VOLUME PREDICTION IN RAIL CARGO FOR OPTIMAL RESOURCE UTILISATION

by

PEDRO CADAHIA DELGADO

A thesis submitted in conformity with the requirements
for the MSc in Economics, Finance and Computer Science

University of Huelva, International University of Andalusia & DB Cargo AG

September 2016

Volume Prediction in rail cargo for optimal resource utilization

Pedro Cadahía Delgado

Máster en Economía, Finanzas y Computación. Especialidad Marketing y Big data.

Dr. Juan Manuel Bravo Caro & Dr. Martin Bodestedt
Universidad de Huelva, Universidad Internacional de Andalucía & DB Cargo AG

2016

Abstract

«The increasing competitiveness of the market forces companies to adapt to new times, the implementation of automated predictive models could be a critical business opportunity.

The automatic program developed in the project uses simple and univariate models to face this challenge and later to verify to what extent it brings an improvement in the company.

The study concluded that the developed program analyzes the data sets efficiently, analyzing hundreds of different combinations and models per time series.

Besides, the methodology saved analytical time compared with traditional methods in addition it implied improvements in predictions accuracy against the company's manual forecasts.»

JEL classification: C22, C53, C55, C63.

Key words: Arima, Sarima, Exponential smoothing, time series, cargo, logistics, SWT, forecasting.

Resumen

«La creciente competitividad del mercado hace que las empresas tengan que adaptarse a los tiempos actuales, la implementación de modelos predictivos automatizados podrían suponer un elemento diferenciador de negocio.

El programa automático desarrollado en el proyecto usa modelos simples y univariantes para afrontar este desafío y posteriormente comprobar hasta qué punto aporta una mejora en la empresa.

Se concluyó que el programa desarrollado analiza los conjuntos de datos de forma eficiente, analizándose por serie temporal cientos de combinaciones y modelos diferentes.

Además, la metodología desarrollada supuso ahorros de tiempo analítico comparado con los métodos tradicionales y supuso también mejoras en predicciones frente a los datos disponibles por la empresa.»

Clasificación JEL: C22, C53, C55, C63.

Palabras Clave: Arima, Sarima, Suavizado exponencial, Series temporales, Cargamento, logística, SWT, Predicción.

Acknowledgments

« I would first like to thank my thesis advisor Dr. José Manuel Bravo of Department of Electronic Engineering of Computer Systems and Automatic at University of Huelva. He consistently allowed this Project to be my own work, but steered me in the right the direction whenever he thought I needed it, as a result I grew up because of this project.

I would like to thank the person who got me the job: Juan Manuel Muro, also I would like to thank the experts who were involved in some way in this project: Dr. Martin Bodestedt, Dr. Timo Schürg and Dr. Matthias Dremmer.

Without their passionate participation and input, the project could not have been successfully conducted.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Pedro Cadahía Delgado»

Agradecimientos

«En primer lugar quisiera dar las gracias a mi asesor de proyecto Dr. José Manuel Bravo del Departamento de Ingeniería Electrónica de Sistemas Informáticos y Automáticos de la Universidad de Huelva. Siempre ha permitido que este proyecto sea mi propio trabajo, pero me llevó en la dirección correcta cada vez que pensó que lo necesitaba, como resultado crecí debido a este proyecto.

Me gustaría agradecer a la persona que me contactó para el trabajo: Juan Manuel Muro, también me gustaría agradecer a todos los expertos involucrados en el proyecto: Dr. Martin Bodestedt, Dr. Timo Schürg y el Dr. Matthias Dremmer.

Sin su participación y contribución apasionada el proyecto no podría haber sido un éxito.

Por último, debo expresar mi profunda gratitud a mis padres por proporcionarme un apoyo infalible y un estímulo constante a lo largo de mis años de estudio y mediante el proceso de investigación y redacción de este proyecto. Este logro no habría sido posible sin ellos. Gracias.

Pedro Cadahía Delgado»

Table of Contents

CHAPTER 1 – INTRODUCTION

"An approximate answer to the right problem is worth a good deal more than an exact answer to an approximate problem."

John Tukey.

1.1 PREVIOUS STUDIES

This thesis resumes the previous project in **DB Cargo AG**, which involved a data exploration and some forecasting approaches.

As a result, the project forecast had better quality in more than **56**% of cases than the data planning. On average, the difference of all forecasts values were lower than planning values. This was a remarkable result, considering that a lot of employees of DB Cargo AG are involved in the planning. Nevertheless, it was pointed out that planning in certain areas gives better estimates for the next year. By supplying information that is not currently detected, the forecast could be further improved.

Currently, the actual value of the **last year** is proposed. The system would suggest the estimated value of the DB Cargo employees and this could adjust the value manually, if necessary. It could be improved by combining the planning and forecasting to get a significant improvement planning.

The **decision tree model** developed in the previous study should be further evaluated and validated over the years on. This means that it should be examined whether the decision tree is still the optimal prediction model and the areas in which the prognosis is better than planning. (Adrian Schmitt, Predictive Analytics zur Absatzplanung im Schienengüterverkehr, DB Cargo Study, April 2016. p.1-8)

The starting point of the current project is implementing the different forecasting models which can predict the net tonnes of sales planning in a good enough quality, also the **Big Data** approach is a factor to implement in this project.

1.2 CONTEXT

In recent decades, the performance of computer systems has been rising in an exponential way, to the same extent also the storage of large amounts of data has increased. Store gigabytes of data volume are more affordable than ever. On the one hand the hardware of computer systems is becoming cheaper; on the other hand, the technical development has increased enormously.

This automatic processing of data volume due to the recent high business competency is more important than ever, the traditional ways are not always affordable because is not possible to analyse in deep detail a massive amount of data by one or more employees with the artisanal procedures. Hence, the efficiency and power of computation suppose a high plus and differentiation very required nowadays to maintain the market share, this is the big data concept applied in one part of the logistics business.

The aim of this thesis is to examine the question whether it is possible to draw from the past actual values demand by planning a mathematical model with a good quality.

Traditional methods for the analysis of data doesn't match with the interest of tracking down information fail due to the enormous amount of data. Therefore, special models can be selected, which can deal with a big volume of data. These are grouped under the notion data mining or big data analytics.

Data mining is used to analyse large volumes of data and thereby uncover patterns and statistical anomalies. A portion of data mining is the **predictive analytics**, which deals with the prediction of future values and trends. Several predictors are combined into a predictive model. Data are collected, created a statistical model, driven predictions and the model validated or amended as appropriate. Predictive analytics is currently being used in many research and application areas such as meteorology, security, economy and marketing.

This thesis has been written in cooperation with DB Cargo AG. This company is part of the Group of Deutsche Bahn AG and is responsible for the **rail freight**, one of the largest European railway companies. In the fiscal **year 2015,** the sales were around **15.5 billion Euros**. This performance was achieved by each day drove average **4,742 trains** of the DB Schenker Rail Germany AG and a total of **329 million tons** of goods were transported. (DB Schenker. “Turnover of the year 2015”. Dbschenker.de. 03/09/2016. <www.dbschenker.de/log-de-en/company/keyfigures.html>)

1.3 RESEARCH OBJECTIVES

It is examined the question: Can values be predicted in most or at least in some cases? This question should be answered by trying to find a better model for the examined areas than the current data plan.

In order to implement a revenue management strategy, combined with necessary resizing measures, the sales forecast on low aggregation levels must be increased, for instance by a clearer incentivisation strategy, to aid this process the provisioning of high quality predictions of future volumes at low-medium aggregation levels plays a central role.

1.4 BUSINESS BACKGROUND

Currently, the **data** of the **sales**, **information and planning system** are planned by **200 employees** of DB Cargo AG. The planning of the previous data suggested by the system can then be adjusted **manually** by the staff. In this procedure, no statistical models are used. Firstly, they propose values without previous existing data. On the other side, possible significant impacts of independent variables for example **type of goods** (**Güterart**) could not have been taken into account. If the amount of the transported goods are constantly decreasing or increasing over the years then there is a delay in the depiction of this development due to the current system. The sales planning prediction is an important but complex procedure in the rail freight.

As an example, the independent variable type of goods listed. Sinks or rises in transport volumes every year continuously, this can be shown delayed in its current form only. The forecast for sales planning in rail freight transport is an important and complex process. The importance to the Company arises from the fact that some of the staff force (200 employees) has to be expended for the current sales planning. In addition, a mismatch between actual and planned values exists, thus the company's development can be poorly planned and costs may increase. The complexity of demand planning is based from the set of factors and number of records. Two values appear in connection with the sales planning.

- The **actual values** (**Ist-Werte**) are the real values of the DB Cargo AG and reflect the actually driven and services provided.

- The **plan values** (**plan-Werte**) are planned by the staff of DB Cargo AG values for the next year.

Currently, this process usually is fully manual, with a forecast quality ranging from very poor to quite good due to several reasons:

- For **account managers** tending to a high number of low-volume customers are at a disadvantage due to the heterogeneous structure of the traffics.
- Some **markets** are in nature very volatile leading to difficulties in proper forecasting (for instance rail construction site business).
- Variance of service quality by supporting functions (i.e. controlling) lead to a variance in the quality of the manual sales forecasts.
- A different **incentive strategy** in different industrial sectors leads to varying motivation level to deliver high quality forecasts.

The aforementioned reasons lead to an overall low quality of sales forecasts at low aggregation levels (i.e. dispatch station to receiving station level). Therefore, these forecasts cannot be used by the planning department for sizing and resizing of the single wagon network. Instead, the resizing is done using an adjusted past data.

1.5 LOGISTICS BACKGROUND

Nowadays the rail freight transport is made by train, where cargo is carried on wagons that are towed by one or more locomotives. The longer the train the lower the unit cost, because it makes better use of the rail wagon network.

Besides, the railroad has **18%** in the national transport market in terms of tons, and especially in the number of freight. (Eurostat. “Freight transport statistics – modal split”. 03/09/2016.< http://ec.europa.eu/eurostat/statistics-explained/index.php/Freight_transport_statistics_-_modal_split >).

At the moment, there is a hard competition into the logistics freight transport business by the Chinese in the steel market, hence is where the steel transport volumes are getting segregated into a more shared market with other logistics competitors. By trying to predict a specific transport carrying steel, the general trend of decreasing steel volumes should be accounted for.

The business knowledge is very important to save time and understand the problem, some concepts shown below will be critical in the first steps: data preparation and exploration.

Single Wagonload traffic is the main concept in this project, the main target will be the forecasting amount in the network that involves this concept. This kind of network faces in many countries in Europe to profitability and quality problems, with the difficulties to adapt to the actual changing market requirements. However, the Wagonload still forms the backbone of rail freight.

1.5.1 SWL Definition:

Single Wagonload Traffic: Transport of freight in individual railway wagons or groups of wagons (the shipment is less than a whole trainload).

1.5.2 SWL Features:

- The **SWL** supply includes grouping and sorting of wagons in order to assembly full trains with different shipments, in order to take advantage of the full train size and, thus, increasing the productivity.

- **Grouping** / **sorting** can take place through marshalling in dedicated yards where each train is disassembled and the groups of wagons are classified to form new full load trains for the next yard, or more simplified arrangement with removal / addition of groups of wagons at intermediate stops.

- Any kind of wagons including the one loaded with combined transport units can be moved in **SWL** supply chain.

1.5.3 Rail Cargo and Network:

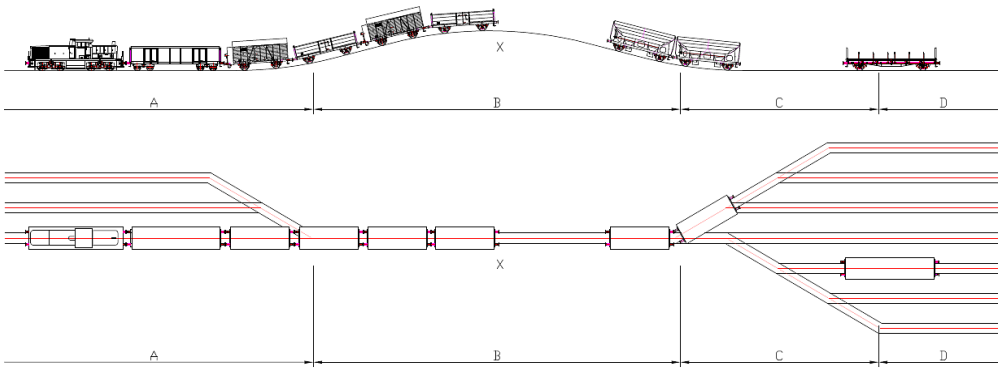
The most important commodities moved by Single Wagonload traffic are: raw materials and chemicals and heavy industry. The total SWL volumes in the main European countries are **75 billion tonnes per km** and an estimated total in EU+ CH with **80-83 billion tonnes per km**. (PWC & University of Rome.“Study on single wagonload traffic in Europe”. European rail freight days, November 2014).

Picture 1: *Structures of SWL. Connected hubs, 2-level connected Hubs and Corridor.*

Basically, the SWL are wagons moving from the **Sending station** (**Versaland**) through some points using several trains to achieve the **Receiving station** (**Empfangsland**), but it is also important to explain the mechanism to classify the cargo in all these tracks.

This activity is managed in the **Classification Yard** or **Marshalling Yard** (American/British English) also known in German as **Ablaufberg,** essentially is a railway yard found at some freight train stations, used to separate railway cars on to one of several tracks. First the cars are taken to a track, sometimes called a lead or a drill. From there the cars are sent through a series of switches called a ladder onto the classification tracks. Larger yards tend to put the lead on an artificially built hill called a hump to use the force of gravity to propel the cars through the ladder.

Freight trains that consist of isolated cars must be made into trains and divided according to their destinations. Thus, the cars must be shunted several times along their route in contrast to a unit train, which carries. This shunting is done partly at the starting and final destinations and partly in classification yards.



Picture 2: Plan and elevation of **Classification Yard-Ablaufberg** – (source: Wikipedia.de)

To sum up, the main objective of the project is to set some basic targets to improve the business by doing predictions in different aggregation levels like Single wagon transport and higher level

of aggregation such as tracks and regions to find some patterns and optimize the resource utilization of rail cargo.

However, the main target is not only to find the best model prediction that has never existed at first attempt, but also it means to improve some bad managing performances between the Sales and Production department by trying to implement the data decision as a link in the management chain. As previously said in the different description levels of data, the main purpose is not only improving the business efficiency with better forecasts but also systematizing the analysis into the IT System Department.

Nowadays due to the recent high business competency it is needed the automatic processing of big data volume, the traditional way is not always possible to analyse in deep detail a massive amount of data by one or more employees with the artisanal procedures. Hence, the efficiency and power of computation suppose a high plus and differentiation very required nowadays to maintain the market share, this is the **Big Data** concept applied in one part of the logistics business.

