



Using big data analytics to support decision-making

Authors: Fabio Mercurio, Lorenzo Malandri and Mario Mezzanica (Interuniversity Research Centre on Public Services (CRISP), University of Milano-Bicocca)

Research managers: Eleonora Peruffo and Gijs van Houten

Eurofound reference number: WPEF21051

Related reports: [Using big data to improve survey sampling](#) and [Using social media and text analytics to investigate marginal phenomena](#)

© European Foundation for the Improvement of Living and Working Conditions (Eurofound), 2022
Reproduction is authorised provided the source is acknowledged.

For any use or reproduction of photos or other material that is not under the Eurofound copyright, permission must be sought directly from the copyright holders.

Any queries on copyright must be addressed in writing to: copyright@eurofound.europa.eu

The European Foundation for the Improvement of Living and Working Conditions (Eurofound) is a tripartite European Union Agency established in 1975. Its role is to provide knowledge in the area of social, employment and work-related policies according to Regulation (EU) 2019/127.

European Foundation for the Improvement of Living and Working Conditions

Telephone: (+353 1) 204 31 00

Email: information@eurofound.europa.eu

Web: www.eurofound.europa.eu

Contents

Executive summary	1
Guide to readers: What do the badges next to chapter titles mean?	2
Chapter 1 – Introduction to Big Data and Analytics.....	3
Data Challenges: the 5 Vs of Big Data.....	6
Process Challenge: data, information, knowledge	8
Chapter 2 – Data Selection and Collection	12
Source Identification	12
Ranking Model.....	14
Engagement and Collection Criteria.....	16
Chapter 3 – Pre-processing and Transformation	19
Pre-processing: Data Quality and Cleaning	20
Data Store: NoSQL in a nutshell	22
Chapter 4 – Data Mining	26
Machine Learning	27
Supervised machine learning	27
Types of supervised machine learning algorithms	29
Evaluation of classification models.....	30
An example of classification algorithm.....	32
Complexity of the model and overfitting	33
Unsupervised Machine Learning	35
Evaluation of Unsupervised Machine Learning	35
Supervised and Unsupervised: A Visual Example.....	35
Deep Learning.....	36
Explainable AI (XAI).....	38
Chapter 5 – Interpretation and Evaluation through examples	40
Data visualisation	40
Data visualisation types.....	40
Data visualisation tools.....	42
Use cases	43
Chapter 6 – Enabling BDA to scale.....	49
Recommendations for conducting a fully-fledged Big Data Analytics project.....	54
R1: Identify the right skills and competences to realise a fully-fledged Big Data Analytics project.....	54
R2: Manage the trade-off between performance and explainability in Machine Learning.	55
R3: Five golden rules for implementing a BDA project on Cloud	56
R4: How to bring BDA into official statistics: understanding the value of smart statistics	58
R5: Address issues related to the representativeness of Web Data	59
References	60

Executive summary

This working paper was produced as part of an [exercise](#) to explore the potential of big data for Eurofound research. The paper provides an overview of the key concepts and main processes used in analysing 'big data' (large amounts of data, mostly online), and can serve as a guide to support planning, conducting or commissioning research of such nature.

In recent years, we have witnessed a massive burst in the availability of online data. These data are generated from various sources, often without control by the individuals or organisations using online services, and the data come in different forms, both structured and unstructured (for example, text, and images). This has two main implications for the way data are managed and analysed. On the one hand, there has been a paradigm shift in the way of reasoning, moving from a top-down, deductive approach – that, starting from one or more hypotheses, collects and analyses data to verify or falsify them – towards a bottom-up inductive approach, which aims to discover hidden patterns to support the decision-making process in social research and in policy making. This data driven approach allows one to better exploit the hidden informative power of big data, distilling relevant insights and, eventually, reconciling the big data with administrative data or official statistics (such combination is also referred to as *smart statistics*) to infer claims about the entire observed population. On the other hand, those enormous and diverse flows of data still need new data management techniques to cope with their complexity.

The paper begins with an introduction (Ch. 1) to the key features, or dimensions, of big data – volume, velocity, variety, veracity, and value - and the main challenges in addressing them. The following chapters (Ch. 2-6) cover the processes as well as types of tools used in Big Data Analytics, starting from the selection, collection, and management of data down to the transformation of the resulting knowledge into business actions. The paper discusses how the data driven AI and algorithms in the big data analytics are used in data mining as well as interpretation. To make the information reader-friendly, there are marked sections which cover methodological aspects (why and what to analyse), techniques (discussing how methodologies work and referring to state-of-the-art solutions), and examples (to explain the approach or to illustrate real cases of big data use). Throughout the report, key issues, critical features, pros and cons in choosing the main elements in a Big Data Analytics project are discussed, including the infrastructure needed and the cost implications.

Recommendations at the end of the paper point to specific competences to employ in realising a fully-fledged big data analytics project; the trade-off between performance and explainability in Machine Learning; rules for implementing big data analysis by using cloud computing services; aspects to address when creating smart statistics by combining results of the big data analysis and official statistics; and considerations for dealing with issues of representativity of data sourced from the internet.

Guide to readers: What do the badges next to chapter titles mean?

To support the reader, each section has one or more badges indicating the nature of the content:



Methodological: the topic will be discussed focusing on the “why” and the “what”;



Technical: the topic will be discussed focusing on “how” the methodology works, listing the state-of-the-art solutions and providing some technical aspects (if applicable);



Example: the topic will be clarified to the reader by using either a working/toy example (realised ad-hoc to clarify the matter) or a use-case scenario taken from a real application.

Chapter 1 – Introduction to Big Data and Analytics



The magnitude of data generated and shared by businesses, public administrations, scientific research, numerous industrial and not-for-profit sectors has increased immeasurably in the last few years. These data include textual content (i.e. structured, semi-structured as well as unstructured), multimedia content (for example videos, images, audio) on a multiplicity of platforms (for example machine-to-machine communications, social media sites, sensor networks, cyber-physical systems, and Internet of Things).

In such a scenario, the term *Big Data (BD)* is used to define large-scale data, usually having a varied and complex structure, which present challenges when they have to be processed, analysed, and presented using typical technologies, methods, and theories. The activity of processing big data and looking for valuable information, correlations and patterns within them is called *Big Data Analytics (BDA)* (Sagiroglu & Sinanc, 2013). Reconciling the information that can be extracted from big data with the quality associated with traditional statistics, including representativeness, is a novel and intriguing field. The aim of this paper is to describe the latest big data techniques and to distil relevant aspects which can be applied by Eurofound in the field of social research.

Big data are usually generated from a variety of online sources such as social media, e-mails, sensors, mobile and desktop applications, online search queries, GPS data and many others, and is summed up to data collected internally from administrative, commercial, and logistic sources.

It is estimated that every second, each of the 5.2 billion active internet users generates about 1.7MB of data¹, and that by 2025, the amount of data created daily on the internet will amount to 463 exabytes². This is not only a gigantic opportunity for companies but also a field of competition among firms and a potential source of information for research. Suffice it to say that, for instance, 68% of Instagram users use Instagram to interact with brands and creators³ and almost 70% of the globe's GDP activities have undergone a digitalisation process. Four of the largest tech companies, Google, Facebook, Microsoft and Amazon, store at least 1,200 petabytes⁴ of information⁵, which constitute their main source of competitive advantage. We can say that for several organisations, the ability to collect, process and use Big Data for decision making is essential in this context. These data are generated continuously by several sources, without any control from the user. For this reason, they are usually described through five implicit characteristics or dimensions (volume, velocity, variety, veracity, and value) and are discussed below.

Any Big Data architecture faces three types of challenges, namely: *data, process, and management challenges*:

1. **Data Challenges:** these are related to the intrinsic complexities and challenges (not only technical) deriving from data complexity;

¹ Source: DOMO, www.domo.com/learn/infographic/data-never-sleeps-9

² Source: World Economic Forum, www.weforum.org/agenda/2019/04/how-much-data-is-generated-each-day-cf4bddf29f/

³ Source: <https://www.kontentino.com/blog/instagram-statistics-to-revamp-strategy/>

⁴ A petabyte is a quadrillion of byte (10^{15}). 1 TB roughly corresponds to 1,5 million CD-ROMs.

⁵ Source: Science Focus, www.sciencefocus.com/future-technology/how-much-data-is-on-the-internet/

2. **Process Challenges:** these are the difficulties that arise due to the process of getting and transforming data into value;
3. **Management Challenges:** these are the often underestimated difficulties that any organisation or company faces when putting into practice decisions based on big data analytics.

Big Data and Big Data Analytics

Big Data (BD) refer to a collection of data, characterised by complexity dimensions according to the five Vs model.

Big Data Analytics (BDA) is the process of extracting meaningful information from those data, being able to transform those data into information, then into knowledge.

While, on the one hand, the computing technologies required to manage these data are keeping pace, on the other hand, the human expertise and competencies required to leverage Big Data are lagging behind. Throughout this paper, we discuss data, process, and management challenges, with examples presented at the end of this document.

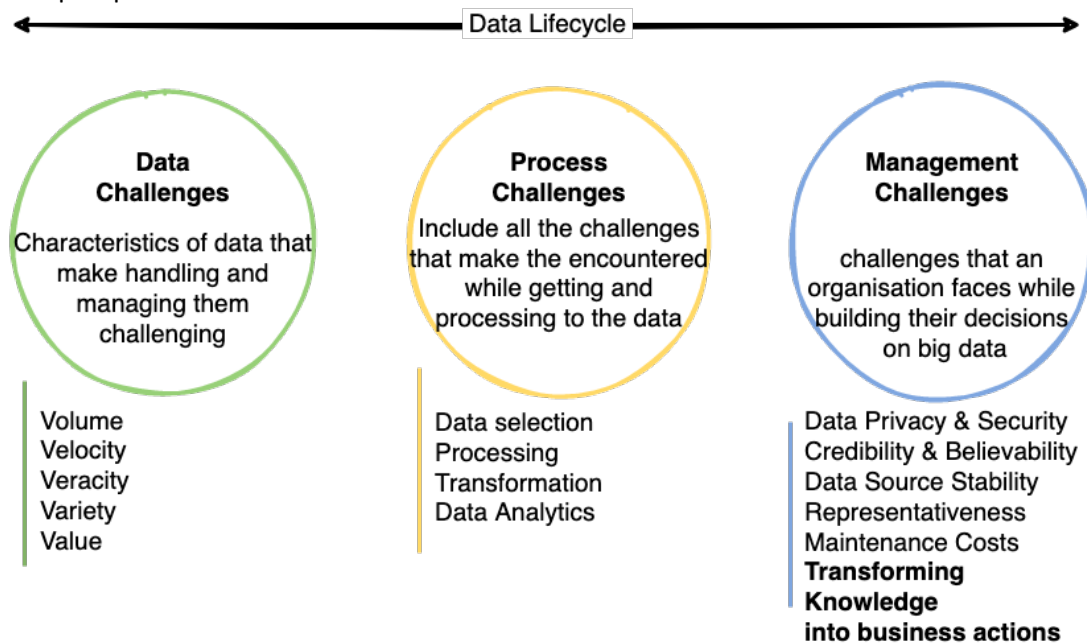


Figure 1. Challenges of the Big Data Lifecycle. Source: created by authors

A paradigm shift in the way of reasoning.

In essence, it can be argued that *any* BDA approach or architecture relies on three key pillars that are discussed in this report, namely: (i) the data availability (i.e. *big data*), which describes the phenomenon to be analysed or explained; (ii) proper algorithms and techniques, which allow data analysis, extracting useful insights (i.e. *data mining*) and (iii) distributed computing, intended as the ability of a system or an architecture to process the large and complex amount of data in a scalable manner (i.e. *cloud computing*).

It is worth mentioning that the use of BDA to support decision making requires a paradigm shift in

the way decisions are made, moving from deductive reasoning to inductive reasoning. When using deductive reasoning, specific statements (hypotheses) are derived from a general statement about a phenomenon (theory). These hypotheses are subsequently tested against empirical observations using statistical analysis. This approach follows a top-down pathway, not only in terms of the formulation of hypotheses, but also in terms of study design. For instance, statistical surveys or medical trials are designed to generate data that allow for the generalisable testing of hypotheses. Similarly, statistical analyses are used to test if a certain population has certain hypothesized characteristics. In all those cases, a dedicated sample is selected with known probability from an existing frame (which sufficiently captures the target population). Conversely, BDA requires decision-makers to employ a different paradigm of reasoning, namely inductive reasoning. This is a reasoning process that starts from specific observations (common patterns or insights coming from the data), moving to specific potential generalisations (hypotheses,) and building towards a theory. This approach is usually called “inductive” (bottom-up or data driven). It aims to unveil hidden patterns in data that were not originally hypothesized with the research question in mind.

Notably, deductive reasoning involves moving from general to specific; for this reason, the conclusion has to be true if the premise is true. Differently, inductive reasoning moves from specific to general; hence, the truth of the premise does not guarantee the truth of the conclusions reached through this way of reasoning.

For the sake of completeness, we have to highlight that these approaches are complementary, as they allow the observation of a phenomenon from different viewpoints. This is the case of *smart statistics*, which aim at putting big data analytics (inductive, bottom-up) into administrative data or official statistics (deductive, top-down), hence extending the ability of statistics to observe and explain a phenomenon, improving the timeliness (real-time statistics) and the coverage (ideally, the entire population that changes continuously). This is far from a straightforward process, and requires computer scientists, statisticians, economists, and domain experts to closely cooperate in the design and implementation of a rigorous methodological framework and system to produce smart statistics.⁶

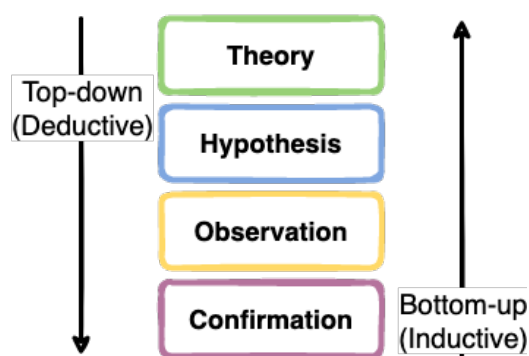


Figure 2. Representation of the top-down (deductive) and bottom-up (inductive or data driven) paradigms for decision making. Source: created by authors.

⁶ https://ec.europa.eu/eurostat/cros/content/trusted-smart-statistics-nutshell_en

Data Challenges: the 5 Vs of Big Data



Data challenges are the challenges related to the characteristics of the data *per se*.

Different researchers have distinct understandings of data characteristics. According to the literature, Big Data should be declined into five dimensions, the so-called five Vs, namely:

1. **Volume.** There are, in the world, more than 5.2 billion active internet users and almost two billion active websites. It is estimated that **every second**, each human being creates 1.7MB of internet data. In 2021, worldwide, Google accounted for 5.7 million searches **every minute**, and almost 575 thousand tweets were posted. At the same time, YouTube users streamed 694 thousand hours of videos, 65 thousand posts were shared on Instagram, Teams connected 100 thousand users, TikTok users watched 167 million videos and Facebook received 44 million views⁷. These are some examples of public data that can be accessed by companies and used to make better decisions and improve their businesses. This is not just a possibility but is already a standpoint of competition and of organisation growth. Although the classic Big Data approach measures volume in terms of bytes, it strongly depends on the domain and the type of target data. For instance, in Social Media Analytics, the number of tweets or Facebook comments collected is certainly more appropriate than the size in bytes. Or, in Labour Market Intelligence, when analysing Online Job Advertisements, the number job advertisements is a suitable volume unit.
2. **Velocity.** This dimension refers to the rate of accumulation of data, both generated internally or collected externally, and is closely linked with the collected volume. The higher the accumulation rate is, the higher the volume collected and, therefore, the need for higher performance computing infrastructures and larger storage systems.
3. **Variety.** This dimension refers to the nature of data, which can be structured, semi-structured or completely unstructured. Dealing with large, unstructured data usually means that, because their heterogeneity is high, more attention should be dedicated to the pre-processing and transformation of data.
4. **Veracity.** This dimension refers to how accurate or truthful a data set is. In a broader sense, it is referred to as the 'quality' of the data that will be used by an automated process to derive insights. In the context of big data, however, data veracity also refers to the trustworthiness of the data source, as data come from a data source external to the organisations whose governance falls outside the control of the user. Removing biases, abnormalities or inconsistencies and duplication are key elements that allow for improving the accuracy of big data. Not surprisingly, the data quality motto 'garbage in, garbage out' (GIGO) still applies to big data, shedding light on the importance of both (i) quality techniques to reduce the noise and to improve the overall data quality and (ii) trustworthiness of the data owner, that result in an improved believability (Wang & Strong, 1996) of the overall process.
5. **Value.** Finally, data have no value *per se* but need to be useful to a specific domain or purpose. The value of data is the knowledge, extracted by a human user, used to explain a phenomenon, support a decision, or make a recommendation in a specific context or domain. Therefore, it is worth mentioning the importance of the analysis of stakeholder needs, which should identify which portion of knowledge is of interest to a given stakeholder and which is not.

⁷ DOMO, www.domo.com/learn/infographic/data-never-sleeps-9

Table 1. The five dimensions of Big Data

Data	Main Characteristics	Example of use
Sensors and IoT	Velocity, Volume	predictive maintenance, fault analysis, tracking workers' stress, tracking and monitoring air quality
Radio Frequency Identification (RFID)	Velocity, Volume	Indoor consumer tracking; basket analysis, logistics (tracking of goods).
Commuters' data (GPS, 4G)	Velocity, Volume, Variety	Monitoring living and working conditions
Hospital Discharge Letter	Variety, Veracity, Volume	Hospital performance analysis, monitoring of infectious diseases
Online Job Ads	Volume, Variety, Veracity	Understanding and monitoring labour market, trend predictions, skill identification, job matching
Fiscal, banking, and monetary data	Volume, Variety, Veracity	Anomaly and fraud detection, identification of tax evaders.

Source: created by authors.

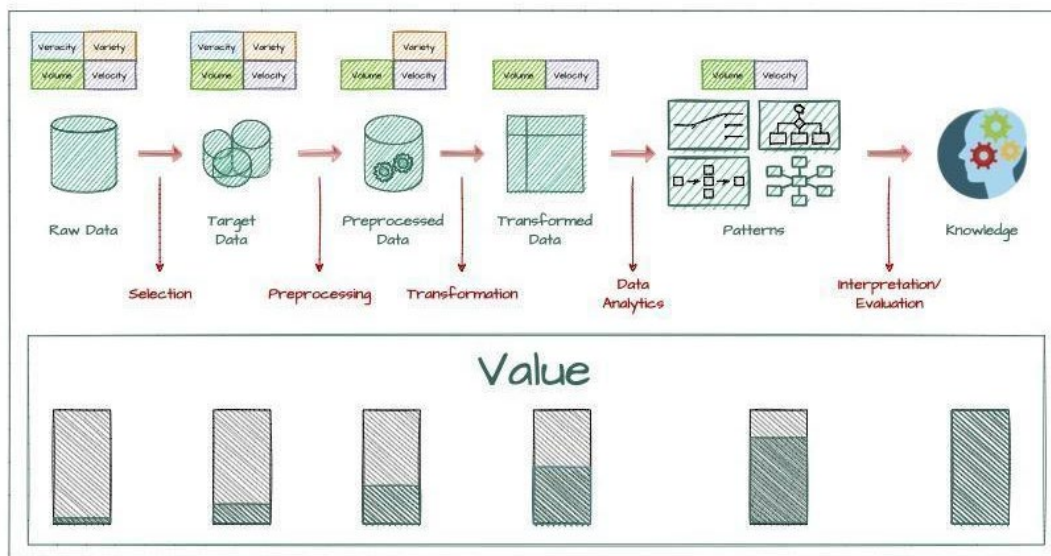


Figure 3. Process Challenges: Big Data Processing Lifecycle. Source: created by authors.

