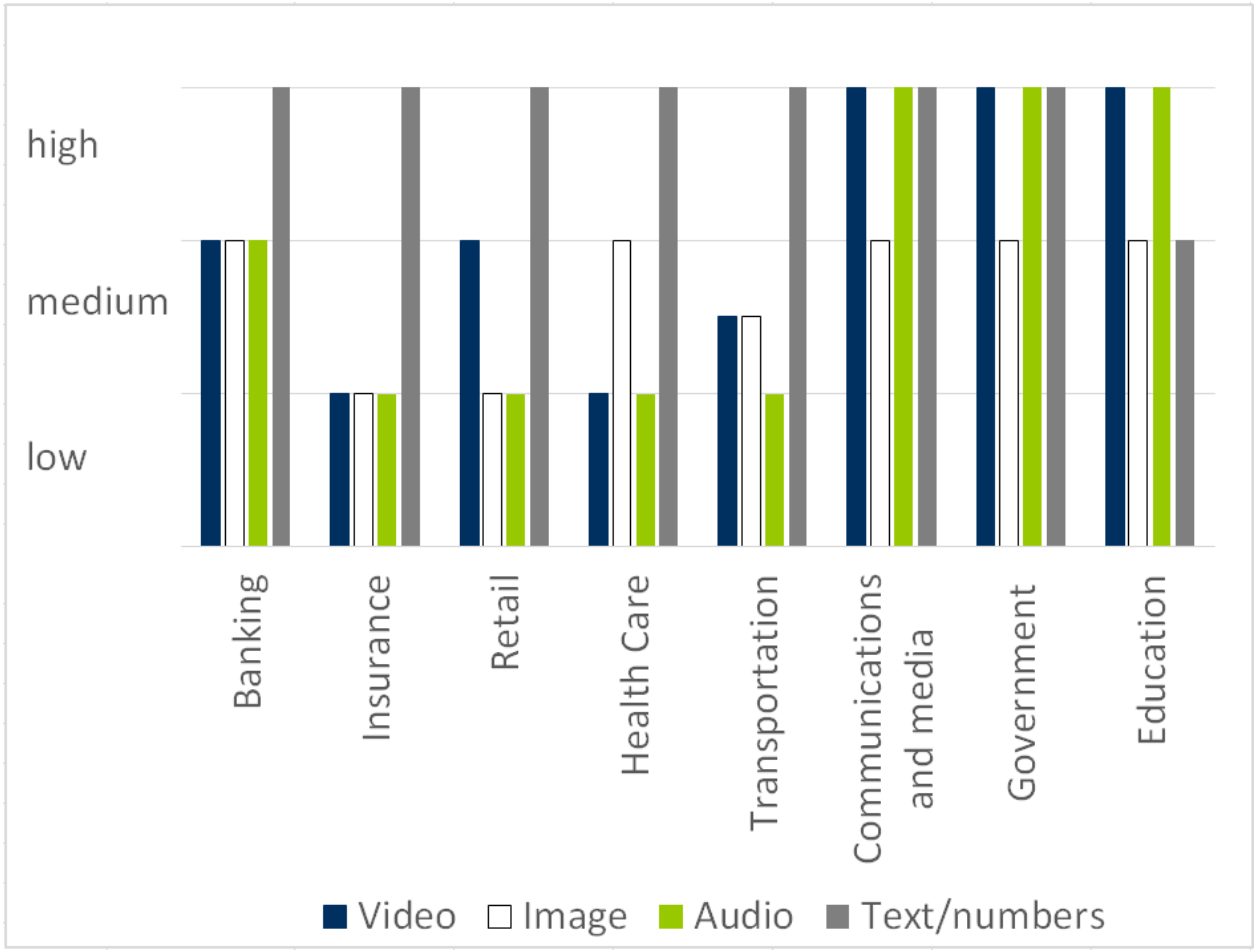
Value
Gartner: Big data is high-volume, high-velocity and/or high variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation.

Real applications of big data

Data never sleeps - actions every minute



Penetration of source types - Important everywhere



Type of data generated and stored by sector

Big data benefits in industries

- **Telecommunications, Media & Entertainment**: 87% of those companies benefit from big data methods. Predicting consumer behavior, development of new products and location-based device analysis, etc.
- **Healthcare**: 60% of those organizations use big data methodologies. Personalized treatments, patient segmentation, predict disease, stock vaccines, etc.
- **Finances, Manufacturing & Banking**: 76% of those service-based companies are using big data techniques. Risk assessment, fraud detection, customer analytics, product quality tracking, etc.

A few examples

- **Subscriber turnover**: Telenor, a Norwegian mobile phone service company, was able to reduce subscriber turnover 37% by using models to predict which customers were most likely to leave, and then lavishing attention on them.
- **Credit scoring**: A credit score is not some arbitrary judgment of credit-worthiness; it is based mainly on a predictive model that uses prior data to predict repayment behavior.
- **Future purchases**: Use of predictive modeling to classify sales prospects as *pregnant* or *not-pregnant*. Those classified as pregnant could then be sent sales promotions at an early stage of pregnancy.
- **Tax evasion**: The US Internal Revenue Service found it was 25 times more likely to find tax evasion when enforcement activity was based on predictive models, allowing agents to focus on the most-likely tax cheats.

Most striking benefits of big data

- **Real-time** information.
- The data is **everywhere** available, in developed and developing countries.
- Data is a by-product of digital conduct and thus **costs** are low (for the data holder).
- Potential to replace or at least complement traditional, costly data sources like **surveys**.
- Potential to improve company's **performance** (value from Big data).

A number of challenges with big data

- **Access to data**: data mostly resides in the private sector, purchase of access to data, data can be a competitive advantage such that sharing is not favourable.
- **Data Policies**: containing privacy, security, property and liability.
- **Technology and techniques**: new types of analysis and new technology for e.g. storage and computing are necessary as well as the human resources.
- **Financial costs**: for all companies or institutions that do not have access to data sources.

Parts in the data science process

Acquire	Prepare	Analyze	Report	Act
Identify data sets	Understand the nature of data	Get to know the prepared data	Communicate results	Apply results
Retrieve data	Clean data	Select analytical technique	Visualize results	
	Structure data	Build model		
	Integrate data if appropriate			