1.6 POSSIBLE USES

As previously said the approach of the project was conceived into a Big Data style. Besides, this solution can be also applied to other several and nonrelated problems, without considering if it is the best solution, but as a good enough approach.

- All the **logistics** world prediction: Loads, number of trains, number of wagons, kind of goods, supply and demand, flights, shipments...
- **Economic world:** sales index, purchases index, market rates...
- **Environment:** weather, harvesting production, population dynamics...

And so on, but the most important thing to highlight is the automatic approach of the solution, with an input file containing data, dates and identifications of observations, the solution offered in this project will provide the best solution within a lot of models considered. Very important in these days when the data amount is massive and the difficulty to process all the information is critical to find a fast solution.

CHAPTER 2 – DATA AND METHODOLOGY

"Without data you're just another person with an opinion"

W. Edward Deming

2.1 DATASET

The data were provided by the company DB Cargo AG from a complex database, the most information of the study was extracted from "**VIPS**" (**Information and Planning System**) which was analysed with data mining methods to create a forecasting model of the data distribution. The whole project was developed in **Sales Controlling System** department of DB Cargo AG.

Controlling is a part of the business management system whose main task is to plan, manage and control all business areas. This is done by capturing, processing and evaluating all data in the enterprise.

One area of controlling is the sales planning. Sales planning is multidimensional (regions, customers, products), which leads to a high complexity. The heart of the marketing plan is the plan-actual comparison. The following performance-based variables are displayed:

- Planned costs
- Actual costs
- Variances of the planned costs of the actual costs
- Variances percentage

The VIPS includes revenue and performance information for rail freight of the company over the period from **2001** to **2014**. Besides, both the real and the revenue performance information planned by the sales department are listed annually in advance. However, in order to create the models, only the actual data sets are used.

Some dataset features:

- These represent the actually driven relations.
- There are **1924679** actual records from the years **2001** to **2014**.
- There appears a little amount of missing values or invalid entries.

As previously mentioned, the application VIPS is the control and planning system of DB Cargo AG for distribution. Further, it is an information system for different user groups (management, production, controlling) within the company. The system includes earnings and performance of

transactions that are settled by machine to customers. In other words, the basis for the values of the invoiced transport orders and the revenues for rendered services. This information can be evaluated according to different criteria and structures. The actual information is taken from an accounting procedure. In addition to the presentation of results the data base includes a planning module, in which the benefits anticipated from the rail transport revenue and performance data are mapped.

The data basis for the results view in VIPS consists on "**the settlement procedure**". It contains in addition detailed waybill information, the calculated revenue data from the cargo loading and billing as well as statistical values and information for the allocation of revenue to revenue-responsible market area. The data is transferred on a provisional, daily presentation of results and a final, monthly presentation of results from the payroll systems. These actual values are compared to plan and forecast figures. The planned and projected values were set by the relevant vendor in the system. The data fields are based on the previous year's actual values. All value fields are the responsible salesperson (competence arises from the customer assignment to Seller) determined. It should be noted that only the share of revenue is shown in VIPS, which are the company DB Cargo AG attributed.

Also, called **pre-cartloads** (**Vorbeifuhren**) are shown in the data base. This is to transport that do not touch the territory of the Federal Republic of Germany (for example, internal transport in France or a transport from Austria to Italy). Also for this traffic revenue and performance information are provided.

2.1.1 Independent Variables

- **Time**: temporal component. Monthly, bimonthly or yearly basis, in principle, the fiscal year begins in January and ends in December of the same year.
- **Market area (Industry Sector):** The customer segments (branches) distributed at DB Schenker Rail Germany AG in six market areas.
- **Shipping location (*Versand*) and receiving location (*Empfangsort*)**:

This exists in various relations. The word "**Relation**" in German means **connection**/**route**/**stretch** between the points. These can be expressed in each of the following ways:

- **Versandland - Empfangsland** / **Ship from Country - Ship to Country** (**36** Countries)
- ***Versand Produktionszentrum - Empfang Produktionszentrum*** **/ Ship from Production Centre - Ship to Production Centre** (**9** production centers and number of **classification**/**connection**/**allocation**)
- **Versand Standort - Empfang Standort** / **Ship from location- ship to location** (**45** locations)
- **Versandbahnhof - Empfangsbahnhof** / **Ship from station** – **ship to station** (**7000** transport stations)

Picture 3: Standort of DB Schenker in Germany (*Source: DB website*)

- **Type of goods**: There are three variables with different characteristics:
  - **20 NST** (Standard Goods Nomenclature for Transport Statistics) formerly known as goods sector (**Güterbereich - GB**)
  - **81 NST** (Standard Goods Nomenclature for Transport Statistics) formerly known as goods group (**Gütergruppe - GGR**)
  - **NHM 4** (Harmonized goods nomenclature) formerly known as (**Gutartnummer- GNR**)

- **Tarifnummer/Tariff number**: indicates which tariff has been negotiated.

- **Eigentumsverhaltnis/Ownership**: Railway or Railroad car.
- **Erlöskunde/Customer service**: Customer, to whom the service is allocated by sales engineering.
- **Grenzübergang/Border crossing**: Border and border crossing.
- **Produkt/Product**: As previously explained is the concept of single wagon load traffic (**Konventioneller Wagenladungsverkehr**) from A to B.

2.2.2 Dependent variables

These are the variables that are planned by the employees of the company. The dependent variables to be predicted are:

- **Tariftonnenkilometer (ttkm)/Net tonne kilometre (ntkm):**

  Net tonne kilometre is the summation of every one tonne moved one kilometre. NTKM takes into account the change in loadings as well as distance between stops (tariff points).
- **Nettotonnen/Net tons (nto)**: weight of the load without car weigh driven by relationship. This contrasts with the gross tons (**Bruttotonnen**).
- **Wagenanzahl/Number of Wagons (wg).**
- **Nettoerlös (Neterl)/Net sales**: Revenue from driven relationship.

2.2 AGGREGATION DATA LEVEL

Because of the massive amount of data, the project has been abridged by narrowing the huge topic down to something fit for the Research. It has been done by identifying traffics of customers with low forecast quality and different logistic stations of study. The data used to the study was divided in two levels of aggregation:

2.2.1 Data aggregation

This term means “the way the values are shown”, it depends on the scope of the logistics value location procedure:

- **Produkt/Customer level**

As previously explained in the SWL, the goods are moved between two stations dispatcher and receiver, all this information is in the string variable named "**relation**", also the number of tonnes of Iron from a company transported in a time aggregation lever per relation (**see point *2.2.2***), also known as "**nto**". Each relation will be considered as a different time series used in the prediction, so the target for this data is the forecasting of goods per relation.

- **Bediensegment/Segment level**

In the Despatching data base are recorded all the wagons with different goods transported from the dispatching station (outcomes). On the contrary, In the Receiving databases are registered all incoming wagons also composed of different goods.

In both cases the forecasting models are focused on predicting the values of the outcomes and incomes to each station, in fact the information has been processed isolated like a different case of prediction study.

The data are from **2001** to **2014**, all the information related to the SWL was provided by DB Cargo AG.

2.2.2 Time Aggregation

Sometimes the granularity of the structure of the data is a big problem in to analyse, identify some patterns and catch the general sense of the data. However, a huge structure could catch to much information n order to perceive the trend and cycles of the data, not only data aggregation level is important to do an accurate prediction but also a correct time step is critical.

The data time step is considered in a monthly and bimonthly basis to catch some patterns weak/strong months like winter and summer stages, in the case that some data trend were conditioned with the season. Besides, the data values are weighted in order to resize them, so it is important to rescale the influence of the holydays in the data.

2.3 CRISP-DM METHODOLOGY

Due to the features of the methodology this project use the **CRISP-DM**, the following point is going to describe the methodology implemented into the project. There are different methods

that every data scientist should know; in fact, some of the most important are **KDD**, **CRISP-DM** and **SEMMA** methods.

SEMMA is shorter methodology than the aforementioned methods and it is implemented with SAS software. However, CRISP -DM methodology is larger than SEMMA and is more focused on Data mining development process and not directed into business objectives as SEMMA method and it is implemented with SPSS software.

CRISP-DM is a European Union project conceived under the **ESPRIT** funding initiative in led by five companies: **SPSS**, **Teradata**, **Daimler AG**, **NCR Corporation** and **OHRA**.

Daimler Chrysler (then Daimler-Benz) was already ahead of most industrial and commercial organizations in applying data mining in its business operations. SPSS (then ISL) had been providing services based on data mining since 1990 and had launched the first commercial data mining workbench (Clementine) in 1994. NCR, as part of its aim to deliver added value to its Teradata data warehouse customers, had established teams of data mining consultants and technology specialists to service its clients' requirements.

This methodology provides a structured approach to planning a data mining project.

Picture 4: CRISP-DM Methodology (*Source: IBM Manual*)

This model is a sequence of events, but almost the totally of the tasks can be performed in a different order and it will often be necessary to backtrack to previous tasks and repeat certain actions. The model does not try to capture all possible routes through the data mining process.

It is shown more information about each phase of the process here:

2.3.1 Business understanding

This initial phase focuses on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a data mining problem definition and a preliminary plan designed to achieve the objectives.

A) **Set objectives** - This means describing the primary objective from a business perspective.
B) **Produce project plan** – Here it is described the plan for achieving the data mining and business goals.
C) **Business success criteria** - Here are set up the criteria to determine whether the project has been successful from the business point of view.

2.3.2 Data understanding

The data understanding phase plays a central role because the wrong interpretations makes always a bad starting point to achieve the goals of the project, this phase starts with initial data collection and proceeds with activities that enable you to become familiar with the data, identify data quality problems, discover first insights into the data, and/or detect interesting subsets to form hypotheses regarding hidden information.

This stage may include:

A) Distribution of key attributes
B) Relationships between pairs or small numbers of attributes
C) Results of simple aggregations
D) Properties of significant sub-populations
E) Simple statistical analyses

2.3.3 Data preparation

The data preparation phase includes all activities needed to form up the final dataset, in other words the data that will be run into the modeling part. Data preparation tasks are feasible to be run into multiple times and not in any recommended order. Tasks would include table, record, and attribute selection, as well as transformation and cleaning of data for modeling tools.

2.3.4 Modeling

To ensure the success several modeling techniques must to be selected and applied to the data, also their parameters must to be calibrated to optimal values. Some techniques have specific requirements on the form of data. Therefore, going back to the data preparation phase is often necessary.

A) Generate test design

The first step before constructing the model is to generating a procedure or mechanism to test the model's quality and validity. For instance, in supervised data mining tasks such as classification, it is common to use error rates as quality measures for data mining models. Therefore, it is typically separated the dataset into train and test sets, build the model on the train set, and estimate its quality on the separate test set.

B) Build model

This step consists on running the modelling tool on the prepared dataset to create one or more models.

B.1) Parameter settings - With any modelling tool there are often a large number of parameters that can be adjusted. List the parameters and their chosen values, along with the rationale for the choice of parameter settings.

B.2) Models - These are the actual models produced by the modelling tool.

2.3.5 Evaluation

During this step, it is assessed the degree in which the model meets the business objectives. In the negative case, is important seek to determine if there is some business reason why the model is performing deficiently. Besides, another ideal option is to test the model on test applications in the real application, if time and budget constraints permit.

The evaluation phase also involves assessing any other data mining results obtained in the process. Data mining results involve models that are necessarily related to the original business objectives and all other conclusions that are not necessarily related to the original business objectives.

A) Review process

At this phase, the resulting models are satisfactory and satisfy business needs. It is appropriate to do a more thorough review of the data mining assurance in order to determine if there is any important factor that has somehow been overlooked.

2.3.6 Deployment

Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that is useful to the customer. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data scoring or data mining process. In many cases it will be the customer, not the data scientist, who will carry out the deployment steps. Even if the analyst deploys the model it is important for the customer to understand up front the actions which will need to be carried out in order to actually make use of the created models.

A) Produce final report

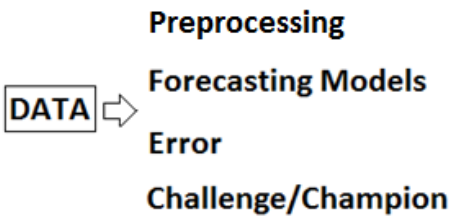
At the end of the project you will write up a final report. Depending on the deployment plan, this report may be only a summary of the project and its experiences or it may be a final and comprehensive presentation of the data mining results.

B) Review project

Assess what went right and what went wrong, what was done well and what needs to be improved.

2.4 CODE'S DESING

At the following picture, it is sketched the structure of the script that get in an automatic way a massive volume of temporal series and calculate all their error to specify which is the best, very important to understand the complexity of the project:



Picture 5: Concept of the script implemented in R.

In automate terms, the more temporal series someone manage the more handicaps will find to have all exceptions under control, also will contribute the different aspects of each time series. However, they were solved by setting some assumptions and requirements in the computation scripts.

2.4.1 Data Pre-processing

Data pre-processing is a data mining technique that involves transforming raw data into an appropriate format. In Real life data is often incomplete, inconsistent, and/or lacking in certain behaviours or trends, and is likely that contain many errors.

Data pre-processing is a proven method of resolving such issues. Data pre-processing prepares raw data for further processing, a critical part to ensure the accuracy of a predictive model.

Tasks in data pre-processing:

- Data cleaning: fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
- Data integration: using multiple databases, data cubes, or files.
- Data transformation: normalization and aggregation.
- Data reduction: reducing the volume but producing the same or similar analytical results.

- Data discretization: part of data reduction, replacing numerical attributes with nominal ones.

2.4.2 Data partition

Partitioning data into training and testing set is an important part of the evaluation of forecasting models. Typically, when you separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing. Analysis Services randomly samples the data to help ensure that the testing and training sets are similar. By using similar data for training and testing, you can minimize the effects of data discrepancies and better understand the characteristics of the model.

After a model has been processed by using the training set, you test the model by making predictions against the test set. Because the data in the testing set already contains known values for the attribute that you want to predict, it is easy to determine whether the model's guesses are correct.

By default, after having defined the data sources for a mining structure, the Data Mining Wizard will divide the data into two sets: one with **80%** of the source data, for training the model, and one with **20%** of the source data, for testing the model. This default was chosen because **80-20%** ratio is often used in data mining, but with Analysis Services you can change this ratio to suit your requirements.

2.4.3 Overfitting

Before to building up the predictive models it is important not to overlook some theoretical aspects.

The training phase of algorithms tries to find the best solution of a mathematic expression for the training data. In this fitting process, the various parameters are optimized until the "best" fit is found. The terms overfitting and generalization play a central role.

Often, the algorithms tend to memorize the training data, which is called an over-adjustment. Besides, the over-adjustment takes place at the expense of generalization for previously unknown instances and often occurs with a small training data. Another reason for an over-adjustment is the noise in the variables, whereby the blurring in the data has a strong influence on the functional context.

