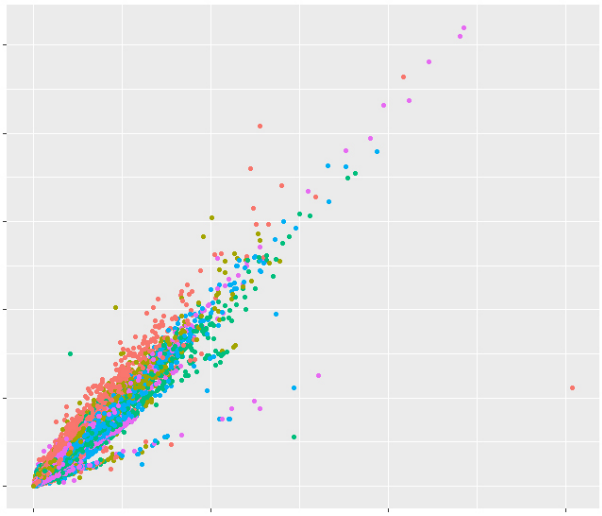
Data Story Telling and Analytics

Case study: An interactive approach using R Shiny, Leaflet and D3

Author:
Georgios Ouzounidis

Supervisor:
Professor Yannis Theodoridis
Department of Informatics, University of Piraeus

A thesis submitted in partial fulfillment of the requirements of the degree of
Master of Science in Geoinformatics



School of Rural and Surveying Engineering
National Technical University of Athens

November 26, 2017

Member of the Supervisory Committee:
Professor Yannis Theodoridis

Chairperson of the Supervisory Committee:
Professor Marinos Kavouras

Member of the Supervisory Committee:
Professor Yannis Theodoridis

Member of the Supervisory Committee:
Assistant Professor Pelekis Nikolaos

November 26, 2017

This study was typeset using these shareware software:

(a) Texmaker 4.5, available at: http://www.xm1math.net/texmaker
(b) MikTeX 2.9.5840, available at: https://miktex.org/download

Contents

List of Figures

List of Tables

Abstract

Understanding the basics of math and graphs is becoming increasingly necessary in many aspects of every day life, as the age in which we live today is the age of data. Either spatial or not, data is everywhere. Statistics, programming, complex processes and further more are behind the results and conclusions extracted from several data sources. This complexity varies from one process to another. However, it always has at least a minimum level of difficulty that requires the contribution of an expert.

The objective of this study is to simplify the way we look into data and propose an interactive approach in which the user would be able to interact, communicate and understand the results of the data analysis without having a statistical and technical background. In this scope, the study describes from the very beginning to the very end the process of knowledge discovery from data and suggests an interactive presentation of the results based on storytelling concepts with a particular focus on geographic data. In order to achieve this, the study is organized as follows:

The first chapter focuses on basic theory concepts of storytelling and especially data storytelling. This chapter also describes the overview of the methodology of extracting knowledge from data and presents a list of web technologies used for visualization of geographic data. Furthermore, it describes data mining techniques with an emphasis on specific classification methodologies and clustering techniques.

The second chapter focuses on the first steps of creating an interactive data storytelling web application. These steps include data preparation and data processing.

The third chapter illustrates the data analysis step. It describes statistical methods and in general mining techniques applied in order to extract knowledge from data. The current case study analysis uses descriptive statistics, probability functions, classification and clustering methodologies as well as predictive analysis. The data used refer to Marineculture industry and are derived from an Aquaculture Company Group located in Greece.

The forth chapter describes the technologies that allow to build an environment with interactive characteristics. This chapter focuses on the implementation of the web application based on data storytelling concepts.

The last chapter contains a review of the whole process and presents ideas on future work.

Περίληψη

Η τελευταία δεκαετία χαρακτηρίζεται από την εκθετική ανάπτυξη των τεχνολογιών διαδικτύου και την εκτεταμένη χρήση των μέσων κοινωνικής δικτύωσης. Χαρακτηριστικό της τάσης των χρηστών διαδικτύου της σύγχρονης εποχής αποτελεί η αυξανόμενη αλληλεπίδρασή τους με τα πλατφόρμες εισαγωγής και ανταλλαγής πληροφοριών. Αυτό έχει ως αποτέλεσμα την συγκέντρωση μεγάλου όγκου πληροφορίας ακόμα και όταν πρόκειται για μικρότερου εύρους εφαρμογές. Πλατφόρμες όπως οι Google, Facebook, Twitter κ.ά. συγκεντρώνουν καθημερινά τεράστιες ποσότητες πληροφορίας που στο παρελθόν συλλέγονταν σε δεκαετίες. Με την ανάπτυξη των τεχνολογιών που αφορούν στην διαχείριση και αποθήκευση δεδομένων, η συλλογή της πληροφορίας αποτελεί πλέον μια από της σημαντικότερες διεργασίες της κάθε εφαρμογής. Συνεπώς η σύγχρονη εποχή δικαίως χαρακτηρίζεται ως η εποχή της πληροφορίας ή διαφορετικά η εποχή των Big Data.

Μεγάλο μέρος των δεδομένων που συλλέγονται είναι αποτέλεσμα εφαρμογών που σχετίζονται και με την γεωγραφική τοποθεσία. Το ποσοστό των χωρικών δεδομένων στο σύνολο των δεδομένων που αποθηκεύονται καθημερινά αυξάνεται σταδιακά. Αυτό συμβαίνει διότι η ανάγκη για τη γνώση της τοποθεσίας γίνεται όλο και περισσότερο ενδιαφέρουσα όσο οι τεχνολογίες που ασχολούνται με την χωρική πληροφορία εξελίσσονται. Οι σύγχρονες πλατφόρμες αποθήκευσης πληροφορίας διαθέτουν ως αναπόσπαστο κομμάτι τους τη χωρική διάσταση των δεδομένων, γεγονός που στο παρελθόν απαιτούσε τη χρήση εξειδικευμένων τεχνολογιών και τη γνώση συστημάτων γεωπληροφορικής. Η εξέλιξη των εργαλείων επεξεργασίας, διαχείρισης, αποθήκευσης, συντήρησης και οπτικοποίησης χωρικών δεδομένων έχει δημιουργήσει νέες προοπτικές ανάλυσης και εξαγωγής συμπερασμάτων από τα δεδομένα.

Παράλληλα, ο αυξανόμενος όγκος δεδομένων που παράγεται καθημερινά δημιουργεί την ανάγκη για την εξαγωγή χρήσιμων συμπερασμάτων. Επιπλέον η παρουσίαση των αποτελεσμάτων, κυρίως όταν πρόκειται για αποτελέσματα που προκύπτουν από μια σύνθετη ανάλυση, παίζει ιδιαίτερα σημαντικό ρόλο στο τελικό στόχο της διαδικασίας που είναι η επικοινωνία στα ενδιαφερόμενα μέρη.

Σκοπός της παρούσας εργασίας είναι η δημιουργία ενός περιβάλλοντος το οποίο θα παρουσιάζει με σύντομο, απλό και κατανοητό τρόπο προς το χρήστη τα απο-

τελέσματα μιας εξειδικευμένης ανάλυσης δεδομένων. Στην επίτευξη του στόχου αυτού συμβάλλει η θεωρία της αφηγηματικής παρουσίασης (Storytelling) και η χρήση διαδραστικών εργαλείων. Το τελικό αποτέλεσμα έχει ως στόχο την επικοινωνία των αποτελεσμάτων της ανάλυσης με όσο το δυνατόν πιο απλό και κατανοητό τρόπο στον χρήστη στον οποίο απευθύνεται, μέσω μιας διαδικτυακής εφαρμογής. Επιπλέον, στόχος της εφαρμογής είναι η δημιουργία κατάλληλων εργαλείων που θα δίνουν τη δυνατότητα στο χρήστη να πραγματοποιήσει βασική ανάλυση και παρουσίαση της γεωγραφικής διάστασης των δεδομένων.

Αναλυτικά, η εργασία ασχολείται με την διαδικασία εξαγωγής συμπερασμάτων από ένα σύνολο δεδομένων και με την επικοινωνία των αποτελεσμάτων μέσω μιας αφηγηματικής παρουσίασης στο πλαίσιο μιας διαδικτυακής εφαρμογής. Η εργασία χωρίζεται σε δύο μέρη: (α) στο πρώτο μέρος αναφέρονται οι βασικές έννοιες και μεθοδολογίες που σχετίζονται με τη διαδικασία εξαγωγής συμπερασμάτων από βάσεις δεδομένων. Επίσης περιγράφονται κεντρικές έννοιες της εργασίας όπως, analytics, storrytelling, kdd process κ.α. που χρησιμοποιούνται στο στάδιο υλοποίησης της εφαρμογής. (β) Στο δεύτερο μέρος, περιγράφονται επιγραμματικά τα στάδια υλοποίησης μιας διαδικτυακής εφαρμογής που έχει ως στόχο την παρουσίαση των αποτελεσμάτων από την ανάλυση μιας βάσης δεδομένων, χρησιμοποιώντας μια αφηγηματικού τύπου προσέγγιση.

Συγκεκριμένα, το πρώτο κεφάλαιο επικεντρώνεται στην αναφορά κεντρικών εννοιών γύρω από τη θεωρία του Storytelling και ιδιαίτερα του Data Storytelling. Στο ίδιο κεφάλαιο περιγράφεται και μια σειρά μεθοδολογιών ανάλυσης δεδομένων και αναλύεται η διαδικασία της εξαγωγής συμπερασμάτων από βάσεις δεδομένων (KDD process). Τέλος παρουσιάζεται μία λίστα με τις διαθέσιμες τεχνολογίες διαδικτύου για την οπτικοποίηση γεωγραφικών πληροφοριών και αναλύονται συνοπτικά οι δυνατότητές τους. Μέρος των μεθοδολογιών που περιγράφονται χρησιμοποιούνται σε επόμενα κεφάλαια, στο πλαίσιο του σχεδιασμού και της δημιουργίας μιας Data Storytelling εφαρμογής.

Το δεύτερο κεφάλαιο επικεντρώνεται στην επιλογή, τη διάθεση και την αποθήκευση των δεδομένων που θα χρησιμοποιηθούν στην εφαρμογή. Συγκεκριμένα, οι πληροφορίες που χρησιμοποιούνται στην παρούσα εφαρμογή έχουν συλλεχθεί από την παγκόσμια τράπεζα πληροφοριών (The world Bank) και μεγάλο ευρωπαϊκό όμιλο που δραστηριοποιείται στον κλάδο της ιχθυοκαλλιέργειας. Τα δεδομένα αφορούν σε ποσότητες και αξία πωλήσεων του ομίλου για την περίοδο από 1/1/2013 εώς 22/8/2017. Τα δεδομένα περιέχουν πληροφορία που αφορά στο σημείο πώλησης σε επίπεδο χώρας, το είδος του προϊόντος (Λαυράκι, Τσιπούρα, Φαγκρί και Κρανιός), το βαθμό επεξεργασίας του τελικού προϊόντος (απεντερωμένα, ολόκληρα και φιλέτα), την κατηγορία μεγέθους, τη γενική κατηγορία (κατεψυγμένο και φρέσκο), την ημερομηνία αποστολής, την ποσότητα αποστολής και την αξία. Συνεπώς, στο στάδιο αυτό πραγματοποιείται η συλλογή και αποθήκευση των πληροφοριών από τις παραπάνω πηγές, η πρώτη επεξεργασία, το καθάρισμα των δεδομένων και η αποθήκευση της τελικής τους μορφής σε μια

αξιοποιήσιμη δομή προς περαιτέρω ανάλυση.

Στη συνέχεια, το τρίτο κεφάλαιο επικεντρώνεται στην ανάλυση των δεδομένων και στην παρουσίαση των κεντρικών συμπερασμάτων της ανάλυσης. Στο πλαίσιο αυτό χρησιμοποιούνται μεθοδολογίες περιγραφικής στατιστικής (Μέγιστη τιμή πλήθους, Μέσος Όρος, Αθροιστική συχνότητα, Συνάρτηση πυκνότητας, Συνάρτηση πιθανότητας κλπ), μεθοδολογίες ταξινόμησης (Μέθοδος ταξινόμησης ίσων διαστημάτων), τεχνικές ομαδοποίησης δεδομένων και μεθοδολογίες πρόβλεψης (Predictive Analysis). Σκοπός της ανάλυσης είναι η εξαγωγή χρήσιμων συμπερασμάτων, που δεν είναι εκ των προτέρων γνωστά και η συμβολή στην βέλτιστη κατανόηση των πηγαίων δεδομένων. Τέλος, στο κεφάλαιο αυτό περιγράφονται τα αποτελέσματα και τα κεντρικά σημεία της ανάλυσης που θα χρησιμοποιηθούν στην υλοποίηση της Data Storytelling εφαρμογής.

Στο τέταρτο κεφάλαιο παρουσιάζεται η διαδικασία της μετατροπής των αποτελεσμάτων της ανάλυσης σε μια διαδραστική εφαρμογή. Το κεφάλαιο αυτό επικεντρώνεται στην δημιουργία ενός περιβάλλοντος που βασίζεται στα κεντρικά στοιχεία της έννοιας του Data Storytelling. Στόχος αποτελεί η παρουσίαση των συμπερασμάτων που έχουν προκύψει από την ανάλυση με ένα αποτελεσματικό και όσο το δυνατόν πιο επικοινωνιακό τρόπο. Για το λόγο αυτό, χρησιμοποιούνται μια σειρά τεχνολογιών που συμβάλλουν στην ανάπτυξη της διάδρασης του χρήστη με την εφαρμογή και στην παρουσίαση των εξαχθέντων συμπερασμάτων με απλό και κατανοητό τρόπο. Αυτό επιτυγχάνεται με τη χρήση κατάλληλων γραφημάτων, με τη συνοδεία κειμένου και με την ανάπτυξη της παρουσίασης των αποτελεσμάτων με ένα γραμμικό τρόπο ώστε να διευκολύνει το χρήστη στην κατανόησή τους. Για τη δημιουργία των γραφημάτων στην παρούσα εφαρμογή χρησιμοποιούνται οι τεχνολογίες Leaflet, GoogleCharts, D3 Highcharter και στην υλοποίηση του συνόλου των διαδικασιών της εφαρμογής οι τεχνολογίες Postgresql και R shiny. Επιπλέον, η χρήση διαδραστικών γραφημάτων στοχεύει στην εμπλοκή του χρήστη και κυρίως στη δυνατότητα να πραγματοποιήσει δυναμικά μια βασική γεωγραφική και στατιστική ανάλυση στα δεδομένα.

Στο πέμπτο και τελευταίο κεφάλαιο της εργασίας αποτυπώνονται συνοπτικά τα συμπεράσματα και οι προοπτικές εξέλιξης και βελτιστοποίησης της προτεινόμενης αλλά και παρόμοιας θεματικής εφαρμογών.

Chapter 1

Story Telling and KDD Process

1.1 Introduction

The first chapter focuses on basic theory concepts of storytelling and especially data storytelling. This chapter also describes the overview of the methodology of extracting knowledge from data and presents a list of web technologies used for visualization of geographic data. Furthermore, it describes data mining techniques with an emphasis on specific classification methodologies and clustering techniques.

1.2 Story telling concepts

Storytelling is one of the oldest forms of art. Since the very beginning of mankind's history, storytelling has been the most powerful and communicative way to share information. This particular type of communication differs from reading and writing as in storytelling the result is adjusted according to the audience and ever more to teller's skills. This means that a fact that that has already happened, or told in case of imaginary stories, is being reproduced in a specific way by the teller for a specific audience. Thus, the story contains characteristics that improve the transmissibility and successfully share the message in a more efficient way (**Smith, 2015**).

However, the word "storytelling" is often used in many ways. This study defines the concept of storytelling as a way of transmitting a message in an entertaining and memorable manner.

According to the international literature, in order for an act to be characterized as a storytelling one, this act has certain features (**National Storytelling Network, 2017**). An act is characterized as a storytelling performance act if it:

- Presents a story that has a beginning, a middle and an end

- Uses words and actions that interact with the audience
- Encourages the active imagination of the listeners
- Communicates the message that the audience will come away with
- Involves the audience to the story
- Makes the audience wonder about the next parts of the story
- Imprints pictures on audience's minds
- Avoids too many complicated words and too much information
- Its content is illustrative and easily memorable

1.2.1 Data storytelling

Many definitions have been stated for the term of data storytelling. One among them is the definition that Howard Dresner published in July of 2015. According to Dresner's definition, data storytelling is described as a set of features within visualization tools that enable a more interactive experience with the data. Moreover, Dresner points that data storytelling is the next big thing in collaborative computing (**Rouse, 2015**).

In the age of information, people are facing an increasing shortage of time and an increasing range of choices. On top of that, people need to be informed in an efficient way. This means that in most cases the way that information is provided becomes even more important than the information itself. At this point is where the use of a storytelling approach is able to create the expected results. Storytelling can be both efficient and powerful when it contains all its critical features (**Kumar, 2014**).

It is a fact that data storytelling is more and more used when it comes to data visualization and information sharing in general. Journalists, designers, developers, consultants, sellers, scientists, marketeers and many other professionals benefit from the use of data storytelling as it is an efficient way to communicate the knowledge that is extracted from a set of data.

In general, as with storytelling, a successful data storytelling is characterized by specific features. It should contain the features mentioned in storytelling and furthermore a successful data storytelling should:

- Have beginning, middle, climax and conclusion
- Be simple and use labels
- Work with questions
- Contain interactive visualizations
- Provide context and directions

- Be narrative and as much close to a linear experience

Based on the above, a data storytelling application should not use purely reader-driven visualizations as used in dashboards and usual reports. In contrary, it should use interactive graphs and features that encourage the participation of the user. Also, its structure should contain parts as beginning, middle, climax and conclusion. The beginning is about the data sources and a short description of the data set. The middle and climax is about questions to be answered and pointing picks respectively. Finally, the conclusion is a short explanation of the results and contains the message that the story is about.

The more different parts of the brain are activated the more memorable and impactful a story is. This means that the content should avoid the extensive use of numbers as much as possible and replace them with simple and everyday words with clear meaning. Although, there are occasions where figures can speak for themselves and these kind of replacement is not necessary.

Data stories should not be treated as exploration tools. They are rather narrative experiences that provide context and direction, not just numbers and charts. The key point of a successful data storytelling visualization is to include all the components to allow the audience to read it in a linear story mode. The more linear an experience is, the more it feels like a story to the audience.

A successful data storytelling presentation is equivalent to its simplicity. Moreover, it should work with questions, labels on graphs and interactive visualizations at each segment of the story. Finally as with every story, also in data storytelling it is very important to know which are the key points of the story, what the knowledge extracted from the data and what is the audience needs to understand. Based on this, a detailed analysis on the data set is always a crucial prerequisite.

1.2.2 Analytics in data storytelling

In order to create a data storytelling, a data set should be initially analyzed and processed. The results of this process will be then used as input in the data storytelling. Based on the purpose of the analysis, the complexity and the type of the data set, the analysis could be anything from a simple statistical analysis to a more complex predictive analysis by using machine learning techniques. And that is what Data Analytics is all about.

Data Analytics is "the process of examining raw data with the purpose of finding patterns and drawing conclusions about that information by applying mining methodologies to derive insights" (**Monnappa, 2017**). Also, Data Analytics is defined as "the process of examining data sets in order to draw conclusions about the information they contain, increasingly with the aid of specialized systems and software" (**Rouse, 2017**). Either way, Data Analytics is a series of actions in order to get from raw data to meaningful conclusions. These technologies and techniques are widely used in various industries. They

play an important role on decision making process as they suggest actions and improvements on company's strategic plans based on the analysis results.

Data storytelling is closely connected with Data Analytics concepts. As mentioned in previous paragraphs a successful data storytelling, among other parameters, should also contain climax. The climax of a data storytelling is the moment when the major and most important results of the analysis are illustrated. There are cases where the results are given or can easily be found. However, there are cases where the results are not obvious and they are hidden in the data set. In these cases, the story can become extremely interesting and for sure innovative. Therefore, a good data storytelling also requires the knowledge of retrieving unexpected results from a set of data in order to emphasize and visualize these findings at the appropriate moments of the story flow.

The next paragraph describes the basic concepts regarding Data Analytics such as Data Mining, Knowledge Discovery Database process and visualization tools, with an emphasis on spatial data.

1.3 Discover knowledge from data

The need to analyze, process and extract knowledge from a large amount of data has been a critical subject for computer scientists and researchers since the early years of databases creation. However, in the last decade, the speed at which data is created and stored has increased exponentially and everything indicates that it will continue to grow. Every day almost 2.5 quintillion bytes of data are created (**Mardell, 2017**), 10% of which are structured data and related to geographic location.

The main reasons that this deluge of data is growing so fast are the development of efficient software and management systems that made available the storage and processing of large amount of data, the increase of global internet population, the increase of cloud-based services and platforms, the evolution of mobile technology and most of all the excessive use of the internet including social media applications in everyday life.

In all modern societies, the storage and maintenance of historical data and furthermore the knowledge extraction from databases has become a matter of great importance. This affects both the private and public sector, in each and every industry. Therefore, as technologies and methodologies for storing and processing large amount of data are constantly evolving, the need of extracting knowledge out of data becomes of great concern for data scientists.

1.3.1 KDD process

In order to become aware of the importance and the necessity of these fields, the basic terms and concepts regarding the processes of Knowledge Discovery in Databases (KDD) and Data Mining (DM) are initially analyzed in the next paragraphs.

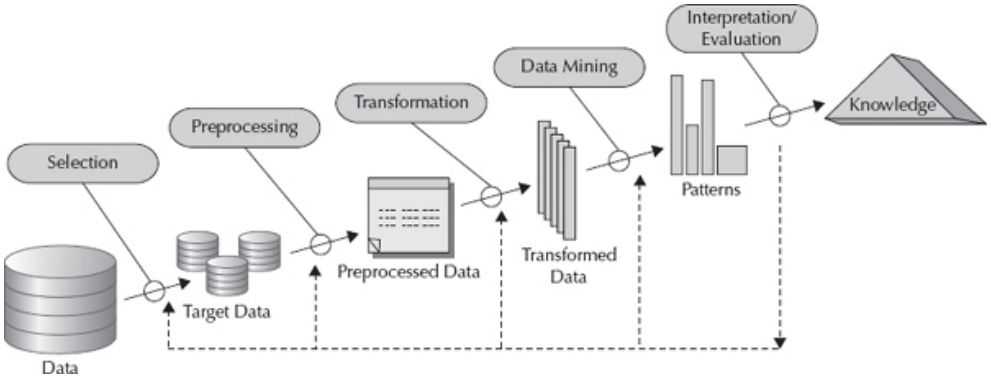


Figure 1.1: Steps of the KDD process (**Walker, 2015**)

Knowledge Discovery in Databases (KDD) is described as an automatic, exploratory analysis and modeling of large data repositories. Thus, KDD "is the organized process of identifying valid, novel, useful, and understandable patterns from large and complex data sets" (**Maimon & Rokach, 2005**).