Process Challenge: data, information, knowledge



The large quantities of data collected and stored by organisations, even in a structured and systematic way, cannot be used as-is for making decisions. The goal of the Big Data Analytics process is to transform data into information and then information into knowledge that can be used by decision-makers, bringing value to the organisation. In the following comments, we will better explain the difference between data, information and knowledge in the Big Data stream.⁸

1. **Data.** Usually, data are collected in a raw form that needs to be processed. Raw data represent a collection of numbers, texts, images, audio, and video files in an unorganised form. Usually, the first step is to select, among the data collected, the portion that can be useful to the organisation, i.e. the target data. Subsequently, data are pre-processed and transformed in a structured way.
2. **Information.** Once data have been processed and transformed, data analytics techniques are performed to extract patterns and correlations from them. Information is the outcome of extraction activities, and it is meaningful to those who receive it in a certain domain.
3. **Knowledge.** Information is transformed into knowledge when it is interpreted, evaluated, and used by people to make decisions. Therefore, it is possible to think of knowledge as information in action in a specific domain, through the experience and judgement of a decision-maker, to tackle a specific task or problem. This goes towards the so-called fact-based decision-making process.

As we move forward in the Big Data Analytics Lifecycle, usually data become more structured and their volume decreases, while their value grows. Figure 3 (above) shows all the steps of the Big Data Analytics Lifecycle (BDAL), from raw data to valuable knowledge.

Real-Life Example in the field of Labour Market Intelligence

Process Challenge: data, information, knowledge, decision

The following example should clarify this matter. Let us imagine collecting online job ads (OJA) from job portals and aggregators about demand in the labour market, as it is conveyed through web portals, with the aim to monitor and observe labour market demands.

Data: online labour market demand is expressed through Job titles and descriptions as raw text. Moreover, they might include the location, skills demanded (digital, non-digital, soft), the name of the company and its sector, as well as the salary. All that information is encoded within the text, expressed using the labour market lexicon, which can vary from business to business.

1. *Volume:* millions of OJAs, a high number of sources of information

⁸ Readers may consider the following for further reading:

Mayer-Schönberger, V., & Cukier, K. (2013), *Big data: A revolution that will transform how we live, work, and think*. Houghton Mifflin Harcourt.

Davenport, T. (2014). *Big data at work: dispelling the myths, uncovering the opportunities*. Harvard Business Review Press.

2. *Velocity*: OJAs published daily, remaining active for an unknown period of time, that changes from source to source
3. *Veracity*: differentiate between duplicated and reposted advertisements, select trustworthy source of information; how should missing information be managed?
4. *Variety*: each source of information has its own structure, which varies from totally structured (the information is organised) to totally unstructured (tree text)
5. *Value*: an overview of the online labour market demand in terms of occupations, skills, territory, and sectors, as accurate as possible

Information: apply all the steps of the process challenge shown above to transform data into information. This includes the following steps:

6. *Data Selection*: identify the most relevant source of information to be included; rank websites to select the ones that provide the information wanted (e.g. skill-based analyses will privilege sources providing skills whilst a trend-based analysis should concentrate on sources that guarantee a stable trend over time)
 - statistics competencies: *how should the criteria to be included in the source model be identified and how should they (i.e. variables) be extracted from sources? How should sources be ranked?*
 - computer science competencies: *how should a huge mass of data at scale be stored? How should data be collected automatically? Do we need API access, or do we need to build up a Web scraper/crawler? How should automatic data collection processes be scheduled?*
 - domain competencies: *how should the right sources be selected? Do we have a selected source that is significant for the labour market? Is there a way to evaluate their representativeness? Is there any official survey to be used as a benchmark?*
7. *Data pre-processing and Transformation*: this includes the data cleaning task to remove noise from the data or inappropriate outliers (if any) and to harmonise the data formats among sources and countries.
 - statistics competencies: *how should data consistency be evaluated? How should data accuracy be measured? How should data significance be estimated? Are we under/over representing jobs?*
 - computer science competencies: *how should duplicate data records be identified? How should missing values be identified? How should big data procedures be built up so they can process and transform raw data into a structured model in a scalable manner?*
 - domain competencies: *how should LM domain synonyms that help in improving data accuracy be identified? How should criteria that characterise missing and duplicated values be identified?*

Knowledge: when information is analysed, interpreted, evaluated through the following steps, and used to make decisions, it becomes valuable knowledge.

1. *Data Analytics:* identify the proper AI algorithms, searching for patterns of interest in a particular representational form based on the analysis purposes. In our example, this includes training text classification algorithms that assign one job class to each job ad, according to predefined classes (e.g. the European Occupation Classification, ESCO).
 - statistics and computer science competencies: *How should the best algorithm be selected? How should the parameters be tuned? How should algorithm effectiveness be evaluated? How should it be implemented at scale?*
 - domain competencies: *which knowledge has to be selected and which has to be discarded? What is the economical meaning behind the resulting knowledge? Which novel insights have been discovered? How should results from the mining process from a labour market perspective be explained?*
2. *Interpretation/Evaluation:* identify indicators, dimensions of analysis and visual paradigms to represent the resulting knowledge. This step is usually driven by domain questions to use the data to improve decisions, including both BI-related questions “Which occupations are growing more? Where? What skills are in demand?” to BA “Which occupations will grow in the future and where? What skills will be most in demand in the next years? What are the new emerging jobs?”

Questions and Answers

About Veracity. Q: Does veracity include issues of self-representation? For instance, most social media data are true when taken as information in the way in which posters want them to be perceived, but not necessarily when taken as information on who they are.

A: *Let us consider the case of getting tweets from Twitter for a specific topic. Veracity considers both: (i) quality issues related to how good the data are. In the case of social media, the quality would consider the number of duplicated entries, the presence/absence of a field needed to perform your analysis (e.g. latitude and longitude are needed if you plan to perform a geographical analysis of tweets), the typos that should be fixed (in case of textual or sentiment analysis); (ii) trustworthiness issue, related to what extent the source of information is trusted. If the same information may be collected from multiple sources, the trustworthiness of a source should be assessed (see the AHP section in the next chapter), differently, you have to trust the only source of data available. In the case of Twitter, there is no way of knowing if - and to what extent - the tweets you are getting represent all the tweets about a specific topic, what rank Twitter uses to provide the tweets and if - and why - Twitter prevents you from obtaining some tweets.*

Concerning the Real-Life Example. Q: How should multilanguage be managed?

A: In the case of the example above, the process described should be performed for each language. This clearly increases the complexity but allows for high precision of the result. In this case, indeed, the use of the ESCO taxonomy is crucial as it is expressed in all the EU languages. Generally speaking, any BDA approach that aims at using textual data should be considered language-dependent; this means that the collecting pipeline, as well as the language models and algorithms, should be tuned for each language from scratch.

Concerning the Real-Life Example. Q: How should missing information be managed?

A: The problem of missing information is key in any statistical-based analysis as it pertains to the reasoning under uncertainty. First, we should distinguish between missing value by design and missing value by process. The former means the datum is missing as it does not exist in the domain of interest. For example, the variable “second name” would be missing for people who do not have a second name. The latter means the datum exists, but it was not collected due to reasons that are outside our control (the most common case). Think of the salary value that is missing in many job ads.

You should decide to adopt either the closed world assumption (CWA) or the open-world assumption (OWA). The former considers what is unknown as false or, simply, it is discarded from analyses. The latter is the opposite, considering that what is not known to be true remains unknown (i.e. null or NA).

Concerning figure 1. Q: Why do variety and veracity disappear along the process?

A: Dealing with Big Data requires “reducing” the complexity as soon as the process goes. This means that dimensions such as Volume and Velocity may remain (i.e. the process continues handling a huge amount of data in real-time) whilst some other dimensions (such as Variety and Veracity) decrease until they disappear. Variety usually decreases as data are harmonised using a common data model to derive indicators and statistics; Veracity usually disappears because the problem of dealing with data quality and trustworthiness is addressed and solved during the processing.

Measuring welfare accounting of new and free technological goods

The fact that almost 70% of the world’s GDP has undergone digitalisation makes it an interesting discussion regarding the contribution to welfare of technological goods. On the one side, they are not accounted in traditional welfare measures, such as GDP, because they are free, but they constitute a valuable service. On the other side, in addition to their benefit, it is useful to understand their cost (e.g. in terms of privacy, environmental cost). Finally, it is still difficult to link the use of free technological goods to the revenues they generate (value of information) because of increasingly complex and indirect revenue streams. The GDP-B (Brynjolfsson, Erik, et al. No. w25695. National Bureau of Economic Research, 2019) could be an interesting starting point for this discussion.

Chapter 2 – Data Selection and Collection



Even though this might seem surprising, the Volume dimension of big data should not be rephrased as “the more, the better”, as the higher the volume of the data, the higher the complexity in terms of veracity and variety as well. This, therefore, requires selecting *the proper source of information before developing a big data analytics solution*.

Domain experts should select the sources of information according to their ability to bring information and then rank according to the analytical purposes. This is the reason behind the need to select and rank web sources.

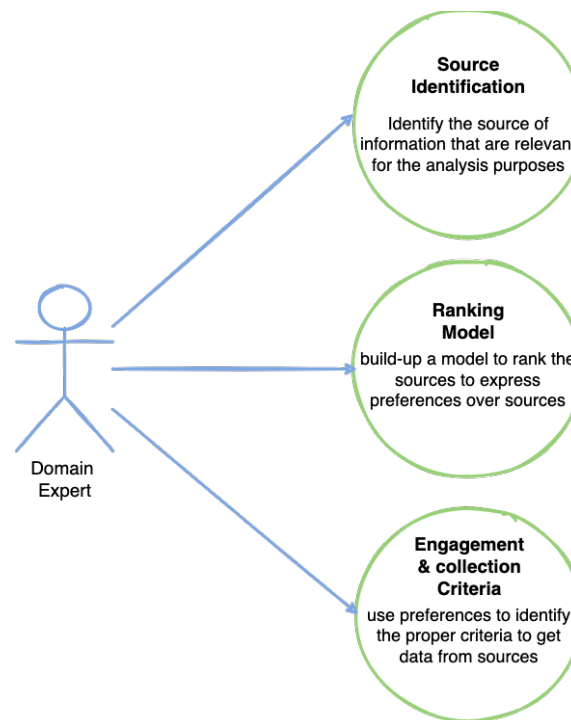


Figure 4. Activities of the data selection and collection process. Source: created by authors.

Source Identification

Generally speaking, a ranking model would represent a partial order relationship between a set of items (i.e. sources of information) to allow analysts and decision-makers to express preferences over items. The ranking is based on a set of variables, both quantitative and qualitative. The latter need to be transformed from qualitative to quantitative to be included in any ranking models.

A hierarchy of criteria might be like the one in Figure 5 and in the box below, where:

3. **structured** means the information can be easily obtained from the source, i.e. the developer can write a regular expression that matches the structure that envelopes the information;

4. **semi-structured** means the information does not have a structure by itself. However, it can be retrieved through a semi-structured format (usually, JSON⁹) that allows information extraction;
5. **unstructured** refers to information that either is not organised in a pre-defined manner or does not have a pre-defined data model. Unstructured data usually refers to a set of text-heavy data but may contain data such as numbers, dates, and facts as well. Note that videos, audio, and binary data files might not have a specific structure, hence they are considered unstructured data.

Real-Life Example

Source selection of Travel Marketplaces

In this example, the aim is to build a real-time monitoring system for the travel marketplace, to allow analysts to quantify and compare the demand for digital travel companies. To this end, a good first step would be selecting a set of “variables” as they emerge from those websites.

1. Variables for capacity and occupancy
 1. Number of hosts (number of hosts renting out one or more listings)
 2. Number of bed places
 3. Number of stays (number of rentals of each listing during the reference period)
1. Time-related variables
 1. Number of nights rented out (number of nights each listing was rented out during the reference period)
 2. Number of overnight stays (number of guest nights spent at each listing during the reference period)
1. Quality related variables
 1. Number of postings per week
 2. Update frequency
 3. Payment methods

The user expert then identifies a set of sources of information to be analysed. In this case, this might include a set of platforms and agencies, $S = \{\text{Trivago, Booking, Expedia, etcetera}\}$

⁹ Javascript Object Notation is the most commonly used data format to exchange data among services. This means that the activity of “collecting” data from a web server is more likely to result in a JSON.

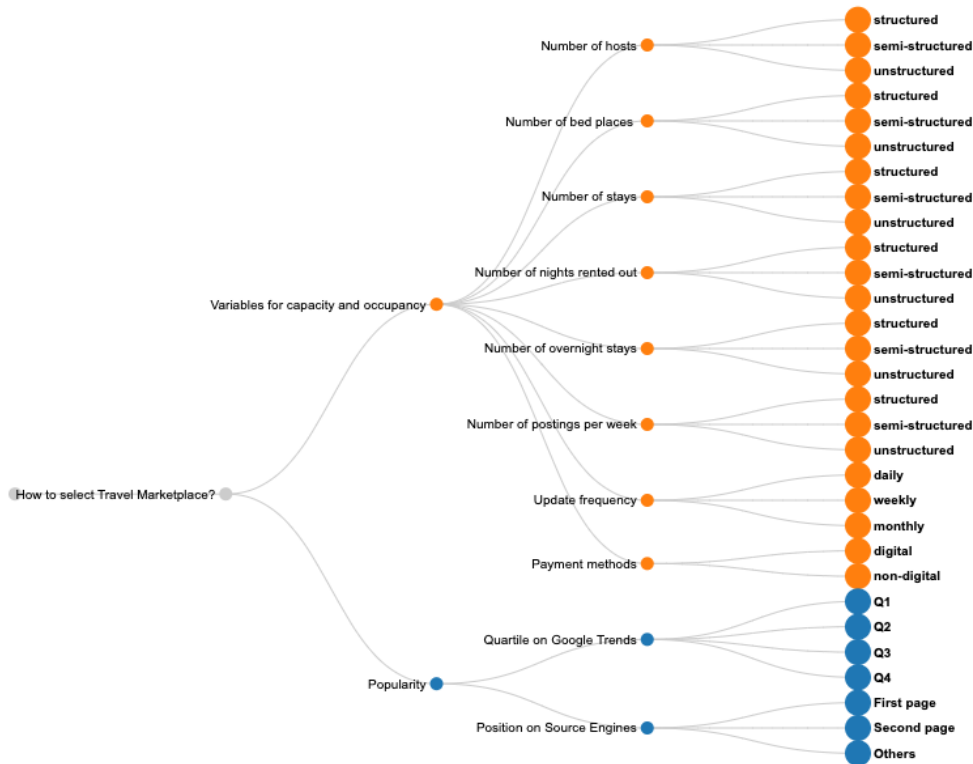


Figure 5. Example of a hierarchy of criteria for the real-life example on travel marketplaces Source: created by authors.

Ranking Model

Why do we need preferences over ranking criteria? To date, the ranking model assigns to each criterion the same importance for producing the final list of websites to be engaged. This means that the ranking model does not allow for expressing the importance of one criterion over another (e.g. for forecasting analysis, having accurate update frequency is more important than having the accepted payment methods).

The goal of this step is to improve the ranking model, enabling stakeholders to assign importance (i.e. the weight) to each criterion based on the analysis purposes. This is beneficial to stakeholders so they can:

1. Have a ranking model that better fits the analysis purposes;
2. Have a list of Web sources that should be engaged with first, as they are crucial for performing the analyses;
3. Understand and estimate the role that certain website or vacancy variables play in the analytics.

To this end, a useful approach consists of employing Multi-Criteria-Decision-Making techniques, which allow users to enumerate the criteria used to decide. A model where each criterion has the same importance would not allow decision-makers to prefer one criterion over another. Indeed, the decision-maker might consider a platform's attribute more important than another. To handle such a scenario, one might plan to use the Analytic Hierarchy Process (AHP), briefly introduced in the following section.

Informal Description. The AHP¹⁰ is an effective technique for dealing with multi-criteria decision-making problems, which allows decision-makers to set priorities of variables to make decisions that fit their variable preferences. By reducing complex decisions to a series of pairwise comparisons, and then synthesising the results, the AHP helps to capture both subjective and objective aspects of a decision. The AHP is a very flexible and powerful tool because the scores, and therefore the final ranking, are obtained based on the pairwise relative evaluations of both the criteria and the options provided by the user. The computations made by the AHP are always guided by the decision maker's experience, and the AHP can thus be considered as a tool that is able to translate the evaluations (both qualitative and quantitative) made by the decision-maker into a multicriteria ranking.

An example is shown in the figure below, where the weighted hierarchy of criteria was weighted by a stakeholder with her/his preferences. As one might note, evaluating the ranking according to the unweighted criteria would consider all the criteria equally important. Differently, the weighted model would encode the user preferences, and, in this case, it highlights that the number of bed places is the most important variable to rank platforms (26%), followed by several nights rented out (20%) and the number of stays/hosts (15%). Surprisingly, the user did not give importance either to the update frequency or to the number of postings, which, in turn, will have a neglectable impact in forming the final ranking. Consequently, a system collecting data from platforms selected through these criteria might have very low coverage in terms of postings and issues in terms of quality, as the update frequency is not valued properly.

The AHP, as a further benefit, helps decision-makers in clarifying the criteria that should be used to select sources and, in turn, helps in clarifying the questions and refining the analysis purposes.

Decision Hierarchy			
Level 0	Level 1	Level 2	Glb Prio.
Select-travel-marketplace	Variables-for-capacity-and-occupancy 0.333	Number-of-hosts 0.250	8.3%
		Number-of-listings 0.250	8.3%
		Number-of-bed-places 0.250	8.3%
		Number-of-stays 0.250	8.3%
	Time-related-variables 0.333	Number-of-nights-rented-out 0.500	16.7%
		Number-of-overnight-stays 0.500	16.7%
	Quality-related-variables 0.333	Number-of-postings 0.333	11.1%
		Update-frequency 0.333	11.1%
		Payment-methods 0.333	11.1%

Figure 6a. AHP hierarchy – unweighted. Source: created by authors.

¹⁰ Saaty, T., L. (1988). *What is the analytic hierarchy process? Mathematical models for decision support*. Springer, Berlin, Heidelberg, 1988. 109-121.

Decision Hierarchy			
Level 0	Level 1	Level 2	Glb Prio.
Select-travel-marketplace AHP	Variables-for-capacity-and-occupancy 0.674 AHP	Number-of-hosts 0.230	15.5%
		Number-of-listings 0.161	10.8%
		Number-of-bed-places 0.378	25.5%
		Number-of-stays 0.230	15.5%
	Time-related-variables 0.226 AHP	Number-of-nights-rented-out 0.889	20.1%
		Number-of-overnight-stays 0.111	2.5%
	Quality-related-variables 0.101 AHP	Number-of-postings 0.648	6.5%
		Update-frequency 0.300	3.0%
		Payment-methods 0.052	0.5%

Figure 6b. AHP hierarchy - weighted according to the user's preferences. Source: created by authors.

Engagement and Collection Criteria

Thanks to the ranking model, the decision-maker can understand why - and to what extent - a source of information is important. Top-ranking sources should be engaged through specific and ad-hoc contracts to allow the system to collect data from them. In any case, there are three ways to engage a source of information, as discussed below.

There are, worldwide, more than two billion active websites; therefore, collecting web data offers huge business opportunities to organisations. While regulations vary from country to country, usually the content of web pages can be freely collected and used, complying with the privacy regulations in force. Organisations may resort to two main techniques to extract data from websites: **web scraping** or **Application Programming Interfaces (APIs)**. Web scraping gives you the option to automatically collect data from any web page through ad hoc tools; it is usually used jointly with a **web crawler** (or spider bot), an instrument that allows you to visit and scrape all the web pages under a certain domain or conditions. APIs are put in place to provide users with direct access to some data or sources.