2.4.4 Forecasting Models

A time series is a collection of observations made variable sequentially in time, in which the observation order is important. Time series values are linked to time instants, therefore a time series analysis involves handling set of two variables; the variable itself of study and the time variable.

The following techniques of time series analysis extract the patterns observed in the past of the variable. In addition, everything considered under the assumption that the structural conditions that make the series under study remain constant.

It is highly thought that the amount of data is directly proportional to the quality of the forecast of a model. "*The longer a temporal series is, the better will be*". However, this is sometimes a wrong concept because "*The more data, the more noise*", it means that is becoming an unforecastable time series.

The type of analysis and the models on which we base the study will depend largely on the type of questions we want to answer. When the historic data correspond to a single variable, time

series analysis often aims to build a model to explain the structure, forecast trends and seasonality of the variable of interest.

To reach the targets, it is established as basics of the study some **Standard Simple Models** and **Univariate Models.**

2.4.3.1 Standard Simple Models

The study begins with a first approach to get the forecasting values with some simple predictive models that sets the maximum error allowed by a more complex chosen model.

- **Average method**

$$y'_t = (\sum_{t=1}^{t} y_t)/T$$

Using the average method, all future forecasts are equal to a simple average of the observed data,

- **Naïve method**

$$y'_t = y_{t-1} + e$$

The naïve method assumes that the most current observation is the only important one and all previous observations provide no information for the future. This can be thought of as a weighted average where all the weight is given to the last observation.

Consequently, each model gives a yield that every model has to pass to be considered as a forecasting improvement.

2.4.3.2 Univariate Models

As previously described in the Data set in **chapter 2**, the first study was launched with univariate series, which means to be autocorrelated with itself, which means that in some way each value is correlated with the past data in any degree, depending on the model chosen.

To forecast this series, it was used the most common and effective univariate models, grouped in two blocks:

- Autoregressive models (**AR**), Moving average models (**MA**), Autoregressive models-Moving average models (**ARMA**), Autoregressive Integrated Moving Average model (**ARIMA**) and Seasonal Autoregressive Integrated Moving Average model (**SARIMA**).
- **Exponential Smoothing** (Simple, Double and Triple Exponential smoothing).

2.4.3.3 Overview of the time series models

Time series was previously defined as "*collection of observations made variable sequentially in time, in which the observation order is important*", but it be also described as a stochastic process, i.e. an ordered sequence of random variables, where the time index t takes on a finite or countable infinite set of value.

- **AR Model**

The notation AR(p) refers to the autoregressive model of order p. The following formula is the model AR(p):

$$y'_t = c + e_t + \sum_{j=1}^{p} \varphi_j \, y_{t-j}$$

where the c is the constant, e_t is the white noise error term and φ_j is the parameter of the model.

- **MA Model**

The notation MA(q) refers to the moving average model of order q. The following formula is the model MA(q):

$$y'_t = \mu + e_t + \sum_{i=1}^{q} \theta_j \, e_{t-i}$$

where the θ_q is the parameter of the model, μ is the expectation of y_t (often assumed to equal 0), and the e_t is the white noise error term.

- **ARMA Model**

AutoRegressive Moving Average models are a class of stochastic processes expressed as (Box and Jenkins, 1970):

$$y'_t = c + e_t + \sum_{j=1}^{p} \varphi_j \, y_{t-j} + \sum_{i=1}^{q} \theta_j \, e_{t-i}$$

Where the parameters are: **φ** and **θ** are model parameters; p and q are the orders of the AutoRegressive (**AR**) and Moving Average (**MA**) processes respectively.

- **ARIMA Model**

If is combined differencing with autoregression and a moving average model, the result is a non-seasonal ARIMA model. The full model ARIMA(p,d,q) can be written as:

$$y'_t = -(\Delta^d y_t - y_t) + \varphi_0 + \sum_{i=1}^{p} \varphi_i \Delta^d y_{t-i} - \sum_{j=1}^{q} \theta_i e_{t-i} + e_t$$

It is called ARIMA (p,d,q) model, where the parameters are: p, order of the autoregressive part; d, degree of first differencing involved and q is the order of the moving average part.

- **SARIMA Model**

A SARIMA model can be written as follows: **ARIMA (p,d,q) ARIMA(P,D,Q)s**

$$y'_t = \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \cdots + \varphi_{Ps+p+Ds+d} y_{t-Ps-p-sD-d} + \delta + U_t + \theta_1 U_{t-1} + \cdots + \theta_{Qs+q} U_{t-sQ-q}$$

Where the second parenthesis (uppercase) is the **seasonal part** of the model and **s** is the number of periods per season.

- **Exponential Smoothing** (Simple, Double and Triple Exponential smoothing).

To do a brief description of the exponential smoothing we will classify as a prediction model that can be classified as:

  - **Simple exponential smoothing**

The simplest of the exponentially smoothing methods is naturally called “simple exponential smoothing” (**SES**). This method is suitable for forecasting data with no trend or seasonal pattern.

When the data do not display any clear trending behaviour or any seasonality, but the mean of the data may be changing slowly over time. It has already considered the naïve and the average as possible methods for forecasting such data.

The smoothing parameter of the SES is alpha (α), where it gets values from $0 \leq \alpha \leq 1$. If α is small (i.e., close to 0), more weight is given to observations from the more distant past. If α is large (i.e., close to 1), more weight is given to the more recent observations.

The equation is the following:

$$y'_t = \alpha y_t + \alpha (1 - \alpha) y_{t-1} + \alpha (1 - \alpha)2 y_{t-2} + \cdots + \alpha (1 - \alpha)n y_{t-n}$$

- **Double exponential smoothing**

This method is used when the data shows a trend. Exponential smoothing with a trend Works much like simple smoothing except that two components must be updated each period level and trend. The level is a smoothed estimate of the value of the data at the end of each period. The trend is a smoothed estimate of average growth at the end of each period. The specific formula for simple exponential smoothing is:

$$y'_t = \alpha y_t + \alpha (1- \alpha)(y_{t-1} + b_{t-1}) \qquad 0 \leq \alpha \leq 1$$

$$b_t = \beta(y_t - y_{t-1}) + (1- \beta) b_{t-1} \qquad 0 \leq \beta \leq 1$$

If β is small (i.e., close to 0), more weight is given to observations from the more distant past. If α is large (i.e., close to 1), more weight is given to the more recent observations.

- **Triple exponential smoothing**

This method is used when the data shows trend and seasonality. To handle seasonality, we have to add a third parameter (**ɣ**). It is introduced a third equation to take care of seasonality. The resulting set of equations is also known as "**Holt-Winters**" method after the names of the inventors.

There are two main HW models, depending on the type of seasonality.

• Multiplicative Seasonal Model

• Additive Seasonal Model

This model is referred by the following formula:

$$y'_t = b_t + b_{2t} + S_t + e_t$$

Where $\mathbf{b_1}$ is the base signal also called the permanent component, $\mathbf{b_2}$ is a linear trend component, $\mathbf{S_t}$ is an additive seasonal factor and $\mathbf{e_t}$ is the random error component

The seasonal factors are defined so that they sum to the length of the season, i.e.

Adding the seasonal component:

$$St = \gamma(y_t - S_t) + (1 - \gamma)\, S_{t-L}$$

Where $0 < \gamma < 1$ is the third smoothing constant. If γ is small (i.e., close to 0), more weight is given to observations from the more distant past. If α is large (i.e., close to 1), more weight is given to the more recent observations.

2.4.5 Error Measurement

Depending on the data, each model performs in a different way, but before to classify the result of a predictive model as good or bad it is important to set which error measurement we are going t use.

Summing up all reviews and books on the subject, there are more than 25 models and more than 15 error measures. In fact, Error measurement plays a big role in tracking forecast accuracy, monitoring exceptions, and benchmarking the forecasting process. Interpretation of these statistics can be tricky, particularly when trying to assess accuracy across multiple items (e.g., SKUs, locations, customers, etc.).

The next questions to solve in the predictive modelling are:

- Are the models equally effective in predicting even if they fit the historical data?

- How much accuracy exist and in which cases each apply?

- What other criteria exist for the selection of forecasting models?

There is significant evidence that a model fits well to historical data, does not necessarily predict well.

But, why do we evaluate the accuracy of a forecast model based on their ability to fit historical data? And also important, are there alternatives to evaluate a model based on their ability to forecast? Hence, one alternative is to divide the information into two sets as described in the data partition point.

For all the considered models, it is necessary to measure the performance of them with accuracy indicators. Not all have the same meaning and the same use, but both are based on the following formula:

$$e_t = y_t - y'_t$$

Where e_t is the forecast error, and y_t the observed or actual value at time t of the time series, and y'_t is the predicted value at time t of the time series.

The error can also be represented in relative and/or absolute terms (%) using the following formula:

$$ea_t(\%) = \frac{|y_t - y'_t|}{y_t} * 100$$

Also in quadratic form:

$$e_t{}^2 = (y_t - y'_t)^2$$

The next table shows the measures most commonly used error where F_t is the forecast previously called y'_t and Y_t is the actual value previously called y_t, the notation used to identify the predicted value. It is shown more than 15 tools to measure forecast error. For example, by choosing the absolute error rather than quadratic, it is got a penalized more the big values of the error.

Medida de error	Fórmula
MSE Mean Square Error	$Media\{e_t^2\}$
RMSE Root Mean Square Error	$\sqrt{MSE}$
MAE Mean Absolute Error	$Media\{e_t\}$
MdAE Median Absolute Error	$Mediana\{e_t\}$
MAPE Mean Absolute Percentage Error	$Media\{p_t\}$
MdAPE Median Absolute Percentage Error	$Mediana\{p_t\}$
sMAPE Symmetric Mean Absolute Percentage Error	$Media\left\{2 \cdot \frac{\lvert Y_t - F_t \rvert}{Y_t + F_t}\right\}$
sMdAPE Symmetric Median Absolute Percentage Error	$Mediana\left\{2 \cdot \frac{\lvert Y_t - F_t \rvert}{Y_t + F_t}\right\}$
MRAE Mean Relative Absolute Error	$Media\{r_t\}$
MdRAE Median Relative Absolute Error	$Mediana\{r_t\}$
GMRAE Geometric Mean Relative Absolute Error	$MediaG\{r_t\}$
RelMAE Relative Mean Absolute Error	MAE / MAE^*
RelRMSE Relative Root Mean Squared Error	$RMSE / RMSE^*$
LMR Log Mean Squared Error Ratio	$\log(\mathrm{Re}lRMSE)$
PB Percentage Better	$100 \cdot Media\{I\{r_t < 1\}\}$
PB(MAE) Percentage Better (MAE)	$100 \cdot Media\{I\{MAE < MAE^*\}\}$
PB(MSE) Percentage Better (MSE)	$100 \cdot Media\{I\{MSE < MSE^*\}\}$

Table 1: Forecasting error measurement, Table adapted Gooijer and Hyndman (2005).

To conclude with the error measurement the most relevant formulas are described below:

2.4.4.1 Akaike's Information Criterion

AIC error penalizes the complexity of the model taking into account the number of variables and is used to select the best model within the same data set. The method of Box & Jenkins has this feature as they use actual and previous values of the independent variables to produce accurate short-term forecasts. The solution given by Akaike is to choose as a function of loss (or criteria specification) the minimum information criterion. Besides, in this project the use of AIC error is discarded because is not useful to interpret different models.

$$AIC = \sqrt{\frac{\sum_{t=1}^{n} e_t^2}{n}} * \exp(\frac{2 * k}{n})$$

2.4.4.2 Mean Absolute Percentage Error

The 100*MAPE (Mean Absolute Percentage Error) measures the size of the error in percentage terms. It is calculated as the average of the unsigned percentage error, as shown in the example below:

$$\frac{100}{N}\sum_{t=1}^{N}\frac{|y'_t - y_t|}{y_t}$$

Many organizations focus primarily on the MAPE when assessing forecast accuracy. Most people are comfortable thinking in percentage terms, making the MAPE easy to interpret. It can also convey information when you don't know the item's demand volume. For example, 4% is more meaningful than the amount of 3,283 cases.

The MAPE is scale sensitive and should not be used when working with low-volume data. Notice that because "Actual" ($\boldsymbol{y_t}$) is in the denominator of the equation, the MAPE is undefined when Actual demand is zero. Furthermore, when the Actual value is not zero, but quite small, the MAPE will often take on extreme values. This scale sensitivity renders the MAPE close to worthless as an error measure for low-volume data.

2.4.4.3 Symmetric Mean Absolute Percentage Error

Symmetric Mean Absolute Percentage Error (SMAPE) is an alternative to MAPE when there is zero or near-zero demand for items. SMAPE self-limits to an error rate of 200%, reducing the influence of these low volume items. Low volume items are problematic because they could otherwise have infinitely high error rates that skew the overall error rate. SMAPE error measurement is expressed in the following formula:

$$\frac{2}{N}\sum_{t=1}^{N}\frac{|y'_t - y_t|}{y'_t + y_t}$$

To Sum up, the error most used is the MAPE because it is easy to interpret, a well-known measure very used by the forecasters. But it favours the forecasts that are below the actual

values. To avoid this problem, there is an error measurement gaining a lot of supporters, the SMAPE. But it also has an unwanted behaviour when the actual value or forecast value is very close to zero.

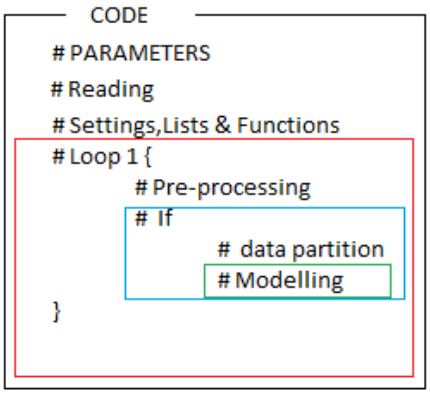
2.4.6 CHALLENGE/CHAMPION

Challenge/champion's task is to evaluate all models for a time series automatically by the criterion selected in parameterisation part. This error measure is applied in the test part of the time series to quantify the accuracy, as a result this criterion is chosen to discern among all models to choose a model as the best one (see point ***2.4.2 "data partition"***). Because it is not interesting to keep all models except the one that predicts better, By using the lowest error as a criterion is an efficient way to save memory and keep the rest unsaved.

2.5 CODE'S OPERATION

The aim of the project is more to get a practical and useful tool to analyse a massive amount of data than a scientific research approach, not only the predictions are a result of the project but also the code is a result by itself, is one of the products of the project.

As previously sketched in the **picture 5,** it was explained the main idea of the script, but in order to understand better how performs the code it is necessary to go through the code in detail by explaining technically how it works and some more pertinent information to understand the output data. Also, to zoom-in in more detail, at the **picture 6** is shown the technical script map to have an overview of the points to explain.



