Escola de Camins
Escola Tècnica Superior d'Enginyeria de Camins, Canals i Ports
UPC BARCELONATECH

TRAFFIC PORT FORECASTING SYSTEMS: TIME-SERIES METHOD APPLIED TO PIRAEUS PORT (GREECE)

TREBALL FINAL DE MÀSTER

Treball realitzat per:
Roberto Páez Álvarez

Dirigit per:
Manuel Grifoll Colls

Màster en:
Enginyeria de Camins, Canals i Ports

Barcelona, 28 de setembre de 2018

Departament d'Enginyeria Civil i Ambiental

Abstract

The objective of this study is to investigate and apply forecasting techniques for the Piraeus Port, the largest Greek seaport and one of the biggest in the Mediterranean Sea. Recently, forecasting and predictions of port freight evolution have received an increasing attention in ports management and logistics fields, due to the impact on optimization or resource assignment produced.

There are many methods to perform these prediction, each one with its own limitations and advantages. In order to achieve the objective of forecasting the port's freight future a method based on Monte Carlo experiments and Markov chains technique is used to predict the port traffic. This methodology belongs to the time-series category. To do so, it is required to pre-process the data provided by the port, investigate the different paths to simulate the evolution of port's freight, calibrate and validate the model and finally perform the prediction for the port freight evolution in time.

The results show a method performance based on the comparison between three commonly used errors in forecasting models (Root Mean Squared, Mean Absolute and Mean Absolute Percent errors).

Through this study, the prediction method is described and then applied to highlight some future sight about the freight traffic evolution at the Greek port using real data provided by itself. Different combinations of distributions and Markov chains are compared to finally end up with a normal distribution with two states chain as the best forecast model for the Piraeus case.

Acknowledgments

I would like to especially thank Manel Grifoll, my study tutor, the effort made and the time dedicated to me during the realization of it. He has been a very helpful tutor, sharing his experience and knowledge with me, helping and encouraging me at all times that I needed. Special thanks to his dedication in non-teaching periods to give me support and his prompt attention whenever required. Thanks for everything.
I would also like to thank the contribution of Thanassis Karlis, the Piraeus port engineer who gave us access to all the data we needed for the container traffic study. I hope the results are profitable for him in return to the time that he has dedicated us.
With this study, my academic period at the Universitat Politcnica de Catalunya is over -or at least for now-, and that is the reason to also mention all the staff (professors, rectors, secretariat, and everyone) of the University and to thank them all for the contribution on making me an engineer and to grow as an individual. It has been a long but a profitable time, thanks.
Finally thank my family and friends, especially my brother Enrique, who have always been supporting me and reinforcing me with positivity when I needed it, and my partner Anna, for the great support I always find, and I have found, in her person throughout my academic career. Thank you all for your support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Global container traffic

The traffic between ports and its continuous growing over the years show its importance on the industrial activity around the world, not only on the merchandise trade, but also on globalizing the production processes.

Figure 1.1 shows the growing positive trend, only affected adversely during the start of the global crisis, of the global traffic in TEUs (20 foot long, or 6.1 metres, intermodal container):

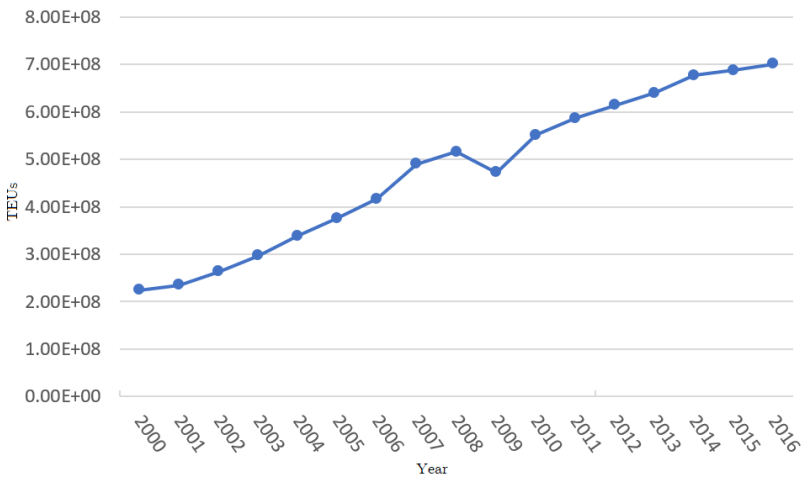


Figure 1.1: Global traffic per year in TEUs. *Source: World Bank data.*

Ports are the main infrastructure for this traffic. They play an important component by linking of global trade (every mode of transport in the supply chain) with maritime transport.