Introduction to big data analyses

General concepts

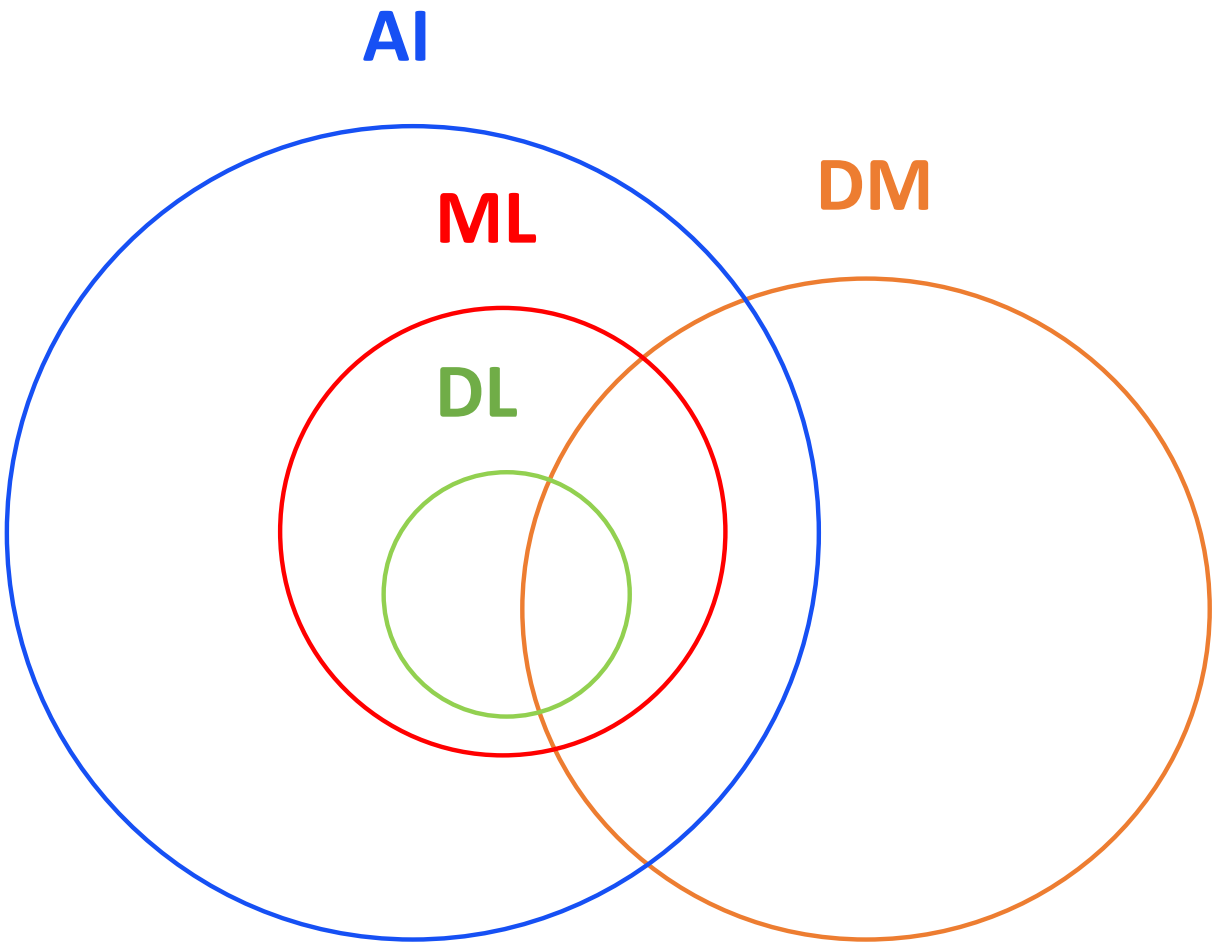
Artificial Intelligence (Wikipedia): It is intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans and animals.

Machine Learning (Wikipedia): It is a sub-field of artificial intelligence. It is the scientific study of algorithms and statistical models that computer systems use to effectively perform a specific task without using explicit instructions, relying on patterns and inference instead.

Deep Learning (Wikipedia): It is part of a family of machine learning methods based on artificial neural networks.

Data Mining (Wikipedia): It is the process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems.

General concepts - graphical representation



Types of statistical machine learning algorithms

Supervised: The algorithm needs that the data scientist acts as a guide to teach the algorithm to which conclusions it should come. It works with explicit inputs and the desired outputs. The y is known and is split into:

- *training* data contain outcomes to train the machine.
- *validation* data are used for select the best performing approach.
- *test* are used for making predictions, which have no outcomes to predict them.

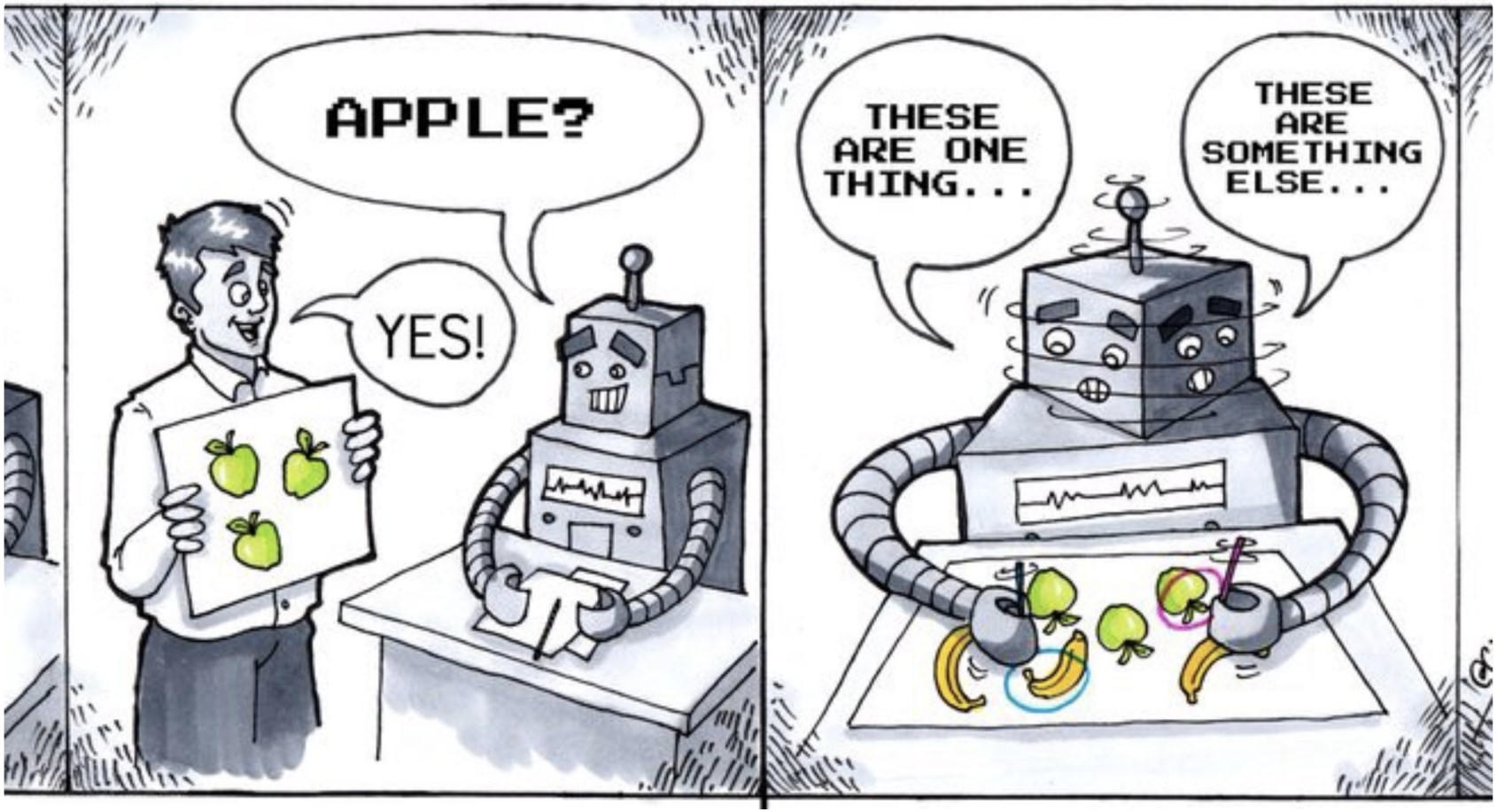
$\Longrightarrow$ Classification, regression models, discriminant analysis, etc.

Unsupervised: The algorithm is able to learn to identify complex structures and patterns of data sets without a data scientist or without using explicitly-provided labels.

- In most of these algorithms there is no a specific way of comparing model performance.

$\Longrightarrow$ Cluster analysis, dimension reduction, association rules, etc.