Picture 6: Predictive code map

2.5.1 Parameterisation

The parameterisation is the part where some variables are set at the top of the code and makes potentially stronger to the users who don't know about the code, programming, time series modelling, etc.

Besides, it is really useful in order to avoid changing the code in several parts spread in the whole document, in fact usually the developer is doing more projects at the same time and with this tool the developer ensures program's functionality.

At the **picture 7** it is set the first step of the parameterisation, where it is possible to change the strings to allocate and read a csv file.

Immediately in the following module there are some parameters to set the time series properly like:

- **timesteps_ per_year**: if the time series has **6** values per year (**bi monthly** basis), **12** values per year (**monthly**) among other possibilities (**quarterly**, **daily**…).
- **sequencystep**: parameter used inside the modelling part to catch the seasonality.
- **trainingpercentage**: By default is **0.8** (80%) as explained in the ***chapter 2***, there is the option to change the percentage.
- **errormeasure**: The value is a **string**; it could be either "**MAPE**" or "**RMSE**" as a criterion of error to choose the wining model.

```
############################## PARAMETERS ###############################################

# File Parameters #
namefile<-"TS_test"       # Name of the file
locationfile<- "C:/Users/Usuario/Desktop/"  # location file

# Time series Setting #
timesteps_per_year= 6     # 12/level aggregation i.e: {"1m","2m","3m","Y"} = {12,6,4,1}
sequencystep= "2 months"  # string argument: "month","2 months", "3 months", "Year"
trainpercentage= 0.8      # numeric between 0-1 (0% -100%)
errormeasure= "MAPE"      # Error in measurement. options: "MAPE","RMSE"
```

Picture 7: R-script's parameterisation part.

The third module is the modelling parameterisation, where the user set the values to go through the different model combinations. In the modelling part of the ***chapter 2*** was explained the theory of the different models implemented in the code.

To sum up, there are 3 different models with several parameters:

- **Arima**(**p**,**d**,**q**): Which means there are $\boldsymbol{n^r}$ combinations (but values of differencing ***d*** gets values between 1-2.
- **Sarma**(**p**,**d**,**q**,**P**,**D**,**Q**): Which means there are $\boldsymbol{n^r} * \boldsymbol{n^r}$ combinations (but we can only choose combinations by summing values of differencing ***d*** and **D** that gets values between 1-2.
- **Exponential smoothing** (**α**, **β**, **γ**): Which means there are different combinations depending on the sequence step (*hw_a_by*, *hw_b_by*, *hw_g_by*).

```
arima_p_from = 0 # Must be an integer.
arima_p_by =   1 # allways use = 1 for arima. Must be integer.
arima_p_to =   timesteps_per_year # usually is <= timesteps_per_year
arima_d_from = 0 # must be an integer.
arima_d_by =   1 # allways use = 1 for arima. Must be integer.
arima_d_to =   2 # allways smaller than 2. Must be integer.
arima_q_from = 0 # must be an integer.
arima_q_by =   1 # allways use = 1 for arima. Must be integer..
arima_q_to =   timesteps_per_year # usually is <= timesteps_per_year
```

Picture 8: R-script's parameterisation part (model's variables).

2.5.2 Reading file, lists & Functions

2.5.2.1 Reading file

The R script reads a csv file that has the following structure as shown in the **table 2.**

Ts	time	value	kpi
Ts 1	01.02.2001	1234	act
Ts 1	01.04.2001	1235	act
Ts 1	01.06.2001	1236	act
Ts 1	01.08.2001	1237	act
Ts 1	01.10.2001	1238	act
Ts 1	01.12.2001	1239	act

Table 2: *Ts1*, default data format of a csv file.

- **ts_id**: Identification of the temporal series.

- **time**: By default is set the format "***dd.mm.yyyy***", necessary to run the code.
- **value:** The historic data is saved in this column.
- **kpi**: This column has 3 possible strings "***act***" (actual value), "***mfc***" (manual forecasts) and "***plan***" (plan value) to classify the data and measure the distance between the prediction and the plan value or manual forecasts.

R script saves the csv into a **data frame**, a genuine object of R language.

Definition: A **data frame** is a table, or two-dimensional array-like structure, in which each column contains measurements of one variable, and each row contains one case. Unlike an array, the data stored in the columns of a data frame can be of various types. I.e., one column might be a numeric variable, another might be a factor, and a third might be a character variable. All columns have to be the same length (contain the same number of data items, although some of those data items may be missing values).

2.5.2.2 Lists Storage

The lists are used as empty boxes to full of the code's results (outputs) and generally they are going to be "***lists of lists***" or "***lists of numeric***".

2.5.2.3 Functions

As a good practice, some functions are set to make visually the code simplest and more modifiable the parameters as well. Besides, the functions are at the top of the script in order to use them later in the loops, without functions the code would be illegible and fuzzy in a long term even to the code developer.

2.5.3 Main Loop

Once the code reaches the main loop, **R** has the **raw data** already saved into a **data frame** variable called "***raw.data****"*.

2.5.3.1 Data preparation

The target of the main loop is going through the "***raw.data****"* and by splitting the information by time series do the data preparation and data partition's part.

Data preparation's object is to assemble the information in a correct way to make a correct analysis.

```
################################### LOOP CODE #############

i<-0 # starting iteration (0)
for(series_name in unique(raw.data$ts_id)){

  # Format Data
  df<-subset(raw.data, raw.data$ts_id==series_name)
  df_plan<-(df[df$kpi %in% c("plan"), ])
  df_mfc<-(df[df$kpi %in% c("mfc"), ])
  df_Act<-(df[df$kpi %in% c("act"), ])

  df_plan$time<- (as.Date(df_plan$time, format = "%d.%m.%Y"))
  df_mfc$time<- (as.Date(df_mfc$time, format = "%d.%m.%Y"))
  df_Act$time <- (as.Date(df_Act$time, format = "%d.%m.%Y"))

  mfc_ts<-preprocess(df_mfc)[c("time","value")]
  plan_ts<-preprocess(df_plan)[c("time","value")]
  Actu_ts<-preprocess(df_Act)[c("time","value")]
```

Picture 9: Beginning of the main loop, data preparation.

- **Functions:**
  - ***unique*** function gets the different values of the data frame's column and
  - ***subset*** function splits the data frame by some criterion, for example the information with the same **ts_id**.
  - ***preprocess*** function is responsible to the pre-processing part.

The loop helps to go through all the different values per ts_id, then is saved in a data frame per ts_id in a dynamic way. Once the information is split by ts_id in a data frame, the pre-processing part is launched.

```
preprocess<-function(dtf){
  Actu_ts<-merge(data.frame(list(time=seq(dtf[order(dtf$time),]$time[1],
            dtf[order(dtf$time),]$time[length(dtf[order(dtf$time),]$time)], by=sequencystep))),
            dtf[order(dtf$time),], all=T)
  Actu_ts$value[is.na(Actu_ts$value)]=0
  return(Actu_ts)
}
```

Picture 10: R-script pre-processing function.

Due to the database settings does not record zeros, the exported data are incomplete time series, then the "***preprocess***" function is used to complete the time series by merging a sequence of a full time series and by replacing the gaps with 0. The function input is the "***raw.data***" (data frame object) and the output is a complete time series (data frame object).

2.5.3.2 IF structure

The **if structure** is used to maintain a good forecasting quality control, because not all the time series are long enough to ensemble a model, in the if sentence is imposed that at least the time series has to be 1 year length. With this premise the data partition is executed.

2.5.3.3 Data partition

Following the good practices to have the code cleaner, the data partition is defined in a function as well.

```
datapartition<-function(Actu_ts,trainpercentage){
  h<-round(length(Actu_ts)*trainpercentage)
  train.end <- time(Actu_ts)[h]
  test.start <- time(Actu_ts)[h+1]
  return(list(test.start,train.end))
}
```

Picture 11: R-script data partition function.

The input of the "***datapartition***" function is the output of the "***preprocess***" function and a variable "***trainpercentage"*** which is a numeric value between 0-100.

Besides, the output of the function is a list of two "***date***" objects to set the beginning of the testing part and the end of the training part, used later in the modelling part.

2.5.3.4 Modelling

The modelling part builds different model combinations by looping through different levels, the models are built with the training part. Furthermore, within the same loop the forecasts, errors and strings are stored in the lists defined at the top of the code.

```
# Arima loop
for (p in seq(arima_p_from,arima_p_to , by=arima_p_by)){
  for(d in seq(arima_d_from,arima_d_to , by=arima_d_by)){
    for(q in seq(arima_q_from,arima_q_to , by=arima_q_by)){
      if (class(try(ARIMA<-arima(train,order=c(p,d,q)), silent=T))=="try-error"){
        Models_arima[[length(Models_arima)+1]]<- NaN
        pred_arima[[length(pred_arima)+1]]<-NaN
        Error_arima[[length(Error_arima)+1]]<-NaN
        label_arima[[length(label_arima)+1]]<-paste("Arima(",p,",",d,",",q,")",sep="")
        series_arima[[length(series_arima)+1]]<-series_name
      }
      else{
        Models_arima[[length(Models_arima)+1]]<-arima(train,order=c(p,d,q))
        pred_arima[[length(pred_arima)+1]]<-forecast(arima(train,order=c(p,d,q)),length(test))
        Error_arima[[length(Error_arima)+1]]<-ErrorFunction(forecast(arima(train,order=c(p,d,q)),length(test))$mean,test,errormeasure)
        label_arima[[length(label_arima)+1]]<-paste("Arima(",p,",",d,",",q,")",sep="")
        series_arima[[length(series_arima)+1]]<-series_name
      }}}}
```

Picture 12: R-script's loop combination model.

2.5.4 Challenge/Champion

The stored lists are tabulated into a data frame to show the results and define the winning model by using a loop, subset the data frame by a unique id and finding the minimum error.

```
for(ts_id in unique(df_arima$series_arima)){
  df1<-subset(df_arima, df_arima$series_arima==ts_id)
  df_arima_win<-rbind(df_arima_win,df1[which.min(df1$Error_arima),])}
```

***Picture 13**:* Winning Model loop.

CHAPTER 3 – RESULTS

"Hiding within those mounds of data is knowledge that could change the life of a patient, or change the world."

Atul Butte, Stanford

The script was developed to works always in a default format (see ***table 2 "****Sample data format of the csv file*"). Furthermore, the study launched the aforementioned combination models through 33 time series, but not all them had a good quality and were either unpredictable or with immeasurable error.

However, the error was immeasurable because of the unpredictability, in other words, the number of zeros, the high volatility of the data and not long enough time series are some of the criteria to skip the analysis of a time series.

As a result, a high % of the time series were appropriate to be analysed and measured with the MAPE explained in the ***point 2.4.4.2***.

3.1 TABLE RESULTS

In this section the winning models are shown by forecasting model, containing **15** analyzable time series in the dataset. The table shown below contains some statistics from each time series analyzed, it helps to understand time series behavior and why each model could perform better from others.

ts	Obs	Zeros	CV	SD
ts1	84	0	49.4	75039.51
ts2	80	35	143.44	4737.03
ts3	65	29	302.13	5033.74
ts4	84	8	110.42	22306.75
ts5	84	0	98.66	150257.57
ts6	72	53	267.04	100.59
ts7	84	0	25.89	15017.78
ts8	33	30	361.29	15.98
ts9	83	34	138.18	303.72
ts10	84	0	33.31	1764781.13
ts11	84	0	26.28	13948.96
ts12	84	6	65.01	135590.96
ts13	84	0	10.97	277325.73
ts14	80	4	107.07	11889.72
ts15	84	0	28.56	63652.48
ts16	84	0	32.94	75068.61
ts17	84	0	22.75	761886.45
ts18	84	1	125.45	2660.09
ts19	82	49	163.16	1873.84
ts20	84	0	68.02	3726.39
ts21	84	0	40.07	24667.06

ts22	84	0	87.75	24497.41
ts23	84	0	22.86	9960.92
ts24	84	0	33.56	35131.28
ts25	84	0	63.73	53566.59
ts26	84	0	65.84	1759.74
ts27	84	0	73.83	1446.4
ts28	84	1	112.05	680.66
ts29	84	0	59.52	418851.97
ts30	84	0	15.99	209360.79
ts31	84	14	147.87	913.2
ts32	83	0	46.96	352230.76

Table 3: *Descriptive data table*

There are some time series with a high coefficient of variation (Standard deviation / mean), big percentage of zeros and short number of observations. It means that not all time series are going to show coherent results in the error measurement because of the aforementioned criteria.

Consequently, some time series are deleted again because they are not well measured, either because they show errors as Infinite value or value higher than 100%.

Time series	ARIMA	SARIMA	ES	Average	Naïve
Ts5	12,86	12,62	11,85	63,00	11,00
Ts7	14,83	7,50	17,80	54,00	8,00
Ts11	8,89	5,58	8,13	37,00	12,00
Ts12	13,49	12,91	14,24	93,00	18,00
Ts14	3,84	3,86	3,25	9,00	3,00
Ts16	6,06	5,84	5,48	37,00	7,00
Ts17	14,58	13,96	15,52	53,00	22,00
Ts18	7,84	4,75	4,30	22,00	24,00
Ts21	10,04	10,65	17,39	6,00	14,00
Ts22	25,23	22,39	21,25	79,00	28,00
Ts24	8,00	8,06	7,82	42,00	10,00
Ts25	9,55	9,45	8,90	22,00	14,00
Ts26	38,80	35,05	34,57	38,00	51,00
Ts28	38,70	65,59	43,00	79,00	29,00
Ts30	6,11	4,33	3,71	14,00	7,00
Mean	14,59	14,84	14,48	43,20	17,20
Std	11,06	16,25	11,44	26,80	12,25

Table 4: Winning models.

Besides, the results show an acceptable performance of the selected models, several time series reach a low percentage of error measurement.

Because the non-scientific approach, it is not tried to draw conclusions about which model works better in this business model. As a result, according to the statistics in ***table 4***, a general model couldn't be established robustly as a general model to work with.

As many scientific researches show, exponential smoothing has better average and standard deviation than the other models, confirming previous studies that show SARIMA and exponential smoothing works better in a short forecasting term compared to other models (P. J. Harrison, Exponential Smoothing and Short-Term Sales Forecasting, 1967).

	ARIMA	SARIMA	ES	Average	Naïve
fi	1	5	8	2	4
Fi	7,14	35,71	57,14	14,28	28,57

Table 5. Error statistics.

Calculations fi and Fi are based on the frequency of model that won in a time series:

- **Cumulative frequency** (***fi***) is the total of the absolute frequencies of all events at or below a certain point in an ordered list of events.
- **Relative frequency** (**Fi**) of an event is the absolute frequency normalized by the total number of events.

At the end of the document it is possible to find the extended tables, where is possible to check the orders of the models and their performance.