In addition to KDD process, Data Mining (DM) as a part of KDD process is a term coined to describe the process of sifting through large databases for interesting patterns and relationships. Although contemporary researchers tend to identify DM with KDD process, DM is more than the core of the KDD process, "involving the inferring of algorithms that explore the data, develop the model for understanding phenomena from the data, analysis and prediction and discover previously unknown patterns" (**Maimon & Rokach, 2005**). Also, DM is referred to as the "non-trivial process of discovering interesting, implicit, and previously unknown knowledge from large databases" (**Han, Kamber & Tung, 200**1).

The KDD process consists of five distinct stages. Data selection (also known as Data Extraction), preprocessing, transformation, data mining and evaluation.

One step before the data evaluation is data mining. In this step, the transformed data are being processed using data mining techniques, as executing clustering algorithms, to search for existing patterns. Thus, DM can be an extremely complex process, particularly when this process applies to large databases or big data.

As an overview, KDD process has become extremely popular field in computer science in the past 10 years. The evolution of database management systems and data visualization software, their upcoming functionalities, the interconnectivity among them and the rapidly increasing amount of data coming from several sources have made Knowledge Discovery in Databases (KDD) and data mining methodologies more important than ever.

1.3.2 Data mining

As data mining tasks become more crucial day by day, data mining tools and data mining techniques are rapidly increasing. Currently, there have been developed a significant number of software that provide scientists and analysts with the appropriate tools to perform data mining tasks and apply mining algorithms. Some of the most frequently used technologies for data mining are programming languages such as R, python, Java, Scala and Julia. Also, desktop software are used for data mining activities. In the list with the more widely used desktop software are RapidMiner, KNIME, Weka, Gephi and GoeDa.

R

R stands for a language and environment used in statistical computing and graphics. Based on the S language and environment, R is considered as a different implementation of S. Even though certain important differences between the two languages or environments can be observed, the same code written for S runs respectively under R.

A wide range of statistical is offered by R as well as the former's graphical techniques. Some of them include linear and nonlinear modelling, classical statistical tests, time-series analysis, classification and clustering. Being an open source, R is highly extensible and a useful tool for the statistical methodology research and mining techniques.

R is engaged with a great ease mathematical symbols and formulas while it also participates in the production of well-designed publication-quality plots. R allows the user to be fully in charge although the defaults and the design choices offered in graphics are so well-designed that can be used without the user's involvement.

In order for R code to be executed, a user can compile and run it on multiple UNIX platforms and other equivalent systems, including FreeBSD, Linux, Windows and MacOS.

Python

Another significant data-mining tool is Python. Python is a strong programming language that can be easily learned. Its high-level data structures are quite effective whereas its approach to object-oriented programming is simple but efficient. Python can be perceived as the perfect language for scripting and high-speed application in many areas on the majority of platforms thanks to its elegant syntax, dynamic typing and interpreted nature.

All of the Python's features and library can be found and distributed for free on the Python Web site, https://www.python.org/. This site apart from Python's main components includes and freely distributes a great number of third party Python modules, programs and tools, as well as extra documentation.

The Python interpreter can be easily expanded with the addition of new functions and data types implemented in C or C++.

Weka

Weka is an open source data mining software. Its most crucial function is to collect machine learning algorithms employed for data mining tasks. The algorithms can either be applied directly to a dataset or called from your Java code.

The tools entailed in Weca can perform data pre-processing, classification, regression, clustering, association rules, and visualization.

Weka is also suitable for creating new machine learning schemes and it is an open source software issued under the GNU General Public License.

RapidMiner

RapidMiner Studio through its well-designed environment and convenient usage has the power and potential to quickly complete any predictive analytic procedure. This fully equipped tool supplies users with various pre-defined data preparation and machine learning algorithms that support all of the former's data science projects. As with the tools described in previous paragraphs, RapidMiner can perform data pre-processing, classification, regression, clustering, association rules and visualization tasks.

KNIME

KNIME is also an open source software that serves data analysis, reporting and integration processes through its modular data pipelining concept. All the KNIME procedures mentioned above can be easily accomplished by the final user because of its easily operated graphical environment.

Although written in Java and based on Eclipse KNIME can evolve its functionality engaging extensions to add plugins.

KNIME effectively cooperates with many other open-source projects such as machine learning algorithms from Weka and the statistics package R project.

Gephi

Gephi is a dynamic platform through which networks and complex systems are depicted and explored. It is a dynamic platform that depicts and explores networks and complex systems as well as hierarchical charts.

It is an open source and through 3D render engine it accelerates the exploration process and displays the results in real time. Using Gephi, the user can perform clustering activities, spatial separation, analysis, and visualizations with great clarity and a wide variety of graphs.

GeoDa

Last but not least, GeoDa is a free software for usability-friendly spatial data analysis. Its purpose, since its creation in 2003, is to translate spatial data into information. The program manages data about the location and this is which differentiates it from other data analysis tools. With the usage of dynamically linked windows it combines maps with statistical analyses and graphs.

1.3.3 Spatial data mining

Many definitions have been stated in literature for the term of spatial data. One of the simpler definition of spatial data, describes spatial data as "information related to the space occupied by objects" (**Kolatch, 2001**). Moreover, spatial data can be defined as any structured or unstructured data that refers to a specific location of a certain area. The area could be a two-dimensional or a multidimensional space, as for example the surface of the earth or an imaginary multidimensional space.

In data science and computer science, spatial data differ from ordinary data. Spatial data are stored in databases with spatial extension. In this way, they use specific data types (point, polygon, line, geometry collection), formats and functionalities, according to the capabilities of each database management system. Thus, Spatial Data Mining (SDM) methods differ from those used in mining regular data. SDM is defined as the process of extracting knowledge, spatial relationships and previously unknown patterns from spatial data.

SDM techniques can be classified into two main categories, the descriptive data mining techniques and the predictive data mining techniques. Furthermore, the basic tasks proposed for SDM include: (a) classification, (b) association rules, (c) characteristics rules, (d) discriminant rules, (e) clustering and (f) trend detection (**Kumar, C. N. S., Ramulu, Reddy, Kotha, & Kumar, C. M., 2012; Sumathi, Geetha & Bama, 2008**).

Classification

Classification is a technique where each object of a data collection is assigned a class, according to a set of rules that determine every class. The number of classes are predefined. The classification rules are based in a certain set of attributes. The scope of a classification algorithm is to create the set of rules that will determine the class of an object. Classification is considered as a predictive supervised data mining method, as it first creates a model according to which the whole dataset is analyzed and the number of classes are predefined (**Sumathi, Geetha & Bama, 2008**).

Association rules

Association rules is a data mining technique where given a collection of objects and their occurrences, creates the rules that will predict the occurrence of an item based on the occurrences of other objects in the collection. The

scope of association rules is to find patterns, which often exist in the database. Association rules in SDM finds rules from the database that are spatially related. Types of spatial relationships includes topological and distance relations among spatial objects combined with logical operators (**Bembenik & Protaziuk, 2004**). As for example: object A completely encloses object B, object A is within a radius of 10 meters from object B, object A comes into contact with the boundary of object B.

Characteristics rules

Characteristics rules, also called as data characterization, is a summarization of general features of objects in a target class and creation of a set of rules for each class. In the case of a spatial database, this technique "takes into account not only of the properties of objects, but also of the properties of their neighborhood up to a given level" (**Kumar, C. N. S., Ramulu, Reddy, Kotha, & Kumar, C. M., 2012**). A step further, discriminant rules is a technique that describes differences between two parts of a database, as for example in spatial data to find differences between cities with high and low unemployment rate (**Kumar, C. N. S., Ramulu, Reddy, Kotha, & Kumar, C. M., 2012**). Data discrimination is also used in characterization in order to identify the rules that will partition the database into classes.

Trend detection

Trend detection is a technique that finds a temporal pattern in time series data. In case of spatial data, trend detection is used for finding patterns of the changes of non-spatial attributes with respect to the spatial relationship among the data, as for example the distance.

Clustering

Last, clustering is the technique of grouping a set of objects into classes, also called clusters, so that each class contains objects with high similarity to one another and as high dissimilarity as possible to objects of other classes. In general, spatial clustering algorithms, based on cluster definition techniques, can be separated into four main categories: (a) partitional clustering algorithms, (b) hierarchical clustering algorithms, (c) density-based clustering algorithms and (d) grid-based clustering algorithms.

Other related work (**El-Zawawy, 2012; Padmavati, 2013; Swathi & Rajesh, 2012**) classify spatial clustering algorithms into more categories, which include the previous four and the addition of the model-based clustering algorithms, the constraint-based clustering algorithms and the locality-based clustering algorithms. Most of clustering algorithms, especially those that have recently proposed, are a combination of more than one methods.

However, the majority of papers and academic work related to the subject

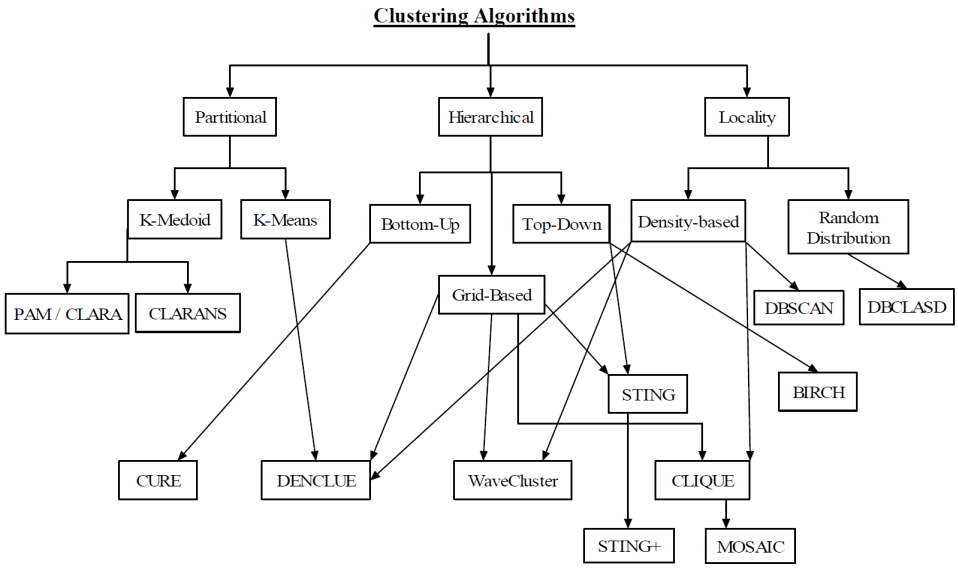


Figure 1.2: Categorization of clustering algorithms (**Kolatch, 2001**)

of spatial clustering refer to the four general categories as the main categorization of spatial clustering algorithms.

The partitioning method is an approach of clustering where a set of n spatial objects is grouped into k clusters based on optimizing an objective criterion or else called a similarity function, which defines the level of similarity among the points. In most cases, the similarity function between two points corresponds to the distance between them. Partitioning was the earliest approach that appeared in spatial clustering and thus is still one of the most cited approach in literature (**Varlaro, 2008**). Most commonly used partitional clustering algorithms include k-means, k-medoids, PAM and CLARA.

The hierarchical clustering algorithms organize a tree data structure of the clusters, also called dendrogram, using an hierarchy structure. In this tree, "the root is considered as a single cluster which involves all the spatial objects in the spatial area whereas the nodes are considered as clusters with only one object. These algorithms operate repeatedly to achieve merging or splitting until a stopping condition is satisfied or the clustering process encompassed all objects" (**Otair, 2013**). Hierarchical algorithms are classified into divisive and agglomerative algorithms. In divisive algorithms the decomposition of the tree is formed in a top-down approach, from root to leaves, whereas in agglomerative algorithms (Ward's method) the decomposition is formed the opposite way, which is a bottom-up approach (**El-Zawawy, 2012**). Clustering algorithms that belong to this category are BIRCH, CURE, STING and DIANA.

In the Density-based method, clusters are defined based on a mechanism of density-connected points. Clusters are defined as dense regions of objects that are separated by low density regions in the data space (**Otair, 2013**). Al-

Algorithm	*Input Parameters*	*Optimized For*	*Cluster Structure*	*Outlier Handling*	*Computational Complexity*
Partitional Methods					
$k-means$	Number of Clusters	Separated Clusters	Spherical	No	$\mathcal{O}(Ikn)$
PAM	Number of Clusters	Separated Clusters, Small Data Sets	Spherical	No	$\mathcal{O}(Ik(n-k)^2)$
$CLARA$	Number of Clusters	Relatively Large Data Sets	Spherical	No	$\mathcal{O}(ks^2+k(n-k))$
$CLARANS$	Number of Clusters, Maximum Number of Neighbors	Spatial Data Sets, Better Quality of Clusters than *PAM* and *CLARA*	Spherical	No	$\mathcal{O}(kn^2)$
Hierarchical Methods					
$BIRCH$	Branching Factor, Diameter Threshold	Large Data Sets	Spherical	Yes	$\mathcal{O}(n)$
$CURE$	Number of Clusters, Number of Cluster Representatives	Arbitrary Shapes of Clusters, Relatively Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n^2 \log n)$
Density-Based Methods					
$DBSCAN$	Radius of Clusters, Minimum Number of Points in Clusters	Arbitrary Shapes of Clusters, Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n \log n)$
$DENCLUE$	Radius of Clusters, Minimum Number of objects	Arbitrary Shapes of Clusters, Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n \log n)$
$OPTICS$	Radius of Clusters (min,max), Minimum Number of objects	Arbitrary Shapes of Clusters, Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n \log n)$
Miscellaneous Methods					
$STING$	Number of cells in lowest level, Number of objects in cell	Large Spatial Data Sets	Vertical and Horizontal Boundaries	Yes	$\mathcal{O}(n)$
$WaveCluster$	Number of Cells for each Dimension, Wavelet, Number of application of Transform	Arbitrary Shapes of Clusters, Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n)$
$CLIQUE$	Size of the Grid , Minimum Number of Points within each Cell	High Dimensional Large Data Sets	Arbitrary	Yes	$\mathcal{O}(n)$
$ScalableEM$	Initial Gaussian Parameters, Convergence Limit	Large Data Sets with Approximately Uniform Distribution	Spherical	No (?)	$\mathcal{O}(n)$

n=number of objects, k=number of clusters, s=size of sample, I=number of iterations

Figure 1.3: Clustering algorithms characteristics (**Andritsos, 2002**)

gorithms belonging to density-based category of spatial clustering algorithms have the ability to detect noise and clusters or arbitrary styles and shapes. The most known method belonging to density-based clustering is DBSCAN.

In the grid-based clustering algorithms, data objects are replaced with a number of cells that that represent the elements that will be clustered. Cells are defined with a specific grid size and form the grid structure. Each cell includes a number of data objects and summarized information, as for example the total number of objects within the cell. The operations of the algorithm for clustering are performed on the cells rather than the whole dataset. In this way the process becomes faster than in other clustering methods (**El-Zawawy, 2012;Otair, 2013**). Among grid-based clustering algorithms are CLIQUE, WAVECLUSTER and DENCLUE.

Spatial clustering algorithms

Spatial clustering is a field with many prospects. Several algorithms have been introduced to literature the last 10 years. Clustering methods usage depends on their complexity, the amount of data, the purpose of clustering and the predefined parameters.

K-means, DBSCAN and Ward's method are among the most used clustering algorithms for spatial data.

K-means

K-means belongs to partitioning spatial clustering algorithms. It is a frequently used clustering method and it is one of the simplest unsupervised

learning algorithms. K-meas define clusters by partitioning all observations into groups in which each observation belongs to the group with the nearest mean. The algorithms operates in iterations until the sum of squares from points to the assigned cluster centres is minimized. The end result of k-means algorithm is the partitioning of the data space into Voronoi cells.

In general, the algorithm aims to minimize an objective function where the key paramenter is the distance between each data point and its cluster center (**Varghese & Unnikrishnan, 2013**).

K-means, in its standard version, uses one input parameter that is the number of the expected clusters. The cluster centers, which are initially defined, are selected randomly among all points. Thus, k-means may produce different results in every execution.

Ward's method

Ward's method belongs to hierarchical spatial clustering methods.This method involves an agglomerative clustering algorithm.

The algorithm starts from the leaves and works up to the root, or in other words operates in a bottom-up approach. Ward's method starts with n clusters of size 1, where n is the total number of observations or total number of points, and continues until all the observations are included into one or the preferable number of clusters.

Ward's algorithm uses one input parameter that is the number of the expected clusters.

DBSCAN

DBSCAN it is a commonly used clustering algorithm. It belongs to density based spatial clustering methods. DBSCAN is a fundamental data clustering technique for finding arbitrary shape clusters and for detecting outliers. DBSCAN defines clusters depending on density where density is defined as the number of points within a certain distance of each other (**Varghese & Unnikrishnan, 2013**).

Moreover, DBSCAN defines clusters depending on density reachability and density connectivity among data points. Density reachability depends on the distance (Eps) between point A from point B and the number of points in point's A neighbors which are within this distance. Density connectivity depends on the existence of a point A which has sufficient number of points in its neighbors and both the points A and B are within the Eps distance.

DBSCAN uses two parameters, Eps and MinPts to control the density of the cluster. Minpts, indicates the minimum number of data points in a neighborhood to define a cluster and Eps indicates the radius of the neighborhoods around a data point.

1.3.4 Visualize results

Data visualization, especially when it comes to big data and even more to real time data, becomes a very challenging task. Thus, there have been developed numerous of tools for data visualization.

This parer focus on tools that meet two specific requirements: (a) they are web based, which means that they use web technologies (javascript, html, css) and they are designed to operate in browsers and (b) they are compatible with geospatial data, which means that they contain features that support the visualization of geospatial data (spatial datatypes, geographic coordinate systems, projections).

This study presents a list of 38 web based javascript libraries for geospatial data and describes briefly the top 5 in terms of developer's usage and popularity among them.

Leaflet

Leaflet is the most commonly used open-source javaScript library. It is built for the development of mobile-friendly interactive maps and it widely used from developers due to its small capacity. It is a cross platform tool and it can be easily extended with plugins and features.

OpenLayers

OpenLayers is the second most popular JavaScript library for developing maps and visualizing geospatial data on the web. As with Leaflet,OpenLayers can be implemented easily and is able to display all kinds of spatial data types like vectors, tiles and markers. OpenLayers is an open-source javaScript library and has a large community of support.

D3

D3 is also a javaScript library with many capabilities that are supported from all modern browsers. D3 is a more comprehensive library that contains many features for data visualization. Among them, it contains features for geospatial data visualization and web maps in general. It is a very promising tool, with many innovations and can be easily integrated with other systems.

Polymaps

Polymaps is another popular free javaScript library for making dynamic and interactive maps in the web. Polymaps provides advanced features and supports a variety of visual presentations for the most types of spatial data. It uses SVG as well as tiles to display spatial information and is one of the most used among developer's community.

Raphaël

Raphaël is one of the smallest but also one of the most widely used javaScript libraries. It is supported from all modern browsers and works with vector graphics on the web. This means that, as with Polymaps, its configuration and development can be easily achieved.

Tools for geospatial data visualization		
#	Name	Type
1	Leaflet	JavaScript Library
2	OpenLayers	JavaScript Library
3	MapBox	Developer Platform
4	Google Maps	Developer Platform
5	Modest Maps	JavaScript Library
6	Polymaps	JavaScript Library
7	D3	JavaScript Library
8	DataMaps	JavaScript Library
9	Raphaël	JavaScript Library
10	jVectorMap	JavaScript Library
11	JQVMap	jQuery plugin
12	GeoChart	JavaScript Library
13	HERE JavaScript API	JavaScript Library API
14	MapQuest	JavaScript Library API
15	Bing Maps	JavaScript Library API
16	AmMap	JavaScript Library
17	BGeoPrisma	Web Map Application
18	GeoExt	JavaScript Library
19	Mapstraction	JavaScript Library
20	gmaps.js	google maps api
21	Kartograph	JavaScript library
22	ArcGIS API for JavaScript	JavaScript library
23	API Javascript ViaMichelin	JavaScript library
24	Geo5	JavaScript library
25	Cesium	JavaScript library
26	WebGL Earth	JavaScript library
27	OSM Buildings	JavaScript library
28	CSSMap	JavaScript library
29	jHERE	HERE Maps API
30	jump	jQuery map plugin
31	jQuery Geo	JavaScript library
32	jQuery Mapael	JavaScript library
33	initmap.js	jQuery plugin
34	iMapBuilder	Web Mapping Application
35	heatmap.js	JavaScript library
36	Cartographer	library for Google Maps
37	Carto	Web Map Application
38	Mango	Web Map Application

Table 1.1: Web based tools for visualization of spatial data

Chapter 2

Data Storytelling Application: Data collection

2.1 Introduction

This chapter introduces and describes the general architecture of building an interactive data storytelling application. It focuses on the first steps that include data preparation and data processing. In particular, this chapter describes the data used for the implementation of the web application and the way they were processed in order to create a clean and meaningful input for next step, the data analysis.

2.2 Process overview

The implementation of a data story application is separated in 6 main steps:

- Step 1 - Define sources and select data.
- Step 2 - Process data.
- Step 3 - Analyze data.
- Step 4 - Design the flow of the storytelling structure.
- Step 5 - Develop the application.
- Step 6 - Implement and deploy the application.

The first step is about identifying the data sources and design the way to retrieve or extract the subset of data needed for the story. This is a more technical process as it contains tasks regarding the connection to the selected data source, the maintenance and storage of the extracted data. The connection could be a either a real time process or a single batch process that collects data from source usually over a period of time. The second step is about data transformations and creation of a data model that will help in the next step of

data analysis. Data analysis is the most interesting part of the whole process. The objective of this step is to discover unknown patterns, understand trends and gain knowledge from the data. The results of this step will be the input and the key elements of the story. Next step is the design of the story. In this step, all knowledge gained from data analysis is presented in a meaningful and communicative way. The last two steps is about the development and the implementation of the application. In these steps, a variety of web technologies are listed and selected in order to achieve the best possible results.