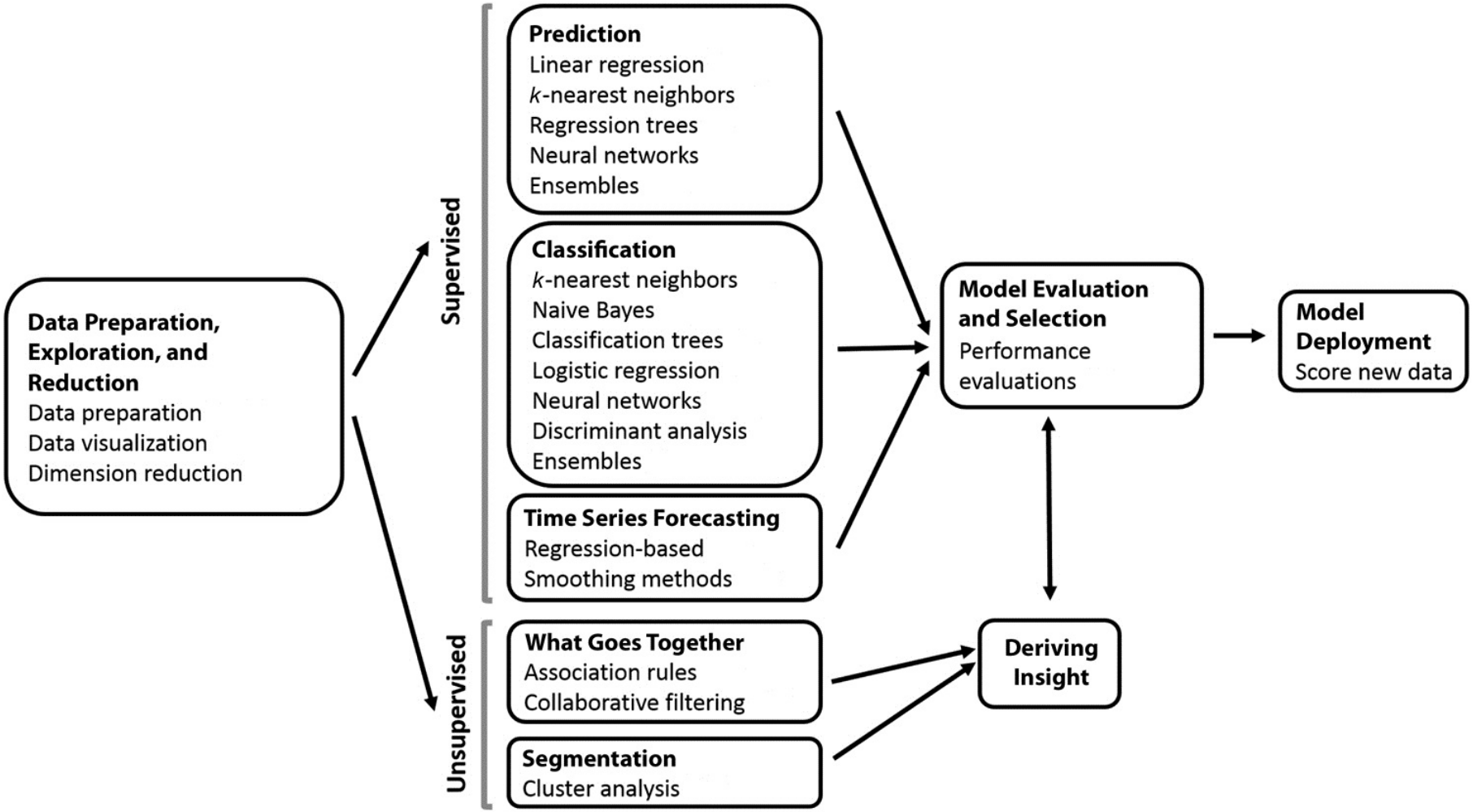
Supervised algorithms



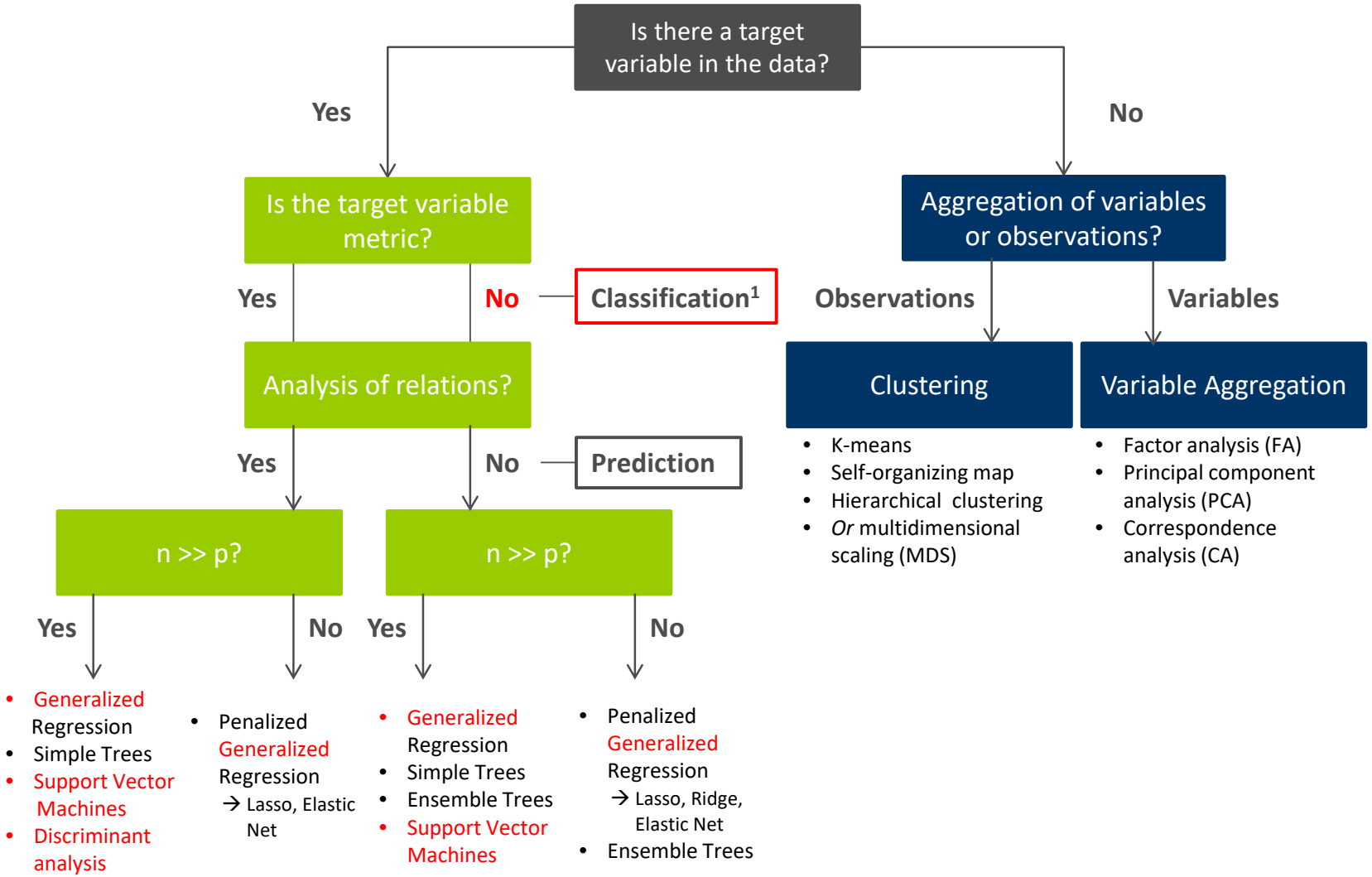
Working with big data sets

1. Develop an understanding of the **purpose** of the data project.
2. **Obtain** the dataset to be used in the analysis.
3. **Understand** the data structure and features by exploring, cleaning, and pre-processing the data (missings, outliers, etc.).
4. Determine the data analysis **task** (dimension reduction, analysis of relations, etc.).
5. **Partition** the data (for supervised tasks) - training, validation, and test data sets.
6. Choose the statistical **techniques** to be used.
7. Use **algorithms** to perform the tasks trying multiple variants.
8. **Interpret** the results of the algorithms.