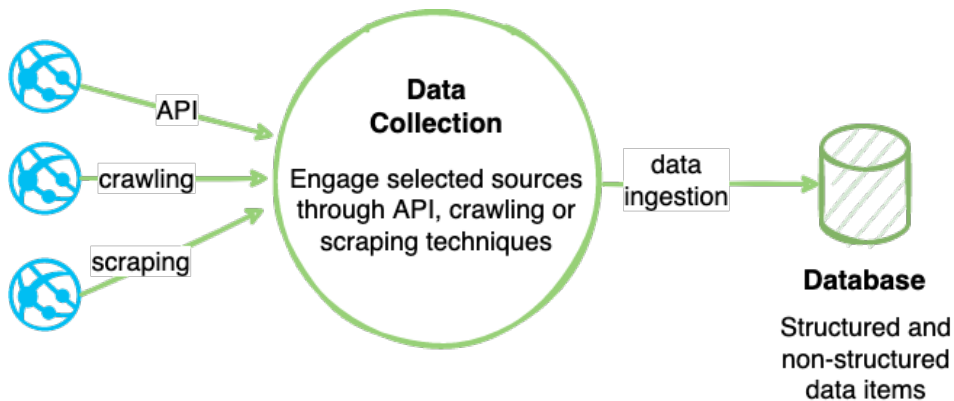
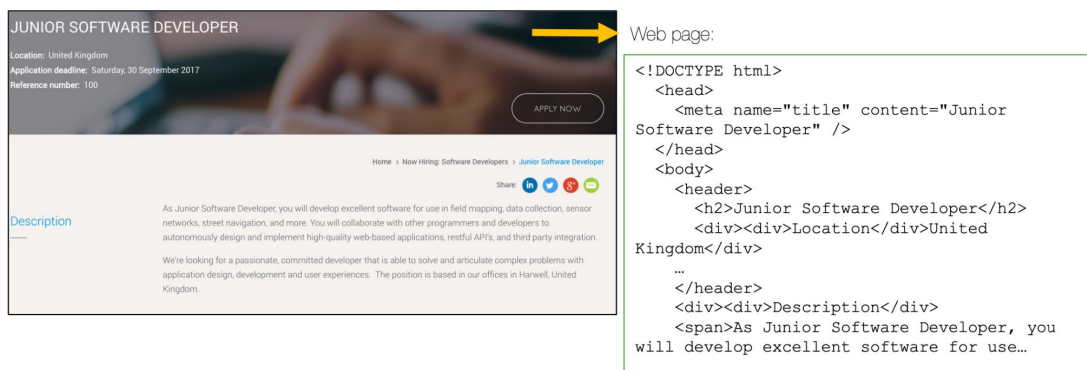


Figure 7. Data collection lifecycle. Source: created by authors.

1. **Web scraping:** the term “web scraping”, also known as “web harvesting” or “extraction”, is a method to extract data from the World Wide Web (WWW) and store it in a database (Zhao, 2017) by means of an ad hoc tool. Web scrapers can be used manually by a user who decides which pages to scrape or who can automatically visit several pages through web crawlers. It usually works in two steps. First, it automatically requests a Web page, then it collects only a limited part of the information from it, leaving out the remaining data. This means that a scraper (partially) knows the Web site structure so that it can identify only the content of interest for the analysis.
2. **Web crawling:** this is a software activity that automatically indexes all the contents of a web page and adds them to a database. It iteratively follows all the hyperlinks included in a domain or which follow certain rules and also indexes that data to the database (including images, tables, and style sheets). A classic example of crawling is the search activity performed by Google. While this activity is valuable *per se*, usually it is used in conjunction with web scraping, to combine the capacity of the crawler to automatically index an enormous number of pages in short times and the flexibility of the web scraper in selecting and formatting the correct information. As an example, a web crawler might download all the products listed within an e-commerce website, whilst a scraper might be developed to collect only product names and prices, leaving out links to banners, comments and meta-data related to the page layout.
3. **API:** Application Programming Interfaces are software components exposed to any programmer to enable data collection. They are usually provided by the source owner (e.g. Twitter, Facebook, Eurostat), which in this way can control the collection process, in terms of which data are emitted, in which structure and rate.

The example below should clarify the difference between crawling and scraping: the former is a bot that systematically browses web portals to download all their pages. An HTML web page encodes the information (semi-structured format).



Web page:

```

<!DOCTYPE html>
<head>
  <meta name="title" content="Junior Software Developer" />
</head>
<body>
  <header>
    <h2>Junior Software Developer</h2>
    <div><div>Location</div>United Kingdom</div>
    ...
  </header>
  <div><div>Description</div>
  <span>As Junior Software Developer, you will develop excellent software for use...

```

Figure 8. An example of webpage crawling. Source: created by authors.

A web scraper can be used either after a crawler collects the page or to directly extract structured data from a website, as in the case below.

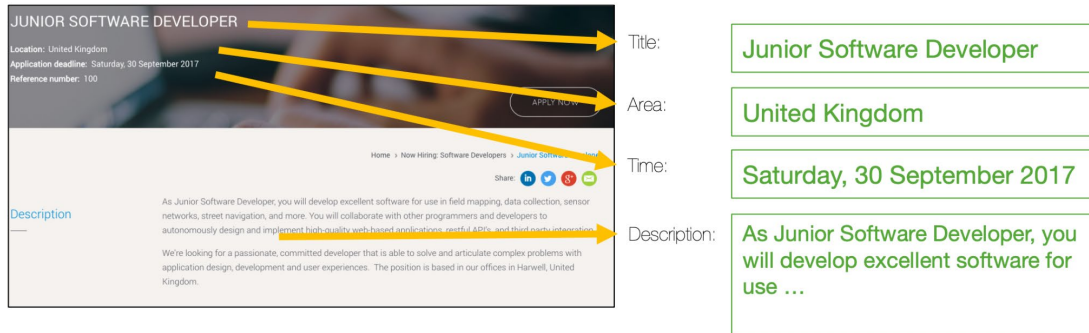


Figure 9. An example of webpage scraping. Source: created by authors.

Questions and Answers

About Data Collection. Q: What is the "best" way to collect data?

A: We discussed three ways to collect data: crawling, scraping and API. First, you should consider that all these techniques try to collect someone else's data. This means you should consider that your approach might change frequently, as soon as the data owner performs (even minimal) changes to the website or data policies as well.

Indeed, losing a portion of data due to scraper misalignment or a block set by the data owner might have an unpredictable impact on your analyses, and this is the reason why having an agreement with the data owner is always preferable, in spite of the techniques used to collect the data.

Once the data owner is informed about - and agrees on - the collection process, any technique can be used. Collecting from API is the best way to proceed.

Chapter 3 – Pre-processing and Transformation



This step deals with the processing and manipulation of data sets to analyse and improve the overall quality, to build up automated procedures and routines, which transform the data according to a unified model of data (if any) through relational paradigms (i.e. SQL) and non-relational ones (i.e. NoSQL). This step aims at preparing the data needed for the next set of data mining activities. This activity is orchestrated by ETL technologies:

1. **(E) Extract:** this stage focuses on the “data extraction” task, which involves both homogeneous and heterogeneous sources of data. During this step, the quality of the selected data is analysed and assessed, and proper cleaning routines are defined until a satisfying quality level is achieved.
2. **(T) Transform:** this stage deals with the transformation of the selected data from one format (original) to another (destination). Note that data cleaning routines can be used to improve the quality of a variable while moving from one format to another. Furthermore, this step requires employing state-of-the-art big data techniques to process massive amounts of data. A survey on big data architecture falls outside the goal of this guide. Readers who would like to know more of this topic can read (Marz & Warren, 2015).
3. **(L) Load:** the load phase is aimed at loading the resulting data into the desired database/data warehouse¹¹ (Power, 2002; Yang & Helfert, 2019), using the data modelling paradigms of the literature, namely, relational and non-relational paradigms. Depending on the data and their characteristics, the load and integration of data into databases/data warehouses might be complex and should be carefully designed. This step is crucial as the next phase of the data lifecycle shown in figure 1 will make use of those data to perform data mining activities.

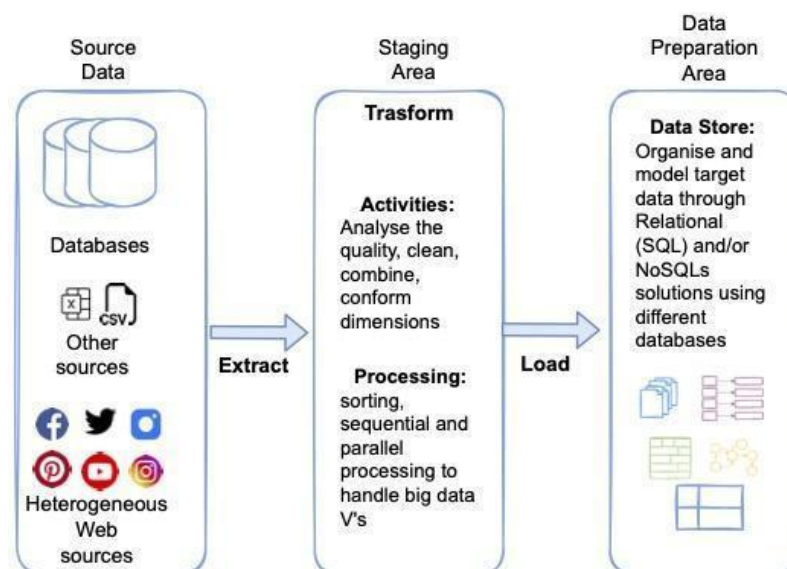


Figure 10. Graphical representation of the ETL process. Source: created by authors.

¹¹ A data warehouse is a (large) collection of data coming from multiple sources, stored centrally, and used for reporting and data analysis.

Please note that this step of the data lifecycle process usually needs a *data engineer* to be accomplished properly whilst the subsequent data mining step involves *machine learning engineers*.

Pre-processing: Data Quality and Cleaning

The term “data quality” (DQ) (Redman, 2001) applies to the whole data lifecycle process, as the term also specifies the meaning: data quality refers precisely to the process of analysing the quality of a dataset. Similarly, data cleaning (or cleansing) indicates the (automated) process that improves the overall quality of the dataset. In the DQ community, there are two well-known principles that deserve to be introduced to the reader:

(i) “GIGO: garbage in, garbage out”. The quality of a dataset or a variable should be expressed from a quantitative perspective to allow a machine to analyse the quality and improve it. Conversely, poor data quality might have unpredictable effects on the analytics and any other process that exploits data. This is the reason why data quality is composed of *dimensions* and *metrics*: dimensions provide a qualitative description of the quality characteristics we are interested in, while metrics provide formulae to measure the quality level reached. Though many dimensions (and, in turn, metrics) have been identified over the years, one of the most adopted classifications is the one provided by Redman(2001), which classifies DQ issues by considering the distinct levels where they can occur:

1. (a) The *intensional* level includes criteria related to the content of the conceptual schema relevance, as well as consistency and flexibility. In other words, it considers the conceptual model used to represent the data;
2. (b) The *extensional* level considers the data values including criteria such as accuracy and correctness of values, timeliness, and completeness of data. This level focuses on the *content*, rather than on the modelling structure;
3. (c) The level of *data representation* focuses on issues related to the data format, e.g. interpretability, portability, adequateness.

The reader can easily note that the (b) extensional level is the one that counts more from a big data analytics perspective.

(ii) “fitness for use”. Even though a quantitative measure allows for estimating the quality of a dataset, a “threshold value” is needed to consider the quality of the data adequate. Informally, data quality and cleaning are defined as “fitness for use” to indicate a satisfying quality level depending on both the user needs and analysis purposes.

An example should help clarify the matter.

Real-Life Example of Data Quality on Labour Data

Context. According to the Italian Labour Law, every time an employer hires or dismisses an employee or an employment contract is modified (e.g. from part-time to full-time, or from fixed-term to permanent), a communication (i.e. an event) is sent to a job registry. That information is called Mandatory Communication (CO). Since 1997, the Italian public administration has been using an Information System, called the “CO System”, where data concerning employment and active labour market policies are stored. Extracting the historical data for each subject from the CO archives allows the observation of the overall evolution of the labour market over a given period, obtaining insightful information about

worker career paths, patterns, and trends, facilitating the decision-making processes of civil servants and policymakers as well.

Data Quality Dimension and Metric. The career of a subject can be seen as a sequence of events, which can be the start/end/transformation of a working contract. We can imagine a consistent career path always begins with a “start” communication, with a number of transformation events, concluding with an “end” communication. However, as the collection of the time series started in 1997, all people who started working before that date are not included in this repository. This represents a quality issue (*dimension: consistency*), mainly due to an Italian labour law that forbids people from having two active contracts in the same time interval unless they are both part-time contracts. This means that “reconstructing” the workers’ careers is mandatory to perform meaningful analyses. A metric to evaluate the consistency of a career should be designed accordingly.

Dimension: Consistency. A working career should always begin with a “start” event. Otherwise, the career is inconsistent.

Metric. Let C be the set of careers and C' the set of inconsistent careers. A metric to estimate the level of dataset inconsistency might consider the ratio of inconsistent careers over the total. Alternatively, one might decide to normalise the value of C , for example, considering only careers before 1997.

Data Cleaning activities. A set of “policies” should be defined to “cleanse” inconsistent careers. One might decide to put a dummy “start” event, establishing the starting date according to a mean value.

Source: Mezzanzanica et al., (2015)



Figure 11. Gartner Magic Quadrant for Data Quality companies. Source: Gartner.

The data quality analysis and improvement tasks have been the focus of a large body of research in different domains, which involve statisticians, mathematicians, and computer scientists, working in close cooperation with application domain experts, each focusing on their own perspective. According to Gartner¹², poor data quality costs organisations \$12.9 million (about €11.3 million) a year. Mostly, over the long-term, poor-quality data affects the complexity of data ecosystems and processes, leading to poor decision making.

This clarifies the presence of many big players that propose solutions to analyse and cleanse data. The 2020 Gartner magic quadrant provides a bird's eye view of the solutions available on the market.

Though all those commercial solutions are effective in practice, worthy of mention is an open-source solution proposed by the community, namely OpenRefine¹³. Formerly proposed by Google, OpenRefine is a tool that allows analysing the quality of any dataset by means of built-in algorithms and metrics, cleansing it through massive operations on the data and exporting the final dataset as well. As a benefit, it allows the user to explore the data visually, then it automatically translates the cleansing activities performed by the user into cleansing routines (no coding is required) and finally allows estimating the impact of each implemented data cleaning activity. Moreover, OpenRefine is equipped with some free plug-ins to perform data reconciliation and augmentation on Wikipedia, to certify the quality of the data, as well as to automate cleaning activities in a batch mode.

Data Store: NoSQL in a nutshell

As mentioned above, the “end product” of the data processing and transformation step is an organised data repository that takes into account the characteristics of the data as well. To better understand this part, we have to distinguish between structured and unstructured data.

Structured data refers to data whose schema follows a specific and pre-defined structure, which easily allows a machine to ‘reason’ on it. Examples of structured data include names, dates, addresses, credit card numbers, stock information, geolocation and more. This, as a consequence, allows the storing of that information in a table-like manner (*aka*, relational paradigm), querying them through the SQL (Standard Query Language). Below is an example (left) of a relational schema that organises structured data, while the right side of the figure shows an example of ‘content’. As you can see, tables are ‘connected’ by means of keys, which allow for data connections. One might note that the customer name is equipped with a numerical ID that acts as a foreign key in the ‘orders’ table. The same happens for the ‘Address’ table. Hence, the row of ‘Orders’ indicates that Martin (CustomerID=1) made an order (ID= 99) that should be shipped to Chicago (ShippingAddressID = Address.ID = 77).

The relational paradigm implies the organisation of data into tables and requires data always to fit the schema. For example, if we planned to store information about surnames, this would require adding a new column to Customer, and deciding how to handle the ‘missing’ value for existing customers. A change (even minor) in the available data information would require re-organising the data model accordingly. Therefore, data can be organised with programming languages specifically designed for managing relational data, usually SQL (Structured Query Language). In the market there

¹² Gartner, data quality report, 2020

¹³ <https://openrefine.org/>

are several tools, open source or otherwise, to manage relational DBs and to query them using the SQL language. Among them, it is worth mentioning DBeaver, Microsoft Visual Studio, MySQL Workbench, Razor SQL, and Oracle SQL Developer.

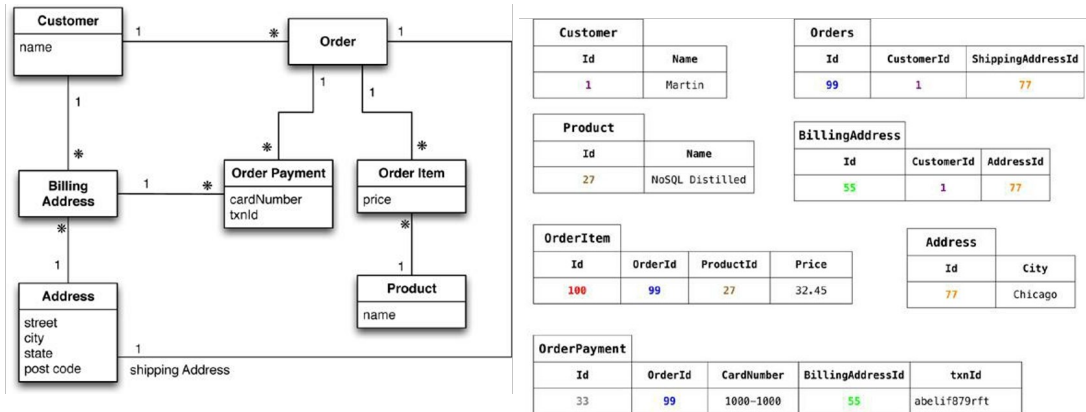


Figure 12. Left: relational schema of a sample database. Right: example of database content fitting the schema on the right. Source: Sadalage and Fowler (2013).

Unstructured data refers to data whose structure is “hidden or unknown” as the schema evolves with the data. Examples of unstructured data often include text, videos, audio files, mobile activity, social media posts, satellite imagery, surveillance imagery. These data typically arrive from *external sources*, whose structure is outside the control of the user. A common example is the case of Twitter: you can get tweets from Twitter, which provides you with information using its own structure (i.e. variables, values, information).

```
// in customers
{
  "id":1,
  "name":"Martin",
  "billingAddress":[{"city":"Chicago"}]
}

// in orders
{
  "id":99,
  "customerId":1,
  "orderItems":[
    {
      "productId":27,
      "price": 32.45,
      "productName": "NoSQL Distilled"
    }
  ],
  "shippingAddress":[{"city":"Chicago"}]
  "orderPayment":[
    {
      "ccinfo":"1000-1000-1000-1000",
      "txnId":"abelif879rft",
      "billingAddress":{"city": "Chicago"}
    }
  ],
}
```

The table content is identical to Figure 12, showing data for Customer, Orders, Product, BillingAddress, OrderItem, Address, and OrderPayment tables.

Figure 13. Left: example of a document DB. Right: example of database content fitting the schema of figure 12 (left). Source: Sadalage and Fowler (2013).

Twitter is free to change this structure, adding or removing information at any time. This, therefore, requires the user that is collecting tweets either to continuously adapt the structure of the data to fit Twitter data or to store the data in a data store that can “follow” the Twitter data schema automatically. This example should help in clarifying why NOSQL data stores (Not Only SQL) have

been growing in importance and popularity in the recent years. Following the schema example above, the structured data (right) are organised using JSON files that allow adding/removing content at any time. Hence, each “row” can have multiple and different attributes, enabling the schema to change according to the data, moving from “schema-fixed” to “schema-free”.