3.2 FORECAST PLOTS

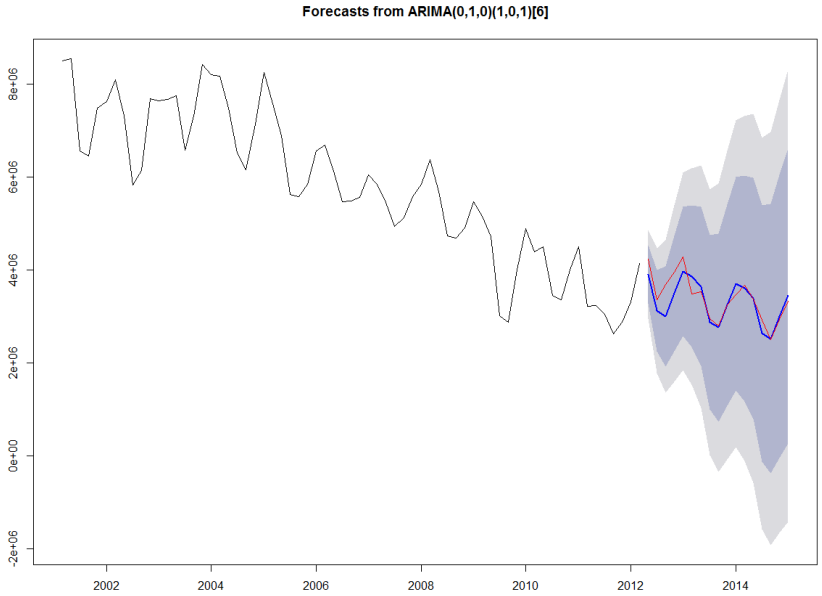
In this section, it is shown a comparison between the models in one time series analyzed, understanding how performed, the rest of plots can be found in the annex section at the end of the document.

The time series selected to compare the performance between the model is Ts11 with the f descriptive table:

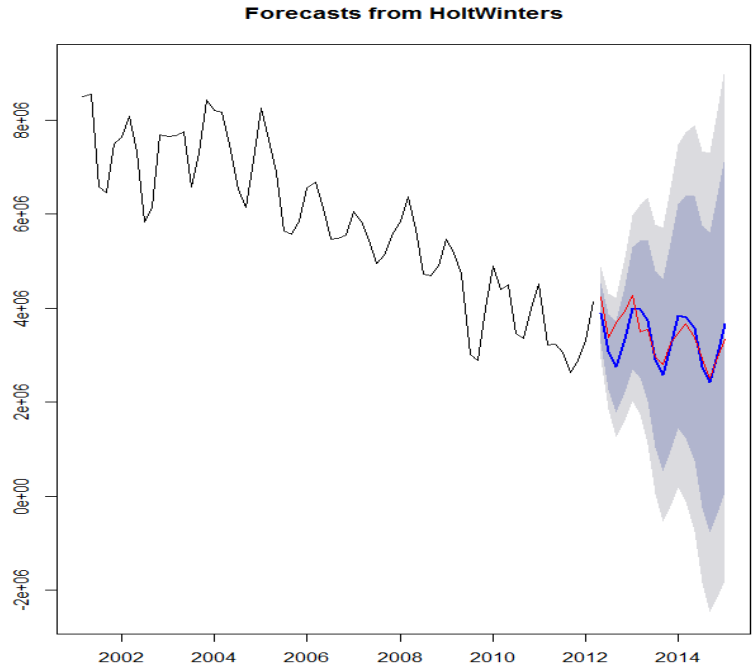
ARIMA	SARIMA	ES	NAÏVE	AVERAGE	Obs	Zeros	CV	SD
8,89	5,58	8,13	37,00	12,00	84	0	26.28	13948.96

Table 6: *Descriptive table, Ts10*

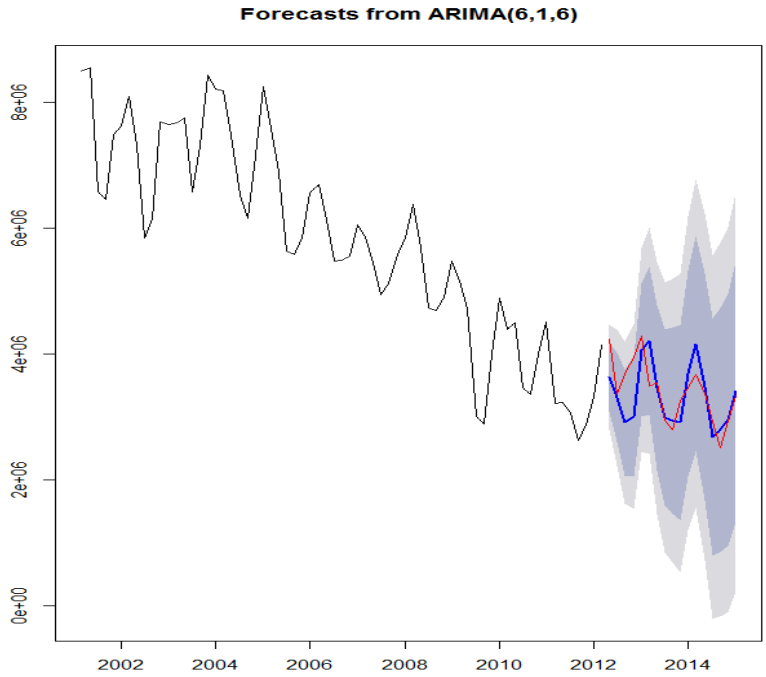
In a dynamic way, the program developed applies several combination models to a time series, after getting all combinations, the program find the minimum MAPE error in each model obtaining the results of the table 8 and plotted in the following pictures, where the red line is the test part, the blue line the forecast and the dark zone is the interval of confidence between 95-85%.



Picture 14: Winner SARIMA Forecast, Ts11, MAPE:5,58.



Picture 15: No-Winner ES, Ts11, ES(0.7,0.1,1), MAPE:8.13.



Picture 16: No-Winner, ARIMA Forecast, MAPE:8.89.

Once again, the developed program chooses among the models the one with the smallest error of prediction, in such a way that a new winning model is chosen and this will be the one used by the user or company as the best model to apply to the time series.

These results may vary according to the step size used for the combination of predictive models, database used and error measurement used. In fact, the user can decide which parameters to use according to the precision and criteria that he wants to use,

In the study carried out, standard parameters were chosen, with a high computational cost, but better results could be obtained by expanding the parameter range with a computational cost higher than the one used to carry out the study.

CHAPTER 4 –CONCLUSIONS

"The temptation to form premature theories upon insufficient data is the bane of our profession."

Sherlock Holmes, fictional detective

4.1 PROJECT CONCLUSIONS

- Programming Languages suppose a high opportunity to process big data volume, implementing this big data approach saves time and Works better than previous methods with lower forecasting errors.

- It is important to emphasize that little improvements can be a big savings of money, in fact the results are forecast in number of tonnes, it means that is highly important to resize the train length in order to save logistics costs, in other words, such a small improvement could suppose millions of Euros.

- In a long-term, the three models are performing better than simplest models, however it is not possible to establish a general winning model between ES, ARIMA and SARIMA. It means that in the script with 80% of a time series length is still small to get short and mid-term forecasts.

- ES shows good predictions by studying the time series in isolation with a high frequency of winning times.

- Naïve model is still close from univariate models, the results could be improved either by applying exogenous variables or narrowing the term of the forecast.

- There are high unpredictable time series due to the change of business contracts, the high volatility, load transport cancellations and some market trends that changes spontaneously.

4.2 AVENUES FOR FURTHER RESEARCH

- Code optimization and automatic plotting in a smart solution (visualization).

- Remove the csv file and Reading method by creating a smart connection between the database and R.

- Develop an automatic process to discern the data patterns to select the correct time series length, sometimes a time series shows a change of contract or market change.

- Either applying exogenous variables or applying more univariates models prepared to the high volatility behavior such as **Arch** and **Garch** models.

CHAPTER 5 – LITERATURE

"You can have data without information, but not information without data."

Daniel Keys Moran, Computer programmer

5.1 SCIENTIFIC REVIEWS

1. P.Someshwar Rao (1978) Forecasting the demand for railway freight services, Journal of transport economics and policy. 19-21.
2. Fite, Jonathon T; Taylor, G Don; Usher, John S; English, John R; Roberts, John N., 2002. International Journal of Physical Distribution & Logistics Management, Volume 32, Number 4, 2002, pp. 299-308(10).
3. Ülo Hunt (2003) Forecasting of railway freight volume: Approach of Estonian railway to arise efficiency, Transport, 18:6, 255-258.
4. Jong-Kil Kim. Ji-Yeong Pak. Ying-Wang. Sung-Il Park.Gi-Tae Yeo. A Study on Forecasting Container Volume of Port Using SD and ARIMA" Journal of Navigation and Port Research International Edition 35. Vol.35, No.4 pp. 343~349, 2011 (ISSN-1598-5725).
5. Weiss AM, Indurkhya N: *Predictive data mining – a practical guide*. San Francisco, California: Morgan Kaufmann Publishers; 1998.
6. Arima model and exponential smoothing method: a comparison. Wan ahmad, wan kamarul; ahamad, sabri. April 2013, AIP conference proceedings Vol 1552 Issue 1 p1312.
7. Lynwood A. Johnson Douglas C. Montgomery and John S. Gardiner. Forecasting and Time Series Analysis. McGraw-Hill,Inc, 2nd edition edition, 1990.
8. Fildes, R., & Makridakis, S., 1995. The impact of empirical accuracy studies on time series analysis and forecasting, International Statistical Review 63, 289-30.
9. Makridakis, S., Hibon, M., 2000. The M3-Competition: results, conclusions and implications International Journal of Forecasting, 16, pp. 451–476.
10. Chien-Chang Chou, Ching-Wu Chu, Gin-Shuh Liang, 2008. A modified regression model for forecasting the volumes of Taiwan’s import containers. College of Maritime Science and Management, National Taiwan Ocean University 2, Pei Ning Road, Keelung, Taiwan, ROC
11. Ling Zhou, Bernhard Heimann, Uwe Clausen. Short term demand forecasting for a typical logistics service provider.
12. P. J. Harrison, 1967. Exponential Smoothing and Short-Term Sales Forecasting. Management Science,13:11 , 821-842

5.2 WEBSITES

13. R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria – URL: http://www.R-project.org/.

5.3 DOCUMENTS

14. Esperanza Catalán, C. (2016). Series temporales. 1st ed. Laboratorio de estadística, pp.17-52.

15. Blank, S. (2013). Why the Lean Start-Up Changes Everything. 1st ed. Harvard Business Review, pp. 4-9.

16. PWC & University of Rome. (2014). Study on Single Wagonload Traffic in Europe. 1st ed. Brussels: EUROPEAN RAIL FREIGHT DAYS, pp.11-17.

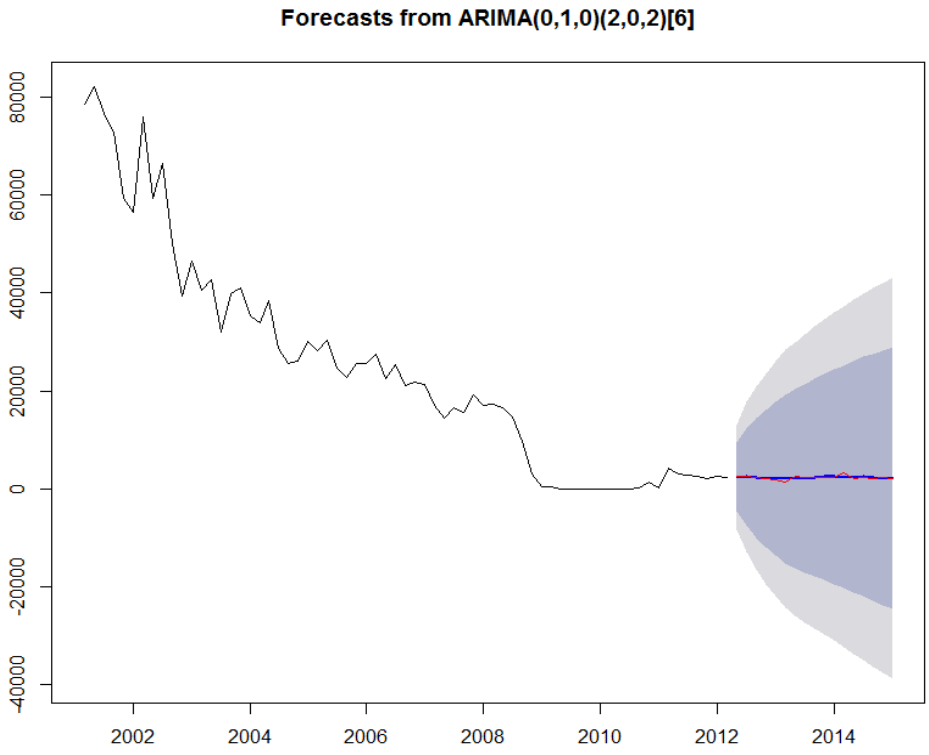
17. IBM. (2011). SPSS Modeler CRISP-DM Guide. 1st ed. IBM, pp. 2-37. URL: http://ftp.software.ibm.com.