Nowadays, ports are handling around the 80 per cent of the global merchandise trade and more than two thirds of its value, which is a huge impact on the global economy. That is why ports need to be competitive and adapt to changes not only in the operational efficiency but also in the economic, regulatory and institutional landscapes. Trade evolution is driven by three basic variables: demand, supply and the policy framework of the markets. This, of course, are the most simple way of defining how the global trade is regulated. Going deeper on all of the variables, it can be stated that trade depends on different factors such as demographics (population growth, urbanization, and others), governance (regulations, transparency) and economics (capital flows, foreign investment, and many others). Port trade evolution is usually associated to the GDP of the country where the port is, and also to the neighbour countries. The global trade received a big hit in late 2008 as the world economy took a sudden and unexpectedly sharp downturn. Ports struggled to keep up with demand and to survive to those years. As a result of a backdrop of weaker global demand and the end of the global crisis, terminal operators have been reconsidering their capacity expansion plans, which is a key factor in order to keep existing in the global leading ports map. This means a great investment for ports and its investors which means that need to be carefully analyzed.

One of the trends in order to provide knowledge about the future situations that a port can live is to use forecasting techniques to determine how and when the expansion plans need to be thought of.

The intrinsic connection between maritime transportation, international trade, and globalization trends are strictly related to economy wellness as seen in Figure 1.1, where a growing in TEUs traffic when economy is also growing can be clearly seen, and a recession with the crisis. While predicting the future changes in global economics is beyond study, it can assume a continuist trend from the last episodes.

1.2 Piraeus port

The Piraeus port is the largest Greek seaport and one of the main ports in the Mediterranean sea and Europe. The port is located in the city of Athens and it is difficult to speak about when it was build, because this port has been serving the city since around 450 years BC. Despite that, start of the port as it is known today can be set in 1924 when major civil works started.

Today, the port has become a huge terminal both for transport and travel services with an area of 39 square kilometres. With three container terminals, it has a capacity for storing more than 6 million TEUs. Also, it is considered the largest passenger port in Europe with a 2.8 kilometres quay length. Finally, the port is completed with a cargo and an automobile terminals, beyond other services it offers. In Figure 1.2 port can be seen, with the passenger terminal at the front and the container one at the far end.

Figure 1.2: Piraeus port aerial view. *Source: Portopia international consortium.*

With about 19 million passengers annually, Piraeus occupies the third place worldwide. It also occupies the 47th position at international level in cargo traffic and the top position among all Eastern Mediterranean ports.

Piraeus port is a key element for the Mediterranean because of its strategic position and infrastructure. It acts as the main gate for Hellenic imports and exports and as a link for the trading between Europe, Asia and Africa, being part of the EMEA area (Europe, Middle East, Africa, see Figure 1.3.

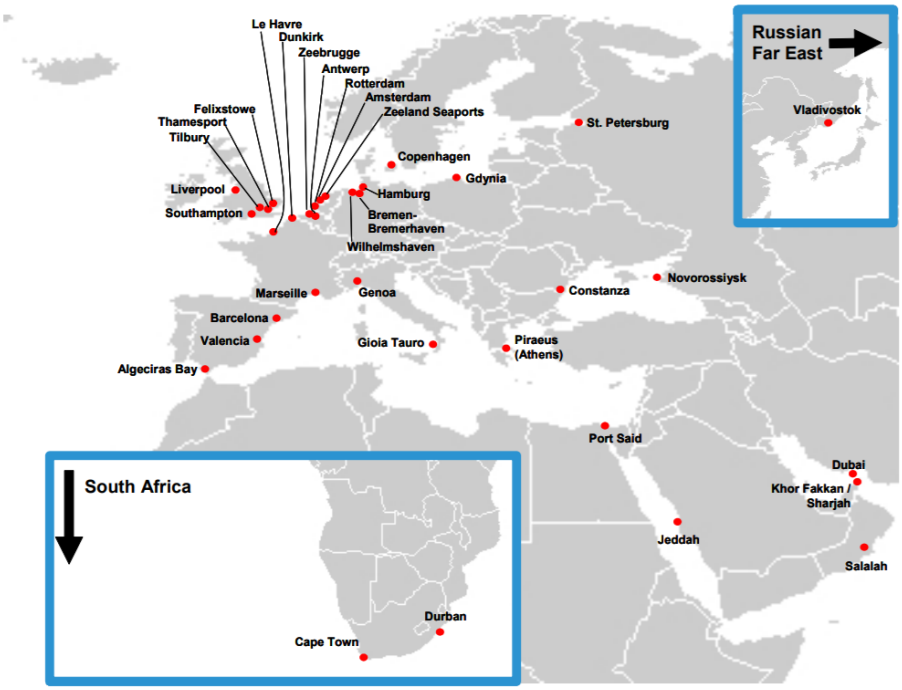


Figure 1.3: Major ports in EMEA region. *Source: RREEF Research.*

The port is managed by the Piraeus Port Authority (PPA) but in 2002 a concession contract was signed between the Greek government and OLP SA company, under which the Greek government grants for 40 years the exclusive right to use and exploitation of land, buildings and facilities of inland port area of PPA to OLP SA.

