

Faculdade de Engenharia da Universidade do Porto

Adaptive Information Visualization Framework

Luís Spínola



Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Daniel Silva

October 31, 2021

Adaptive Information Visualization Framework

Luís Spínola

Mestrado Integrado em Engenharia Informática e Computação

October 31, 2021

Abstract

The human brain can quickly become overwhelmed with the amounts of data computers can process. Consequently, data abstraction is necessary for a user to grasp information and identify valuable patterns. Usually, data is abstracted in a pictorial or graphical format. In conjunction with interactiveness, it is possible to adapt to how data is abstracted to obtain more detailed or specific conclusions in real-time.

ASAE is the Portuguese Authority of Food and Economic Security organisation. Its services still have relatively limited resources to visualise the gathered data as the current systems do not integrate and abstract all information effectively. While a new platform is in development to solve many of the organisation problems, including the one stated, there is the need for a solution that eases the process of creating visualisations. Existing solutions for this purpose do not account for the user individual preferences and necessities, and so, consequently, the lack of a personalised experience may hinder the user's comprehension of graphical information and, therefore, their decision-making process.

The main objective of this thesis is to critically examine how visualisation and geovisualization techniques can help analyse information and aid the decision-making process in the context of ASAE's data and then conceptualise a solution that facilitates the introduction of data abstraction techniques on the new platform. To help achieve this goal, a literature review was conducted in which some essential guidelines and techniques in the scope of this work were described and analysed.

This dissertation proposes a user-centred framework that eases the process of creating visualisations for the developers of a platform while still offering the end-user a personalisable experience. Nowadays, the standard user demands more personalisation on the systems he uses. As complex platforms offer an extensive set of visualisations, there is also the necessity to explore ways to apply previous personalisations to new visualisations encountered. With that hypothesis as a basis, the suggested framework offers an adaptive nature. The adaptiveness of the proposed solution offers a different approach based on the user personalisation and visualisation selection. The conceptualised solution was later prototyped.

The resultant prototype was included in the platform and tested to ensure the information about the different spatial data is transmitted to the user in a quick and effective process. It showed that users are pleased with the usability of the prototype and proves that they desire control on the configuration of their visualisations. Thus, the framework proposed is expected to improve the way patterns are recognised and decisions are made based on the available data of an organisation.

Keywords: Information Visualisation, Geovisualisation, Visual Analytics, Visualisation Techniques

Resumo

O cérebro humano pode facilmente ficar sobrecarregado com a quantidade de dados que os sistemas informáticos conseguem processar. Consequentemente, uma abstração de dados é necessária para que um utilizador compreenda todas as informações e identifique padrões relevantes. Normalmente, os dados são resumidos num formato pictórico ou gráfico. Em conjunto com interatividade, é possível adaptar a forma como dados são abstraídos para obter conclusões mais detalhadas ou específicas em tempo real.

A ASAE é a organização portuguesa de Autoridade de Segurança Alimentar e Económica. Os serviços da ASAE ainda têm recursos relativamente limitados para visualizar os dados coletados, pois o sistema atual não integra e abstrai todas as informações de forma eficaz. Enquanto uma nova plataforma está em desenvolvimento para resolver muitos dos desafios da organização, incluindo o indicado, há uma necessidade de uma solução para facilitar o processo de criação de visualizações. Soluções existentes para este propósito não tem em conta as preferências e necessidades individuais de um utilizador, consequentemente a falta de uma experiência personalizada pode dificultar a compreensão da informação gráfica e piorar o processo de tomada de decisões.

O objetivo principal é examinar criticamente como as técnicas de visualização e geovisualização podem ajudar a analisar informações e auxiliar no processo de tomada de decisão no contexto dos dados da ASAE e depois conceptualizar uma solução que facilite a introdução de técnicas de abstração de dados numa plataforma. Para ajudar a atingir o objetivo, foi realizada uma revisão de literatura na qual algumas diretrizes e técnicas essenciais no contexto deste trabalho foram descritas e analisadas.

Esta dissertação propõe uma framework centrada no utilizador que facilite o processo de criação de visualizações para os programadores de uma plataforma enquanto oferece ao utilizador final uma experiência personalizável. Atualmente, o utilizador exige mais personalizações nos sistemas que usa. Como plataformas complexas oferecem um conjunto extenso de visualizações, há também a necessidade de explorar métodos para aplicar personalizações feitas anteriormente nas novas visualizações encontradas. Com essa hipótese como base, a framework sugerida oferece uma natureza adaptativa. A adaptação da solução proposta oferece uma abordagem diferente baseada na personalização do utilizar e seleção de visualizações. A solução conceptualizada foi mais tarde prototipada.

O prototipo resultante foi incluído na plataforma e testado para certificar que a informação sobre os diferentes dados espaciais são transmitidos ao utilizador de forma rápida e eficiente. O prototipo mostrou que os utilizadores estão satisfeitos com a usabilidade do prototipo e prova o desejo de controlar as configurações das

suas visualizações. Por isso, é expectável que a framework sugerida melhore a forma como padrões são identificados e as decisões são feitas com base nos dados de uma organização.

Contents

1	Introduction	1
1.1	Context and Motivation	1
1.2	Objectives	2
1.3	Methodology	3
1.4	Document Structure	3
2	Background Knowledge and Related Work	5
2.1	Information Visualisation	5
2.1.1	Context	5
2.1.2	Techniques	7
2.1.3	Techniques Taxonomy	16
2.1.4	Dashboards and Key Performance Indicators	19
2.1.5	Design	22
2.1.6	Colour	25
2.1.7	Evaluation	28
2.1.8	Conclusion	31
2.2	Geovisualisation	31
2.2.1	Context	31
2.2.2	Techniques	32
2.2.3	Tools and Technologies	34
2.2.4	Practical Uses	36
2.2.5	Conclusion	37
2.3	Interaction	37
2.4	Visualisation Recommendation and Adaptive Visualisation	42
2.4.1	Visualisation Recommendation	42
2.4.2	Adaptive Visualisation	43
2.4.3	Conclusion	48
3	CIGESCOP Overview and Solution Architecture	49
3.1	CIGESCOP project	49
3.1.1	Current Technologies	50
3.1.2	Potential Technologies	52
3.1.3	Visualisation Module and Problems to Solve	54
3.2	AIVF: Architecture	55
3.2.1	Context	55
3.2.2	Architecture	56
3.2.3	Functionalities	57
3.3	Conclusion	59

4 AIVF: Implementation	61
4.1 Technologies Used	61
4.1.1 Development of the Prototype	62
4.1.2 Information Visualization Techniques	62
4.2 Chart Types Used	63
4.2.1 Information visualisation graphics	63
4.2.2 Geovisualization graphics	64
4.3 Families of Graphics	65
4.4 Personalisation of Graphics	66
4.4.1 Personalisation Options	67
4.4.2 Interface Layout	72
4.5 Input	72
4.6 Adaptiveness	74
4.6.1 Database Tables	75
4.6.2 Adaptive Component	79
4.7 Conclusion	80
5 Usability Study and Results	89
5.1 Usability Testing	89
5.2 The Study	90
5.2.1 Metrics Saved	90
5.2.2 Introduction	90
5.2.3 Gathering Information About the Participant	91
5.2.4 Usability Questions	91
5.2.5 Adaptability Questions	93
5.3 Results	95
5.3.1 Participant Information	95
5.3.2 Answers About the Usability	96
5.3.3 Answers About the Adaptability	96
5.4 Conclusion	97
6 Conclusions and Future Work	99
6.1 Work Done	99
6.2 Background Knowledge Acquired	99
6.3 Usability Study and Results	100
6.4 Future Work	100
Bibliography	101
A Questionnaire	107
B Test Results	109

List of Figures

2.1	Minard’s full figurative map of 1869	7
2.2	Example of bar and line charts	9
2.3	Example of scatter plot and histogram	10
2.4	Density estimates of the sepal lengths of three different iris species . .	11
2.5	The problem with pie charts	11
2.6	Changing the order of the categories in a radar chart	12
2.7	Parallel Coordinate Plot	13
2.8	Comparison of four Treemap layout algorithms	14
2.9	Four different ways of presenting a table	15
2.10	Information Visualisation Data State Reference Model	17
2.11	Quality metric pipeline	19
2.12	Use of color in data visualization	26
2.13	Simulation of CVD	28
2.14	Bar chart to demonstrate the problem with the traffic light colours . . .	29
2.15	Rise and fall of specific terms to describe the digital cartography concept	32
2.16	Example of some geovisualisations	34
2.17	Example of a flow map	34
2.18	Visualisation of traffic in Rome between 12:00 and 13:00 UTC	35
2.19	A space–time cube of Napoleon’s march in Russia	41
2.20	Scatter plot of types of graphics identified by emerging clusters	45
2.21	Visualisations showing the same data (analysis setting/monitoring setting)	46
2.22	The components of the adaptive and adaptable system and the system flow	46
2.23	Adaptive taxonomy calculation steps	47
3.1	High level representation of the project architecture	50
3.2	A more holistic view of Django’s architecture.	51
3.3	AIVF: Architecture Diagram.	57
4.1	AIVF: Charts. Bar chart	64
4.2	AIVF: Charts. Side by side bar chart	65
4.3	AIVF: Charts. Scatter Plot, Sankey Diagram, Bullet graph and Gauge chart	66
4.4	AIVF: Personalisation. <i>y tick</i> and <i>x interval</i>	68
4.5	AIVF: Personalisation. Simplification of values and showing values di- rectly on the visualization	69
4.6	AIVF: Personalisation. Colour picker.	70
4.7	AIVF: Personalisation. Interpolation	82

4.8	AIVF: Personalisation. Percentage and pie plot radius	83
4.9	AIVF: Charts. Heat Map and Hexagonal Map	83
4.10	AIVF: Personalisation. How the personalisation options are shown. Example 1.	84
4.11	AIVF: Personalisation. How the personalisation options are shown. Example 2.	84
4.12	AIVF: Personalisation. How the personalisation options are shown. Tool-tips.	85
4.13	AIVF: Charts. Column Map	86
4.14	AIVF: Charts. Bubble Map	87
4.15	AIVF: Database Tables.	87
5.1	Dashboard test page. Part 1.	93
5.2	Dashboard test page. Part 2.	93
5.3	Dashboard test page. Part 3.	94
5.4	Dashboard test page. Part 4.	95
A.1	Questionnaire. Page 1.	107
A.2	Questionnaire. Page 2.	108
A.3	Questionnaire. Page 3.	108
B.1	Participant information results	109
B.2	Usability results	110
B.3	Adaptation results	111

Listings

4.1	Template of the structure of the input JSON	73
4.2	AIVF: Database Tables. Example of <i>graph_options</i> content.	76
4.3	AIVF: Database Tables. Example of user history of personalisation on the <i>one_numerical</i> family.	77

List of Tables

5.1 Participant engagement.	95
5.2 Participant information results.	96
5.3 Usability results.	96
5.4 Adaptation results.	97

Abbreviations

AIVF	Adaptive Information Visualization Framework
ASAE	Authority of Food and Economic Security (<i>Autoridade de Segurança Alimentar e Económica</i>)
CSS	Cascading Style Sheets
CVD	Color Vision Deficiency
FEUP	Faculty of Engineering of the University of Porto (<i>Faculdade de Engenharia da Universidade do Porto</i>)
geoVis	Geovisualisation
GIGESCOP	Intelligent Center of Operational Management and Control (<i>Centro Inteligente de Gestão e Controlo Operacional</i>)
GIS	Geographic Information System
HCI	Human-Computer Interaction
IA.SAE	Artificial Intelligence in Food and Economic Security (<i>Inteligência Artificial na Segurança Alimentar e Económica</i>)
infoVis	Information Visualisation
JSON	JavaScript Object Notation
KML	Keyhole Markup Language
KPI	Key Performance Indicator
LIACC	Artificial Intelligence and Computer Science Laboratory (<i>Laboratório de Inteligência Artificial e Ciência de Computadores</i>)
MVC	Model, View, Controller
MVT	Model, View, Template
PCP	Parallel Coordinate Plot
UI	User Interface
WebGL	Web Graphics Library

Chapter 1

Introduction

There is an extreme need for companies and public administration to digitise their services and processes. This need has been translated into software mostly in the last two decades. As is the case of ASAE¹ (a Portuguese specialised authority responsible for critical tasks like food safety and economic surveillance), the problem is that many organisations have several systems to cater to different aspects. Consequently, there is now the possibility of applying new technologies that improve previously implemented procedures and allow the development of new products that encompass all the information about an organisation in one unique system.

1.1 Context and Motivation

IA.SAE, which means Artificial Intelligence in Food and Economic Security, is a project created in late 2018 that resulted from a partnership with FEUP (Faculty of Engineering of the University of Porto) through LIACC (Artificial Intelligence and Computer Science Laboratory) and ASAE. It aimed at promoting food safety, public health, consumer protection and safeguarding market rules and free competition between economic operators through the development of risk analysis models and the selection of economic agents to be inspected, based on the most recent artificial intelligence (AI) and computational learning techniques using the databases available and to be developed at ASAE.

A platform came as a sequence of the IA.SAE project called CIGESCOP (Intelligent Center of Operational Management and Control). One of the requirements to satisfy the CIGESCOP objectives consists of a system interface with interactive geo-referenced visualisations, real-time visualisations, key performance indicators and information based on maps and routes. This document has that requirement as a basis and presents a process to satisfy it. A more detailed view of the project is presented in section 3.1.

Nowadays, analysing the data collected is a must for organisations worldwide as important patterns and problems can be identified that would not be noticed otherwise. However, large amounts of available information make it extremely difficult for a user to manually explore and analyse data. So, mechanisms to guide the user and

¹More information about the organisation: <https://www.asae.gov.pt>

abstract all data obtained are a necessity. The visualisation process is about converting all kinds of data into graphical representations, and there are numerous methods to transform a set of data into a picture/representation.

When developing a platform to expose graphical visualisations, there needs to be a particular concern with the overall design, usability and interactivity, besides the effectiveness of the methods used. The problem starts when the complexity of the data raises to a point where it is not easy to apply strategies to expose valuable and critical patterns. Considering the diversity of needs, preferences, and requirements of different organisations and users, the set of visualisations should provide intuitive solutions to reduce efforts to identify and absorb critical information.

1.2 Objectives

This work's goal is to develop a solution to produce visualisations based on the data provided by ASAE, and include it in the system resulting from the IA.SAE project called CIGESCOP. The visualisations should adapt taking into consideration user necessities through the use of personalisation. The resultant visualisations should fit in a single uniform theme, adapt to a user and allow him to absorb information effectively and personalise the environment according to his preferences and necessities.

To comply with the expressed necessities, a data visualisation tool is the most adequate solution. As the CIGESCOP platform is an enormous project and visualisations are a constant throughout the system, developing visualisations one by one from the ground up is not a reasonable solution. Consequently, there is the need for some kind of tool or framework that produces graphical representations that respect the state of the art in their regard and satisfy all the other requirements of the system.

A data visualisation tool produces representations based on the underlying data. The traditional tools for that purpose do not comply with all the necessities stated and usually offer a one-size-fits-all solution that does not take into consideration user needs and necessities, resulting in an unsatisfactory data exploration process. It is then the main objective of this thesis to conceptualise a solution that satisfies the following requirements:

- Must be developer-friendly and easily deployable, as the system requires an extensive set of visualisations, and it is unreasonable to allocate that task to a single developer.
- Should offer the capabilities to graphically display performance indicators, the most helpful information visualisation representations and equally valuable geographic visualisation representations.
- Must consider the user as an individual, offering an extensive set of personalisation options that satisfies its preferences.
- Should adapt user visualisations taking into consideration its preferences as the size of the system makes it unreasonable to ask a user to personalise each graphic available/inserted manually.

1.3 Methodology

Although visualisations highly depend on the type of data in question, it is always beneficial to learn about the successful use of different techniques and later apply those that seem appropriate. Hence, a literature review was conducted to find what has been done in related areas and the advantages and disadvantages that those analysed projects present to achieve the primary goal stated in this document.

With the research done, it was concluded that developing an adaptive visualisation framework based on user personalisation that could later be used to produce intelligent visualisations and geographic representations was an adequate approach to solve the problems posed by the objectives previously stated.

The ASAE organisation provided the data used in the course of this work. It contains the information collected over the years. The data contains private and sensible information. The collaboration with the stakeholders was continuous, that was extremely helpful as stakeholders are part of the target user, and it is essential to have the ability to test the system with direct users.

Later, the proposed solution was prototyped. The developed prototype was then deployed in the CIGESCOP platform to conduct usability tests to evaluate and validate the usefulness of the framework and its capabilities to solve the problems caused by the objectives of this thesis.

In conclusion, the objectives of this dissertation could only be completed by taking a deep dive on research topics related to the wanted solution goals, to collect valuable information to develop a framework that satisfied this project needs but that also could be used in other projects by offering a different perspective in comparison to similar solutions.

1.4 Document Structure

This document offers a total of five additional chapters. This section briefly presents the remaining content of this document.

Chapter 2, named **Background Knowledge and Related Work**, contains an analysis of the state of the art of relevant topics in this thesis's scope. The most important research fields explored were information visualisation, geovisualization and interaction. It finishes with an overview of the topic of visualisation recommendation and the results of the research on adaptive visualisation systems.

Chapter 3 (**CIGESCOP Overview and Solution Architecture**), starts by offering a more detailed view of the platform resultant from the CIGESCOP project. On the first section the current state of the platform is described as well as potential technologies that could be used to achieve the solution proposed. After, there is an enumeration of the necessities on the CIGESCOP platform in relation to visualisation. Then the architecture of the conceptualised framework is presented that solves the problems previously enumerated.

Next comes chapter 4, with the title of **AIVF: Implementation**. This key chapter explains the process of implementing the prototype for the solution found and conceptualized to satisfy the necessities of the objectives formerly presented.

Chapter 5 (**Usability Study and Results**) is where it is explained how tests were conducted to evaluate the prototype of the solution proposed and the results are also presented.

The document ends with chapter 6, named **Conclusions and Future Work**. In that self-explanatory chapter, the overall conclusions are presented as well as a brief description of possible future work.

Chapter 2

Background Knowledge and Related Work

This chapter contains four sections. The first is **Information Visualisation** and is divided into several sub-sections, starting with a brief context of the topic, an enumeration and description of relevant techniques and methods in the area and a look towards classifications and selection methods for those technologies. The following topic presents dashboards and explains in some detail Key Performance Indicators (KPIs). The next sub-section introduces design concepts and describes essential guidelines when creating an interface to present the techniques discussed before. The section ends with a sub-section about colour complemented by presenting the results of research on colour vision deficiency. The second section is **Geovisualization** and is also divided into several sub-sections. The first one for a context and the second one to describe some techniques. The following part describes tools and technologies that were used in a scientific context, and the section ends with the description of some actual practical uses on the topic. Section 3 (**Interaction**) offers some useful insights for both fields of infoVis and geoVis. The fourth and last section explores the concepts of **Visualisation Recommendation** and **Adaptive Visualisation** and the development done on those research areas. This last section explores this two concepts together as they complement each other, it presents some of the models and frameworks proposed over the years on those topics.

2.1 Information Visualisation

Information visualisation is a field in which the major objective is to abstract and help understand complex data by producing user-friendly visualisations with the raw data obtained. Although an almost ancient topic, its most recent form is relatively recent as it came with the digital era. However, visual illustrations have been used for centuries to reveal patterns and tell stories. This specific technology-related discipline which is nowadays called information visualisation originated in the 1980s alongside the rise of the first machine graphics applications [[Bailey and Pregill, 2014](#)].

2.1.1 Context

Investigation in visual design, human-computer interaction, graphics, psychology, and many others are all of extreme importance for information visualisation. The need

to consider information visualisation arises from the fact that the data available for people to achieve a conclusion is usually originated from an inadequate method for that purpose. The basic idea is to present data in some visual way, allowing a human being to view and interact directly with the data. The goal is to generate critical thinking about the matter in question by offering an intuitive overview and showing relevant connections in the abstracted data.

A classic in the field of information visualisation considered one of the best drawn historical graphics and referenced by several authors is the famous Minard's map of Napoleon's 1812 campaign into Russia, the "Carte figurative des pertes successives en hommes de l'Armée Française dans la campagne de Russie 1812-1813" (see Fig. 2.1). In [Spence, 1980] the historical map is shown as a valuable representation that poses new questions; in [Kraak, 2003] several geovisualization techniques are used to try to expose the information of the painting in different ways; in [Bailey and Pregill, 2014] it is given as an example of visual storytelling; in [Goebel, 2014] it is provided as an example on a chapter that talks about the author informal observations about good visualisations. The painting offers a very compelling diagram that presents very few interpretation problems with a simple visualisation that exposes complex information. The brown line represents the march towards Moscow and the black one the retreat. That same line width is proportional to the number of soldiers left in the army, and relevant information is subtly displayed to complement it, such as the temperature. It is a remarkable example of a visualisation of data because it quickly exposes considerable valuable information and raises important questions like the ones exposed by Robert Spence in [Spence, 1980]:

"Did thin ice on the Berezina River cause the huge loss of soldiers, or did they just (understandably!) desert?'. 'What caused a significant number to head North just after marching a few miles towards Moscow?'"

There is a general agreement that in Fig. 2.1, Jacques Minard effectively captures the devastating passage of time during military campaigns of that time and its persistent disease and logistics problems with just an image. Although highly praised as one of the best visual representations, one can still argue that the need to abstract data makes it necessary to resume or omit some of the information. Therefore, there are other ways to expose a specific set of information that can be just as valuable.

"Alternative pictures of the same data will expose hypothetical relationships in the data that were simply not previously considered; for example, the Napoleon diagram includes a chart near the bottom that shows the change in temperature during the campaign, but it is not so easy to create hypotheses about the weather's impact on the size of the army as it traveled." [Goebel, 2014]

Some information has to be resumed or even omitted in scenarios with several variables. Thus, it becomes difficult to evaluate a visualisation since many different approaches to the same data set can result in valuable information towards shared or different goals. In conclusion, no specific visualisation theory can guide decisions about consolidating all large data sets and converting them into successful visual representations [Spence, 1980].

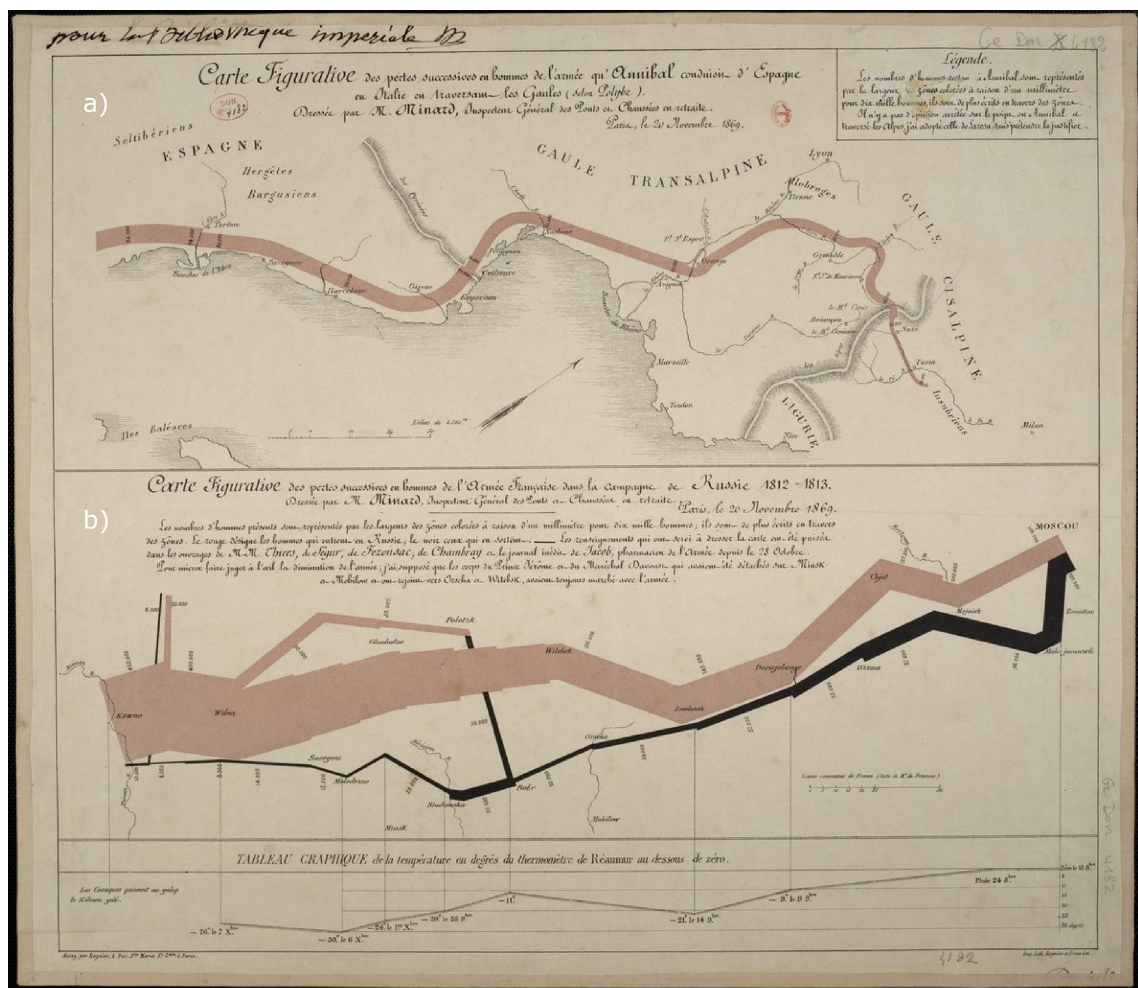


Figure 2.1: Minard's full figurative map of 1869. a) Hannibal's march across the Alps. b) March of Napoleon's army towards Moscow, and its retreat.

An optimal way to present a visualisation does not exist as concluded before. However, research work on the area is extensive, and so in the following sections of this chapter, a more in-depth overview is taken towards the state of the art of suggested guidelines and techniques.

2.1.2 Techniques

There is an extensive list of techniques available to represent information as well as the patterns it may contain. This type of techniques are essential in aiding the user in the creation of a mental model, transmitting information clearly and transparently. Data visualisation is not meant to be looked at for its aesthetic arguments and features but rather to convey useful information [Wilke, 2019]. This part of the chapter aims to present some techniques that could be effectively used in this work context.

When opting for any of these techniques, two concerns are raised. The first concern is the type of data, which can be numerical (quantitative), categorical (qualitative), or ordinal. Numerical data can be measured, aggregated [Wexler et al., 2017]

and mathematical operations can be executed with it. Quantitative data can be further divided into two sub-types: discrete and continuous data. On the other hand, categorical data represents qualities or characteristics. The numbers involved cannot be used in mathematical operations (an example from [Rumsey, 2010], is when numbers are used but do not have real numerical meaning like using 1 for male and 2 for female). The ordinal type of data falls in between of numerical and categorical data [Rumsey, 2010], this type of data does appear in the categorical form. However, it can be treated like numerical data (for example, a rating from 0 to 5 to classify some system can then be handled mathematically to make a median).

The second concern is the user characteristics; as stated in [Forsythe et al., 2016], for anyone who has never witnessed an information visualisation system before, understanding it might not be an easy task. In the opposite side, more capable users can handle more complex visualisations. So techniques should take into consideration the target reader.

For most techniques, the classical 2D Cartesian coordinate system is in use, where the x and y-axis run orthogonally to each other, and more often than not there is the need to specify the range and unit of each one, as stated in [Wilke, 2019]. In the same book, the use of grids in the graphics background to guide the reader is discussed. The grid line should be adapted concerning the data presented. Ideally, the grid should have an aspect ratio that allows the reader to detect differences in the data's positioning. If the x and y-axis belong to the same unit the spacing of the grid should be equal as the same distance along both axes will correspond to the same number of units of data; on the other hand, if the two axes correspond to different units the grid and the axis itself can be stretched or compressed still resulting in an accurate visualisation. Some common techniques are listed and described below:

- **Bar chart.** In this type of chart, the bars can be vertical or horizontal. The length of each bar represents a measure, and as stated in [Wexler et al., 2017], sorting the bars can become very helpful, as a common task when using this type of chart is to detect and compare the most prominent and most miniature measures. One variant is the **stacked bar chart**, where a bar is divided into sections, with a section being a sub-bar. Consequently, similar sections share the same visual variable between bars. Another variant is a **grouped bar chart**. As stated in [Wilke, 2019], we are frequently interested in analysing more than one categorical variable simultaneously. In this type of graphic, the bars along the axis are determined by a categorical variable. Then inside each bar, a group of sub-bars is created concerning the second categorical variable. Example of a bar, stacked bar, and grouped bar chart available at Figs. 2.2a, 2.2b, 2.2c, respectively.
- **Bullet Graph.** Bullet graphs work in a similar manner to bar charts but have extra visual elements to offer more context ¹. This type of graphic facilitates the evaluation of performance. There is an indicator that represents the actual value to be evaluated (Feature Measure) in the middle of the graph. There is also a line that marks the target value (Comparative Measure) and the colours in the background are of use combined with labels to present range ratings.

¹Source: https://datavizcatalogue.com/methods/bullet_graph.html

Obviously, the number of range ratings should be kept at a feasible minimum to prevent over-clustering the visualisation.

- **Line chart.** It is often called a **line plot** or **line graph**. It can be described as a simple but powerful graphical representation that connects data points with a continuous line and is mostly used to represent changes over time. Usually, a specific measurement is shown on the y-axis, and the interval scale is available on the x-axis (see Fig. 2.2d). This type of representation is not limited to only one line, as long as the chart is readable, double lines or more are useful to help the reader compare specific trends over interludes of time.

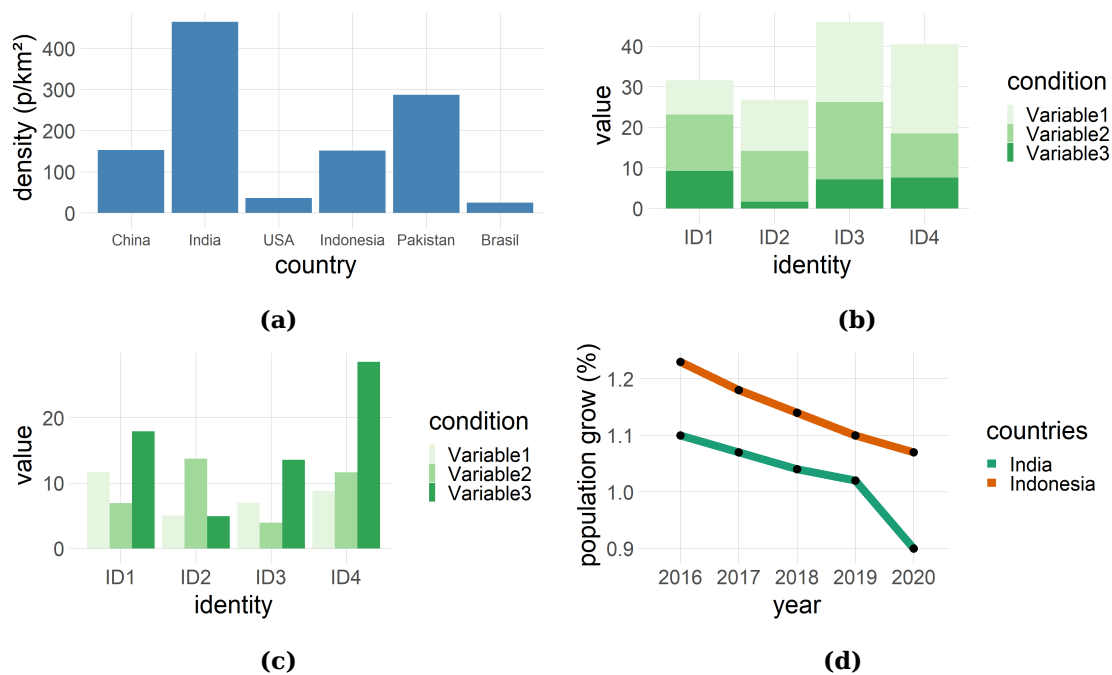


Figure 2.2: a) Example of a bar chart (2020). b) Example of a stacked bar chart. c) Example of a grouped bar chart. d) Example of a line chart.

- **Scatter plot.** Also denominated as **scatter chart**, **scatter graph**, among others. It is a chart in which points usually represent one single entity and are placed in a graphic where the x and y-axis are attributes relative to the context (see Fig. 2.3a). It is useful to discover relationships between two variables and so its purpose is to allow the reader to compare two different measures [Wexler et al., 2017]. A variation of the scatter graph is called **bubble chart** in which an additional dimension of data is displayed as the area of the points.
- **Histogram.** This type of technique results in a graphic where each column width represents an interval and the height shows how much data is contained in that specific interval (see Fig. 2.3b). Although it serves a different purpose compared to bar charts, the display of both methods only differs on the x-axis where the columns of a histogram do not have space between them. According to [Nuzzo, 2019] this type of visualisation is useful to detect shape features in the distribution and to compare sub-groups in the data. The author calls the

columns as “bins” and points out that the bin number/bin width is a tuning parameter that should be experimented with to achieve the correct balance as a histogram with too wide bins faces the risk of omitting important distributions and with too many bins (too narrow bins) it can result in an overpopulated representation. This statement is supported by [Wilke, 2019] that claims that as most visualisation programs generate by default a histogram with a specific bin width, it is of most importance to confirm if the resulting width is appropriate to display the data effectively.

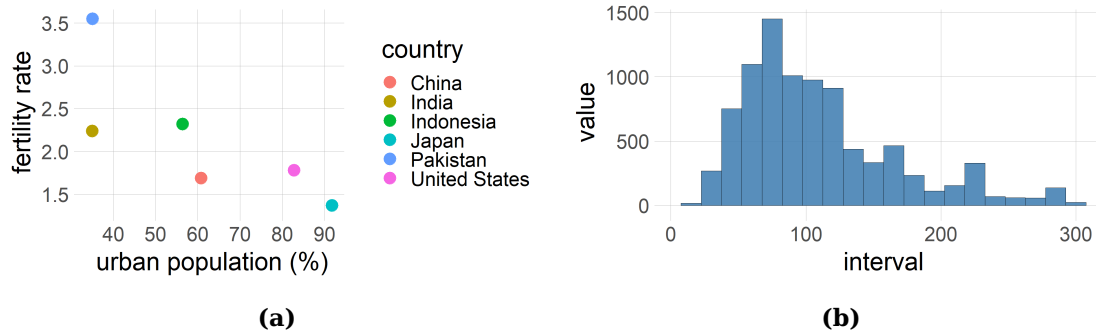


Figure 2.3: a) Example of a scatter plot. b) Example of a histogram.