As a drawback, the database becomes ignorant of the relationships between the tables, which means that it is no longer able to recognise that the ‘city’:‘Chicago’ attribute of billingAddress is the same as shippingAddress.

To solve the issue of storing and querying unstructured data, as well as structured data whose schema frequently changes, in recent years, the well-known NoSQL movement brought to the fore new data model paradigms that differ significantly from the classical relational model (i.e. key-values, document databases, column-oriented and graph-db). All these new paradigms share some interesting features compared with a classical relational model, such as a flexible schema that can evolve to always fit the data (in other words, schema-free) and the ability to horizontally scale over multiple machines, a key element for dealing with big data at scale.

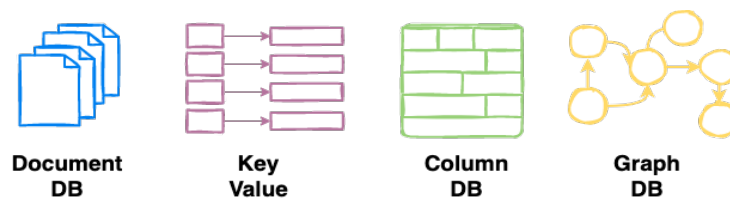


Figure 14. NoSQL data store paradigms. Source: created by authors.

DocumentDB: document database is the most used NoSQL paradigm that enables the modelling of each data row as a document, each with its own structure and schema specified through JSON or XML format. This means that there are no empty attributes within documents: if a given attribute is not found, it was not set or not relevant to the document. Furthermore, new attributes can be created without the need to define them or to change the existing documents. The schema is simply derived from the data (i.e. schema-free approach). They are particularly useful for storing data from the web (event logging, blogging platforms, storing IoT and sensor data, Web and real-time analytics, e-commerce, etcetera). Examples include Amazon documentDB and MongoDB.

Key-Value DB: key-value data model means that a value corresponds to a key (unique). Although the structure is simpler, the query speed is higher than a relational database, and this allows for supporting mass storage and high concurrency. These databases are quite useful to store session information: every web session is unique and is assigned a unique sessionId value. With a key-value store, everything about the session can be stored and accessed easily. Example: MongoDB.

Column-DB: column-oriented databases store data by columns rather than rows. This means a columnar database provides access to the most relevant elements, which increases the speed of a query even in a database containing millions of records. A column-oriented database is then quite useful for any read-intensive application, as well as to identify duplicates among millions of records. Examples: HBase and Cassandra.

GraphDB: graph-databases organise the data like a network, in terms of nodes (i.e. row of a classical relational database) and edges (relationships among rows). Both nodes and edges can have attributes that specify the semantics. Notably, graph-databases differ from document, key-value and column databases as the former are based on the concept of relationships among nodes whilst the

latter is ignorant of the relationships in the data. A query on a graph-database is a graph-traversal (i.e. a path) from one point to another. Graph-databases should be preferred to exploit state-of-the-art algorithms in (social) network analysis and graph algorithms. Examples: Neo4j and TitanDB.

Questions and Answers

About Distributed Computing. Q: What is the benefit of having NoSQL data store from a computation perspective?

A: The use of NoSQL data stores enables the schema of the data to be free of evolving (i.e., data define the schema). This, in turn, allows the use of distributed computing to speed-up the computational task over multiple machines. The cornerstone of this computational paradigm is “map-reduce”, that is inspired by the Roman “divide et impera” strategy. Simply put, mapreduce is a programming paradigm that allows dividing the computation into tasks that can be executed independently by several machines (the “map” task). Then, the “reduce” task combines together all the results.

About Data Quality and Cleaning. Q: Does OpenRefine send data to a cloud server if they remain protected locally?

A: OpenRefine works like a webserver (i.e., you open a web page) but it runs locally. This means no data are shared or transferred outside the machine. All the data and cleaning activities remain within the local machine.

Chapter 4 – Data Mining



Although the two terms are sometimes used interchangeably, data mining is a subset of data analytics. Specifically, data analytics is the general term used to describe all the actions performed, from the collection of raw data to the generation of (data-driven) knowledge. The term ‘Data Mining’ usually refers to a specific step of this process, aimed at extracting meaningful information and patterns from large sets of data recorded in one or more databases generated from the previous steps. Data mining is composed of two key elements: a set of data, managed through one or multiple databases, and the techniques used to derive patterns from them. These techniques are called machine learning or mathematical learning methods because their goal is to “learn” general rules from a set of past observations, sometimes called experience, in a bottom-up fashion. As explained in detail in Chapter 1, these methods imply a different paradigm compared to classical ones, like statistics and On-Line Analytical Processing (OLAP). Statistical analyses allow the validation of hypotheses formulated by the decision-maker, while OLAP enables users to query databases starting from an intuition or request. They both fall under the umbrella of rule-based methods, which are top-down approaches that only permit to validate the hypotheses formulated by the user. Conversely, given their data-driven nature, learning methods allow the extraction of information and the generation of predictions in a bottom-up fashion, providing new knowledge to the user who validates it. They are designed to unveil patterns from large sets of data, including unstructured ones, mining novel information to be incorporated into the iterative decision-making process.

Real-Life Example

Rule-based vs data-driven analysis

Let us imagine the following example. We are going to build an anti-spam system, able to automatically skim undesired e-mails. In the following bullets, we are going to present a rule-based and a data-driven approach to this problem, to clarify how they work.

1. Rule-based approach

In this kind of approach, the expert user defines a set of rules that, if true, usually identify spam e-mails. Some rules could be:

- a) A large amount of clickable text (hyperlinks)
- b) Special words (winner, reward, etcetera)
- c) Longest words (they increase the page rank)
- d) Number of images

The problem with this approach is that there may be more rules than the user can think of and define, and they change over time.

2. Data-driven approach

In this approach, the algorithm learns by itself which words and features usually identify spam e-mails. The user should provide a large set of e-mails and, for each one, specify if it is spam or not (i.e. the class). The learning algorithm learns which

words in the text and features (hyperlinks, images, sender provider, etcetera) are correlated with the spam and the non-spam class.

The problem with this approach is that a conspicuous amount of training data with the correct label (spam, non-spam) is needed.

Machine Learning

The term machine learning refers to a set of data-driven algorithms that learn data patterns in an autonomous manner. This usually happens in a two-step process: firstly, the algorithm is trained to learn patterns and rules from a set of data, then what it has learned is **tested** on new data. For instance, regression models can derive the functional relationship between a set of attributes and a target variable. Those models have two primary goals: **interpretation** and **prediction**.

While the former aims at identifying regularities in the data that can be understood and used by domain experts, the latter is drawn to anticipate the value of one or more random variables to estimate the probability of future events. For instance, a company in the manufacturing industry might be interested in using machine learning to analyse data from various kinds of sensors. Interpretation analysis unveils the rules correlated with a certain family of faults, such as vibrational or temperature increments of some component, allowing the user to identify the causes of the problem and to take action against them. At the same time, prediction analysis could anticipate a fault event, allowing timely maintenance to be scheduled when needed.

Before delving into machine learning methods, it is important to understand the main logic behind the way they function. To do this, we need to subdivide learning models into two main groups: **supervised** and **unsupervised** algorithms. The difference lies in the fact that supervised learning, as the name suggests, is guided by a target attribute representing the class of the record, which acts as a supervisor, or a teacher, for the algorithm.

Supervised machine learning

In supervised machine learning, the algorithm is trained using a target attribute as a supervisor. This attribute is, for each observation in our data, the real value of the dependent variable we want to understand or predict. For this reason, the target variable is also called ground-truth. For instance, in the above examples, the ground-truth is represented by the binary variable indicating whether some values measured by the sensors caused a failure or a malfunction in the system. The value of the sensors in this case represents the features, or independent variables, usually denoted with the X letter, while the fault/not fault variable is the target or dependent variable, usually denoted with y . Given the ground-truth, a supervised learning algorithm is trained to learn the relations (function) between the values of the feature attributes and the values of the target attributes. Once this relation has been learned, it can be analysed for interpretation purposes, or it can be applied to new data (for which we do not know the value of the target function) for prediction. In the previous example, the values measured by the sensors for every week of the last year can be used as features. Each week is an observation, and for each observation the target value is a binary variable indicating whether the system malfunctioned in the following week or not. If a machine learning algorithm is trained with those data, the relation between the value of the sensors and the target

variable allows to understand which are the physical values (temperature, vibration, etcetera) that are most correlated with fault events (interpretation). But what if the aim is to understand the chances that the system will break next week? The learned function can be applied to the features measured in the current week, for which we do not have a target value, because an eventual break might happen in the future. In this case, the algorithm is going to output a prediction on the values of the target attribute, based on what it has learned from past data.

The problem of data labelling. In supervised learning, having a correct and meaningful target attribute is necessary because it is the ‘teacher’ of the machine learning algorithm. However, having high quality labelled data is one of the main challenges in data mining. In the aforementioned example, the labels are produced by collecting the history of failures and the related values measured from the sensors. While it is not always easy, even in today’s organisations, to have in place an integrated system for data collection among the company areas, and important data is often not stored (and therefore lost), in this case, if the historical data is collected correctly, we can be sure of its veracity. However, in several cases the quality of the labelling is not guaranteed, and this can be critical for the performance of the model. Following the well-known ‘garbage-in, garbage-out’ principle of data quality, the more the labels move away from the real value of the target variable, the less the output of the supervised model will be reliable. Who assigned the labels? Through which process? Which difficulty did they encounter? Usually, labels can be either the output of a previous decision process or manually annotated (Cowie et al., 2011; Monarch, 2021; Pustejovsky & Stubbs, 2012). Neither of these processes is exempt from errors. An example of error in the first case is the bias in historical data. Let us suppose that we want to train an algorithm to help the institution make loan-granting decisions, and we rely on historical data, where the label is granted/not granted for each request. If there is a bias against some features of subgroups of applicants (i.e. gender or ethnic groups), the algorithm is going to learn those biases and systematically make unfair decisions again. An example in the second case are the diseases specified in discharge letters of hospital patients. In several cases the classification task is carried out by hospital operators, who are under considerable pressure and affected by time constraints, making this task an error-prone and time-consuming activity. Finally, it is worth noticing that the manual labelling activity is not only error-prone, but also costly and time consuming, especially in the case of big data. For instance, if we want to have Online Job Advertisements labelled with the corresponding occupation of the official European taxonomy, ESCO, we need to label a considerable number of advertisements for each of the possible values of the target variable, i.e. the 426 occupations of the taxonomy. Moreover, because the labour market is evolving over time, these data labels need to be monitored and updated.

Real-Life Examples		
Target variable for examples of machine learning applications.		
In case of unsupervised applications, there is no target variable.		
Data	Example of use	Example and type of the target variable
Sensors and IoT Machine data	Predictive maintenance	System up Yes/no (Binary)
Radio Frequency Identification (RFID) Purchasing history	Basket analysis	-
Commuters' data Commuting time	Monitoring living and working conditions	Working turnover Years (numeric)
Hospital Discharge Letter Health records	Monitoring the correct tracking of diseases of patients	Correct disease class Standard taxonomy (categorical)
Online Job Ads Job descriptions and CVs	Job matching	Job/CV matching Yes/no (Binary)
Banking data Frequency, GPS data, amounts	Fraud detection	Fraud transaction Yes/no (Binary)

Types of supervised machine learning algorithms

Supervised machine learning algorithms can be further divided into two groups, based on the type of the target variable. Regression algorithms aim to predict the value of a numerical target attribute. On the contrary, classification algorithms have a categorical target variable, which can assume two or more discrete values called classes. The goal of these algorithms is to predict the correct class of observations. For instance, in predictive maintenance, a regression algorithm could be trained to predict the number of days before the next failure, while a classification algorithm could be trained on two classes to predict if tomorrow the system will be down or up. Examples of well-known

families of classification algorithms are classification trees, support vector machines, Bayesian classifiers, logistic regression, ensemble methods such as random forests and neural networks.

Evaluation of classification models

After a classification model has been trained, it is of primary importance to evaluate how well it can predict the target class of additional, unseen data. The goodness of the model has a dramatic impact on the reliability of the decision made based on it. When training a learning model, it is good practice to try different algorithms or different hyper-parameters, then choose the one with the best performance. In this sense, what is usually advisable is to have a model that performs well on new data, rather than on the data used for training, to reduce risk of overfitting. Classification models cannot be evaluated with the same metrics and instruments used for regression, which usually measure the distance between the prediction and the real value of the target variable, because the output is categorical. For this reason, they are usually evaluated through ad hoc metrics, which can be inferred from a decision table called **confusion matrix**. In case of binary classification, a confusion matrix is a two-entry matrix which compares the real labels of our data and the ones predicted by the classifier.

Confusion Matrix		Actual Values	
		Negative (-1)	Positive (+1)
Predicted Values	Negative (-1)	TN=990	FN=5
	Positive (+1)	FP=0	TP=5

Table 2. Example of confusion matrix. Source: created by authors.

Table 2 shows the structure of the confusion matrix. The rows correspond to the predicted class, while the columns are the real values. The two rectangles (bottom right and top left) represent the number of instances that have been correctly classified. **TP (true positives)** in the bottom right corner is the number of instances belonging to the positive class that have been classified as positive. Similarly, **TN (true negatives)** is the number of negative class elements classified as class negative. In the red rectangles, the wrongly classified observations are marked. **FN (false negatives)** are the positive instances classified as negative, meaning that the algorithm classified them as negative, but they are not. In the same way, the **false positives (FP)** are the observations classified as positive, but that are negative.

From this matrix, we can compute four important metrics:

Accuracy: the ratio of correctly classified observations over the total number of observations	$\frac{TP + TN}{TP + TN + FP + FN}$
Recall: the fraction of positive instances over all the instances classified as positive	$\frac{TP}{TP + FN}$
Precision: the fraction of instances classified as positive over all the positive instances	$\frac{TP}{TP + FP}$
F1-score: harmonic mean of precision and recall	$2 \times \frac{Precision \times Recall}{Precision + Recall}$

To better understand the meaning of these four metrics, let us observe the figure on the right. The outer square contains all the observations while the inner circle contains the observations classified as positive by the algorithm, also called **retrieved** elements. The observations on the left side are the **relevant** items, i.e. the items belonging to the positive class, while the elements on the right are the negative ones.

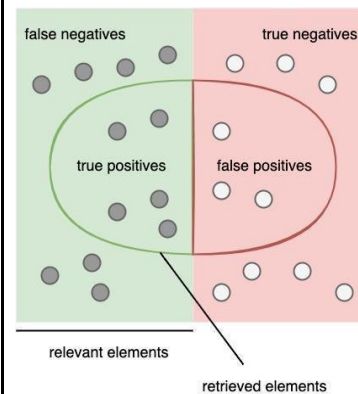
The **accuracy** thus represents the fraction of correctly classified observations, both positive and negative, that is, the ratio between the elements in the circle and the entirety of the observations.

The **recall** is the fraction of relevant elements that are retrieved.

The **precision** is the fraction of retrieved elements that are relevant.

The **F1-score** is the harmonic mean between precision and recall.

Recall, precision, and F1-score are calculated in relation to one class, the positive one. They are class measures and can be similarly computed for each class. Accuracy, to the contrary, is a unique measure for all the classes.



How many retrieved items are relevant?

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

How many relevant items are retrieved?

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

Source: created by authors.

Accuracy is the most used measure in classification because it gives a synthetic result for all the classes. However, sometimes it is not enough, since the good performances on one class can obscure poor performances on the other, especially if the dataset is unbalanced, i.e. the numerosity of one class is much greater than the other. For instance, in Table 2, the accuracy is 99.5% because 995 observations are correctly classified and five are not. However, the performances on class +1 are rather poor, because of the ten accuracies belonging to the class, only five are correctly classified, that is, the recall on the positive class is only 50%.

The decision to give more importance to precision or to recall depends on the problem being analysed. For example, in the case of a classifier that should suggest whether a certain patient should undergo surgery or not, a low recall would mean that few of the positive observations, the patients in need of surgery, have been retrieved. In other words, many people who need to undergo surgery are discharged. If the precision is low, it means that more people than the ones in need of surgery are kept in the hospital and undergo more medical examinations. In this case, the cost of a false negative, i.e. not retrieving a relevant element, is higher than the one of the false positive, i.e. subjecting a patient who does not need surgery to more tests. Therefore, in this case, recall is more important than precision. In a different case, for example a classification algorithm put in place to detect flawed objects in a production line, the reasoning would be different. If the production cost of the item is particularly low, checking if an item is flawed might cost as much as the value of the product itself. It may make sense economically to inspect only the pieces that are surely flawed and leave some faulty pieces to be sold and, eventually, returned by the final customers. Consequently, a high ratio between retrieved elements is desirable, (marked as broken by the classifier) and relevant ones (actually broken). In this case, precision should be preferred to recall.

An example of classification algorithm

A family of simple yet powerful classification algorithms are *classification trees*. These algorithms derive a set of hierarchical rules to split the observations into the target classes. Let us consider as an example a sentiment classification problem, which is a problem aimed at understanding if an online review expresses a positive or negative opinion based on the words included in it. Specifically, let us imagine analysing hotel reviews posted online.

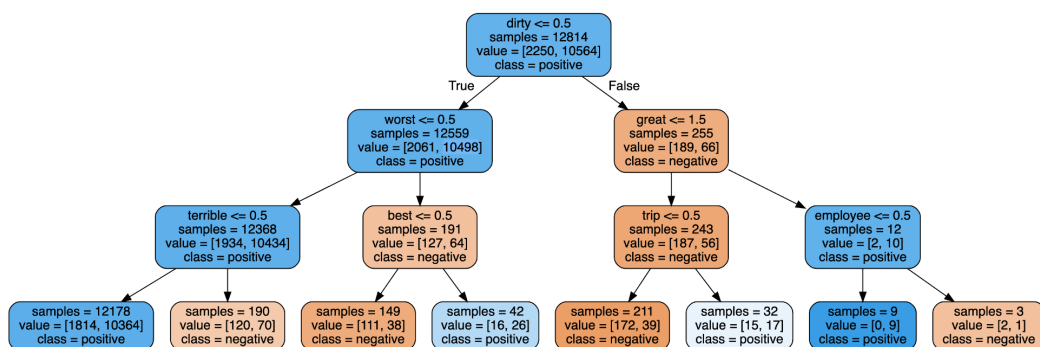


Figure 15. Example of a decision tree classifier. Source: created by authors.

Figure 15 shows a decision tree model trained to classify 12,814 reviews into two classes, positive and negative. As this is a text mining problem, the features are the words contained in the reviews. The top node of the tree, called root, contains all the observations. The *value* field indicates that in this node there are 2,250 instances of class negative and 10,564 of class positive. Therefore, the

node is labelled as class *positive*. The first decision criterion is based on the word in the top of the node: 'dirty'. If the word dirty appears in the review not more than 0.5 time (i.e. if it does not appear at all), we proceed on the left branch (*true*), otherwise on the right one (*false*). The number of samples on the following level of the tree tells us that the word 'dirty' appears in 255 reviews and does not appear in 12,559. If we follow the right branch, we find a second criterion based on the word 'great'. If it appears twice or more (≤ 1.5 is false), we follow the right branch again, otherwise we go for the left one. We follow this process of simple binary criteria until we find a terminal node. In these nodes, called leaves, we find the final classification of the number of samples indicated. Let us observe the bottom left leaf. It classifies 12,178 samples. Among them 1,814 are negative and 10,364 are positive, thus the class of this leaf is *positive*. The reviews in this leaf respond to the three criteria followed to get here: the word 'dirty', 'worst' and 'terrible' are not in the review: this is the **classification rule**. In this example there are eight different rules, each following three classification criteria. Four leaves are positive (different shades of blue), and some are negative (orange). All the negative examples in positive leaves, like the 1,814 in the bottom left node, are misclassified, and specifically false positives. Similarly, in the second leaf from the left, there are 70 false negatives. With these values for all the leaves we can compute the accuracy, recall, precision and F1 of this classification.