Due to the big hit the global crisis meant to Greece, the port closed its container terminals during almost all the 2008 months, until the company Cosco Pacific signed a contract for the terminals concession. This is going to be a big breach in the data gathered by the port.

1.3 Predictive modeling opportunities

One of the best ways for successful management of certain transport companies, specially when speaking about infrastructure, is traffic demand planning. The main reason for it is that if there is a higher supply than demand it leads to the failure in the utilization of port infrastructure and suprastructure, and to the lack of cost-effectiveness, but it is also needed to be careful not to over-plan in order to avoid misspending money.

The predictive models are very useful on this task. A definition of what a predictive model is can be: *"Predictive modeling is a name given to a collection of mathematical techniques having in common the goal of finding a mathematical relationship between a target, response, or dependent variable and various predictor or independent variables with the goal in mind of measuring future values of those predictors and inserting them into the mathematical relationship to predict future values of the target variable."*

The utility of the predictive models is then clear by relating first and second paragraphs above. Predictive models are a very useful tool in order to avoid misspending valuable resources (optimization), or to take a decision on how to spend them (resources assignment) to improve the incomes. There are different types of prediction models that will discussed later in this study when defining the one to be performed with Piraeus port case.

1.4 Objectives of the study

The main objective for this study is to perform a predictive analysis over the case of Piraeus Port container traffic. With the collaboration of the Piraeus Port, real data is available for this study in order to generate a forecast picture about the future of the container traffic evolution on the port with the Monte Carlo experiments and Markov chains technique method.

Objectives of the study can be briefly described in the following points:

1. Investigate the different ways of carrying out a predictive analysis, and choose

the best fit for the type of data that the port has provided.

2. Develop the predictive model with the aid of MATLAB software.

3. Test and calibrate the predictive model and set-up alternatives within it with the real data, performing convergence analysis.

4. Apply the best model and obtain results for the prediction of container traffic of Piraeus port.

5. Highlight conclusions of all the carried out work, and observe which are the weaknesses and strengths of it.

Definitely providing a good forecasting sight of the situation which can be useful to the port is the aim of this study, without pretending to be the unique element to be taken into account in a decision making framework.

Chapter 2

State of the art

Through this chapter, all the background behind the forecasting analysis is explained. From the different ways to attack the case to how to calibrate and test them, for example with different types of errors consideration.
There are many options to take into account when speaking about predictive modeling, so a brief touch over them is helpful to understand how a predictive model works more than throwing random assumptions. With it, it is deeply defined the option used for this case.

2.1 Predictive models scenario

From now on the study will focus on freight transport predictive models, since one of these is going to be applied for the Piraeus port forecast.
Speaking about port traffic forecast, there are two different ways traditionally followed by the specialists to perform their analysis, the GDP method and the ones based on data and knowledge on field.

2.1.1 Gross Domestic Product method

The GDP method is based on one main assumption: economy determines the demand for the freight. It is a simple but a correct assumption as it has been shown at Figure

1.1 with the positive growing trend of the global traffic broken at the economic crisis time. Nevertheless, it is not economy the only factor determining the evolution and therefore simplifying the prediction to only this can lead to mistaken results. This method works with multipliers and the GPD, for instance freight transport demand growing twice the GPD, the key of it is to find the appropriate multipliers. Usually, this method is combined with one of the following ones to improve its accuracy.

2.1.2 General techniques

This techniques are based on the expertise and the data obtained through the operation in time, some of them using expertise and some others statistics. There are three differentiated kinds of forecasting techniques, depending on what kind of data they rely on, that can be classified:

1. **Qualitative models**: these are methods that rely on qualitative data such as experts opinions, information about special events, and may or may not take past into consideration. This is the reason why they are commonly used when there is a lack of data for any reason. Some examples of this method are: Visionary forecast, Market research, Panel consensus, Delphy method or Historical analogy. These methods can be useful when there is no data or it is desired to use non numerical data as well, it also incorporate the experience or advising from experts. That fact can become a huge disadvantage, because if one individual input is wrong and prevails, the whole method can fail.

2. **Time series analysis**: these kind of models rely entirely on historical data and search patterns and changes in them. They are clearly statistical methods. Since this study is going to use the historical data provided by the port, this is going to be the way the forecast model is going to be developed.
   Some examples for this method are: Exponential smoothing, Moving average, X11, Box-Jenkins or Trend projections.
   Time series analysis is one of the most well-known statistical techniques to make

predictions assuming patterns will continue in the future. This can become an advantage as it is shown in Figure 1.1 where it can be seen a clear growing trend, but also a disadvantage when predicting turning points such as the global crisis. That is the main vulnerability of these methods.