2.3 Data sources

The data used in this study are collected from three main sources:

- The World Bank, http://databank.worldbank.org
- Google DataSet Publishing Language, https://developers.google.com/publicdata/
- An Aquaculture Company Group located in Greece

2.3.1 The World Bank

Data from the World Bank refers to the period between 1.1.2013 and 31.12.2016 and concerns two global indicators, total population and GDP per capita per country.

Total population is based on the de facto definition of population, which counts all residents regardless of legal status or citizenship. The values used are midyear estimates and they are derived from:

- United Nations Population Division. World Population Prospects
- Census reports and other statistical publications from national statistical offices
- Eurostat: Demographic Statistics
- United Nations Statistical Division. Population and Vital Statistics Report (various years)
- U.S. Census Bureau: International Database
- Secretariat of the Pacific Community: Statistics and Demography Programme

Sources were last updated on 2.8.2017.

code charac	pop2013 double precision	pop2014 double precision	pop2015 double precision	pop2016 double precision
KWT	3598385	3782450	3935794	4052584
CAN	35155451	35544564	35848610	36286425
ALB	2895092	2889104	2880703	2876101
ARE	9006263	9070867	9154302	9269612
AUT	8479375	8541575	8633169	8747358
BEL	11182817	11209057	11274196	11348159
BGR	7265115	7223938	7177991	7127822
BLR	9465997	9474511	9489616	9507120
CHE	8089346	8188649	8282396	8372098
CYP	1143896	1152309	1160985	1170125
CZE	10514272	10525347	10546059	10561633
DEU	80645605	80982500	81686611	82667685
DNK	5614932	5643475	5683483	5731118
ESP	46620045	46480882	46447697	46443959
EST	1317997	1314545	1315407	1316481

Figure 2.1: Sample data: Total population per country

GDP per capita is gross domestic product divided by midyear population. GDP is the sum of gross value added by all resident producers in the economy plus any product taxes and minus any subsidies not included in the value of the products. It is calculated without making deductions for depreciation of fabricated assets or for depletion and degradation of natural resources. Values are derived from World Bank national accounts data and OECD National Accounts data files.

Sources were last updated on 2.8.2017.

code2 charact	gdp2013 double precision	gdp2014 double precision	gdp2015 double precision	gdp2016 double precisio
KWT	48399.90581	42996.40812	28975.40108	40668.82
CAN	52413.72116	50440.43376	43315.70044	42157.92799
ALB	4414.72314	4575.763787	3954.022783	4146.89625
ARE	43350.64268	44449.74035	39101.74689	37622.20746
AUT	50504.71532	51322.63997	43665.00947	44176.51522
BEL	46510.38647	47439.39684	40356.875	41096.1573
BGR	7674.860559	7853.335191	6993.47736	7350.795801
BLR	7978.825443	8318.429294	5949.110677	4989.254611
CHE	84658.88768	85814.58857	80989.84024	78812.65069
CYP	27907.96736	27340.88382	23075.1127	23324.20174
CZE	19916.01939	19744.55861	17556.9243	18266.54969
DEU	46530.91143	47902.65288	41176.88158	41936.05858
DNK	61191.19263	62425.5392	53014.64416	53417.66428
ESP	29210.09342	29600.47225	25683.84565	26528.49179
EST	19029.7746	19941.45532	17074.92091	17574.68736

Figure 2.2: Sample data: GDP per capita per country ($)

2.3.2 Google DataSet Publishing Language

Google DataSet Publishing Language (DSPL) is a data and metadata format designed from Google to support powerful and interactive visualizations.

From DSPL is derived data regarding the coordinates of each country. The coordinates are essential for the visualization of geographic data and for further geographic analysis.

	latitude double precision	longtitude double precision	countryname2 character varying(256)	country2 character varying(256)
1	29.31166	47.481766	Kuwait	Kuwait
2	56.130366	-106.346771	Canada	Canada
3	41.153332	20.168331	Albania	Albania
4	23.424076	53.847818	United Arab Emirates	United Arab Emirates
5	47.516231	14.550072	Austria	Austria
6	50.503887	4.469936	Belgium	Belgium
7	42.733883	25.48583	Bulgaria	Bulgaria
8	53.709807	27.953389	Belarus	REPUBLIC OF BELARUS
9	46.818188	8.227512	Switzerland	Switzerland
10	35.126413	33.429859	Cyprus	Cyprus
11	49.817492	15.472962	Czech Republic	Czech Republic
12	51.165691	10.451526	Germany	Germany
13	56.26392	9.501785	Denmark	Denmark
14	40.463667	-3.74922	Spain	Spain
15	58.595272	25.013607	Estonia	Estonia

Figure 2.3: Sample data: Coordinates per country

2.3.3 Aquaculture Company

The central point of this story is the Marineculture industry. For this purpose, data has been collected from an Aquaculture Company Group located in Greece and refer to information about the orders of the company, the amount of retail sales and the quantity of fish, in the period between 1.1.2013 and 22.8.2017. The subset data was derived from a relational database schema hosted in an MS SQL Server.

Sources were last updated on 24.9.2017.

country character va	itemline text	itemcategory text	itemgroup text	itemsize character vary	quantity double pre	sales double p	issuedate date
GREECE	Fresh	Whole	Seabream	2000+	0	0	2013-01-01
GREECE	Fresh	Whole	Seabream	2000+	0	0	2014-01-01
GREECE	Fresh	Whole	Seabream	2000+	0	0	2014-01-01
GREECE	Fresh	Whole	Seabream	SPECIALCUTS	0	0	2015-01-01
GREECE	Fresh	Whole	Seabream	300-600	0	0	2016-01-01
BELGIUM	Fresh	Whole	Meagre	4000+	35	277	2014-06-27
GREECE	Fresh	Whole	Meagre	4000+	25	25	2017-01-21
ITALY	Fresh	Whole	Seabream	STORTI	220	726	2017-07-14
SPAIN	Fresh	Whole	Meagre	4000+	296	1656	2017-06-23
USA	Fresh	Whole	Seabass	600-800	3150	25214	2017-03-20
ROMANIA	Fresh	Fillet	Seabream	400-600	90	828	2016-12-21
ROMANIA	Fresh	Fillet	Seabream	400-600	6	70	2017-03-01
ROMANIA	Fresh	Fillet	Seabream	400-600	6	64	2017-03-11
ROMANIA	Fresh	Fillet	Seabream	400-600	6	71	2017-07-15
ROMANIA	Fresh	Fillet	Seabream	600-800	6	71	2017-03-04
ROMANIA	Fresh	Fillet	Seabream	600-800	12	142	2017-03-08
ROMANIA	Fresh	Fillet	Seabream	600-800	6	71	2017-03-11
ROMANIA	Fresh	Fillet	Seabream	600-800	12	142	2017-03-15
ROMANIA	Fresh	Fillet	Seabream	600-800	30	354	2017-03-18
ROMANIA	Fresh	Fillet	Seabream	600-800	6	71	2017-03-22
RUSSIA	Frozen	Whole	Seabass	400-600	3439	18056	2013-02-22
RUSSIA	Frozen	Whole	Seabass	400-600	1132	6850	2013-07-11
RUSSIA	Frozen	Whole	Seabass	400-600	1243	7521	2013-10-07

Figure 2.4: Sample data: Orders from Aquaculture Company Group

2.4 Data processing

Data processing is about transforming and creating a data model that will help in the next steps. Thus, data from all sources are integrated into a single location. For this reason, a PostgreSQL relational Database is used to store and transform the extracted data.

The objective of this step is to clean the data by handing accordingly null values and corrupted records, as well as make appropriate transformations in data types and values to create a useful output. The final output should contain all the information needed for the analysis.

Thus, in this study, data from each source is imported in a relational database created in PostgreSQL. As a next step, new variables are created to indicate the time, the day, the location and the type of every transaction. As a transaction is characterized each record from the final table that corresponds to a single purchase order. The data denormalization is accomplished by creating keys and relationships among all initial tables and then aggregating data in one single table.

In conclusion, as a final output from this step, a denormalized table is

created that contains the following information per transaction:

- country, the destination of the order
- lon, the longitude of the destination
- lat, the latitude of the destination
- line, whether the order refers to frozen or fresh fish
- category, the level of processed fish (Whole, Fillet, Gutted)
- product, the type of fish (Seabass, Seabream, Meagre, Red Seabream)
- size, the group of sizes of the fish (600-800, 800-1000 etc)
- date, the date of the order (2017-03-18 etc)
- day, the weekday of the order (Friday, Monday etc)
- year, the year of the order (2017, 2016 etc)
- month, the month of the order (12, 09 etc)
- quantity, the quantity of the ordered fish measured in kgrs
- sales, the cost of the order measured in euros

The total number of rows in the final table equals to 2,476,350 records that corresponds to almost 138MB of size in the database.

Chapter 3

Data Storytelling Application: Data Analysis

3.1 Introduction

This chapter illustrates the data analysis step. It describes statistical methods and in general mining techniques applied in order to extract knowledge from data. This case study analysis uses descriptive statistics, probability functions, classification and clustering methodologies as well as predictive analysis.

Data analysis is one of the most interesting parts of the whole process. In general, data analysis is an extremely challenging and demanding process that requires expertise in various fields of computer science.

As an overview, the objective of this step is to discover unknown patterns, understand trends and gain knowledge from the data in order to use them in the storytelling application. This chapter presents the findings of the analysis and describes the methodologies used.

3.2 Methodologies

Data analysis includes numerous methodologies for mining and advanced analysis of data. In this case, the methodologies used are the following:

- network analysis
- descriptive statistics
- time series analysis
- geographic analysis
- equal interval classification
- customized clustering
- predictive analysis

3.2.1 Network analysis

As mentioned in the previous chapter, the input to this process is a denormalized table. Denormalization is a technique used on a previously-normalized database in order to gather all necessary information in one or more tables. In this case, all information is grouped into one table.

Although this structure is very comfortable when applying calculations on data, it is quite uncomfortable to understand the business model behind the data. As for example, in this case, it is not very obvious which products can be sold fresh, or which categories are included under fresh or frozen line, or which sizes may exist in more than one product etc. Thus, a basic analysis that can be applied on the data is a network analysis. A network analysis is able to answer these questions and furthermore illustrate connections among data that will help understand the business model. Actually, network analysis corresponds to the visualization of a collection of interconnected components.

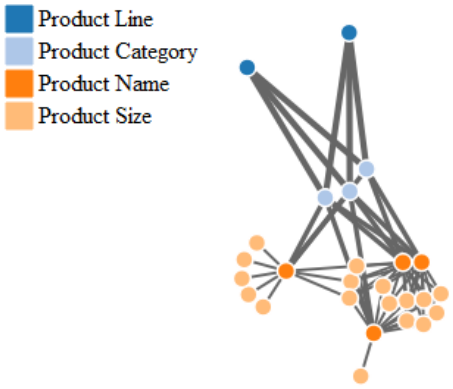


Figure 3.1: Network analysis

As a result, using a network visualization for fields line, category, name and size from the denormalized table, it becomes more readily understandable the possible path for each type of product inside the product line of the company.

It is more clear that the production is separated in two main production lines: the Fresh and the Frozen fish. Moreover, the fish are separated in three categories: (a) Gutted, (b) Fillet and (c) Whole. Its category contains its products and every product its own sizes. Same products may exist in more than one categories as showed in the network. The company produces 4 types of fish (Red Seabream, Seabass, Seabream and Meagre) and 17 distinct sizes in total (0-200, 200-300, 300-400, 300-600 etc).

3.2.2 Statistical analysis

The next paragraphs illustrate the results of the statistical analysis. Statistical analysis is applied in order to get minimum and maximum values per variable

of the data set, calculate frequencies, cumulative frequencies and descriptive statistics, investigate distributions and analyze time series.

Analysis of total sales

As a first result, the following graphs and tables show that Italy, France, Greece, Portugal and Spain are distinguished from the rest countries in terms of total sales.

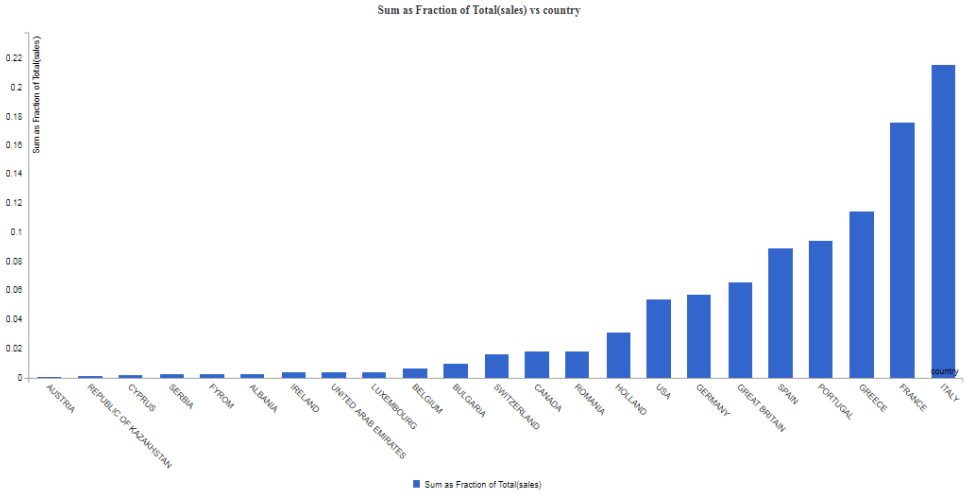


Figure 3.2: Percentage of total sales per country

country \ issueyear	2013	2014	2015	2016	2017	Totals
ITALY	4.2%	4.5%	4.7%	4.6%	3.0%	**21.1%**
FRANCE	3.5%	3.4%	3.6%	3.8%	2.9%	**17.2%**
GREECE	1.6%	1.9%	2.5%	3.0%	2.1%	**11.2%**
PORTUGAL	2.2%	1.7%	2.2%	1.7%	1.3%	**9.2%**
SPAIN	1.3%	1.6%	1.9%	2.2%	1.7%	**8.7%**
GREAT BRITAIN	1.4%	1.3%	1.4%	1.5%	0.8%	**6.4%**
GERMANY	0.9%	1.2%	1.4%	1.2%	0.9%	**5.6%**
USA	1.0%	1.1%	1.1%	1.3%	0.8%	**5.3%**
HOLLAND	0.7%	0.6%	0.7%	0.7%	0.5%	**3.1%**
ROMANIA	0.2%	0.3%	0.4%	0.5%	0.4%	**1.8%**
CANADA	0.3%	0.3%	0.4%	0.5%	0.4%	**1.8%**
SWITZERLAND	0.3%	0.2%	0.4%	0.4%	0.3%	**1.6%**
RUSSIA	0.8%	0.2%				**1.1%**
BULGARIA	0.1%	0.2%	0.2%	0.3%	0.2%	**1.0%**
BELGIUM	0.2%	0.2%	0.2%	0.1%	0.1%	**0.7%**
LUXEMBOURG	0.1%	0.1%	0.1%	0.1%	0.0%	**0.4%**

Figure 3.3: Percentage of total sales per country per year

issueyear	2013	2014	2015	2016	2017	Totals
Totals	141,988,823.00	141,387,730.00	155,000,178.00	161,340,071.00	111,896,855.00	711,613,657.00

Figure 3.4: Total sales per year (euro) (1)

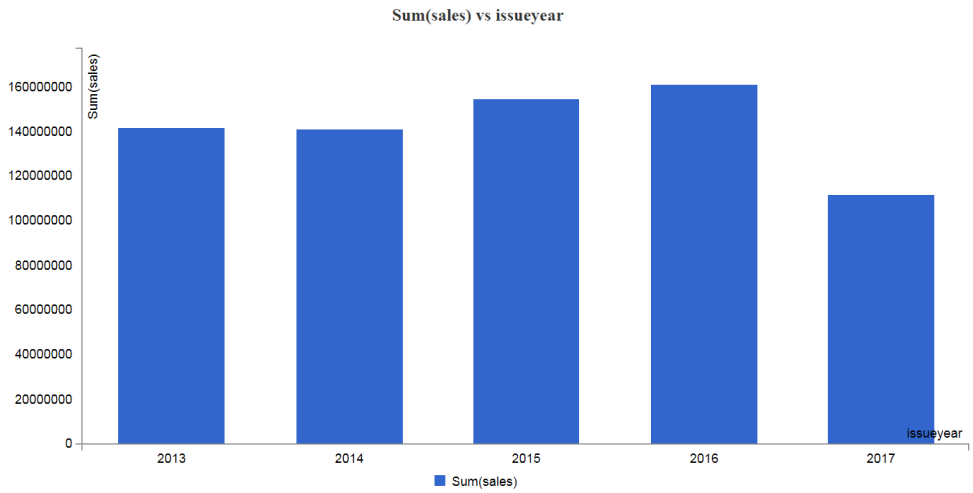


Figure 3.5: Total sales per year (euro) (2)

issueyear	2013	2014	2015	2016	2017	Totals
Totals	28,386,224.00	25,890,567.00	26,473,916.00	28,406,980.00	20,073,609.00	129,231,296.00

Figure 3.6: Total quantity per year (kgrs) (1)

A further analysis of the data shows that the total quantity of sales for the period of study (01.01.2013-22.08.2017) is equal to 129,231.296 tonnes of fish and that only a share of 12.2% of this quantity is sold domestically. The rest is exported to the rest 37 countries. Therefore, an important result is that 90% of total sales is derived from exports. Moreover, the largest importers by volume are the countries of Italy, France, Portugal and Spain.

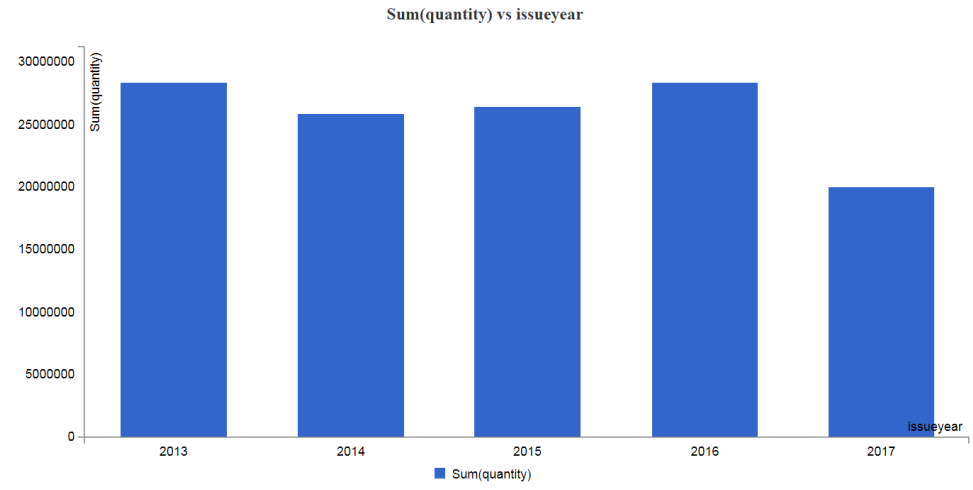


Figure 3.7: Total quantity per year (kgrs) (2)

Analyze metrics per month

Another result showed that sales between May and September reach an average of 44.62% of the total annual sales. Also, July is the month with the largest number of sales with an average of 14 millions while in February sales are in their lowest point with an average of 10.460 millions. Moreover, in July the company produces the largest quantity, almost an average of 2,500 tonnes. The month with the lowest amount of volume is February and the average quantity of production for the same month is 1,924 tonnes.

Also, in July of 2016 the total sales of the company reached the number of 15,738 million euros. This is the largest number in sales per month, from January of 2013 until August of 2017, although the maximum average price per month is depicted in May.

The following graphs illustrate the total sales, quantity and average selling price per month.

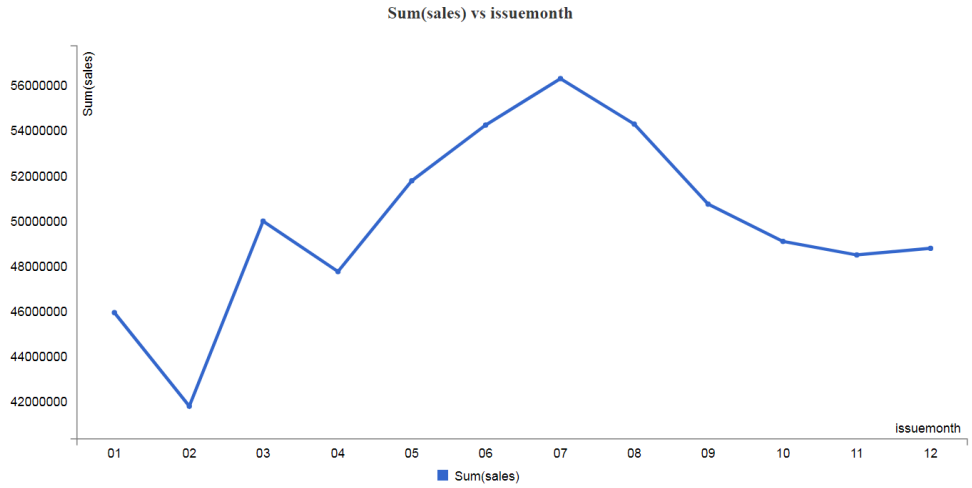


Figure 3.8: Total sales per month (euro)

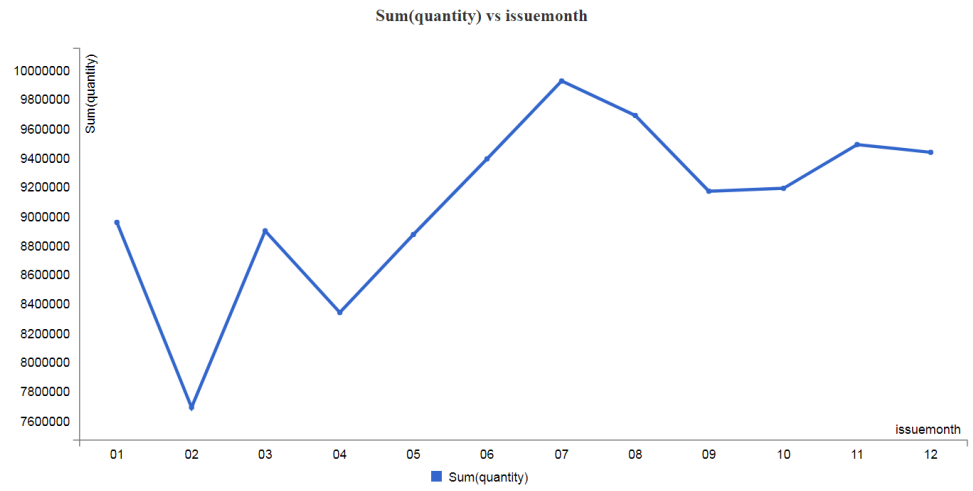


Figure 3.9: Total quantity per month (kgrs)

The selling price depends on the type, the size, the product line - meaning whether it is processed or not -, the consignor country and the date of order.