How to analyze big data sets - A first approach

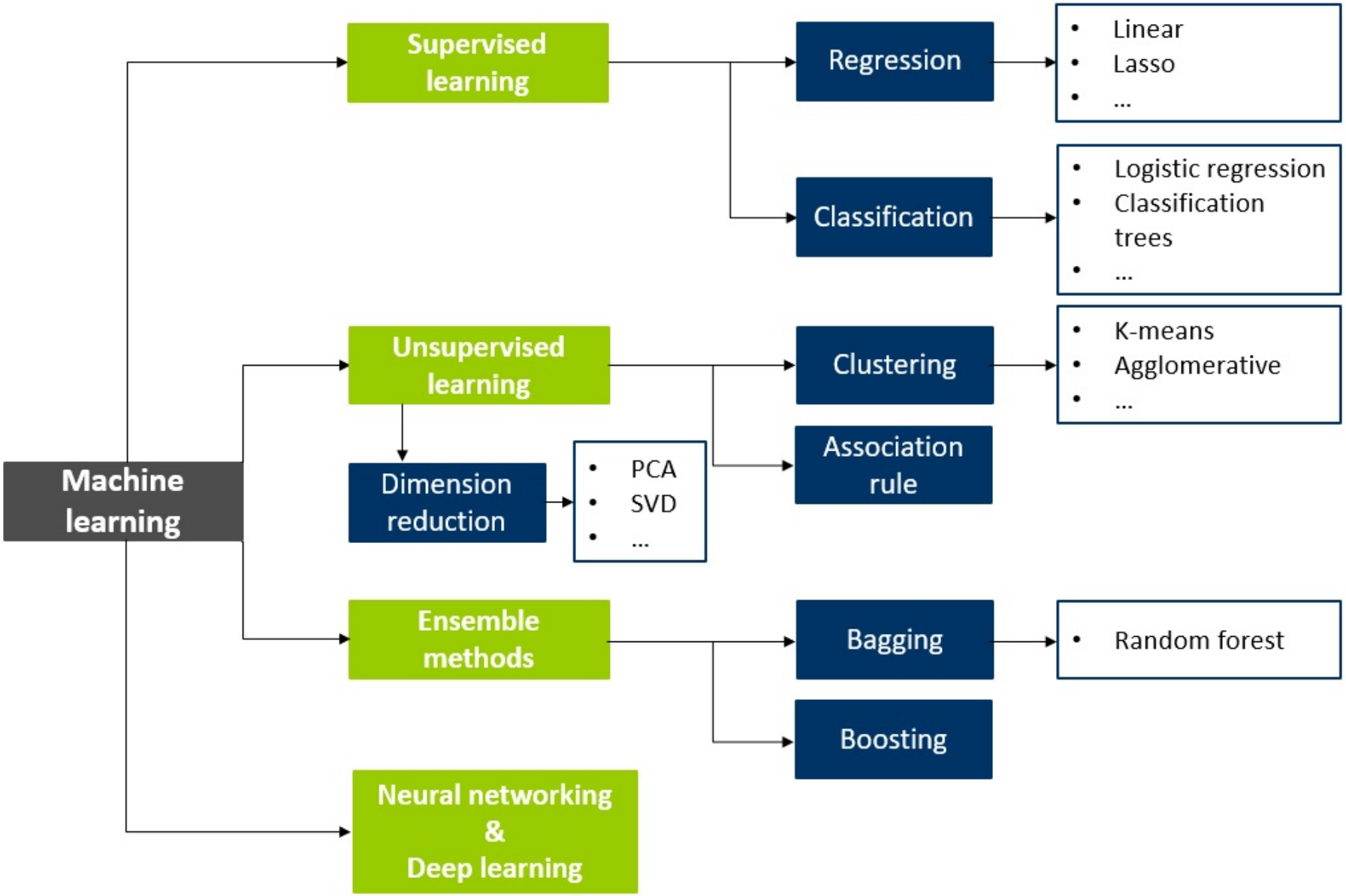


How to analyze big data sets - A second approach



[1] Methods especially for non-metric target variables are marked red

How to analyze big data sets - A third approach



Classification vs. Prediction(regression)

Classification: Examine data where the classification is unknown, with the goal of predicting what that classification is. Similar data where the classification is known are used to develop rules. Predicts categorical class labels. Examples:

- Recipient of an offer can respond or not respond.
- Bus can be available for service or unavailable.

Prediction: is similar to classification, except that we are trying to predict the value of a numerical variable (e.g., amount of purchase) rather than a class (e.g., purchaser or non-purchaser).

Predictive analytics: classification, prediction, and to some extent further methods constitute the analytical methods employed in predictive analytics. The term predictive analytics is also used to include data pattern identification methods such as clustering.

Classification vs. Prediction(regression) - graphical representation

Classifier

To predict the class of objects whose class label is unknown

Buy | Does not buy

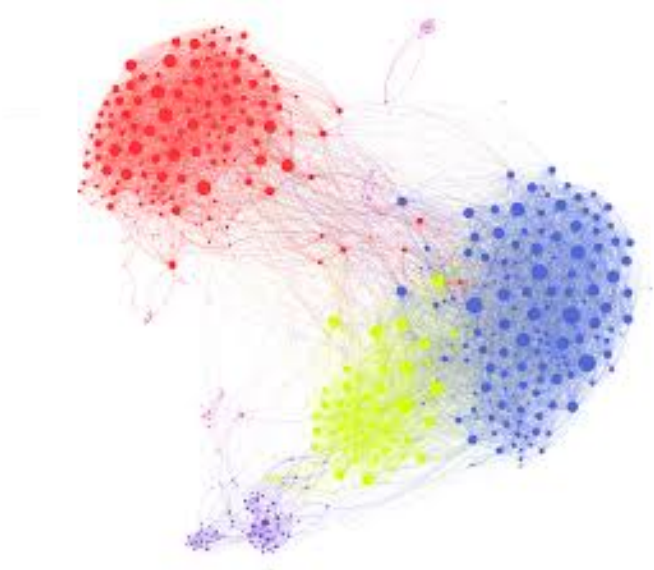
Predictive model

To predict unknown values of a numerical variable

1000 AED (amount of purchase)

Most used methods

Association rules, clustering, classification(prediction)



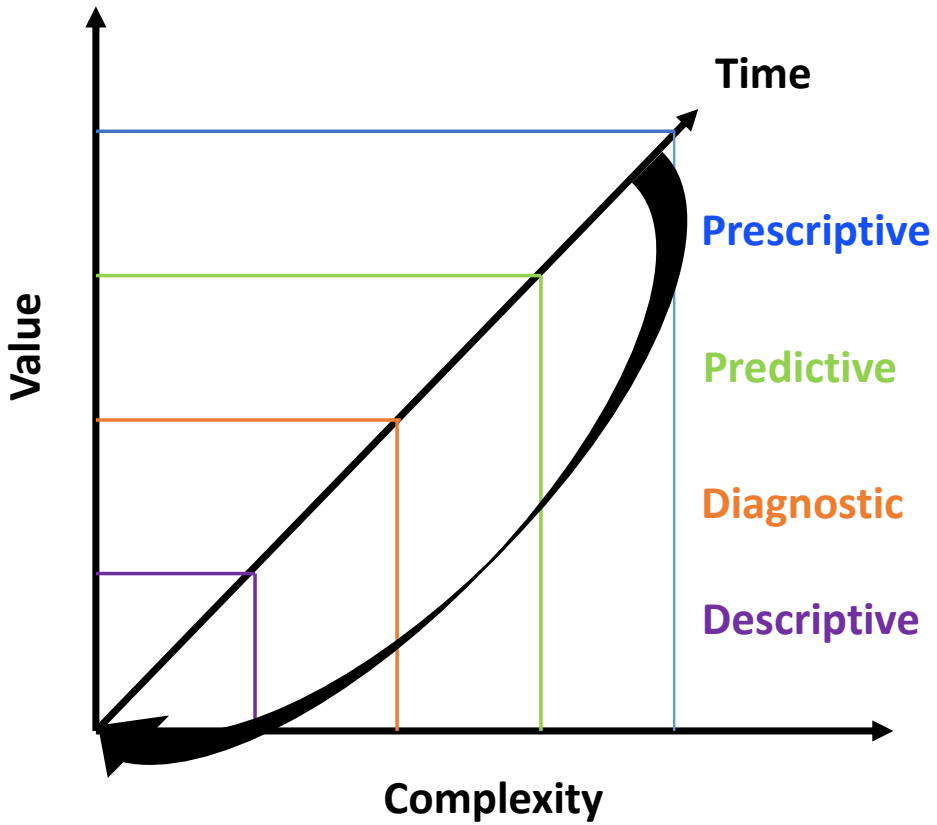
Data reduction vs. Dimension reduction

Data reduction: The process of consolidating a large number of records (or observations) into a smaller set is termed data reduction. Methods for reducing the number of cases are often called clustering. Predicts numerical values. Examples:

- Rather than dealing with thousands of product types, an analyst might wish to group them into a smaller number of groups.
- A marketer might want to classify customers into different groups.