Complexity of the model and overfitting

Modern machine learning methods can be so powerful and complex that they excessively reflect the peculiarity of the examples in the training set and are therefore less capable of generalisation. This phenomenon, called 'overfitting', produces learning models based on the specific characteristics of the training data, rather than on the general features of the classes we want to discriminate against. Henceforth, a model which overfits the training data will most likely not be generalisable, with a consequent drop of performances on unseen data.

To better explain this concept, we can use the example of the decision tree above, reducing it to two extreme cases. The first is the case in which we force the tree to have just one level:

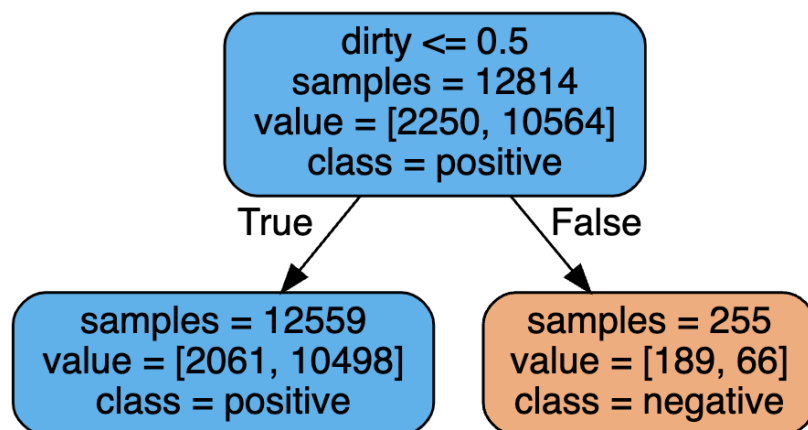


Figure 16a. Example of underfitting in classification trees.
Source: created by authors.

In this case, we have one single rule composed by a single criterion. On the left branch we have 12,559 reviews that do not contain the word dirty. The model does not specify any further rule and

assigns those elements to the positive class, without considering any word contained in the review but the word 'dirty'. This simple algorithm is **underfitting** the data because it provides loose rules that, in many cases, will not be able to discriminate between classes. As an example, consider the following review: 'I hated this place. Staff were rude and the breakfast was disgusting. On top of this, we paid for the room like a 5-star suite'. This review, clearly negative, does not contain the word 'dirty', thus would be classified as positive.

To explain overfitting, let us for instance consider the following slice of a complex tree (the whole tree would be difficult to visualise):

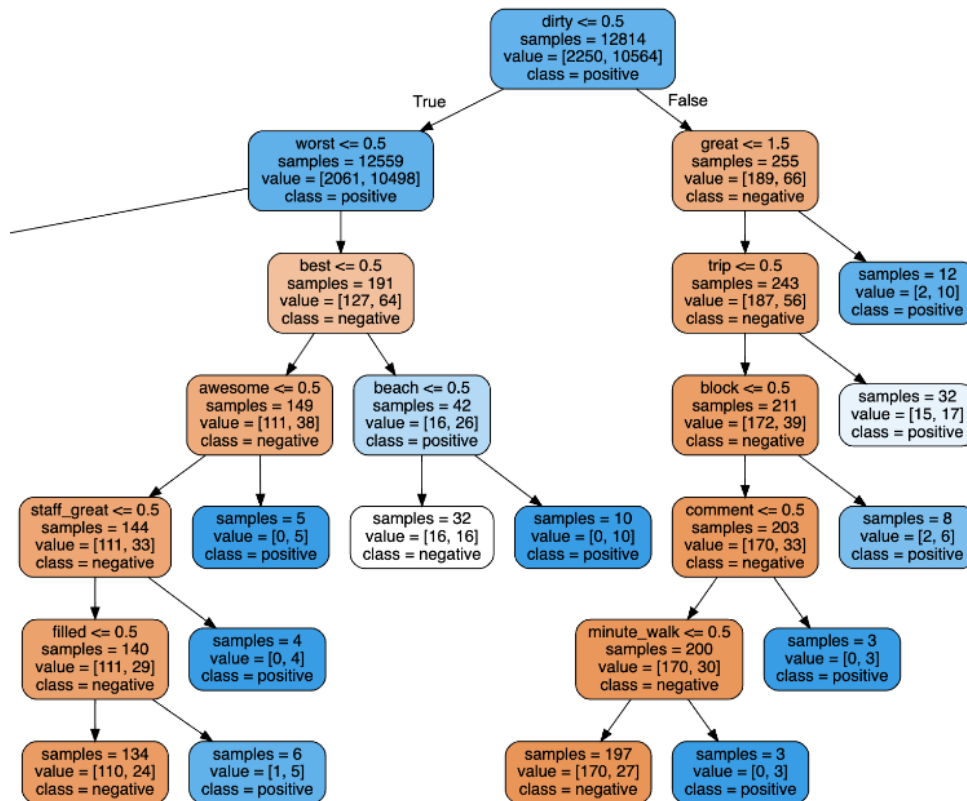


Figure 16b. Example of overfitting with classification trees

Source: created by authors

The bottom right leaf is reached with the following rule: there should be the word 'dirty' and twice the word 'great', then there should be the words 'trip', 'block', 'comment' and 'minute walk'. If all those conditions are met, we classify the review as positive. However, this rule is extremely specific, and it is difficult to find a review meeting all the criteria. Indeed, only tree samples follow this rule. Modern machine and deep learning methods are so powerful that they could even generate a rule for each observation in the dataset and get 100% accuracy on training data. The problem with **overfitting**, however, is on the test set. New online reviews will unlikely contain all the words in the rules represented above. For this reason, test performances would drop dramatically. The choice of hyper-parameters, like the number of levels in classification trees, is as important as the choice of the model itself, if not more, and should always be performed looking at the performances on test data.

Unsupervised Machine Learning

In unsupervised machine learning, the algorithm is not guided by a target variable, but aims at finding recurring patterns and similarities in the database. As an example, think about a retail corporation that wants to identify groups of customers displaying a similar purchase behaviour, based on their past sales receipts. On the one side, this would enable them to plan marketing campaigns targeted to each cluster and tailored to their usual shopping behaviour. On the other hand, a product usually bought by some members of the cluster could be proposed to all the other members of the same group because they usually buy similar products.

Examples of two well-known families of unsupervised machine learning algorithms are **clustering** and **association rules**. Clustering algorithms are used to create groups of similar observations (clusters) based on their features. Association rules are used to find correlation and co-occurrences among dataset elements (for example shopping baskets with similar purchase patterns).

Evaluation of Unsupervised Machine Learning

For unsupervised machine learning, there is no ground-truth to use as a benchmark. Usually, these methods are evaluated on how well they can create groups of observation that have internal homogeneity and maximum inhomogeneity with other groups. For instance, a common indicator for the evaluation of the quality of clustering is the silhouette coefficient, defined as:

$$sil(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

The value $a(i)$ is the average distance between the observation i and all the elements in its cluster, while $b(i)$ is the average distance between i and the closest cluster to which i does not belong to. The silhouette score is bound between -1 and 1, and the greater it is, the better. If the silhouette is negative, it means that there is a cluster which is not the one of i , but is closer.

Supervised and Unsupervised: A Visual Example

Looking at figure 17, the same observations can be made, but while some of them have a class label (colour) in the right plot, they do not in the left one.

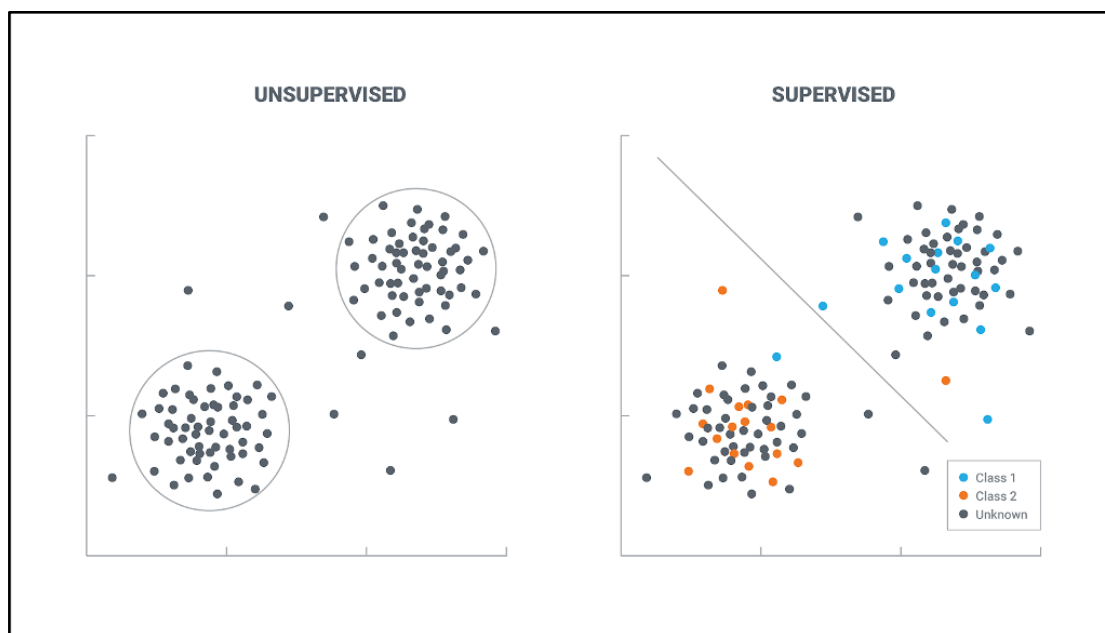


Figure 17. Supervised vs Unsupervised Classification (source: <https://www.informatec.com/en/machine-learning>).

A supervised learning algorithm takes as input the features (location in the space) of the labelled observations, i.e. the orange and blue ones, and builds a function (or model) that distinguishes between the two (the straight line separating the space on the right). Based on the learned model, a label can be assigned to the observations of unknown class, hypothesising that the points above the separation line belong to class 1 and the ones below to class 2. If we do not know the classes of some observations, we can use an unsupervised approach (left plot) and find a function that separates the points in the clearest way possible based on their position in the features' space (the two circles). In this case, we do not know what these two groups represent, since we do not have the supervision of the labelled orange and blue points to tell us the class of similar points.

Deep Learning

The term Deep Learning indicates **a family of machine learning methods, falling under the umbrella of neural networks, with multiple levels of representation**. The first part of this definition indicates that deep learning methods, like all other machine learning methods, are data driven, that is, they learn rules and patterns from previous data, without being explicitly programmed. The second part explains how those methods differ from classical machine learning. Traditional machine learning algorithms are fed with a set of features and output a value for the target variable. The input data of the learning algorithm can be imagined as a table, where each row represents an object, or observation, and each column represents a particular feature that all the observations possess. In the predictive maintenance example (if a system breaks down or not, see above), one column would represent the intensity of the vibrations, one the temperature, etcetera. Those features are collected or generated by the user and, in some cases, engineered by a domain expert, in the sense that multiple features can be blended if it is considered necessary. For instance, when classifying job adverts based on the skills found in the description, one could group together similar features in a single representative one ('feature engineering').

However, in some cases, 'feature engineering' is not feasible. In image classification, for instance, each pixel represents a feature, and modern images can be composed of millions of pixels.

Moreover, the target object is not always in the same portion of the image, making it impossible for the user to tell which pixels grouped together could form a meaningful representation.

On the contrary, neural networks are methods that allow a machine to be fed raw data and to automatically discover the representations needed for detection or classification. This is possible because they are composed of different layers. In the first layer, the initial features are inputted (in the above example, pixels and their colour). The features are then blended into a second layer, whose number of features is determined by the user. These intermediate features are called neurons, and the information flowing from one neuron to another is called activation, since the architecture of neural networks is loosely inspired by the architecture of the human brain. Information then flows from the first layer to the second, from the second to the third and so on up to the last one. The features (neurons) of the last layer are then used as input features to classify the observation in the target classes. These layers are called hidden, because it is the algorithm itself that decides how to combine the initial features to generate new intermediate ones. As shown in figure 18, the algorithm learns different intermediate features that combine together the input pixels into shapes that are specific to the three output classes (car, person, animal). For classification tasks, higher layers of representation amplify aspects of the input that are important for discrimination and suppress irrelevant variations. This is particularly true when the number of features is high, like in image classification or in text classification, where each possible word represents a different feature. In these cases, deep learning algorithms have proven to be disruptive compared to classical machine learning, enabling a number of important applications including machine translation, autonomous drive and so on. For this reason, this family of algorithms has earned its own name, 'Deep Learning', where the term "deep" refers to the multiple hidden levels of representation.

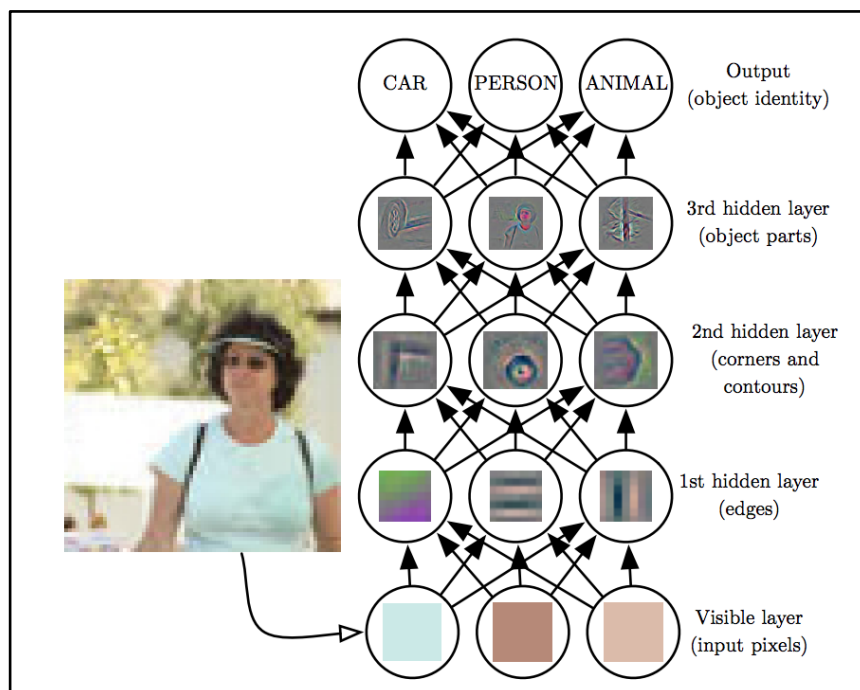


Figure 18. Exemplification of a Neural Network (source: <https://www.deeplearningbook.org/>)

Explainable AI (XAI)

The recent preponderance of machine-learning methods in data analysis has fostered the culture of overemphasising model performances, rather than studying their rationale and the reasons behind their errors. Machine learning and deep learning models provide a response to the input data they are fed, building complex learning models that are opaque to humans. However, the notion of AI models as black boxes strongly limits their acceptance in several fields, especially in mission-critical ones, such as medicine, law, finance and many others, which 'directly affect the health, safety and welfare of the public and in which failure could cause loss of life, serious injury or significant loss of assets or privacy' (Laplante et al., 2020). Garnering the trust of domain experts and final users is as important as the model's performance, and this cannot be separated from the notion of interpretability.

In 2017, the American Defence Advanced Research Projects Agency (DARPA) introduced the explainable AI (XAI) initiative, designed to create a toolkit of machine learning techniques to produce more explainable models to garner human users' trust and foster the adoption of the latest generation of artificial intelligence technologies. XAI is becoming crucial in several industries and applications.

The GDPR (General Data Protection Regulation), adopted by the European Union in 2018, defines the right to explanations (Art. 13-15), asking the data controllers to provide data subjects with information about 'the existence of automated decision-making, including profiling, referred to in Article 22(1) and (4) and, at least in those cases, meaningful information about the logic involved, as well as the significance and the envisaged consequences of such processing for the data subject'. The focus is on the adjective meaningful, as this implies that any stakeholder should be able/put in a position/supplied with the information to understand the logic behind an automated process, even without any technical knowledge.

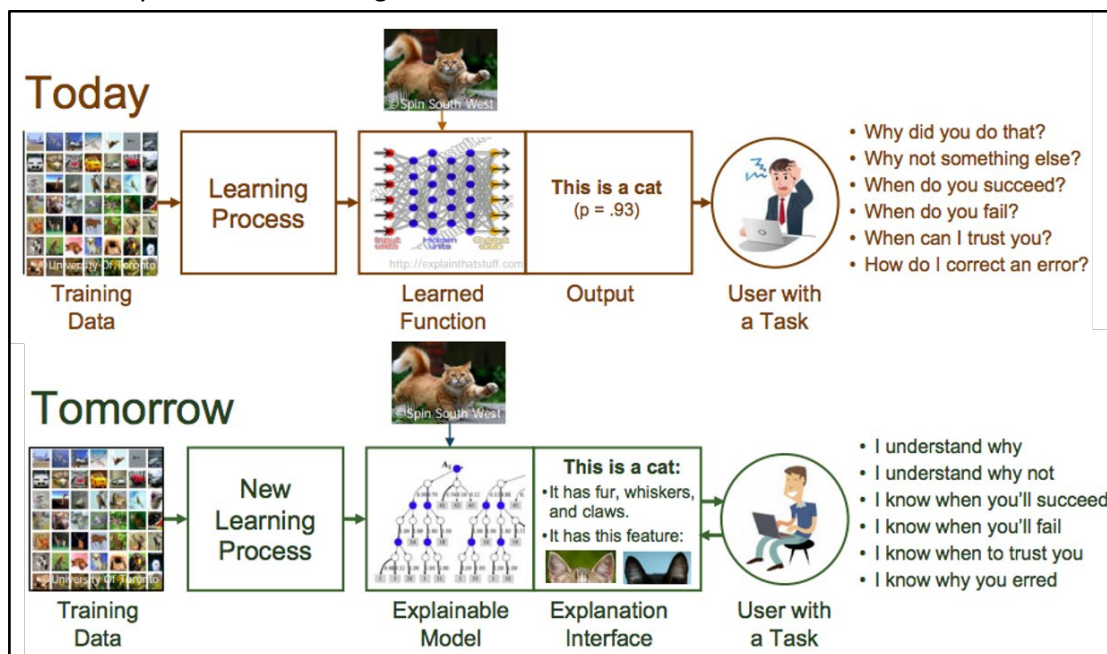


Figure 19. Graphical representation of XAI (source: DARPA)

As the goal of any big data analytics project is to obtain knowledge from data to *support human decision making*, the role of XAI becomes key, as XAI can effectively support the decision maker in understanding the rationale behind a machine's suggestions or decisions. Not surprisingly, interest in

XAI has been increasing in the recent years. A very practical example is in the field of recruitment, criminal justice, and healthcare, where the massive use of AI has stirred a debate about bias and fairness. Yet, human decision making in these, and other, domains can also be flawed, shaped by individual and societal biases that are often unconscious. Will an AI's decisions be less biased than human ones? Or will an AI make these problems worse? As machine-learning algorithms use historical data to learn, how can the learning process be tuned to avoid incorporating historical unfair and biased decisions? And most importantly, how can it be guaranteed that AI decisions do not hide novel (unknown) biases? The goal of XAI is to address these issues to empower the so-called "Human-AI teaming" in decision making. Not surprisingly, in late 2021 the European Commission clarified the key role of XAI to create a human-centric and trustworthy AI in the EU approach to AI.¹⁴

¹⁴ Fostering a European approach to Artificial Intelligence. European Commission, COM(2021)

Chapter 5 – Interpretation and Evaluation through examples



In the final step of the Big Data Analytics Lifecycle, the information flowing from the previous steps is turned into knowledge and used to make business decisions. Consequently, this is the step that, more than the others, implies the engagement of the expert user, i.e. a user with expertise in the domain but not necessarily knowledgeable about using big data. In this section, we are going to present the two founding steps in this process: the *interpretation* and *evaluation* of data through visualisation, and the impact of data-driven decisions explained by successful use cases.