- **Density plot.** Also known as **kernel plot**. Much like histograms, this graphic’s appearance varies in function of a value, in this case, the bandwidth, and it faces similar representation problems when the right choice isn’t made [Wilke, 2019]. It uses a statistical technique called kernel smoothing to reproduce the values. Contrary to histograms that allow only the direct comparison of maximum two distribution (by making two separate histograms and then rotating one by 90 degrees and the other by minus 90 degrees and merging them [Wilke, 2019], the same method used to make a population pyramid) with density plots is possible to compare more than one distribution through the use of colour, labelling and transparency (see Fig. 2.4), as our optical mechanism is more suitable to detect minimum distances rather than vertical ones, although, it is generally worse than histograms when the objective is to detect gaps in a distribution.
- **Pie chart.** The authors of [Wexler et al., 2017] recommend against the use of pie charts. They claim that this type of graphic representation does not work well with our visual system. To prove their point, they present an image (see Fig. 2.5) where they show how hard it can be for a human to make accurate estimates of angle sizes. The authors suggest that it is hard to guess that the second “pie” has the same 25 percent “slice” as the first because it is not aligned to an axis. The third one has a “slice” of 13 percent as an example of how hard it is to estimate a value accurately when it involves angle sizes. Their claim is applicable if accurate estimates are the goal, but even if accurate estimates are needed in some cases, it is possible to label all “slices” of the pie chart clearly. However, other types of representations are preferred. This point is supported by the claim presented in [Wilke, 2019], that if the data is encoded as a distance as is the case of the bar chart, our optical system perceives it more precisely than when data is encoded through a combination of two or more distances that together generate an area as is the case of a pie chart “slice”.

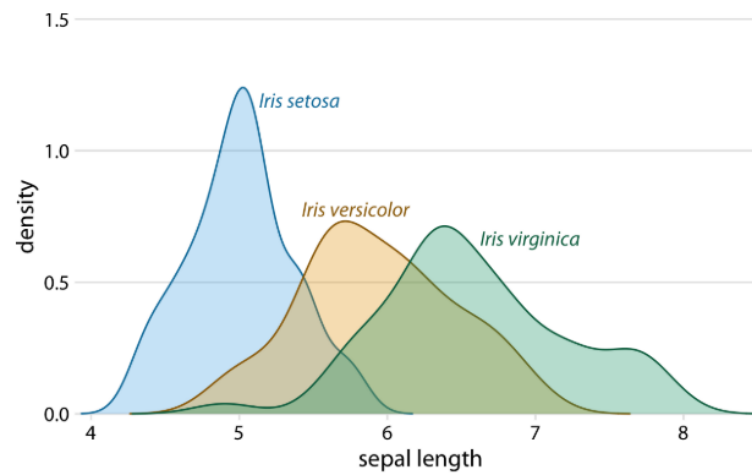


Figure 2.4: Density estimates of the sepal lengths of three different iris species. Each density estimate is directly labeled with the respective species name [Wilke, 2019].



Figure 2.5: The problem with pie charts. What percentage of each pie does the blue segment represent? [Wexler et al., 2017].

- Radar chart.** It is also known as star, spider or web chart. The attributes scale from the centre when increased, usually six to eight attributes are used but more or less are applicable. A line connects the attribute values, thus creating a polygon. It is a method to display multivariate data, but it is still limited. It is possible to go for more than the classic eight attribute chart. However, if this number is increased too much, the resultant graphic can become saturated with information, this is called overplotting. This type of representation, just like the pie chart, is often recommended against. One reason is that the reader might focus on the shape created by the chart, and it happens that the shape changes drastically with the ordering of the values (see Fig. 2.6). One other reason is that radar charts are harder to read than the more traditional bar or line chart, those methods can display the same information most of the times. Still, there are potential ways where radar chart use can be justified [Few, 2005]: the information to be presented contains multiple measures that demand different quantitative scales; the goal of the chart is to expose the symmetry of the variables rather than to compare their magnitudes; perhaps the data fits the round frame because it might be intuitively circular by nature.

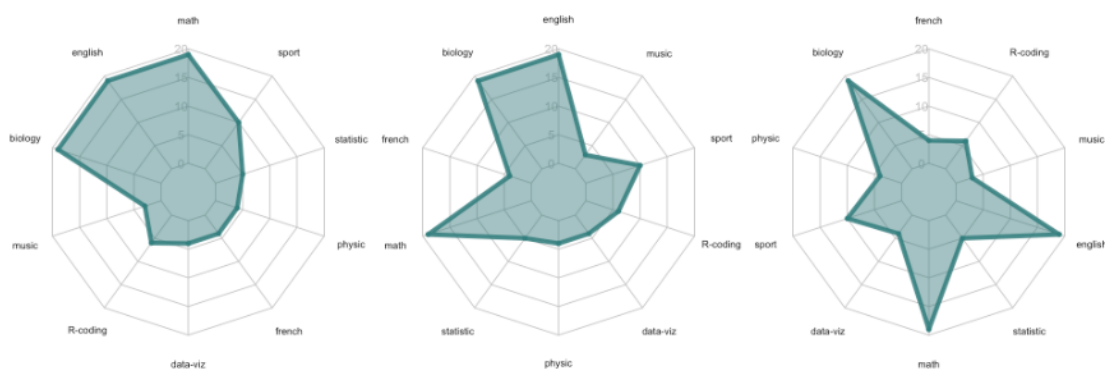


Figure 2.6: Changing the order of the values in a radar chart (<https://www.data-to-viz.com/caveat/spider.html>).

- Parallel Coordinate Plots.** This method presents a way of visualising and analysing high-dimensional data, each variable has its own axis, which are placed parallel to each other. Each axis scale can differ according to the unit of measurement of the respective attribute. The values are plotted as a series of lines connected across all the axes (see Fig 2.7). All techniques have their weakness, and parallel coordinate plots are no different. There are limitations in the number of values displayed: if the number is too high, the plot can become over-cluttered and even illegible. This is due to the fact that the non-interactive form of a PCP (Parallel Coordinate Plot) may have prevented the widespread acceptance as a useful statistical representation following the year after its introduction [Edsall, 2003]. Hence interactivity can be used to highlight a collection of lines distinguishing them from the others thus removing most of the noise. This type of representation along side the radar chart is useful when there is a reasonable high amount of attributes but a low amount of values.

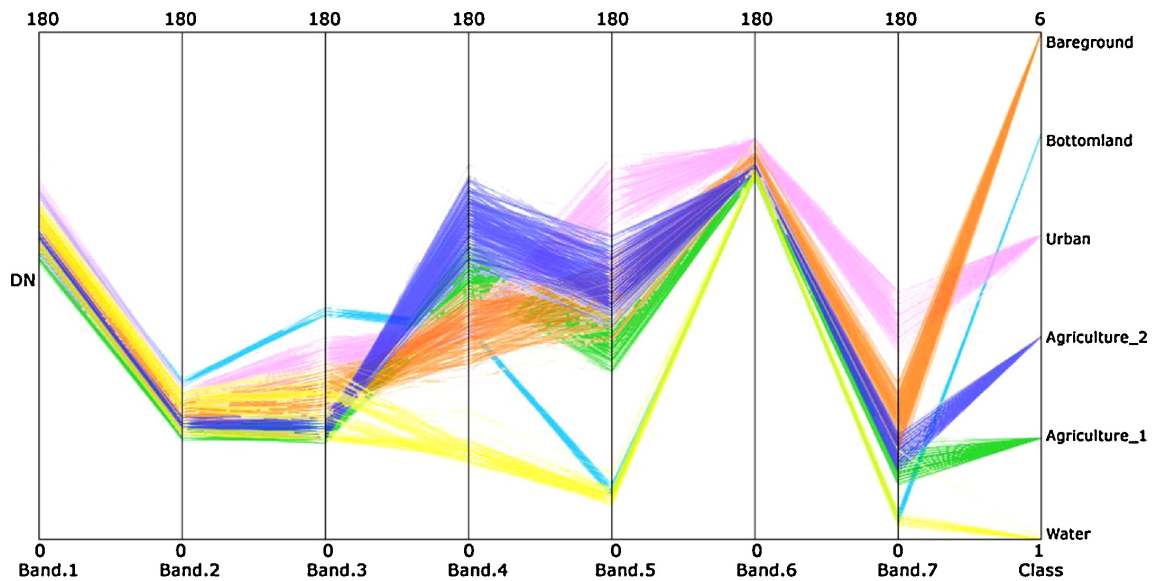


Figure 2.7: Parallel Coordinate Plot of sample data [Ge et al., 2009].

- Mosaic Plots.** A graphical method that facilitates the identification of relationships between two or more categorical values. According to Friendly in [Friendly, 2002], a mosaic plot is a graphical method to display the values in a contingency table cross-classified by one or more factors. To make a mosaic plot an enclosing rectangle is divided into smaller rectangles whose areas represent the proportions [Wilke, 2019]. With the use of interaction to support the traditional mosaic plot, it is possible to cope with categorical data of almost any type [Theus, 2012].
- Treemap.** It is a method for displaying hierarchical data using nested images, usually rectangles. Each branch of a tree is given a rectangle, which is then divided into smaller rectangles representing lower branches. Each rectangle's size has an area related to a specified dimension of the data, and colour can be of use to help clarify the different rectangles and their meaning. It is a method that offers a fair and efficient use of screen space. Although it should be taken into consideration that the human brain can't do valuable comparisons between an attribute like the area, it is not an easy task to compare areas of rectangles with different lengths and heights just with the human eye. Consequently, treemaps are not well suited for tasks involving precise comparisons. Although designed to display hierarchical data, it is of use when there are many categories to visualise. The traditional treemapping algorithms are limited to rectangular shapes. In [Balzer and Deussen, 2005], an approach called Voronoi Treemaps is presented that aims at eliminating problems with the aspect ratio and identifying the hierarchical structure that rectangles can bring. A visual example from that article can be seen in Fig. 2.8.
- Dendrogram.** Its a diagram representing a tree, it shows the hierarchical relationship between information. To interpret a dendrogram, the focus needs to be on the height at which two objects are joined together. It is a summary, and

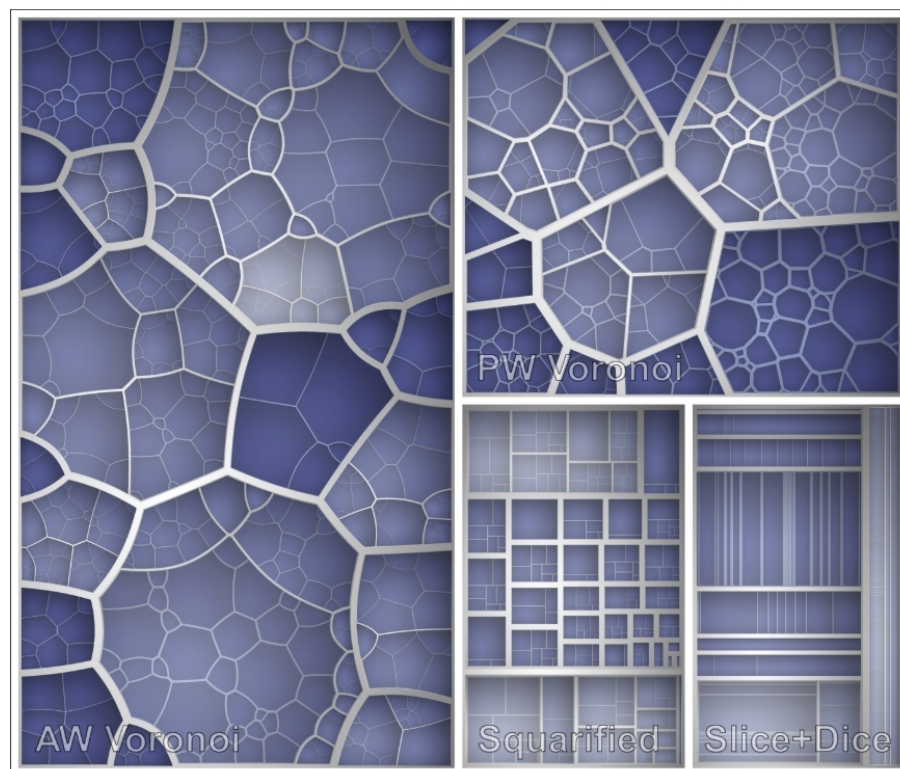


Figure 2.8: Comparison of four Treemap layout algorithms, at first, the top hierarchy level was subdivided with the Squarified Treemap algorithm, then for each of the four subareas according to its label a different layout algorithm was used (a brighter color indicates a lower hierarchy level) [Balzer and Deussen, 2005].

like most abstraction of data, some information is lost in the process. The dendrogram can either be used with a hierarchic dataset and show the connections between the nodes explicitly or represent the results of a clustering algorithm².

- **Tables and highlight tables.** When there is the need to show raw numbers, a simple table might be the best solution. Yet, because of their simplicity they don't always receive the attention they deserve [Wilke, 2019]. When applicable, a highlight table to facilitate the reading of exact values can also be used [Wexler et al., 2017]. In his book [Wilke, 2019], Wilke suggests key rules when constructing a table (see Fig. 2.9). The rules are: to avoid vertical lines; to not use horizontal lines between data rows (except as dividers between the title row and the rest and identical cases); when placing text in columns, it needs to be left-aligned except if the text contains just one character; if so, it should be centred, the number columns should be right-aligned and use the same number of decimal digits; the header fields need to be aligned with the data they represent.

a ugly

Rank	Title	Amount
1	Star Wars	\$71,565,498
2	Jumanji	\$36,169,328
3	Pitch Perfect 3	\$19,928,525
4	Greatest Showman	\$8,805,843
5	Ferdinand	\$7,316,746

b ugly

Rank	Title	Amount
1	Star Wars	\$71,565,498
2	Jumanji	\$36,169,328
3	Pitch Perfect 3	\$19,928,525
4	Greatest Showman	\$8,805,843
5	Ferdinand	\$7,316,746

c

Rank	Title	Amount
1	Star Wars	\$71,565,498
2	Jumanji	\$36,169,328
3	Pitch Perfect 3	\$19,928,525
4	Greatest Showman	\$8,805,843
5	Ferdinand	\$7,316,746

d

Rank	Title	Amount
1	Star Wars	\$71,565,498
2	Jumanji	\$36,169,328
3	Pitch Perfect 3	\$19,928,525
4	Greatest Showman	\$8,805,843
5	Ferdinand	\$7,316,746

Figure 2.9: Four different ways of presenting a table. Table "a" and table "b" don't follow the rules previously listed [Wilke, 2019].

In [Elzer et al., 2011] it is introduced the argument that information graphics are a form of language. Just like it is possible to convey an unintended message by using inappropriate words, intonation and the likes, it is possible to misrepresent data in a representation. This fact can be used to improve the visualisation, as stated in [Wilke, 2019] the visualisation of data is half art and half science, and the problem is to achieve an aesthetically pleasing visualisation without getting in the way of the science. All representations can make use of communicative signals when convenient. A communicative signal is, for example in a bar chart to use colour, shade or texture to distinguish a bar from the others [Elzer et al., 2011]. On the other end of the spectrum, it is possible to degrade the state of a visualisation. Some techniques that depend on an axis can have their information distorted if the axis does not start at

²Source: <https://www.data-to-viz.com/graph/dendrogram.html>

zero, for example, a bar chart, and it is mandatory that graphics are not misleading. Other types of graphical visualisations like the line chart do not require the axis to start from zero as starting from another value is usually more relevant and does not distort the comparison of data.

This chapter tries to present standard techniques that in some way or another could be of use in the development of projects that share similar objectives to the ones stated in the document. However, this listing only scratches the surface of available techniques. Different variations of currently available representations are common as well as the proposal of new methods, as a consequence, it is unfeasible to list all available techniques and their variations, it is although mandatory to keep an eye on research on new ways of displaying data but even more important to pay attention to the ongoing research on the more traditional visualisations. In the design sub-section a more in depth view of principles to aid the implementation of this kind of techniques is taken.

2.1.3 Techniques Taxonomy

With so many techniques available there is the need for rules that help generalize the picking of a certain representation. This aggregate of rules and classifications is usually called taxonomy, and several taxonomies for information visualisation have been explored over the years. Visualisation taxonomies have two objectives, which are to guide the users and guide research [Tory and Möller, 2004]. It categorizes visualisations and so it aids in discovering representation ideas for a specific need, plus it helps researchers place their research with more focus towards a specific goal even facilitating the process of finding similar studies.

One of this taxonomies is based on a model called Information Visualisation Data State Reference Model (see Fig. 2.10) also known as Data State Model [Chi, 2000]. This model offers many components, which can be divided in Data Stages, Data Transformations and Within Stage operators.

- **Data Stages.**
 - **Value.** The data into its raw state.
 - **Analytical Abstraction.** This next state is when meta-data is created.
 - **Visualisation Abstraction.** The information that is visible on a technique.
 - **View.** The result visualisation from the prospective of the end user.
- **Data Transformations.**
 - **Data Transformation.** Receives as input the raw data (Value) and transforms it into some sort of analytical abstraction.
 - **Visualisation Transformation.** This step receives the return of the last transformation and turns it into visualisable content by further abstracting the data.
 - **Visual Mapping Transformation.** As the last step this transformation receives the visualisable content and turns it into a graphical representation.

- **Within Stage operators.** These operators do not directly change the underlying information structures. Each data stage has a related within stage operator (Within Value, Within Analytical Abstraction, Within Visualization Abstraction, and Within View).

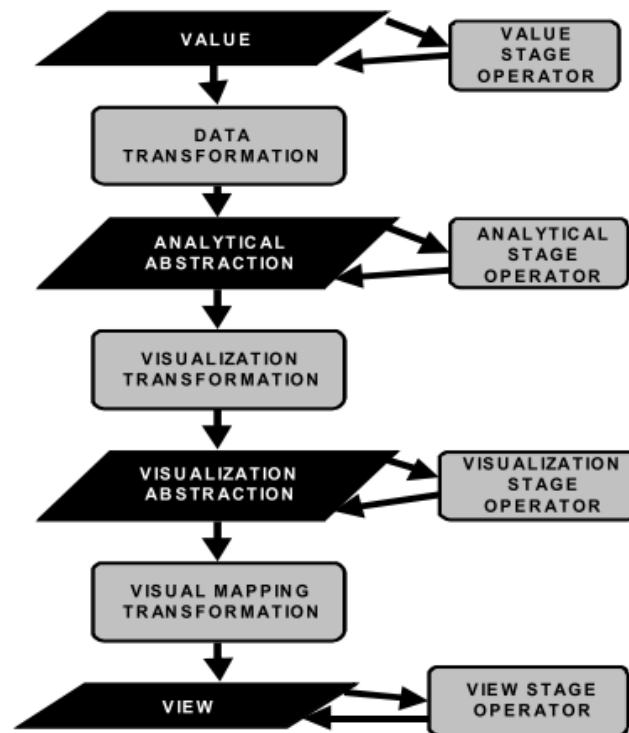


Figure 2.10: Information Visualisation Data State Reference Model [Chi, 2000].

The Data State Model can be described as a set of classification processes for visualisations according to how the operators are used [Bertini et al., 2011]. One of the strengths of this model is that it shows the modular steps towards the implementation of visualisations, and so in a development context it eases the process of implementing several visualisations by reusability of the modules [Chi, 2000].

In [Shneiderman, 2003], the author presents a type by task taxonomy that considers seven different data types. These types of data represent the problems readers aim to overcome, and are:

- **One dimensional.** Examples given by the author are textual documents and alphabetical lists of names. The user perhaps needs to find a specific name in the alphabetical list and so the method used should cater to that.
- **Two dimensional.** Refers mostly to planar and map data. In this context users want to find a path between elements or a specific location for example. For this the author recommends geovisualisation techniques (some are enumerated later in this document).
- **Three dimensional.** This data type represents "real-world objects such as molecules, the human body, and buildings have items with volume and some

potentially complex relationship with other items." [Shneiderman, 2003]. Some problems users have to overcome are on the scope of orientation.

- **Multi dimensional.** Refers to a large proportion of the data available. Elements have a certain amount of attributes that need to be represented with for example points in a n-dimensional space. One example given by the author to represent this type of data is a parallel coordinate plot.
- **Temporal.** As the name suggests this type of data is used to represent changes or comparisons over time. The reader in this context might want to have a more detailed look at some specific point in time, and related actions. Some examples of representations are a histogram or a density plot.
- **Tree.** Refers to data with an hierarchical nature. It usually has a root and all the other elements have a connection to one parent. Example of representations approached before are the mosaic plot and the tree-map.
- **Network.** There are many kinds of networks. It differs from trees as the elements have multiple connections.

To complete the proposed taxonomy, Shneiderman looks at several tasks that directly influence the use of a representation:

- **Overview.** This task argues that a user should have an overview of all the information.
- **Zoom.** When a reader is presented with an overview of the representation in question he can then use zoom for example to get more detailed insights.
- **Filter.** Refers to the importance of filtering options either to remove unwanted information or explore the data.
- **Details-on-demand.** The need to offer details by selecting or just hovering over some specific point.
- **Relate.** States that a representation should benefit the patterns found finding useful relationships in data.
- **History.** Refers to the technical functionality to allow the user to undo and replay actions.
- **Extract.** This task refers to how useful it is for an user to have the option to extract the information that he got throughout filtering either saving it to a file or sharing it, as so on.

Most research prototypes have usually managed only one type of data type but for relevant commercial solutions to come out they need to handle more than one of those [Shneiderman, 2003]. The tasks (overview, zoom, filter, relate, history, extract) are all mandatory for a decent representation and so when the moment to choose a technology comes it needs to comply with all of those tasks.

In [Bertini et al., 2011] a pipeline to help on transforming raw high-dimensional data into interactive representations is suggested (see Fig. 2.11) based on a previously established pipeline presented here (Data State Model). This pipeline allows

for the exploration of several different ways of representing high-dimensional data, trying to find the most adequate one, thus creating several alternatives while at the same time evaluating the different options. It follows a cycle that in the end of its interactions most likely produces a valid and effective representation. This pipeline has 3 core processes:

- **Data Transformation.** This process exist to make sure the source data is right-fully formatted in a similar manner to how it is done in the Data State Model.
- **Visual Mapping.** In this step "data dimensions are mapped to visual features to form visual structures." [Bertini et al., 2011].
- **View Transformation.** This process consists of rendering the visual structures obtained.

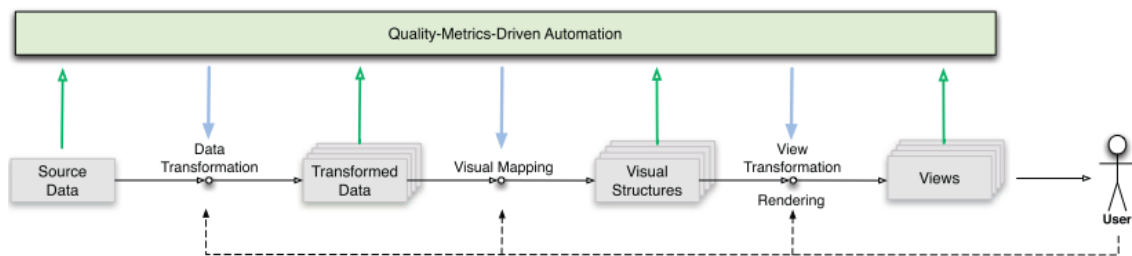


Figure 2.11: Quality metric pipeline [Bertini et al., 2011].

This pipeline encourages the user to be active and in control on all processes while the quality-metrics-driven automation represents how quality measures fit into the process. The automated metrics get information from the current stage and use it to influence the proceeding processes.

2.1.4 Dashboards and Key Performance Indicators

This part presents broad concepts that are important to get insights on. It is important to look at information on dashboards as they are the most common method to aggregate several data related visualizations. It is also extremely relevant to grasp the concept of KPIs (key performance indicators) and their importance.

2.1.4.1 Dashboards

The techniques just enumerated, on their own, might not be enough to help the user abstracting data effectively. Most organisations need to monitor their business but have several different services to track the information gathered. For the user whose job is to discover useful patterns, several services result in resource and time-consuming tasks. Consequently, there is a necessity for tools that combine the most relevant techniques to a specific data set. Data dashboards try to solve this problem.

Dashboards deal with complexity by limiting the whole of visible metrics to a single interface. In [Pauwels et al., 2009], the authors define a dashboard as a compilation of fundamental performance metrics that are interconnected and reflect on short and long-term interests relevant to an organisation. As dashboards agglomerate all data

from available and relevant sources for a specific organisation, they can be considered a data-driven decision support system [Yigitbasioglu and Velcu, 2012]. In the same article, the authors state that a dashboard represents an iceberg's tip when concerning the data because it is what the user sees first. When necessary, he can analyse further to uncover the causes of uncharacteristic behaviours. That drilling down of data can be successfully achieved through the possibility of interaction. So in [Yigitbasioglu and Velcu, 2012], the authors suggest a definition of a dashboard that emphasises their interactivity: "A visual and interactive performance management tool that displays on a single screen the most important information to achieve one or several individual and/or organisational objectives, allowing the user to identify, explore, and communicate problem areas that need corrective action."

Generally, a dashboard should be designed with the objective to allow a user who does not have experience as an analyst to quickly get information without sitting through spreadsheets and raw databases. The users of a dashboard use it to obtain valuable abstractions of information, and a lot of the time, those abstractions need to be shared with the user peers. Although a dashboard can be a highly interactable system, a typical method to facilitate sharing a dashboard is implementing the functionality to convert for a document format (for example, PDF) as a static dashboard [Wexler et al., 2017]. That functionality can be complemented with the addition of customisation by allowing specific parts of a dashboard to be included in the final document.

Besides graphics, a dashboard uses legends, filters, text, colours, and symbols to help the user focus on critical data entries offering an overview and allowing to access more detailed information with interaction. It is then relevant to point out that although a dashboard might have a splendid design, it is only useful if the techniques used successfully abstract the input data. The opposite can also apply, problems occur if the techniques are useful but the overall aesthetics interfere with the use of the dashboard, so a section that explores the concepts of design is presented next.

2.1.4.2 Key Performance Indicators

In [Kaplan and Norton, 1996] the authors compare the guiding of an organisation with the piloting of a plane. The pilot cannot effectively guide with no metrics such as a fuel gauge or altimeter, for example, and people who make decisions are no different. They need a full set of reliable metrics about the organisation's environment and performance to successfully achieve the best outcomes in the future.

Key performance indicators are mostly quantitative information and express the structures and processes of an organisation [Badawy et al., 2016]. Such metrics aid everyday decision making and guarantee progress towards a particular goal. Performance measures are not used to aid the implementation of strategic initiatives but rather to guarantee that an organisation's members and actions are well spent and defined towards a specific critical success factor [Parmenter, 2012]. KPIs' general concept is based on taking technical data and presenting it in a relevant and transparent manner to the entity evaluated. [Peterson, 2006] provides some general norms to establish what KPIs are:

- Represented in percentages, ratios and averages instead of raw numbers.

- Expressed in graphical ways that clearly indicate the state like tachometers or thermometers instead of pie or bar charts.
- Capable of providing a temporal context and highlight change instead of presenting raw data.
- A set of methods that drive critical action.

KPIs are mostly used when considering commercial environments. However, in [Parmenter, 2012] it is stated that KPIs are as crucial for government and non-profit agencies as they are for the private sector, and that to implement those measurement practices there needs to be a radical change in the method performance measurement and management is addressed. The author complements this by claiming that government and non-profit organisations might have a scarcity of resources and that using performance indicators to evaluate and optimise their use is beneficial for numerous people.

Most organisations work with the wrong measures, and a lot of them are not real KPIs as only a few organisations have explored what a real KPI is. So many of those companies have measures that they name as key performance indicators incorrectly [Badawy et al., 2016], to describe what they really have, four methods of performance indicators are presented [Parmenter, 2012]:

- **Key result indicators.** Present an overall view of how an organisation is performing.
- **Result indicators.** Provide information on what has been done.
- **Performance indicators.** Transmit information on how a specific team or department is delivering.
- **KPIs.** State how the organisation is doing in performance in terms of a critical success factor that by being monitored offers the possibility to increase performance drastically.

One vital fact to explore in the effects of performance indicators is the possibility of unintended behaviours. Every performance measure might have a negative consequence, and it is a myth that most measures implicate an increase in performance [Parmenter, 2012]. To demonstrate this, the author presents the example of a hospital environment: The managers were concerned with the time it took to treat victims in the accident and emergency ward. So they decided to apply a measure to the time a patient took from the registration in the institution until the end of the treatment. The result was the nursing personnel asking the paramedics to leave less critical patients in the ambulance waiting until an adequate doctor was ready, with the intention of improving the average time of the measure implemented, this measure could then hurt other services of the same organisation that are just as important for the overall well-being of the environment. This example emphasises the importance of carefully introducing measures even if they directly help in a relevant factor prioritising quality rather than the quantity of those same measurement indicators. Another important fact is that in most cases, a KPI that is more than a few days old is almost useless [Parmenter, 2012], and so KPIs should be updated in real-time when applicable.

2.1.5 Design

The overall design of a system is of paramount significance. A healthy design is regularly unnoticeable because it feels intuitive and natural. Accordingly, if a user does not notice problems in an interface, it is most likely because it does not get in the way of the tasks he wants to finish. It is human nature to commit errors. So, systems should cater to that starting on the design, reducing chances of occurring mistakes and minimizing the consequences when mistakes do happen.

Design is a word that can mean much in information visualization, from how an interface is constructed to how graphics and figures are produced. According to Wilke in [Wilke, 2019], if you are a scientist, analyst or anything else that needs to prepare technical documents it is of most importance to have the ability to make compelling visualizations, more often than not as figures. The starting point is that in an optimal design, the interface is intuitive and usable as well as aesthetically pleasing. However, the view's design must never cause a disadvantage in usability. In this section, several relevant terms are described as well as essential principles that were presented over the years.

Often obstacles exist that render the design not so invisible. Maybe it is not self-explanatory, as a system should always aspire to be as intuitive as possible. Perhaps it is challenging to use, and forgettable, as bad aesthetics or poor performance are more than sufficient to disengage a user from a system and make it extremely likely that he won't come back. It could be distracting, as a design must avoid overwhelming a user. Although the amount of simplicity achievable depends on the context, simplicity in a system interface should always be a priority.

Following the discussion on how to represent information bestowed in the last sections, it is now opportunistic to examine how those representations can be best presented. According to Spence, there are three principal resources ready for beneficial exploitation when designing a presentation of data [Spence, 1980]:

- **Space.** This term depicts the size of the display. Nowadays, it can go from a small phone screen to a television screen that occupies an entire wall.
- **Time.** The overall time available for the completion of a specific task.
- **Characteristics of the user.** This resource refers to the human visual processing system. Although multiple factors can adjust its effectiveness, the human optical system is unalterable, and so the designer must cater to it.

The human eye receives light and transforms it into electrical signals that end up flowing to the brain. Our perception of size varies according to various factors like the visual angle and perception of depth [Rozanski and Haake, 2017]. Visual acuity is a term utilised to describe the human ability to discern small details. Contrast and luminance are linked, and so higher levels of brightness mean better visual acuity to a certain extent. As the eye perceives colour by handling light of different wavelengths, and because the fovea (part of the eye responsible for sharp central vision) is 3 to 4% sensitive to blue, that colour has a slightly lower acuity level than all the others [Rozanski and Haake, 2017]. However, this usually does not result in the use of blue being a disadvantage as the difference is minimum. The importance of colour in the

construction of an interface is enormous, and so a section about colour is introduced later in this document.

The concept of aesthetically pleasing depends somewhat on personal taste. However, as the resources ready for beneficial exploitation when designing a presentation of data are enumerated, it is essential to look at some design principles and guidelines that take full advantage of the resources presented (space, time and user characteristics). With that in mind, this section contains a sub-section that lists and describes some valuable design principles.

2.1.5.1 Design Principles

As a sound visualisation system is not possible without a good design, accordingly to Norman in [Norman, 2002], two of the essential characteristics of a good design are **Discoverability**, a user must easily make sense of what actions are available and where and how to execute these actions; considered by the author one of the design principles (presented below) and **Understanding**, how one is supposed to use a system. A user must always have a clear view of what is going on. Several of the design principles directly influence the overall understanding of a system.

Design guidelines are a combination of rules and recommendations on applying design principles. Design principles are high-level design goals, generally based on theories of HCI. The guidelines are specific and context-dependent rules that can be followed to achieve the principles. A design's priority is to accommodate human needs, capabilities, and behaviour, and a good design requires an excellent machine to person communication. Effective communication might be relatively easy to achieve when no errors occur, and processes work harmoniously. However, when things go wrong, it is imperative to communicate to a human what went wrong and perhaps what the user can do to solve it [Norman, 2002]. The book enumerates seven fundamental principles of design, **discoverability** (mentioned before), and six others:

- **Feedback.** There is a reaction for every action, or at least it should be as continuous information about the results of activities, and the current state must be available.
- **Conceptual model.** It refers to how the design projects all information needed and explains, usually in a significantly simplified manner, how a system works leading to understanding and a feeling of control.
- **Affordances.** The term refers to an indicator that allows the user to understand a particular product's purpose. Affordances determine what actions are possible and when the affordances of something are perceptually obvious, it is easier to know how to interact with it. This term makes sense for interaction with physical objects. So the author introduced a term in a later edition of the book called **signifiers** that better explains how to handle interaction with virtual "objects" like an interface.
- **Signifiers.** This term refers to the communication signals that transmit where the action should take place. Fair use of signifiers guarantees discoverability. Thus the feedback is well communicated and intelligible. According to Norman, signifiers are far more relevant to UI designers than are affordances.

- **Mappings.** It means the interconnection between two sets of elements that signals cause and effect.
- **Constraints.** Refers to the act of limiting the set of possible actions to increase clarity.