Dimension reduction: is reducing the number of variables. Dimension reduction is a common initial step before deploying data methods, intended to improve predictive power, manageability, and interpretability. Typical example is principal component analysis (PCA).

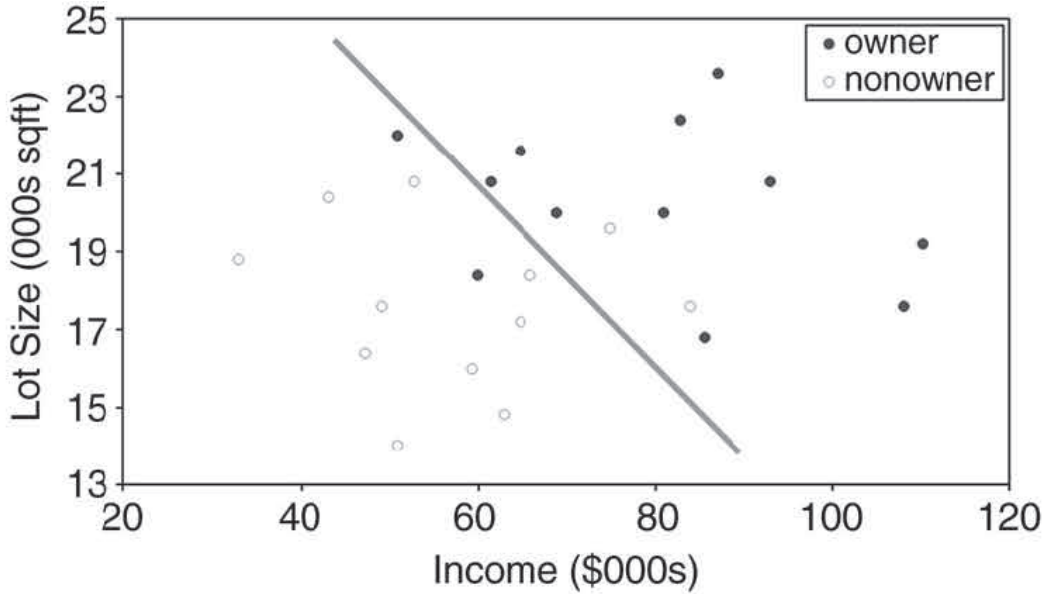
Complexity of big data methods



Why are there so many different methods?

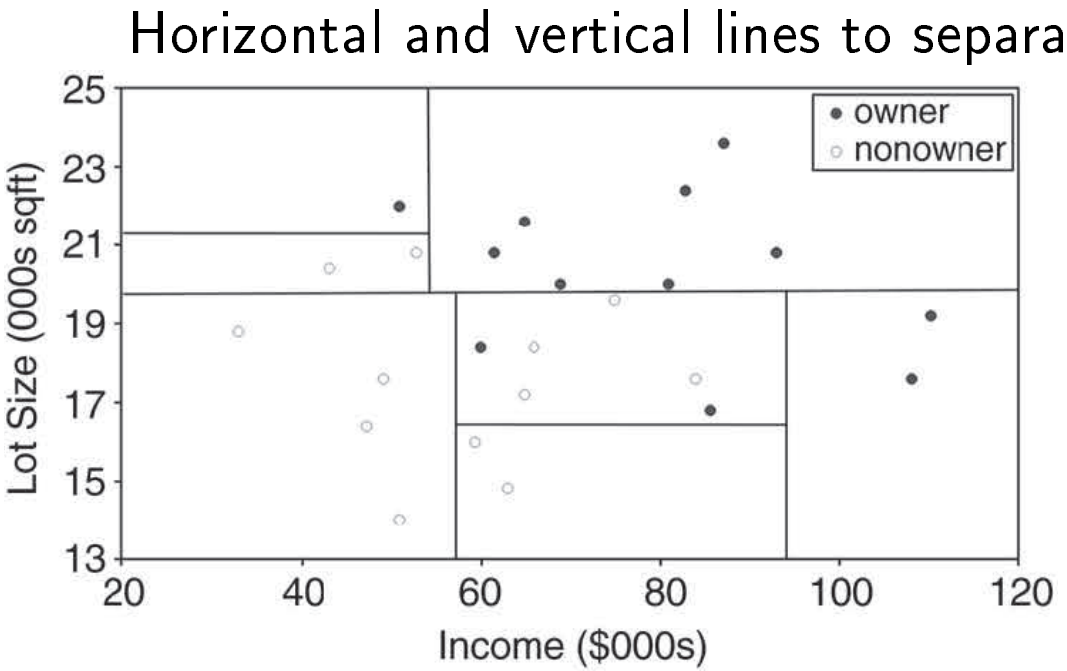
- Each method has advantages and disadvantages.
- Usefulness of a method depend on factors like size of the data set, the types of patterns that exist in the data, how noisy the data are, etc.

Example: Goal is to find a combination of household income level and household lot size that separates buyers and non-buyers.



Single diagonal line to seperate

Why are there so many different methods?



- Different methods can lead to different results, and their performance can vary.
- It is customary to apply several different methods and select the one that appears most useful for the goal at hand.

R-based course

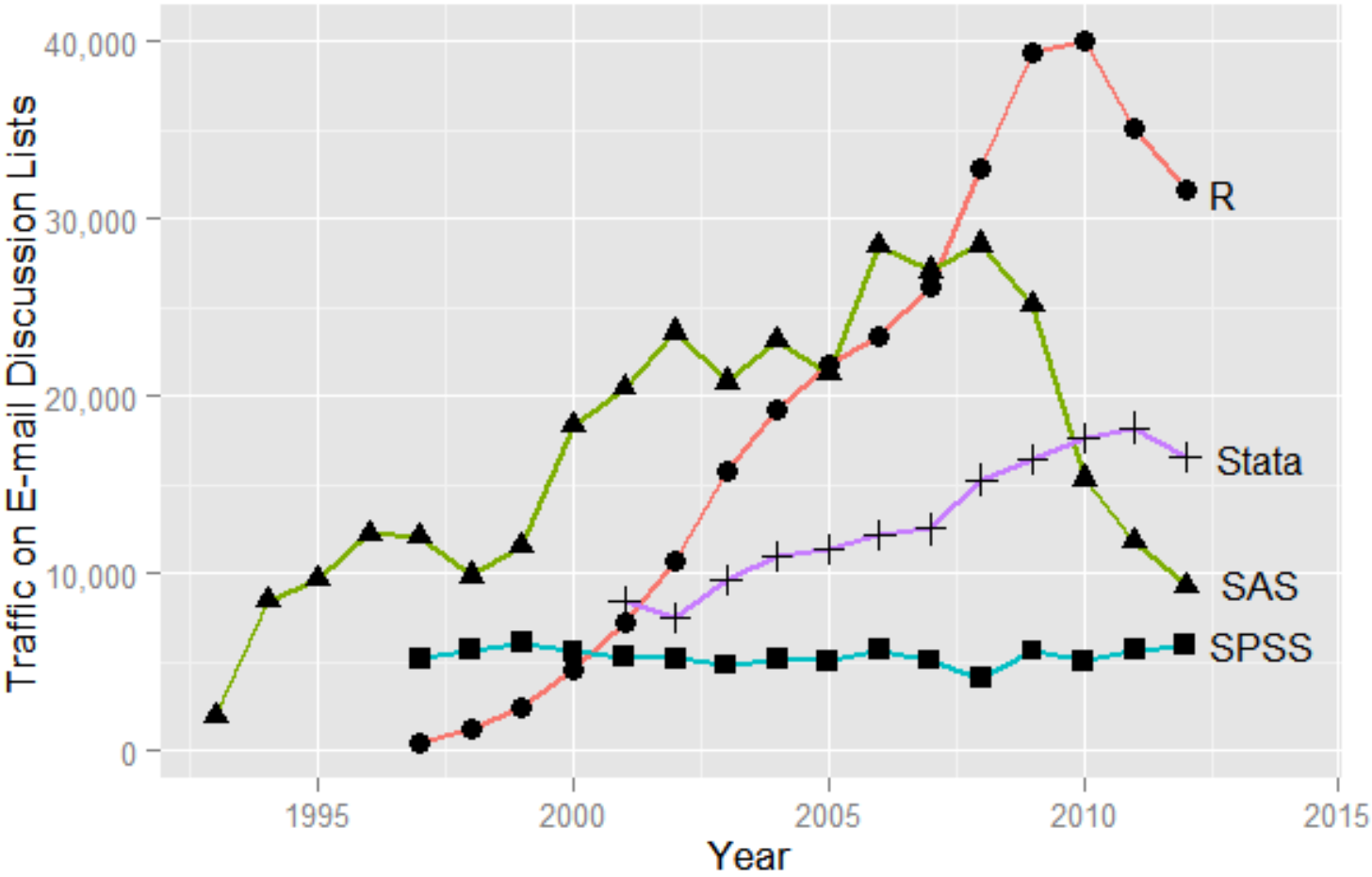
Why you should learn R for data science?