3. **Causal models**: these models rely on specific information about the relationship that elements in a system may have and also taking special events into account. These are the most sophisticated forecasting tools and they try to express mathematically those relationships. Some examples are: Regression, IO model, Leading indicator, Diffusion index or Econometric model. Their strongest aspect is the finding of relationships between elements. Despite that, these methods require a higher cost and time in regard to the other types. Also, variables with insignificant coefficients are automatically discarded due to the principles of econometric, and the relationship between elements can be in constant changes.

In this study, a time-series method is the method selected to perform the forecast of the container traffic in Piraeus port. The real data provided by the port is just the monthly distribution of containers that come in and out of the port, and with that it is useless to proceed with a causal or a qualitative method.

2.2 Model background

In this section the background for the predictive model that is going to be applied is described. As stated in the previous section, and regarding to the type of data the port has delivered, a time-series method is going to be used for the study's purpose. The idea of the process is to apply Markov chains combined with the Monte Carlo experiment. With this, probabilities of going from positive container traffic growths to negative or viceversa can be obtained from data, to finally use the Monte Carlo experiment to generate forecast volumes based in the data distribution and the Markov chains results.

The methodology applied will be deeply explained in the next chapter. For this section some basic concepts about the elements mentioned above are explained.

2.2.1 Markov chains

A Markov chain can be described as a series of states, $U = u_1, u_2, ..., u_N$, and a process starting in one of those states and jumping from one to another. Each move is called *step*. For Piraeus case, that is a positive or a negative growth each month or year. The process is at state u_i and moves to u_j with a probability p_{ij} which is not dependent on the state previous to u_i. Also, the process could remain at the same state, and that would generate the probability p_{ii}. These probabilities p are called the *transition probabilities*, and they are what it is needed to find in order to define the Markov chain. An initial probability distribution, defined on U, specifies the starting state, usually set by knowledge.

A simple example of a two states Markov chain can be seen at Figure 2.1. The states are E and A. Each number besides an arrow represents the probability of the Markov process changing from one state to another state in the way the arrow does. As an example, if the state is currently A, the probability of remain at the same state for the next step is 0.6.

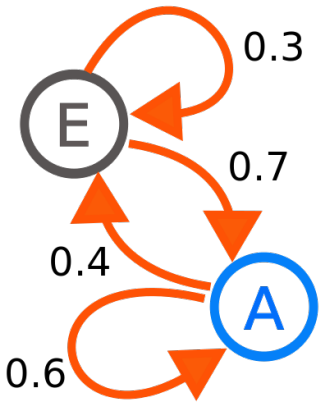


Figure 2.1: Two states Markov chain. *Source: Unknown (many sites).*

The *transition matrix* of this example is the matrix that performs a square array of all the probabilities involved on the chain:

$$T = \begin{pmatrix} 0.3 & 0.7 \\ 0.4 & 0.6 \end{pmatrix} \tag{2.1}$$

Where the rows are the states of origin and the columns the end ones. For example position $(1, 2)$ in this case is the way from state E to A, with a probability of 0.7. As it can be observed, the sum of all the probabilities with the same state as origin is 1, this corresponds to the rows of the transition matrix.
There are many interesting theorems about the transition matrix and what can be achieved with probability vectors, but as it is not going to be required for the Piraeus port, it will not be shown in this study.

2.2.2 Monte Carlo experiment

The Monte Carlo experiment or Monte Carlo simulation is named after the Monte Carlo borough in Monaco City, which is very famous due to its casino and gambling games such as roulette or dices. The simulation comes precisely for the random phenomena involved in those gambling games, generating random values each time a user plays them. The Monte Carlo experiment is very useful when solving engineering problems as it can deal with a large number of random variables, different distribution types, and highly nonlinear engineering problems.
In this method, the properties of the distributions of random variables are investigated by the use of simulated random numbers. Usually the asymptotic properties of an estimator are known but its finite sampling ones are not.
Generally the Monte Carlo experiment is carried out following this simple scheme, with possible variations:

1. Define a domain for the random samples.

2. Generate the random samples by following a probability distribution over the

domain that can be obtained from previous data knowledge.

3. Apply a deterministic computation to all the inputs generated.

4. Gather the results.

A common use of the Monte Carlo simulation is to obtain the area of a figure, such as the circle in Figure 2.2.