Data visualisation

What is the best visualisation for the output of a data analytics process? The answer is not univocal, because data visualisation is not only a tool for showing results, but a way to communicate the results. In particular, **visual analytics** is the science of “*analytical reasoning facilitated by interactive visual interfaces*” (Thomas & Cook, 2006). Visualising data in charts, graphs or maps helps the decision maker and the audience in general to identify patterns and thereby develop data-driven actions. This, in turn, helps organisations make better decisions. Therefore, before deciding which kind of visualisation to use, it is important to evaluate the four questions below:

What? What kind of data do I want to present? How many dimensions does it have? Is it structured or unstructured? How many samples do I need to present?

Who? Is the final user an internal stakeholder, i.e. working for a different division of the same organisation, or an external one (e.g. European or national level policy makers, social partners)? Are they familiar with the topic? Do they understand the data underlying the visualisation? Do they have a good visualisation literacy?

Where? Where is the explanation going to be displayed? What is the cultural context? Is the audience well-acquainted with big data analytics? Are there time or format constraints to be met?

Why? What is the objective? And what message do I want to communicate? Do I want to explicitly show the main insights or let the user explore the data?

Data visualisation types

The **most common types of data visualisation** (presented below) are divided into three groups: Graphics/plots, Texts, Images and Reports.

Graphics/Plots include the most popular methods in the literature. Such popularity is rooted in the presence of tools and the relative simplicity of generating such graphics. This group consists of the following types: Bar plots, Line plots, Trees, Heatmap plots, Histograms, Scatter plots and Bubble plots.

1. Bar Plots are the most used method in this category and can be further divided into Horizontal and Vertical Bar Plots.
2. Line Plots vary from simple vertical bar plots to sophisticated custom plots made to represent a particular subject, often mixed with other types of methods, such as destiny plots, or changing hue to add additional attributes.

3. Heatmap Plots (not to be confused with Heatmap Images) are simple yet immediate visual representations that show a numeric range with a colour scale. Figure 20 (left) shows the number of monthly flights in US between 1949 and 1960. On the vertical axis there are months and on the horizontal one years. The intersection cell represents the number of flights for a given month in a given year. The right bar indicates the colour grades and the corresponding values, which in this case are also reported inside the heatmap cells. The colour grades allow viewers to immediately perceive the values represented.
4. Histograms look quite like bars, but they represent the empirical distribution of the targeted variable. Among the presented methods so far, they are the most technical methods as their interpretation might be complicated for the lay user without statistical knowledge.
5. Scatter Plots map two variables into a two-dimensional space, where each observation is represented as a dot. In addition, a third categorical dimension can be represented by the different colours of the dots, usually called hue.
6. Bubble Plots, visually similar to scatter plots, can be considered as an augmented version of scatter plot where a fourth numerical dimension is represented by the size of the bubble (dot). For instance, in figure 20 (right), visualising some car features, the size of the dot represents the weight of the car, the vertical axis its consumption in miles per gallon, the horizontal one the power in HP while the hue (colour) represents the production country.

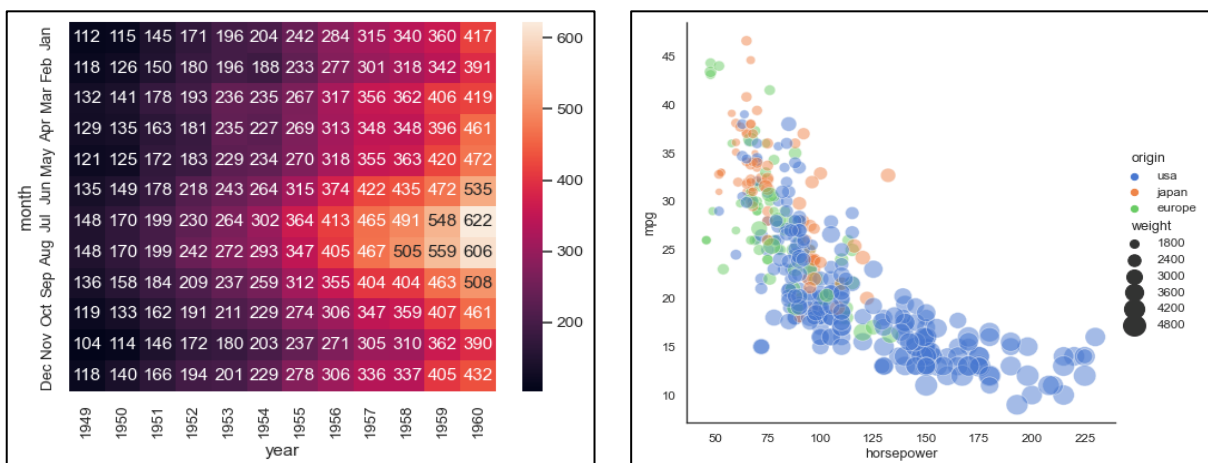


Figure 20. A heatmap with colour grades and values (left) and a bubble plot with varying point size and hue (right). (Source: <https://seaborn.pydata.org/>)

Images. Image-based presentations are considered more sophisticated than the previous group (plots/graphics) and, at the same time, they are more limited since they can be applied only if the target input is an image. The main types in this category are image heatmap, saliency masks and image manipulation.

7. Image Heatmaps, not to be confused with heatmap plots, use an image as their base and add different layers of visualisation, mostly coming from continuous data. The visualisation layers usually are correlated with the task performed.
8. Saliency Masks are similar to image heatmaps as they utilise an image as their basis, but instead of adding heatmaps of values, they partially mask/cover the image. A difference from image heatmaps is that saliency masks usually represent characteristics of the image itself and are not related to a particular task. For instance, in figure 21, the second image

(heatmap) might represent the features used by a classifier to classify the parrots, while the third one shows the features with greater sharpness.

9. Image Manipulation, being the least sophisticated method in its family, consists of adding indicator shapes to an image in order to indicate a specific part of the image.



Figure 21. Example of a picture (left) with associated image heatmap (centre) and saliency mask (right). Source: (Le Meur & Baccino, 2013). The image heatmap usually represents features correlated with the task performed. For instance, the colour scale could represent the importance of the portions of the image to classify it as parrots. The saliency mask usually represents perceptual features of an image, i.e. where an observer would look independently from any task performed; in this case, for instance, areas with greater sharpness.

Reports. Although this family is close to the text category (described below) reports have a more structured approach compared to texts and often are combined with other methods (e.g. graphics). The main techniques in this category are tabular reports, decision tables and graphical table reports.

10. Tabular Reports. The most basic method in this family reports and conveys the desired message in a structured and direct manner.
11. Decision Table. Like tabular reports, decision tables use the tabular structure, but since they solely represent the rules and mostly no other info, they have less flexibility in the data types and other representations.
12. Graphical Table Report. This method, which uses tabular reports as basis, integrates other methods in a very flexible way which allows the customisation of the table based on the specific message to be communicated.

Texts. This group contains methods that use text as their basis. Please note that it does not mean that the output of these representations is necessarily expressed in natural language, but it indicates that the main message is conveyed through text and not the other techniques mentioned above.

Data visualisation tools

Data visualisation tools can be code tools or visual tools.

Code tools

Code tools make use of ad hoc libraries, such as *matplotlib*, *seaborn*, *ggplot2*, *plotly*, and R shiny. One advantage of code visualisation tools is their flexibility. They do not have a set of fixed options but can be customised as needed. Moreover, code tools can be integrated directly into the data mining pipeline and are developed to handle complex data. However, the drawback of using tools of this kind is that their flexibility comes at the cost in terms of design time and that coding does not allow a direct visualisation of the underlying data.

Visual tools

Excel, *Tableau* or *RawGraphs*, *PowerBI*, both as desktop apps and online services, are examples of visual tools. They are immediate tools as they allow to see the data source, usually in the form of a

table, and the output plot. Once the data is loaded, the user can easily create visualisations, reports, and dashboards through the user interface.

Use cases

This section provides some examples of applications of BDA to different domains. Each use case reports a description of the problem to be addressed, the benefits for the organisations, the data used and with the challenges that were addressed. For each case, whenever possible, we provide external details and references.

Sentiment analysis of customer reviews

Nowadays, customer reviews and online ratings have a tremendous impact on purchasing behaviour. Online buying is gradually replacing in-store shopping in several industries, including, but not limited to, fashion, electronic goods, and food. The same holds for many services, which are sold and managed completely online. Some examples are bank accounts and insurances. Before concluding a purchase, usually customers carefully examine online reviews to make a pondered choice. Furthermore, even in the case of in-store shopping, customers could collect information and opinions online before visiting the shop, to get more information about the product and to choose the best brand or retailer to visit. Therefore, online reviews are not only a driver for customer choices but also a valuable source of information for companies. Having positive reviews is important for the reputation and exposure of the company and drives increased sales. Moreover, customer reviews are a free source of information that can be a great source of knowledge for the company.

Sentiment analysis (SA) studies the recognition of opinions, feelings, and emotions from text objects, usually using AI tools and machine learning. Online reviews are collected and analysed, usually by product or service type. The text of the review is transformed from natural language to a tabular format, which can be used as input for a machine learning algorithm. Usually, the presence or absence of a word in a sentence corresponds to a feature. If the reviews have a related numerical rating, it can be used as the target value to train the algorithm. The simplest case is the binary one, where high values of rating correspond to the positive class and low ones to the negative. Once the algorithm is trained, it can be used in several ways. For instance, if there are reviews without a rating, such as social media comments, the pretrained model can be used to predict the sentiment. For a recent survey on sentiment analysis, see Birjali et al., 2021.

Benefits

The benefits of analysing customer reviews are multiple. The main ones are listed below.

1. Timely and improved understanding of customers, their needs, and complaints.
2. Timely and improved understanding of pros/cons of competitor and company products.
3. Timely and improved understanding of pros/cons of competitor and company services.
4. Savings in questionnaires and other costly and time-consuming customer/user feedback systems, allowing staff to save time and energy for higher-level tasks.

Data

Large quantities of online reviews, preferably with a label indicating the sentiment of the comment.

Challenges

1. Natural language varies across domains. The meaning of several words is often related to the specific context they are used in, and each context has its own metaphors and special

words. Moreover, the human language also varies over time. Therefore, the collection of data for sentiment is time and domain dependent.

2. Collecting social data, both through scraping and APIs, should comply with the applicable privacy regulation.
3. It is not easy to find labelled reviews and comments in every domain.

Classifying hospital discharge letters

Discharge letters are used within hospital Information Systems to track patients' diseases during their hospitalisation. Such records are commonly classified over the standard taxonomy set out by the World Health Organization, the International Statistical Classification of Diseases and Related Health Problems (ICD-10). Classifying discharge letters with the correct code is crucial to allow hospitals to be refunded by Public Administrations based on the health service provided. The goal is to use machine learning techniques to classify discharge letters into the class representing the correct disease.

Benefits

In many practical cases, the classification task is carried out by hospital operators, who are often working under pressure, making this task an error-prone and time-consuming activity. This process might be improved by applying machine learning techniques to reduce human error and, at the same time, accelerate the process.

Data

Historical DiL classified by hospital operators according to the ICD-10 taxonomy

Challenges

1. Data quality: historical data, as mentioned above, might contain errors that, if not processed and amended, are used as examples by the classifier to build the classification models, which will be biased and will repeat those errors when used.
2. In these kinds of sensitive processes, the interaction between human evaluators, data mining techniques and patients who will receive their outcome must be carefully designed to garner the trust of all the actors involved. The human evaluator, usually the doctor, should be able to understand the reasons behind the predictions of the algorithm and the patient should be aware of how their data is being used.

Alignment of national taxonomies in the labour market

The AI4ESCO action (Giabelli et al., 2022), developed within the EU Programme for Employment and Social Innovation, EaSI (2014 - 2020)¹⁵, used AI for the construction of a high-quality mapping table between the Italian (CP) and the European (ESCO) job taxonomy. The need for common policies across EU states, coupled with the internationalisation of labour demand and supply, increased the importance of having a common language that allows EU member states to communicate about the labour market. ESCO is the official labour market taxonomy of the European Union. It identifies skills/competencies, qualifications, and occupations relevant to the EU labour market, education, and training in 25 European languages.

15 <https://ec.europa.eu/social/main.jsp?catId=1081>

AI4ESCO, which has already been developed and deployed for Italy, is a general AI-based method that automatically maps two different taxonomies, namely ESCO and CP-2011 (i.e. the Italian labour market taxonomy). The mapping was done automatically by an AI algorithm and the results were then validated by domain experts. It acts as a translator from the national labour market languages to a common language. AI4ESCO followed the scheduled activities, namely (i) to train the algorithm to encode the similarities of both CP2011 and ESCO (occupation pillar) taxonomy; (ii) to suggest a list of matches between the classifiers; and (iii) to use experts to validate the results and to estimate the final accuracy of the approach.

Benefits

AI4ESCO provides (i) a bridge between CP-2011 and ESCO that has been produced by a machine trained to encode the lexicon similarities of both classifiers with a high accuracy (>80%). This, as a consequence (ii) reduces the human effort required to build a map from scratch, concentrating the human-effort on validating the approach. For the whole EURES community, it is: (i) an approach that is domain-dependent (i.e. it can be used to bridge from/to the ESCO occupation pillar) but language-independent as it can be applied to any other language encoded in ESCO; (ii) a formalisation that clarifies all the steps and allows for the reproducibility and transparency of the approach.

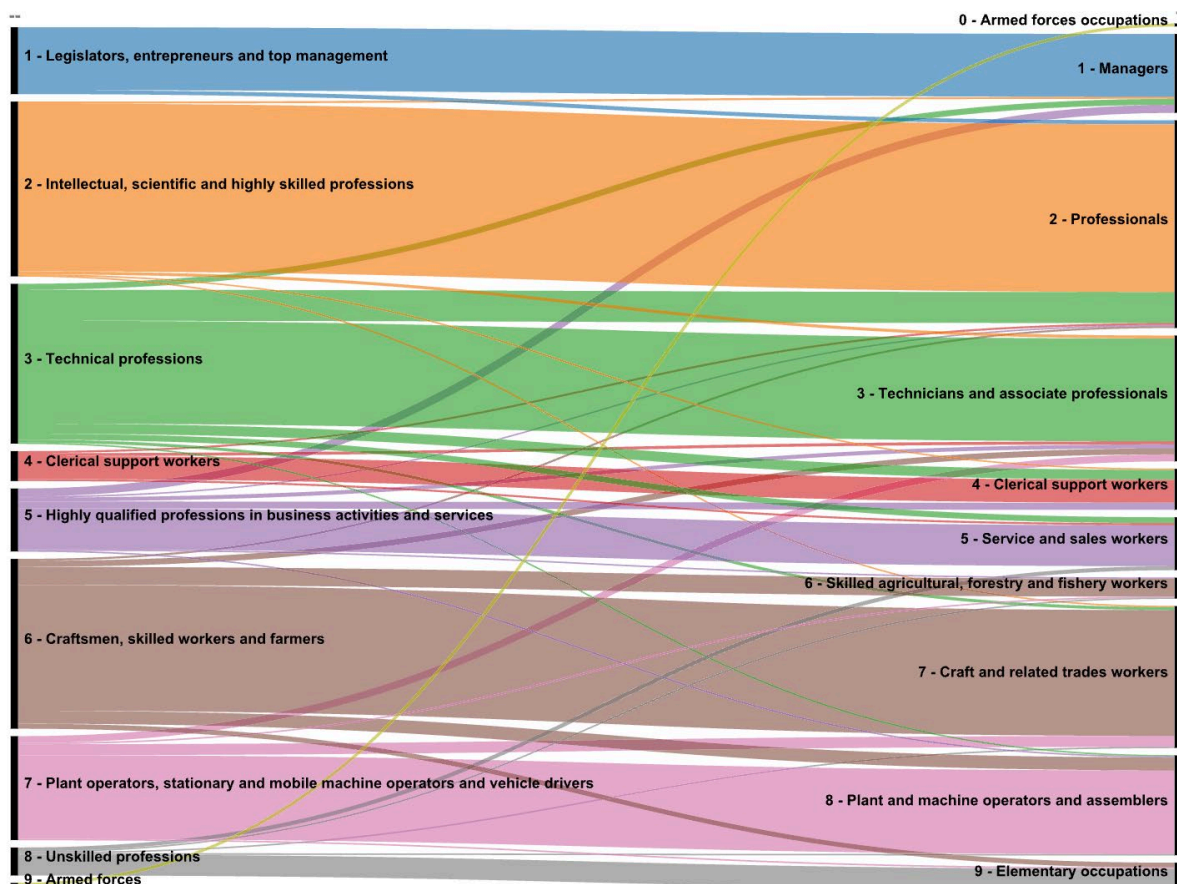


Figure 22. Alluvial diagram of the mapping from CP (left side) to ESCO (right side) for first levels (lower granularity). Source: (Giabelli et al., 2022)

Data

ESCO; National taxonomy (i.e. Italian CP-2011)

Challenges

1. Results need to be evaluated by domain experts.
2. ESCO and the national taxonomies usually have different granularities.

Data driven person-job fit

The recent explosion of web data related to the labour market, such as job advertisements, allows for the timely modelling and understanding of complex labour market phenomena using AI. This raised the interest of both academics and practitioners toward data-driven person-job fits. This is an unsupervised problem that consists in the extrapolation of patterns such as: 1) the requested skill(s) for an occupation in different countries and over time, as it emerges from the labour market and 2) the usage of terms representing job positions in the job market over time and across countries. For instance, in figure 23, the scatterplot illustrates the clustering of Information and Communication Technologies (ICT) occupations as they appear in around 2 million advertisements published in UK in 2018, according to a study by (Malandri et al., 2021). Each coloured marker represents one of the ESCO groups in the legend. The position in the map represents how those terms are used similarly, as it has been extracted by unsupervised AI algorithms. We can see that, in general, occupations belonging to the same ESCO group are close in the scatter plot. However, this is not true for all the occupations. For instance, looking at the zoom-box, the “database administrator” occupation, which belongs to the “2511: System Analysts” ESCO group, represented by a green triangle, in the plot is on the upper right zone, closer to the “2521: Database designer and administrators” group, represented by a blue square, than to its own ESCO group. This means that, in the UK labour market in the year 2018, the use of the word “database administrator” was different from the meaning in ESCO.

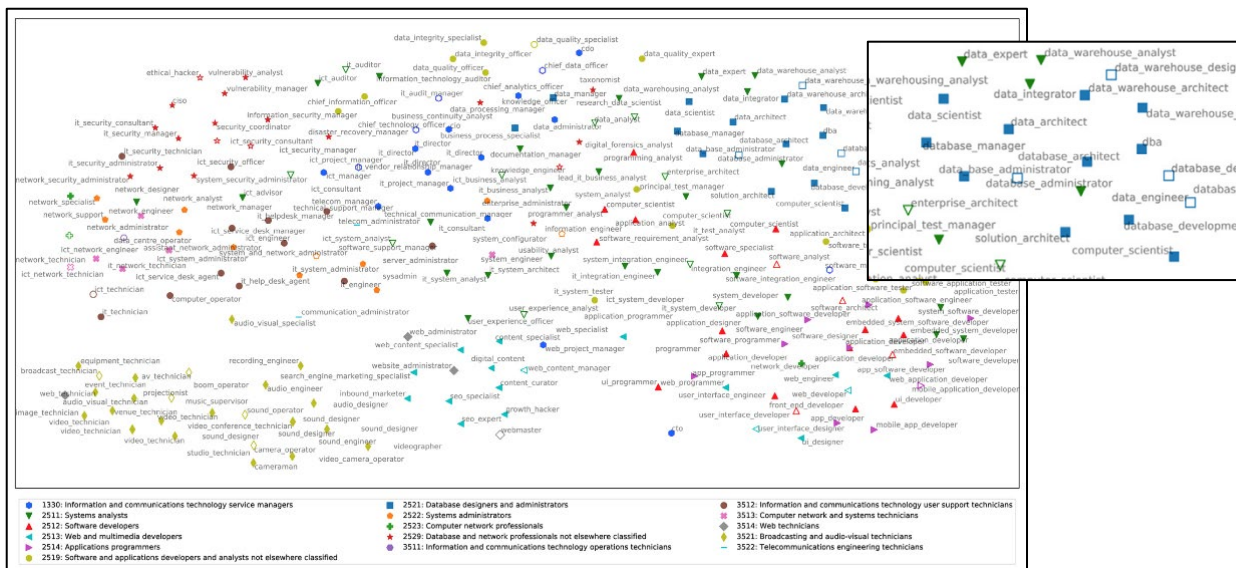


Figure 23. ICT job positions in the UK, 2018. The colours and shapes of the markers represent the ESCO categories, while the relative position of points and clusters is based on their use in the Job Advertisements. Source: created by authors.