Some reasons:

- R is becoming the *lingua franca* data science - the most widely used software and it is rising in **popularity**.
- R is in heavy use at several of the best companies who are hiring data scientists - also very popular among academic **scientists**.
- It's often said that 80% of the work in data science is *data manipulation* - R has some of the best data **management** tools.
- R is one of the best *data visualization* tools around, as of 2015.
- When you are ready to start using (and learning) *statistical methods*, R has some of the best tools and resources.

Why you should learn R for data science?

A vital user community with many open source packages.



More analytical tools

Data sets in this course

Life Expectancy (WHO) data set

The *Life Expectancy (WHO)* data set

- This data set contains life expectancy at birth (years) and 20 related variables from year 2000 to 2015 for 193 countries. It is collected by the World Health Organization (WHO) and United Nations.
- Several numerical variables are included. They are related to immunization, mortality, economical, and social factors.

Possible tasks:

- To find predicting variables for the life expectancy.
- A supervised task, where the goal is to explain numerical and categorical variables
  ($\Longrightarrow$ Regression).

Some variables in the WHO data set

Variable	Name
Country	`Country`
Year	`Year`
Life Expectancy in age	`Life.expectancy`
Adult Mortality Rates of both sexes	`Adult.Mortality`
Number of Infant Deaths per 1000 population	`infant.deaths`
Gross Domestic Product per capita	`GDP`
Developed or Developing status	`Status`
Alcohol, recorded per capita (15+) consumption	`Alcohol`
Hepatitis B immunization coverage	`Hepatitis.B`
Measles - number of reported cases	`Measles`
Population of the country	`Population`
Number of years of Schooling	`Schooling`

WHO data set: Overview

The `str()`-command displays the internal structure of an R object.

```
    > str(Daten)
'data.frame': 2938 obs. of  22 variables:
 $ Country                        : chr  "Afghanistan"
 $ Year                           : int  2015 2014 ...
 $ Status                         : chr  "Developing"
 $ Life.expectancy                : num  65 59.9 ...
 $ Adult.Mortality                : int  263 271 ...
 $ infant.deaths                  : int  62 64 66 ...
 $ Alcohol                        : num  0.01 0.01 ...
 $ percentage.expenditure         : num  71.3 73.5 ...
 $ Hepatitis.B                    : int  65 62 64 67...
 $ Measles                        : int  1154 492 430...
 $ BMI                            : num  19.1 18.6 ...
 $ under.five.deaths              : int  83 86 89 ...
 $ Polio                          : int  6 58 62 67...
 $ Total.expenditure              : num  8.16 8.18 ...
```

Hitters data set

The *Hitters* data set

- This data set contains the numbers of errors, putouts and assists made by 322 major league players from the seasons of 1986 and 1987.
- 20 numerical variables are included.

Possible tasks:

- To predict baseball players' salaries.
- A supervised task, such as tree-based methods for regression and classification
  ($\Longrightarrow$ Classification and regression trees).

Some variables in the Hitters data set

Variable	Name
Number of times at bat	`AtBat`
Number of hits	`Hits`
Number of home runs	`HmRun`
Number of runs	`Runs`
Number of runs batted	`RBI`
Number of walks	`Walks`
Number of years in the major leagues	`Years`
Number of times at bat during his career	`CAtBat`
Number of hits during his career	`CHits`
Annual salary on opening day	`Salary`
Number of put outs	`PutOuts`
Number of assists	`Assists`
Number of errors	`Errors`

Hitters data set: Overview

The `head(#)`-command displays the variable values of the first # elements.

```
# Loading libraries and the data

library(tree)
library(ISLR)
data("Hitters")

# Check values of the first element

> head(Hitters,1)
                  AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits
-Andy Allanson      293   66     1   30  29    14     1    293    66
                  CHmRun CRuns CRBI CWalks League Division PutOuts
-Andy Allanson         1    30   29     14      A        E     446
                  Assists Errors Salary NewLeague
-Andy Allanson         33     20     NA         A
```

Decathlon data set

The *decathlon* data set

- This data set contains the results of decathlon events during two athletic meetings in 2004: Olympic Games in Athens (August) and the Decastar in France (September).
- 13 variables for each athlete (27 men): performance for each of the 10 events, total number of points, final ranking and identifier for the meeting.
- Order of the events: 100 metres, long jump, shot put, high jump, 400 metres (first day) and 110 metre hurdles, discus, pole vault, javelin, 1500 metres (second day).

Possible tasks:

- An unsupervised task, where the goal is to combine variables ($\Longrightarrow$ PCA).

Variables in the decathlon data set

Variable	Name
100-meter sprint in seconds	`X100m`
Long jump in metres	`Long.jump`
Shot put in metres	`Shot.put`
High jump in metres	`High.jump`
400-meter race in seconds	`X400m`
110-meter hurdles in seconds	`X110m.hurdle`
Discus in meters	`Discus`
Pole vault in metres	`Pole.vault`
Javelin in metres	`Javeline`
1,500-metre run in seconds	`X1500m`
Final ranking	`Rank`
Total number of points	`Points`
Identifier for the meeting (Decastar/Olympics)	`Competition`

The decathlon data set: Overview

The `head()`-command returns the first parts of a vector, matrix, table, data frame or function.

```
# Loading libraries and the data
library(FactoMineR)
library(factoextra)
data(decathlon2)

# Additional information regarding
> head(decathlon2)
           X100m X400m Discus Javeline X1500m Rank Points
SEBRLE     11.04 49.81  43.75    63.19  291.7    1   8217
CLAY       10.76 49.37  50.72    60.15  301.5    2   8122
BERNARD    11.02 48.93  40.87    62.77  280.1    4   8067
YURKOV     11.34 50.42  46.26    63.44  276.4    5   8036
ZSIVOCZKY 11.13 48.62  45.67    55.37  268.0    7   8004
McMULLEN   10.83 49.91  44.41    56.37  285.1    8   7995
```