Although several authors present valuable principles for developing a design, those are just to help guide the whole process, as they do not guarantee a perfect end result. The principles just enumerated were conceived to work for the majority of products, and so they only present general insights. It is then useful to look at more specific principles. Tufte explored principles more tailored for the field of infoVis, and he used Minard's figurative map (approached in this document before) to describe the six Fundamental Principles of Analytical Design [Tufte, 2006]:

- **Comparisons.** Refers to the contrasts and differences exposed in a representation and how they should be relevant to the context. Tufte goes further with this point by claiming that if a visualisation purpose it to aid critical thinking for the reader, it must display comparisons.
- **Causality and Underlying Structure.** Describes the act of having a cause to what is shown within a representation.
- **Multivariate Analysis.** This brief principle states that representations should present more than two variables. Tufte's main justification for this principle is that the real world is multivariate.
- **Integration of Evidence.** Refers to the use of extra elements on graphics such as words and images for the sake of integrating enough evidence in a representation.
- **Documentation.** This principle refers to how graphics should be rightfully labelled with scales, sources, and more relevant information.
- **Focus on Content.** This last principle explains that if the content being described is of low quality, it cannot be salvaged by a representation.

The principles of analytical design offer a set of rules that prioritize assisting the reader of a graphic achieving critical thinking. Tufte claims the principles he presents have effect independently of language, culture, technology or even time (as he states that these principles are applicable for "the first map scratched into stone 6,000 years ago, and also to modern scientific displays") [Tufte, 2006].

Another recent approach is the principles of effective graph design [Zacks, 2020]. This set of principles also has a focus on graphic design. Still, they offer a more user-centred design approach:

- If there is the need for quantitative-based decisions, graphics that use position or length are preferred opposed to the ones that use area or intensity.
- The organisation of patterns in the visual features should reflect proper grouping in the data.
- Visual illusions are to be avoided as they might distort the perceiving of the data.

- The number of visual comparisons needs to be as minimum as possible.
- Minimise demands on aspects related to the user memory.
- If it is possible, standard conventions should be followed.
- If there is the absolute need to break standard convention, it is mandatory to respect broader information, for example, on how values map into a position.
- Make use of the cycle of experimenting, gathering feedback, iterating.

Although, the principles of effective graph design just described are based on theory and collected data, every data set still offers specific context problems, so it might happen that two principles conflict [Zacks, 2020].

Most designs are subjected to continuous incremental innovation by being tested every time they are used and modified if a relevant problem is discovered. As stated in [Norman, 2002], most interfaces evolve through incremental innovation by continuous testing and consequent refinement. This process is referred to as hill climbing in the book and is described as the secret to incremental innovation. It works as the name suggests if a change is made to a system, it is tested. It keeps getting tested and consequently changed until the alarming functionalities turn into valid ones. If no more functionality flaws are detected, then the system is a local high of a hill and not necessarily at an optimal state. Consequently, if done correctly, it does aid in reaching the top of the hill, the problem is if the current design is not the best hill to climb in the first place.

2.1.6 Colour

Colour is obviously a tremendously important factor to understand and master in data visualization. In [Wexler et al., 2017], it is said that colour should not be used just to enliven a dull visualization and that statement is supported with the claim that many information visualizations do not use any colour besides black and white and are still useful and beautiful. Colour should be used with a clear purpose, that could be, for example, to get the attention of the user, highlight a portion of data or distinguish elements or categories. The authors of [Wexler et al., 2017] claim that colour should be used in three principal ways. The authors also make available an image (see Fig. 2.12) that effectively demonstrates the several uses colour can have. The uses are based on common-sense and so are straightforward, but it is still relevant to describe them:

- **Sequential.** A method suited for ordered data. Use of a single colour that gains saturation the higher the data value, so light colours mean low data values and dark ones mean high data values. The colour scheme must guarantee that it changes consistently across all its scope [Wilke, 2019].
- **Diverging.** This hue use can be described as two sequential colour schemes put together but with emphasis on the mid-range values and extremes or a range diverging from a midpoint. The method goes from dark to low saturation with the first colour scheme, and when it meets the middle a neutral shade is used, then it goes from the lightest colour to the darker one of the second sequential tone scheme. The extremes usually have contrasting hues for a clear view.

- **Categorical** (or qualitative). Offers a simple use of colour. Each element/category has a different colour, similar but distinguished colours can be used to represent related categories, but usually, the colours picked are clearly distinct from each other and are easily distinguished between themselves. In the last case, in the set of colours used, none should stand out as to not create the impression of an order [Wilke, 2019].
- **Highlight or alert.** Colour contrast is used to ensure the user sees a piece of specific information. The colour to highlight something depends on the context, but the colours commonly accepted in our culture for an alert are usually yellow, orange or red depending on the severity.

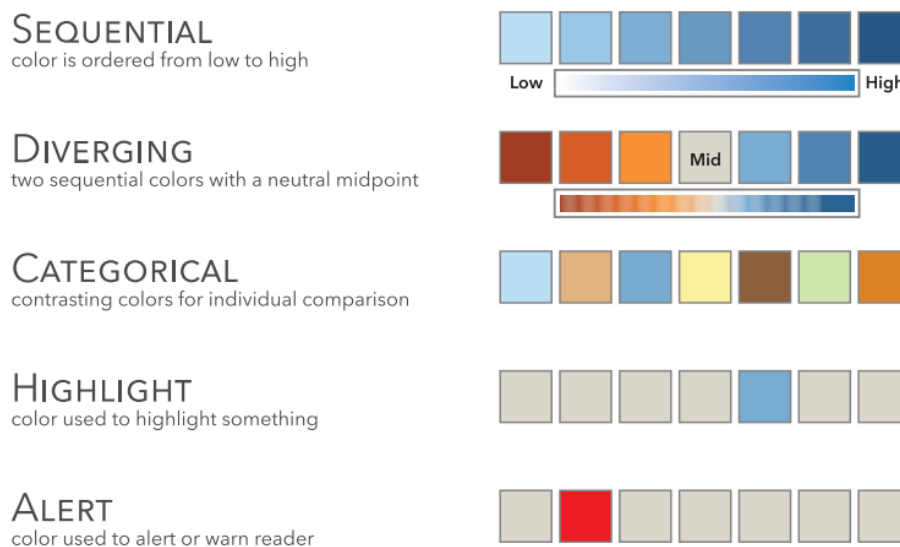


Figure 2.12: Use of color in data visualization [Wexler et al., 2017].

Some other aspects are essential to have in mind when applying colour to an interface. The distinction of colours used in a display should not be affected by changes in contrast. The hues used should also correspond to common customs and user expectations when necessary, as colour conventions are culturally dependent [Rozanski and Haake, 2017] (for example, red means danger for western cultures but in China symbolizes happiness and good fortune [Rozanski and Haake, 2017]). For the sake of accessibility, when displaying indicators through colours, it should not be the only signal as many users and potential users might have some type of colour vision deficiency. This condition is further explained in the next sub-section.

2.1.6.1 Colour Vision Deficiency

This deficiency is mostly hereditary and is caused by the weakness or the lack of one of three kinds of cones within the eye required to observe all hues. So although this deficiency is usually called colour blindness, that term is not entirely accurate because people who suffer from colour vision deficiency can see shades but cannot equivalently recognise distinct colours as the rest of the population [Wexler et al., 2017].

People with normal colour vision use all three types of light cones correctly and are known as trichromats. Some people might have what is known as anomalous trichromacy, with this condition all three cones are used but one of the cones perceives light slightly out of alignment. Subjects with this condition can have an almost perfect colour perception to almost absence of perception on the tint in question. Each one of the types of anomalous trichromacy can be described as a weakness to see a specific colour and are called:

- **Protanomaly.** Reduced sensitivity to red light.
- **Deuteranomaly.** Reduced sensitivity to green light.
- **Tritanomaly.** Reduced sensitivity to blue light and is an extremely infrequent condition.

On the other hand, if a person only has two light cones capable of perceiving colour, their condition is called dichromatic colour vision. They cannot perceive a specific section of the light spectrum. The three types of this kind of CVD (Colour Vision Deficiency) are:

- **Protanopia.** The lack of long-wave cones.
- **Deuteranopia.** The lack of medium-wave cones.
- **Tritanopia.** The lack of short-wave cones.

Protanomaly/Protanopia and Deuteranomaly/Deuteranopia are the most common CVD types. People with these conditions similarly perceive the world, and it occurs more on males than females. According to the statistics presented in [Hasrod and Rubin, 2016], one in every twelve men has some kind of red-green colour vision deficiency compared to only one in every two hundred women. This is because the gene that is responsible for this condition is carried on the X chromosome and it is mostly a hereditary disease³. People with this kind of deficiency confuse hues in the red-yellow-green spectrum. An even more severe type of CVD exists, called monochromacy, and is extremely rare. People who suffer from this condition can see no colour and see different grey shades that go from white to black. A simulation of what might happen for a person with these conditions is available in Fig. 2.13⁴.

Subjects with protanomaly/deuteranomaly have poor red-green hue discrimination and so have difficulty understanding differences between the colours in the red-yellow-green spectrum. With the condition of the more severe protanopia, pure reds cannot be seen and look black, orange-like colours might look like very dim yellows. Orange, yellow and green shades appear as similar yellow hues, and purple tints are indistinguishable from blues. With deuteranopia light greys can be confused with pale pinks, mid spectrum reds with mid browns and blue and green colours might be considered grey and pink⁵.

³More information available at <https://www.colourblindawareness.org/colour-blindness/causes-of-colour-blindness>

⁴Image source: <https://www.colourblindawareness.org/colour-blindness>

⁵More information available at <https://www.colourblindawareness.org/colour-blindness/types-of-colour-blindness>

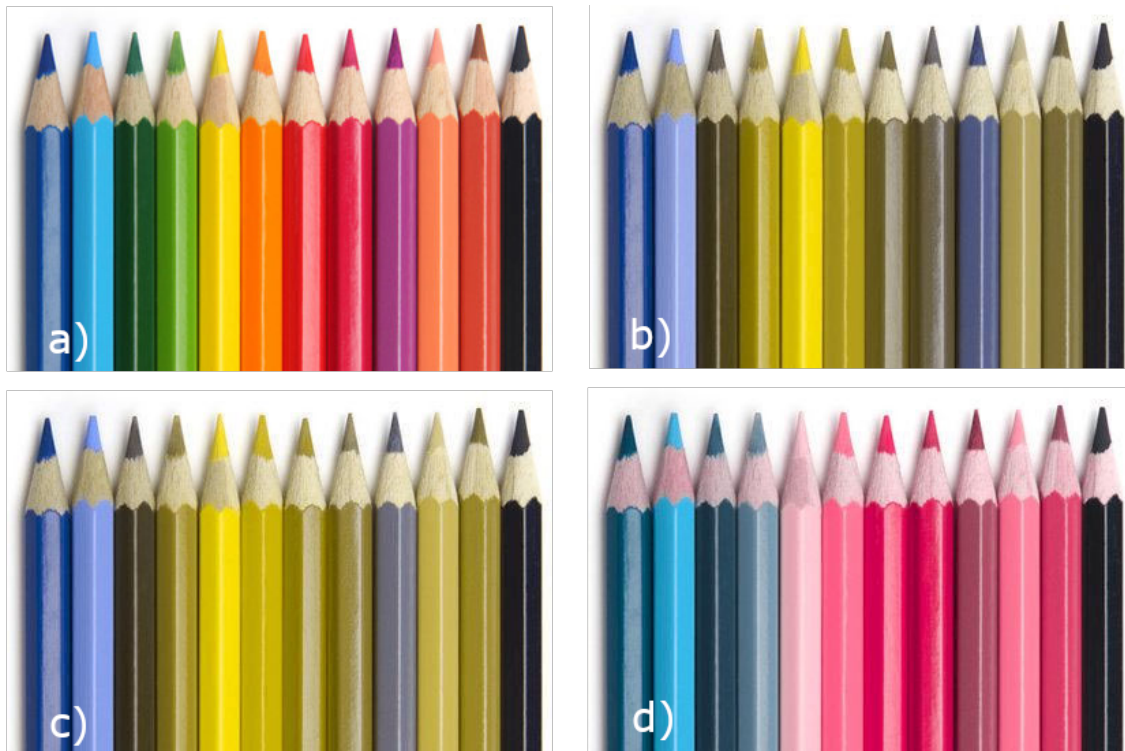


Figure 2.13: Simulation of CVD. a) Normal Vision. b) Vision with protonopia. c) Vision with deuteranopia. d) Vision with tritanopia.

In [Wexler et al., 2017] it is claimed that a problem with people with CVD is in distinguishing the colours red and green. The combination of green and red is approached because the traffic light colours are commonly used in our culture to signify “good” and “bad”. However, differentiating colour for someone with CVD is more complicated than the red and green problem previously stated. A solution presented by the authors recommends using blue to replace green for “good” and orange for “bad” (see Fig. 2.14) and although it might still be a problem for some people those cases are rare. Another way to adapt for people with CVD is by using different indicators besides colours such as icons, arrows or labels, or to show another visual variable, such as stripe patterns. Another option is to have an option either related to a specific visualization or tight to the whole system to adapt the colours for a specific situation.

2.1.7 Evaluation

Each type of information visualisation technique requires a specific line of study to understand if it is the most appropriate solution [Freitas et al., 2002], the authors point out that to evaluate infoVis techniques three usability issues should be taken in consideration:

- **The visual representation.** Represents the overall quality of the static visualisation.
- **The usability of the interface.** Refers to the interaction presented in the technique.

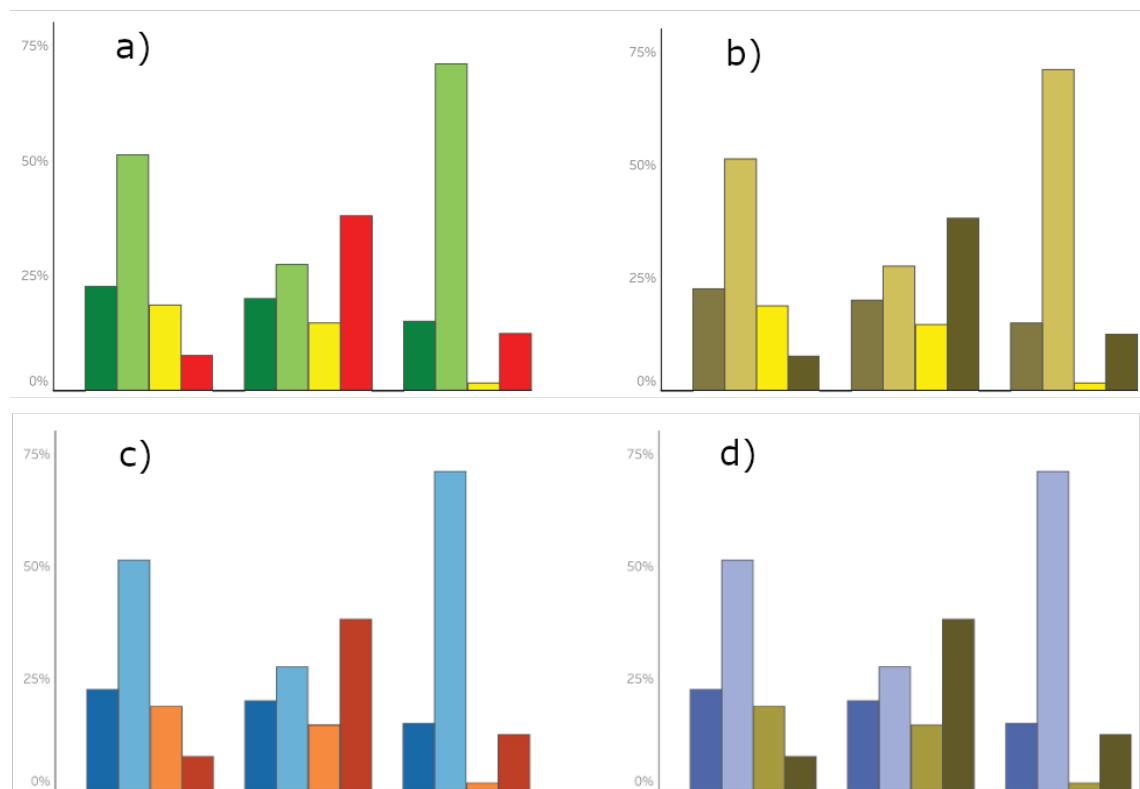


Figure 2.14: Bar chart to demonstrate the problem with the traffic light colours [Wexler et al., 2017]. a) Traffic Light Colours. b) Protanopia simulation to illustrate the problem. c) Possible CVD friendly colours. d) Protanopia simulation to illustrate a possible solution.

- **The usability of data.** Refers to the quality of the data used in the representations.

It is of most importance to look at some examples of what has been suggested to validate infoVis systems. Obviously, our own eyes can be critical concerning a graphical representation's effectiveness. Suppose a person tries to critically judge their own graph creation, for example. In that case, the result is a bias evaluation as the creator already understands the message the visualizations is trying to convey [Zacks, 2020], the author goes further by stating that it is unfeasible to disable the creator bias which makes it hard to "see through the eyes of nonexperts".

One type of evaluating is called Heuristic Evaluation which is a widely used method on the field of HCI [Forsell and Johansson, 2010]. Heuristic Evaluation in HCI is a method that uses usability experts to make judgments based on how the system satisfies a set of predefined heuristics. In the context of infoVis systems, the heuristics from HCI do not satisfy the needs for a successful validation, and so new heuristics were introduced [Forsell and Johansson, 2010]:

- **Encoding of information.** "Perception of information is directly dependent on the mapping of data elements to visual objects" [Forsell and Johansson, 2010].
- **Minimal actions.** Refers to the number of steps needed to complete a task.
- **Flexibility.** Refers to the different ways to achieve a specific objective, and so refers as well to customisation.
- **Orientation.** Describes useful supportive features, for example, redo/undo.
- **Spatial organization.** Refers to the layout organisation of a system. Important characteristics to keep in mind are legibility and the possibility of distortion of some visual representations.
- **Consistency.** This heuristic concerns the consistency between components of the same system.
- **Recognition rather than recall.** This concept was approached before in this document. Refers to a person's memory and how it should not be required for one to remember specific information.
- **Prompting.** When several actions are available, the user should have ways to easily predict what the outcome of each one is.
- **Remove chunk.** Refers to any extra information or graphics that are perhaps distracting.
- **Reducing of a data set.** If the data abstraction methods are effective, taking into consideration their context.

Although these heuristics are not necessarily the optimal ones for information visualisation, they are a potential candidate as they originated from a study that started with over 60 heuristics [Forsell and Johansson, 2010].

The validation of a system is highly dependent on the system itself, and evaluation methods need to be adapted. For example, in [Yin et al., 2015], the authors present

a web-based geovisualisation system for the visualisation of multi-modal urban accessibility. To validate the application's usefulness, they gathered insights from six participants directly connected to the system.

2.1.8 Conclusion

In this section, a deeper look was taken towards the field of information visualisation. At first, the field's context is discussed; this was done with the aid of the highly referenced (throughout books and articles) Minard's map. Much research on infoVis is on improving current techniques and developing new ones. Some standard techniques were looked up and described in this section, which helped gain useful insights, such as the fact that there is much literature that recommends against the use of pie charts and similar representations. After, methods to classify techniques were introduced. Many authors on this topic try to present their own taxonomy although sometimes as an incremented version of a previously established one.

A broader look was taken towards dashboards and the KPIs they often contain. It is interesting to point out again that all KPI have a disadvantage and so one of the objectives is to find the type of KPI that makes the disadvantage as small as possible. Although a highly opinionated topic, it is always important to look at the design and its principles. Another extremely relevant topic in infoVis is the use of colour and the CVD condition. Lastly, a brief research on evaluating infoVis systems was conducted. Most of research on HCI is on evaluating digital systems, and although some were introduced over the years to cater to the field of infoVis, it still offers small amounts of literature.

2.2 Geovisualisation

This topic is connected to the previously presented topic of information visualisation. As the name suggests, its focus is on geospatial information exposure. This field goes further by making use of recent developments in technology for the use and update of digital maps in real-time.

2.2.1 Context

Cartography is an ancient topic, with the digital era, the more recently included concepts of dynamism and interactivity offered new possibilities. Consequently, several terms have been used over time to describe this new type of digital cartography, some examples are geovisual analytics, digital cartography, geovisualisation (sometimes shortened to geoVis) and computer cartography, as a consequence, nowadays it might be challenging to draw a line between all of those terms [Çöltekin et al., 2017]. Taking into consideration that the term geovisualization (abbreviated form of the expression geographic visualisation) started slowly replacing other similar terms since around 1990 (see Fig. 2.15) it is the term used in this document. Figure 2.15 was produced replicating the same experiment as in [Çöltekin et al., 2017] but accounting for the most recent years (1960 until 2019) as the latter accounted for years 1960 till 2008.

About 80% of digital data generated nowadays has some kind of information about geospatial referencing [Zichar, 2013]. That data is collected using "ground surveying,

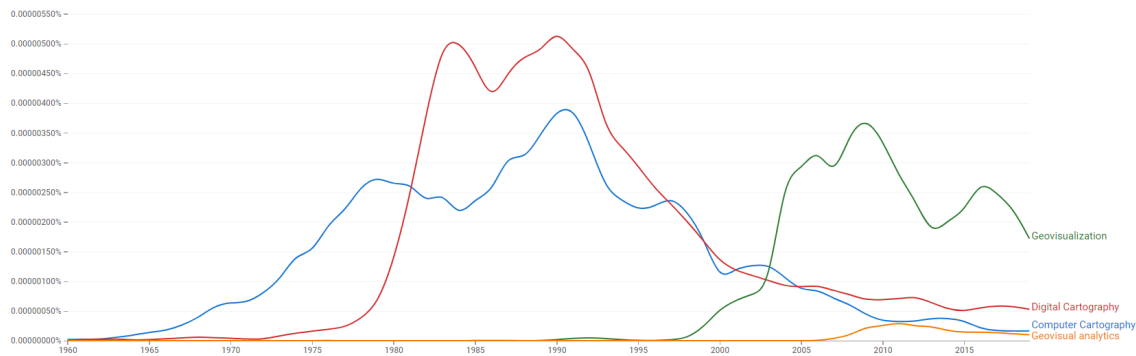


Figure 2.15: Rise and fall of specific terms to describe the digital cartography concept in an extensive book data set using Google’s online Ngrams tool (<https://books.google.com/ngrams>).

photogrammetry, and remote sensing and more recently through laser scanning, mobile mapping, geo-located sensors, geo-tagged web contents, volunteered geographic information, global navigation satellite system tracking and so on" [Li et al., 2016]. As spatial data sources drastically evolve in quality and quantity, the problem of finding meaning on that data rises. Consequently, methods to develop interactive maps, that can successfully help on generating insights from unstructured and complex data, must keep up with all the information provided. A good visualization eases the process of analyzing and investigating collected geospatial data, so it has always been an important topic in geographically related fields [Al-Kodmany, 1999].

Geovisualization englobes the interactive examination of geo-referenced information and is crucial to aid people in identifying worthy patterns. Digital maps come as platforms to display information and render a compelling visual gateway for data evaluation. Considering the subject nature, it highly depends on the development of other topics such as information visualization, geographic information systems, cartography, exploratory data analyses, and image analysis [Kilsedar and Brovelli, 2020].

A GIS is a digital tool for mapping and analysing information and patterns. It integrates everyday database operations with the different visualisation and geographic analysis advantages granted by maps. The use of such systems has collected attentiveness from public and private organisations, mostly because of their capacity to successfully manage and analyse spatial data [Balasubramani et al., 2020]. Most open source GIS software is single and often oriented to a certain application [Andrienko et al., 2010], but nowadays plenty of tools to achieve a working GIS are available. Knowing that geospatial data has been consistently used more on the web rather than desktop applications, some recent researches evaluated the tools available for web geovisualisation, with focus on open-source solutions. The overall results show that the use of freely available tools is enough for publication of high-quality geographical information on the web [Balla et al., 2020, Kilsedar and Brovelli, 2020].

2.2.2 Techniques

- **Point map.** Also known as dot mapping, as the name suggests, uses a point to represent a variable (see Fig. 2.16 a)). Points are easy to map and show the

exact position of the geospatial information. They also give general information useful to compare different areas of a map. A good method to display data with a wide distribution of geospatial data. The method's big weakness is that overlapping points are not detectable.

- **Bubble map.** Circular shapes are represented over designated geographical regions. The area is proportional to the dimension of the value it represents⁶. They can represent two different variables, one signified by its size and another by varying its colour. Although good for comparing proportions, a problem similar to the one identified on the point map technique exists, in this case, bubbles can overlap other bubbles. This flaw can be eased by introducing: interactivity to allow zoom on specific parts of the map; transparency for each bubble; a method to click each bubble to offer a way to the reader to get more detailed information⁷.
- **Choropleth Map.** Represents information using different and impactful colour schemes for the different countries, cities, regions or some customized division. In Fig. 2.16 c) the colour scheme goes from the typical green that represents a low amount of fires in a specific region to the typical yellow to the red that represents increasing fire incidents. With choropleth maps conclusions towards a specific geographical area with normalized values can be successfully achieved, although the results might sometimes be deceiving as areas may be heterogeneous regarding different types of entities. An alternative called **cartogram** exists, in which the regional areas are inflated concerning their data dimension. This alternative was introduced to correct the problem of regions looking important because they happen to have a bigger territory. However, this technique distorts the real map and needs to be presented with an introduction or by showing the original map before⁸.
- **Hexbin map.** Similar to choropleth mapping but the division is made so that each region represents an equal area thus removing the bias that can happen in choropleth maps, but the drawback is that it can become a difficult task for the map reader to recognize a specific area. However, this can be solved by introducing labels on top of the hexagonal data⁹.
- **Heat map.** Works a lot like choropleth mapping but it is not restricted to geographical boundaries (see Fig. 2.16). Heat maps are versatile if the objective is to get the concentration per unit of the area; hence the method can provide information about significant clusters. However, it does need much-georeferenced information as input to achieve satisfactory results and conclusions cannot be achieved towards an exact position but rather an area.
- **Connection Map.** In this technique, points are connected by lines that usually represent the shortest route. As they account for the earth curvature, the result is rounded lines that give a pleasant visualization. An effective method to display geospatial connections, relationships and map routes¹⁰.

⁶More information at https://datavizcatalogue.com/methods/bubble_map.html

⁷Source: <https://www.data-to-viz.com/graph/bubblemap.html>

⁸More information: <https://www.data-to-viz.com/graph/cartogram.html>

⁹Source: <https://www.data-to-viz.com/graph/hexbinmap.html>

¹⁰Source: https://datavizcatalogue.com/methods/connection_map.html

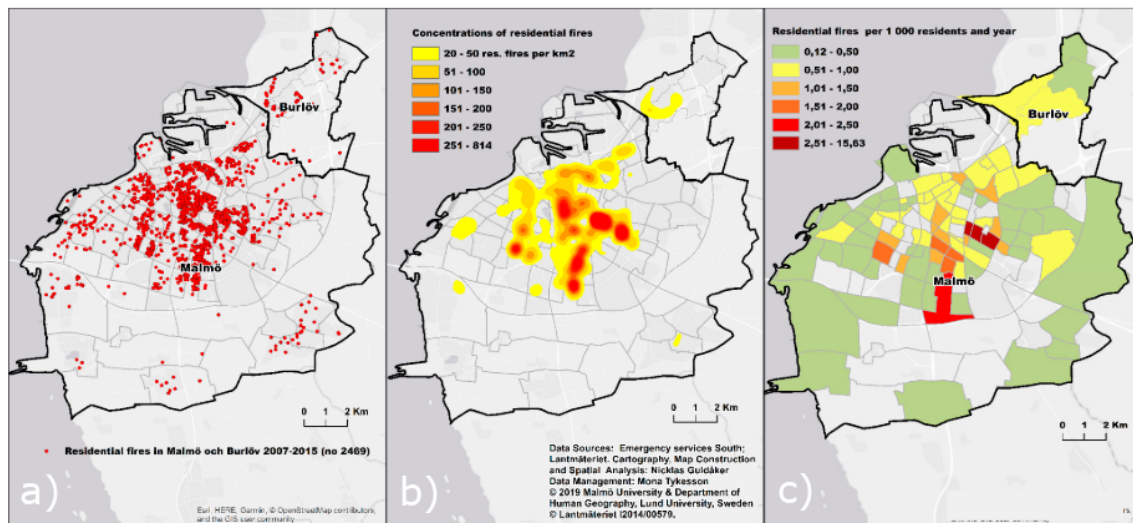


Figure 2.16: Example of some geovisualisations. (a) Point map, (b) Heat map, (c) Choropleth map. Municipalities of Malmö and Burlöv, Sweden. The residential fire data period is 2007–2015. Adapted from [Guldåker, 2020].

- **Flow map.** In this type of map usually lines are used to connect origins to destinations creating a visualization great to see events like aviation routes, traffic and human or animal migrations (see Fig. 2.17¹¹). The lines are usually represented by the shortest route between the two points, just like a connection map. The line's thickness represents the amount of data presented in that specific flow. Arrows are of common use to clearly present the way data travels.

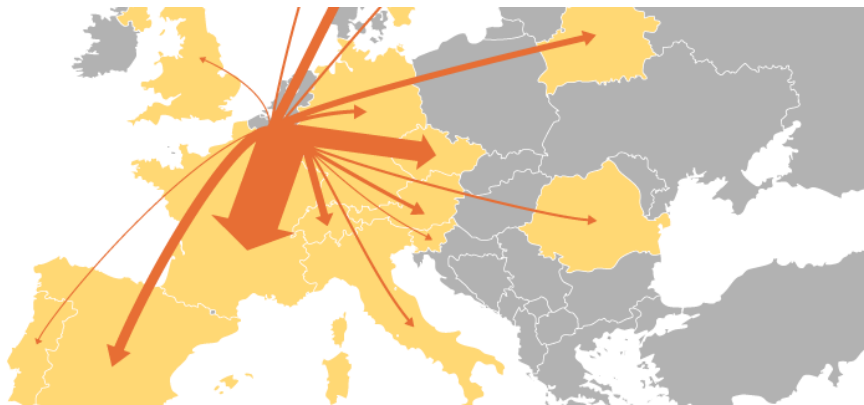


Figure 2.17: Example of a flow map.

2.2.3 Tools and Technologies

In [Balla et al., 2020] one tool discussed was the Keyhole Markup Language, an extensible markup language notation that describes data regarding characteristics with a geospatial location to achieve the goal of geovisualisation. Most GIS software and virtual globes can import and export such type of files. However, it is stated that

¹¹Image source: https://datavizcatalogue.com/methods/flow_map.html

factors were found which restrict the visualisation of KML objects (default display, formatting limitations, and others). Another set of tools were Google APIs which are JavaScript interfaces that can be utilised to manage the functioning of a map and are convenient to access and manage KML files, and the QGIS¹² (Quantum GIS) geovisualisation tool and its modules (qgis2web¹³, qgis cloud¹⁴, Qgis2threejs¹⁵) which is a GIS tool that requires no coding skills. In the same article, some freely accessible georeferenced databases are praised with the focus on OpenStreetMap¹⁶ and ASTER Global Digital Elevation Map¹⁷.

In [Kilsedar and Brovelli, 2020], open-source software was used mostly because it allows for a more freely customisation opposed to closed source options. For building digital map on the web two APIs are presented: NASA Web WordWind¹⁸ and CesiumJS¹⁹. Both of those technologies use WebGL, JavaScript, HTML5 and Cascading Style Sheets Level 3, in Fig. 2.18 an example from the same article of a geovisualisation using CesiumJs on a virtual globe is displayed. The presented technologies offer outstanding performance, dynamic, interactive, and cross-platform visualisation of a world with terrain and two and three-dimension geospatial information layers. Another markup language is presented in the article, CityGML, a standard data model for reproducing 3D objects representing relations in topographic objects. It can also be converted into the Keyhole Markup Language.

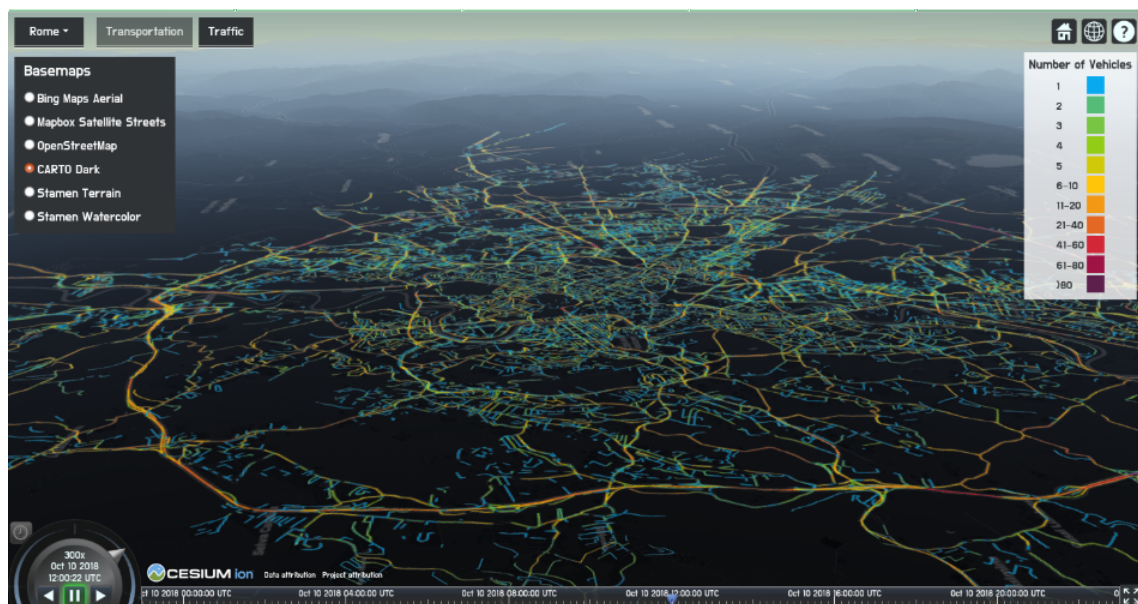


Figure 2.18: Visualisation of traffic in Rome between 12:00 and 13:00 UTC on a virtual globe created using CesiumJS [Kilsedar and Brovelli, 2020].