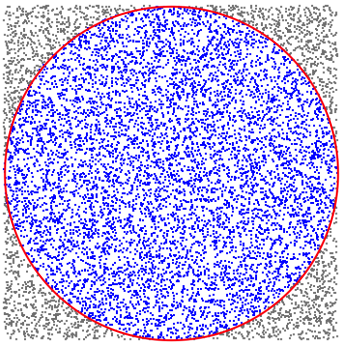


Figure 2.2: Monte Carlo experiment example. *Source: StackExchange.*

The figure is drawn over a domain and random points inside it are generated. For instance, to know the value of π:

1. Define the domain as a square one by one

2. Generate random values with a uniform distribution.

3. Draw a circle with radius one.

4. The relationship between the number of points inside the circle and outside is the same that between both areas respectively. Since it is known that a circle has area πR^2 and both radius and the square area are 1, the value of π can directly be obtained.

There is no much more theory background at Monte Carlo experiment, as it is a basic concept that can be tangled as much as the user wants with complex probability distributions or more complex necessities than a simple area computation.

2.3 Validation

Once the model is built, it is required to assess whether the results are interesting or not. Since this is a predictive model, it is hard to evaluate if a prediction will be valuable without knowing the future, but there is an easy solution when dealing with the enough data and that is to perform predictions over real data and see how close the predictions are.

For the models validation it is suggested to address to the errors computation. According to *Peng et al (2009)*, in order to assess forecasting models, when they are statistical models, it is recommended and commonly employed to compare three different errors:

- Root mean squared error (RMSE),
- Mean absolute error (MAE),
- and the Mean absolute percent error (MAPE).

2.3.1 Root mean squared error (RMSE)

The RMSE is defined by the following formulation:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Y_i - \breve{Y}_i)^2)}{n}} \tag{2.2}$$

The RMSE depends on the scale of the dependent variable. It should be used as relative measure to compare forecasts for the same series across different models. The smaller the error, the better the forecast.

2.3.2 Mean absolute error (MAE)

The MAE is defined by the following formulation:

$$MAE = \frac{\sum_{i=1}^{n} \left| Y_i - \breve{Y}_i \right|}{n} \tag{2.3}$$

This error is also dependent on the scale of the dependent variable but it is less sensitive to large deviations than the squared loss.

2.3.3 Mean absolute percent error (MAPE)

The MAPE is defined by the following formulation:

$$MAPE = \frac{100 \sum_{i=1}^{n} \left| \frac{Y_i - \breve{Y}_i}{Y_i} \right|}{n} \tag{2.4}$$

This error computation is scale independent, which is an advantage from the other two. However, MAPE has the problem of asymmetry and instability when the original value is small. It is affected by:

1. Equal errors above the actual value result in a greater MAPE.

2. Large percentage errors occur when the value of the original series is small.

3. Outliers may distort the comparisons in empirical studies.

Chapter 3

Predictive model methodology

In this chapter, the methodology followed to obtain the final results of the predictive model is discussed in order to allow the next chapter to show the results of the whole forecasting process. The data provided by the Piraeus port is presented and how it has been processed as well. After that, a scheme on what the MATLAB code has been asked to do is also explained, jumping across the main keys that have been dealt with, as for example the number of simulations, dealing with Markov chains, the random value generation for the Monte Carlo simulation, and other issues.
The main path is the one described in the previous chapter, performing a time-series predictive model with the combination of Markov chains and Monte Carlo simulation, but within that there are many different ways to take and some of them have been compared in order to obtain the best forecast traffic volumes possible.

3.1 Piraeus port data

Data process is the first step the predictive model needs before even starting to code with MATLAB. This section shows everything related to the data provided by the port, which consists fundamentally in container movements:

- Imports: number of TEUs received by the port to be stored at yard.

- Exports: number of TEUs departing from the port yard with another port as a destiny.
- Transhipment: number of TEUs involved in the operation of moving containers from one transport mode to another (can be ship-ship or even railway and road).

Figure 3.1 shows the gross detail of all this operations yearly for the port. Import and export are always considered together since it is interesting to see them as one and compare it to the volume of transhipments. Usually, the import-export volume is the one strongly related to the GDP and the evolution of economy whilst transhipment is more unpredictable not following economy growths and recessions.