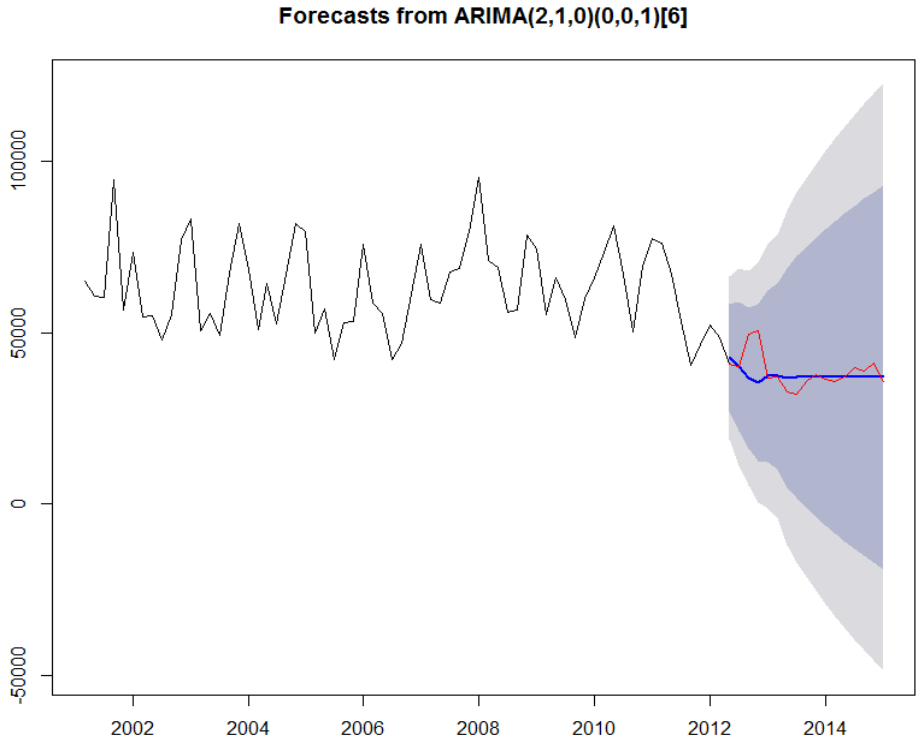
CHAPTER 6 – ANNEX

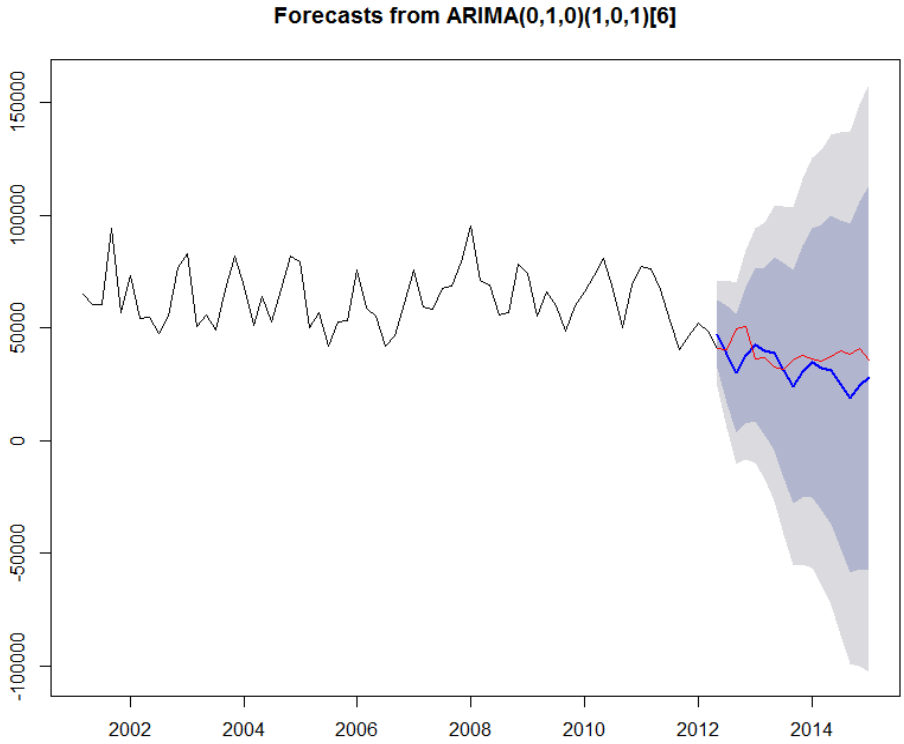
6.1 TABLES

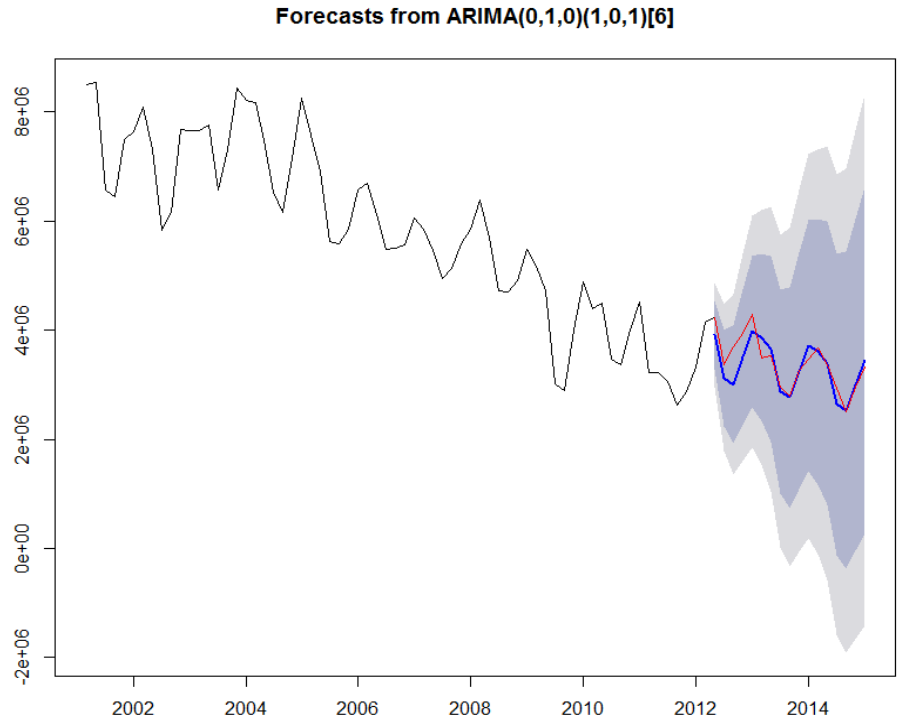
Ts	ARIMA	MAPE	SARIMA	MAPE	ES	MAPE
Ts1	Arima(1,2,4)	172,84	sarima(0,1,0,2,2,0)	102,57	ES(0.4,0.9,0.3)	37,96
Ts2	Arima(0,1,0)	100,00	sarima(0,1,0,0,0,0)	100,00	ES(0.1,0,0)	Inf
Ts3	Arima(0,1,0)	100,00	sarima(0,0,0,0,1,0)	100,00	ES(0.1,0,0)	Inf
Ts4	Arima(0,1,4)	10,57	sarima(0,1,0,2,0,2)	9,82	ES(0.4,0.2,0.7)	12,34
Ts5	Arima(0,1,0)	12,86	sarima(2,1,1,1,0,0)	12,62	ES(0.9,0,1)	11,85
Ts6	Arima(0,1,0)	100,00	sarima(0,1,0,0,0,0)	100,00	ES(0.1,0,0)	Inf
Ts7	Arima(0,2,6)	14,83	sarima(2,1,0,0,0,1)	7,50	ES(0.7,0.2,0.2)	17,80
Ts8	Arima(0,1,0)	100,00	sarima(0,1,0,0,0,0)	100,00	ES(0.1,0,0)	Inf
Ts9	Arima(0,1,0)	100,00	sarima(0,1,0,0,0,0)	100,00	ES(0.1,0,0)	Inf
Ts10	Arima(6,1,6)	8,89	sarima(0,1,0,1,0,1)	5,58	ES(0.7,0.1,1)	8,13
Ts11	Arima(3,2,0)	13,49	sarima(0,1,1,0,2,0)	12,91	ES(0.3,0.6,0.9)	14,24
Ts12	Arima(0,0,0)	Inf	sarima(0,0,0,0,0,0)	Inf	ES(0.1,0,0)	Inf
Ts13	Arima(5,0,3)	3,84	sarima(0,0,0,2,0,2)	3,86	ES(0.3,0.2,0.2)	3,25
Ts14	Arima(0,0,0)	Inf	sarima(0,0,0,0,0,0)	Inf	ES(0.1,0,0)	Inf
Ts15	Arima(3,0,6)	6,06	sarima(1,0,2,2,1,1)	5,84	ES(0.5,0.4,0.5)	5,48
Ts16	Arima(3,0,5)	14,58	sarima(1,0,1,2,0,0)	13,96	ES(0.3,0,0.1)	15,52
Ts17	Arima(6,2,3)	7,84	sarima(0,1,0,0,1,2)	4,75	ES(0.9,0,1)	4,30
Ts18	Arima(0,0,0)	Inf	sarima(0,0,0,0,0,0)	Inf	ES(0.1,0,0)	Inf
Ts19	Arima(0,1,0)	100,00	sarima(0,1,0,0,0,0)	100,00	ES(0.1,0,0)	Inf
Ts20	Arima(6,1,2)	10,04	sarima(0,1,1,0,0,2)	10,65	ES(0.3,0.1,0.3)	17,39
Ts21	Arima(6,2,6)	25,23	sarima(0,1,0,1,0,2)	22,39	ES(0.2,0.1,0.5)	21,25
Ts22	Arima(0,1,3)	29,40	sarima(0,1,0,2,0,2)	24,12	ES(0.6,0.1,1)	21,26
Ts23	Arima(4,2,5)	8,00	sarima(2,1,0,2,0,1)	8,06	ES(0.3,0.4,0.1)	7,82
Ts24	Arima(2,1,2)	9,55	sarima(0,1,0,1,0,0)	9,45	ES(0.6,0.1,0.1)	8,90
Ts25	Arima(1,2,6)	38,80	sarima(1,0,1,2,2,1)	35,05	ES(0.1,0.1,0.7)	34,57
Ts26	Arima(0,1,5)	13,63	sarima(0,1,0,1,0,0)	14,26	ES(0.3,0.9,0.4)	12,73
Ts27	Arima(0,2,1)	38,70	sarima(1,0,0,0,2,1)	65,59	ES(0.8,0.4,0.9)	43,00
Ts28	Arima(3,2,4)	31,17	sarima(1,1,0,2,0,1)	40,46	ES(0.2,0.6,0.4)	42,49
Ts29	Arima(3,1,3)	11,71	sarima(0,1,2,1,0,0)	12,76	ES(0.3,0.4,1)	9,19
Ts30	Arima(1,2,4)	6,11	sarima(0,1,1,1,1,2)	4,33	ES(0.9,0.1,0.6)	3,71
Ts31	Arima(0,0,0)	Inf	sarima(0,0,0,0,0,0)	Inf	ES(0.1,0,0)	Inf
Ts32	Arima(0,0,0)	100,99	sarima(1,0,1,0,0,2)	100,99	ES(0.5,1,0.3)	87,91