¹²Official website: <https://www.qgis.org>

¹³Source: <https://github.com/tomchadwin/qgis2web>

¹⁴More information at <https://qgiscloud.com>

¹⁵Source available at <https://qgis2threejs.readthedocs.io/en/docs/>

¹⁶Source: <https://www.openstreetmap.org/>

¹⁷More information available at <https://asterweb.jpl.nasa.gov/gdem.asp>

¹⁸Official website: <https://worldwind.arc.nasa.gov/web>

¹⁹Official website available at <https://cesium.com/cesiumjs/>

“Unfortunately, there is a common and growing conception that geovisualization and the geospatial information science as a whole has been or may be taken over by Google Earth” [Andrienko et al., 2007]. Those kinds of virtual globes on their own are by no means enough to work as an effective support to complex problems, and good support requires tools and methods that help articulate a particular problem goal. Recognising valuable information from the amounts of data available and applying intuitive and relevant interaction is a specific task for each problem. If a geovisualisation platform is not done carefully, the result might prevent the user from achieving desirable and optimal conclusions. Therefore, the data directly shown in a map and its interactivity must be carefully designed and extensively evaluated [Robinson et al., 2017].

Although virtual globes (3D viewers), such as Google Earth, Nasa World Wind, OpenWebGlobe, etc, are not sufficient for an optimal geovisualisation system on their own, recent research shows that virtual globes are pleasant and comfortable to utilise and permit contextualising the information represented more adequately compared to three-dimension maps [Kilsedar and Brovelli, 2020]. Hence recent geoVis methods have gone pass by the static 2D products and three geographic dimension models through proposing compelling and interactive 3D and 4D data representations, still, the techniques and devices that visualise data in 3D need additional expansion and research, as they are still lacking [Kilsedar and Brovelli, 2020].

2.2.4 Practical Uses

It’s not hard to find relevant and practical studies that use geovisualisation techniques as a tool to extract information, and many have shared characteristics with some of the goals stated in this document.

An example from 2015 was already approached in this document. The authors present a web-based geovisualisation system for the visualisation of multi-modal urban accessibility [Yin et al., 2015]. The authors describe the resultant platform as “a good example of how a platform helps researchers better understand accessibility patterns in a geographical area”.

A study was done with the objective of describing “spatial patterns of infant mortality and preterm” in a city [Root et al., 2020]. The conclusions were that the techniques they applied stimulated relevant discussions not only about the study in question but also about other areas where similar geovisualisation techniques could have use.

Another article [Heinzlef et al., 2020] describes a methodology for enhancing resilience (ability to withstand, learn, adjust and recover from the effect of a hazard) to floods that concentrate on the accessibility encouraged by geovisualisation techniques for risk management to promote stakeholders involvement and understanding of the correspondent topic.

Just recently (2020), a study used such techniques intending to aid and improve the spatiotemporal allocation of emergency vehicles in Lahore, Pakistan, the results were “beneficial for effective resource planning and for understanding the complexities of a highly urbanised city” [Maqsood et al., 2020].

Also in 2020, [Guldåker, 2020] describes how diverse geovisualisation techniques such as point data, heat mapping, and choropleth mapping can complement one another and be utilised in preventive fire work (see Fig. 2.2.3).

2.2.5 Conclusion

This section explores some components of geovisualisation. It is a field that is highly connected to the one of information visualisation. It starts with a brief context that also introduces the concept of GISs. Common techniques were described as well as some broad reasons to choose a particular one. Lots of technologies for the geovisualisation of data were introduced in the literature, although there was a lack of comparison between similar technologies and methods to do so. To finish, it was relevant to look at successful applications of geovisualisation techniques.

2.3 Interaction

Information visualisation systems, widely speaking, are divided into two components, **representation** and **interaction**. Representation stands for the overall data exposition on the screen and how that same representation gets rendered. Interaction represents the whole process of dialogue between humans and the system itself as the user explores the information. As stated in [Yi et al., 2007], interaction eventually triggers some change in the representation, and although treated as different components, representation and interaction are intrinsically connected and positively contribute to the final experience.

There are several interaction styles explored over the years. The classical style of interacting with a computer is through the command-line interface, using parameters that the user has to memorize. Other less complicated products present menu options on just one screen, usually as text-based options or through the use of simple icons or images. One style that gained popularity is the WIMP interface and is generally the norm in every computer's system interface.

Every letter in WIMP stands for an element of the proposed interface. "W" stands for windows, as in areas of the screen that contain any information that can be resized and moved; "I" stands for icons and represent an existing window. "M" means menus and refers to information in the form of a list that can be interacted with using the cursor. "P" means pointers and refers to the different shapes a mouse cursor can have to help the user understand how it functions. The Wimp interface can contain more elements, as stated in [Yi et al., 2007], elements like buttons that are isolated regions that can be select to invoke an action accordingly, or toolbars that function similarly to menus or even dialogue boxes used to bring the attention of the user to transmit critical information.

As a WIMP interface contains other programs that subsequently rely on the same interface strategies, the interactions become consistent in the whole system allowing users to apply previously known skills common to most programs. Considering standard interaction interfaces is of most importance because in interaction, not only the visualisation itself matters. The former experience of the user and the environmental settings all play a role in shaping the learning process [Forsythe et al., 2016].

The information visualization research has generally concentrated more on visual encoding than on interaction, yet interaction is mandatory for an interface [Lam, 2008]. Interaction is an intangible concept, so it is hard to design and evaluate and unlike the representation component, interaction does not have many examples that show how to implement and design it for a system [Elmqvist et al., 2011].

No interaction means static visualisations, which can still be useful, but interaction is of extreme value when overcoming obstacles imposed by significant numbers of data variables. Interaction in visualisation systems might get confused with the definition of interaction in HCI. According to [Yi et al., 2007], a difference is that interaction in information visualisation focuses on changing and adjusting visual representations rather than entering data into systems with forms, which is more a focus of study for the field of human-computer interaction. Nevertheless, they are similar topics, as the authors suggest in [Elmqvist et al., 2011] that research on HCI was essential for them to introduce the concept of fluidity and flow in the context of interaction on infoVis systems. They classify fluidity in information visualisations as an "elusive and intangible concept characterised by smooth, seamless, and powerful interaction; responsive, interactive and rapidly updated graphics; and careful, conscientious, and comprehensive user experiences." To better explain this concept, the authors present a set of properties of fluid interaction in infoVis systems:

- **Encourages flow.** This property states that the interaction must be designed to encourage continuous action and consequent total immersion.
- **Allows for direct manipulation.** This property according to the authors originated from four principles: clear visualisation of the components of relevance; physical actions instead of other kinds of more complex syntax; fast, intuitive and easily reversible operations that instantly impact at least one component of interest; and a method that facilitates the use for users with low amounts of knowledge on the system.
- **Lowers the gulfs of action.** There are two gulfs, the gulf of evaluation that is the difference from what the user perception is of the system state compared to what it actually is. And the gulf of execution, which is the difference of the actions that can be performed in a system compared to the user's actual intentions on the system.

Users acquire interface usability skills mostly through exposure and repetition. More often than not, typical interaction functionalities will feel the most familiar even though they may not render the best feasible implementation. So design judgments made in the first stage of appearing technologies are frequently selected uncritically when more suitable options may exist [Harrower and Sheesley, 2005]. In [Dix et al., 2003] a great example to confirm this point is the one of the keyboards layout, the vast majority of keyboards follow the layout "QWERTY" which is not optimal for typing but comes from the times of the typewriter, they complement this claim by informing that the "DVORAK" layout reduces fatigue and increases typing speed by as much as 15%. A wide range of users needs to be in consideration, and although the average user expects common and simple interactions the power user expects advanced features characterized by the desire to make the most use out of a system. A balance between innovation and standard and complex features must be achieved not to overwhelm

the common user, allowing him to comfortably accommodate new functionalities and new approaches to previously known features, and to please the power user having features and functionalities that the typical user can ignore or does not even notice.

Users have many necessities for features to help filtering data. Some possibilities are "numeric range sliders, alphas-sliders for names or categories, or buttons for small sets of categories" [Shneiderman, 2003] and are called dynamic queries. It is a well-known fact that an action done by the user in an interface should offer some kind of feedback less than one hundred milliseconds after the action was executed and Shneiderman also talks about this fact in the article. This raises the problem of having such a large data-set that computations take more than the desired milliseconds to display relevant feedback.

Interaction in a geovisualisation system (cartographic interaction) includes everything related to how the end-user manipulates a map. A map is a practical vessel for depicting geographical variation in data but is limited in the number of variables it can show, interactive techniques are generally applied to accommodate the presence of several variables, the most common approach is through the power of interaction to smooth the path for geographical filtering [Turkay et al., 2014]. Current GIS science is still inadequate in handling geographic data's temporal nature. Also, methods to represent time-dependent geographical information in static maps are only applicable or adequate for small data sets and small intervals of time [Andrienko et al., 2010].

A study was conducted that identified some broad goals of interaction in a geospatial context [Roth, 2013]:

- **Procure.** This goal refers to the interactions that can be executed to obtain data about the representation in question.
- **Predict.** Refers to the interactions that can be done to predict an outcome with the current state as a base. This point is then a combination of "Procure" to get a full insight into the state and the user's overall knowledge to make valid estimates.
- **Prescribe.** Describes the interactions executed to alter a possible unwanted outcome. This point is then a combination of the last two, as knowledge of the current state is necessary as well as an understanding of what will happen.

Research shows that in the context of map visualization no panning/zooming method is highly efficient for every situation, in an article the advantages and disadvantages of most methods are stated, the methods studied in [Harrower and Sheesley, 2005] were:

- **Directly Re-position the Map.** Works like a grab and drag method, the user clicks the screen and can pan by moving the mouse; consequently, the entire map can be used to pan. It is a simple and natural approach. The disadvantage is that there is no scale connection among the panning and the map.
- **Smart Scroll Bars.** Behaves similarly to traditional scroll bars but only appear when necessary and their range dynamically changes to show what proportion of the document is currently noticeable.

- **Rate-Based Scrolling.** The mouse is permanently in the middle of the screen and pans accordingly to the user movements with the mouse. The big disadvantage is that it takes the mouse hostage, conceding the possibilities of using it to perform other tasks.
- **Keyboard Controls.** Can be faster than other browsing options plus requires no screen real estate. Disadvantages are that such controls might be overwhelming to a novice user and it usually restricts panning to four directions or possibly eight.
- **Zoom and Re-center Under Mouse Click.** A hybrid method that merges zooming and panning into a unique mouse click. High level of precision and optimal if a target is already on screen but impractical if not.
- **Navigator Tabs/Interactive Compass.** Consists of navigator tabs located on the map's extremities and interactive direction indicators to restrict the panning choices to either four or eight directions. Can be easily understood by the user but they do not allow for user-defined measures of action and might be impractical to transverse long distances.
- **Navigator Window.** Capacity to draw a zoom box and grab it and move it around. The user can pan and zoom to exactly the scale and location he wants to in one click and drag operation.
- **Specify Explicit Coordinates.** Self-explanatory on how it works, extremely useful for a user that wants to view a specific location, the input could be in the form of a street address, geographic coordinates, etc.
- **Zoom Box.** Drag and draw a box directly on the map that becomes the map's new extent. The best method if the target is on screen; it also has a minimal footprint, it does not provide global orientation clues, impractical to zoom out, and consequently hard to find a target off-screen.

Consequently, it is concluded that "pan/zoom methods have several attributes, indicating that they are best implemented when matched to particular users and map browsing tasks" [Harrower and Sheesley, 2005].

The feel of zooming and panning a map is crucially vital for a GIS; several studies point towards Google Maps being the most popular navigation system by long margins. Google maps present a hybrid of several panning/zooming methods. The most used one is the method to directly re-position the map (grab and drag), which some advantages are its simplicity and naturalness, it also supports keyboard navigation using the arrow keys and the plus and minus keys to zoom in and out respectively, and it is possible to zoom in and re-centre under the use of a double mouse click. Being the most popular, most users are accustomed to that type of navigation; as stated before users expect similar systems to offer similar functionalities. So the interactivity of a map regarding panning/zooming should have as basis those methods. The addition of more methods should be considered if the characteristics of the set of problems the geovisualisation is trying to help solve require so.

One effective interactivity method is to have several options to graphically represent the same data. [Kraak, 2003] argues that the use of different graphics stimulates

visual thinking and that such representations can help expose patterns that are not significantly evident when conventional map presentation techniques are employed, that same paper uses geovisualisation techniques to expose the information presented in the previously mentioned Minard's map of Napoleon's campaign in different ways. One of the representations applied used the technique called space-time cube where the third dimension can be implemented from a distinct panorama, in Fig.2.19 the X and Y-axis describe the geography and the Z expresses time.

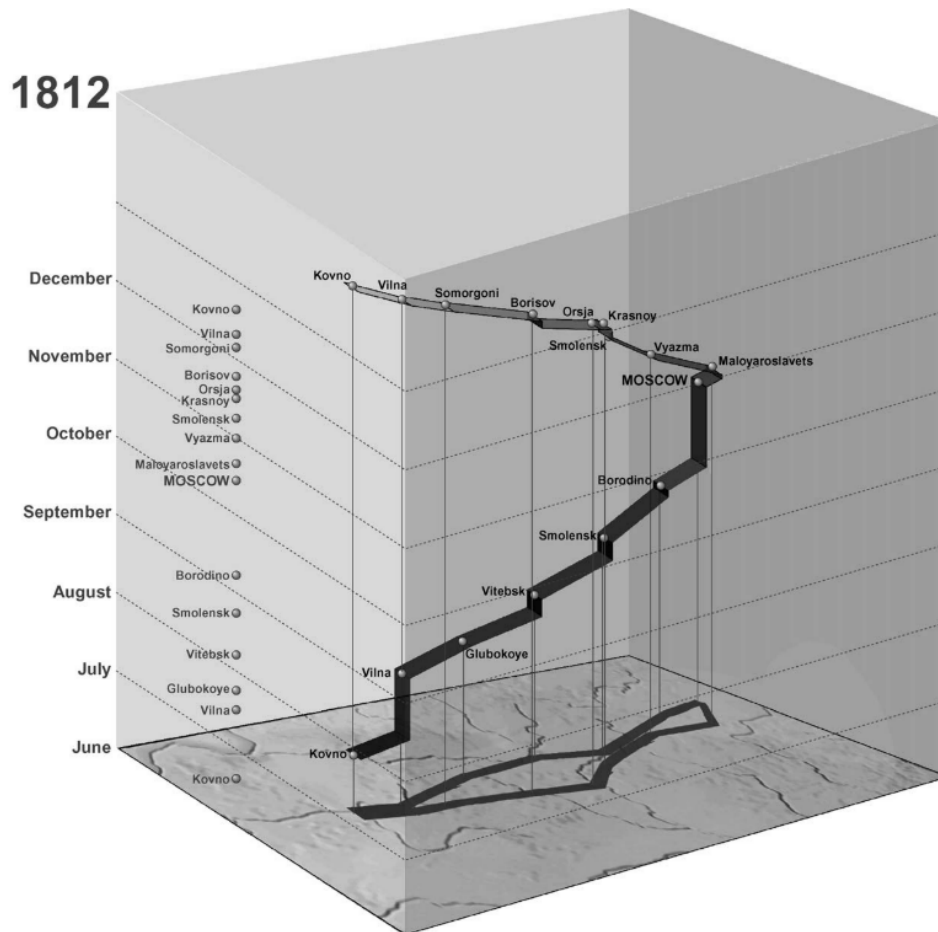


Figure 2.19: A space-time cube of Napoleon's march in Russia [Kraak, 2003].

Having concluded that users acquire usability skills mostly through experience and that there is more than one optimal way of graphically representing the same data there is the hypothesis that information visualisations should be adapted taking the user in consideration. A study with sixteen analytic experts was conducted and showed that when given the option to choose a graphic representation to see some information, there was much variability amongst the experts [Poetzsch et al., 2020]. This introduces the concept of adaptive visualisation. The topic of adaptive visualisation serves the purpose of improving the experience of a user by offering the possibility to intuitively change representations taking in account several users features that could be provided or concluded from the actions a user makes [Ahn and Brusilovsky, 2013].

In this section, general insights were acquired in the context of interaction. From more broad concepts like the fluidity and flow interaction to more concrete studies, for example, ways of manipulating a digital map. The section ends with the exploration of a set of processes to aid the creation of interactive visualisations.

2.4 Visualisation Recommendation and Adaptive Visualisation

Although data visualisation is widely used as a tool in a data analyst toolbox, this process more often than not still involves manual generation of visualisations through tools that demand extra tedious work like Excel and Tableau. Also, the sizes of databases by norm grow substantially but the amount of human attention and time do not necessarily grow with that trend staying constant in most cases [Vartak et al., 2016]. Consequently, with larger sets of information users need to test more attributes and experiment with a larger domain of visualisations before coming up with suitable visualisations. One other aspect to take in mind is that more and more people are performing data analysis, this new set of users have varying levels of skills in statistical and programming techniques supporting even more the need for intuitive and easy to use tools for data analysis [Vartak et al., 2016]. Two interesting research fields that approach the problem referenced are visualisation recommendation and adaptive visualisations. This section is based on those fields and how both offer valuable information but differ on their approach to this problem.

2.4.1 Visualisation Recommendation

The purpose of any visualisation recommendation system is to create, classify and recommend visualisations with the inserted data in an automated way. This visualisation recommendation cycle speeds up the process of achieving useful insights on some set of information data.

According to [Qian et al., 2020] a complete visualisation recommendation system should recommend a set of visualisations organised by relevance, by choosing the proper displays for a selection of variables from a database the system "greatly reduces the amount of time, cost, and effort that human spend in insight discovery process."

As stated in [Gotz and Wen, 2009], normal business users while experts in their area don't typically possess the skills to choose a particularly suitable visual metaphor of all the options available. So visualisation recommendation systems were introduced to prevent companies having to hire professional analysts that do have high levels of visualisation and analysis skills but lack the knowledge on the domain they are working on. In conclusion, visualisation recommendation systems came as a realization that supporting average users is extremely important. Furthermore, in the same article, the current type of these systems can be categorized in 3 categories:

- **Task-based systems.** This type of system classification uses formal visual task descriptions as input to build appropriate visual representations. This means having in most cases some information on the user purpose to use a visualisation.

- **Data property-based systems.** This type of system bases its recommendation on the properties of the information sets being analysed.
- **Hybrid systems.** As the name suggest it is a combination of the previously presented systems. This more complex type of system classification uses the properties of the data in question and also some type of user intent to recommend a valid and adequate visualisation.

The argument provided by [Gotz and Wen, 2009] is that while visualisation recommendation systems already exist and are proved to help reduce the skill barrier they remain hard to utilize by the common user. And so, in that article a specific type of visualisation recommendation system is introduced and it is called **BDVR** (Behaviour-Driven Visualisation Recommendation) and it suggests an algorithm with two stages:

- In the beginning stage the algorithm observes the end-user interactions throughout the completion of his tasks. With that information it applies a rule-based procedure to identify semantically significant interaction patterns.
- In its next stage the identified patterns are consequently applied to yet another algorithm that handles the recommendation process.

This visualisation recommendation method was later applied to a prototype and tested with 20 participants. The results according to the article showed that BDVR does boost user's task completion by reducing the time used in each task as well as reducing task error rate when in comparison to systems that do not use the user's behaviour to suggest recommendations.

2.4.2 Adaptive Visualisation

As the name implies adaptive visualisation is a field that proposes strategies to ameliorate information visualisations by including adaptation. Naturally it differs from non-adaptive visualisations because it takes in consideration specific contexts or user traits, and in consequence visualisations in a system vary from user to user.

2.4.2.1 Context

In the last years research has been conducted that proves that users behave as individuals with completely different desires and so their needs, abilities and preferences have a tremendously significant impact on their performance and satisfaction when using visualisation techniques [Carenini et al., 2014]. This point is supported by the claims in [Toker et al., 2013] that states that there is more and more confirmation that a user cognitive abilities and personality have an extreme influence on the efficacy of the data representations they consult.

[Toker et al., 2013] confirmed the point made on the last paragraph by testing users with the use of an eye tracking device and having them perform tasks on visualisation techniques (bar and radar graphs). Furthermore the authors state that more research is necessary to understand the impact of simple properties of information visualisation such as size, color as well as shape. On that account it is crucial to continue to scrutinize the possibilities of new kinds of adaptive information visualisations that are extremely user centred.

It is also relevant to present the difference between an adaptive system and an adaptable one. While an adaptive system has the capacity and tools to dynamically change behaviour according to some established factors, an adaptable system provides the end-user with mechanisms that grant them the power to alter the structure complexion. Nonetheless, research has shown that the best path to achieve more successful results is by joining both methods [Tan et al., 2007].

While the topic of visualisation recommendation has more focus on selecting the right types of graphics for a certain set of data, the topic of adaptive visualisation focuses more on taking in consideration the users diversity and so the data representation should be adapted to the context but also to the user characteristics [Poetzsch et al., 2020]. In that same article several studies were conducted that justify the need for adaptation and presents some new perspectives:

- The first study purpose was to prove that there is value in using a user-adaptive approach for data visualisation. To achieve this, analytic experts were gathered and asked to choose a type of encoding for different sets of data. The result of the study demonstrated that there is in fact a substantial variability in visualisation preferences for the same data set between participants.
- The second study explored "how user traits impact on the perception of different data visualization encodings, and hence laid the groundwork for adapting to traits.". This study focused on the user potential prior experience, visual literacy and cognitive capabilities. While the first study proved that user that are in the same category of experts have different visualisation preferences between themselves, this study tried to identify what types of graphical representations are more adequate for a certain type of user taking in consideration the user's prior experience, visual literacy and cognitive capacities. With this study, three clusters of visualisation types were identified named "Good Standard", "Suboptimal Standard" and "Multivariate Diagrams". By evaluating the types of graphics in each group with metrics such as the number of participant that did not understand the visualisation and the average of errors per task using a specific type of graph, a set of graphics was concluded as belonging to an expert area (see graph on Fig. 2.20 for a clear visualisation of that study)
- The third study tried to understand how the purpose of the user with the data could take place to change the visualisation requirements. For this, two different user states were taking in consideration. An analyzing state and a monitoring state (see Fig. 2.21 for an example).

2.4.2.2 Models and Frameworks

An extended set of adaptation frameworks have come out over the years. Although included in the same theoretical set, a lot of those frameworks have different purposes, so it is important to analyse the state of art on their regard.

One adaptive system is the **ERST**, which stands for External Representation Selection Tutor. This system is evaluated in [Votano et al., 2004]. This system adapts its visualisation in a subtle but effective way. It offers hints and advises to the user and besides that it hides display forms deemed unsuitable for a specific user. It takes

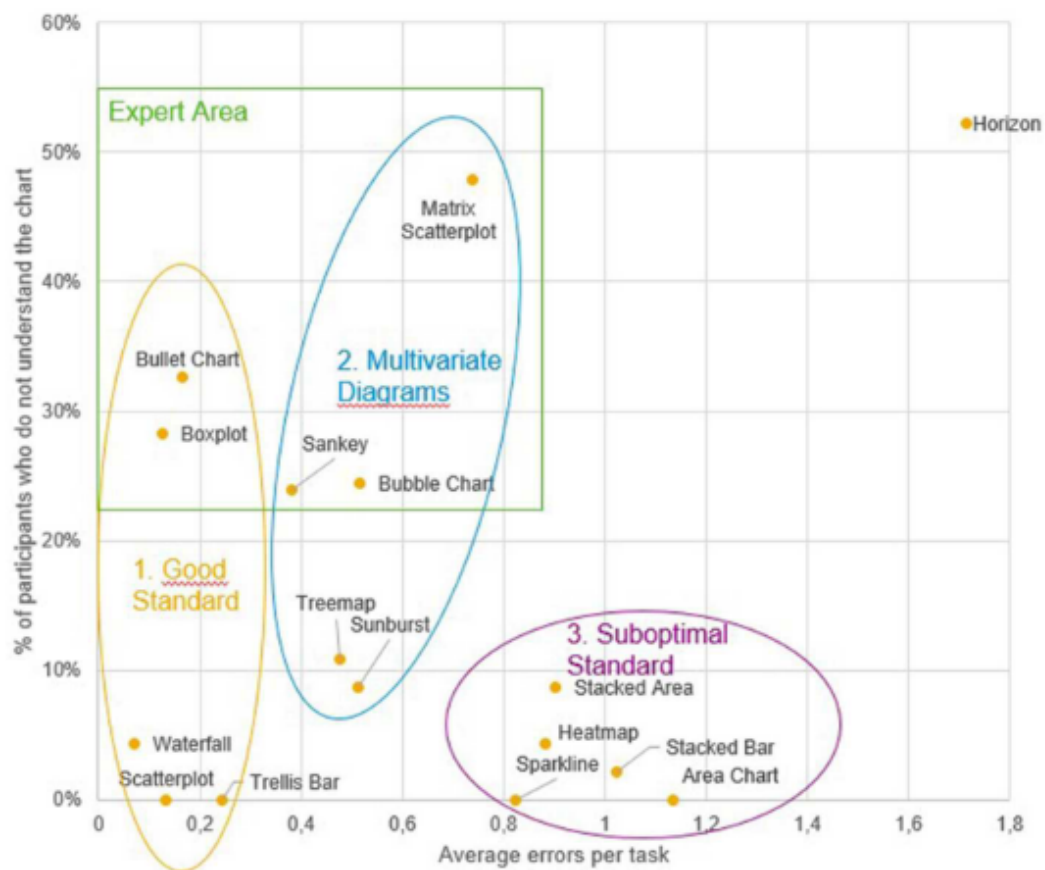


Figure 2.20: Scatter plot of types of graphics identified by emerging clusters [Poetzsch et al., 2020].

in considerations the user error rate in a particular visualisation and then limits the range of information displays taking in consideration those factors. It also recommends graphs taking in consideration the time spent by the user base of a system on selecting a visualisation.

Another completely different approach but not less important is the framework presented in [Tan et al., 2007], which is an adaptive and adaptable tool to improve the accessibility to web graphics for the portion of the population that is visually impaired. The previously presented framework follows a component based path:

- The first component is called Sub-Application Database, shortened as SD. This standalone application handles the aggregated data on graphic segments and translates it towards components that are "haptic, tactile, audio (or a combination of any)" [Tan et al., 2007].
- In second comes the Context Manager layer, also known as CM. This component receives the information provided by the Sub-Application Database part and besides storing it, also manages all the information that is required for the adaptation features. The information required for the adaptation lineaments are as described in the article that presents this framework: "(a) graphical contents, (b)

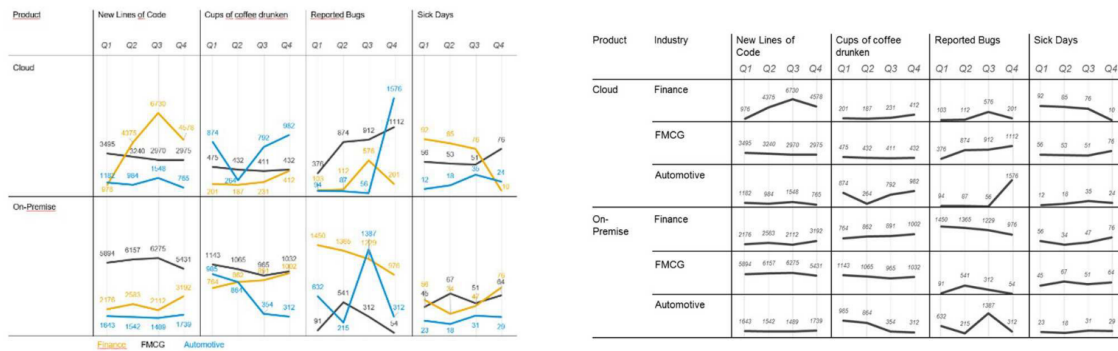


Figure 2.21: Visualisations showing the same data in an analysis setting first and then in a monitoring setting [Poetzsch et al., 2020].

system configuration and hardware/software components ... (c) requirements and feedback provided by each sub-application and (d) user's profile and preferences".

- The next component is the GCS, which stands for Graphical Content system. It offers the launching interface and gathers data in relation to the content explored by the end-users.
- The following component is the Control Centre (CC). This sub-application provides the user with the tools to alter their profile and change system settings and personal preferences. This layer offers the adaptable features of the framework.
- Lastly comes the Core Processor Module (CPM). This last sub-application handles all the Adaptive logic. It takes advantage of the resources accumulated by the GCS and CM layers and in the end provides what it concludes to be the most adequate and efficient final interface for the user.

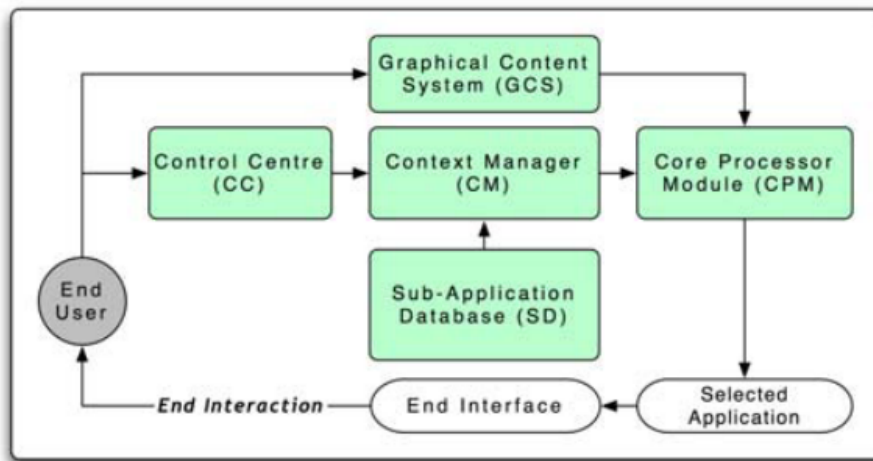


Figure 2.22: The components of the adaptive and adaptable system and the system flow [Tan et al., 2007].

Using the studies presented in the context part of this subsection as a basis the authors from that same article ([Poetzsch et al., 2020]) proceeded to present a user

adaptive visualisation taxonomy, which offers three calculation steps that help determining the layout, encoding and specifications (see Fig. 2.23):

- The layout part handles the organization of graphics with consideration for the dimension of a certain data set. Thus, in this first step, it should be decided if the data set in question has more dimensions that can be effectively displayed in a graph or if it necessary to come up with a layout to support several representations.
- The second step handles the visual encoding of data by offering different types of valid graphical representations, the differentiating between a novice user and a power user approached before is in this step as an essential factor to consider when deciding on the representations and their features. Furthermore, in this step, the author also differentiates the user's two objectives: an analysis setting or a monitoring setting. In the case of an analysis setting it is the goal to find patterns and compare values, on the other hand, on the case of a monitoring setting it is essential to get exact information hence to get a clear view of the state in every moment (see Fig. 2.21 for an example of the same data displayed two times taking in consideration the two settings).
- The third and final step considers others factors that constitute a factor and also affect the perception and understanding of charts are the color scheme used and the size of the chart itself. However the article claims the optimal chart size cannot be determined exactly, although it claims it should be large enough to allow for an accurate depiction of the visual indicators but also small enough that it does not cause unintended shifts of attention.

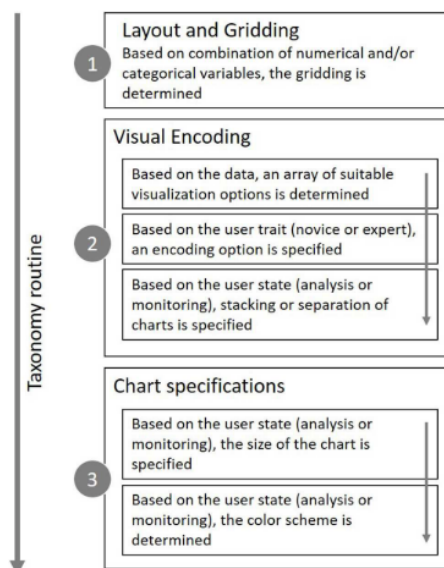


Figure 2.23: Adaptive taxonomy calculation steps to determine layout, encoding and specifications [Poetzsch et al., 2020].

2.4.3 Conclusion

On this section a deeper look was taken towards the fields of visualisation recommendation and adaptive visualisation. This research was mandatory to detect how these concepts are applied to diverse situations and to get an idea of how they work. Furthermore it was an important review to acquire information on how to contribute to this research field.

Chapter 3

CIGESCOP Overview and Solution Architecture

This chapter explains the planning of the work done. At first, the current state of the CIGESCOP software being developed is described as well as the technologies already in use and the ones that could be in use. After such description comes the approach applied to the problem created by the development of the visualisation module of the CIGESCOP system.

3.1 CIGESCOP project

The introduction chapter explained that the ASAE organization is a specialised government authority. ASAE uses several systems that were developed in different time periods in a sectorial perspective. IA.SAE was a project that aimed at helping the functioning of the ASAE organisation. From that project there was a resultant prototype that offered functionalities like route/inspector allocation and handling complaints related to health safety and public health, for example. In sequence of the IA.SAE project another one was created called CIGESCOP. It has access to pretty much all data available at ASAE, besides being connected to an ASAE older system called GESTIGAE.

CIGESCOP's web application gets many of its features based on the mentioned prototype that was constructed in 2019/2020. Several people are working on the system, so development has already begun. There are modules for route generation, text classifiers, among others.

As just stated, the system is already in the development phase, and so it already has a defined architecture (see Fig. 3.1). The architecture presented clarifies on where the data comes from. The Django¹ framework is used as a platform to gather all the data and the different back-end modules. Two different, but connected to the same back-end, client-side applications will be developed in the project's scope, a web application and a mobile application.

One of ASAE's main tasks is to supervise all kinds of commercial entities to ensure they follow all the national and European legislation regarding the services they offer.

¹More information: <https://www.djangoproject.com/>

In ASAE's context, two or more inspectors constitute a brigade, and several missions are completed daily using several brigades. Different types of services demand a specific visualisation, and so the CIGESCOP system takes in consideration the methods of exposition to the web application: the tablets the field inspectors have; the possibility of having a big screen used to show the state of the brigades in real-time; a normal desktop setup to visualise the overall metrics and abstractions in a dashboard with the possibility to generate reports based on the information available. Consequently, the visualisations of the system must be suited for all screens.

The choice of technologies and tools to use for this kind of project was of most importance. Not opting for the best available frameworks and libraries could result in limitations that could appear mid-development, and when it does happen it might be too costly to adapt the work already done to a new technology that fits the needs. So, as there is a broad list available when discussing web technologies and tools, there is also the extreme necessity to critically evaluate a technology before using it. This section is used to transmit information about technologies chosen for the development of the platform, and other technologies that could have been potentially used are discussed as well.