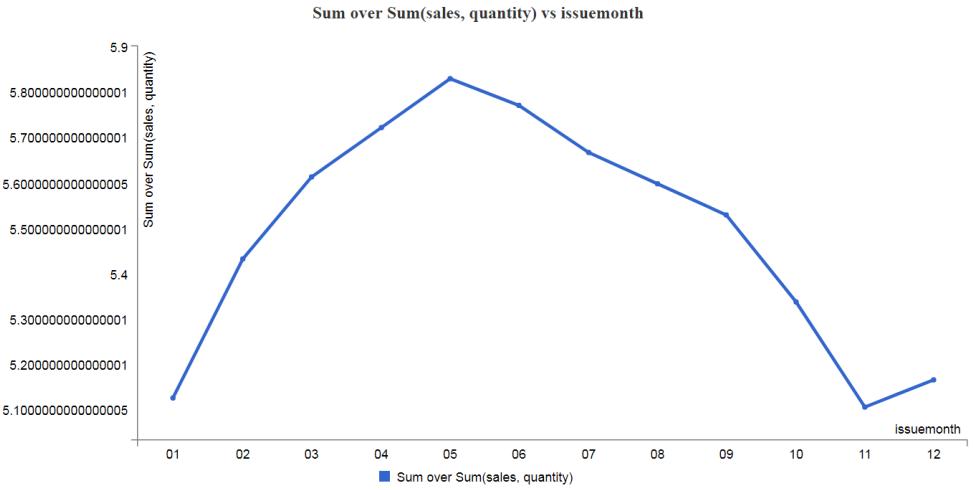


Figure 3.10: Average selling price per month (euro/kgrs)

Analysis of sizes

Statistical analysis on sizes shows that smaller sizes are selling more than larger ones. In addition, 2 out of 17 sizes are the holders of 73.3% of total sales between January of 2013 and December of 2016. These sizes are (300-400) and (400-600). In the third place with 10.2% of total sales is the size of (600-800) and in the forth place with 6.9% is the size of (200-300), which is also one of the smallest group of sizes.

The following graphs present the results of the analysis of the size variable.

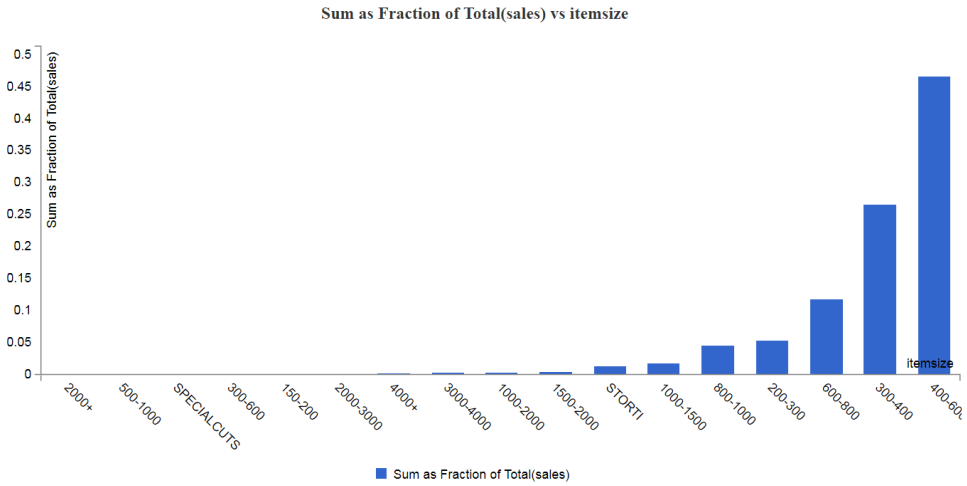


Figure 3.11: Sum of sales as fraction of total sales per size (euro)

itemgroup \ itemsize	2000+	500-1000	SPECIALCUTS	300-600	2000-3000	4000+	1500-2000	3000-4000	1000-2000	150-200	1000-1500	STORTI	800-1000	200-300	600-800	300-400	400-600	Totals
Meagre		0.9%	0.0%		12.1%	17.4%		24.1%	26.8%		11.3%	7.4%						100.0%
Red seabream	0.2%		0.0%				0.6%			0.0%	2.4%	7.3%	4.3%	1.9%	13.4%	8.7%	61.2%	100.0%
Seabass			0.0%	0.1%			0.5%			0.6%	1.4%	2.3%	3.5%	10.1%	10.1%	30.5%	40.8%	100.0%
Seabream			0.0%	0.1%			0.0%			0.2%	0.6%	2.7%	2.7%	4.7%	10.4%	28.8%	49.8%	100.0%
Totals	**0.0%**	**0.0%**	**0.0%**	**0.1%**	**0.1%**	**0.2%**	**0.2%**	**0.3%**	**0.3%**	**0.4%**	**1.1%**	**2.6%**	**3.0%**	**6.9%**	**10.2%**	**29.0%**	**45.6%**	100.0%

Figure 3.12: Sum of sales as fraction of total sales per size and type

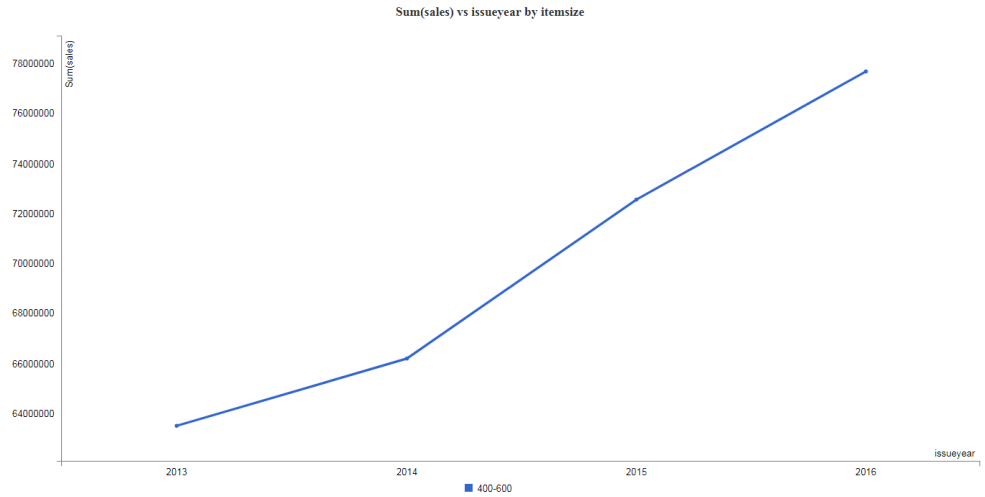


Figure 3.13: Sum of sales for size 400-600 per year (euro)

The total sales for size (400-600) increased in every year for the study period, starting from 63.540 million euros and ending up to 77.708 million euros, which lead to an increase of almost 22%.

Analyze metrics per product

Analyzing data per product type, the results showed that in 2015, total production of Seabream was less than in previous years. However, in the same year, the selling price of Seabream reached its maximum value with an average of 5.84 euros/kg. Moreover, in July of 2015 the selling price was at its highest points. The top 5 average selling prices per month for Seabream where from May to September of 2015 (6.22 euros/kg). Also, figures regarding the quantity produced for Seabream indicates that the demand for Seabream is increasing over the years.

The tables below present the numbers for total sales and selling price per product over the study period.

itemgroup \ issueyear	2013	2014	2015	2016	2017	Totals
Meagre	7.49	8.12	6.83	5.60	5.18	**6.20**
Red seabream	5.12	5.77	5.99	6.43	6.19	**6.06**
Seabass	5.57	5.83	5.84	6.00	6.01	**5.85**
Seabream	4.61	5.18	5.84	5.38	5.19	**5.22**
Totals	**5.00**	**5.46**	**5.85**	**5.68**	**5.57**	**5.51**

Figure 3.14: Average selling price per year and product (euro/kgrs)

itemgroup \ issueyear	2013	2014	2015	2016	Totals
Meagre	225,120.00	160,458.00	302,442.00	484,810.00	**1,172,830.00**
Red seabream	63,552.00	262,042.00	322,789.00	374,086.00	**1,022,469.00**
Seabass	10,915,401.00	10,326,277.00	11,260,801.00	13,084,529.00	**45,587,008.00**
Seabream	17,182,151.00	15,141,790.00	14,587,884.00	14,463,555.00	**61,375,380.00**
Totals	**28,386,224.00**	**25,890,567.00**	**26,473,916.00**	**28,406,980.00**	**109,157,687.00**

Figure 3.15: Total quantity per year and product (kgrs)

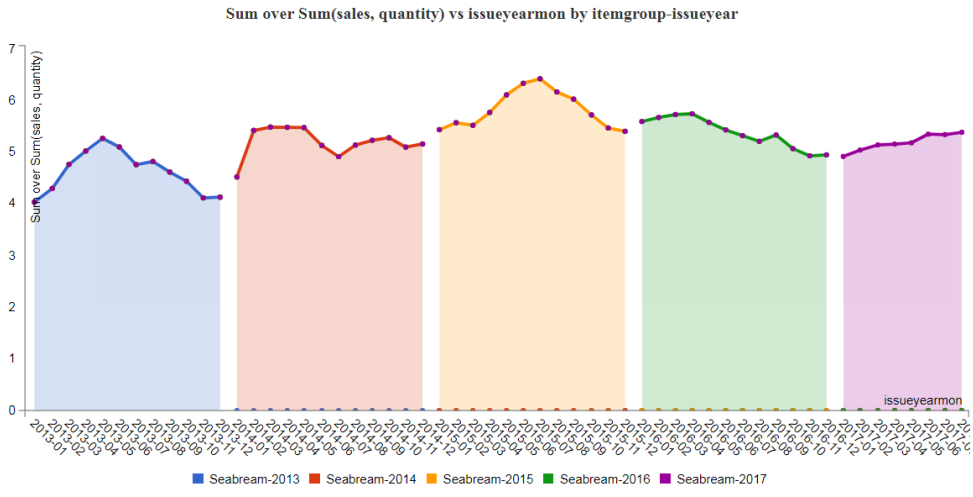


Figure 3.16: Average selling price per month for seabream (euro/kgrs)

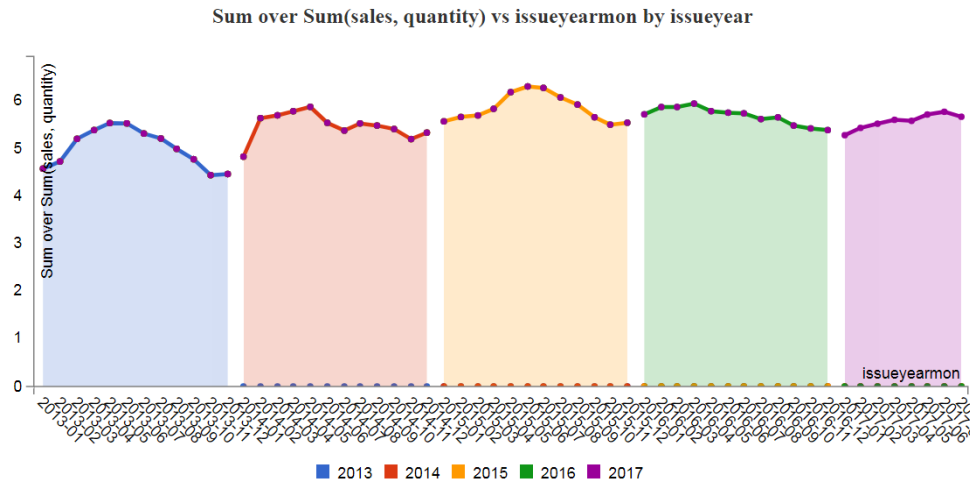


Figure 3.17: Average selling price per month for all products (euro/kgrs)

On the other hand, the graph below shows that the selling price of Meagre has been decreasing over the last months. This confirms a previous conclusion that small sizes have a greater demand than larger ones, as sizes for Meagre range from (500-1000) to (4000+).

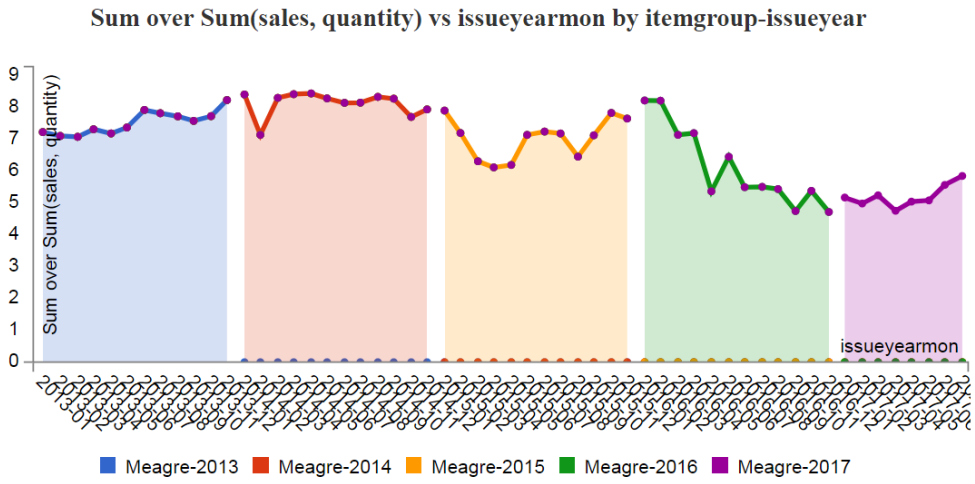


Figure 3.18: Average selling price per month for meagre (euro/kgrs)

Analysis of fish category

According to the analysis of average selling price through the period of study, the results showed that Fillet is sold 1.84 times more than Gutted and 2.45 times more than Whole fish. Despite the difference in the selling price among the three categories, Whole fish is the dominant category in sales as almost 90% of total revenue is obtained from the sales of Whole fish.

Similar proportions are occurred also in quantities. Analysis showed that 93.75% of total quantity produced follows the path of the Whole fish while only the 6% is sold in the market as Fillet or Gutted fish.

The tables below show the figures regarding total sales per fish category.

itemcategory \ issueyear	2013	2014	2015	2016	Totals
Fillet	12.81	12.92	12.99	12.79	**12.88**
Gutted	6.40	7.17	7.29	7.13	**7.00**
Whole	4.81	5.22	5.61	5.41	**5.26**
Totals	**5.00**	**5.46**	**5.85**	**5.68**	**5.49**

Figure 3.19: Average selling price per year and category (euro/kgrs)

itemcategory	Fillet	Gutted	Whole	Totals
Totals	**5.5%**	**4.9%**	**89.6%**	**100.0%**

Figure 3.20: Sum of sales as fraction of total sales per category

3.2.3 Geographic analysis

The company has 38 point of sales all over the world. Each point represents a country, from Canada to Singapore. Results from geographic analysis showed that the top 10 countries in sales are Italy, France, Greece, Portugal, Spain, Great Britain, Germany, USA, Holland and Romania. Also, between the period from January of 2013 to December of 2017, these 10 countries held almost 90% of total sales (almost 535 million euros). The table below describes the percentage of total sales for the top 10 countries per year as fraction of the total sales of the company.

country \ issueyear	2013	2014	2015	2016	2017	Totals
ITALY	4.2%	4.5%	4.7%	4.6%	3.0%	**21.1%**
FRANCE	3.5%	3.4%	3.6%	3.8%	2.9%	**17.2%**
GREECE	1.6%	1.9%	2.5%	3.0%	2.1%	**11.2%**
PORTUGAL	2.2%	1.7%	2.2%	1.7%	1.3%	**9.2%**
SPAIN	1.3%	1.6%	1.9%	2.2%	1.7%	**8.7%**
GREAT BRITAIN	1.4%	1.3%	1.4%	1.5%	0.8%	**6.4%**
GERMANY	0.9%	1.2%	1.4%	1.2%	0.9%	**5.6%**
USA	1.0%	1.1%	1.1%	1.3%	0.8%	**5.3%**
HOLLAND	0.7%	0.6%	0.7%	0.7%	0.5%	**3.1%**
ROMANIA	0.2%	0.3%	0.4%	0.5%	0.4%	**1.8%**

Figure 3.21: Sum of sales as fraction of total sales for top 10 countries in sales

In addition, the total quantity produced for the same countries and period was 97,688 tonnes of fish that is equivalent to 89.49% of the total quantity. The next table shows the proportion of total quantity per country as fraction of the total quantity of the company.

country \ issueyear	2013	2014	2015	2016	Totals
ITALY	29,882,685.00	32,014,523.00	33,278,760.00	33,039,405.00	**128,215,373.00**
FRANCE	24,649,637.00	24,311,554.00	25,571,835.00	27,117,775.00	**101,650,801.00**
GREECE	11,595,617.00	13,762,932.00	17,891,608.00	21,325,752.00	**64,575,909.00**
PORTUGAL	15,966,469.00	12,395,242.00	15,313,933.00	12,310,464.00	**55,986,108.00**
SPAIN	9,352,530.00	11,246,100.00	13,581,092.00	15,918,258.00	**50,097,980.00**
GREAT BRITAIN	9,727,136.00	9,242,075.00	10,309,956.00	10,685,589.00	**39,964,756.00**
GERMANY	6,713,130.00	8,445,774.00	9,979,609.00	8,682,206.00	**33,820,719.00**
USA	7,203,708.00	7,712,978.00	7,705,048.00	9,303,291.00	**31,925,025.00**
HOLLAND	4,905,454.00	4,029,275.00	4,669,769.00	5,039,093.00	**18,643,591.00**
ROMANIA	1,654,452.00	2,104,632.00	2,871,097.00	3,669,491.00	**10,299,672.00**
Totals	**121,650,818.00**	**125,265,085.00**	**141,172,707.00**	**147,091,324.00**	**535,179,934.00**

Figure 3.22: Sum of quantity per year for top 10 countries in sales (kgrs)

Customized clustering of countries

A cluster analysis of countries characteristics based on GDP per capita, total population and total sales showed that Russia, although has a GDP close to Romania and population 8 times more than Romania's, it is grouped with countries with medium sales. This result heavily dependent on the fact in August of 2014 Russia stopped purchasing fish products from the company. This might be caused by Russia's embargo that banned European Union food imports in August of 2014. And that is also the case with Ukraine.

Below is a time series graph of total sales for Greece, Russia and Ukraine and it shows how sales for Russia and Ukraine moved downwards after December of 2013 and reduced to zero in August of 2014.

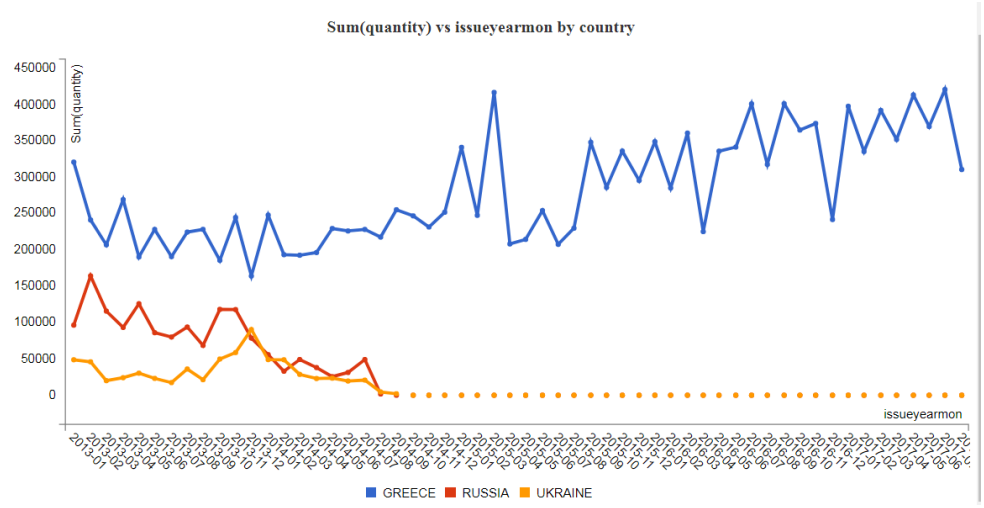


Figure 3.23: Sales trend of Greece, Russia and Ukraine

The customized cluster analysis depends 70% on the number of the average annual total sales per country, 15% on the number of total population in 2016 and 15% on the number of GDP per capita in 2016. According to this parameters, the analysis produced 3 groups of countries.

Another conclusion from this customized cluster analysis is that although Greece, Portugal, Estonia and Czech Republic have great similarity in terms of GDP and total population, they are grouped in different clusters. Greece and Portugal belong to the first group whereas Estonia and Czech Republic belong to the second. This is also due to the fact that Greece is the location of company's head office and Portugal has a very high rate in sales (4th place) whereas sales figures for Estonia and Czech Republic show a discontinuous demand for fish.

Classification of countries

Using the classification method of equal intervals with 7 classes, the results showed that in case of total quantity for year 2016, the 7th class contains

22 countries. That is also the case with total sales. This indicates that top countries in sales exceed by far countries with lower sales.

In 2016, 84% of the countries that had at least one transaction were European and the countries with the higher sales were Italy, France, Greece, Spain and Portugal. Actually, almost 67% of total sales in 2016 came from 16% of total countries or from 5 Mediterranean countries.

The following tree-map graphs show the difference among countries in terms of total sales and total quantities.