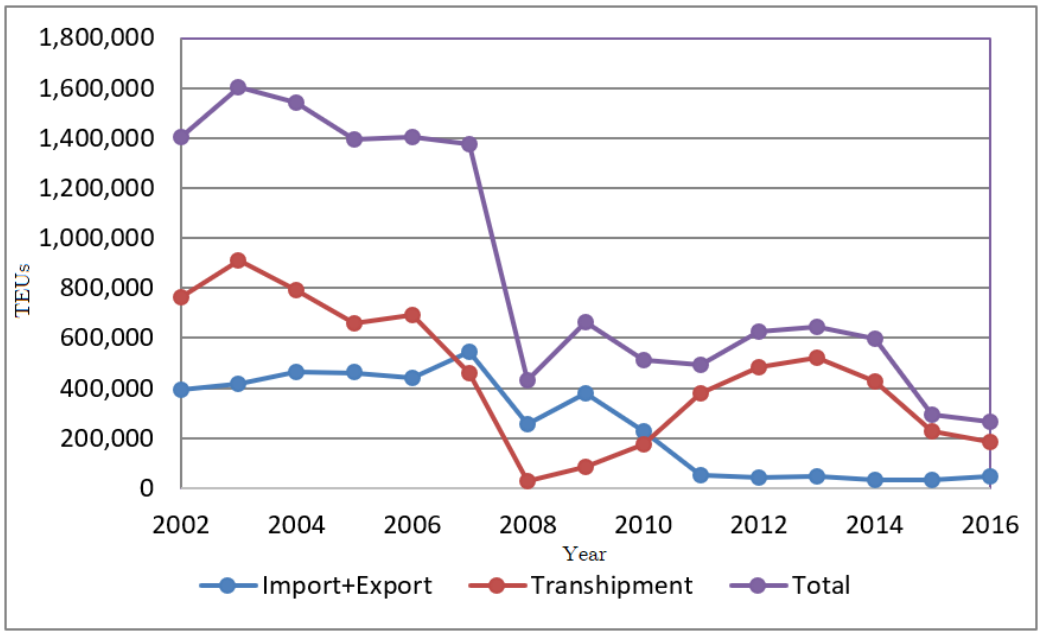


Figure 3.1: Piraeus yearly imports, exports and transhipments in TEUs. *Source: Self made.*

There are two interesting notes that the port adds to this data:

- There was a a strong industrial action due to the global crisis that closed the terminal some months between 2008 and 2009. This can be clearly seen as the collapse in the purple line representing the total of TEUs traffic at the terminal in Figure 3.1.

- In 2010 an intra-port competition started with another provider. There is no monthly data for the other operator, which is not a big deal since the study can be based on just the provider that has been operating since 2002 and it is going to produce less disturbance due to intern issues to each provider.

It is verified that the import-export volumes have more relationship with the economy wellness: growing until the crisis, struggling some years after that, and finally starting to grow again last years. Transhipment is more unpredictable and it is related to other factors, it is seen at the graph that its growths and decreases are not in line with the imports and exports trends. Port also provided monthly data for import, export and transhipment for the periods 2003-2008 and 2011-2016 which are interesting to see:

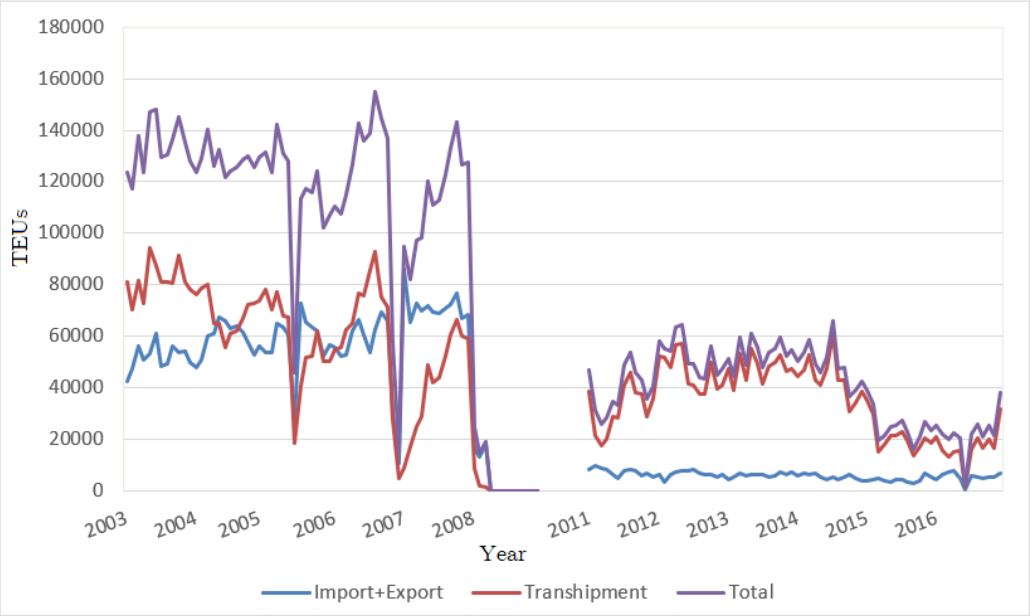
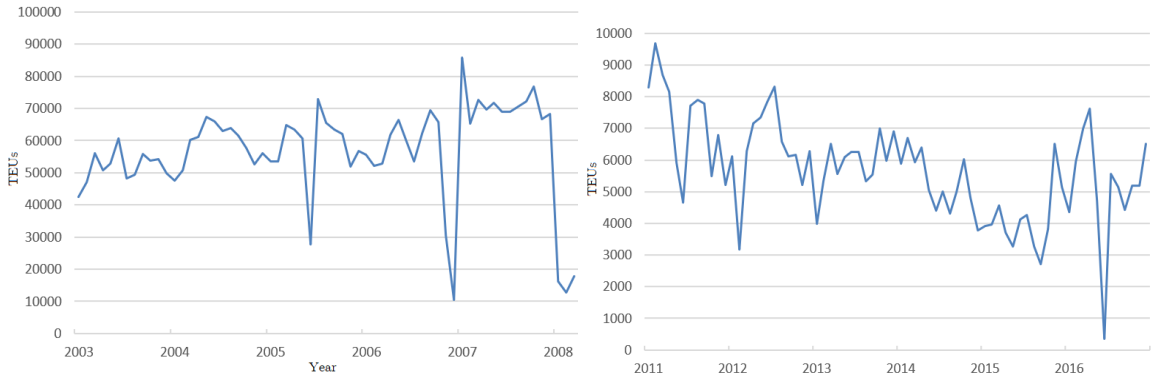