Benefits

The labour market is constantly changing, and its lexicon varies among countries. A data-driven approach can show in which way an occupation is different across different countries, for instance in terms of requested skills. To give an example, in Table 3 some of results from Giabelli et al., (2022) are included. In this paper, the authors use unsupervised AI to assess the most suitable job for the users comparing their skills and the ones required by the occupation, as they emerge from Online Job Advertisements collected in 2018 in the United Kingdom, Germany, and France. To clarify the

matter, Table 3 shows an example of calculation of the skill gap between the starting occupation of the user and the arriving one. The starting occupation is defined as the one the user is currently holding in a specific country, called therefore starting country (in this case, *Web and multimedia developers* in the UK). The system then proposes a set of arriving occupations, i.e. the occupations in a certain country, in this case Germany, which are similar to the starting occupation in terms of skills required. It can be seen that the most similar one, with a rank of 0.56 (details on how to calculate this rank are explained in the paper) is the category of *Web Technicians* (second column). The third column lists the skills usually missing in the starting occupation to match the skills of the arriving one; the last columns are a measure of the importance of those skills for the arriving occupations, the Revealed Comparative Advantage (RCA). In the example, the system suggests that for UK *Web and multimedia developers* who want to move to Germany, the most similar occupations in the UK are *Web Technicians*. However, the former should cover a skill-gap consisting of three skills they do not possess. Among this, the first one, *perform online data analysis*, has maximum importance for *Web Technicians* in Germany (RCA=1).

Table 3. Skill Gap for UK “Web and multimedia developers” moving to ICT occupations in Germany.

Rank	Arriving Occupation	Gap Skills	RCA
0.56	Web Technicians	perform online data analysis	1
		social media management	0.7476
		social media marketing techniques	0.6638
0.2	Application programmers	provide software testing documentation	1
		Android	0.8744
		mobile operating systems	0.8227
		iOS	0.7802
0.18	Software developers	perform business analysis	0.6131
		Apache Maven	1
		business model	0.7631
		establish customer rapport	0.657
		Jboss Agile development	0.6342
			0.6225

Data

Online Job Advertisements, ICT, UK, 2018

source: <https://www.cedefop.europa.eu/en/tools/skills-online-advertisements>)

Challenges

1. Web data are usually unstructured, dirty and domain dependent. Therefore, the pre-processing and data cleaning phase is time consuming and challenging.
2. The data collection phase faces all the problems related to web data harvesting discussed in Section 2.

Fairness in loan granting

Loan granting is an important activity for banking institutions. In recent years, Machine Learning algorithms have found fertile ground in credit scoring for loan applicants. They have shown good performance in the prediction of the probability of repayment, using the applicants' demographic and credit data as input. However, a black box Machine Learning might systematically make unfair decisions in relation to some population subgroups without the possibility of understanding why. XAI methods, which allow to open the black-box and to better understand their behaviour, were found to be particularly useful in this context and, specifically, to assess two different kinds of fairness:

1. Individual fairness: given two individuals belonging to two different classes of a sensitive variable (male and female, for instance) but with all the remaining features equal (income, job status, etcetera), do they have the same probability of having a loan granted?
2. Group parity: do different subgroups of people have the same number (in percentage) of loans granted? For instance, 40% of males and 40% of females have a loan granted. In some cases, group parity is not compatible with individual fairness. For instance, if women have, on average, a lower salary than men, with all other features equal, only one of the two can hold.

XAI methods can help understand whether those two conditions hold and, if not, why. For instance, if there is no group parity in the decision, XAI will explain why (e.g. because one group has higher salary, or more stable job contracts).

Benefits

The benefit of verifying individual fairness is to check if the data or the algorithm are biased. For instance, training can be carried out using historical data on loan-granting decisions where, for several reasons, individual fairness does not hold. Group parity can help enforce fairness policies. In the above example, if a policy maker wants to balance the salary gap between the two subgroups, they can implement policies to encourage banks to apply group fairness, which makes it easier for the disadvantaged subgroup to have access to credit.

Data

Data is usually tabular data, including company data with demographic features related to the customers and third-party data on the credit history of the applicants.

Challenges

1. There is always a trade-off between the accuracy of machine learning model and its explainability. Simple algorithms using simple rules that can be explained cannot account for all the cases, especially if there is a high number of features.

The implementation of mitigation strategies may have negative economic effects in a first phase, and, in any case, it needs to be accepted by the users.

Chapter 6 – Enabling BDA to scale



Nowadays, cloud computing represents a key element to allow BDA projects to scale. This chapter introduces the general characteristics of cloud computing services to guide the reader in understanding cloud computing.

Cloud computing, as defined by NIST¹⁶ in 2011, can be considered an approach for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. servers, storage, networks, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. **The characteristics of cloud computing** can be summarised as follows:

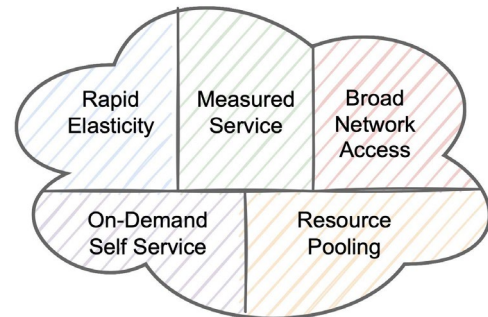


Figure 24. Characteristics of cloud computing

1. **On-demand self-service.** This feature enables consumers to unilaterally provision computing capabilities (e.g. disks, virtual machines) as needed, automatically. Furthermore, it allows consumers to view a service catalogue via a Web-based user interface and use it to request a service. Consumers can easily ‘book’ a resource according to their computation/storage needs. Furthermore, should the needs change over time, consumers can resize (increase/decrease) the computational power or the storage capacity at any time.
2. **Broad network access.** Computing capabilities are available over the network, accessible from many client platforms (e.g. desktop, laptop, mobile). This, clearly, requires the organisation to have a high-performance network (in both upload and download). Note that an organisation might decide to implement a private cloud service, in which case VPN services are used to guarantee security when accessing network resources.
3. **Resource pooling.** The provider’s computing resources are pooled to serve multiple consumers using a multitenant model. Hence, resources are assigned from the pool according to consumer demand while consumers have no control or knowledge over the exact location of the provided resources. To give an example, this approach is similar to the district heating service, where the generated heat is distributed through a system of insulated pipes for residential and commercial heating requirements, such as room heating and water heating.
4. **Rapid elasticity.** Computing capabilities can be elastically provisioned and released according to consumer needs. For example, the user can ‘resize’ a machine, adding or removing memories or CPUs from a web interface. This, in turn, gives the consumer almost unlimited scalability.
5. **Measured service.** Cloud computing provides consumers with a real-time system or interface that continuously monitors resource consumption and generates reports (including cost reports). Furthermore, most cloud computing services allow the generation of a complex set of rules to tag resources.

The characteristics of cloud computing concerning On-Premises solutions can be delivered through three different models, providing different levels of services. ‘On-Premises’ is a term used to identify

¹⁶ US National Institute of Standard Technology - <https://www.nist.gov/>

solutions installed on the organisation's/client's own servers. The figure below represents different levels of services offered by a cloud compared to an On-Premises solution, which requires the organisation to handle and manage all the components of the system's architecture, from guaranteeing the network connectivity up to the full functionality of all the applications (i.e. software) used.

When talking about cloud computing, the suffix 'as a service' is used to describe how a block of the stack shown in Figure 23 is handled by the cloud provider.

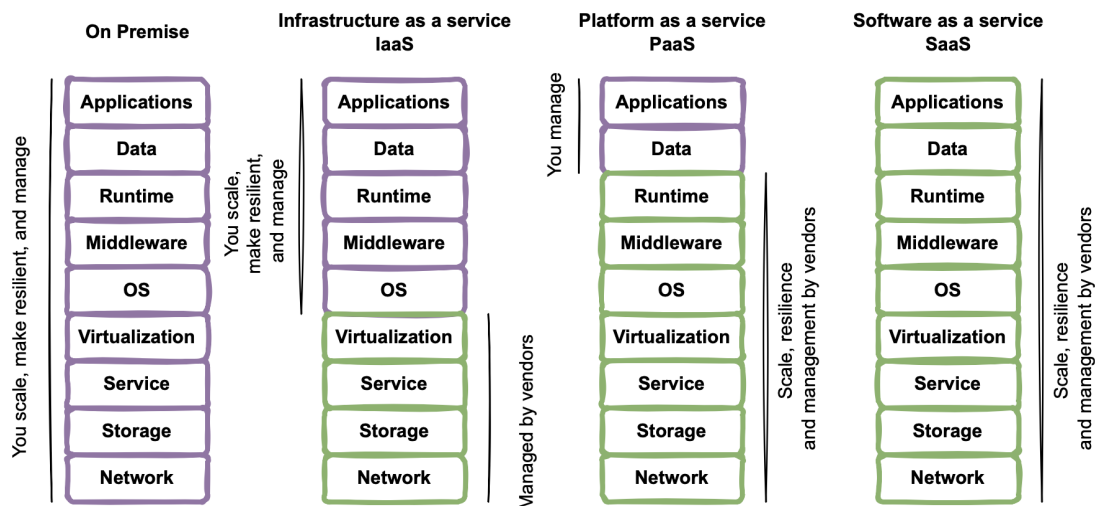


Figure 23. Service models of a cloud architecture

- 1. IaaS (Infrastructure as a Service).** This represents the basic level of abstraction, and possibly one of the most used. An example of IaaS is a *virtual machine (VM)*, which is a computing resource that uses software instead of a physical computer to run programmes and deploy apps. A virtual machine runs within another machine, providing the user with the sense that the machine used physically exists whilst it is '*virtualised*' within another (bigger) one. Each virtual machine runs its own operating system and functions separately from the other VMs, even when they are all running on the same host. Consumers deploy their software, choosing among the OS provided. On the other hand, computing resources (e.g. processing power, memory, storage, and networking components) are offered as service provider. The figure below is a snapshot of the virtual machine catalogue of one of the most used cloud providers. As one might note, the cost of a machine depends on the size despite the actual usage of the machine when it is on.

The IaaS solution is quite helpful if the consumer needs to have control over the machine and special 'licences' of software to be used/installed. Any solution deployed using IaaS can scale as much as the bigger instances of VMs scale. Additional benefits include: (i) the consumer does not have to worry about power consumption, physical servers, storage, failure, SLA, etcetera; (ii) the consumer can easily resize the machine according to novel requirements. On the other hand, the consumer is expected to manage the operating system (updates, software installed, licences, users, configurations, etcetera). Finally, (ii) backups are the customer's responsibility.

Instance types (506)

Filter instance types

Instance type	vCPUs	Architecture	Memory (GiB)	Storage (GB)	Storage type	Network performance	On-Demand Linux pricing
x1.32xlarge	128	x86_64	1952	3840	ssd	25 Gigabit	13.338 USD per Hour
x1e.32xlarge	128	x86_64	3904	3840	ssd	25 Gigabit	26.688 USD per Hour
x2idn.32xlarge	128	x86_64	2048	3800	ssd	100 Gigabit	13.338 USD per Hour
x2idn.metal	128	x86_64	2048	3800	ssd	100 Gigabit	13.338 USD per Hour
x2iedn.32xlarge	128	x86_64	4096	3800	ssd	100 Gigabit	26.676 USD per Hour
x2iedn.metal	128	x86_64	4096	3800	ssd	100 Gigabit	26.676 USD per Hour
c5.24xlarge	96	x86_64	192	-	-	25 Gigabit	4.08 USD per Hour
c5.metal	96	x86_64	192	-	-	25 Gigabit	4.08 USD per Hour
c5a.24xlarge	96	x86_64	192	-	-	20 Gigabit	3.696 USD per Hour
c5ad.24xlarge	96	x86_64	192	3800	ssd	20 Gigabit	4.128 USD per Hour

Figure 24. Snapshot from a web page of an IaaS cloud provider

Create database

Choose a database creation method [Info](#)

Standard create


You set all of the configuration options, including ones for availability, security, backups, and maintenance.


Easy create


Use recommended best-practice configurations. Some configuration options can be changed after the database is created.


Engine options


Engine type [Info](#)


Amazon Aurora


MySQL


MariaDB


PostgreSQL


Oracle


Microsoft SQL Server


Version

PostgreSQL 13.4-R1

Figure 25. Snapshot from a web page of a PaaS cloud provider

2. Platform as a Service (PaaS). Within a PaaS¹⁷ model, consumers deploy consumer-created or acquired applications onto the provider’s computing platform. This means the computing platform is offered as a service while the consumer controls deployed applications. A database running on cloud services is an example of PaaS. The DBMS¹⁸ is ‘offered’ by the

¹⁷ For example, AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, OpenShift (this list does not constitute an endorsement).

¹⁸ Database management system is the ‘software’ that allows users to work on the database, in terms of creating/updating and reading data.

cloud service. The consumer can pick a DBMS from a catalogue, choosing the size of the machine on which the database runs. Such a solution is preferable to IaaS as the consumer/organisation does not need to worry about the machine on which the DBMS is running (OS updates, security, etcetera). Furthermore, the machine can be resized (upgraded or downgraded) at any time through a simple web interface. Finally, backups are also provided as a service. A drawback is that data access is still the consumer's responsibility.

- 3. Software as a Service (SaaS).** This represents the highest level of abstraction as consumers use providers' applications running on the cloud infrastructure. SaaS, also known as cloud application services, is the most commonly used option for business. SaaS uses the internet to deliver applications, which are managed by the cloud service providers, to its users. The vast majority of SaaS applications run directly through web browsers. We daily use SaaS software, such as Gmail, Dropbox, Salesforce, and Webex. While, on the one side, a SaaS model eliminates the need to have IT staff download and install applications on each computer, on the other, the integration with existing apps and services can be a major concern.

As recently observed by a Eurostat Survey¹⁹, the majority of companies use SaaS as a service model for e-mail, office software, finance, or accounting software applications, such as ERP, CRM, or security software applications. A considerable proportion (about 74%) used also at least one cloud Infrastructure as a Service (IaaS), e.g. to host a database or storage service, while slightly over one out of five (21%) enterprises used PaaS. Nonetheless, the pervasiveness of cloud computing among different sectors is quite evident from Figure 26, which clearly shows a growing trend.

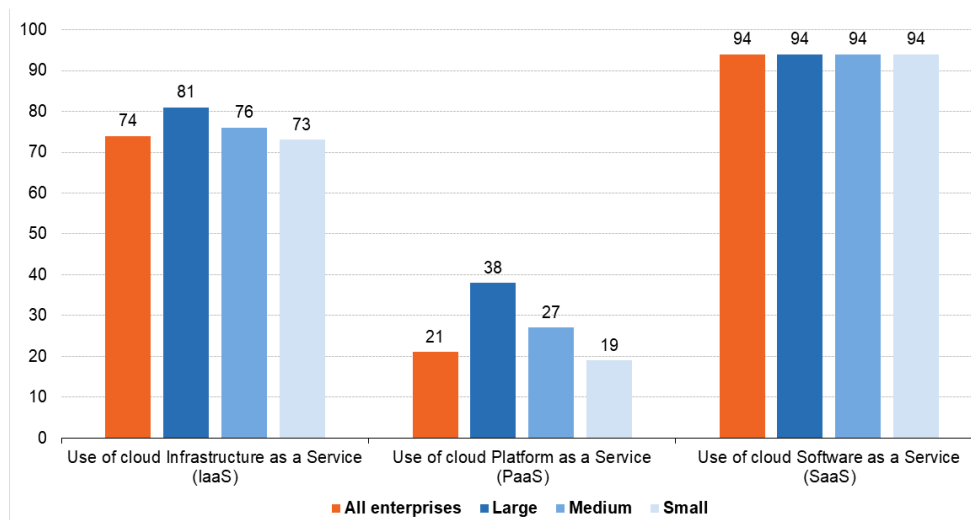


Figure 26. Types of cloud computing services used by service model (% of enterprises using the cloud), EU 2021. Source: Eurostat (online data code: isoc_cicce_use)

¹⁹ https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Digital_economy_and_society_statistics_-_households_and_individuals

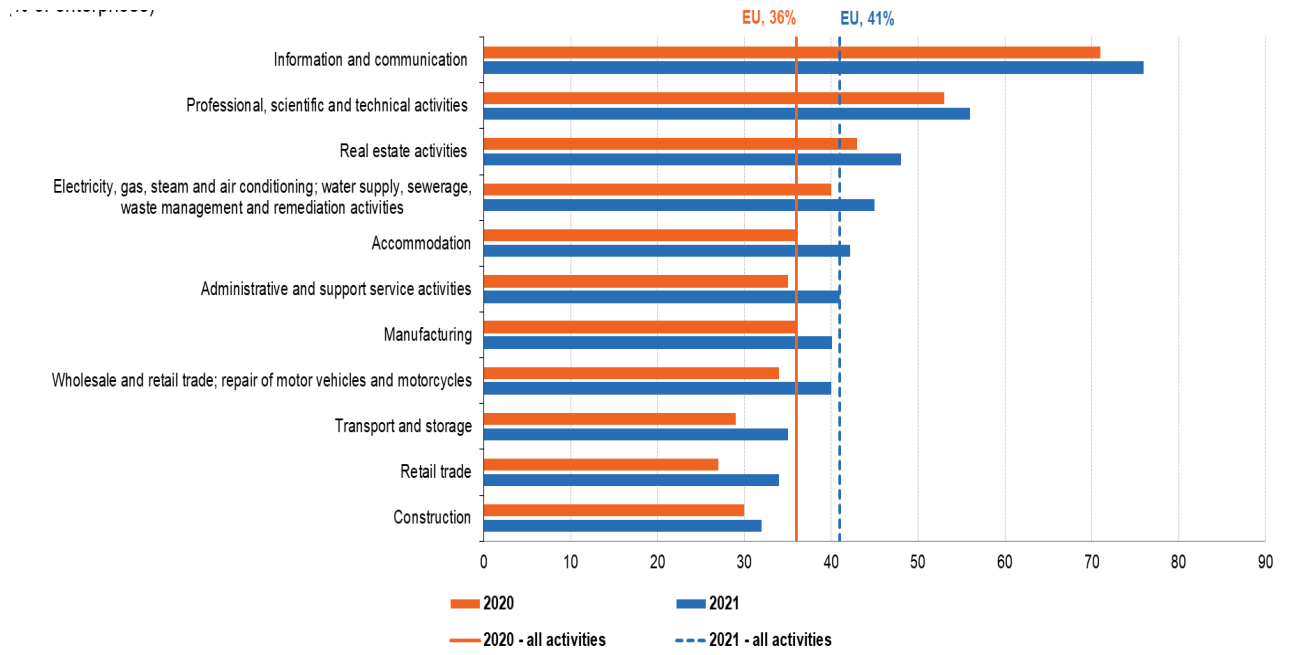


Figure 27. Use of cloud computing services by economic activity (% of enterprises using the cloud), EU 2020 and 2021. Source: Eurostat (online data code: isoc_cicce_use)

Recommendations for conducting a fully-fledged Big Data Analytics project

R1: Identify the right skills and competences to realise a fully-fledged Big Data Analytics project.