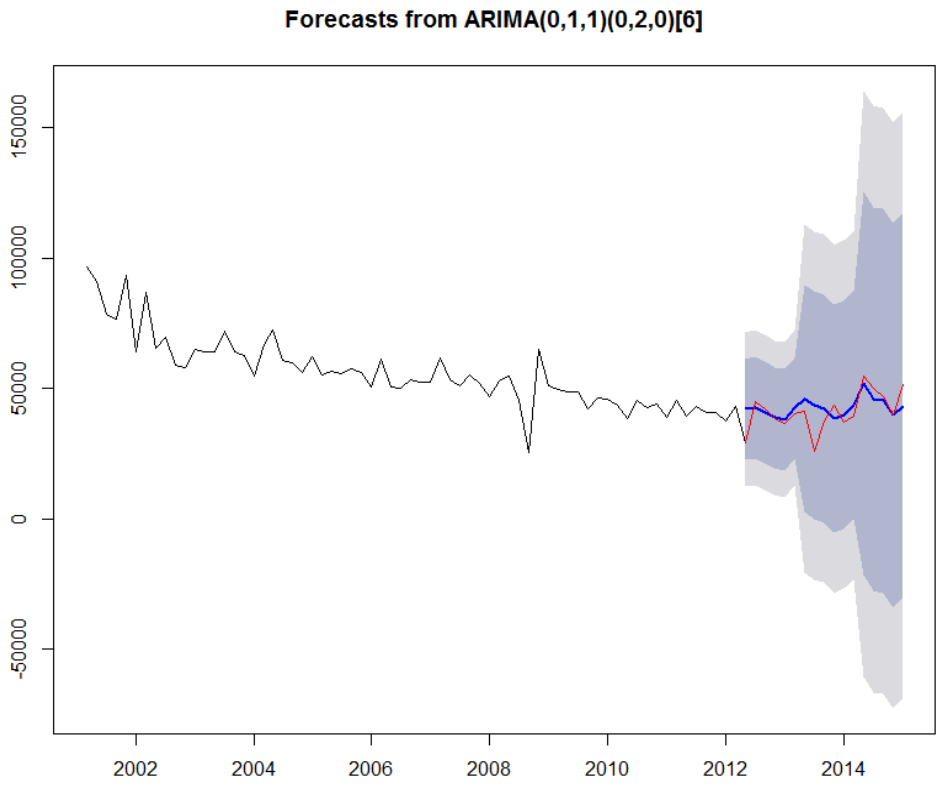
6.2 ANNEX PLOTS

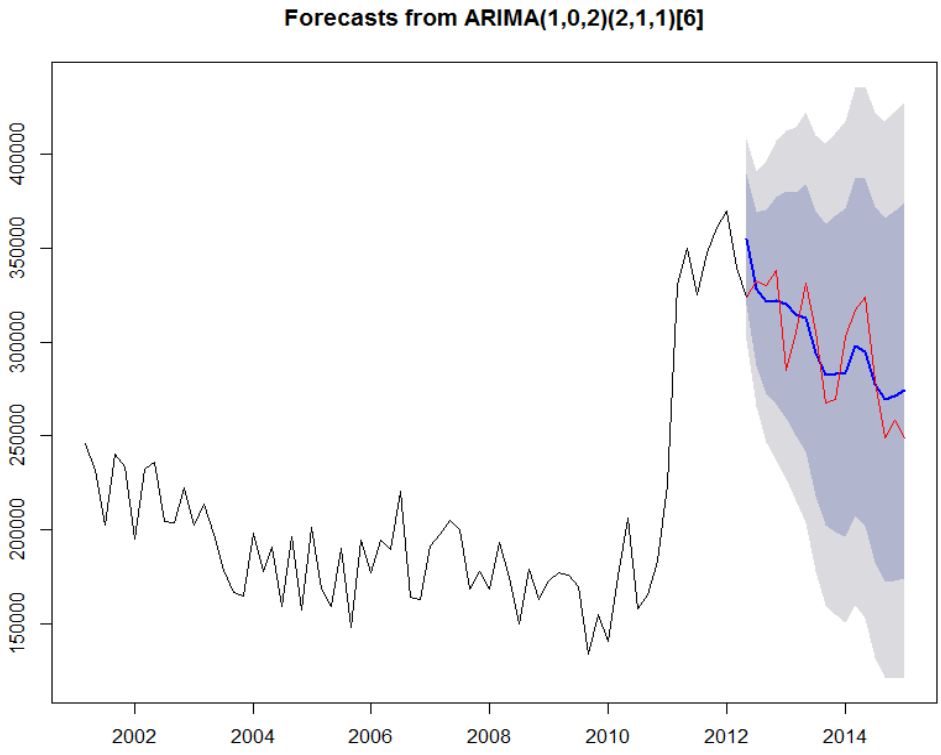


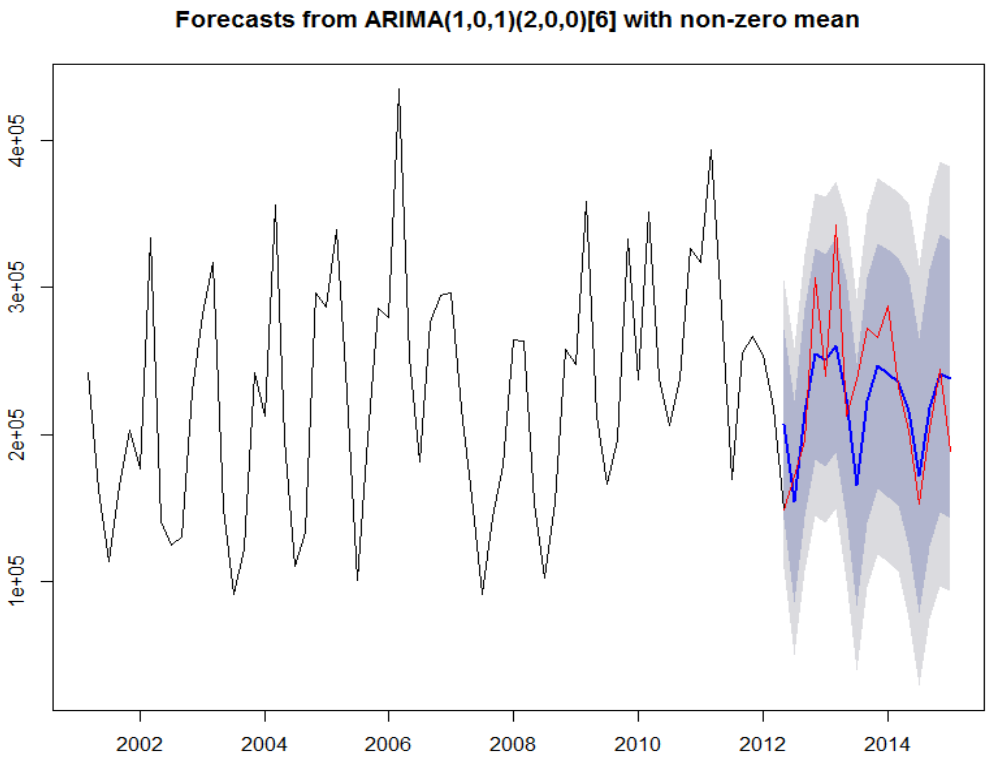


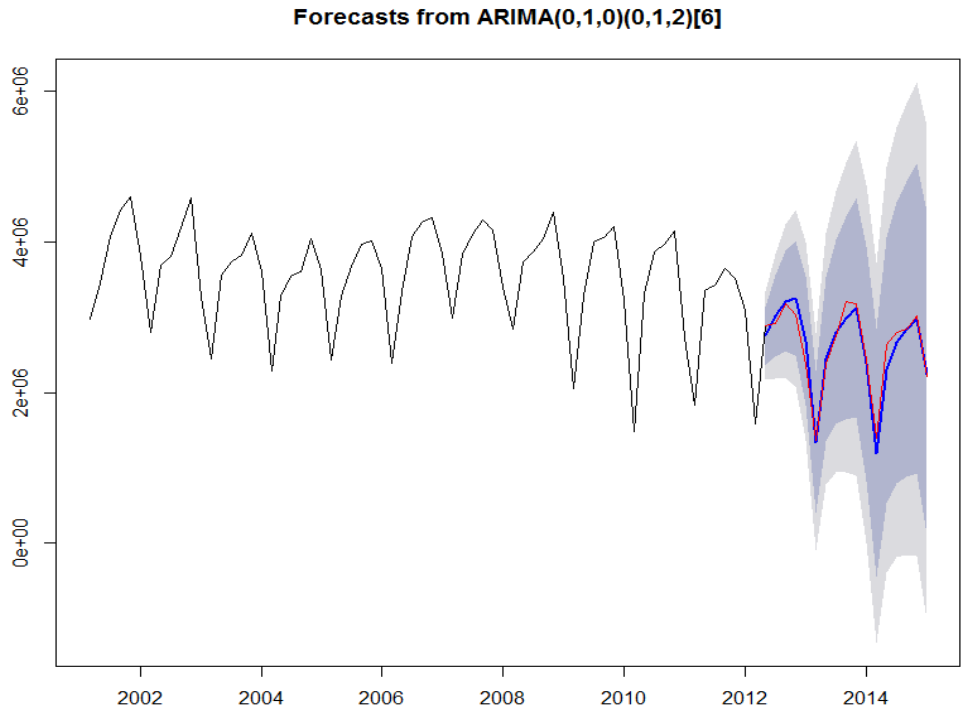

Forecasts from ARIMA(0,1,0)(1,0,1)[6]
150000
100000
50000
0
-50000
-100000
2002
2004
2006
2008
2010
2012
2014

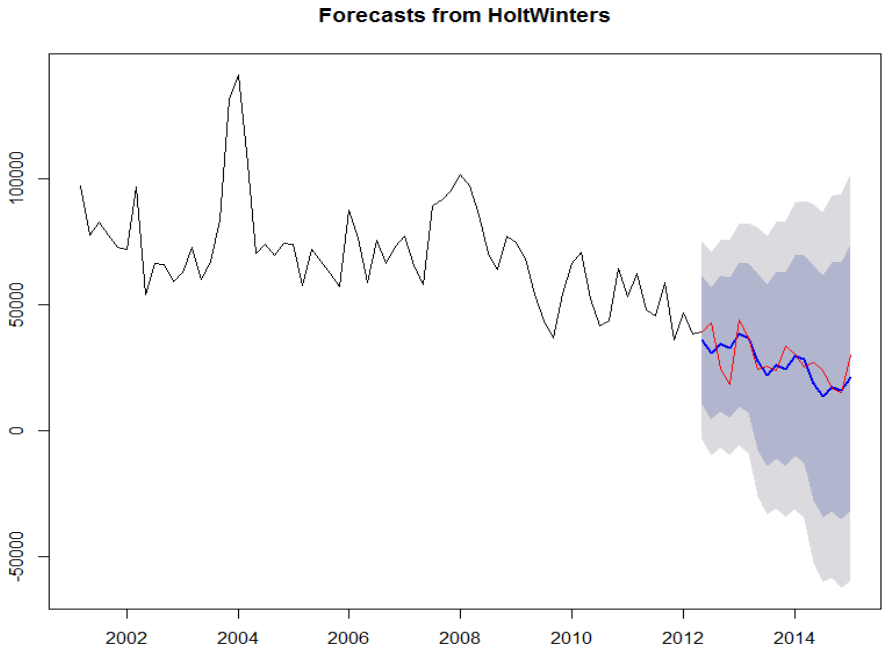

Forecasts from ARIMA(0,1,0)(1,0,1)[6]
8e+06
6e+06
4e+06
2e+06
0e+00
-2e+06
2002
2004
2006
2008
2010
2012
2014


Forecasts from ARIMA(0,1,1)(0,2,0)[6]
150000
100000
50000
0
-50000
2002
2004
2006
2008
2010
2012
2014

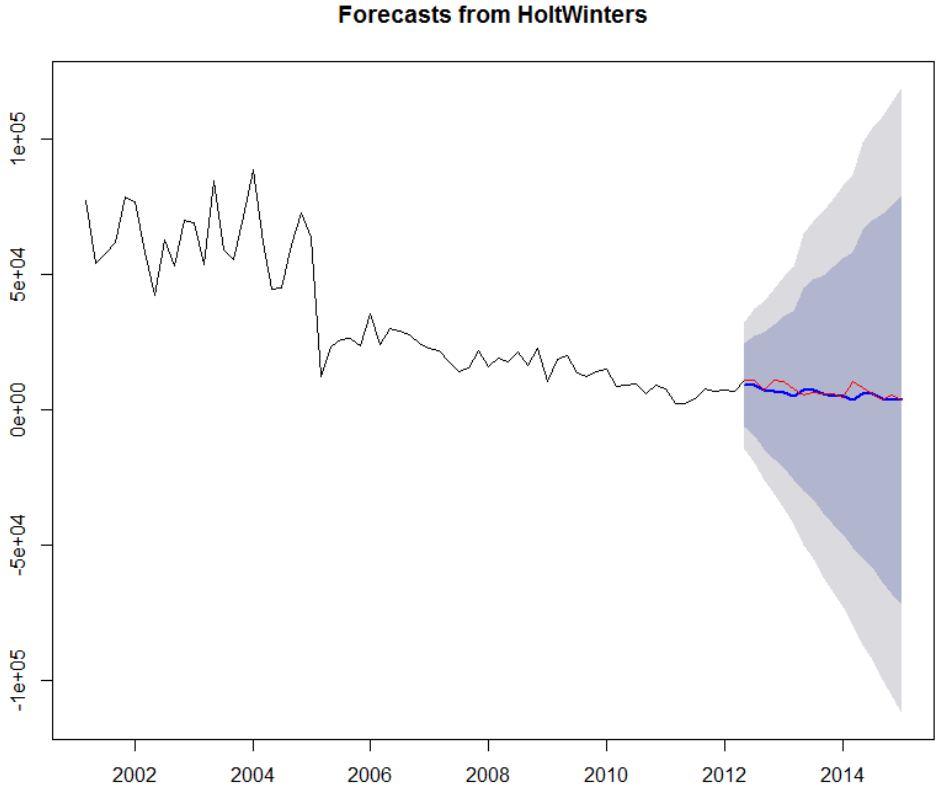

Forecasts from ARIMA(1,0,2)(2,1,1)[6]
400000
350000
300000
250000
200000
150000
2002
2004
2006
2008
2010
2012
2014


Forecasts from ARIMA(1,0,1)(2,0,0)[6] with non-zero mean
4e+05
3e+05
2e+05
1e+05
2002
2004
2006
2008
2010
2012
2014

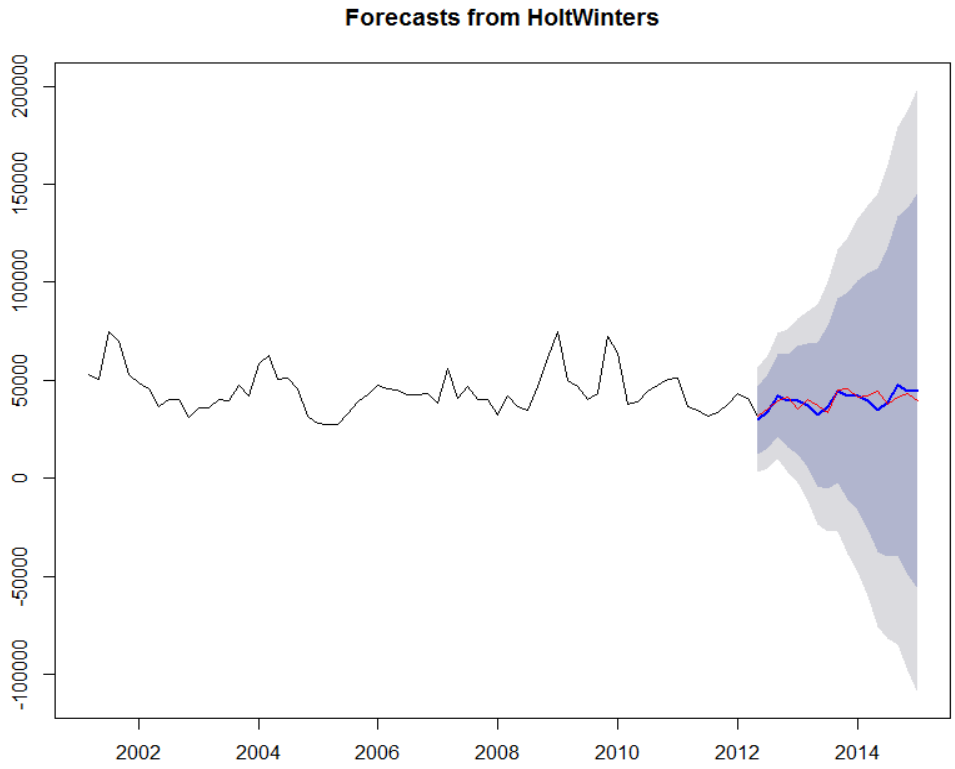

Forecasts from ARIMA(0,1,0)(0,1,2)[6]
6e+06
4e+06
2e+06
0e+00
2002
2004
2006
2008
2010
2012
2014



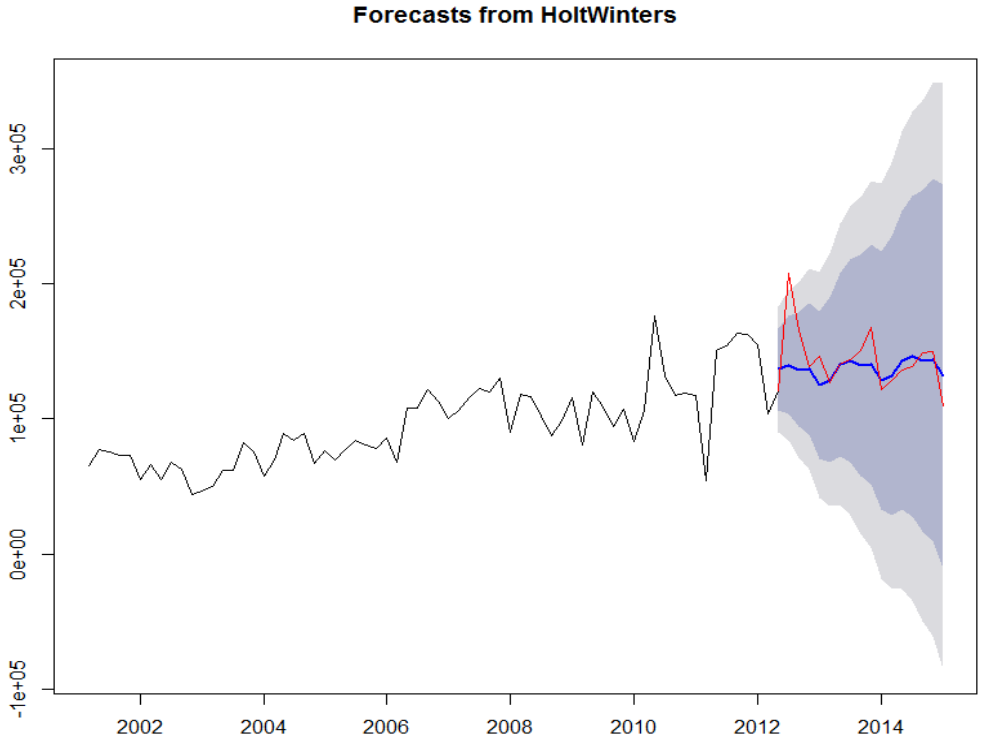
ES (0.2, 0.1, 0.5).