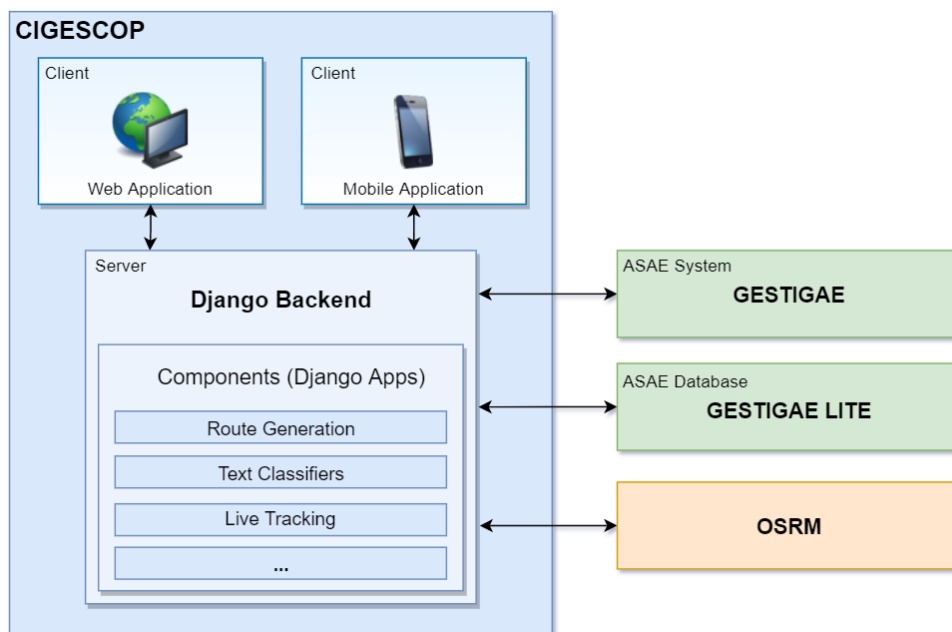


Figure 3.1: High level representation of the project architecture.

3.1.1 Current Technologies

Nowadays, projects use dozens if not hundreds of technologies and those same technologies consequently demand a more significant number of dependencies. Here, it is presented some of the most notable technologies and dependencies of the project:

- **Django.** This high-level Python framework is widely used and open source. The well-documented framework follows an original architectural pattern called

MVT (Model, View, Template), although it can be compared to an MVC (Model, View, Controller) architecture, the difference is that the Django framework does some of the work done by the controller part with the aid of templates (see Fig. 3.2). A Django project contains at least one component called a Django app but can have more as it will be the case on CIGESCOP. In this context, an app is a sub-container that in theory, can be used in other projects without drastic modifications needed. This modular division of how it works supports code reuse and eases the process of having multiple developers working on the same project. Django also offers, as expected, integrated security against Cross-site scripting, Cross-site request forgery, SQL injection, Clickjacking besides many others.

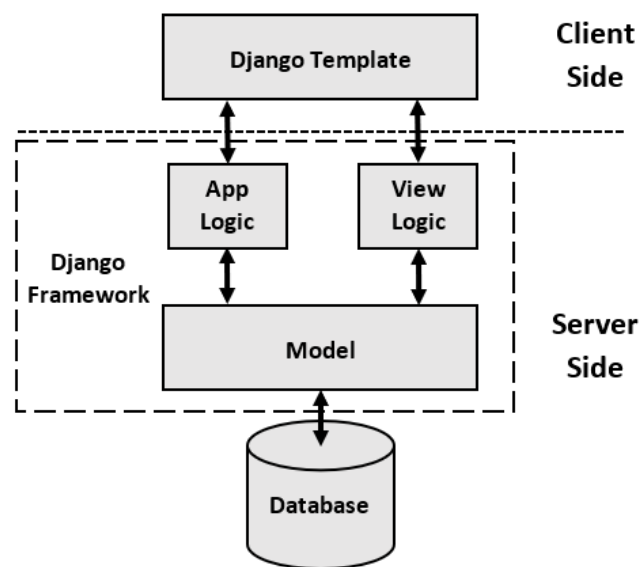


Figure 3.2: A more holistic view of Django’s architecture (<https://djangobook.com/mdj2-django-structure>).

- **Docker**². Docker is a widely supported tool that provides mechanisms that use operating system-level virtualisation to package software in an isolated environment known as a container. The container then becomes the component that is used to distribute the application. One of the purposes of this is to save time between developing and running in a production environment.
- **Bootstrap 4**³. The application claims to be the world’s most popular front-end open-source toolkit on their website. It consists of a free framework for CSS (a common and important web technology, used in combination with HTML to present web content). Bootstrap is an aggregate of code written in HTML, CSS and JavaScript with the main purpose of saving time on the developer side when it comes to writing CSS.

²Official website: <https://www.docker.com>

³More information available at: <https://getbootstrap.com>

- **AdminLTE 3**⁴. Another technology interesting to list as it highly influences the overall view. It claims to be a responsive HTML template based on Bootstrap 4. This tool's focus is to serve as a template for dashboards, and as it is built with a modular design in mind, it allows for easy customisation.
- **DataTables**⁵. This technology consists of a jQuery plug-in that facilitates searching, sorting and pagination. It is a free and open-source plug-in that is interesting to point out because of the way it benefits interaction in tables.

3.1.2 Potential Technologies

Information visualisation is an extensive field, and so many technologies can be used to develop visualisations or aid the ones that do. There is some research on this type of technologies with a particular focus on the web, but as there is a constant stream of new technologies, it is impossible to describe all of them critically.

When evaluating a tool, some of the more prominent factors are the task completion time, and the task completion correctness [Nazemi et al., 2015]. The focus is on finding web-based technologies that are freely available and allow effective interactivity to enable further exploring and filtering of data, and customisation. With this sub-section comes the description of some technologies that might seem suitable to include in the development of the visualisation framework.

The traditional options for data visualisation in applications were non-interactive plotting libraries [Wozny, 2015]. The author gives GNUPlot⁶ and matplotlib⁷ as examples and refers to JavaScript libraries as an emerging way to deliver a highly interactive data visualisation. JavaScript was known as a popular tool to validate web forms. However, now it has evolved much further, and it is in use all over the world by providing the majority of interactive applications. Also, it is compatible with all modern browsers [Roy, 2015]. The following listing describes some well-known free libraries for data visualisation in the web:

- **D3.js**⁸. An open-source JavaScript library designed to take advantage of the capabilities of modern web standards (CSS3, HTML5, SVG) that was developed and is maintained by the Stanford Visualization Group from Stanford University [Roy, 2015]. D3.js is widely used to manipulate documents based on data, providing features for interactions, animations and complex visualisations. A lot of other libraries are built on top of this one, one example of that is **nvd3.js**⁹ that aims at providing reusable charts without hurting the normal functioning of the d3.js library; another example is **rickshaw**¹⁰ which is a library for the creation of interactive graphs, that includes a lot of common chart types by default and allows the addition of more components with extensions. However, some might require the jQuery library for compatibility [Wozny, 2015].

⁴Official website: <https://adminlte.io>

⁵Source and more information: <https://datatables.net>

⁶More information: <http://www.gnuplot.info>

⁷Source and more information at <https://matplotlib.org>

⁸Official website available at <https://d3js.org>

⁹More information at <https://nvd3.org>

¹⁰Source and more information: <https://github.com/shutterstock/rickshaw>

- **Vis.js**¹¹. A dynamic library designed to handle large amounts of data by offering features to manipulate and interact with the data. The open-source library was first developed and maintained by Almende B.V. This Dutch research company works in the fields of information and communication technologies [Roy, 2015], recently in 2019 the original creators decided to abandon the project. It is now maintained solely by the community.
- **Cytoscape.js**¹². An open-source JavaScript library with a focus on interactive network visualisations. The open-source project was funded by the U.S. National Institutes of Health, National Centre for Research Resources. It can be described as a graph library for graph analysis and visualisation [Roy, 2015].
- **Sigma.js**¹³. Another JavaScript library dedicated to graph drawing. Its purpose is to ease the process of publishing networks on a web page. The tool offers features to render the graphics in either the HTML5 canvas or in WebGL and to make the visualisations interactable.
- **Google Charts**¹⁴. Described in the library website as a powerful, free and simple to use tool. It is a JavaScript library that offers by default several types of chart that support interactivity and offer customisation options.
- **Chart.js**¹⁵. A community-maintained library that can be used to generate a total of eighth different chart types that offer interactivity and can be customised. The eighth charts include bar, line, area, pie, bubble, radar, polar, and scatter charts.
- **Plotly.js**¹⁶, **Plotly.py** and **Plotly.R**. A graphing library that has a Python, an R language and a Javascript version. All libraries are free and open-source and allow for the publication of interactive graphs online. The JavaScript library is used to power the Python and R modules and its build on top of d3.js and stack.gl.
- **React**. React is an open-source web framework designed to handle the building of user interfaces. There are many libraries specially conceded for data representation in this framework. One example is **React-vis**¹⁷, a library created and supported by Uber that consists of a collection of react components that promote simplicity and flexibility. Another example is **Recharts**¹⁸, a library built on React components to handle the rendering and D3.js sub-modules as dependencies.

The visualisation of information relative to maps is also a task of information visualisation, and there is also many tools and libraries available like there is for specific tasks of infoVis. Here are some technologies that might fit the context of this work for geovisualisation:

¹¹More information available online at <https://visjs.org>

¹²Official website: <https://js.cytoscape.org>

¹³Official website available at <http://sigmajs.org>

¹⁴Source: <https://developers.google.com/chart>

¹⁵More information: <https://www.chartjs.org>

¹⁶Official website: <https://plotly.com/javascript>

¹⁷Source and more information at <https://uber.github.io/react-vis>

¹⁸More information available at <https://recharts.org>

- **CesiumJs**. This Javascript library was already approached in this document. It is an open-source solution for the visualisation of virtual globes that claims to be extremely precise as it was first built to track satellites.
- **Leaflet.js**¹⁹. Originally created by software engineer Vladimir Agafonkin and now developed by a community of contributors, Leaflet is an open-source JavaScript library specially designed to provide mobile-friendly, cross-browser interactive maps. A library designed with a focus on usability and simplicity; nevertheless, its functionalities can be extended with plugins to provide specific map interaction, include extra layers and related features.
- **Deck.gl**²⁰. Deck.gl is an open-source visualisation framework started and maintained by Uber. The framework claims to be highly customisable, to have several methods of interactive event handling that allow for the highlighting, picking and filtering of data on a map, and to allow for the integration of the major base-map providers. This technology proposes a layered approach to data visualisation, that allows for the reuse of layers that can be adapted to a specific data set. The technology also offers an extensive amount of customisable layers already developed and established. With the layered architecture and WebGL technology to render the graphics, the framework offers state of the art performance and accuracy.
- **Folium**²¹. Folium its a python library that uses Leaflet.js strengths on building maps. It eases the process of manipulating data in python.
- **Kepler.gl**²². It offers a tool for the geospatial visualisation especially designed for large data sets. This open-source tool is built on top of Deck.gl.

This enumeration of tools and libraries for both information visualisation and geo-visualisation only includes technologies suitable for production, but there are still a lot more that could be inserted in this listing. The point is to expose the amount of existing free tools for visualising data on a web page. As expected, JavaScript is by far the language that supports the most popular libraries.

3.1.3 Visualisation Module and Problems to Solve

The CIGESCOP platform is divided in several modules. As formerly mentioned there are several developers working on the project and its different modules. This dissertation and the work produced in this thesis comes as a means to solve the necessities of the visualisation module. The two main requirements are:

- Visualisations of performance indicators, adequate information visualisation graphics and means to display georeferenced information to effectively abstract ASAE's data that follow a specific theme.
- A method to configure and personalise the visualisations encountered to either cater the visualisations to a user preferences or to change the visualisations to

¹⁹Source: <https://leafletjs.com>

²⁰Source and more information at <https://deck.gl>

²¹More information at: <https://python-visualization.github.io/folium>

²²Official website: <https://kepler.gl>

include enough information to be exported to documents (for reports and similar means) as sometimes some information is hidden behind graphic interaction.

The CIGESCOP platform is a complex project. There is the visualisation module that focuses mostly on the necessity of a dashboard for the system but all its modules will in one way or the other demand visualisations. The framework later presented in this document uses the ASAE system as a case study but it is intended to work with any system with similar necessities, such as:

- Necessity to produce key performance indicators and several kinds of information and geo visualisation graphics.
- For each visualisation enclosed in the framework needs to exist an extensive and scalable set of personalisation options.
- There is the need to provide means to a developer to easily and quickly implement a visualisation in the system, providing a simple workflow when considering adding graphics to a system.
- The graphics are adapted taking in consideration the choices of personalisation of an individual user, new graphics encountered in a platform are adapted to follow previously encountered visualisations and, thus, follow a specific and unified theme.

3.2 AIVF: Architecture

These kinds of frameworks are designed as any other type of framework to facilitate programmer's work by offering a set of tools with a specific purpose and, in this case, to encourage data analysis for the end-user. Additionally, developers want versatile frameworks that can be added to new or existing projects, that can be in a wide variety of programming languages and technologies; consequently, the framework conceptualized is no different and works as an independent component that offers cross-platform/technology deployability.

3.2.1 Context

One of the first facts mentioned before is that visualization techniques are often the optimal way to achieve valid conclusions on considerably extensive sets of information. For that reason, the framework to be conceptualized and later prototyped needed to include at least the essential sorts of graphical data visualisation. Another conclusion from research presented in previous chapters is that different types of visualizations can be equally valuable to present to a user for the same type of data for two distinct reasons:

- Different insights can be achieved by having different perspectives on a set of data, and different perspectives can be achieved by having several visualisations for the same type of data.
- All end-users whose job is to analyse data have different preferences, and as shown before, one type of visualisation can be optimally intuitive to a user but

not to the other. Therefore it is of most importance for a framework to offer the option for a user to change between graphical visualisations.

A lot of the frameworks analysed for the state of the art offer users the option to switch between visualisations but put too much weight in the graphical preferences of the whole user base even though it is proven that all users react to visualisations in distinctive ways. With the stated facts in mind, the approach proposed by this thesis does not put as much weight on the graphic selection each user makes but instead on the individual personalisation done in each graphic.

As analysed in previous chapters, frameworks based on adaptation were conceptualised before. Many were even implemented as prototypes, and some even as final or close to final products. It was also stated that the current direction research is taking is to offer an adaptive system that also offers adaptable functionalities. As explained before while an adaptive system has the capacity and tools to dynamically change behaviour according to some established factors, an adaptable system provides the end-user with mechanisms that grant them the power to alter and personalize a structure. Taking into consideration the context of this thesis, a different type of framework was a necessity. With the potential of being part of a more extensive system currently in development, the developed framework needed to be easy to use and understand by the developers that wish to implement it.

The ultimate goal of the conceptualised framework is to give the end-user full power to customise the graphics he encounters when using a specific platform. This part accounts for the adaptable behaviour of the framework. When encountering new graphical representations, they get adapted according to the user's previous customisation and graphic selection decisions. The adaptability also encloses the overall user base customisation decisions but with less weight as one of the more meaningful conclusions of the research done was that users are entirely different from one another. The emphasis on the adaptable part of this visualisation framework is what differentiates this concept from previous ones.

3.2.2 Architecture

The framework in its core is divided into four components/layers. A visual representation of the flow between layers is presented in Fig. 3.3.

The first layer is known as **Data Input**. The Data Input layer receives the data in JSON format or transforms it to that same format. This base layer is the simplest one; the metadata (explained in the next subsections) of the JSON object is created and appended to the data fetched from the database. Even though only one database is represented in Fig. 3.3 as the fetching source for this layer, naturally, the information can come from any other source or multiple sources. This process is done by the developer, as he is the one that prepares the data and appends the necessary metadata. As this process is completed by the developer, he chooses if he wants to do this task in the back-end of the platform by creating an API that returns the data in JSON format and has the metadata already appended or client-side by requesting the data (converting it to JSON if not already in that format) and appending the header (metadata).

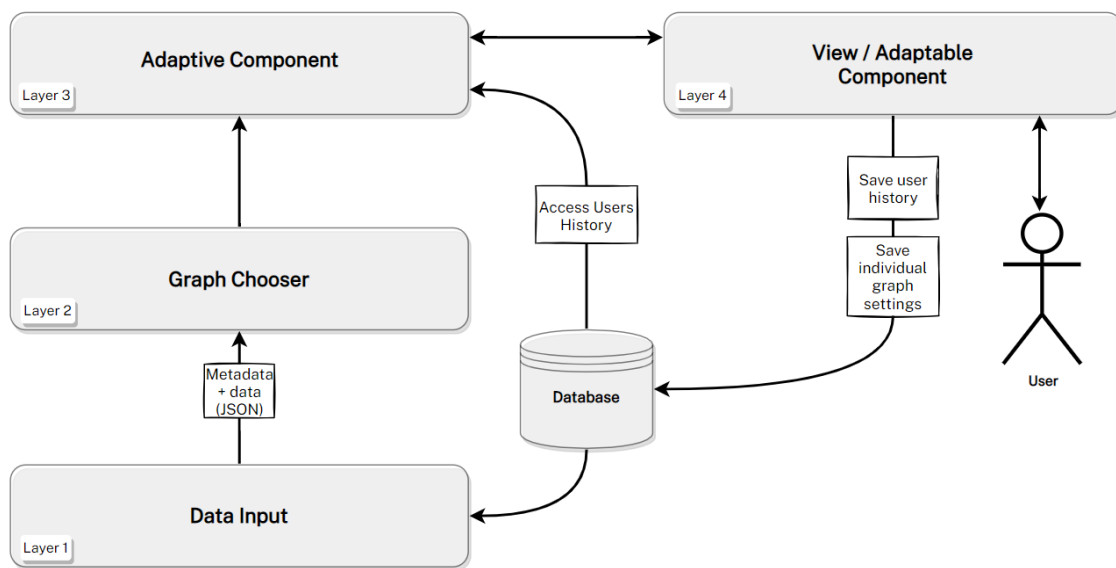


Figure 3.3: AIVF: Architecture Diagram.

The second layer (**Graph Chooser**) uses all the information from the JSON passed from the Data Input layer and selects the adequate graphic family for the data (this concept is later explored). From this layer and forward, the process takes place on the client-side of the system.

The next layer is the adaptation layer which is called **Adaptive Component**. This third layer, at first, looks if the user in question has preceding records on the system in use, and if he does not, it then looks for the preceding records of all the user base. It uses all that gathered information to adapt the selection of the graphics the user is going to encounter as well as the personalisation options for each one of those graphics. Although represented in Fig. 3.3 as the user information coming from the same place as the source of the data, normally, this information does not need to come from the same place.

The final layer is the stage that handles the visualisation part and offers adaptable components for the user (the **View/Adaptable Component**). Besides handling the visual representations, this layer also handles the process of personalisation configuration by the user. Another task this layer has is to communicate with a database to store individual graphic information as well as a history of preferences to later use in the Adaptive Component. This layer belongs to the front-end of a system, and so the end-user of a specific platform sees this layer exclusively. Here he sees the respective visualisations, the option to change the selection of possible visualisations and also has access to the personalisation options.

3.2.3 Functionalities

- The framework is suitable for displaying visualisations of multiple data sets; it is independent of the technologies used in the rest of a system, either in the back-end or front-end.

- An object is used as input at the data layer, and an instance is created. The graph chooser allocates the data to the respective family; if a new modified file is sent through the same instance in run-time, the graphic in the current selection updates the view according to the new data. Thus the visualisations of the framework can be used to represented changes in data in real-time.
- For each input to the data layer, a graphic is created. Besides the data sets, a header with metadata needs to be appended.
- The framework needs a communication with a database to save each graphic set of preferences and a user history of changes.
- Each graphic on a platform using the framework is independent of each other; although they share the same database table to keep a history of alterations, they do not directly connect with one another.
- When a graphic is loaded on a page, first, it is checked if there is a table with the respective user id, if there is not an entry in the database for the graphic in question, it is checked if there are enough history records of the overall user base to adapt the graphics for the new user, if there is not, the graphic is displayed on its default state. The default state considers the recommendations obtained for each visualisation during the state of the art research. If the user base is enough to obtain a valuable adaptation, the graphics are adapted for the new user, considering that information.
- Once a user has a history of his selection and personalisation options, the framework no longer requires assessing other user's history to adapt his visualizations.
- Once a graphic in a family is adapted, its particular configurations are saved individually, and it is no longer adaptable; this is done so that the user does not see its graphics configurations changed when refreshing a page or coming back to a system. This is because it is the user decision if he wants a graphic that is utterly different from the pattern the other visualizations follow. The information stated only applies to a specific graphic in a graphic family, all the other graphics in the family that the user might switch to are adapted if it is the first time they get selected per individual visualization.

A necessity to include a metadata appended to the data to be displayed in the framework was mentioned before. This is because visualisations are divided into families of graphics, which is yet another concept explored in this section. In conclusion the input submitted to the framework has two separate fields, one called header which contains the metadata and another called data which contains as the name suggest, the data set.

This concept of families of graphics was introduced to classify each type of data, and it serves as a component in the framework that is scalable. When seeding data to begin the flow in the framework the programmer needs to specify which family his data complies to. A family of graphics is no more than a set of specific graphics that serve a certain purpose. When a family is chosen and the graphic is produced the end user can change the selected visualisation between graphics that are considered to belong to that specific family.

More information needs to be included in the metadata like the variables to take in consideration on the data and in some cases some extra information that is optional but that can be useful. This concept is exemplified later in the prototype implementation section 4.5 including uses of the metadata fields that are optional.

3.3 Conclusion

In this chapter, several points that led to the conceptualisation of the AIVF framework were approached. As stated, frameworks already exist that serve well their purpose. After research on the matter was conducted and having this thesis included in the IA.SAE project the concept of a framework that catered mainly for the user as an individual that wants complete control over his data analysis system was introduced as it fitted the needs of the visualisation module of the project and also offered a framework with a different perspective.

After offering an overview of the idea of the framework, in the following chapter, the implementation of said visualisation framework is discussed. There, more practical examples of the functioning can be found as well as figures of graphics using the implemented framework in the CIGESCOP platform.

Chapter 4

AIVF: Implementation

This section of the document approaches the implementation cycle of said prototype. First, it introduces the graphics available in the prototype developed and after it explains the concept of graphic families and which ones are available in the prototype. Just after, most of the personalisations options also available in the prototype are introduced and described. Then after a short subsection on interaction the technologies used on the development of the prototype are talked about.

After introducing a proposed architecture and conceptualising the framework, it was essential to develop a working prototype with the intent to arrange tests with real users. As said in the previous chapters, this thesis is enclosed in the CIGESCOP project. The prototype developed was later added to the CIGESCOP platform for testing purposes. In this chapter, some images will appear to demonstrate in visual form how some functionalities of the conceptualized AIVF work; on those graphics, the title, legends, and axis naming are hidden not to expose what might be sensible and private information of the ASAE organization.

4.1 Technologies Used

As approached in the background knowledge section, visualisation on the web in the current times cannot be static as the user expects more. Thus, to keep in line with user expectancy, all the graphics involved in the framework offer interaction primarily via hovering in a visualisation element and getting its values.

This interaction process is eased as almost all graphic technologies in current times offer this functionality. Still is a vital aspect to point out. In the default state of most graphics, to get the exact values of the elements, the user needs to use interaction. Nevertheless, as explained in the last subsection, the visualisations can be configured such that the exact values are displayed in the graphics themselves.

Besides the typical hovering properties the maps included also offer the option to navigate with the traditional 2D map panning functionalities but also offer the option to explore them in 3D. Another use of interaction in the personalisation options is through the use of tool-tips to help clarify what each option does to the user.

The prototype could have been implemented in a lot of different technologies and still offer the same properties, that is because there is an extensive set of valid technologies to develop this kind of project. This section approaches the technologies used.

4.1.1 Development of the Prototype

- **React**¹. React is an open-source web framework designed to handle the building of user interfaces. There are many libraries specially conceded for data representation in this framework. This framework was chosen as it saves a lot of precious time in JavaScript development.
- **Material-UI**². A set of react components that eases the process of the design aspects of a system. In all aspects a great tool to fasten the process of developing prototypes.

4.1.2 Information Visualization Techniques

- **Recharts**³. A library built on React components to handle the rendering and D3.js sub-modules as dependencies. It was the most used library in terms of graphical visualizations in this prototype.
- **Google Charts**⁴. Described in the library website as a powerful, free and simple to use tool. It is a JavaScript library that offers by default several types of chart that support interactivity and offer customisation options. Used because it offers a great set of information visualization techniques.
- **React-d3-speedometer**⁵. This very specific library built on top of d3 was of great use to offer a gauge graphic option in the prototype.
- **Patternfly Charts**. Although a very promising graphic library in the prototype it was used only to offer the option of a bullet chart.
- **Deck.gl**⁶. Deck.gl is an open-source visualisation framework started and maintained by Uber. The framework is highly customisable, and has several methods of interactive event handling that allow for the highlighting, picking and filtering of data on a map, and also allow for the integration major base-map providers. This technology proposes a layered approach to data visualisation, this allows for the reuse of layers that can be adapted to a specific data set, the technology also offers an extensive amount of customisable layers already developed and established. With the layered architecture and WebGL technology to render the graphics, the framework offers state of the art performance and accuracy. This technology was used in all the maps currently available in the prototype because of its fast loading times and because it allows the user to navigate like a traditional 2D map but also also offers 3D functionalities.

¹Official website: <https://reactjs.org>

²More on: <https://material-ui.com>

³More information available at <https://recharts.org>

⁴Source: <https://developers.google.com/chart>

⁵Official website: <https://www.npmjs.com/package/react-d3-speedometer>

⁶Source and more information at <https://deck.gl>

4.2 Chart Types Used

To introduce the concept of families of graphics that serve each a particular purpose, and since they share visualisations among them, it is appropriate first to introduce all types of graphics enclosed in the framework prototype so far.

4.2.1 Information visualisation graphics

- **Bar chart.** This classical type of chart can work as the well-known simple bar chart, but depending on the amount of data series it can be used as a stacked or grouped bar chart. As a stacked and grouped chart can display the same amount of dimensions, one of the user personalisation options in the case of a bar chart with more than one dimension is to either display it as stacked or grouped bars, there is also the option to display a layout of simple bar charts each one containing one dimension (see Fig. 4.1 for an example of this behaviour).
- **Side-by-side bar chart.** An additional version of a bar chart that in the majority of situations facilitates the comparison of two numerical dimensions (example at Fig. 4.2).
- **Line chart.** A regular type of chart already explored and explained in section 2.1.2 subsection. When multiple data dimensions exist it can be separated into individual graphics.
- **Area chart.** Example of this kind of chart contained in the framework in Fig. 4.4. Follows the same pattern as the bar chart. When multiple data dimensions exist it can be used in a stacked or overlapped configuration, or divided into individual graphics.
- **Pie chart.** Example of this chart and some of its configurations in Fig. 4.8. This type of chart as explained in the techniques of information visualisation its not advisable as it does not work well with our visual system. With that in consideration it is still a graph widely known and desired by a lot of users.
- **Scatter chart.** Example in Fig. 4.3 a). Extremely useful visualisation to discover relationships between two variables.
- **Sankey diagram.** Example available in Fig. 4.3 b). Simplified version to fit the family it is inserted. It could later be adapted to fit a hierarchical graph family.
- **Bullet chart.** Representation of this KPI graphic in Fig. A solid visualisation for the display of key performance indicators. Example in Fig. 4.3 c).
- **Gauge chart.** Example of this classic performance indicator graphic in Fig. 4.3 d). Because of its nature it suffers from the same problem as the pie chart but it is a popular visualisation as well in this case for the representation of KPIs and as such it had to be included in the prototype.

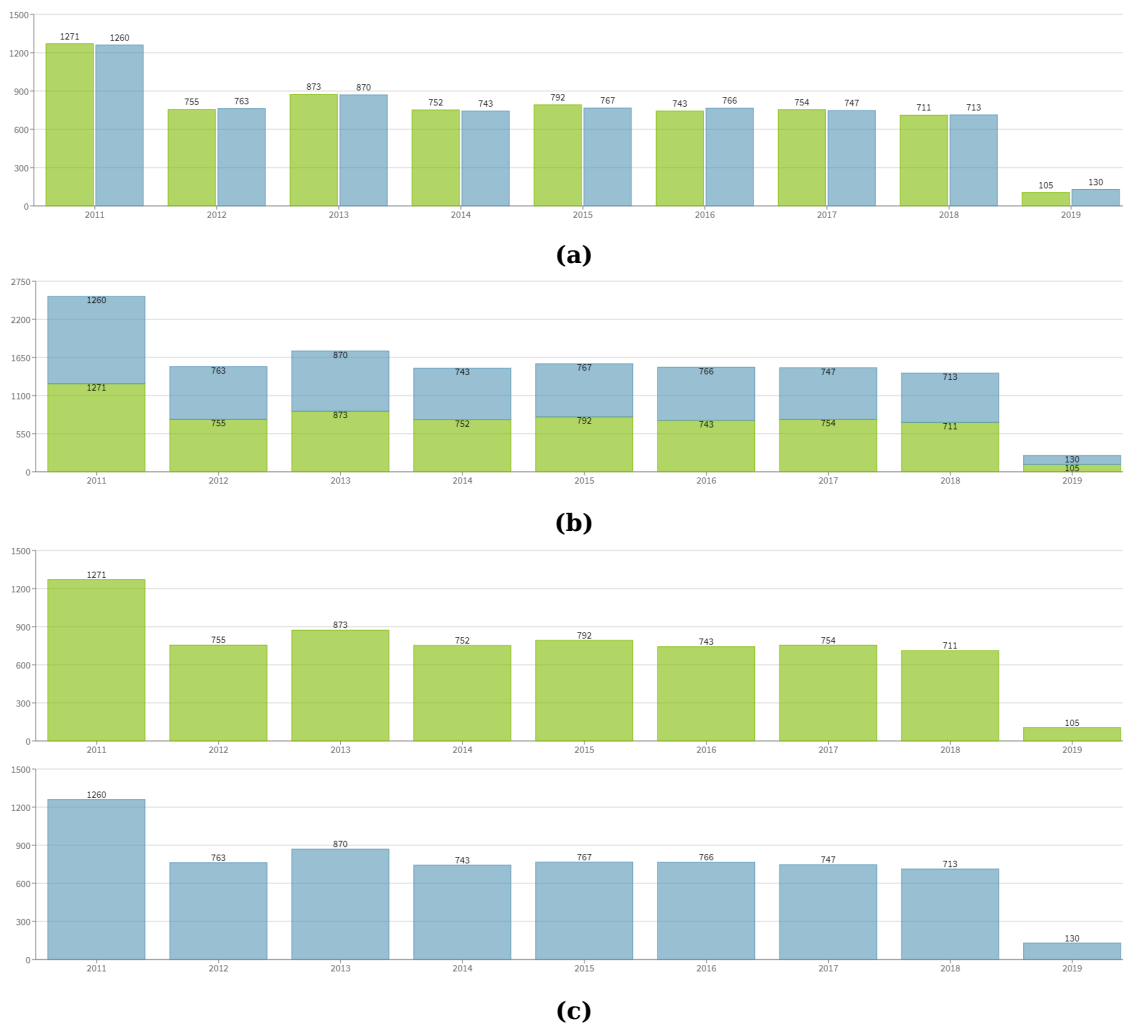


Figure 4.1: AIVF: Charts. Bar chart: a) Example of a grouped bar chart. b) Example of a stacked bar chart. c) Example with the dimensions separated into simple bar charts.

4.2.2 Geovisualization graphics

- **Heat map.** A classic density map (see Fig. 4.9 a) for example).
- **Hexagon map.** A density map that offers a new perspective in comparison with the heat map (example in Fig. 4.9 b)).
- **Column map.** A map used to show values by single coordinates (example at Fig. 4.13)
- **Bubble map.** Alternative to the column map (example at Fig. 4.14).
- **Icon map.** A simple map that displays custom icons.
- **Path map.** A map that connects consecutive coordinates with a line.

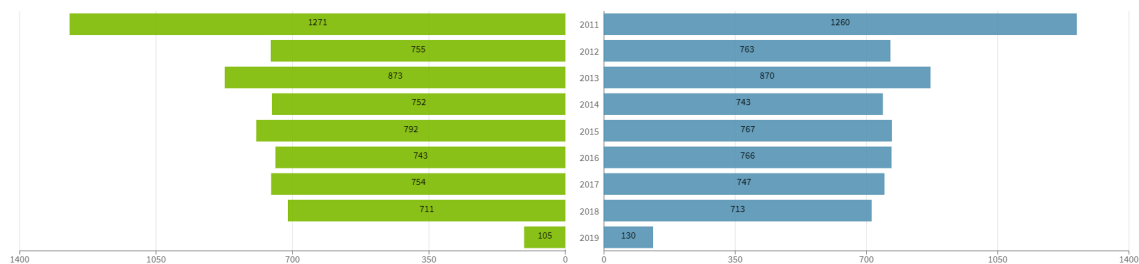


Figure 4.2: AIVF: Charts. Side by side bar chart.

4.3 Families of Graphics

As stated there is a division of graphics in the platform in different "families", each one with its concrete purpose. When seeding data to begin the flow in the framework the programmer needs to specify which family his data complies to. For adaptation purposes the personalisation options chosen in a specific family by a user have a bigger weight than the ones that do not belong to the same family. So far, the existing families are:

- **One numerical.** This family of graphics is used to represent numerical data that is simple and represents only one numerical value per description. It offers the options to change between a bar chart, a pie chart, a line chart, an area chart or a simplified version of a Sankey diagram.
- **Two numerical.** This specific family of graphics is used when the objective is to compare two numerical values. The graphic selection options are the side-by-side bar graph, scatter plot, line and area graph, and a normal bar chart.
- **Time series.** As the name suggests, this family is used to explore information that is displayed over time from one to n numerical values. It offers line, area and bar charts.
- **Performance.** This set of graphs is used to display relevant and straight to the point KPIs. It has two options, a bullet plot or a gauge representation.
- **Geo.** The first family on geographic information represents values at specific coordinates. Options available are a column and a bubble map.
- **Geo dens.** This type of graphic family takes in a great deal of coordinates and resumes them into density graphs. The graphics it offers are a heat map and a hexagonal map (see Fig. 4.9 for an example).
- **Geo icon.** Although the name is straightforward, it is still appropriate to explain the purpose of this type of map. It displays icons at specific locations, and to complement it, mouse hover tool-tips can be added to each particular icon. The typical location icon is the default icon in use, but the developer can add another one of his likings.
- **Geo path.** At last, comes the type of map to produce paths and routes. This type of map can be used together with the icons map to display a path and with the icons display several stops at a particular route, or to display a start and endpoint, for example.