Figure 3.2: Piraeus monthly imports, exports and transhipments in TEUs. *Source: Self made.*

The break 2009-2010 in data is reflected in the graph. Again it is seen that the transhipment is way more unpredictable as a trend for the future can be foreseen in the import-export line. This will impact directly on the predictive model questioning it if it can be applied for totals or transhipments and imports-exports separately.

There will be another point to take into account when dealing with monthly data:

the seasonality demand. The fluctuations induced by the month or season of the year that is going on can mean significant deviations from the global or yearly trend. It is difficult to add this element into a time-series method but there are some other ways to deal with it if this is very pronounced and really affects the forecast.



(a) Period 2003-2008 (b) Period 2011-2016

Figure 3.3: Piraeus monthly imports-exports in TEUs. *Source: Self made.*

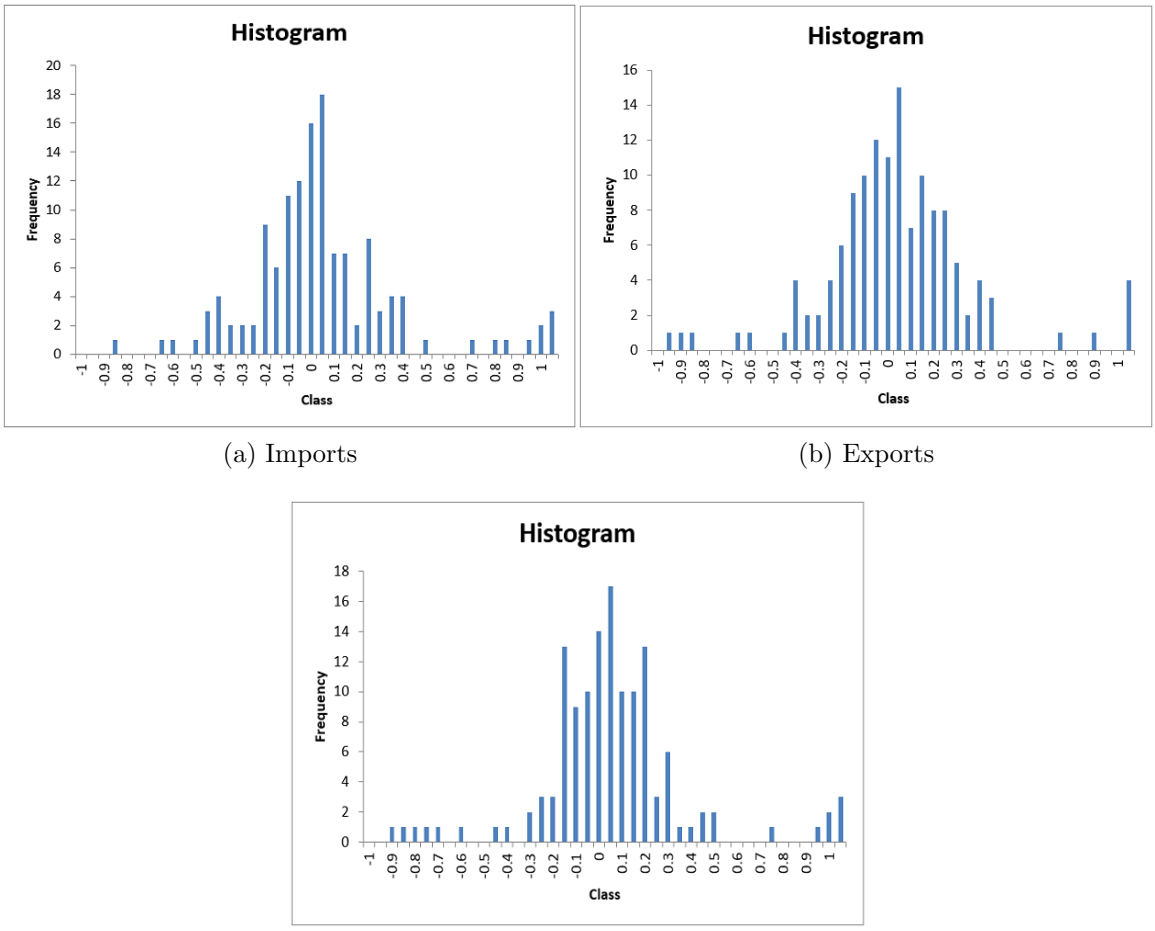
Figures 3.3a and 3.3b show the monthly data for the import-export traffic. The two periods are separated to have a better sight of the trends, since the second period is in a lower scale TEUs order.