As highlighted in chapter 1, the big data processing lifecycle is a complex process that involves several actors and requires a wide range of diversified skills. Although job titles can vary, the most commonly identified actors are:

- **Data engineer:** data engineers are experts in the management of data. Their role is not limited to the design of databases and data warehouses, but includes data collection, quality control, pre-processing, and the preparation of data for data analytics tasks. The umbrella of “data engineers” may include database developers, data architects, database administrators, etcetera. Important skills and knowledge for data engineers are a good knowledge of DBMSs, query languages, database development tools, ETL tools, the ability to estimate the duration of works and queries, performing load test, etcetera.

- **Data scientist:** data scientists are experts in the modelling of data and in the solution of complex analytical problems. They must be able to cope with different data sources and formats and with large amounts of data. Therefore, they must be able to manage and transform the pre-processed data so that they can be fed to statistical modelling algorithms. Data scientists can be mathematicians, computer scientists or statisticians. The most important skills and knowledge they must possess are the knowledge of statistical models, including Machine Learning, for information extraction, and the knowledge of programming languages to implement them, together with the ability to think analytically, manage research data and interpret the results.

- **Data analyst/Statistician:** data analysts are experts in the analysis, interpretation, and presentation of data, and in the validation of models. They ensure the analysis provides consistent results and evaluate their impact on business. Data analysts are closer to business and less technical than data scientists, and their skills and knowledge include executing analytical and mathematical calculations and modelling, performing data mining, analysing big data, and using business intelligence and office tools, together with statistics, query languages and data quality assessment tools.

- **Domain expert:** domain experts are pivotal in every data analysis process, from the selection of data and variables to the validation of the outputs of the analysis. Furthermore, domain experts will understand the value, impact, and possible applications of the analysis, both in the initial, strategic phase of the process and in the evaluation and interpretation phase.

Below, the main actors for each phase of the big data processing lifecycle are presented, from the collection of raw data to the evaluation of the output.

Data collection and selection: data engineer, domain expert;

Data pre-processing and transformation: data scientist, domain expert;

Data Analytics: data analyst, data scientist, domain expert;

Interpretation/evaluation: data analyst, domain expert.

R2: Manage the trade-off between performance and explainability in Machine Learning.

As highlighted in chapter 4, Machine Learning algorithms may have different degrees of complexity. This fact should be accounted for, not only to prevent overfitting, but also to select the desired level of interpretability of the model. Complex models, such as deep neural networks or support vector machines, are mathematically complex, with millions of parameters. This usually leads to increased prediction performances but, at the same time, makes the model difficult to understand and explain. The classification trees in Figures 16a and 16b in chapter 4 are a good example. The first is a very simple model; if the TripAdvisor comment has the word 'dirty' in it, the review will be negative. However, such a model cannot have good performances, because a review could be negative for several reasons, not just the cleanliness of the room. On the contrary, the tree in figure 16b uses long rules, each one using several words. It is difficult for a human being to have a general understanding of the model and the characteristics that make a review good or bad based on it. It is important to specify that a model can be very complex without overfitting. It depends on the complexity of the data. The example of hotel review classification is a typical example of a complex problem, where the model should evaluate several word combinations in the reviews, and where there could be several reasons and different words to express a negative or positive review. Therefore, users often have to choose between complex models, which can reach high accuracies by creating large sets of complex rules, or a simple, more easily explainable model with poorer performances. An approach to manage this trade-off is to use two different models. A first model, usually powerful and complex, is trained to reach the best performances possible. A second model, called surrogate model, is used to evaluate and explain the first one. The second model is not designed to have high accuracy, but to use simpler rules, with those rules to cover a certain proportion of the ones used by the complex model. With these settings, the first model will be used for prediction purposes providing high accuracies, while the surrogate model will support the decision maker to better understand the decisions of the first model and possible errors.

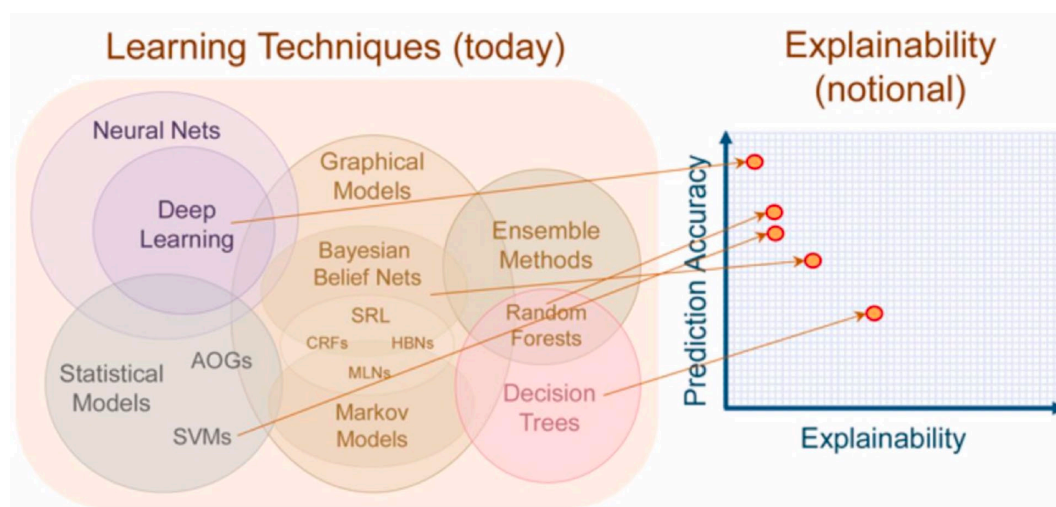


Figure 28. Prediction accuracy vs explainability of machine learning algorithms (source: DARPA)

R3: Five golden rules for implementing a BDA project on Cloud

As discussed in this report, the use of cloud computing services is mandatory to allow BDA projects to scale out. Below we report *five critical issues* that should be taken into account when executing a BDA project over the cloud.

1. **Define the service model.** As discussed in Chapter 5, there are several service models that any cloud-based service can provide. Usually, a BDA project requires to select among (i) IaaS, which provides infrastructure services that are cost-effective and highly scalable, such as storage and computing. The Cloud provider deals with security and provisioning and gives high flexibility to the user on how exactly to use the infrastructure; and (ii) PaaS, which allows users to build their own apps without having to deal with OS and other details. To design their apps, developers can employ pre-built software components, which reduces the amount of code they need to write from scratch. Usually, the cost for the more managed services is higher than IaaS; however, the time to deploy and maintain apps is lower.
2. **Big Data requires Big Data Services.** A common mistake is to set up an environment to deal with big data, writing code through libraries/tools that are not suited for that. Indeed, not all libraries and services allow for distributed computing over multiple machines. As an example, the Pandas library of Python, which is used for data management, is not able to properly scale over multiple machines. There are often appropriate alternatives - in this case, the Koalas library allows one to use the same well-known interface of Pandas while allowing for big data, multi-machine processing.
3. **Try to estimate the cloud costs.** Before deploying any architecture, the ongoing costs should be estimated. The costs depend on many different factors. For example, when estimating computing costs, it is necessary to evaluate:
 - a) The size of each individual machine in terms of CPUs and memory;
 - b) The quantity of machines to be deployed;
 - c) The operating system (Windows often costs more than Linux servers);
 - d) The pricing strategy (reserving machines for years ahead gives discounts when compared to on-demand usage).

Because of the large number of variables involved in the calculation, cloud services often provide *calculators* that can aid you in choosing the best services for your business. For example, AWS provides the handy Pricing Calculator²⁰ and Optimizator²¹ where you can estimate the overall monthly cost of the chosen services.

Monthly Cost Example for the following services:

1. **Storage - AWS S3**- 1 TB storage, 100,000 requests, 100 GB download (internet)
2. **Compute - AWS EC2** - 16 vCPUs, 64 GB RAM, 8 hours/day, 30 GB EBS storage.
3. **Big Data Distributed Compute - AWS EMR** - 5x instances of 16 vCPUs, 32 GB RAM, 8 hours/day.

²⁰ <https://calculator.aws/#/estimate>

²¹ <https://www.wellarchitectedlabs.com/cost/>

Estimate summary						
Upfront cost			Monthly cost		Total 12 months cost	
0.00 €			396.26 €		4,755.09 €	
Detailed estimate						
Name	group	region	upfront cost	monthly cost		
1. Amazon Simple Storage service(S3)	No group applied	US East (Ohio)	0.00 €	34.60 €		
Description: S3 Storage						
Config summary: S3 Standard Storage (1000 GB per month) DT inbound: Internet (1000 TB month), DT Outbound: Internet (100 GB per month)						
2. Amazon EC2	No group applied	US East (Ohio)	0.00 €	150.30 €		
Description: EC2 Compute						
Config summary: Operating system(Linux), Quantity (1), Pricing strategy (On-demand Instances), Storage amount (30GB), Instance type (m6g.4xlarge)						
3. Amazon EMR	No group applied	US East (Ohio)	0.00 €	203.10 €		
Description: EMR Big Data						
Config summary: Number of master EMR nodes(1), EC2 Instance (c5.4xlarge), Utilization (8 Hours/Day) Number of core EMR nodes (4), EC2 instance (c5.4xlarge), Utilization(8 Hours/Day)						

Figure 29. Example of cost-estimating calculator, price in Euro as at 12 September 2022. Source: Amazon

- 4. Apply Autoscaling and Load Balancing.** Scaling horizontally (i.e. over multiple machines) is one of the main advantages of deploying a Cloud Architecture (it is referred to as Rapid Elasticity in Ch.6). Many Cloud providers enable the system to automatically detect surges in user activity, and to “scale out” (deploy more machines) whenever needed - e.g. when the response time of a web application increases over a chosen time threshold. This helps keep the user experience optimal, while at the same time minimising costs, because the additional machines are booted only when required, and shut down otherwise. Additionally, the workload can be divided into multiple machines using Load Balancers, which automatically send requests to whichever machine is less used. In this way it is possible to ensure that the work is equally distributed and is less likely to suffer from the malfunctions of a single machine.
- 5. Set up cost reports and monitoring.** As previously mentioned, the cost of cloud services can grow in an unpredictable manner. A key recommendation is to set up groups - with budget quotas - and triggers that act like rules to automatically disable a specific service or user group as soon as they reach the budget threshold. Setting up these triggers and rules is a time-consuming and error-prone activity, which might require several attempts before the “right” tune that fits the users and system requirements is found. Nonetheless, this activity

can effectively make the difference in cost management because it allows to optimise resource consumption at a very fine-grained level.

R4: How to bring BDA into official statistics: understanding the value of smart statistics

Official Statistics were designed to provide a quantitative representation of society, the economy and the environment for several reasons: purposes of public interest, for policy design and evaluation, and as basis for informing the public debate (Ricciato et al., 2020). These statistical systems contained – and were structured for – data from traditional sources such as surveys (including censuses) and administrative registers. Nowadays, for official statistics to keep up with the world transformation and to continue being useful and informative, a change is needed (and is being designed).

Our world is characterised by a constant production of data: thanks to the use of the Internet of Things (IoT), our whole ‘data ecosystem’ has been changed. New (*smart*) technologies that create continuous real-time data dominate our realities. The resulting large amount of data (Big Data) that is generated has great potential for the enrichment of Official Statistics. This is the reason that led Eurostat to introduce the ‘smart statistics’ term as the future extended role of official statistics in a world permeated with smart technologies. In the future, statistics themselves will become a smart technology embedded in smart systems that will transform ‘data’ into ‘information’. On this regard, Eurostat is working to put Online Job Ads collected from the Web into official statistics as well.

However, a shift in the procedures and paradigms of Official Statistical Systems is needed to adapt to these new data. There is a need for statistical offices to rethink the way they operate: data collection and analysis must change, as must statistical production. In addition, together with the significative opportunities that this transformation could bring to official statistics, a main concern has arisen that needs to be addressed. It regards the link between smart statistics and their trustworthiness, which is the main characteristic needed for Official Statistics (and statistics in general). Notably, the term smart statistics also applied to any quantitative research that makes use of data with the goal of inferring claims about the entire population.

To build trust into smart statistics, the data life cycle needs to be auditable, transparent, with guarantees of accuracy and privacy by design. This is what the main Statistics Offices are working on, trying to provide new rules and new framework(s) that will allow these data to become valuable information, creating Trusted Smart Statistics²². Generally, *trusted* smart statistics assume that only a selected (trusted) and verified set of sources is used to produce smart statistics, as we discussed regarding the importance of selecting web sources through a multi-criteria decision-making approach (e.g. AHP previously mentioned and described). Any BDA process that makes use of web data should begin with a landscaping exercise in which (i) a set of quality criteria are enumerated by stakeholders and domain experts, then (ii) web sources are ranked by the extent to which they fit these criteria, and this ranking is monitored over time to guarantee the credibility of the final results.

²² https://ec.europa.eu/eurostat/cros/content/trusted-smart-statistics-nutshell_en;
<https://www.istat.it/it/archivio/257157>

R5: Address issues related to the representativeness of Web Data

Interest in the problem of inferring the ability to represent a population from web data has been growing in recent years, given the rising usage of web data in many domains. Here, the question related to what extent Web data (and, in turn, the analytics derived from them) can represent the population becomes crucial. In this respect, it is important to distinguish between the concept of representativeness and significance. Regarding dataset or analytics provided, representativeness refers to the ability of matching the (defined) characteristics of the population observed or inferred. An example should clarify the matter. When considering a system collecting job ads taken from websites, the OJA representativeness of the real labour market demand would be crucial if analysing the historical time series of the demand and trends. Indeed, one potential drawback of online ads is that they capture only advertisements posted on the Internet and may not be representative of the universe of advertisements, and this possible bias varies across sectors and occupations. Generally speaking, the issue of representativeness cannot be fully addressed. However, we can use an approximation by means of post stratification techniques. If reliable claims about the population are required, data from other sources are needed- not just to weight against, but to rigorously test whether claims are generalisable. Eurostat (2021) discuss the properties of OJAs in the context of the European labour market and propose some techniques to address the issue of representativeness. Regarding the time pattern of OJAs, Lovaglio et al. (2020) show that online advertisements display the same time series behaviour of advertisements obtained from official statistics, both overall and at sectoral level. Moreover, it is worth mentioning that if the analysis focuses on the skill distribution within occupation, any bias that online advertisements may have, is likely to be greatly weakened.

In essence, the issues related to the representativeness of web data should be addressed by (i) identifying either surveys or official statistics that might represent a proxy of the population, (ii) building models to weight classes and groups that are under/over-estimated in the web data according to the official statistics, and (iii) compare the trends of web data against official statistics to verify if (and to what extent) the trends are significant, in spite of the bias that unrepresentative data might contain.

References

- Birjali, M., Kasri, M., & Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226, 107134. <https://doi.org/10.1016/j.knosys.2021.107134>
- Cowie, R., Cox, C., Martin, J.-C., Batliner, A., Heylen, D., & Karpouzis, K. (2011). Issues in data labelling. In *Emotion-oriented systems* (pp. 213–241). Springer.
- Eurostat (2021), *Experimental Statistics: Labour Market Concentration Index using OJA data*, Eurostat methodological note, Luxembourg. [https://ec.europa.eu/eurostat/documents/7894008/13101260/Methodological_note.pdf/4718223c-761d-0b40-a240-99c6801ef813?t=16278925084491ef813\(europa.eu\)](https://ec.europa.eu/eurostat/documents/7894008/13101260/Methodological_note.pdf/4718223c-761d-0b40-a240-99c6801ef813?t=16278925084491ef813(europa.eu))
- Giabelli, A., Malandri, L., Mercorio, F., & Mezzanzanica, M. (2022). WETA: Automatic taxonomy alignment via word embeddings. *Computers in Industry*, 138, 103626. <https://doi.org/10.1016/j.compind.2022.103626>
- Laplante, P., Milošević, D., Serebryakov, S., & Bennett, D. (2020). Artificial Intelligence and Critical Systems: From Hype to Reality. *Computer*, 53(11), 45–52. <https://doi.org/10.1109/MC.2020.3006177>
- Le Meur, O., & Baccino, T. (2013). Methods for comparing scanpaths and saliency maps: Strengths and weaknesses. *Behavior Research Methods*, 45(1), 251–266. <https://doi.org/10.3758/s13428-012-0226-9>
- Lovaglio, P. G., Mezzanzanica, M., & Colombo, E. (2020). Comparing time series characteristics of official and web job vacancy data. *Quality & Quantity*, 54(1), 85–98. <https://doi.org/10.1007/s11135-019-00940-3>
- Malandri, L., Mercorio, F., Mezzanzanica, M., & Nobani, N. (2021). MEET-LM: A method for embeddings evaluation for taxonomic data in the labour market. *Computers in Industry*, 124, 103341. <https://doi.org/10.1016/j.compind.2020.103341>
- Marz, N., & Warren, J. (2015). *Big data: Principles and best practices of scalable real-time data systems*. Manning.
- Mezzanzanica, M., Boselli, R., Cesarini, M., & Mercorio, F. (2015). A model-based evaluation of data quality activities in KDD. *Information Processing & Management*, 51(2), 144–166. <https://doi.org/10.1016/j.ipm.2014.07.007>
- Monarch, R. M. (2021). *Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI*. Simon and Schuster.
- Power, D. J. (2002). *Decision support systems: Concepts and resources for managers*. Greenwood Publishing Group.
- Pustejovsky, J., & Stubbs, A. (2012). *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. O'Reilly Media, Inc.
- Redman, T. (2001). *Data Quality: The Field Guide*. Digital Press.
- Ricciato, F., Wirthmann, A., & Hahn, M. (2020). Trusted Smart Statistics: How new data will change official statistics. *Data & Policy*, 2, e7. <https://doi.org/10.1017/dap.2020.7>
- Sadalage, P. J., & Fowler, M. (2013). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison Wesley.
- Sagiroglu, S., & Sinanc, D. (2013). Big data: A review. 2013 *International Conference on Collaboration Technologies and Systems (CTS)*, 42–47. <https://doi.org/10.1109/CTS.2013.6567202>
- Thomas, J. J., & Cook, K. A. (2006). A visual analytics agenda. *IEEE Computer Graphics and Applications*, 26(1), 10–13. <https://doi.org/10.1109/MCG.2006.5>
- Wang, R. Y., & Strong, D. M. (1996). Beyond Accuracy: What Data Quality Means to Data Consumers. *Journal of Management Information Systems*, 12(4), 5–33. <https://doi.org/10.1080/07421222.1996.11518099>

- Yang, Q., G., M., & Helfert, M. (2019). Analysis of data warehouse architectures: Modeling and classification. In *Proceedings of the 21st International Conference on Enterprise Information Systems (ICEIS 2019)*, pages 604-611, Heraklion, Crete, Greece. <https://doi.org/10.5220/0007728006040611>
- Zhao, B. (2017). Web Scraping. In L. A. Schintler & C. L. McNeely (Eds.), *Encyclopedia of Big Data* (pp. 1–3). Springer International Publishing. https://doi.org/10.1007/978-3-319-32001-4_483-1

WPEF21051

The European Foundation for the Improvement of Living and Working Conditions (Eurofound) is a tripartite European Union Agency established in 1975. Its role is to provide knowledge in the area of social, employment and work-related policies according to Regulation (EU) 2019/127.