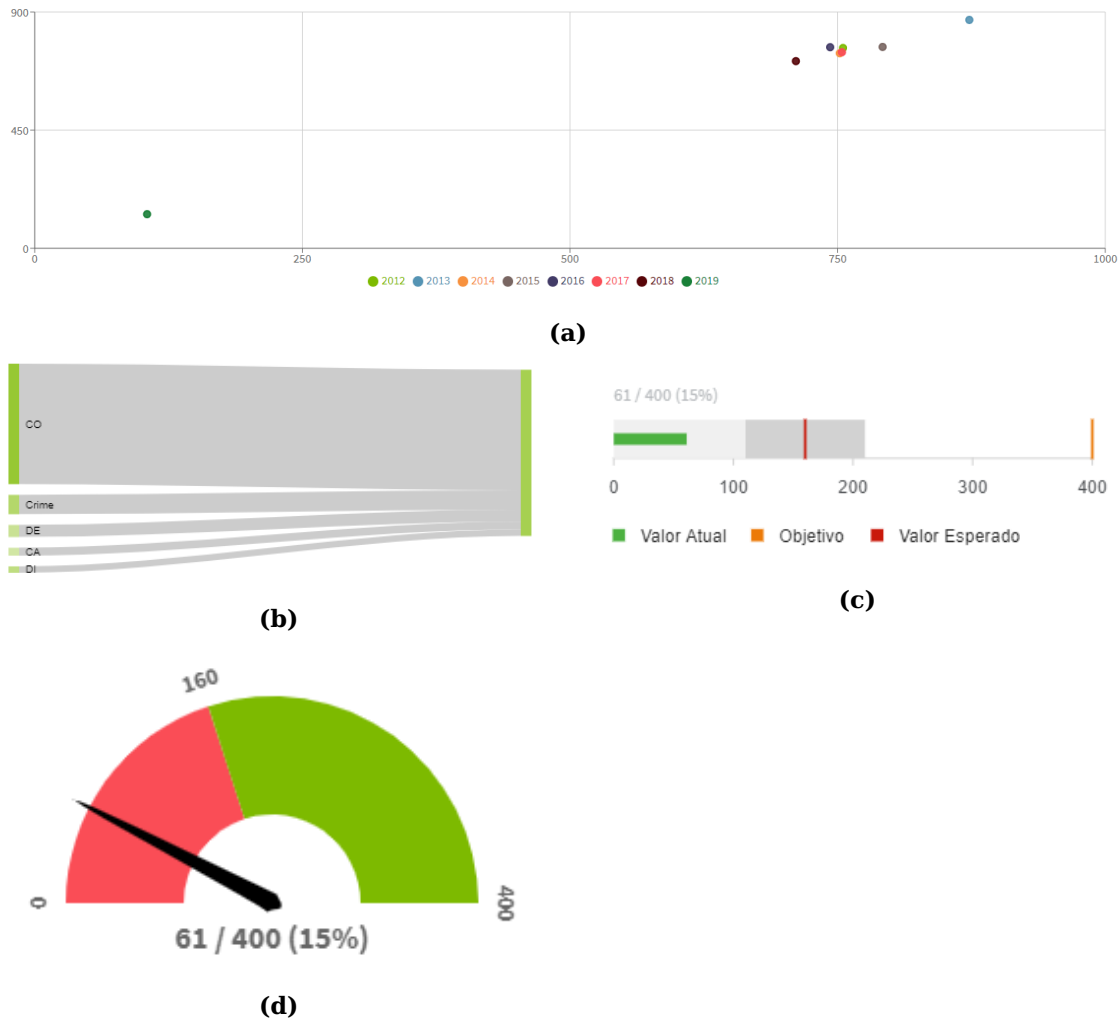


Figure 4.3: AIVF: Charts. **a)** Scatter Plot. **b)** Sankey Diagram. **c)** Bullet graph. **d)** Gauge chart

The reason for this division was to give full power to the developer on the visualisation his data produces. This way, when seeding data the developer knows what visualisations the end user will have as options. The division of the families in this prototype follows the most common norms for each type of data. Also, these families of graphics were the first ones to be implemented in the framework as they suited the needs of ASAE's data for the most part. Nonetheless, as pointed out, more families can be added by using existing graphics, granting them a different objective and combining them with newly introduced types of information visualizations or creating a new family with only freshly instituted graphic visualizations.

4.4 Personalisation of Graphics

Graphics share characteristics, even more so if they belong to the same category. Even wholly different graphics like a bar chart and a bubble map share some traits.

That is why the adaptation bases itself on the user personalisation of the visualizations. In environments that depend highly on data analysis, new graphics are proposed and created at a fast rate. Having the possibility to personalize is something the regular user craves, but a user does not want to go for the same process for every graphic that comes up.

The one characteristic that all graphs have in common is the use of colour. Although colours serve different purposes in visualizations like stated in the subsection 2.1.6 (for example, in Fig. 4.9 the colour works as a sequential colour scheme and in Fig. 4.1 it works as a qualitative colour scheme) in most cases the user wants all the graphics available in, for example, a dashboard to share the same colour scheme. It is also imperative to give the user the option to configure the colour scheme by a specific graphic, as stated in the section on colour vision deficiency.

Much like colour, there are many characteristics that graphs share between themselves. The framework was conceptualised such that new personalisation options could be easily added over time, thus enriching its functionality. This subsection addresses some of the personalisation options available in the developed prototype of AIVF. Those options available approach factors that are crucial for good graph readability, like labels and the use of a grid, for example, and others that cater for aesthetic decisions or aid in exploring data.

4.4.1 Personalisation Options

- One characteristic that is also very common in a big part of the visualisations available was named **y tick**. It simply refers to the number of values of the y axis. It is helpful for the user to get a less spaced interval of numbers on said axis to achieve more precise comparisons. This option is displayed as a slider of integers. The number displayed in the slider is the number of ticks available in the y axis, although is it also suitable to have in mind that the height of the graphic directly affects the maximum number of values that can fit in the referred axis. Another convenient use of this option is to set it to the value zero, which means that the y axis disappears. This option is available in most graphics that use the traditional 2D Cartesian coordinate system like the bar, line, area and scatter plots (see Fig. 4.4 for an example).
- An option that is more indicated for specific cases was called **x interval**. As the name infers, it adjusts the interval of values of the x-axis. This configuration can be tweaked by the user in a slider, and its default value is 0. For each increment of this slider, the values shown on the x-axis are divided by half. This option can reveal itself to be extremely useful for the user when the values on the x-axis follow an intuitive order (for example, in the *time_series* family where usually in the x-axis the values are incremental dates), intending to remove visual clutter. In congruence with the last discussed personalisation configuration (*y tick*) this option is offered in the bar, line, area, and scatter charts (see Fig. 4.4 for an example on the behaviour of this configuration).
- It is also possible for the user to decide if he wants a **grid** in the background of the graphic or not. Furthermore, it is possible to tweak the grid's opacity, only

show a horizontal or vertical line (or both), and display the grid as a continuous line or stroke line.

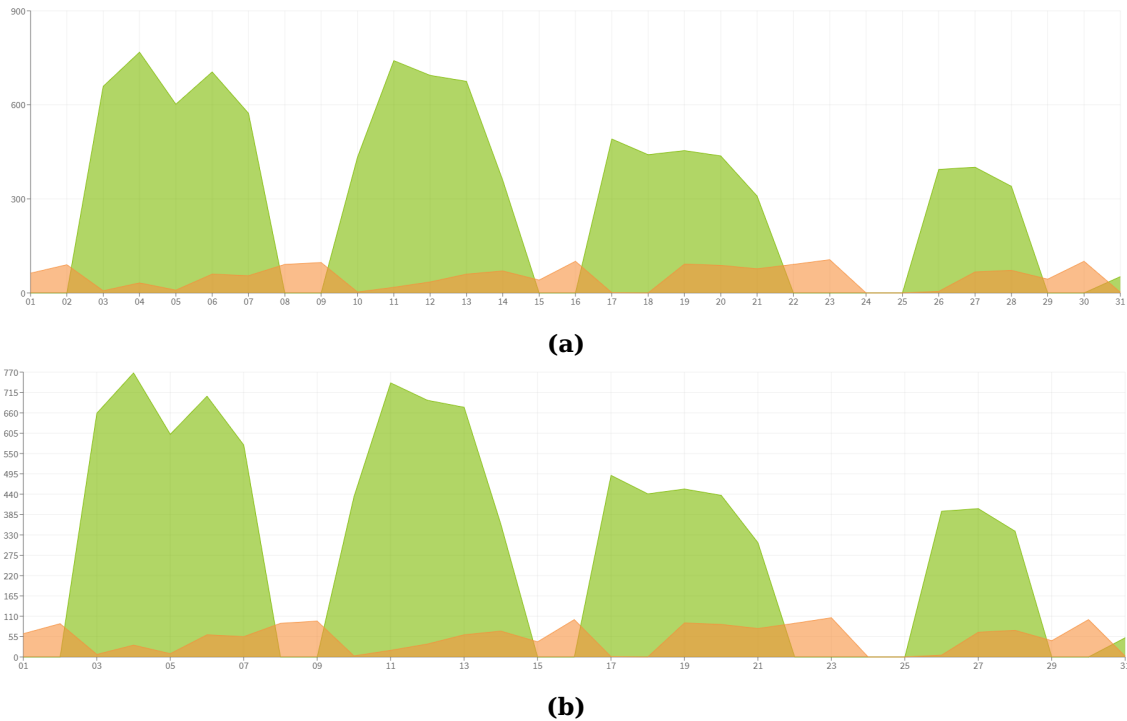


Figure 4.4: AIVF: Personalisation. **y tick** and **x interval**. **a)** The **y tick** value is set to 4 and the **x interval** is set to 0. **b)** The **y tick** value is set to 15 and the **x interval** value is set to 1. As in the example the x-axis represents days and such the use of the **x interval** configuration can be used without losing information for the sake of reducing visual clustering

- This configuration is meant to help the user improve the readability of his information visualisations. It is called **simplify**. It is a Boolean option. Hence as expected, it can be either true or false. This option is extremely valuable to cater for large values in the visualisations. When positioned in the state of true this option adapts how the numerical values on the axes are represented. For example, if a value is 100 000, it represents it as 100k; if that value has one more 0 as in 1 000 000, it is represented as 1m and so forth (see Fig. 4.5 a)).
- Another personalisation option available is called **label list**. It is of great use when the user wants to visualize the exact values presented in a graph without making use of interaction. It is another option that is a Boolean. With this choice, the user can, for example, show the values of a bar chart on top of every single bar. Furthermore, the user can manually adjust the location of those values if he desires via having the option to relocate those same values up and down and rotating them. Besides the last option explained that improves the readability of large numbers on the axes, it is possible to apply the same process to these values (the user chooses if he wants both values simplified, one or the other or none, example of this behaviour in Fig. 4.5 b)). This option is exceptionally

valuable when configuring graphics to be exported as PDF or some identical file format as those solutions obviously do not offer interaction. Another interesting use is as an alternative to display the graphic values when the user decides to use the y-axis tick option previously discussed at set it at 0, thus eliminating the y axis.

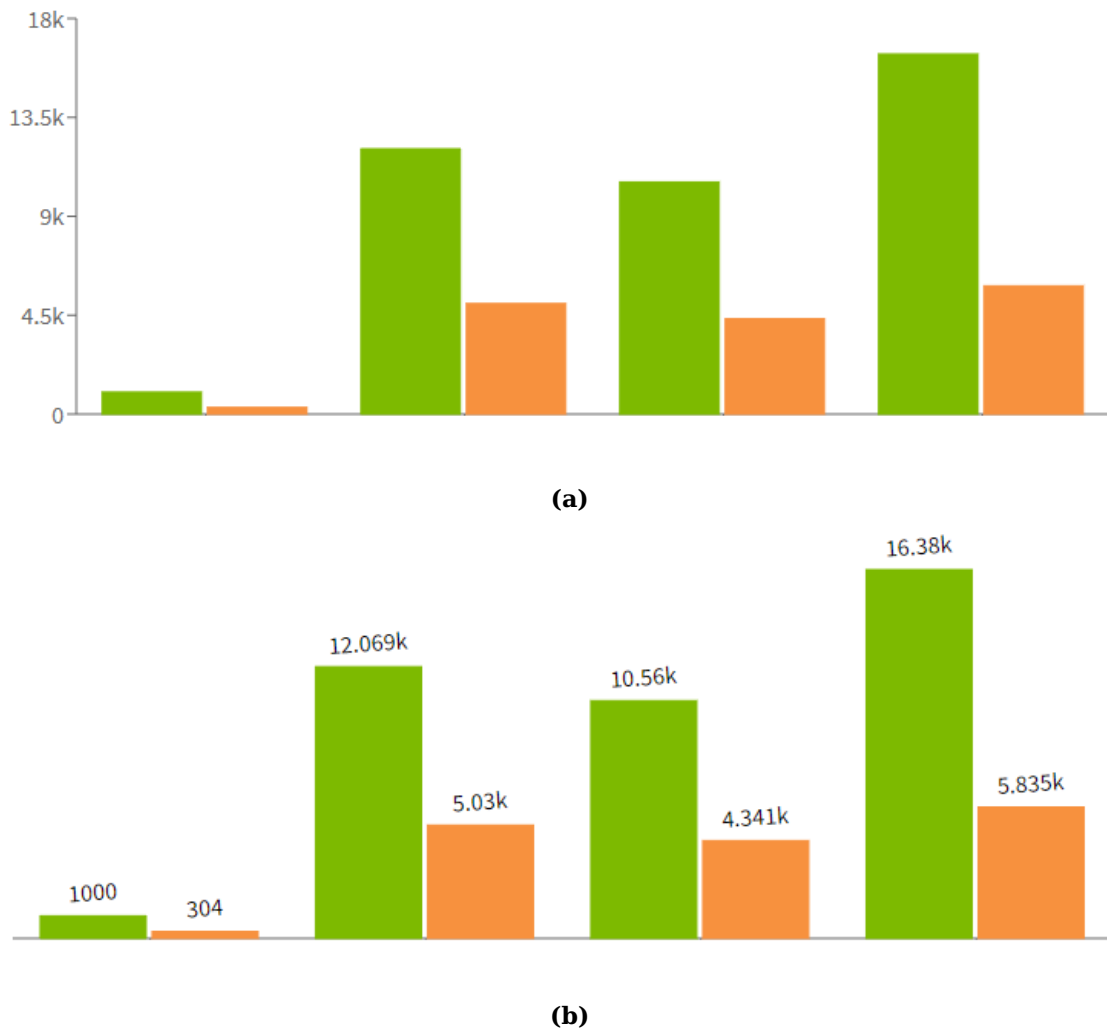


Figure 4.5: AIVF: Personalisation. Simplification of values and showing values directly on the visualization. **a)** Example of the values on the y-axis with the options **simplify** set as true. **b)** Example of the configuration **y tick** set to zero, and the option **label list** set to true, with a slightly panning up of the values and a almost unnoticeable rotation. The simplification of the label list values is also set to true.

- One other configuration is the option to show the **legend** of the graphic or not and to choose its position. It could be on top of the graphic, the bottom or even the middle (although the middle position is not recommended as it more often than not would create visual clustering). The legend can also be aligned to the right, left or centre independently from its position.

- In consideration for the graph dimensions the **height** is a value that is available as a personalisation option. It is set through the use of a slider. The dimensions of the width are left for the behaviour of the container a specific graphic is enclosed on.
- One of the more important options that can influence a lot the usability of the visualizations available is the possibility to change **colour**. A colour scheme is provided to the framework when it is used in a platform. It is a task of the developer to provide tools to the user base to change the colour scheme of all graphics. The framework handles the alteration of colour per graphic. Per each color in use it offers the option to open a colour picker and choose the desired hue. The user select his desired colour and the picker offers the default colour scheme in the quick selection part (see Fig. 4.6, the default colour scheme is at the bottom).

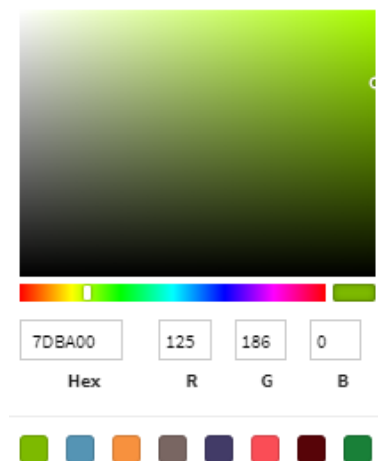


Figure 4.6: AIVF: Personalisation. Colour picker. The default provided colour scheme is at the bottom for a quick selection.

- One other option related to colour is the **opacity**. It was decided to offer the opacity as a separate slider from the colour picker to facilitate the user job and avoid problems like changing the opacity colour by colour when needed. Sometimes under-looked, the option to configure the amount of opacity is advantageous to avoid visual clustering when overlapping information (an example is the area chart graph in Fig. 4.4).
- When applicable and it exists more than one set of information for comparison, there are options to show all the information on the same graphic if the Boolean option **grouped** is set to true (for example, the area chart displayed in Fig 4.4), if it is not then the data is presented in individual graphics. Furthermore, in bar and area charts is possible to present the grouped information in a **stacked** method (as shown in Fig. 4.1 b), the same concept applies for the area chart).
- In cases that information is connected by lines such as the line and area graphics, it is possible to change the **interpolation** via the selection menu. This

option, in most cases, is merely aesthetic but, in some cases, might reveal itself helpful in the data exploration process. The currently available interpolation methods are linear, monotone, step and basis (example of this behaviour available at Fig. 4.7).

- There exist some configurations that are family-specific, like in the case of the `one_numerical` for example, it is possible to **order** the data in the graphic by descending or ascending order and choose the y axis **scale** to be of regular or logarithmic scale. Alternatively, some are graphic specific; for example, it is possible to adjust the size of the pie plot, not only the **outer radius** but also the **inner** one (example in Fig. 4.8); in line charts, it is possible to adjust the size of the line; in scatter plots it is possible to change the size of the dots.
- In performance graphics and in the pie plot, it is possible to display the values in **percentage** rather than its raw state. Furthermore, it is possible to choose the number of decimals places (behaviour explanation in Fig. 4.8).

The personalisation options introduced just now are primarily applicable for information visualization techniques. It is also relevant to introduce some applicable for geovisualisation techniques, even though graphics from both areas share some traits (legend, height and colour, for example) and are taken into account in the adaptation component later explained.

- Some minor personalisation options exist to cater for usability, such as **maximum zoom** that restricts the maximum zoom applicable in a map to a value the user desires and a Boolean called **restrict** the map to the where the data is to avoid having the user waste time navigating through non-relevant zones.
- An interesting option contained in the framework is called **borders** and shows relevant borders for the organisation the CIGESCOP project is based on. It is also possible to choose the colour inside the borders and adjust the opacity. These borders could be adapted to serve any organisation's necessities by just changing the polygon coordinates to relevant ones as in the prototype this values are hard-coded. In this case, there is the option to show Portugal district borders and two other relevant options (as the one exemplified in Fig. 4.9 a)). In the corresponding figure, there is an example in b) where the borders are turned off, and a **background map** was added, which is another option where several maps are available to be picked (a normal map, a relief map and a terrain map like the one in the indicated figure).
- Just like the personalisation options for information visualisation techniques, there are not such relevant configurations that can be done, such as changing the **icon size** when icons are available and changing the **path thickness** when paths are encountered.
- It is possible to configure how the density works in a heat map. For example, a user can change the **intensity** via slider. This value is multiplied by the total weight in a location to obtain the final weight. As such, if the intensity is below one, it biases the output colour to go closer to the end of the spectrum, while the opposite happens if it is a value bigger than one. One other configuration

available is setting the **threshold**, this value goes from 0 to 1, and the bigger it is, the fewer relevance points with low weight are displayed. Finally, it is possible to adjust the radius. The radius means the length of zones used to calculate the number of coordinates inside it and then calculating the amount of density. In the density charts, just like other charts, it is possible to change the colour scheme.

- In the same kind of heat map comes the hexagonal map. In its options, it also has a **radius** value to set; this intuitively means the radius of each hexagonal displayed on the map. The higher the density, the bigger the hexagonal length, the map can be used as a 3D map, and interaction is available to hover on every single hexagonal and get its value, but to ease the process for the user, it is possible to set a **scale** that then is multiplied with the actual lengths to calculate its height.

4.4.2 Interface Layout

It is also important to present how these personalisation options are shown to the end-user in this specific implementation of the framework. To open the personalisation menu of the framework, the developer needs to assign a button or an equivalent component. The menu, when opened, appears as a sidebar that shows in front of the rest of the system; that sidebar can also be docked to the left or right side of the user screen depending on his preferences. Exemplification of this behaviour can be seen in Fig. 4.10 and Fig. 4.11.

In the examples provided, all the text is in Portuguese as this version of the framework was implemented to be included in the CIGESCOP platform, which is in development to cater to ASAE needs (a Portuguese organisation). The first thing that appears in the sidebar is always the graph selection; it is divided into sections that follow the same pattern independently of the graphic in the current selection. One other interesting aspect is the colour theme. The colour theme of the options (as seen in the suggested figures are based on a shade of green that represents the primary colour of the ASAE organisation), this is independent of the default colour scheme provided to be the default scheme for the visualisations but can also be altered by the developer to fit his needs by changing the values of the interface colours in the prototype's code.

A useful usability functionality not stated so far is the use of tool-tips for each option presented in the menu. The tool-tips appear after a user hovering an option for more than 2 seconds and disappear 0.5 seconds after the user releases selection of that options. These times can be easily changed in the implemented prototype to other values. An example of this behaviour is available in Fig. 4.12.

4.5 Input

As mentioned, the framework accepts data in JSON format in its lower layer. Besides containing the data, a header needs to be attached to the JSON containing information on how to display the information. Let us look, for example, at how the JSON file should be constructed to produce the graphics in Fig. 4.13 and Fig. 4.14:

```

1 {
2   "header":{
3     "type":"geo",
4     "id":["Name"],
5     "value":["Value1","Value2"],
6     "coordinates":["coords"],
7   },
8   "data": [
9     {
10      "Name": "Name1",
11      "coords": [41.160304, -8.602478],
12      "Value1": 9097,
13    },
14    {
15      "Name": "Name2",
16      "coords": [41.160304, -8.502478],
17      "Value2": 2643,
18    },
19    {
20      "Name": "Name3",
21      "coords": [39.2419779, -8.6821562],
22      "Value1": 3597,
23    },
24    {
25      "Name": "Name4",
26      "coords": [39.2419779, -8.5821562],
27      "Value2": 451,
28    },
29    ...
30  ]
31 }

```

Listing 4.1: Template of the structure of the input JSON

Looking at the example, one can identify two main components in the JSON, the header and the data. In the header, the developer defines what values to present, having the option to send any number of values he wants (leaving the decision of handling visual clustering to the programmer). As seen in 4.13 the green columns account for the *Value1* of the example and the blue ones for the *Value2*. Per order of the value array, the framework uses the order of the colour scheme array. The *id* setting in the header is optional as it is then used in the interaction to display alongside the value of a column when the graph reader hovers it. The *coordinates* field in the header is there to indicate where the coordinates are on the data; it is also optional as if it is not indicated, the framework looks for them by the keyword *coordinates* in the data.

After an example to get an idea on how it works it is important to discuss how it works in a more structured manner. As seen in the example the metadata is a small

component of the JSON and it is labelled as header. The same logic maintains for all families. The simple structure is the following:

- **type.** This is where the key word that represents the family is inserted. On the prototype, the keywords for the families available are *one_numerical*, *two_numerical*, *time_series*, *performance*, *geo*, *geo_dens*, *geo_icon* and *geo_path*.
- **id.** In the example, this entry is optional. It classifies each entry of data. This is used in the geo maps just to describe an item when it is hovered, but when using 2D visualisations, this *id* becomes necessary to classify the data (usually shown on the x-axis).
- **value.** This entry is used to specify the values on the data. This way, a developer does not need to filter his data and can just specify the necessary values. As shown in the example, the *value* entry receives an array with the names of the fields to display in the graphic. Naturally, the length of the array should take into consideration the family in use; if it is the *one_numerical* family, only one value is allowed; in the *two_numerical* family, an array of two elements is necessary; in the *time_series* family the array can have a length of *n* and so on. This field name is never shown in the visualisation, it is only used to identify what the developer wants to show in the graphic.
- **coordinates.** This value is used in map visualisations to point to the field containing the coordinates. If not included it is assumed the coordinates field is called "coordinates".
- **preferences.** This extra entry is optional in all uses of the framework. It receives an object containing the personalisation settings of a graph if the developer wishes to change its default view.

4.6 Adaptiveness

When it comes to the developed framework there is a previously defined architecture. Before going into more depth on the adaptiveness is it also relevant to discuss the main features:

- Each graphic has a relatively extensive set of personalisation options, from merely aesthetic options to options intended to help the user best understand the data they are looking into.
- There is one main algorithm, which is administered to new graphics on a platform when a user has seen and personalised at least one graphic in that same system. The algorithm considers the latest *n* set of personalisation options a user has chosen for each family of graphics.
- One purpose is to facilitate the process of creating graphics for the developer, releasing much valuable time for the programmer to focus on the quality of the queries to the database.
- Another purpose is to save the user time. Since new graphics that appear in a platform get adapted in consideration for previous user decisions, it saves the user valuable time on future occasions.

- The framework is divided into several families of graphics (explained later in this section).
- A prototype of the framework was developed such that it can be expanded and scaled. More personalisation options could be added, more graphics could be implemented to complement the existing family distribution, or even new families of graphics could be introduced.
- The framework was implemented such that it can be used in any system. It was designed to accept data in JSON format as it seems to be the best and most versatile data format available.
- Since it accepts JSON as an input format, it can be used and deployed over any other programming language.

4.6.1 Database Tables

To include this framework in a system/platform there is the need for a connection to a database to take full advantage of its full functionalities. As such, the creation of two database tables is required. As each graphic that a user encounters can be personalized, there is the necessity for methods to save that information. The option picked was to create a table for that purpose. Another significant factor was to save a history on the options, and for that goal, another table is required.

4.6.1.1 Table 1 - Individual Configuration

This table is used to keep track of every graphic enclosed in a platform (see Table 1 in Fig. 4.15). For each graphic available, an entry is created per user. In one table is possible to save all the information regarding all the visualisations on the family that graphic is inserted on. The organisation of the variables of the table is the following:

- **id_user.** As the name suggests, this variable contains the user's identification; it could be the traditional integer identifier or a specific string as long as it is unique per platform user. In conclusion it is a foreign key to the users table of a system.
- **page_name.** It keeps an identifier of the page it belongs to.
- **page_index.** It represents the index of the graphic on a specific page (identified by the variable `page_index`). Together with the two last variables (`id_user`, `page_name`) it makes the table unique.
- **graph_options.** This variable contains the bulk of the table. As text, it represents all the personalisation configurations that affect the family of graphics the visualisation belongs to.
- **selected.** This simple integer variable represents the index of the type of graphic the user last selected in a specific family.

The most crucial aspect to keep in mind in this table is the variable *graph_options*, as it contains all the personalisation information. The text it contains follows a JSON

structure. The decision to follow such structure was its versatility, as when new configuration options are added to any graphic, it does not disrupt previously functioning nor requires changes to be made to the table in question.

```

1 {
2   "0": {
3     "yTick": 7,
4     "colors": [
5       "#fa4d56",
6       "#5594b4",
7       "#f7913e",
8       "#796662",
9       "#423b67",
10      "#fa4d56",
11      "#570408",
12      "#198038"
13    ],
14    "opacity": 0.9,
15    "scale": 0,
16    "simplify": false,
17    "order": 2,
18    "height": 200,
19    "legend": true,
20    "legend_pos": "bottom",
21    "legend_align": "center",
22    "brush": false,
23    "labelList": true,
24    "labelList_position": "top",
25    "labelList_offset": 5,
26    "labelList_angle": 0,
27    "labelList_simplify": false,
28    "grid": true,
29    "grid_horizontal": true,
30    "grid_vertical": true,
31    "grid_opacity": 0.2,
32    "grid_stroke": false
33  },
34  ...
35 }

```

Listing 4.2: AIVF: Database Tables. Example of *graph_options* content.

Listing 4.2 shows an example of what the variable *graph_options* might contain. In this specific case, one can see that each set of options is named after the index of the type of graphic in the family.

In the example, it is clear that the user has visited the selection 0 (the other options are omitted) from that family, although not indicated in the example because

such information is not required in this database table. For example purposes, imagine there were options for index 1 and 3 as well, this set of configurations are from a graphic that belongs to the *one numerical* family, and it means the user has switched this type of graphic between a bar chart (represented by index 0), a pie chart (represented by index 1) and an area graph (represented by index 3). The last graph selected and the one the user will encounter when coming back to the page in which this graphic is contained is stored in the *selected* variable of the table.

4.6.1.2 Table 2 - User History

This second table handles the recording of a history of personalisation options selected by the user. (see Table 2 in Fig. 4.15). Opposite from the last table presented, this one only requires one entry per user, and so the unique variable of this table is *id_user*.

Besides the user identification variable, there is one to represent each family of graphics available in the framework. Consequently, if a fresh family of graphics is added to the framework, a new column needs to be added to this table to represent that same family. Each of these columns behaves much like the column *graph_options* explained earlier. The whole set of options is stored per type of graph of the family as well as a history.

Considering j to be the number of alterations necessary for a set of options to be stored in the history and n the number of sets stored in the history per family graphic. Both those numbers could be changed in the framework. The higher the n value, the bigger the history record will be; having a more extended history generally means having more information on which to base the adaptation, but obviously, the bigger this number gets, the larger the amount of computation needed for the history to be analysed. The j variable can also be changed in the framework; the smaller it gets, the more communications will be needed to the database in run time; if it gets too big, the system might miss important user decisions that could later be relevant for the adaptation process.

```

1 {
2   "0": {
3     "0": {
4       (options)
5     },
6     "1": {
7       (options)
8     },
9     "selected": 1,
10    "last_updated": true
11  },
12  "1": {
13    "0": {
14      (options)
15    },
16    "1": {

```

```

17         (options)
18     },
19     "selected": 1,
20     "last_updated": false
21 },
22 "2": {
23     "0": {
24         (options)
25     },
26     "1": {
27         (options)
28     },
29     "selected": 1,
30     "last_updated": false
31 },
32 "3": {
33     "1": {
34         (options)
35     },
36     "selected": 1,
37     "last_updated": false
38 },
39 "4": {
40     "1": {
41         (options)
42     },
43     "selected": 1,
44     "last_updated": false
45 }
46 }

```

Listing 4.3: AIVF: Database Tables. Example of user history of personalisation on the *one_numerical* family.

The history storing is first organised by index (the n value just mentioned). In Listing 4.3, that value is set to the number 5; thus, the entries go from 0 to 4. So, each index serves as the label for the objects containing the personalisation information. In those objects, the information stored is the set of configurations of the graphics being modified, the object of that set is identified with the index of selection of the graphic in its specific family.

When there is the need to store information, the framework looks at the current object in the database and looks for a value of n that does not already have information on the specific graphic being modified. If it is found, the information is stored there, the variable *selected* is set to the index of that graphic and the variable *last_selected* is set to true in the index that was just updated. If all objects named with the n indexes already have information on that specific graphic, the information on the index after the one with the variable *last_selected* set to true is modified.

4.6.2 Adaptive Component

The algorithm that handles the adaptation of graphics is simple, and highly user-focused. While previously presented frameworks of this kind opt for an approach primarily based on the whole user base of a system, sometimes even given a more significant weight for certain users considered experts, this proposed approach focus on the user as an individual. If the user base is relevant enough, it does consider that information if the user who requires adaption is new and does not have a record of configurations done.

As explained in the latter subsections, the programmer decides how a particular data query is presented by allocating it to a specific family of graphics. The framework was thought as a means to produce graphics either as a solo on pages or as dashboards for controlled environments. The AIVF facilitates the programmer's work by producing graphics when data is introduced, and a family specified. The adaptation component affects the final user solely. Although at the beginning of usage, a new user might have to do more clicks to personalise the graphics than frameworks that enforce an adaptation algorithm to a whole user base. Nonetheless, in the long run, it should, in theory, result in a user having to do zero alterations to a newly added/encountered graphic on a platform, with the reason being that the adaptation component gave a much heavier weight to the personal user decisions rather than the ones from the user base.

- By default a graphic only suffers adaptation if it is the first time a user sees it. This behaviour can be changed by the user. By default it is turned off as it could become a nuisance for a user to configure a graphic to its liking to later see it getting changed because all the other graphics follow another pattern. In conclusion, the user decision always comes first.
- If it is the first time a user uses a system using the AIVF framework and there is not an extensive enough user base to create a new "default", the first graphics he sees are on the default state.
- If a user configures a graphic before trying other graphics then if he selects other graphics they will suffer adaptation taking in consideration the changes previously made.
- When in need of adaptation the system queries for the user history object (table 2).
- As stated, the weight values could be changed, but for the sake of presenting an example, arbitrary values will be used. For the specific personalisation options the graphic in the process of adaptation requires, the algorithm looks for those same options in the history. It counts for each option what was the most recurrent selection by the user, but takes into consideration were each option was applied:
 - If the history options are from the same type of graphic in the same family, those options count three times more.
 - If the options are from the same type of graphic but belong to another family, the options count two times more.

- If the options do not meet any requirement, they count the default (no scaling).
- Besides those rules, if the options history has the check of *last_selected* those configurations count two times more, as they are the most recent set of options.
- If it is the first time a user uses a system using the AIVF framework and there is an extensive enough user base, the same process applies but with the history from other users.

In conclusion, the system looks for each option on the graph being adapted in the history and counts which option has been selected more times. This makes sense for Boolean values but stops being effective if the option is on a continuous scale. If the continuous scale is small (for example, a slider with a selection from 0-5), it works like a Boolean value as the system just counts the value selected more times, but in other cases, this approach is unreasonable. In those cases, the system looks for values between specific intervals. For example, the height selection of a graphic goes between 200 and 1000 pixels; when looking for a history on the selection of height, the system looks for values on intervals of 50 pixels and counts them and so the adapted graph can only have an initial height that is a multiple of 50 and falls in the respective selection interval. This approach is harder to apply to colour and is something that needs improving in this implementation of the framework as at the moment, the colour selection is made by counting matching colours rather than colour intervals.