Figure 3.24: Treemap of total sales per country for 2016

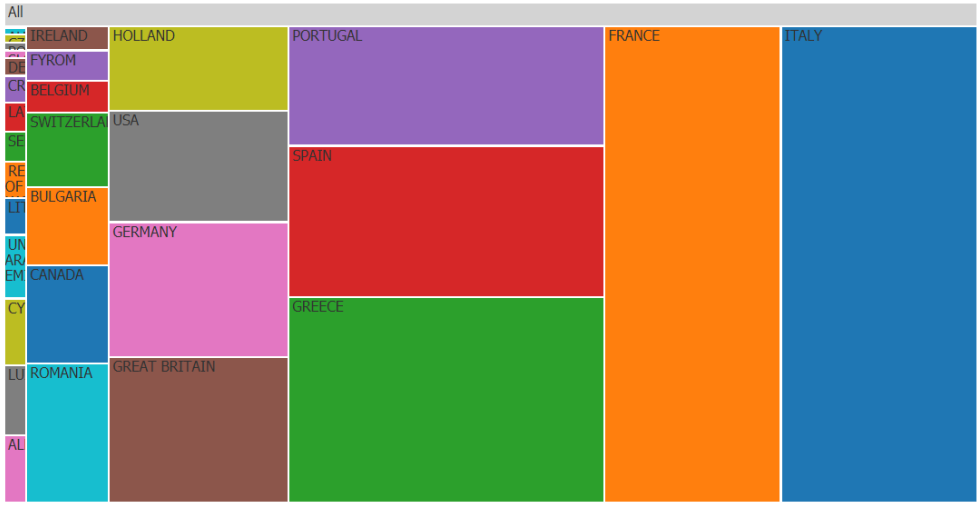


Figure 3.25: Treemap of total quantity per country fro 2016

According to this classification methodology, the top 5 countries belong to five different classes. Also, the cumulative frequency of sales, which is defined

as the sum of all previous frequencies up to the current point when frequencies are ordered from the smallest to the largest, indicates that countries with lower sales have a very small contribution in total sales.

country	Totals
ITALY	20.5%
FRANCE	16.8%
GREECE	13.2%
SPAIN	9.9%
PORTUGAL	7.6%

Figure 3.26: Sum of sales as fraction of total sales in 2016 for top 5 countries

3.2.4 Predictive analysis of total sales

Finally, a predictive analysis applied on time series data regarding the number of total sales of the company showed that the prediction of sales per month for the next 12 months is very positive. Sales will have a constant decline until February of 2018 and after that point they will increase and in July of 2018 will reach the number of almost 16.9 millions, which is the highest value of sales per month from January of 2013.

The model is created using the ARIMA algorithm and is based on sales data from January 2013 to August 2017. The model is also based on the trend and considers a 12 month (yearly) seasonality.

The graph shows the predictive values for total sales.

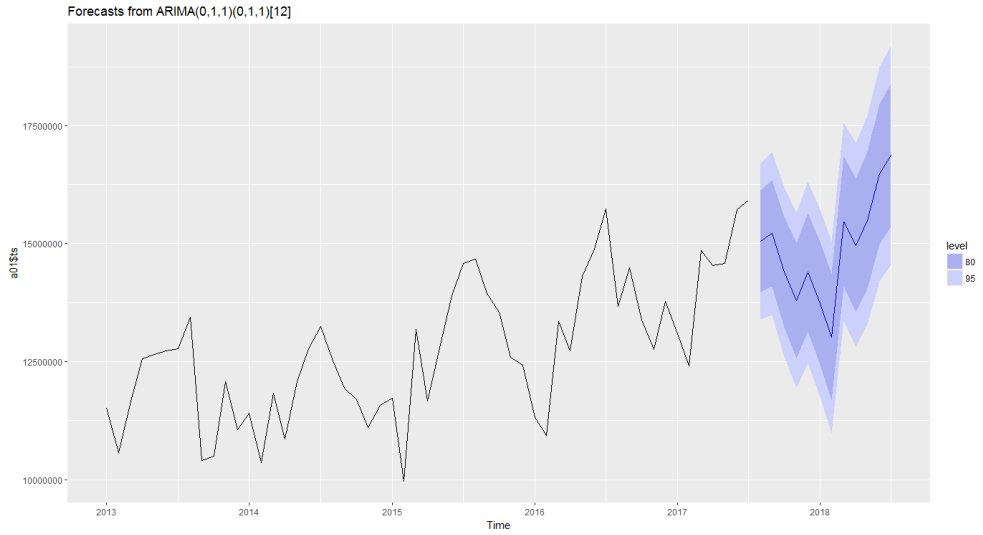


Figure 3.27: Total sales forecast per month from Sep.2017 to Aug.2018

Chapter 4

Data Storytelling Application: Design and Implementation

4.1 Introduction

This chapter presents the process of transforming the findings of the previous paragraphs into an interactive storytelling application. It focuses on the implementation of the web application based on data storytelling concepts. The objective here is to present the knowledge gained from data analysis in a meaningful and communicative way. Also, in this step a variety of web technologies is explored in order to select the most suitable tools to achieve the best possible results.

4.2 Storytelling structure

The story is divided into three parts. The first part contains the main title of the story, an introductory paragraph and some general questions in which the story is about to give answers.

The second part describes the key points of the analysis. In particular, this part illustrates the summary of the results obtained by data analysis in a standard presentation template that is repeated until the end of second part. The template consists of a title that describes the subject of the paragraph accompanied by plain text and interactive graphs. The results presented are numbered in order to build a more linear storytelling experience to the user. The first key points presented are more general and as the story continues, they become more specific and more complex. This also applies to the level of user's interaction. At first, the user is able to interact with the graphs with simple actions like 'on mouseover' and 'popup windows' while in the end the user is able to configure the variables displayed, change the values of a specific variable and customize the visualization of the graph.

The third and last part contains a brief conclusion of the whole story. Screen shots of the application can be found in Appendix B.

4.3 Development and implementation

The next paragraphs provide information on the technologies used for the development of the storytelling web application and describe in detail the results of the implementation.

4.3.1 Technologies

The technologies used for the implementation of the web application, as with the most web applications, are separated in the front-end and back-end technologies. In this case, back-end technologies can be summarized in server side R and sql language for the design of services and interaction with the database. In the front-end, which is the part that interacts with the user, the application uses javascript, css and R language with several modules.

For the implementation of the interactive maps and graphs a list of R modules is used. The main ones are Leaflet, GoogleCharts, D3 and Highcharter. The whole application is developed within R Shiny environment.

4.3.2 User interaction

The main objective of this data storytelling approach is to create a highly interactive environment so that the user will be able to configure and customize specific objects. It is very important that the interaction should not increase the complexity of the story, but on the contrary, should obtain its simplicity. This way, the user can become part of the story by having a more active role.

In this case, the second part of the story contains 9 sections. Each one introduces a result from the data analysis and is implemented with a standard format to help the user easily understand and become familiar with the story. In order to achieve a successful communication between the content and the user, every section consists of one or more objects with interactive elements. These objects describe in a graphical way the main concept of each section.

section 1. An introduction to Company's production line and products

The first section is a visualization of the data model. The purpose of this section is to introduce the main variables and basic concepts of Marineculture industry used in the story.

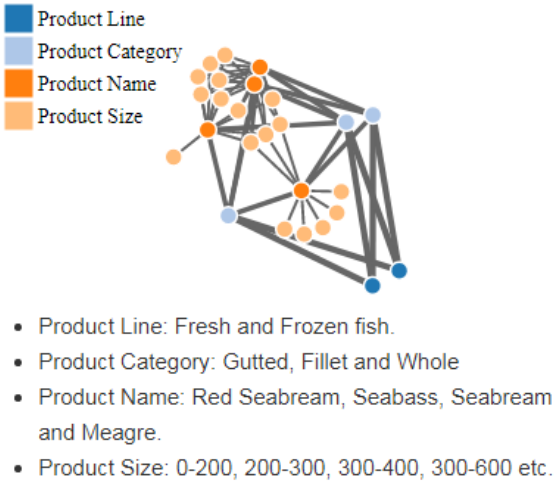


Figure 4.1: Interactive network analysis visualization using D3 library

The user is able select a specific node from the graph and see its possible connections. As for example, if the user selects the product 'Meagre' then only the connected nodes to this product are showed with more intense colors. This way the user can understand the possible path of the product, meaning its sizes and categories. This visualization is even more useful when the data model consists of multiple dimensions with a larger complexity.

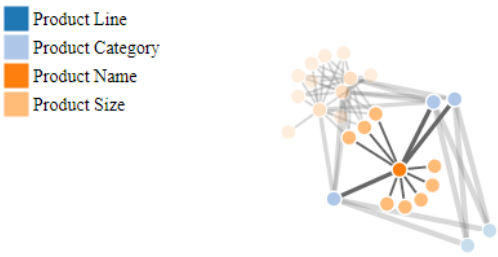


Figure 4.2: Interactive network analysis visualization using D3 library

section 2. Total sales derived from exports

The second point of the story talks about the percentage of total sales derived from exports. This large difference is illustrated with a pie chart. Except from a small window (pop up) that appears when the user places the mouse on the graph, in this section there are no further interactions.

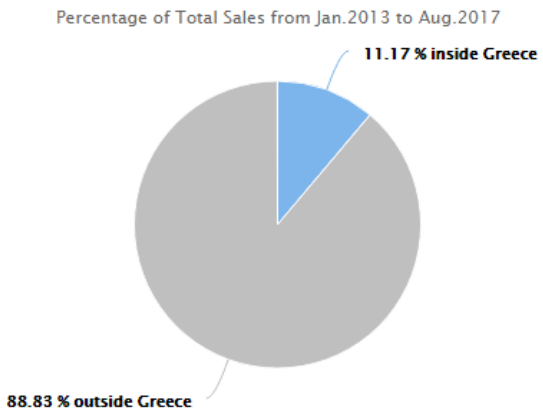


Figure 4.3: Pie chart visualization of sales

section 3. High sales figures are depicted between May and September

The third point presents the period of the year with the highest sales. At this point, the user is able to interact with a line chart and select the preferred variable for visualization. The user can select more that one variables in the same graph. This way makes easier for the user to understand the trends and compare them among different variables (sales, quantity, selling price).

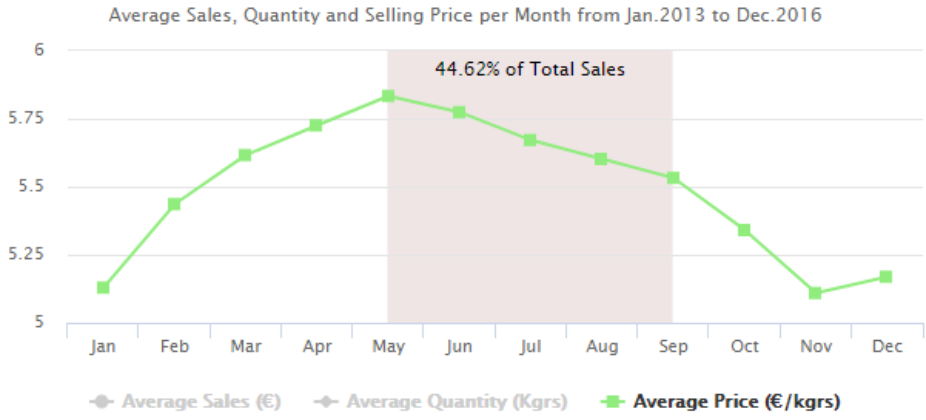


Figure 4.4: Interactive line chart visualization of average selling price per month using highcharter library

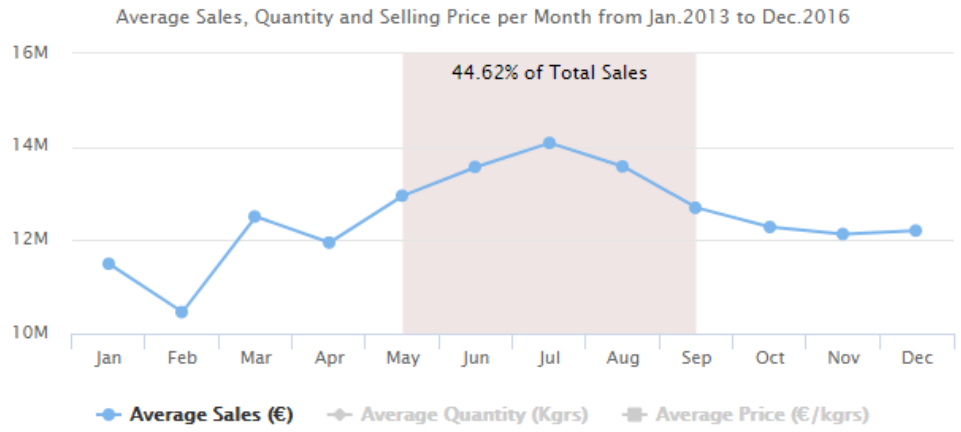


Figure 4.5: Interactive line chart visualization of average sales per month using highcharter library

section 4. Demand for small sizes is greater than larger sizes

The forth point of the story compares the total sales and total quantity produced among different sizes. In this section, the user is able to change the variable as well as the plot type of the graph.

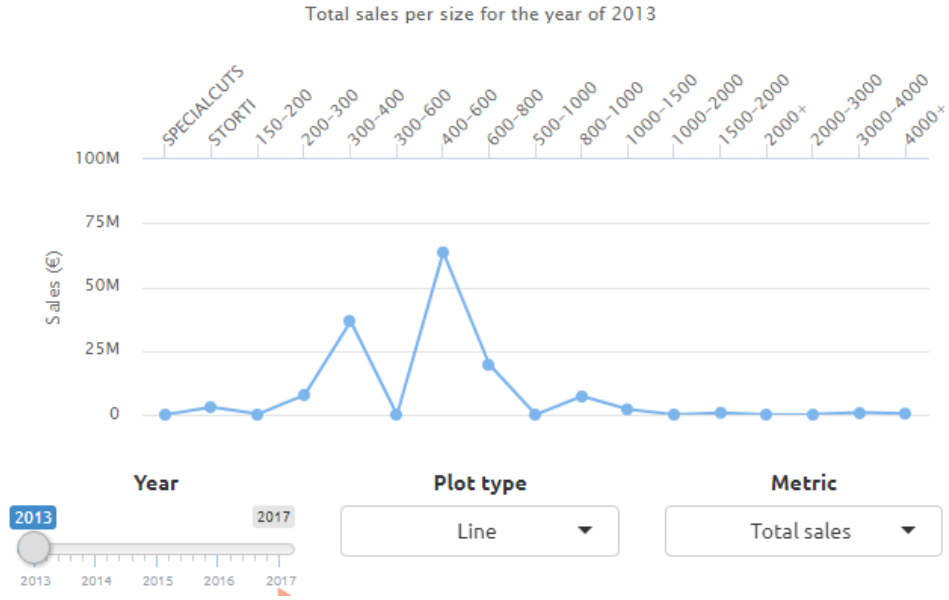


Figure 4.6: Interactive line/scatter/column visualization of total sales and total quantity per fish size in time using google charts library

Also, another interaction is the time line on the left beneath the graph. This gives the opportunity to the user to select a specific year by dragging

the place holder parallel to the time line. Moreover, the user can generate the animation mode by selecting the play button under the time line. In this way, the changes of the variables during time can be traced easier and furthermore the illustration becomes more pleasant for the user. At this point, the user has become more active and more engaged with the story.

section 5. In 2015 Seabream had a raise of 7 billion euros in sales

The fifth section is about the large raise of Seabream in sales in 2015. By this point, the user is in the middle of the story and the graph and the corresponding text are pointing a pick.

In this section, the user can interact with the graph the same way as in section 3.

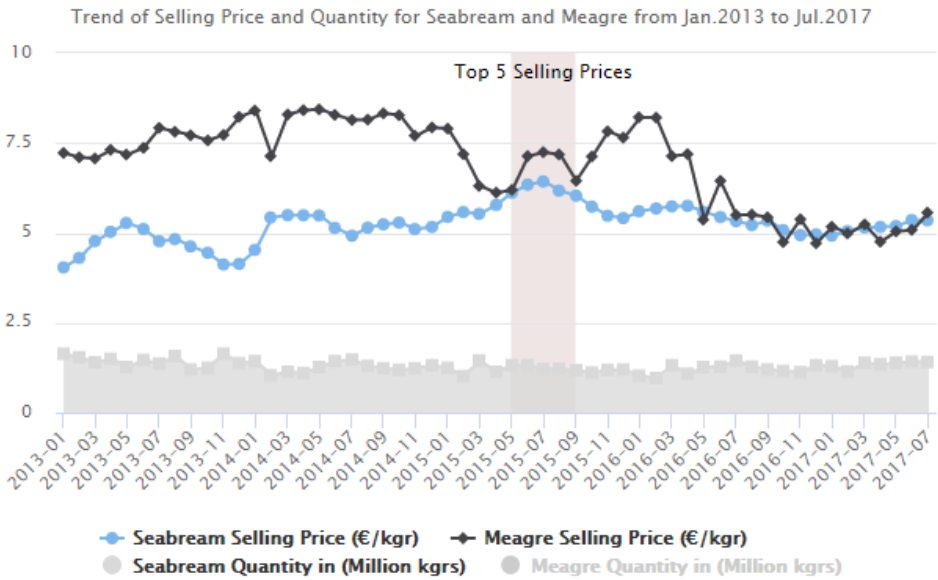


Figure 4.7: Interactive time series visualization of selling price and quantity per fish type using highcharter library

section 6. Price category: Whole vs Fillet vs Gutted

The sixth point describes the price differences among categories of fish. Here the user can select the preferred variable as in previous charts and retrieve the exact value per month or in total.

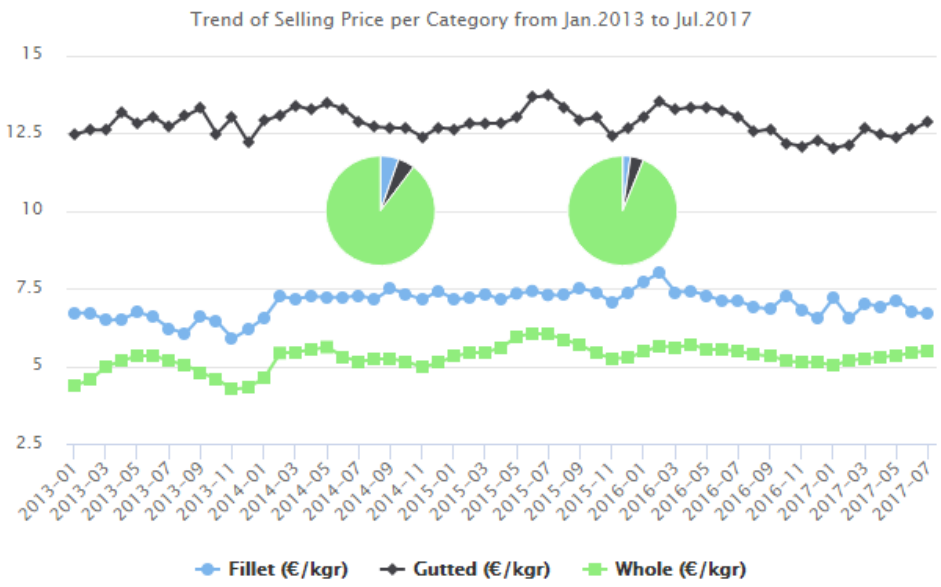


Figure 4.8: Interactive time series complex visualization (pie,line chart) of selling price per fish category using highcharter library

section 7. Geographic destination of sales: clustering countries

This section describes the data from a geographic perspective. Sales are analyzed from a geographic point of view. The user is introduced to a basic visualization regarding the point of sales.

In the first graph, the user has the exact possible interactions as in previous graphs. However, the second graph gives the user the opportunity to play with the data. In particular, the user is able to define a scatter plot view based on categories and graph's boundaries (zoom in/out).

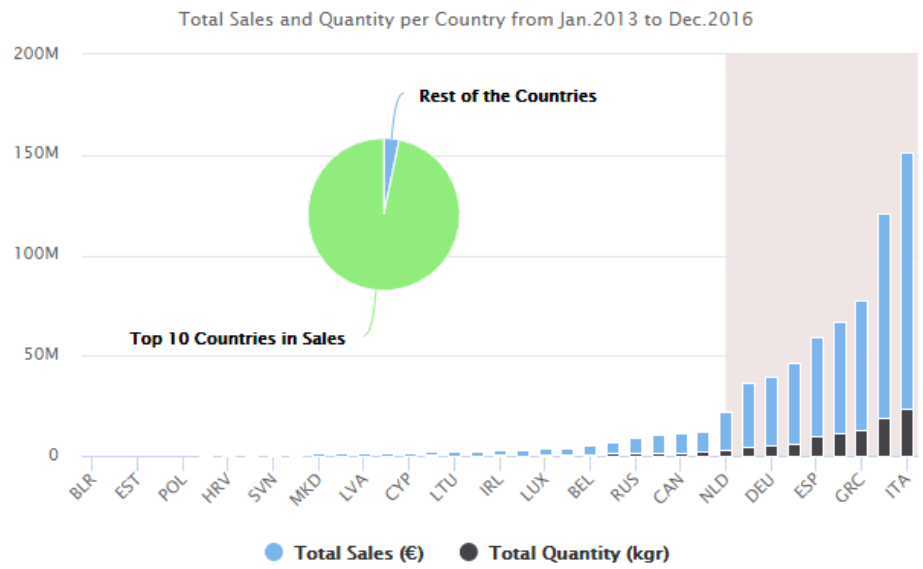


Figure 4.9: Interactive complex (bar and pie chart) visualization of total sales per country using highcharter library

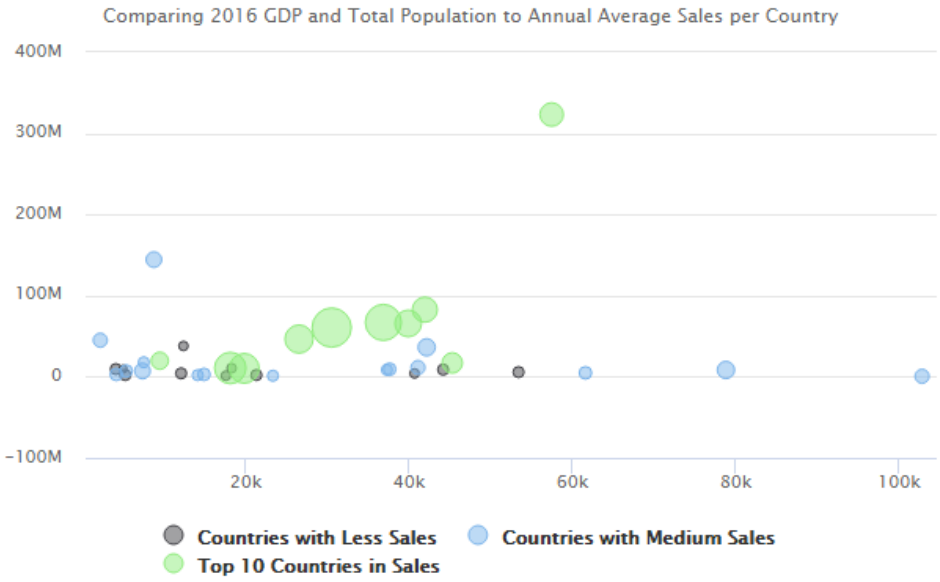


Figure 4.10: Interactive scatter plot visualization of countries clustering using highcharter library

section 8. Mapping and classification analysis of destinations

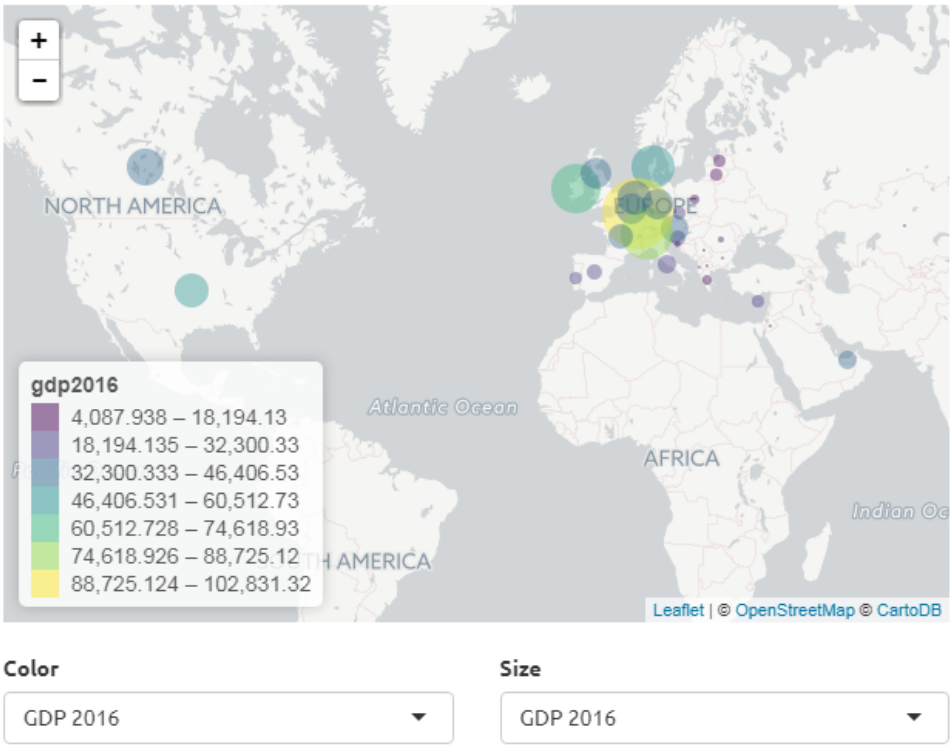


Figure 4.11: Interactive geographic map using OSM maps and leaflet library - visualization of countries classification based on GDP per capita in 2016

This section is the heart of the story as this is where the geographic analysis takes place. In this section, the user can have the highest level of interaction with the data.