The decathlon data set: Overview

The `str()`-command displays the internal structure of an R object.

```
# Additional information regarding
> str(decathlon2)
'data.frame': 27 obs. of  13 variables:
 $ X100m         : num  11 10.8 11 11.3 11.1 ...
 $ Long.jump     : num  7.58 7.4 7.23 7.09 7.3 7.31 ...
 $ Shot.put      : num  14.8 14.3 14.2 15.2 13.5 ...
 $ High.jump     : num  2.07 1.86 1.92 2.1 2.01 2.13 ...
 $ X400m         : num  49.8 49.4 48.9 50.4 48.6 ...
 $ X110m.hurdle: num  14.7 14.1 15 15.3 14.2 ...
 $ Discus        : num  43.8 50.7 40.9 46.3 45.7 ...
 $ Pole.vault    : num  5.02 4.92 5.32 4.72 4.42 ...
 $ Javeline      : num  63.2 60.1 62.8 63.4 55.4 ...
 $ X1500m        : num  292 302 280 276 268 ...
 $ Rank          : int  1 2 4 5 7 8 9 10 11 12 ...
 $ Points        : int  8217 8122 8067 8036 8004 ...
 $ Competition : Factor w/ 2 levels "Decastar","OlympicG"
```

name	100m	Long.jump	//	Javeline	1500m	Rank	Points	Competition
SEBRLE	11.04	7.58		63.19	291.7	1	8217	Decastar
CLAY	10.76	7.4		60.15	301.5	2	8122	Decastar
Macey	10.89	7.47		58.46	265.42	4	8414	OlympicG
Warners	10.62	7.74		55.39	278.05	5	8343	OlympicG
\\								
Zsivoczky	10.91	7.14		63.45	269.54	6	8287	OlympicG
Hernu	10.97	7.19		57.76	264.35	7	8237	OlympicG
Pogorelov	10.95	7.31		53.45	287.63	11	8084	OlympicG
Schoenbeck	10.9	7.3		60.89	278.82	12	8077	OlympicG
Barras	11.14	6.99		64.55	267.09	13	8067	OlympicG
KARPOV	11.02	7.3		50.31	300.2	3	8099	Decastar
WARNERS	11.11	7.6		51.77	278.1	6	8030	Decastar
Nool	10.8	7.53		61.33	276.33	8	8235	OlympicG
Drews	10.87	7.38		51.53	274.21	19	7926	OlympicG

Active individuals
Active variables
Supplementary **quantitative** variables
Supplementary **qualitative** variable
Supplementary individuals

Working data.

The decathlon data set: Overview

The `summary()`-command gives some descriptive statistics.

```
> apply(decathlon2.active,2,summary)
        X100m Long.jump Shot.put High.jump X400m
Min.    10.44      6.80    12.68      1.86 46.81
1st Qu. 10.84      7.21    14.17      1.93 48.70
Median  10.97      7.31    14.57      1.98 49.20
Mean    10.99      7.36    14.54      2.00 49.31
3rd Qu. 11.14      7.54    15.00      2.08 49.86
Max.    11.64      7.96    16.36      2.15 51.16

        X110m.hurdle Discus Pole.vault Javeline X1500m
Min.           13.97  37.92       4.40    50.31 262.10
1st Qu.        14.15  42.27       4.66    55.32 271.55
Median         14.34  44.72       4.90    57.19 278.10
Mean           14.50  44.85       4.84    58.32 278.52
3rd Qu.        14.87  46.93       5.00    62.05 283.55
Max.           15.67  51.65       5.40    70.52 301.50
```

Boston Housing data set

Boston Housing data set

- The Boston Housing data contain information on census tracts in suburbs of Boston.
- Several measurements are included (e.g., crime rate, pupil-teacher ratio).
- 14 variables for each of the 506 houses.

Possible tasks:

- A supervised predictive task, where the outcome is the median value of a home.
- A supervised classification task, where the outcome is the binary variable *CAT.MEDV* that indicates whether the home value is above or below $\$30,000$.
- An unsupervised task, where the goal is to cluster houses.

Variables in the Boston housing data set

Variable	Name
Crime rate	CRIM
Percentage of residential land zoned for lots over 25,000 ft^2	ZN
Percentage of land occupied by nonretail business	INDUS
Does tract bound Charles River (= 1 if tract bounds river)	CHAS
Nitric oxide concentration (parts per 10 million)	NOX
Average number of rooms per dwelling	RM
Percentage of owner-occupied units built prior to 1940	AGE
Weighted distances to five Boston employment centers	DIS
Index of accessibility to radial highways	RAD
Full-value property tax rate per $10,000	TAX
Pupil-to-teacher ratio by town	PTRATIO
Percentage of lower status of the population	LSTAT
Median value of owner-occupied homes in $1000s	MEDV
Is median value of owner-occupied homes in tract above $30,000 (CAT.MEDV = 1) or not (CAT.MEDV = 0)	CAT.MEDV

Boston housing data set: Overview

The `head()`-command returns the first parts of a vector, matrix, table, data frame or function.

```
> head(Daten)
  CRIM ZN INDUS   RM  AGE   DIS RAD TAX LSTAT MEDV CMEDV
0.006 18  2.31 6.57 65.2 4.090   1 296  4.98 24.0     0
0.027  0  7.07 6.42 78.9 4.967   2 242  9.14 21.6     0
0.027  0  7.07 7.18 61.1 4.967   2 242  4.03 34.7     1
0.032  0  2.18 6.99 45.8 6.062   3 222  2.94 33.4     1
0.069  0  2.18 7.14 54.2 6.062   3 222  5.33 36.2     1
0.029  0  2.18 6.43 58.7 6.062   3 222  5.21 28.7     0
```

Boston housing data set: Overview

The `str()`-command displays the internal structure of an R object.

```
> str(Daten)
'data.frame': 506 obs. of  14 variables:
 $ CRIM     : num  0.00632 0.02731 0.02729 0.03237 ...
 $ ZN       : num  18 0 0 0 0 0 12.5 12.5 12.5 12.5 ...
 $ INDUS    : num  2.31 7.07 7.07 2.18 2.18 2.18 7.87 ...
 $ CHAS     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ NOX      : num  0.538 0.469 0.469 0.458 0.458 ...
 $ RM       : num  6.58 6.42 7.18 7 7.15 ...
 $ AGE      : num  65.2 78.9 61.1 66.6 96.1 100 85.9 ...
 $ DIS      : num  4.09 4.97 4.97 6.06 6.06 ...
 $ RAD      : int  1 2 2 3 3 3 5 5 5 5 ...
 $ TAX      : int  296 242 242 311 311 311 ...
 $ PTRATIO  : num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 ...
 $ LSTAT    : num  4.98 9.14 4.03 2.94 5.33 ...
 $ MEDV     : num  24 21.6 34.7 33.4 36.2 28.7  ...
 $ CAT..MEDV: int  0 0 1 1 1 0 0 0 0 0 ...
```

Airline data set

Airline data set

- Contains information on all commercial US Domestic Flights between 1987 and 2008 (1.6 gigabytes compressed, 12 gigabytes uncompressed)
- Over those 22 years there are over 123 million observations of 29 variables
- The data set, that was used in the 2009 American Statistical Association challenge is comprised of variables of different types: most of the variables are metric (discrete or continuous) valued like Distance or ArrDelay (Arrival Delay), then there are nominal variables like Diverted (Yes or No?) and furthermore factors (nominal with many levels) like Origin or Dest (Destination)
- In the context of this course the year 2008 will be used for demonstration purposes, containing around 7 million observations
- In the year 2008 there are 28.5 million observations missing in total (mostly reasons for flight delays)

Variables in the Airline data set

Variable group	Variables
Dates	Year (1987-2008), Month (1-12), DayofMonth (1-31), DayOfWeek (1-7)
Arrival/Departure times	actual: DepTime, ArrTime, scheduled: CRSDepTime, CRSArrTime
Flight determinants	UniqueCarrier (carrier code), FlightNum, TailNum (plane tail number)
Flight time (in minutes)	ActualElapsedTime, CRSElapsedTime, AirTime
Origin/Destination	IATA airport code: Origin, Dest
Distance (in miles)	Distance
Delay (in minutes)	ArrDelay, DepDelay, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay, LateAircraftDelay
Cancellation	Cancelled (yes=1/no=0), CancellationCode (A = carrier, B = weather, C = NAS, D = security)
Diversion	Diverted (yes=1/no=0)
	TaxiIn, TaxiOut

Airline data set: Overview

Let's check the structure of this data set:

```
> str(flights.small)
'data.frame': 9787 obs. of  18 variables:
$ Month             : int  12 1 4 7 5 5 9 12 6 10 ...
$ DayofMonth        : int  3 14 22 31 4 1 5 26 24 8 ...
$ DayOfWeek         : int  3 1 2 4 7 4 5 5 2 3 ...
...
$ AirTime           : int  42 113 40 47 59 63 111 134 61
   36 ...
$ ArrDelay          : int  3 -1 9 -5 0 -15 -2 11 -2 5 ...
$ DepDelay          : int  -6 10 12 2 -2 -9 11 -4 2 -4 ...
$ Distance          : int  201 869 214 258 402 376 775
   1036 358 116 ...
$ TaxiIn            : int  14 5 2 3 5 8 4 19 4 8 ...
$ TaxiOut           : int  29 6 30 13 18 10 7 12 6 15 ...
```

USArrests data set

The *USArrests* data set

- This data set contains statistics about violent crime rates in each of the 50 US states in 1973. Those include arrests per 100,000 residents for assault, murder, and rape. The percent of the population living in urban areas is also given.
- 250 observations on 4 numerical variables are included.