The adaptation component handles the graph selection the user sees on a graph for the first time it is loaded, although a user is able to change to any of the graphs presented in a family at any time. The most relevant part of the adaptation is through the personalisation options of each graphic; this is possible because graphics follow specific rules. Different graphics share a lot of the same characteristics, for example, colour, labels and grid.

As stated, when the user is new, and the user base is big enough, the adaptation process could take into deliberation the options of the user base. This works in the same manner as it would if it considered the user history, but it takes a considerable amount of other users history. After said adaptation, the user starts building its own history and so there is no need to take into consideration the rest of the users.

4.7 Conclusion

The prototype developed fits the needs of the requirements of the visualisation module of the CIGESCOP platform. The implementation also shows the usefulness of the AIVF concept and its simplicity towards the developer who uses it and the end-user on the interaction with the visualisations and its personalisation options.

Many of the figures in this chapter were of graphics produced using the prototype and ASAE's data displaying how the personalisation options can be used to achieve graphics that cater to the end-user necessities and preferences. It was also explained how convenient it can be for the developer just to submit the data and a small object

containing metadata and instantly provide a whole set of configurable visualisations to the user.

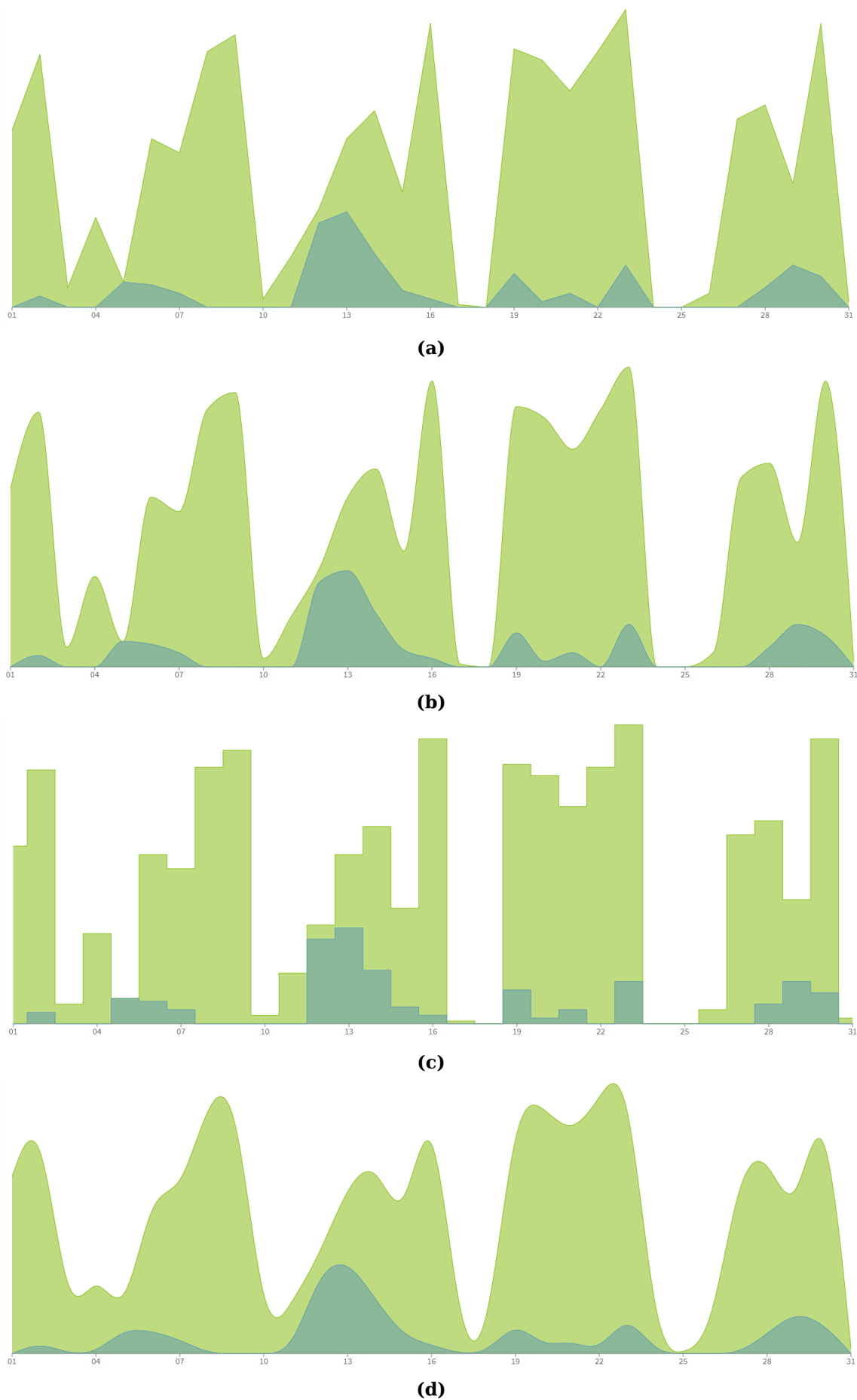


Figure 4.7: AIVF: Personalisation. Interpolation. **a)** Linear. **b)** Monotone. **c)** Step. **d)** Basis.

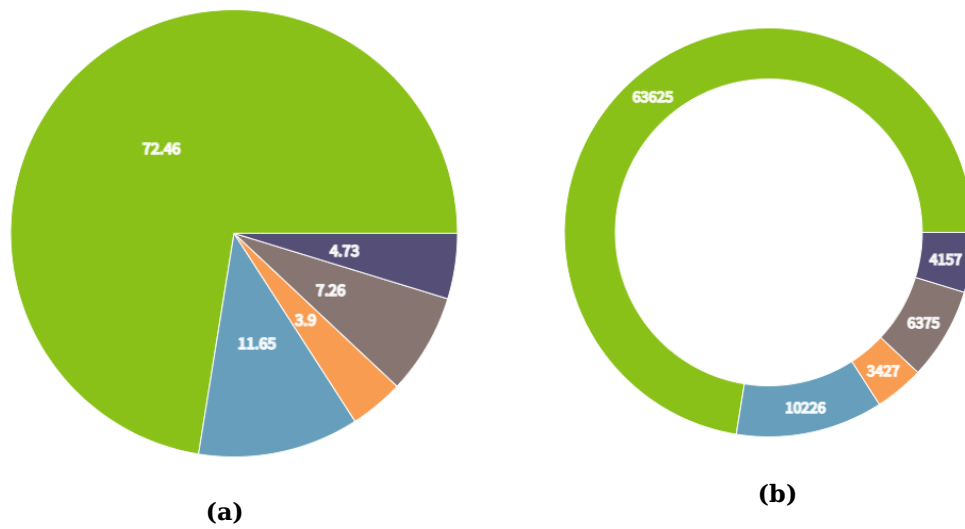


Figure 4.8: AIVF: Personalisation. Percentage and pie plot radius. **a)** The values are set to appear in percentage form with two places; also, the inner radius is set to 0. **b)** The values are in their raw state, and the inner radius is set to be less than 50 pixels than the outer radius.

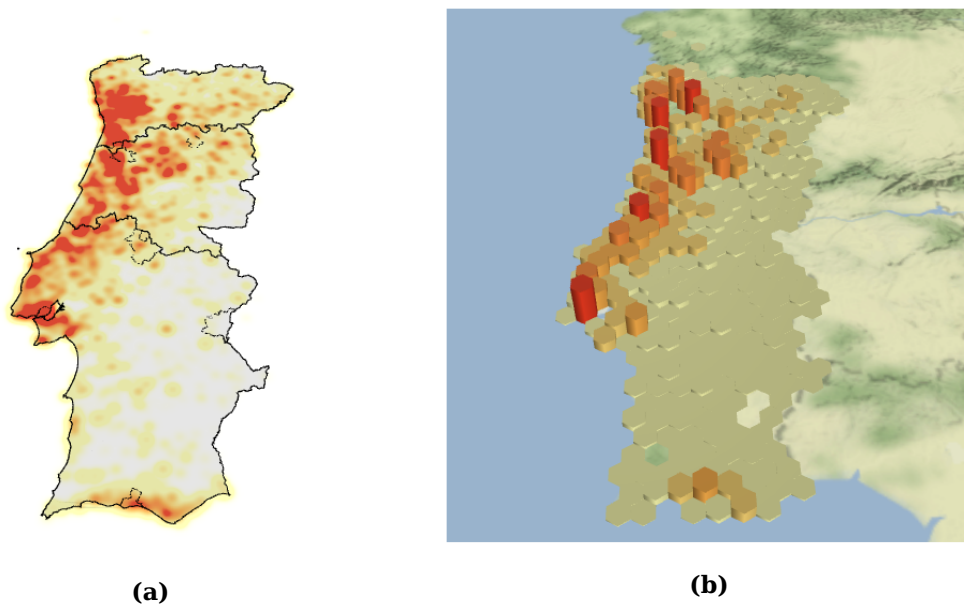


Figure 4.9: AIVF: Charts. **a)** Example of a Heat Map. **b)** Example of a Hexagonal Map. Both graphs hold the same data, the personalisation options have been tweaked to serve as an example.

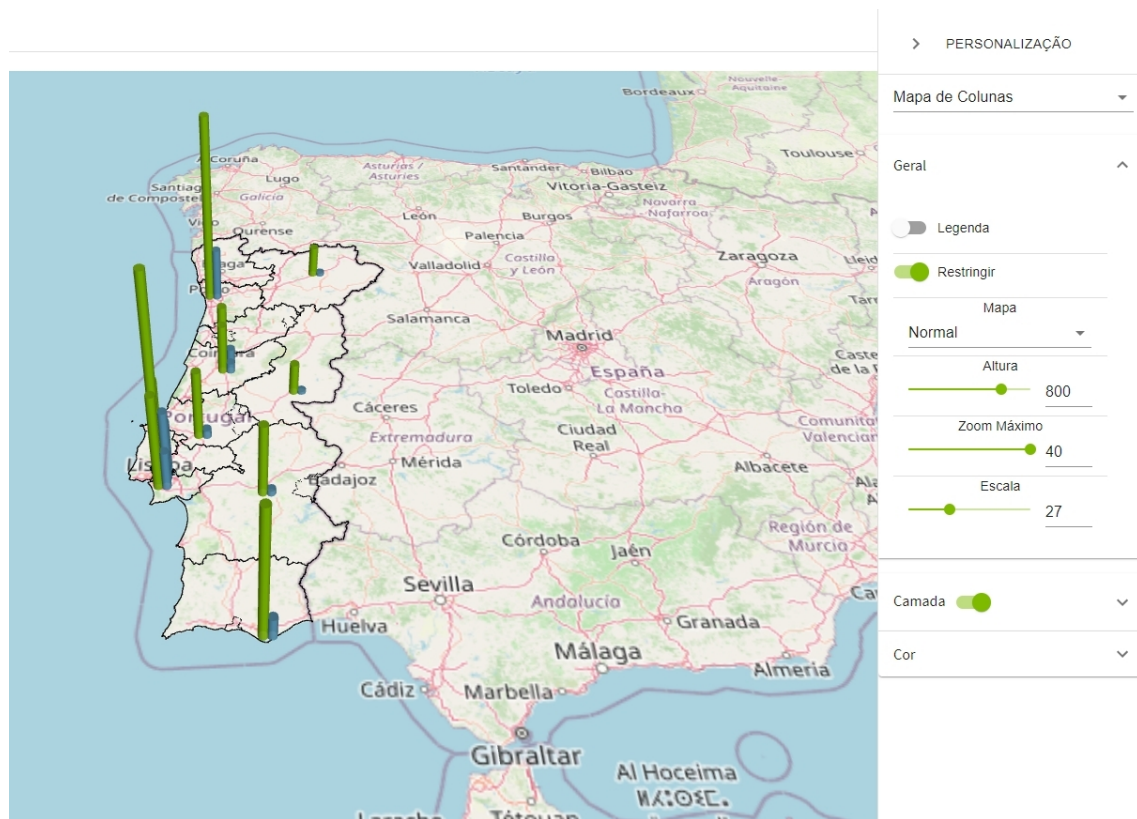


Figure 4.10: AIVF: Personalisation. How the personalisation options are shown. Example 1.

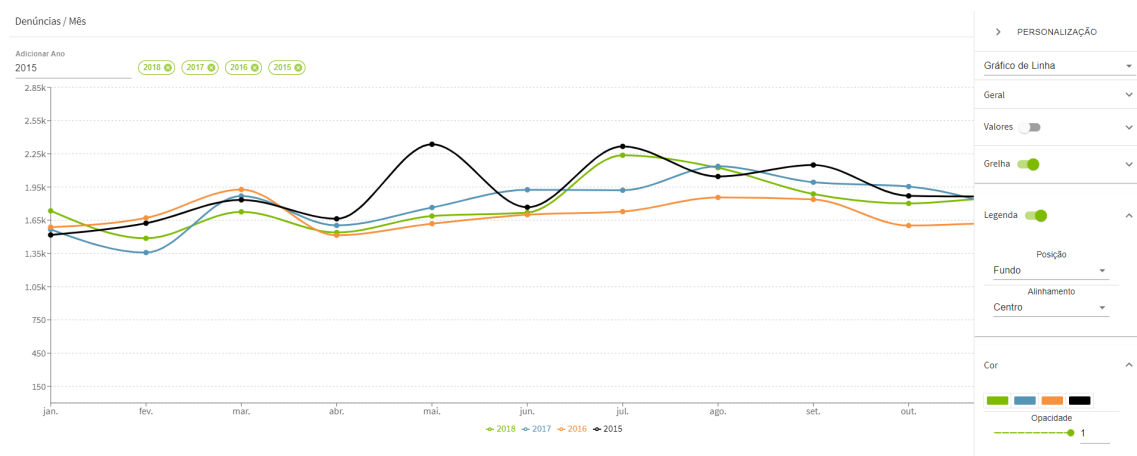


Figure 4.11: AIVF: Personalisation. How the personalisation options are shown. Example 2.

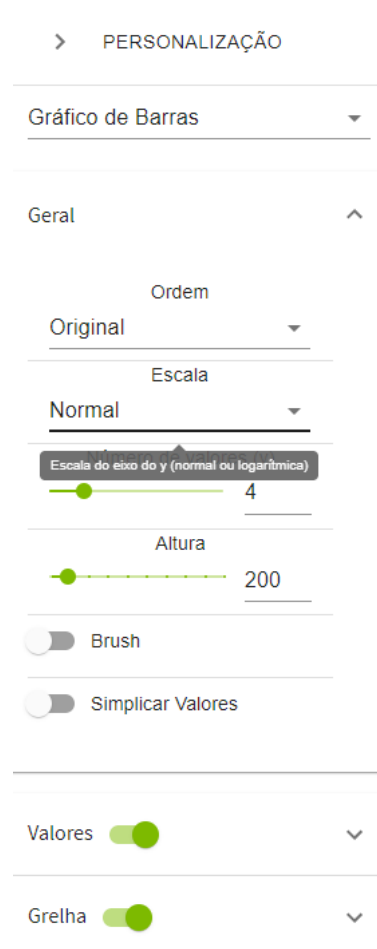


Figure 4.12: AIVF: Personalisation. How the personalisation options are shown. Tool-tips.



Figure 4.13: AIVF: Charts. Column Map.

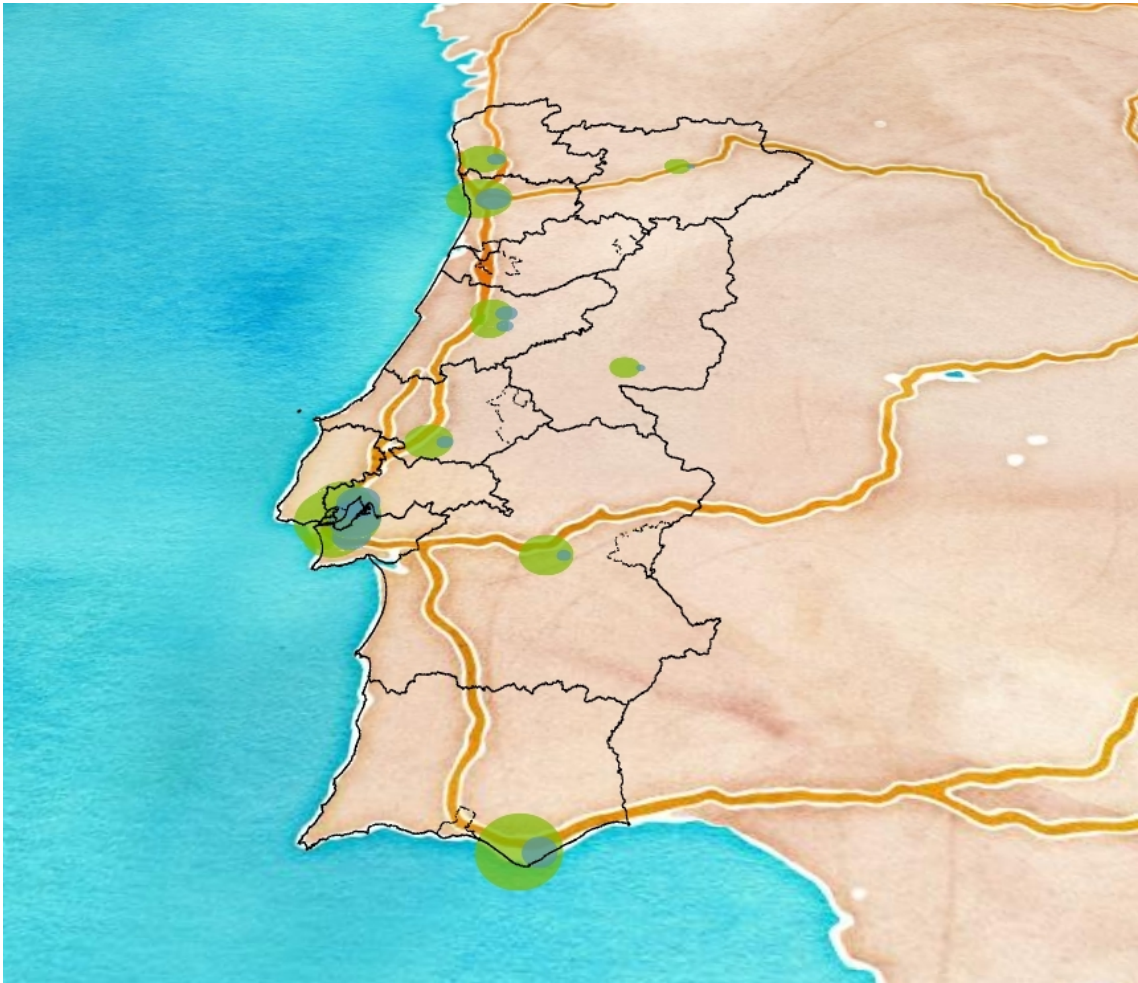


Figure 4.14: AIVF: Charts. Bubble Map.

Table 1	Table 2
id_user	id_user
page_name	one_numerical
page_index	two_numerical
graph_options	time_series
selected	...

Figure 4.15: AIVF: Database Tables.

Chapter 5

Usability Study and Results

To conduct usability studies on the implemented framework prototype there was a necessity to use it in a platform. As mentioned in previous chapters, the framework was conceptualised to solve the visualisation necessities of the GIGESCOP platform. Like any other framework, there is a component of versatility, and so it can be implemented in a whole extensive set of systems, but it was essential to test it on the platform whose problems it tried to solve.

The two necessary tables were created (see section [4.6.1](#)) alongside a third one for the sole purpose of storing metrics (later explained in more detail).

5.1 Usability Testing

Usability testing measures the quality of some piece of software, usually from the perspective of potential users. As the name suggests, usability studies directly measure the usability of a system, which indirectly evaluates the quality and functionality of what is being tested.

These types of tests are significant, as when discussing human-computer interaction, following the most recent research results is not enough to get certainty that a system component works.

According to [[Folmer et al., 2003](#)] usability can be described as having four primary attributes. This fact, taking into consideration what is stated in the article, is the result of extensive research from several authors, and the resultant consensus is the following four attributes:

- **Learnability.** Implied by its naming, this attribute refers to the learning process towards a new system. It describes how easily and quickly can a user begin to work with a system and be productive concerning the system objective.
- **Efficiency of use.** This attribute simply refers to the number of specific tasks a user can accomplish in a platform per time unit.
- **Reliability in use.** Refers to the reliability of a system regarding errors and the time it takes to recover from said errors.
- **Satisfaction.** This last attribute caters for the overall opinions the users elaborate when using the system.

The enumerated attributes can be measured by asking users opinions after having them use the system to be evaluated. Thus, the tests were done with the intention to get insights into all those four attributes.

One other important point to discuss is the potential users of a system and how ample their age and characteristics are, and so it is adequate to have a set of study participants that represent different perspectives.

5.2 The Study

A different approach was applied to this testing process. Usually, people gather in one place and follow a script to perform certain tasks on a system while a person oversees their reaction and satisfaction with the system's functionalities. In this case, participants were given temporary new credentials to access the system, and a script was provided with the task they should perform alongside the questionnaire and some other aspects. This way, participants had a free experience (like the potential user will have), not worrying about time limits or being forced to use the system for more than they would desire.

The questions presented throughout the questionnaire (see appendix A) are meant to address the attributes explored in the last section: learnability, the efficiency of use, reliability in use, and overall satisfaction.

The questionnaire was made available with the use of the famous Google Forms platform¹.

5.2.1 Metrics Saved

The metrics saved are simple and straightforward. As declared, an extra database table was created to store valuable information for the tests:

- Timestamps are saved on the first time the user accesses the system, which is when the user starts the testing process as new accounts were given to the participants to complete the usability testing process.
- Every time a history of personalisation options is saved, another timestamp is saved to get the overall time the user spends on the system.
- Another value saved is the number of times a user changes configuration options to get an overall idea of how much the participant spent playing with the personalisation options.

5.2.2 Introduction

At first, in the script, there was a brief description of the framework and how it proposes to solve the problems of visualisation in a system. This description was done such that the participant had a clear idea of what they were testing while not giving too much information as it was intended for the participants to have the same experience in the platform as any new potential user.

¹More information available on <https://www.google.com/forms/about>

After the description, a short statement was included to inform the participants that the results from the questionnaire and the metrics saved were handled as a whole to guarantee anonymity for the participants.

Next to the statement guaranteeing anonymity, another small paragraph explained that all the questions about the system were to be answered using a Likert scale. In this case, the Likert scale used a range from 1 to 5 with the values meaning:

- 1 - Totally disagree.
- 2 - Disagree.
- 3 - Does not agree but also does not disagree (neutral opinion).
- 4 - Agree.
- 5 - Strongly agree.

5.2.3 Gathering Information About the Participant

After the introductory section of the questionnaire and before the user tried the system, a few questions were shown. First classic questions are presented, such as asking the gender of the participants and date of birth to get an idea of the range of the participants' perspectives. After these two questions, the questionnaire only presents questions answered with the Likert scale or optional open answers. The questions on the information gathering section are the following:

- **1 - I deal with data thought graphics with frequency.** This statement is essential to get information on the participant's data analysis expertise, making it worthwhile to understand the range of participants' skills on such a critical factor to the context of the framework.
- **2 - It is important to have options to personalise a website or application.** It was interesting to get the participant opinion of how important it is to have the option to personalise a platform on its whole.
- **3 - It is important to have the option to personalise graphic visualisations.** Besides wanting to know the participants' opinion on the possibility of personalisation of a whole system, it was also essential to confirm if the users also want to configure their visualisations.

5.2.4 Usability Questions

After quickly gathering information on the participants, they are asked to visit the system for the first time. At first, they see a typical page asking for credentials to enter the system. After login in, the participants are redirected to a dashboard with several graphics containing ASAE's data. At this point, they are asked to explore the dashboard and do as many modifications as they desire on the personalisation of the visualisations. After exploring the questions about usability are presented:

- **1 - The system was easy to use.** This standard usability question is beneficial because if the participants think the system evaluated is not easy to use, it means there are critical problems with the developed solution.

- **2 - I would like to use this type of system with frequency.** This complements the last point, the system could be easy to use, but the participant might not desire to use it with frequency. This point might also reveal critical problems with the solution if the answers are negative.
- **3 - The system was inconsistent.** If the answers to this point are negative, it means there were too many errors, and bugs and/or the organisation of the interface is not consistent and needs to be worked out.
- **4 - The system is easy to learn for most people.** Besides having the information of what the test participant thinks of the easiness of using the developed solution, it is also important to gather opinions on how they think other potential users experience on the system could be.
- **5 - The interface is pleasant.** Sometimes a system or platform works as it is supposed to, it is easy to understand, and it is even strongly consistent but is aesthetic unpleasing; this question purpose is to gather information on the design aspect of the solution and if it needs to be revised.
- **6 - For data exploration, having the option to change visualisation is useful.** This question is used to obtain information if the user would prefer that the optimal visualisation was picked for them without the option to change to another type of graphic because it could be overwhelming, for example.
- **7 - The personalisation options were explicit.** A more specific point about the developed prototype. As there are many personalisation options and the plan is to add more eventually, it is mandatory to gather information on how the potential users see the current options.
- This section ends with an optional open answer question. The question asks what personalisation options caused confusion or were not understood. This type of question is always an excellent addition to transition sections. It offers the option for the test participant to give a more personal opinion in correlation to what is being evaluated. Usually, more participants skip on this type of question than those who write an answer, as this type of question has to be optional not to overwhelm the participant. The responses are usually filled with valuable content on how to improve the evaluated solution.

5.2.4.1 The Dashboard

As stated, the participant is asked to visit a page and introduce his credentials, being later redirected to a page containing a dashboard—the graphics on that dashboard abstract some information of the ASAE’s organisation. As also mentioned before, that information is private, and so the images here, like in past sections of this document, are censored with black bars to hide what which graphic represents. Another factor is that for this work, the information that could be fetched from ASAE’s data could be only up to the year of 2019 for security reasons because of how sensible and private their information is, as explained previously. To improve the readability the dashboard presented in this section of the tests is divided in 4 images in this document (see Figs. 5.1, 5.2, 5.3 and 5.4).

This testing dashboard was done such that it offers some interesting insights on ASAE's data but also so that it offers all families of graphics available at the moment in the implemented prototype of the framework. Each visualisation is enclosed in a container. The container offers three to four icons depending on the context: One icon to minimise the graphic; another one to toggle filters on and off when they are included; one icon to open the personalisation options (example of this behaviour on Figs. 4.10 and 4.11); and one last icon to maximise a graphic to occupy the entire viewport of the page.

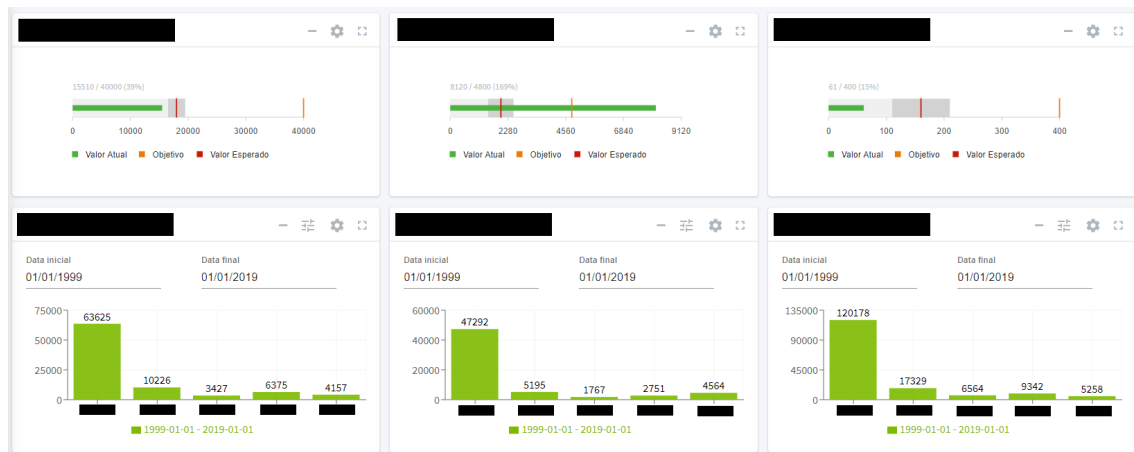


Figure 5.1: Dashboard test page. Part 1.

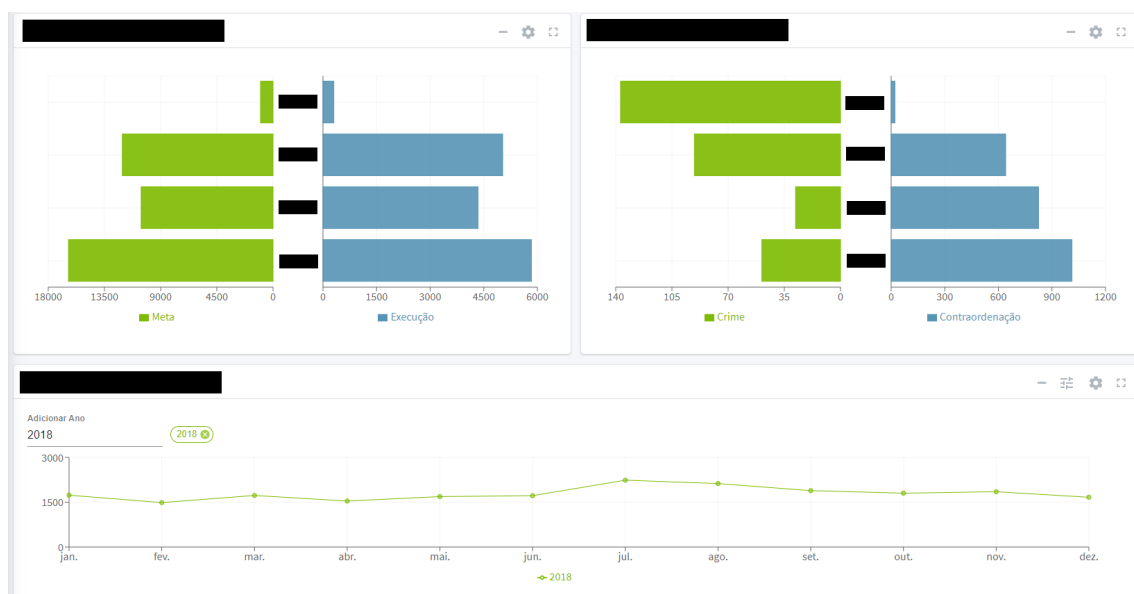


Figure 5.2: Dashboard test page. Part 2.

5.2.5 Adaptability Questions

In the last section, the participants visited a page with several graphics and had the option to configure them as they pleased. In this section, the focus falls on the



Figure 5.3: Dashboard test page. Part 3.

framework's adaptive capabilities, so the participant is asked to visit yet another page, but this time the graphics enclosed are adapted, taking into consideration the decisions made on the page from the last section. This allows for the inclusion of a section on the questionnaire about adaptiveness. This section is also the last step to complete the participation in this study. The questions on this section were the following:

- **1 - The changes made in the last visited page are reflected onto this new one.** Straightforward question for the participant that evaluates the adaption process of the framework directly.
- **2 - The changes made in the last visited page are reflected onto this new page but are not correctly applied.** This question is meant to understand if there are problems with the adaptation algorithm.
- **3 - I would prefer that newly encountered graphics did not take into consideration personalisations made before.** Point made to understand how the participants feel about the adaptive component of the proposed solution.
- **4 - The type of graphics I prefer are shown.** This last question to be answered with the Likert scale is meant to gather information on the graphic selection part and if the suitable information visualisation techniques are presented to the user.

An optional open answer question is presented to finish the questionnaire, asking the participant his overall opinion about the system's efficiency and the functionalities presented to them. It also questions if the participant would like to see any new visualisation introduced into the platform.

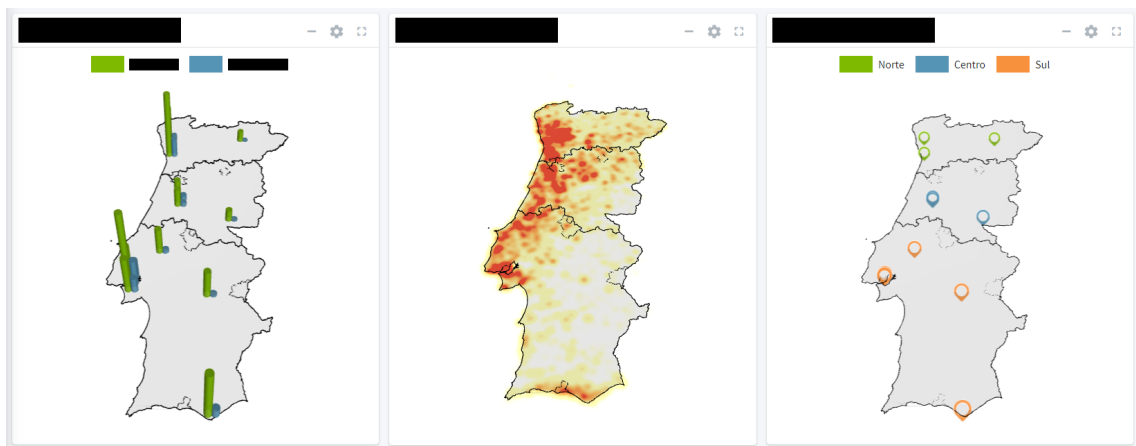


Figure 5.4: Dashboard test page. Part 4.

5.3 Results

As stated the data available on the testing dashboard was private and so the system had to be tested with people that work on the IA.SAE project and have signed a confidentiality document. A total of five participants were included in this test.

5.3.1 Participant Information

There was a total of five participants. Of those five, four were male and one was female. The ages ranged from 25 to 47 years old.

In the guide participants were asked to use the prototype but it was not specified for how long. Those metrics were saved. The time using the system ranged from five minutes to around fourteen. One other aspect saved was the number of changes. The number of changes is a counter that is incremented for every configuration change. For example if it is a Boolean change from true to false the counter is incremented, if it is set back to true the counter is incremented again, but if it is a continuous value change such as height or color the counter is incremented for every three seconds the participant spends on that specific option. This metric was useful to get an idea of the engagement of the participants towards the prototype. The exact values are available on table 5.1

Id	Tempo	Número de mudanças
1	5 minutos, 5 segundos	17
2	9 minutos, 9 segundos	61
3	10 minutos, 14 segundos	87
4	11 minutos, 45 segundos	46
5	13 minutos, 55 segundos	207

Table 5.1: Participant engagement.