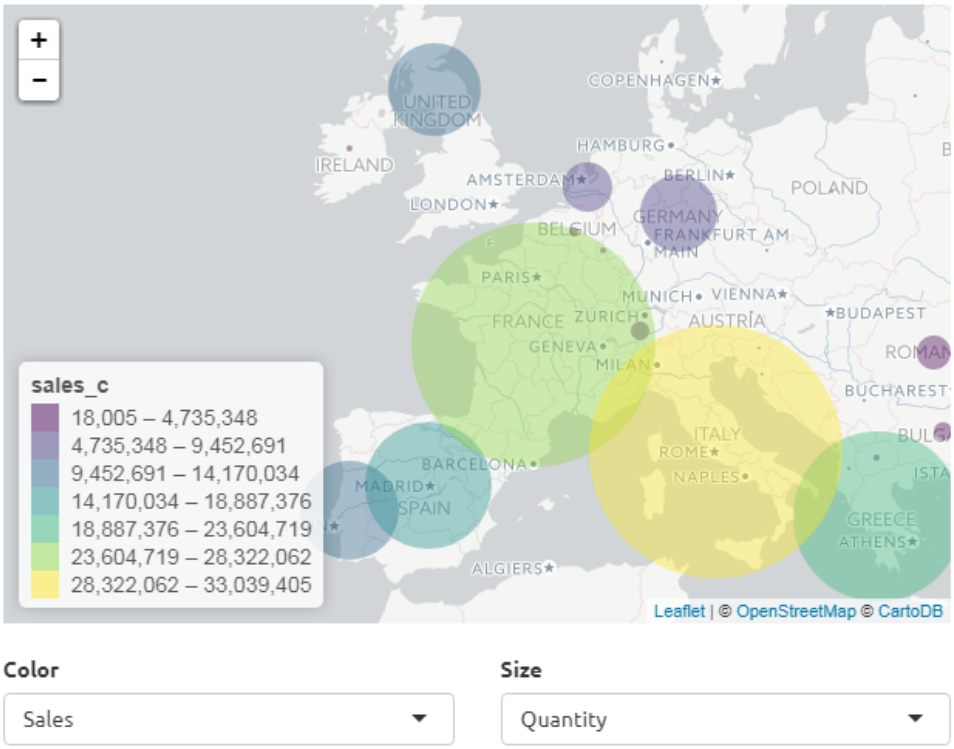


Figure 4.12: Interactive geographic map using OSM maps and leaflet library - visualization of countries classification based on total sales

The map visualization, which is implemented based on Leaflet library, can classify on the fly the countries according to multiple variables (total sales, total quantity, cumulative sales frequency, number of orders, GPD per capita, total population). The classification method applied (equal interval) is one of the simplest classification methods and that makes easier for the user to understand the result of the map. The user can select and visualize the graduated variables in either sized or colored circles.

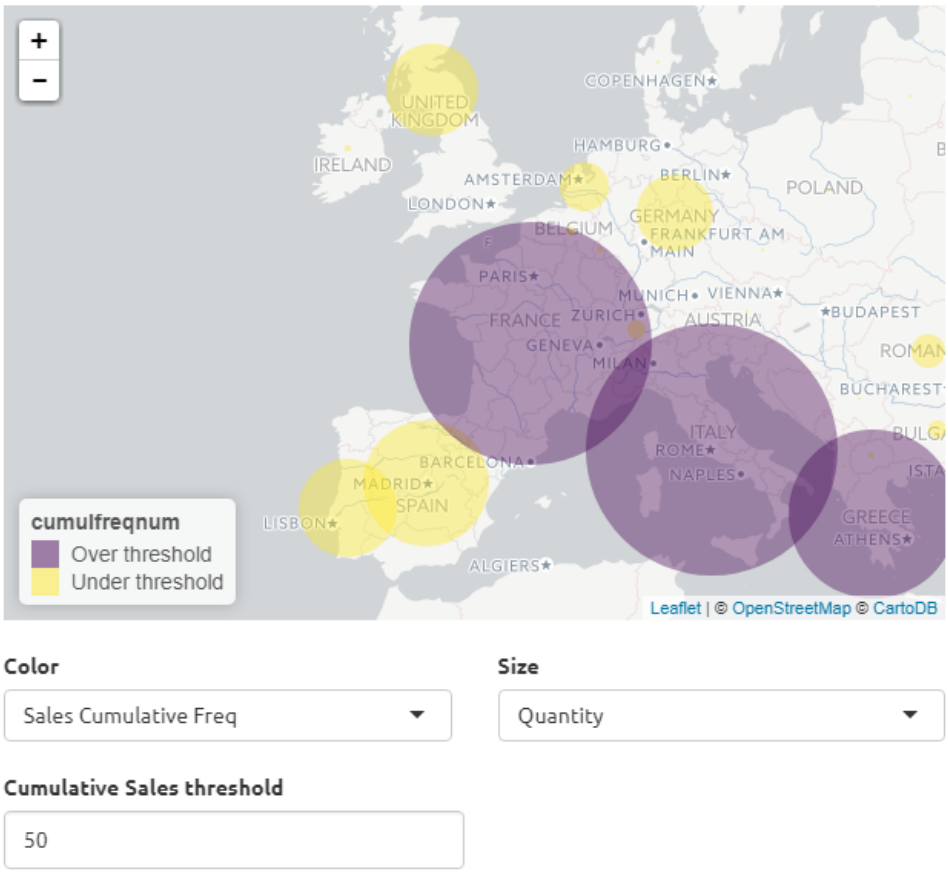


Figure 4.13: Interactive geographic map using OSM maps and leaflet library - visualization of countries classification based on input parameter

Moreover, the user is able to interact with the classification by changing some input parameters. As for example, the user can define the exact number for the cumulative frequency of sales and then separate the countries in two classes, the one that contains countries under this threshold and the opposite.

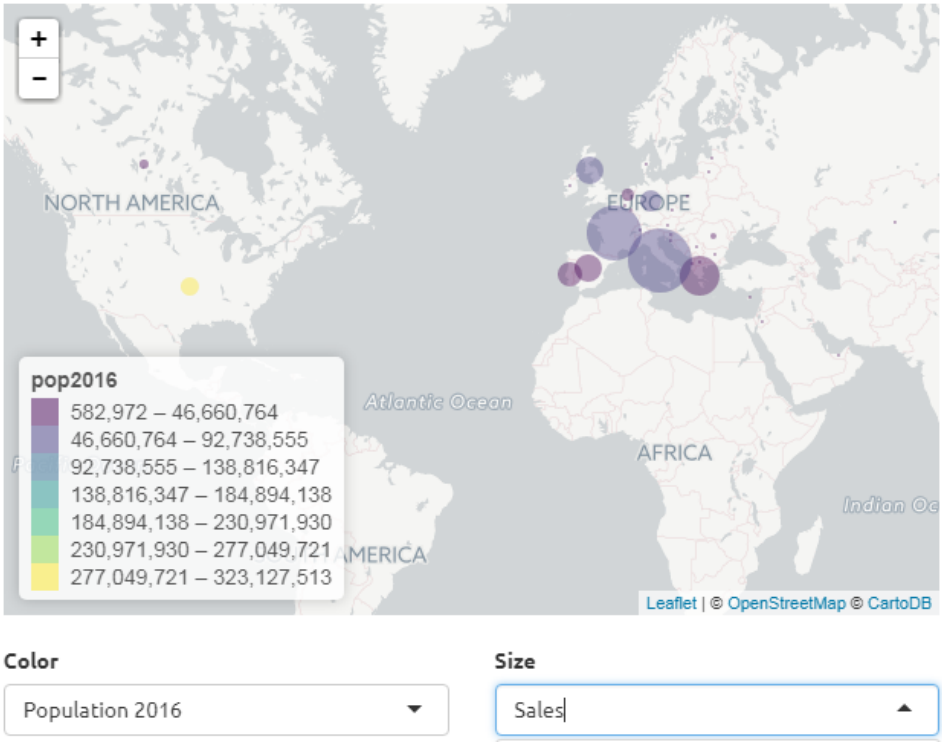


Figure 4.14: Interactive geographic map using OSM maps and leaflet library - visualization of countries classification based on total population in 2016

The graphs under the map are reproduced every time the map changes its boundaries. The graphs are configured to display only the countries that are visible in the map. For example, if the user sets a map view that contains only South America then all the graphs from this section will accordingly have as input data countries from South America, in this case USA and Canada. This way the user is able to make a basic geographic analysis by dragging the map and then looking into the graphs.

The figures below show the graphs that depend on map's configuration.

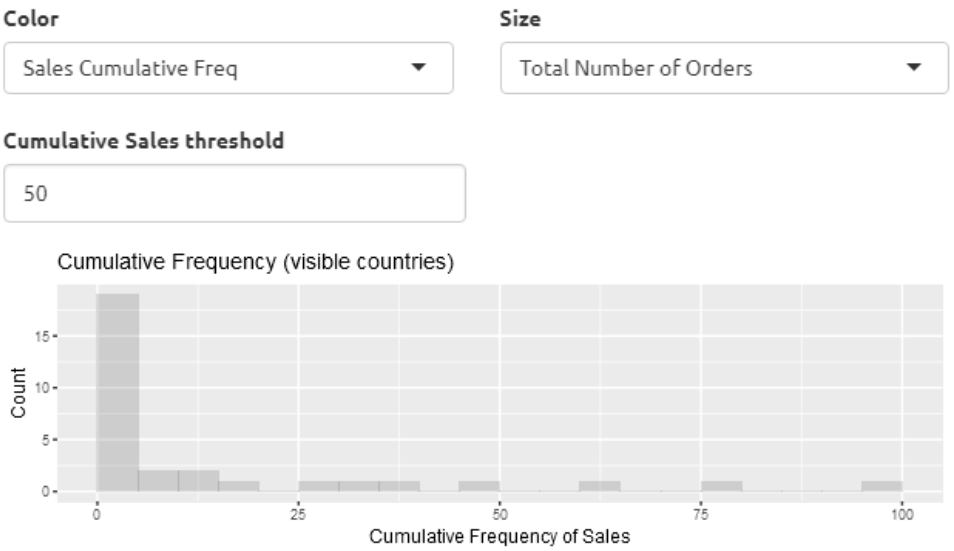


Figure 4.15: Interactive bar chart visualization of countries using ggplot library

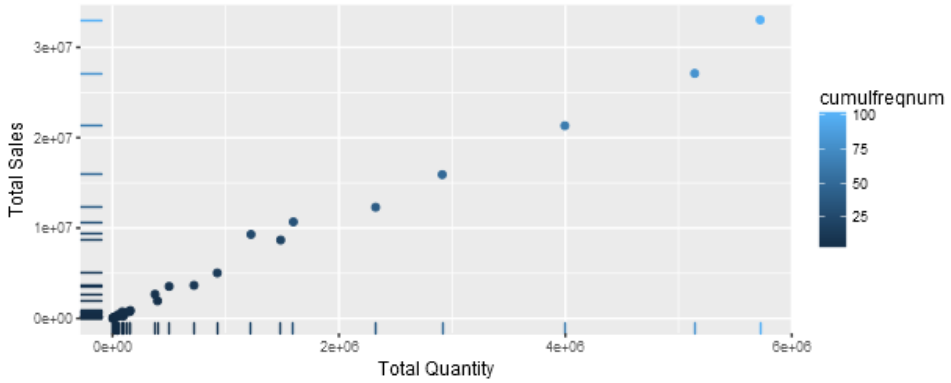


Figure 4.16: Interactive scatter plot visualization of countries using ggplot library

In this section, the user can compare almost all the metrics of the data from a geographic approach, which is the purpose of the current data storytelling implementation.

section 9. Sales forecast for 2018

The last section of the second part of the story presents a forecast on total sales. Here the level of interaction returns to basic and the user is able to get the exact number of sales for the predictive period. The graph in section 9 may have only two possible views that are displayed in the next figures.

After this last point, the story continues with the conclusion where no interaction between the user and the application takes place.

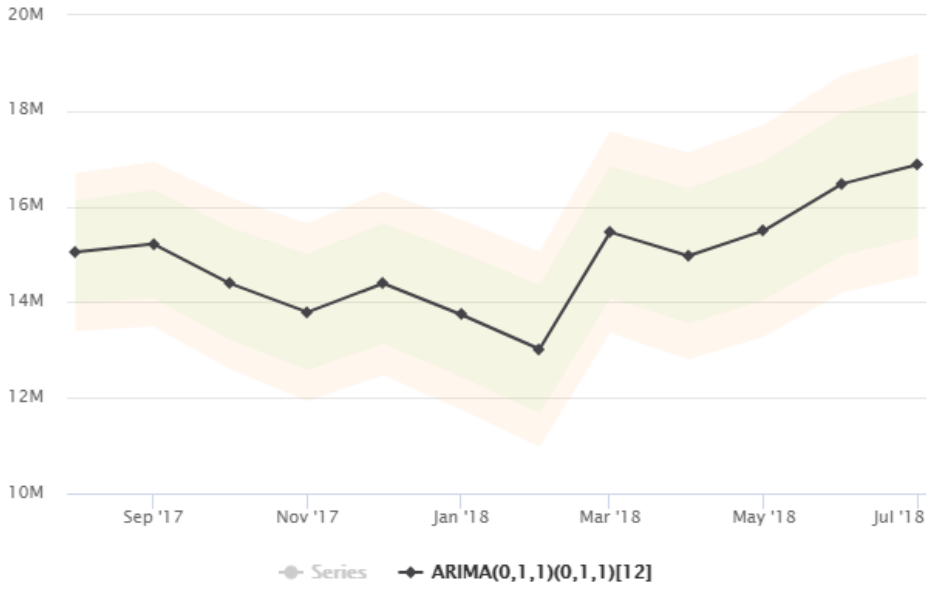


Figure 4.17: Interactive line chart visualization of forecasting algorithm using highcharter library (1)

Figure 4.18: Interactive line chart visualization of forecasting algorithm using highcharter library (2)

Chapter 5

Conclusion

5.1 Summary

This study presented the end to end development of a data storytelling web application based on concepts of storytelling theory and on specific data analysis and mining methodologies by using cutting-edge technologies and with an emphasis on geographic data visualization and interactive graphs. It described one by one the steps of creating a data storytelling web application, from the step of data collection to application deployment.

In general, the current web application implemented with the use of PostgreSQL, R Shiny package and a list of R libraries that are mentioned in the Appendix. Leaflet.js used for the implementation of the geographic visualization and Open Street Maps for the basemap integration of the web map. As a result, the current implementation presented a sample of the capabilities of the technologies used as it described the hole process step by step and that made impossible to take advantage of all the opportunities provided. However, this overall implementation revealed possibilities for future work.

5.2 Future work

Data storytelling applications and their related work have started to appear very recently. Therefore, they have plenty of room for improvement and with this study there are several lines of research arising.

One of the major concerns is the level of application's interaction with the user. The current implementation included text, various type of graphs and map visualization. In this case, the user was able to interact with the data by selecting options only within the graphs and as a result those options changed the output of the graph. In a future case, this action can be generated by a text selection or even more a selection of a specific word that indicates the following changes in the graphs.

Also, a another challenging task is the design of a data storytelling application based on spatio-temporal data sets by using the most recent visualization

tools and spatial related techniques. This scenario becomes even more interesting when new ways of presentation are considered. New ways of presenting data can either include technologies that produce 3D visualizations or those that create environments of augmented reality.

Except from the presentation of the data storytelling application and the level of its interaction with the user, another improvement can concern the level of analysis. This is a very difficult task and also a task of great importance. The current application presented an equal interval classification of points according to the user's selection. Other methodologies can also be integrated and generated on the fly in order to illustrate meaningful information through the implementation of a single object, either a web map or a plain table.

Finally, the entire design and user experience can be enriched with the use of motion pictures, animated infographics and audio material. This kind of content is going to engage further the audience and contribute on increasing the application friendliness.

Bibliography

[1] Andritsos, P. (2002). Data clustering techniques. Rapport technique, University of Toronto. Department of Computer Science.

[2] Bembenik, R., & Protaziuk, G. (2004). Mining spatial association rules. In Intelligent Information Processing and Web Mining (pp. 3-12). Springer Berlin Heidelberg.

[3] El-Zawawy, M. A. (2012). Efficient techniques for mining spatial databases. arXiv preprint arXiv:1206.0217.

[4] FAO, (2016). The State of World Fisheries and Aquaculture 2016. Contributing to food security and nutrition for all. Rome. 200 pp.

[5] Han, J., Kamber, M., & Tung, A. K. H. (2001). Spatial Clustering Methods in Data Mining: A Survey.

[6] Kolatch, E. (2001). Clustering algorithms for spatial databases: A survey. PDF is available on the Web, 1-22.

[7] Kumar, P. (2014). Data Storytelling: A Definition? Retrieved from https://priyakumar.wordpress.com/2014/03/19/data-storytelling-a-definition/

[8] Kumar, C. N. S., Ramulu, V. S., Reddy, K. S., Kotha, S., & Kumar, C. M. (2012). Spatial data mining using cluster analysis. International Journal of Computer Science & Information Technology, 4(4), 71.

[9] Maimon, O., & Rokach, L. (2005). Decomposition methodology for knowledge discovery and data mining. Data mining and knowledge discovery handbook, 981-1003.

[10] Malshe, R. (2017). What is the difference between Data Analytics, Data Analysis, Data Mining, Data Science, Machine Learning, and Big Data? Retrieved from https://www.quora.com/What-is-the-difference-between-Data-Analytics-Data-Analysis-Data-Mining-Data-Science-Machine-Learning-and-Big-Data-1

[11] Mohammad Valadkhani, M. (2016). Knowledge Discovery in Data (KDD) Process. Retrieved from https://www.linkedin.com/pulse/knowledge-discovery-data-kdd-process-mohammad-valadkhani

[12] Monnappa, A. (2017). Data Science vs. Big Data vs. Data Analytics. Retrieved from https://www.simplilearn.com/data-science-vs-big-data-vs-data-analytics-article

[13] National Storytelling Network, (2017). What is Storytelling? Retrieved from http://www.storynet.org/resources/whatisstorytelling.html

[14] Otair, D. (2013). Approximate k-nearest neighbour based spatial clustering using kd tree. arXiv preprint arXiv:1303.1951.

[15] Padmavati, V (2013). Review of Spatial Algorithms in Data Mining.

[16] Rouse, M. (2015). Data storytelling. Retrieved from http://searchcio.techtarget.com/definition/data-storytelling

[17] Rouse, M. (2017). Data analytics (DA). Retrieved from http://searchdatamanagement.techtarget.com/definition/data-analytics

[18] Sadowski, J. (2011). How can storytelling play a role in the communication of concept brands? Retrieved from https://milanvaassen.wordpress.com/2013/11/14/how-can-storytelling-play-a-role-in-the-communication-of-concept-brands/

[19] Smith, A. (2015). February 25, 2015. The Power Of Storytelling. Retrieved from https://www.ceros.com/blog/the-power-of-storytelling/

[20] Sumathi, N., Geetha, R., & Bama, S. S. (2008). Spatial data mining—techniques trends and its applications. Journal of Computer Applications, 1(4), 28-30.

[21] Swathi, K. G., & Rajesh, K. N. V. S. S. K. (2012). Comparative analysis of clustering of spatial databases with various DBSCAN Algorithms. IJRCCT, 1(6), 340-344.

[22] Varlaro, A. (2008). Spatial clustering of structured objects (Doctoral dissertation, University of Bari, Italy).

[23] Varghese, B. M., & Unnikrishnan, A. (2013). Spatial clustering algorithms - An overview. Asian journal of computer science & information technology, 3(1).