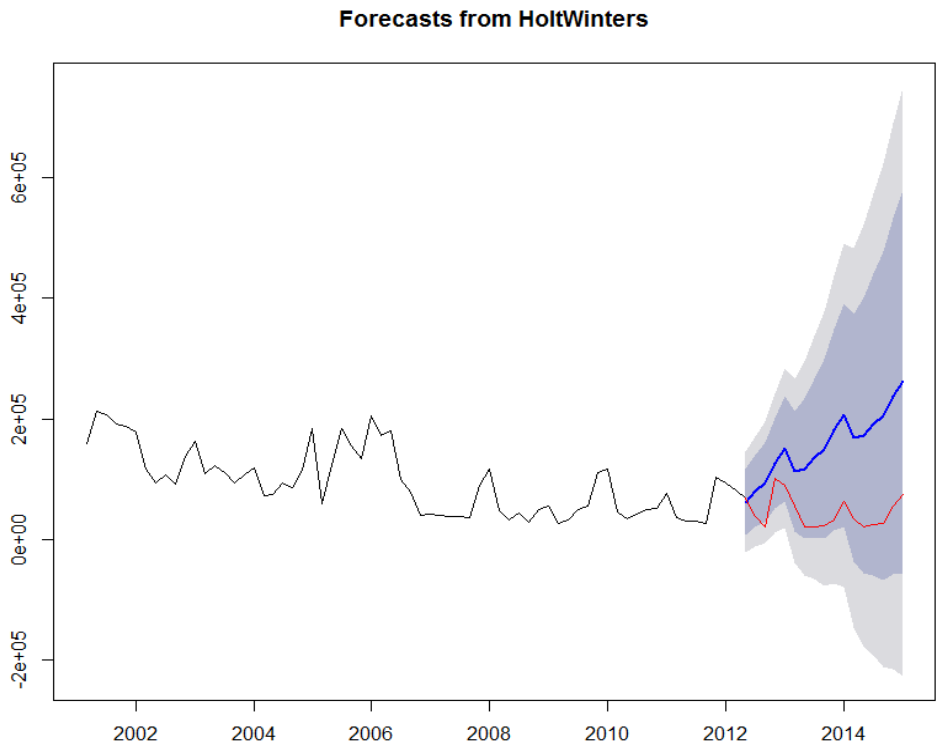
ES (0.3, 0.4, 0.1).

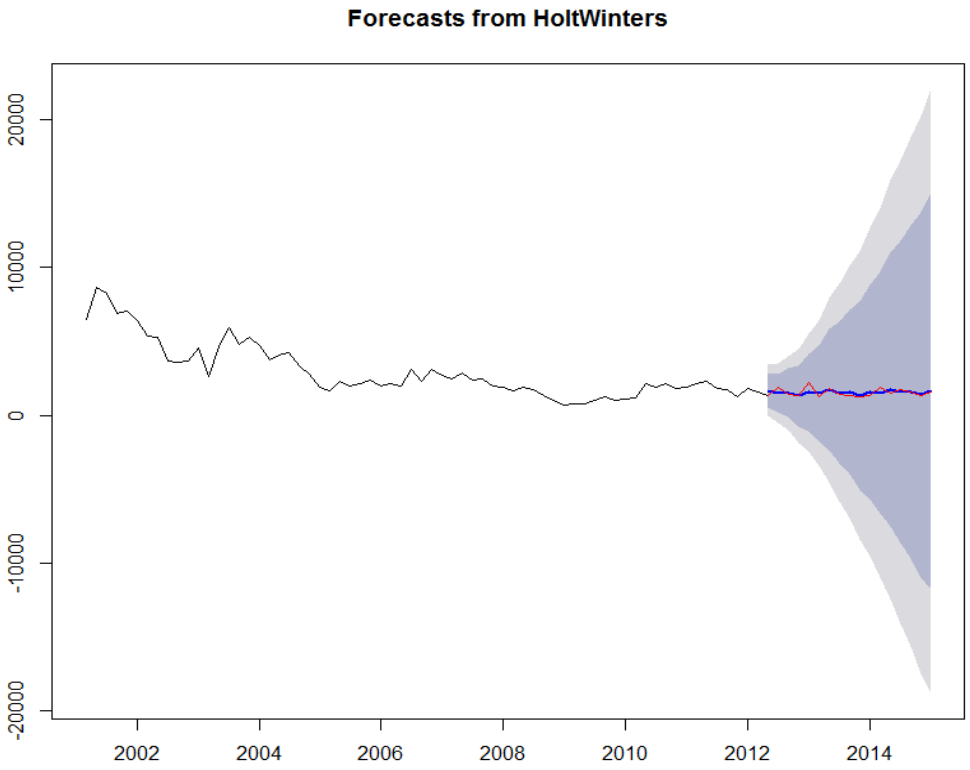


ES (0.6, 0.1, 1).

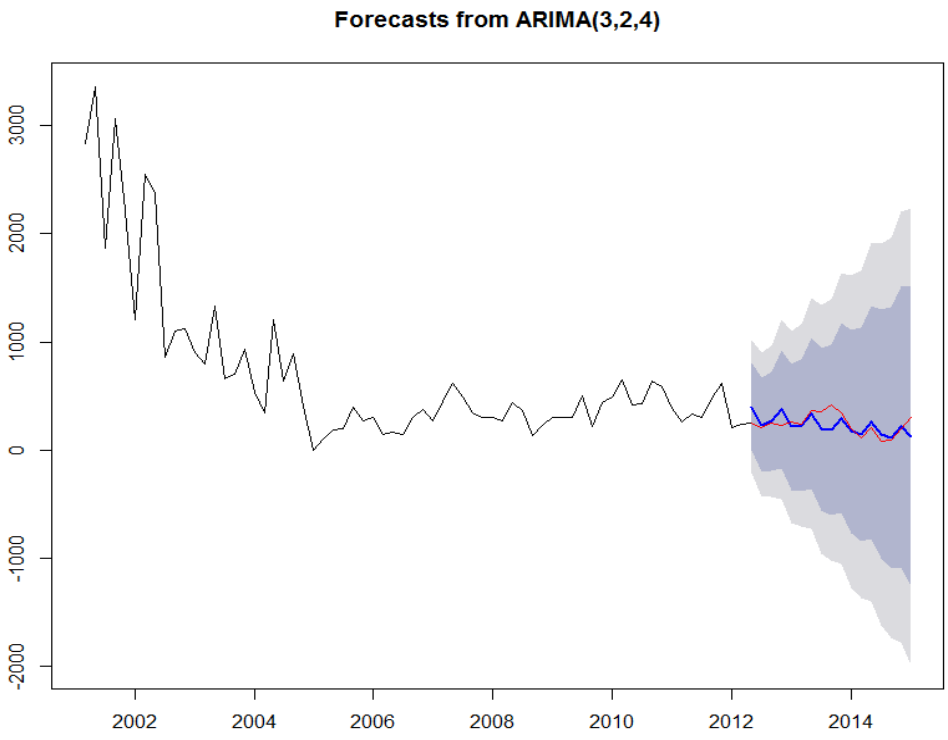


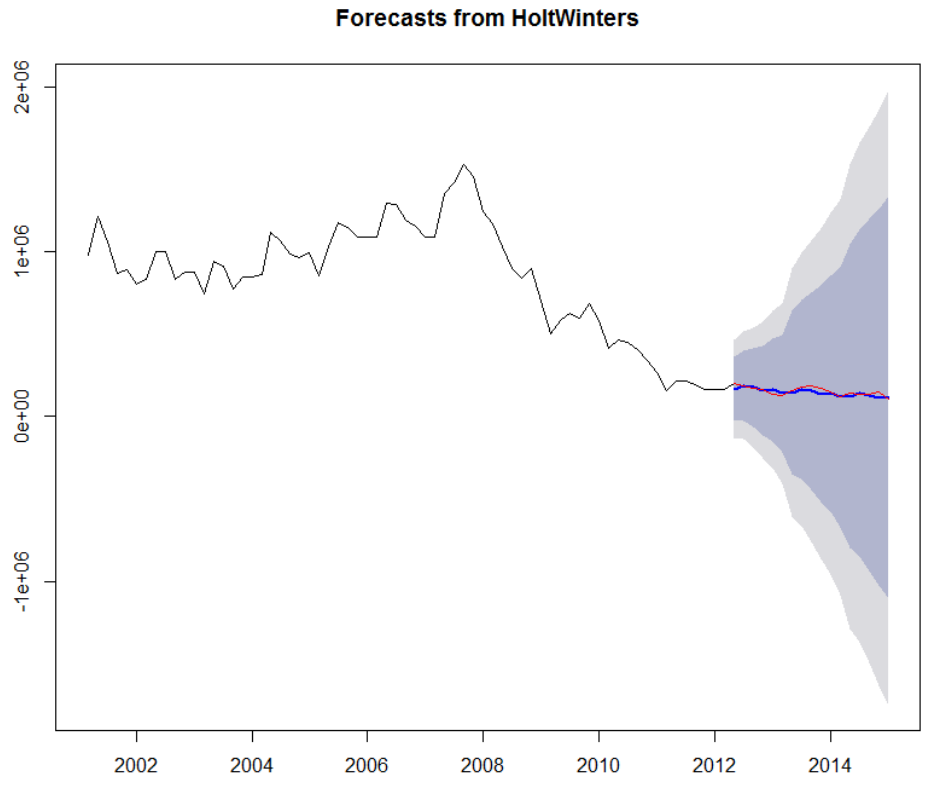
ES (0.6, 0.1, 0.1)


Forecasts from HoltWinters
6e+05
4e+05
2e+05
0e+00
-2e+05
2002
2004
2006
2008
2010
2012
2014

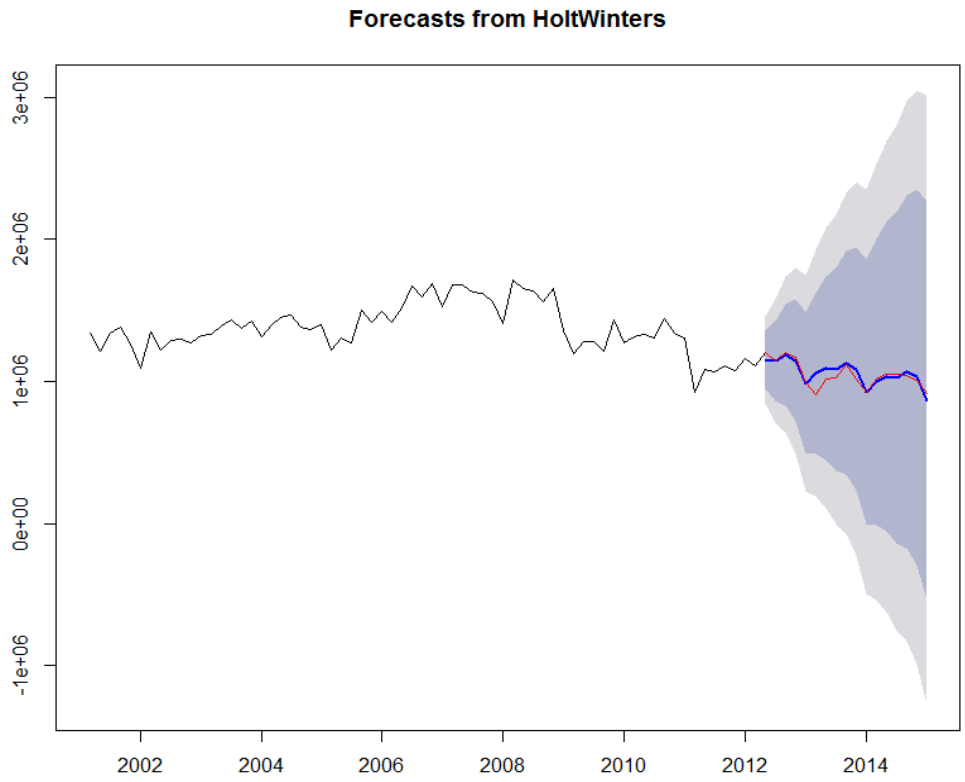

Forecasts from HoltWinters
20000
10000
0
-10000
-20000
2002
2004
2006
2008
2010
2012
2014

ES (0.3, 0.9, 0.4).

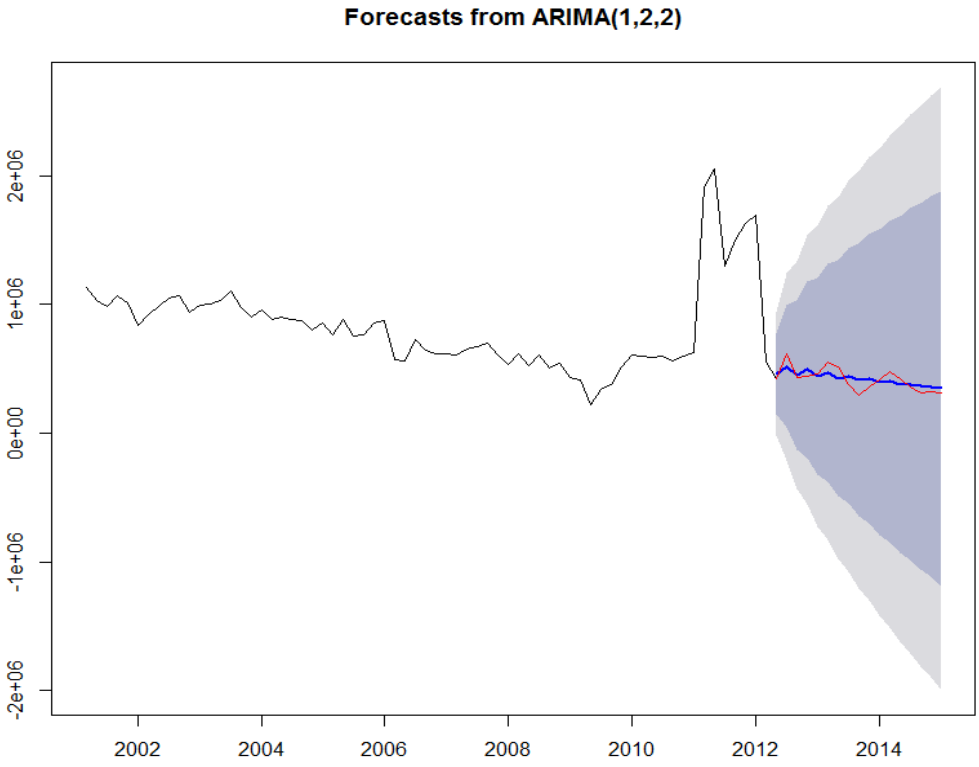

Forecasts from ARIMA(3,2,4)
3000
2000
1000
0
-1000
-2000
2002
2004
2006
2008
2010
2012
2014


Forecasts from HoltWinters
2e+06
1e+06
0e+00
-1e+06
2002
2004
2006
2008
2010
2012
2014

ES (0.3, 0.4, 1).


Forecasts from HoltWinters
3e+06
2e+06
1e+06
0e+00
-1e+06
2002
2004
2006
2008
2010
2012
2014

ES (0.9, 0.1, 0.6).


Forecasts from ARIMA(1,2,2)
2e+06
1e+06
0e+00
-1e+06
-2e+06
2002
2004
2006
2008
2010
2012
2014