The first question to be answered with the Likert scale, which questioned the user experience with data analyses, demonstrated that even though the participant pool

Question id	1	2	3	4	5
1	0	1	1	1	2
2	0	0	0	1	4
3	0	0	0	2	3

Table 5.2: Participant information results.

was relatively small they had different characteristics. The second question demonstrated that users have a desire to have options to personalise a system. The third and last question of the introductory section complemented the result from the last question by having all users agreeing or agreeing strongly that graphics should have personalisation options. Results in graph [B.1](#).

5.3.2 Answers About the Usability

Question id	1	2	3	4	5
1	0	0	0	0	5
2	0	0	0	2	3
3	5	0	0	0	0
4	0	0	0	1	4
5	0	0	0	0	5
6	0	0	0	1	4
7	0	0	0	2	3

Table 5.3: Usability results.

The first question on this section asked the participant if the system was easy to use. The answers were unanimous with the only selection being strongly agree. The following question asked if the participant would like to use this kind of system with frequency. The answers were positive but not unanimous. Some participants chose agree and others strongly agree. The question about the inconsistency of the system was also extremely positive with again an unanimous response. The next point gathered information on how easy the system is to learn for new users. Again the answers were positive. The same goes for the next question about the how pleasant is the interface of the platform. The next two questions offer again positive responses about changing the type of visualisation for better data exploration and about the personalisation options being explicit. Results in graph [B.2](#).

5.3.3 Answers About the Adaptability

The first question of this segment of the questionnaire was about the personalisation options done in the previous page being reflected in this new one. The results were not as positive as was the norm in the usability questions. One participant totally disagreed and one other agreed, the rest strongly agreed. The next inquiry was about having personalisation option reflected in this new page but not being applied correctly. The result were again not unanimous. Two users totally disagreed, one was neutral but of the other two, one agreed and one strongly agreed. The next question

Question id	1	2	3	4	5
1	1	0	0	1	3
2	2	0	1	1	1
3	2	1	2	0	0
4	0	0	1	3	1

Table 5.4: Adaptation results.

asked the opinion of the participant on if they preferred if the newly found graphics did not take in account the personalisation done previously. Two users were neutral, one disagreed and two strongly disagreed. The next question was about showing the correct type of visualisation in this new page according to the selection on the last. One user was neutral towards his inquiry, three other agree and the last one strongly agreed. Results in graph [B.3](#).

5.4 Conclusion

Although the studies were performed with a small participant pool the results were beneficial towards the evaluation of the prototype.

In terms of usability the results were clear. Naturally, an interface can always be improved, but the participants seemed pleased with the interface of the prototype and the way the configuration process works. One important aspect of this is the learnability, with participants spending relatively low amounts of time with the system and still performing a lot of changes and also agreeing that the personalisation options were explicit it means they had no trouble seeing the system for the first time and performing the tasks it was designed to offer.

On the other hand on the adaptability part some confusion appeared. Some participants were satisfied with the adaptation process but the fact that the decision was not unanimous raises problems. There was a positive response towards the need of adaptation but this aspect of the framework needs improvement taking in consideration the answers from all participants.

Chapter 6

Conclusions and Future Work

In this last chapter it is presented a small summary of what has been done and the results of said work. To finish a overview of possible future work is discussed.

6.1 Work Done

This thesis had as the main objective to study and conceptualise a solution to handle the creation of visualisations on the CIGESCOP system. In order to solve the problem presented it was necessary to, in the first place, carefully review the state of art on topics related to the topic of this thesis, deepening the knowledge on topics such as information visualisation, geovisualisation and visualisation recommendation and adaptive visualisation. After knowing the existing approaches to solve problems with some similarity to the one stated, a solution was conceptualised that offered a different perspective and fulfilled the requirements of the visualisation module and had a big focus on personalisation as requested. Later, a prototype of the framework was developed that followed the specifications of the proposed solution. After, the prototype was included in the CIGESCOP system and a dashboard was developed with the prototype by creating effective queries to the ASAE's database. To finish a usability test was conducted on the dashboard created.

6.2 Background Knowledge Acquired

Some conclusions are listed in the scope of the state of the art performed in the preparation. In terms of infoVis, this study allowed to get significant insights into what are KPIs and how they are widely miss-understood. Techniques mostly through graphical representations were explored while also getting insights on how to better use those techniques by looking at some known taxonomies. To complement the study on how to better use those graphics, research was done towards the topic of design, that consequently lead to a literature review on colour. With this research, it was possible to learn how to represent data and present overall good usability considering well known design principles.

In terms of geovisualisation, it allowed to learn about what a GIS is and what it consists of. Then, techniques to expose information on a map were studied. After, it was relevant to learn about technologies that can replicate such techniques. A particular research was taken towards the use of virtual globes oppose the more traditional

approaches in two dimensions, although convenient for some aspects of human perception, 2D map visualisation has strengths that sometimes are hard to replicate in virtual globes. Then relevant practical uses were researched. Moreover, a study was conducted towards general concepts on interactivity, such as the concept of fluidity and flow. Also, a look was taken at technologies to reproduce and aid the visualisation of data. To finish the research on adaptive visualisation and recommendations was useful to detect how these concepts are applied to diverse situations and to get an idea of how they work.

6.3 Usability Study and Results

The framework was conceptualised with an extreme focus on personalisation. A concept of families of graphics for specific situations was presented to aid the developer on the process of creating visualisations using the framework. The prototype showed that it could enrich a system that offers exploration of data and that it could potentially save user time if the system has a lot of visualisations because of the perspective of the adaptiveness of the framework. Another focus of the framework was the simplification of the process of creating visualisations by a developer and including it into a system following a cohesive theme.

Even though the study was conducted with a small number of participants another interesting impact of this work is the result of the study that implicates that users strive for personalisation and are extremely pleased when solutions with a great set of personalisation options are offered to them compared to non personalisable traditional solutions. In terms of the prototype developed the tests were extremely positive when the subject in question was the usability and quality of the interface. When the adaptive process was studied the results were more inconclusive. Some participants enjoyed the process and understood its potential usefulness, while others were somewhat confused with the adaptation process.

6.4 Future Work

In terms of future work there is a possibility to follow a lot of different paths. The prototype developed like all software developed could be improved, and there are always ways in which it could be optimised. The usability study showed that the prototype of the framework has a lot of potential, with that in mind, its adaptive component and algorithm could be improved as some users got somewhat confused with its current functioning. Another path would be to keep on making the prototype more complex by adding more and more configuration options and new types of visualisations. The framework was conceptualised to facilitate the process of data exploration for the user but also to ease the process of introducing visualisations to a system by the developers. And so, another interesting path for future work would be to conduct tests to prove the framework effectiveness from the programmer perspective. There is also the more ambitious perspective of creating mechanisms to provide the user with the power of making their own queries to the database and in conjunction with the proposed framework create their own dashboards and similar solutions.

Bibliography

- [Ahn and Brusilovsky, 2013] Ahn, J. W. and Brusilovsky, P. (2013). Adaptive visualization for exploratory information retrieval. *Information Processing and Management*, 49(5):1139–1164. 10.1016/j.ipm.2013.01.007.
- [Al-Kodmany, 1999] Al-Kodmany, K. (1999). Using visualization techniques for enhancing public participation in planning and design: Process, implementation, and evaluation. *Landscape and Urban Planning*, 45(1):37–45.
- [Andrienko et al., 2010] Andrienko, G., Andrienko, N., Demsar, U., Dransch, D., Dykes, J., Fabrikant, S. I., Jern, M., Kraak, M. J., Schumann, H., and Tominski, C. (2010). Space, time and visual analytics. *International Journal of Geographical Information Science*, 24(10):1577–1600. 10.1080/13658816.2010.508043.
- [Andrienko et al., 2007] Andrienko, G., Andrienko, N., Jankowski, P., Keim, D., Kraak, M. J., MacEachren, A., and Wrobel, S. (2007). Geovisual analytics for spatial decision support: Setting the research agenda. *International Journal of Geographical Information Science*, 21(8):839–857. 10.1080/13658810701349011.
- [Badawy et al., 2016] Badawy, M., El-Aziz, A. A., Idress, A. M., Hefny, H., and Hosam, S. (2016). A survey on exploring key performance indicators. *Future Computing and Informatics Journal*, 1(1-2):47–52. 10.1016/j.fcij.2016.04.001.
- [Bailey and Pregill, 2014] Bailey, J. and Pregill, L. (2014). Speak to the Eyes: The History and Practice of Information Visualization. *Art Documentation: Journal of the Art Libraries Society of North America*, 33(2):168–191. 10.1086/678525.
- [Balasubramani et al., 2020] Balasubramani, B. S., Badhrudeen, M., Derrible, S., and Cruz, I. (2020). Smart Data Management of Urban Infrastructure Using Geographic Information Systems. *Journal of Infrastructure Systems*, 26(4):06020002. 10.1061/(asce)is.1943-555x.0000582.
- [Balla et al., 2020] Balla, D., Zichar, M., Tóth, R., Kiss, E., Karancsi, G., and Mester, T. (2020). Geovisualization techniques of spatial environmental data using different visualization tools. *Applied Sciences (Switzerland)*, 10(19):1–15. 10.3390/APP10196701.
- [Balzer and Deussen, 2005] Balzer, M. and Deussen, O. (2005). Voronoi Treemaps. *Proceedings - IEEE Symposium on Information Visualization, INFO VIS*, pages 49–56. 10.1109/INFVIS.2005.1532128.
- [Bertini et al., 2011] Bertini, E., Tatu, A., and Keim, D. (2011). Quality metrics in high-dimensional data visualization: An overview and systematization.

- IEEE Transactions on Visualization and Computer Graphics*, 17(12):2203–2212. 10.1109/TVCG.2011.229.
- [Carenini et al., 2014] Carenini, G., Conati, C., Hoque, E., Steichen, B., Toker, D., and Enns, J. (2014). Highlighting interventions and user differences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1835–1844, New York, NY, USA. ACM. 10.1145/2556288.2557141.
- [Chi, 2000] Chi, E. (2000). A taxonomy of visualization techniques using the data state reference model. In *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, pages 69–75. IEEE Comput. Soc. 10.1109/IN-FVIS.2000.885092.
- [Çöltekin et al., 2017] Çöltekin, A., Bleisch, S., Andrienko, G., and Dykes, J. (2017). Persistent challenges in geovisualization—a community perspective. *International Journal of Cartography*, 3(sup1):115–139. 10.1080/23729333.2017.1302910.
- [Dix et al., 2003] Dix, A., Finlay, J. E., Abowd, G. D., and Beale, R. (2003). *Human-Computer Interaction 3rd Edition*. Pearson. 978-0130461094.
- [Edsall, 2003] Edsall, R. M. (2003). The parallel coordinate plot in action: Design and use for geographic visualization. *Computational Statistics and Data Analysis*, 43(4):605–619. 10.1016/S0167-9473(02)00295-5.
- [Elmqvist et al., 2011] Elmqvist, N., Moere, A. V., Jetter, H.-C., Cernea, D., Reiterer, H., and Jankun-Kelly, T. (2011). Fluid interaction for information visualization. *Information Visualization*, 10(4):327–340. 10.1177/1473871611413180.
- [Elzer et al., 2011] Elzer, S., Carberry, S., and Zukerman, I. (2011). The automated understanding of simple bar charts. *Artificial Intelligence*, 175(2):526–555. 10.1016/j.artint.2010.10.003.
- [Few, 2005] Few, S. (2005). Keep Radar Graphs Below the Radar—Far Below. *Perceptual Edge*, (May):1–5.
- [Folmer et al., 2003] Folmer, E., Gulp, J. V., and Bosch, J. (2003). Scenario-based Assessment of Software Architecture Usability. *Assessment*, (September 2013):61–68.
- [Forsell and Johansson, 2010] Forsell, C. and Johansson, J. (2010). An heuristic set for evaluation in information visualization. In *Proceedings of the International Conference on Advanced Visual Interfaces - AVI '10*, page 199, New York, New York, USA. ACM Press. 10.1145/1842993.1843029.
- [Forsythe et al., 2016] Forsythe, K. W., Marvin, C. H., Valancius, C. J., Watt, J. P., Aversa, J. M., Swales, S. J., Jakubek, D. J., and Shaker, R. R. (2016). Geovisualization of mercury contamination in Lake St. Clair Sediments. *Journal of Marine Science and Engineering*, 4(1):19. 10.3390/jmse4010019.
- [Freitas et al., 2002] Freitas, C. M., Luzzardi, P. R., Cava, R. A., Winckler, M., Pimenta, M. S., and Nedel, L. P. (2002). On evaluating information visualization techniques. *Proceedings of the Workshop on Advanced Visual Interfaces AVI*, pages 373–374. 10.1145/1556262.1556326.

- [Friendly, 2002] Friendly, M. (2002). A Brief History of the Mosaic Display. *Journal of Computational and Graphical Statistics*, 11(1):89–107. 10.1198/106186002317375631.
- [Ge et al., 2009] Ge, Y., Li, S., Lakhan, V. C., and Lucieer, A. (2009). Exploring uncertainty in remotely sensed data with parallel coordinate plots. *International Journal of Applied Earth Observation and Geoinformation*, 11(6):413–422. 10.1016/j.jag.2009.08.004.
- [Goebel, 2014] Goebel, R. (2014). A sketch of a theory of visualization. *IVAPP 2014 - Proceedings of the 5th International Conference on Information Visualization Theory and Applications*, (January 2014):218–221. 10.5220/0004852702180221.
- [Gotz and Wen, 2009] Gotz, D. and Wen, Z. (2009). Behavior-driven visualization recommendation. In *Proceedings of the 14th international conference on Intelligent user interfaces*, pages 315–324, New York, NY, USA. ACM. 10.1145/1502650.1502695.
- [Guldåker, 2020] Guldåker, N. (2020). Geovisualization and geographical analysis for fire prevention. *ISPRS International Journal of Geo-Information*, 9(6). 10.3390/ijgi9060355.
- [Harrower and Sheesley, 2005] Harrower, M. and Sheesley, B. (2005). Designing better map interfaces: A framework for panning and zooming. *Transactions in GIS*, 9(2):77–89. 10.1111/j.1467-9671.2005.00207.x.
- [Hasrod and Rubin, 2016] Hasrod, N. and Rubin, A. (2016). Defects of colour vision: A review of congenital and acquired colour vision deficiencies. *African Vision and Eye Health*, 75(1):1–6. 10.4102/aveh.v75i1.365.
- [Heinzlef et al., 2020] Heinzlef, C., Becue, V., and Serre, D. (2020). A spatial decision support system for enhancing resilience to floods: Bridging resilience modelling and geovisualization techniques. *Natural Hazards and Earth System Sciences*, 20(4):1049–1068. 10.5194/nhess-20-1049-2020.
- [Kaplan and Norton, 1996] Kaplan, R. S. and Norton, D. P. (1996). *The Balanced Scorecard: Translating Strategy Into Action*. Harvard Business Review Press. 10:0875846513.
- [Kilsedar and Brovelli, 2020] Kilsedar, C. E. and Brovelli, M. A. (2020). Multidimensional Visualization and Processing of Big Open Urban Geospatial Data on theWeb. *ISPRS International Journal of Geo-Information*, 9(7). 10.3390/ijgi9070434.
- [Kraak, 2003] Kraak, M. J. (2003). Geovisualization illustrated. *ISPRS Journal of Photogrammetry and Remote Sensing*, 57(5-6):390–399. 10.1016/S0924-2716(02)00167-3.
- [Lam, 2008] Lam, H. (2008). A Framework of interaction costs in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1149–1156. 10.1109/TVCG.2008.109.

- [Li et al., 2016] Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A., and Cheng, T. (2016). Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115:119–133. 10.1016/j.isprsjprs.2015.10.012.
- [Maqsood et al., 2020] Maqsood, U., Tahir, A., Fatima, K., and Rahman, A. (2020). Interpreting rescue vehicle patterns using geovisual analytics for spatiotemporal resource allocation. *Arabian Journal of Geosciences*, 13(14):1–12. 10.1007/s12517-020-05643-w.
- [Nazemi et al., 2015] Nazemi, K., Burkhardt, D., Hoppe, D., Nazemi, M., and Kohlhammer, J. (2015). Web-based Evaluation of Information Visualization. *Procedia Manufacturing*, 3(Ahfe):5527–5534. 10.1016/j.promfg.2015.07.718.
- [Norman, 2002] Norman, D. (2002). *Design of Everyday Things*. Basic Books.
- [Nuzzo, 2019] Nuzzo, R. L. (2019). Histograms: A Useful Data Analysis Visualization. *PM and R*, 11:309–312. 10.1002/pmjr.12145.
- [Parmenter, 2012] Parmenter, D. (2012). *Key Performance Indicators for Government and Nonprofit Agencies*. John Wiley I& Sons, Inc. 9783540773405.
- [Pauwels et al., 2009] Pauwels, K., Ambler, T., Clark, B. H., LaPointe, P., Reibstein, D., Skiera, B., Wierenga, B., and Wiesel, T. (2009). Dashboards as a service: Why, what, how, and what research is needed? *Journal of Service Research*, 12(2):175–189. 10.1177/1094670509344213.
- [Peterson, 2006] Peterson, E. (2006). The Big Book of Key Performance Indicators. *Web Analytics Demystified*, page 266. -10: 0974358428.
- [Poetzsch et al., 2020] Poetzsch, T., Germanakos, P., and Huestegge, L. (2020). Toward a Taxonomy for Adaptive Data Visualization in Analytics Applications. *Frontiers in Artificial Intelligence*, 3(March):1–16. 10.3389/frai.2020.00009.
- [Qian et al., 2020] Qian, X., Koh, E., Rossi, R. A., Malik, S., Du, F., Lee, T. Y., Kim, S., and Chan, J. (2020). ML-based visualization recommendation: Learning to recommend visualizations from data. *arXiv*.
- [Robinson et al., 2017] Robinson, A. C., Demšar, U., Moore, A. B., Buckley, A., Jiang, B., Field, K., Kraak, M. J., Camboim, S. P., and Sluter, C. R. (2017). Geospatial big data and cartography: research challenges and opportunities for making maps that matter. *International Journal of Cartography*, 3(1):32–60. 10.1080/23729333.2016.1278151.
- [Root et al., 2020] Root, E. D., Bailey, E. D., Gorham, T., Browning, C., Song, C., and Salsberry, P. (2020). Geovisualization and Spatial Analysis of Infant Mortality and Preterm Birth in Ohio, 2008-2015: Opportunities to Enhance Spatial Thinking. *Public Health Reports*, 135(4):472–482. 10.1177/0033354920927854.
- [Roth, 2013] Roth, R. E. (2013). An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization. *IEEE transactions on visualization and computer graphics*, 19(12):2356–65. 10.1109/TVCG.2013.130.

- [Roy, 2015] Roy, S. (2015). Effectiveness of JavaScript Graph Visualization Libraries in Visualizing Gene Regulatory Networks (GRN). (April).
- [Rozanski and Haake, 2017] Rozanski, E. P. and Haake, A. R. (2017). *Human-computer interaction*. 978-0-13-046109-4.
- [Rumsey, 2010] Rumsey, D. (2010). *Statistics Essentials For Dummies*. 978-1-119-59030-9.
- [Shneiderman, 2003] Shneiderman, B. (2003). The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *The Craft of Information Visualization*, pages 364–371. Elsevier. 10.1016/B978-155860915-0/50046-9.
- [Spence, 1980] Spence, R. (1980). *Information Visualization An Introduction*, volume 3. 10.1016/S0166-4115(08)61732-X.
- [Tan et al., 2007] Tan, C. C., Yu, W., and McAllister, G. (2007). An adaptive & adaptable approach to enhance web graphics accessibility for visually impaired people. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1539–1542, New York, NY, USA. ACM. 10.1145/1240624.1240856.
- [Theus, 2012] Theus, M. (2012). Mosaic plots. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(2):191–198. 10.1002/wics.1192.
- [Toker et al., 2013] Toker, D., Conati, C., Steichen, B., and Carenini, G. (2013). Individual user characteristics and information visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 295–304, New York, NY, USA. ACM. 10.1145/2470654.2470696.
- [Tory and Möller, 2004] Tory, M. and Möller, T. (2004). Rethinking visualization: A high-level taxonomy. *Proceedings - IEEE Symposium on Information Visualization, INFO VIS*, (May):151–158. 10.1109/INFVIS.2004.59.
- [Tufte, 2006] Tufte, E. (2006). *Beautiful Evidence*. Graphics Press. 1930824165.
- [Turkay et al., 2014] Turkay, C., Slingsby, A., Hauser, H., Wood, J., and Dykes, J. (2014). Attribute signatures: Dynamic visual summaries for analyzing multivariate geographical data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2033–2042. 10.1109/TVCG.2014.2346265.
- [Vartak et al., 2016] Vartak, M., Huang, S., Siddiqui, T., Madden, S., and Parameswaran, A. (2016). Towards visualization recommendation systems. *SIGMOD Record*, 45(4):34–39. 10.1145/3092931.3092937.
- [Votano et al., 2004] Votano, J., Parham, M., and Hall, L. (2004). Evaluation of ERST. *Chemistry &*, pages 154–167.
- [Wexler et al., 2017] Wexler, S., Shaffer, J., and Cotgreave, A. (2017). *The Big Book of Dashboards*.
- [Wilke, 2019] Wilke, O. C. (2019). *Fundamentals of Data Visualization*. 9781492031086.

- [Wozny, 2015] Wozny, S. (2015). Web based data visualization solutions in quality assurance. 1(1):3–7.
- [Yi et al., 2007] Yi, J. S., Kang, Y. A., Stasko, J. T., and Jacko, J. A. (2007). Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231. 10.1109/TVCG.2007.70515.
- [Yigitbasioglu and Velcu, 2012] Yigitbasioglu, O. M. and Velcu, O. (2012). A review of dashboards in performance management: Implications for design and research. *International Journal of Accounting Information Systems*, 13(1):41–59. 10.1016/j.accinf.2011.08.002.
- [Yin et al., 2015] Yin, S., Li, M., Tilahun, N., Forbes, A., and Johnson, A. (2015). Understanding Transportation Accessibility of Metropolitan Chicago Through Interactive Visualization. In *Proceedings of the 1st International ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics*, pages 77–84, New York, NY, USA. ACM. 10.1145/2835022.2835036.
- [Zacks, 2020] Zacks, J. M. (2020). Designing Graphs for Decision-Makers. *Policy Insights from the Behavioral and Brain Sciences*, 7(1):52–63. 10.1177/2372732219893712.
- [Zichar, 2013] Zichar, M. (2013). Geovisualization-related issues with cognitive aspects. *4th IEEE International Conference on Cognitive Infocommunications, CogInfoCom 2013 - Proceedings*, pages 503–508. 10.1109/CogInfoCom.2013.6719299.

Appendix A

Questionnaire

Framework Adaptativa de Visualização de Informação (Protótipo) - Teste de Usabilidade

*Obrigatório

Descrição

A parcela a ser testada de seguida tem como objetivo agilizar o processo de desenvolvimento de uma plataforma no que toca a visualizações gráficas de informação, enquanto ao mesmo tempo oferece ao utilizador algum controlo no que toca a personalizar os vários tipos de gráficos que vai encontrando e faz com que as várias opções de personalização do utilizador sejam tidas em conta quando forem encontradas/adicionadas novas visualizações em determinada plataforma.

As respostas a este questionário são anónimas assim como as métricas recolhidas ao usar o protótipo cujo endereço é disponibilizado de seguida.

Este tipo de testes/questionário serve para perceber a utilidade do protótipo desenvolvido para o utilizador final. A maioria das questões será respondida com uma escala de Likert (discordo totalmente, discordo, não concordo nem discordo, concordo, concordo totalmente, 1-5 respetivamente).

Género

☐ Feminino

☐ Masculino

Data de Nascimento

Data

dd/mm/aaa:

Lido com avaliação de dados através de gráficos com frequência. *

12345

Discordo totalmente☐☐☐☐☐Concordo totalmente

É importante ter opções para personalizar um sistema (website / aplicação). *

12345

Discordo totalmente☐☐☐☐☐Concordo totalmente

É importante ter a opção de personalizar gráficos individualmente. *

12345

Discordo totalmente☐☐☐☐☐Concordo totalmente

Seguinte

Página 1 de 3

Este formulário foi criado dentro de Universidade do Porto: [Denunciar abuso](#)

Google Formulários

Figure A.1: Questionnaire. Page 1.

Usabilidade

Para responder às próximas perguntas é necessário utilizar a parcela do sistema em questão. Para isso utilize o seguinte endereço e introduza as credenciais fornecidas: [redacted] (a página pode demorar a carregar no primeiro acesso). Após a introdução das credenciais será redireccionado para uma página, nessa mesma página explore os gráficos e as opções de personalização durante alguns minutos.

Achei o sistema fácil de usar. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Acho que gostaria de usar o sistema com frequência. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Achei o sistema muito inconsistente. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

A maioria das pessoas aprenderiam a usar este tipo de sistema rapidamente. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

A interface do sistema é agradável. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Em termos de exploração dos dados disponíveis mudar o tipo de gráfico original é útil. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

As opções de personalização encontradas foram explícitas. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Que opções causaram confusão ou não foram percebidas? (Opcional)

A sua resposta

Página 2 de 3

[Anterior](#) [Seguinte](#)

Este formulário foi criado dentro de Universidade do Porto. [Denunciar abuso](#)

Google Formulários

Figure A.2: Questionnaire. Page 2.

Adaptabilidade

Uma vez terminada a secção anterior e para terminar siga para este novo endereço que representa novos gráficos encontrados (são iguais aos da página anterior, mas para âmbito dos testes representam novos gráficos) que ao serem vistos pela primeira vez são adaptados com algumas das opções de personalização da fase anterior: [redacted]. Veja a página e proceda para responder a este ultimo conjunto de questões.

As mudanças efectuadas na página anterior estão reflectidas nesta nova página. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

As mudanças efectuadas estão reflectidas nesta nova página mas não estão aplicadas correctamente. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Preferia que os novos gráficos encontrados não tivessem em conta as opções de personalização aplicadas anteriormente. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Os tipos de gráficos que prefiro estão presentes. *

1 2 3 4 5

Discordo totalmente ☐ ☐ ☐ ☐ ☐ Concordo totalmente

Opinião geral em relação à Framework, nomeadamente eficiência e funcionalidades ou tipos de gráficos que devem estar presentes mas não estão. (Opcional)

A sua resposta

Página 3 de 3

[Anterior](#) [Submeter](#)

Este formulário foi criado dentro de Universidade do Porto. [Denunciar abuso](#)

Google Formulários

Figure A.3: Questionnaire. Page 3.

Appendix B

Test Results

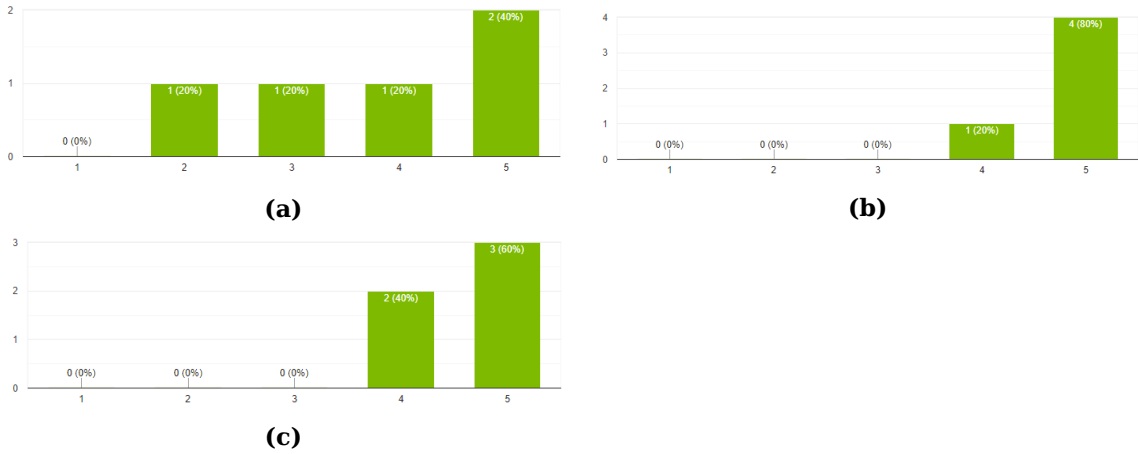


Figure B.1: Participant information results. **a)** 1 - I deal with data thought graphics with frequency. **b)** 2 - It is important to have options to personalise a website or application. **c)** 3 - It is important to have the option to personalise graphic visualisations.

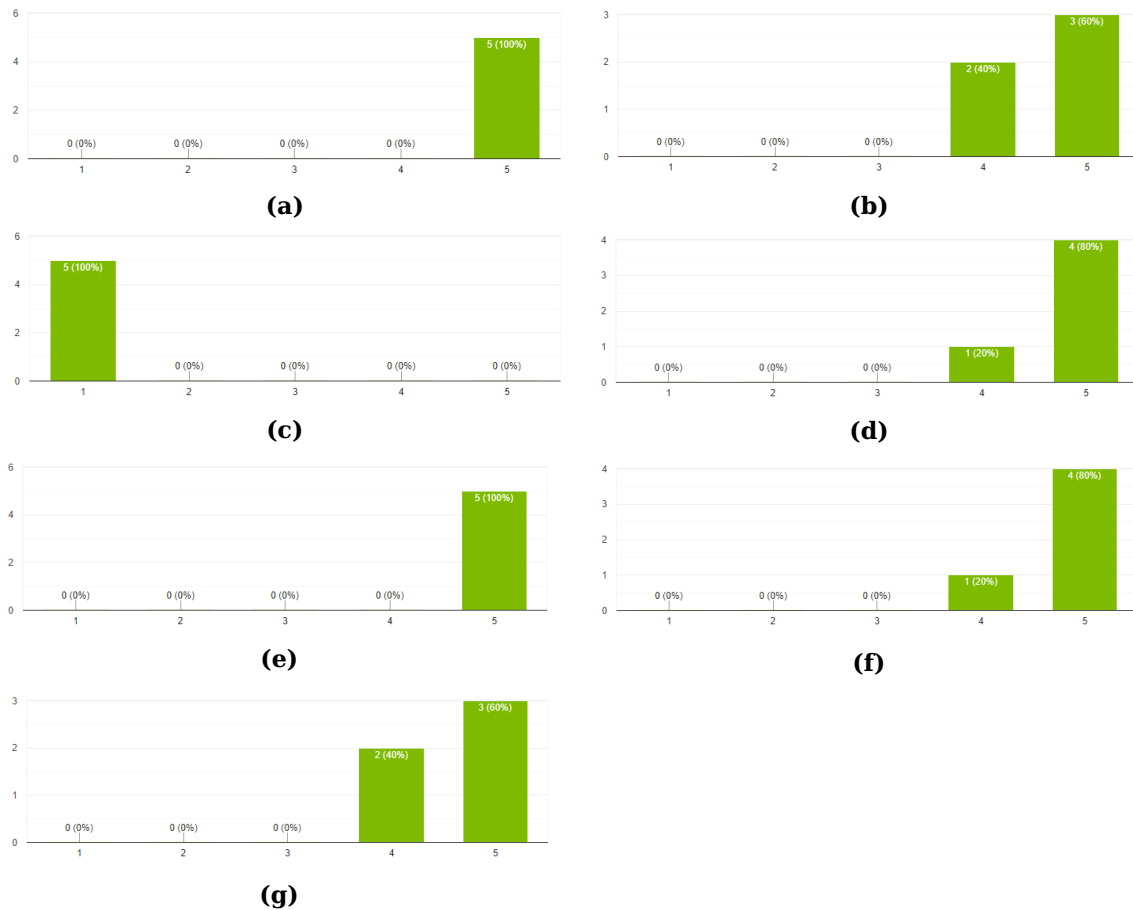


Figure B.2: Usability results. **a)** 1 - The system was easy to use. **b)** 2 - I would like to use this type of system with frequency. **c)** 3 - The system was inconsistent. **d)** 4 - The system is easy to learn for most people. **e)** 5 - The interface is pleasant. **f)** 6 - For data exploration, having the option to change visualisation is useful. **g)** 7 - The personalisation options were explicit.

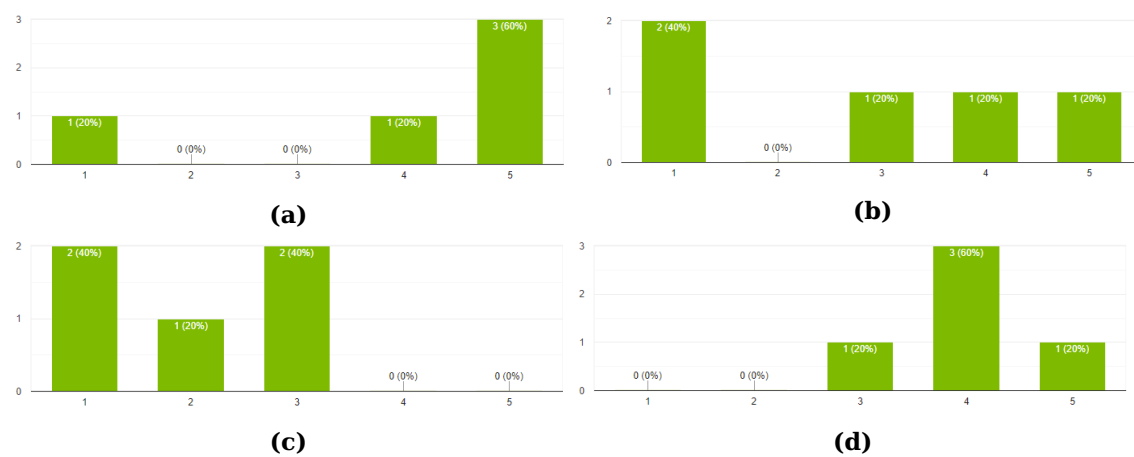


Figure B.3: Adaptation results. **a)** 1 - The changes made in the last visited page are reflected onto this new one. **b)** 2 - The changes made in the last visited page are reflected onto this new page but are not correctly applied. **c)** 3 - I would prefer that newly encountered graphics did not take into consideration personalisations made before. **d)** 4 - The type of graphics I prefer are shown.