[24] Walker B. (2015). Every day big data statistics. Retrieved from http://www.vcloudnews.com/every-day-big-data-statistics-2-5-quintillion-bytes-of-data-created-daily

Appendix

R packages used

1. library("leaflet")
2. library("RColorBrewer")
3. library("scales")
4. library("lattice")
5. library("dplyr")
6. library("ggplot2")
7. library('networkD3')
8. library("highcharter")
9. library('sqldf')
10. library('forecast')
11. library(shiny)
12. library(googleCharts)

R server scripts

sever.R

```
library("leaflet")
library("RColorBrewer")
library("scales")
library("lattice")
library("dplyr")
library("ggplot2")
library('networkD3')
library("highcharter")
library('sqldf')
library('forecast')

# allow uploading large files ---
if (Sys.getenv('SHINY_PORT') == "")
  options(shiny.maxRequestSize = 10000 * 1024 ^ 2)

shinyServer(function(input, output, session) {
  # Map integration ---------------------------------

  output$map <- renderLeaflet({
    leaflet() %>%
      addProviderTiles(providers$CartoDB.Positron) %>%
      setView(lng = -1,
              lat = 47.45,
              zoom = 4)
  })

  zipsInBounds <- reactive({
    if (is.null(input$map_bounds))
      return(datainput1[FALSE, ])
    bounds <- input$map_bounds
    latRng <- range(bounds$north, bounds$south)
    lngRng <- range(bounds$east, bounds$west)

    subset(
      datainput1,
      latitude >= latRng[1] & latitude <= latRng[2] &
        longtitude >= lngRng[1] & longtitude <= lngRng
          [2]
    )
  })
```

```
centileBreaks <-
  hist(plot = FALSE,
       datainput1$cumulfreqnum,
       breaks = 20)$breaks

output$histCentile <- renderPlot({
  if (nrow(zipsInBounds()) == 0)
    return(NULL)

  ggplot(data = zipsInBounds(), aes(zipsInBounds()$
     cumulfreqnum)) +
    geom_histogram(breaks = centileBreaks,
                   alpha = .2) +
    labs(title = "Cumulative Frequency (visible
       countries)") +
    labs(x = "Cumulative Frequency of Sales", y = "
       Count")
})

output$scatterCollegeIncome <- renderPlot({
  if (nrow(zipsInBounds()) == 0)
    return(NULL)

  ggplot(zipsInBounds(),
         aes(x = sales_q, y = sales_c, color =
            cumulfreqnum)) + geom_point() + geom_rug()
             + xlab("Total Quantity") + ylab("Total
            Sales")
})

observe({
  colorBy <- input$color
  sizeBy <- input$size

  if (colorBy == "cumulfreqnum") {
    colorData <-
      ifelse(
        datainput1$cumulfreqnum >= input$threshold,
        "Over threshold",
        "Under threshold"
      )
    pal <-
      colorFactor("viridis", colorData) # "viridis", "
         magma", "inferno", or "plasma"
  } else {
```

```
    colorData <- datainput1[[colorBy]]
    pal <-
      colorBin("viridis", colorData, 7, pretty = FALSE
          ) # "viridis", "magma", "inferno", or "plasma
          "
  }

  radius <-
    datainput1[[sizeBy]] / max(datainput1[[sizeBy]]) *
        600000

  leafletProxy("map", data = datainput1) %>%
    clearShapes() %>%
    addCircles(
      ~ longtitude,
      ~ latitude,
      radius = radius,
      layerId = ~ country,
      stroke = FALSE,
      fillOpacity = 0.4,
      fillColor = pal(colorData)
    ) %>%
    addLegend(
      "bottomleft",
      pal = pal,
      values = colorData,
      title = colorBy,
      layerId = "colorLegend"
    )
})

showZipcodePopup <- function(country, lat, lng) {
  selectedZip <- datainput1[datainput1$country ==
     country, ]
  content <- as.character(
    tagList(
      tags$h4(tags$strong(HTML(
        sprintf("%s", selectedZip$country)
      ))),
      tags$br(),
      sprintf(
        "Cumulative frequency of Sales 2016: %s",
        as.integer(selectedZip$cumulfreqnum)
      ),
      tags$br(),
```

```
sprintf(
  "Total sales 2016: %s euros",
  format(
    as.integer(selectedZip$sales_c),
    big.mark = ",",
    scientific = FALSE
  )
),
tags$br(),
sprintf(
  "Total sales quantity 2016: %s kgs",
  format(
    as.integer(selectedZip$sales_q),
    big.mark = ",",
    scientific = FALSE
  )
),
tags$br(),
sprintf(
  "Total number of orders 2016: %s",
  format(
    as.integer(selectedZip$cnt),
    big.mark = ",",
    scientific = FALSE
  )
),
tags$br(),
tags$br(),
sprintf(
  "Population 2016: %s people",
  format(
    as.integer(selectedZip$pop2016),
    big.mark = ",",
    scientific = FALSE
  )
),
tags$br(),
sprintf(
  "Gross Domestic Product 2016: %s euros",
  format(
    as.integer(selectedZip$gdp2016),
    big.mark = ",",
    scientific = FALSE
  )
),
```

```
        tags$br()
      )
    )
    leafletProxy("map") %>% addPopups(lng, lat, content,
        layerId = country)
  }

  observe({
    leafletProxy("map") %>% clearPopups()
    event <- input$map_shape_click
    if (is.null(event))
      return()

    isolate({
      showZipcodePopup(event$id, event$lat, event$lng)
    })
  })

  # Network integration ---------------------------------

  output$force <- renderForceNetwork({
    forceNetwork(
      Links = datainput2links,
      Nodes = datainput2nodes,
      Source = "source",
      Target = "target",
      Value = "linkvalue",
      NodeID = "nodeid",
      Group = "nodegroup",
      opacity = 1,
      fontSize = 20,
      zoom = FALSE,
      linkDistance = JS("function(d){return d.value * 
          20}"),
      linkWidth = JS("function(d) { return Math.sqrt(d.
          value)*1.9; }"),
      legend = TRUE,
      colourScale = JS(
        'force.alpha(1); force.restart(); d3.
            scaleOrdinal(d3.schemeCategory20);'
      )
    )
  })

  # Highchart integration -------------------------------
```

```
# Calculate sales
diff13 <- reactive({
  if (input$metric == 'sales') {
    datainput0 %>%
      filter(as.integer(issueyear) == input$year[1])
        %>%
      group_by(rank = as.integer(substr(itemsizerank,
          1, 2)) + 1) %>%
      summarise(total = sum(sales)) %>%
      arrange(rank)
  } else {
    datainput0 %>%
      filter(as.integer(issueyear) == input$year[1])
        %>%
      group_by(rank = as.integer(substr(itemsizerank,
          1, 2)) + 1) %>%
      summarise(total = sum(quantity)) %>%
      arrange(rank)
  }
})

selected_years_to_print <- reactive({
  paste(input$year[1])
})

output$hcontainer <- renderHighchart({
  if (input$metric == 'sales') {
    hc <- highchart() %>%
      hc_add_series(
        data = diff13()$total,
        type = input$plot_type,
        name = "Sales",
        showInLegend = FALSE,
        color = "#7cb5ec"
      ) %>%
      hc_yAxis(
        title = list(text = "Sales (euros)"),
        allowDecimals = FALSE,
        max = 80000000
      ) %>%
      hc_xAxis(
        categories = c(
          "SPECIALCUTS",
          "STORTI",
```

```
            "150-200",
            "200-300",
            "300-400",
            "300-600",
            "400-600",
            "600-800",
            "500-1000",
            "800-1000",
            "1000-1500",
            "1000-2000",
            "1500-2000",
            "2000+",
            "2000-3000",
            "3000-4000",
            "4000+"
          ),
          tickmarkPlacement = "on",
          opposite = TRUE
        ) %>%
        hc_title(text = "",
                 style = list(fontWeight = "bold")) %>%
        hc_subtitle(text = paste("Total", input$metric,
           "per size for the year of",
                                  selected_years_to_print
                                     ())) %>%
        hc_tooltip(valueDecimals = 2,
                   pointFormat = "Item Size Rank (1-17)
                      : {point.x} <br> Sales: {point.y}
                      euros")
    } else {
      hc <- highchart() %>%
        hc_add_series(
          data = diff13()$total,
          type = input$plot_type,
          name = "Quantity",
          showInLegend = FALSE,
          color = "#90ed7d"
          ) %>%
        hc_yAxis(
          title = list(text = "Quantity (kgrs)"),
          allowDecimals = FALSE,
          max = 15000000
        ) %>%
        hc_xAxis(
          categories = c(
```

```
            "SPECIALCUTS",
            "STORTI",
            "150-200",
            "200-300",
            "300-400",
            "300-600",
            "400-600",
            "600-800",
            "500-1000",
            "800-1000",
            "1000-1500",
            "1000-2000",
            "1500-2000",
            "2000+",
            "2000-3000",
            "3000-4000",
            "4000+"
          ),
          tickmarkPlacement = "on",
          opposite = TRUE
        ) %>%
        hc_title(text = "",
                 style = list(fontWeight = "bold")) %>%
        hc_subtitle(text = paste("Total", input$metric,
            "per size for the year of",
                                 selected_years_to_print
                                   ())) %>%
        hc_tooltip(valueDecimals = 2,
                   pointFormat = "Item Size Rank (1-17)
                       : {point.x} <br> Quantity: {point.
                       y} Kgrs")
    }

    hc
})

# Highchart integrations
    ------------------------------------


# point 9 forecast
output$highchartforecast <- renderHighchart({
  a01 = sqldf(
    "Select issueyearmon, sum(sales) sales from
       datainput3 where issueyearmon <> '2017-08'
       Group By issueyearmon order by issueyearmon"
```

```
    )
    a01$ts = ts(
      a01$sales,
      start = c(2013, 1),
      end = c(2017, 7),
      frequency = 12
    )
    d.arima <- auto.arima(a01$ts)
    x <- forecast(d.arima, level = c(95, 80), h = 12)
    hchart(x)
  })

  # point 2 pie chart
  output$highchart2 <- renderHighchart({
    a02 = sqldf(
      "Select 'Sales inside Greece', sum(sales) sales
          from datainput3 where country = 'GREECE' union
          Select 'Sales outside Greece', sum(sales) sales
          from datainput3 where country <> 'GREECE' "
    )
    a02$saleRatio = as.numeric(format(round(100 * a02$
        sales / sum(a02$sales), 2), nsmall = 2))
    highchart() %>%
      hc_title(text = "") %>%
      hc_subtitle(text = "Percentage of Total Sales from
          Jan.2013 to Aug.2017") %>%
      hc_add_series_labels_values(
        c(
          paste(as.character(a02$saleRatio[[1]]), '%
              inside Greece'),
          paste(as.character(a02$saleRatio[[2]]), '%
              outside Greece')
        ),
        a02$saleRatio,
        type = "pie",
        name = "Percentage of sales",
        colorByPoint = TRUE,
        size = 200,
        color = c('#7cb5ec', '#bfbfbf')
      )
  })

  # point 3 area chart
  output$highchart3 <- renderHighchart({
    a03 = sqldf(
```

```
    "Select issuemonth Month, sum(sales) Sales, sum(
      quantity) Quantity, sum(sales)/sum(quantity)
      Price from datainput3 where issueyear != '2017'
      group by issuemonth"
  )

  highchart() %>%
    hc_subtitle(text = "Average Sales, Quantity and
      Selling Price per Month from Jan.2013 to Dec
      .2016") %>%
    hc_xAxis(
      plotBands = list(
        list(
          from = 4,
          to = 8,
          color = "rgba(100, 0, 0, 0.1)",
          label = list(text = "44.62% of Total Sales")
        )
      ),
      categories = c(
        'Jan',
        'Feb',
        'Mar',
        'Apr',
        'May',
        'Jun',
        'Jul',
        'Aug',
        'Sep',
        'Oct',
        'Nov',
        'Dec'
      )
    ) %>%
    hc_add_series(name = "Average Sales (euros)", data
      = a03$Sales/4) %>%
    hc_add_series(name = "Average Quantity (Kgrs)",
      data = a03$Quantity/4) %>%
    hc_add_series(name = "Average Price (euros/kgrs)",
      data = a03$Price)
})

# point 5 area chart
output$highchart5 <- renderHighchart({
  a051 = sqldf(
```

```
    "Select issueyearmon YearMon, sum(sales) Sales,
       sum(quantity) Quantity, sum(sales)/sum(quantity
       ) Price from datainput3 where itemgroup='
       Seabream' and issueyearmon != '2017-08' group
       by issueyearmon"
  )
  a052 = sqldf(
    "Select issueyearmon YearMon, sum(sales) Sales,
       sum(quantity) Quantity, sum(sales)/sum(quantity
       ) Price from datainput3 where itemgroup='Meagre
       ' and issueyearmon != '2017-08' group by
       issueyearmon"
  )

  highchart() %>%
    hc_subtitle(text = "Trend of Selling Price and
       Quantity for Seabream and Meagre from Jan.2013
       to Jul.2017") %>%
    hc_xAxis(categories = a051$YearMon,
             plotBands = list(
               list(
                 from = 28,
                 to = 32,
                 color = "rgba(100, 0, 0, 0.1)",
                 label = list(text = "Top 5 Selling
                    Prices")
               )
             )) %>%
    hc_add_series(name = "Seabream Selling Price (
       euros/kgr)", data = a051$Price) %>%
    hc_add_series(name = "Meagre Selling Price (euros/
       kgr)", data = a052$Price) %>%
    hc_add_series(
      name = "Seabream Quantity in (Million kgrs)",
      data = a051$Quantity /
        1000000,
      type = 'area',
      color = '#d9d9d9'
    ) %>%
    hc_add_series(
      name = "Meagre Quantity in (Million kgrs)",
      data = a052$Quantity /
        1000000,
      type = 'area',
      color = '#d9d9d9'
```

```
    )
})

# point 6 area chart
output$highchart6 <- renderHighchart({
  a06fillet = sqldf(
    "Select issueyearmon YearMon, sum(sales) Sales,
       sum(quantity) Quantity, sum(sales)/sum(quantity
       ) Price from datainput3 where itemcategory='
       Fillet' and issueyearmon != '2017-08' group by
       issueyearmon"
  )
  a06gutted = sqldf(
    "Select issueyearmon YearMon, sum(sales) Sales,
       sum(quantity) Quantity, sum(sales)/sum(quantity
       ) Price from datainput3 where itemcategory='
       Gutted' and issueyearmon != '2017-08' group by
       issueyearmon"
  )
  a06whole = sqldf(
    "Select issueyearmon YearMon, sum(sales) Sales,
       sum(quantity) Quantity, sum(sales)/sum(quantity
       ) Price from datainput3 where itemcategory='
       Whole' and issueyearmon != '2017-08' group by
       issueyearmon"
  )
  a06 = sqldf(
    "Select itemcategory, sum(sales) Sales, sum(
       quantity) Quantity, sum(sales)/sum(quantity)
       Price from datainput3 where issueyearmon !=
       '2017-08' group by itemcategory"
  )
  a06$quantityRatio = as.numeric(format(round(
    100 * a06$Quantity / sum(a06$Quantity), 2
  ), nsmall = 2))
  a06$SalesRatio = as.numeric(format(round(100 * a06$
     Sales / sum(a06$Sales), 2), nsmall = 2))
  highchart() %>%
    hc_chart(type = "line") %>%
    hc_subtitle(text = "Trend of Selling Price per
       Category from Jan.2013 to Jul.2017") %>%
    hc_xAxis(categories = a06gutted$YearMon) %>%
    hc_add_series(name = "Fillet (euros/kgr)", data =
       a06gutted$Price) %>%
    hc_add_series(name = "Gutted (euros/kgr)", data =
```

```
        a06fillet$Price) %>%
      hc_add_series(name = "Whole (euros/kgr)", data =
        a06whole$Price)  %>%
      hc_add_series_labels_values(
        a06$itemcategory,
        a06$quantityRatio,
        type = "pie",
        name = "Percentage of total Quantity",
        colorByPoint = TRUE,
        center = c('65%', '38%'),
        size = 70,
        dataLabels = list(enabled = FALSE)
      ) %>%
      hc_add_series_labels_values(
        a06$itemcategory,
        a06$SalesRatio,
        type = "pie",
        name = "Percentage of total Sales",
        colorByPoint = TRUE,
        center = c('35%', '38%'),
        size = 70,
        dataLabels = list(enabled = FALSE)
      )
  })

  # point 7 area chart
  output$highchart71 <- renderHighchart({
    a071 = sqldf(
      "Select 'Top 10 Countries in Sales' label, 100*sum
        (sales_c)/599716802 percent from datainput1_
        legacy where cumulfreqnum > 12 union Select '
        Rest of the Countries' label, 100*sum(sales_c)/
        599716802 percent from datainput1 where
        cumulfreqnum <= 12 "
    )
    a071$percent = as.numeric(format(a071$percent,
       digits = 4))
    highchart() %>%
      hc_subtitle(text = "Total Sales and Quantity per
        Country from Jan.2013 to Dec.2016") %>%
      hc_chart(type = "column") %>%
      hc_plotOptions(column = list(
        dataLabels = list(enabled = FALSE),
        stacking = "normal",
        enableMouseTracking = FALSE
```

```
    )) %>%
    hc_xAxis(categories = datainput1_legacy$code,
       plotBands = list(
      list(
        from = 28,
        to = 37,
        color = "rgba(100, 0, 0, 0.1)",
        label = list(text = "")
      )
    )) %>%
    hc_add_series(name = "Total Sales (euros)", data =
       datainput1_legacy$sales_c) %>%
    hc_add_series(name = "Total Quantity (kgr)", data
      = datainput1_legacy$sales_q) %>%
    hc_add_series_labels_values(
      a071$label,
      a071$percent,
      type = "pie",
      name = "Percentage of Sales (%)",
      colorByPoint = TRUE,
      center = c('35%', '38%'),
      size = 100,
      color = c('#7cb5ec', '#90ed7d')
    )
})

output$highchart72 <- renderHighchart({
  datainput1_legacy$salesGroup = ifelse(
    datainput1_legacy$cumulfreqnum > 12,
    'Top 10 Countries in Sales',
    ifelse(
      datainput1_legacy$cumulfreqnum <= 12 &
        datainput1_legacy$cumulfreqnum > 1,
      'Countries with Medium Sales',
      'Countries with Less Sales'
    )
  )

  highchart() %>%
    hc_subtitle(text = "Comparing 2016 GDP and Total
       Population to Annual Average Sales per Country"
       ) %>%
    hc_chart(zoomType = "xy") %>%
    hc_add_series(
      dataLabels = list(enabled = TRUE,
```

```
                                format = "{point.label}"),
          data = datainput1_legacy,
          hcaes(
            x = gdp2016,
            y = pop2016,
            z = sales_c/4,
            group = salesGroup
          ),
          color = c('#434348', '#7cb5ec', '#90ed7d'),
          type = "scatter"
        ) %>%
        hc_tooltip(
          useHTML = TRUE,
          headerFormat = "<table>",
          pointFormat = paste(
            "<tr><th colspan=\"1\"><b>{point.label}</b></
                th></tr>",
            "<tr><th>Country</th><td>{point.country}</td><
                /tr>",
            "<tr><th>GDP 2016</th><td>{point.x} euros</td
                ></tr>",
            "<tr><th>Population 2016</th><td>{point.y}
                people</td></tr>",
            "<tr><th>Total Sales</th><td>{point.z} euros</
                td></tr>"
          ),
          footerFormat = "</table>"
        )
    })

})
```

global.R

```
datainput1 = readRDS("data/datainput1.rds") # <> 2017

datainput2links = readRDS("data/datainput2links.rds")
datainput2nodes = readRDS("data/datainput2nodes.rds")

datainput0 = readRDS("data/datainput0.rds")

# data3trans2, data3country = source tables in local
    postgres/project2

# reduce granularity from date to month
datainput3 = readRDS("data/datainput3.rds")
years <- unique(as.integer(datainput0$issueyear))
# point 7
datainput1_legacy = readRDS("data/datainput1_legacy.rds"
    )
```

Application screen shots

Geographic analysis and insights into Mariculture

In 2014, world per capita fish supply reached a new record high of 20 kg . Aquaculture remains an important source of food, nutrition, income and livelihoods for hundreds of millions of people around the world. As fish is one of the most-traded food commodities worldwide with more than half of fish exports by value originating in developing countries, this fact raises a number of questions about the aquaculture industry in general.

Which countries import the largest amounts of fish? What are the processing steps of a saltwater fish company from production to sale? Which sizes and categories are in high demand? In which months the fish is sold with the higher price? What is the prediction for the future production and sales?

This report intends to answer these questions using a detailed analysis based on data from a Major European Company Group*.

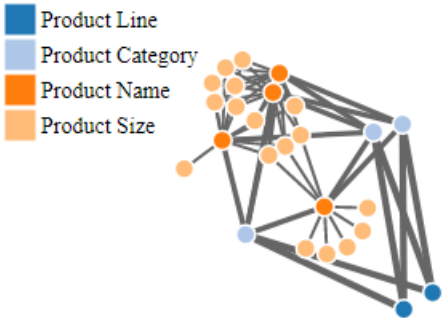
Figure 5.1: Part 1: Introduction

1 An introduction to Company's production line and products

The production is separated in two main production lines: the Fresh and the Frozen fish. At the first step of the process, when the fish reach the appropriate sizes, they are taken for further processing.

Thus, the fish are separated in three categories: (a) Gutted, (b) Fillet and (c) Whole. Its category contains its products and every product its own sizes. Same products may exist in more than one categories.

The company produces 4 types of fish and 17 distinct sizes in total.



- Product Line: Fresh and Frozen fish.
- Product Category: Gutted, Fillet and Whole
- Product Name: Red Seabream, Seabass, Seabream and Meagre.
- Product Size: 0-200, 200-300, 300-400, 300-600 etc.

Finally, according to the production line that are going follow, fish are packed and prepared for dispatch. The diagram describes the possible paths of every fish based on its size and type (Product Name).