Possible tasks:

- To classify states based on their criminal conditions.
- An unsupervised task for finding similar states ($\Longrightarrow$ Clustering).

Variables in the USArrests data set

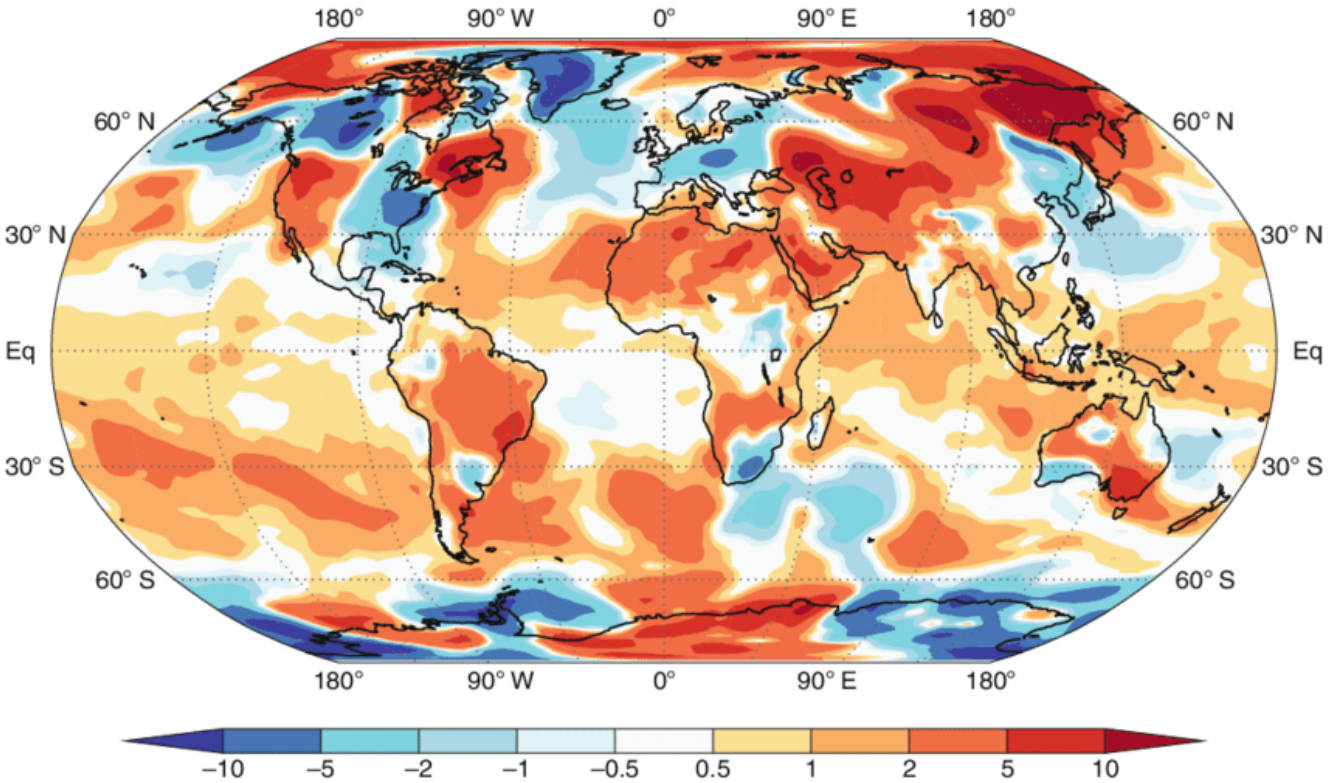
Variable	Name
Murder arrests (per 100,000)	`Murder`
Assault arrests (per 100,000)	`Assault`
Percent urban population	`UrbanPop`
Rape arrests (per 100,000)	`Rape`

USArrests data set: Overview

The `summary()`-command shows some basic descriptive statistics.

```
  # Check basic descriptive statistics of the variables
> summary(USArrests)
     Murder          Assault         UrbanPop
         Rape
 Min.   : 0.800   Min.   : 45.0   Min.   :32.00   Min.
    : 7.30
 1st Qu.: 4.075   1st Qu.:109.0   1st Qu.:54.50   1st Qu
    .:15.07
 Median : 7.250   Median :159.0   Median :66.00   Median
    :20.10
 Mean   : 7.788   Mean   :170.8   Mean   :65.54   Mean
    :21.23
 3rd Qu.:11.250   3rd Qu.:249.0   3rd Qu.:77.75   3rd Qu
    .:26.18
 Max.   :17.400   Max.   :337.0   Max.   :91.00   Max.
    :46.00
```

Minute-weather data set



The *Minute-weather* data set

- This data set contains weather-related measurements such as air pressure, maximum wind speed, relative humidity etc. This was captured in San Diego, over a three-year period from September 2011 to September 2014 and contains raw sensor measurements captured at one-minute intervals.
- It has 1,587,257 measurements and 13 variables.

Possible tasks:

- An unsupervised task for finding similar elements or supervised methods to predict or classify them
  ($\Longrightarrow$ Clustering, Regression, Classification).

Variables in the Minute-weather data set

Variable	Name
Timestamp of measure	`hpwren_timestamp`
Air pressure measured at the timestamp	`air_pressure`
Air temperature measure at the timestamp	`air_temp`
Wind direction averaged	`avg_wind_direction`
Wind speed averaged	`avg_wind_speed`
Highest wind direction	`max_wind_direction`
Highest wind speed	`max_wind_speed`
Smallest wind direction	`min_wind_direction`
Smallest wind speed	`min_wind_speed`
Amount of accumulated rain	`rain_accumulation`
Length of time rain	`rain_duration`
Relative humidity	`relative_humidity`

Minute-weather data set: Overview

The `str()`-command shows the structure of the data set.

```
> str(Daten)
'data.frame': 1587257 obs. of  13 variables:
 $ rowID                : int  0 1 2 3 4 5 6 7 8 9 ...
 $ hpwren_timestamp     : chr  "2011-09-10 00:00:49" ...
 $ air_pressure         : num  912 912 912 912 912 ...
 $ air_temp             : num  64.8 63.9 64.2 64.4 ...
 $ avg_wind_direction: num  97 161 77 89 185 ...
 $ avg_wind_speed       : num  1.2 0.8 0.7 1.2 0.4 ...
 $ max_wind_direction: num  106 215 143 112 ...
 $ max_wind_speed       : num  1.6 1.5 1.2 1.6 1 3 ...
 $ min_wind_direction: num  85 43 324 12 100  ...
 $ min_wind_speed       : num  1 0.2 0.3 0.7 0.1 2 ...
 $ rain_accumulation : num  NA 0 0 0 0 0 0 0 0 ...
 $ rain_duration        : num  NA 0 0 0 0 0 0 0 0 0 ...
 $ relative_humidity : num  60.5 39.9 43 49.5 ...
```