Figure 5.2: Part 2: Story point 1

Figure 5.3: Part 2: Story point 2

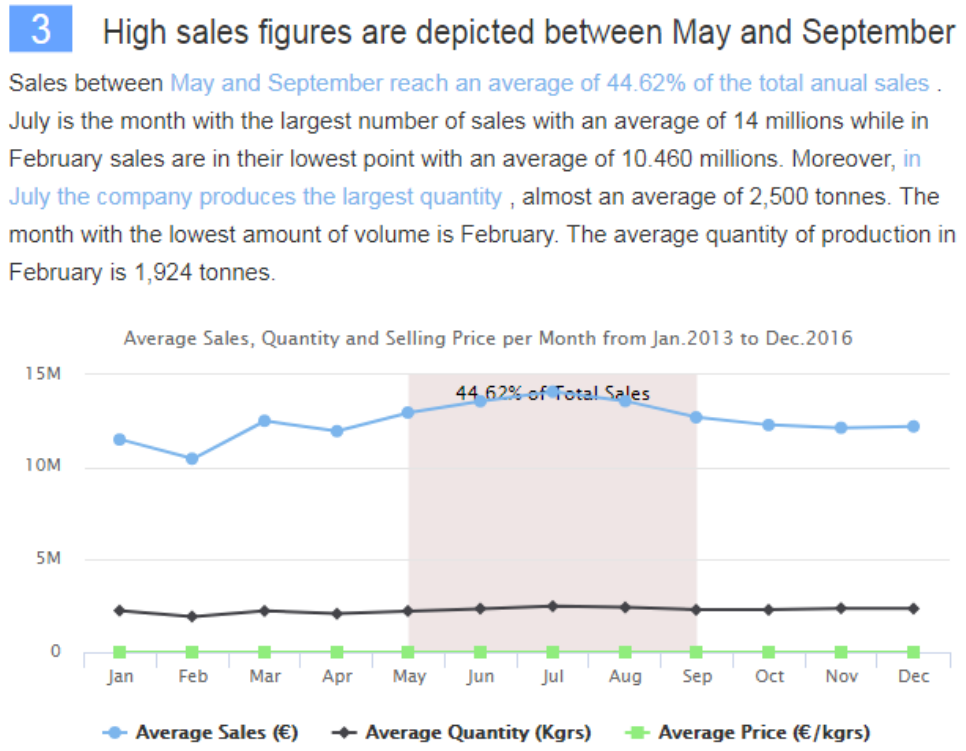


Figure 5.4: Part 2: Story point 3

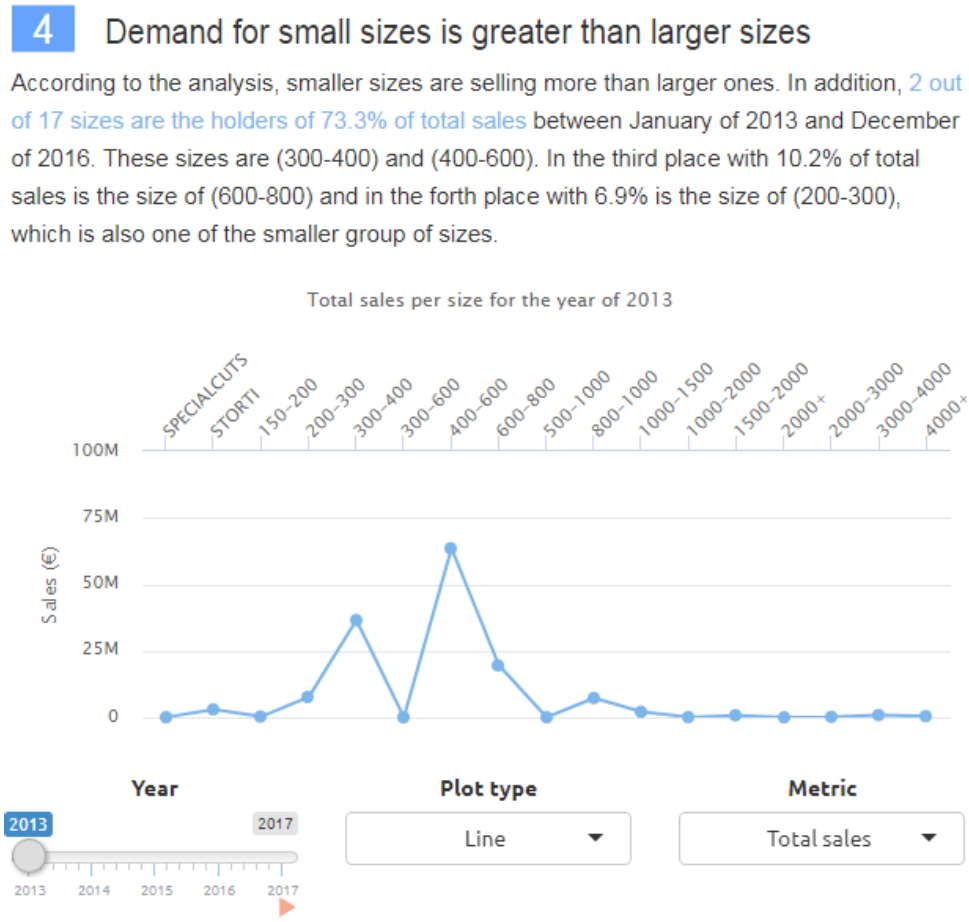
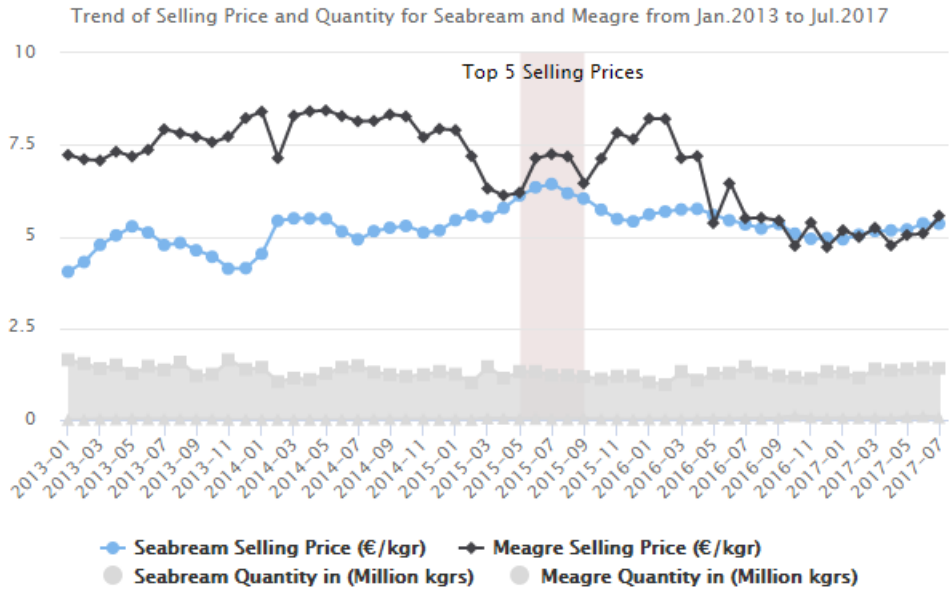


Figure 5.5: Part 2: Story point 4

5 In 2015 Seabream had a raise of ~ 7 billion € in sales

In 2015, total production of Seabream was less than previous years. However, in the same year, the selling price of Seabream reached its maximum value with an average of 5.84€/kg. Moreover, in July of 2015 the selling price was at its highest value between January 2013 and July. The top 5 average selling prices per month for Seabream where from May to September of 2015 (6.22€/kg). This indicates that the demand for Seabream is increasing over the years.



On the other hand, the selling price of Meagre has been decreasing over the last months. This confirms a previous conclusion that small sizes have a greater demand than larger ones , as sizes for Meagre range from (500-1000) to (4000+).

Figure 5.6: Part 2: Story point 5

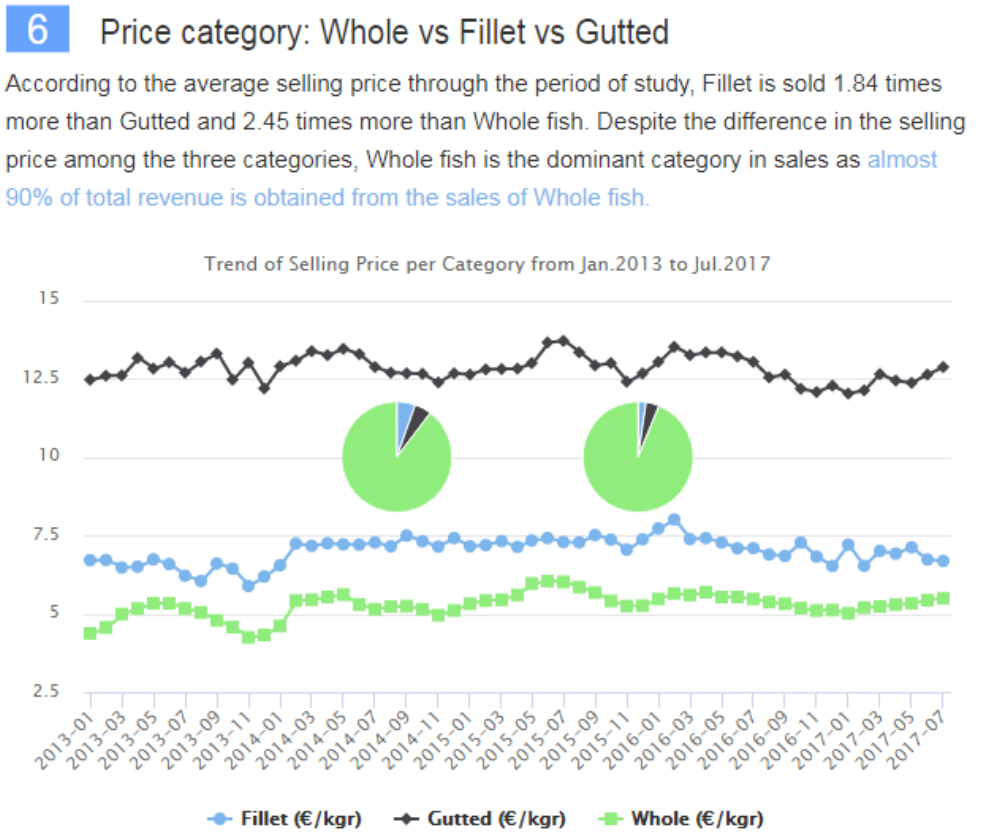


Figure 5.7: Part 2: Story point 6

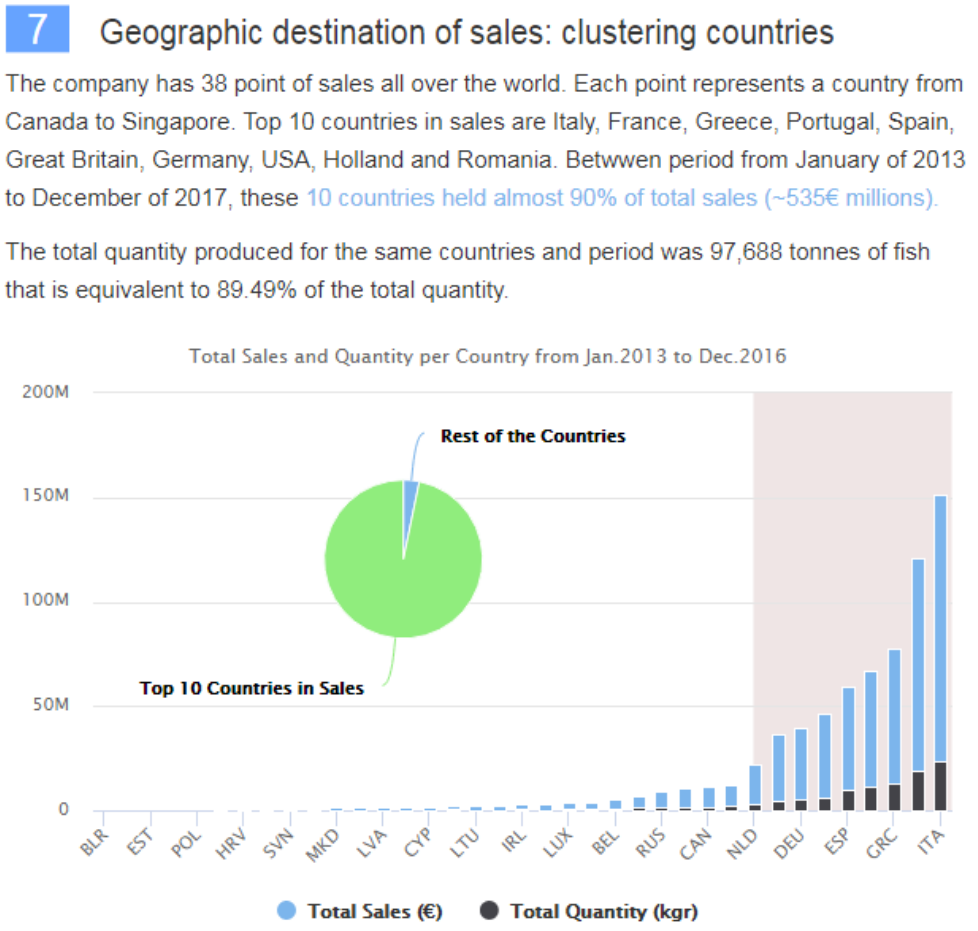
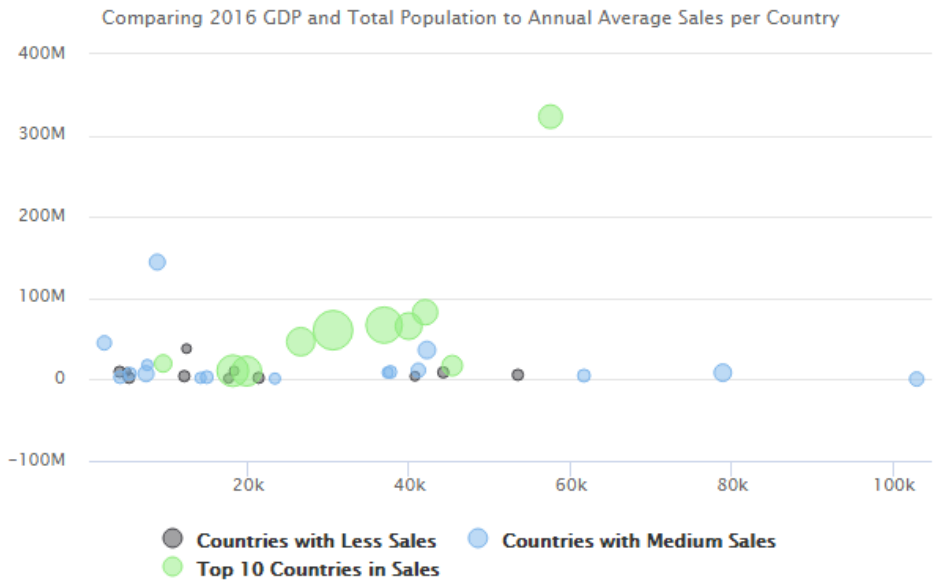


Figure 5.8: Part 2: Story point 7 (1)

A cluster analysis of countries characteristics shows that Russia, although has a GDP close to Romania and population 8 times more than Romania's, it is grouped with countries with medium sales. A further analysis shows that in August of 2014 Russia stopped purchasing fish products from the company. One reason for this is that this might be caused by Russia's embargo that bannned European Union food imports in August of 2014. And that is also the case with Ukraine.



Another coclusion from this cluster analysis is that although Greece, Portugal, Esthonia and Chech Republic have great similarity in terms of GDP and total population, they are grouped in different clusters. Greece and Portugal belong to the first group whereas Esthonia and Chech Republic belong to the second. This is also due to the fact that Greece is where company's head office are located and Portugal has a very high rate in sales (4rth place) whereas sales figures for Esthonia and Chech Republic show a discontinuous demand for fish.

Figure 5.9: Part 2: Story point 7 (2)

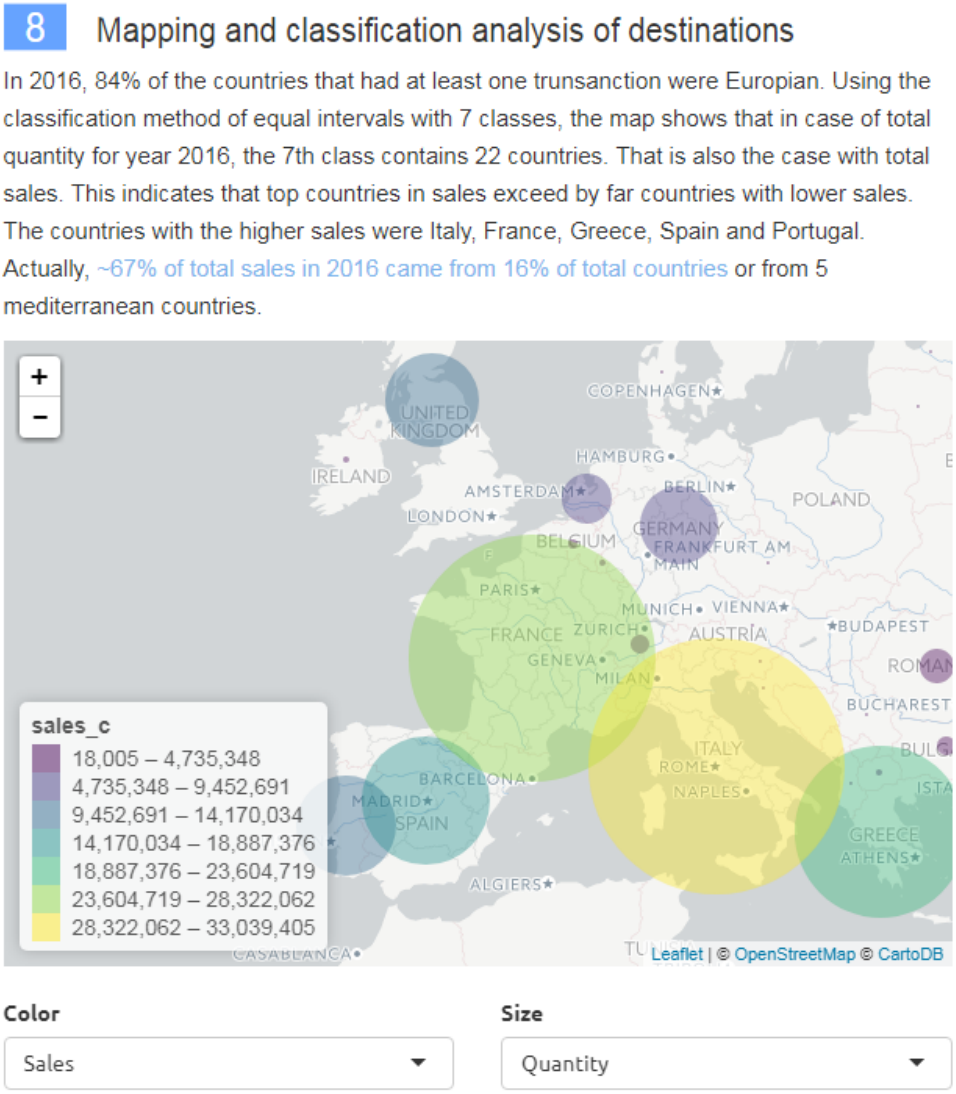
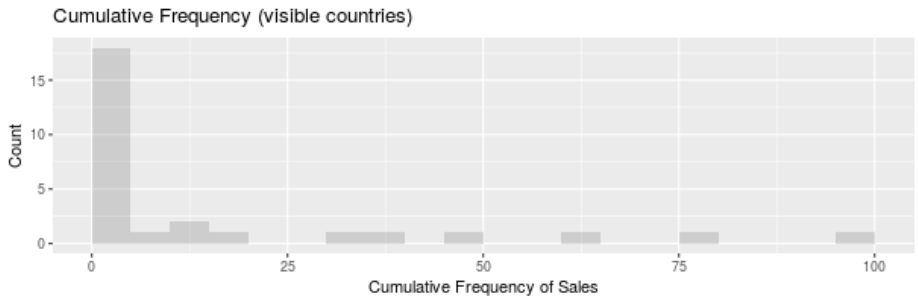
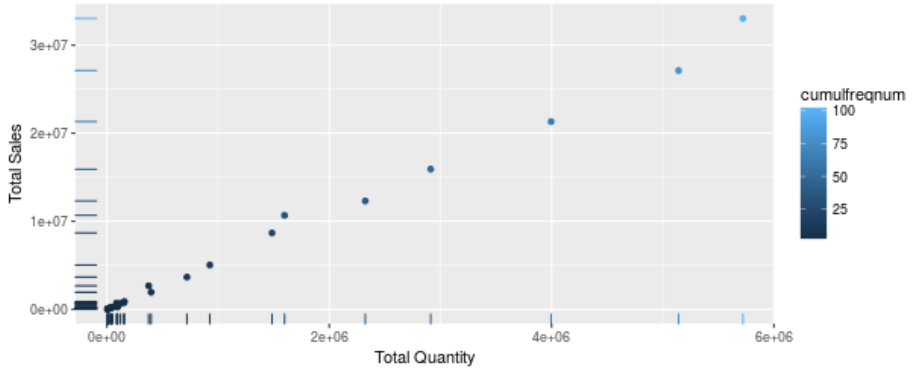


Figure 5.10: Part 2: Story point 8 (1)



Analyzing sales from January 2013 to December 2016 , numbers showed that these 5 mediterranean countries , which in this case represent 13,5% of total countries, held 66% of total sales.



According to countries classification, the top 5 countries belong to five different clsses . The scatter plot describes the reason of this result. The top 5 five countries on the upper right of the plot are shown to have larger distances among them whereas countries with lower sales and quantity are very close to each other and thus can be easily grouped together.

The cumulative frequency of sales, which is defined as the sum of all previous frequencies up to the current point when frequencies are ordered from the smallest to the largest, indicates the small contribution of countries with lower sales to total sales.

Figure 5.11: Part 2: Story point 8 (2)

Figure 5.12: Part 2: Story point 9

Conclusion

In summation, in view of this analysis, Marineculture is a dynamic growing sector and can contribute measurably to the completion of the increasing demand for food in the next decades.

The main conclusions of this analysis are briefly summarised below:

- Countries that import and consume the majority of these specific types of fish products are the mediterranean countries (Italy, France, Greece, Spain and Portugal).
- The consumers prefer fish categories with smaller sizes than categories with larger sizes.
- Selling prices are higher during summer months, especially in July.
- The demand for Marineculture products will grow in the near future.

*The data have been collected from an Aquaculture Company Group located in Greece and refer to information about the orders of the company, amount of retail sales and quantity of fish, in the period between 1.1.2013 and 22.8.2017. Latest update: 24.9.2017

**FAO, (2016). The State of World Fisheries and Aquaculture 2016, FAO.Contributing to food security and nutrition for all. Rome, 200 pp.

Figure 5.13: Part 3: Conclusion