As it is seen in Figure 3.3a, there is a continued growing trend (despite a few collapses) until the year 2008 when the crisis starts. Also in Figure 3.3b the slow recovery from the crisis can be deducted, with a decreasing trend until the ends of 2015, moment in which the port seems to be starting to recover again.

The other thing required from the data are the growths distributions, in order to apply Monte Carlo experiment, it is needed to generate random numbers with a probability distribution that can be obtained from the raw data. Figure 3.4 shows the histograms for the growths of each type of traffic. The growths can easily be obtained with:

$$Growth = \frac{X_{i+1} - X_i}{X_i} \tag{3.1}$$

where X refers to the monthly or yearly data.



(a) Imports

(b) Exports

(c) Transhipments

Figure 3.4: Piraeus monthly growths histograms for each type of traffic. *Source: Self made.*

All of the histograms are showing a possible normal distribution, most specifically the imports and the exports (which reinforces the arguments given above). This is going to be discussed when choosing the appropriate probability distribution in the Monte Carlo experiment section.

With this global visualization of the data the next step is to start building a code for the predictive model, and comparing the different ways to achieve it.

3.2 Markov chains for the Piraeus case

Once the data process is finished, the MATLAB code starts by implementing it. It is chosen which data is to be introduced, whether it is monthly, yearly and how many time steps are wished. After that, the growths are computed with the formulation shown at equation 3.1.
With this, the Markov chain has to be defined.

3.2.1 Two states Markov chain

It will consists on testing the probabilities of having positive or negative growths for the year X_{i+1}, so a start can be a simple two states chain (positive and negative growths) that will generate four probabilities:

- Positive to positive, PP.
- Positive to negative, PN.
- Negative to positive, NP.
- Negative to negative, NN.

And hence, the transition matrix of the chain would be:

$$T = \begin{pmatrix} PP & PN \\ NP & NN \end{pmatrix} \tag{3.2}$$

Just to remember, all the rows of this matrix must sum up one, so for example $NP + NN = 1$. Figure 3.5 shows the scheme of the previous transition matrix.

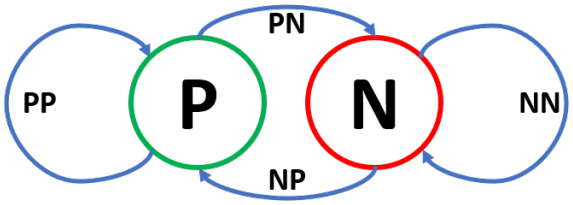


Figure 3.5: Markov chain with two states case. *Source: Self made.*

This probabilities will be computed from the raw data growths that have been obtained. With the use of conditionals at MATLAB, it is checked the growth of the step i and the growth of the step $i+1$, and successively for all the growths obtained in order. With it the number of positive to positive, positive to negative and the rest of it can be known, and just with a simple division find the probabilities. For example:

$$PP = \frac{Number\,of\,positive\,to\,positive\,growth\,steps}{Number\,of\,steps\,with\,a\,positive\,origin} \tag{3.3}$$

And similarly for the other states, the transition matrix is fulfilled for each case it is wanted to test.

Once the transition matrix is defined, a growth matrix is generated with the size $[Number\,of\,time\,steps,\ Number\,of\,simulations]$, where the number of time steps refer to how many months or years are to be predicted, and the number of simulations is the number of times MATLAB will generate predictions in order to obtain the mean of all of them.

For each position, a random number between zero and one is generated, and the previous time step growth is checked (whether it is positive or negative) so it is known if the origin is a P or a N, and finally depending on the random number generated select if the current step will be positive or negative. For example if the transition matrix is:

$$T = \begin{pmatrix} PP & PN \\ NP & NN \end{pmatrix} = \begin{pmatrix} 0.6 & 0.4 \\ 0.7 & 0.3 \end{pmatrix} \tag{3.4}$$

Then if the previous year is negative and the random number is 0.6 (<0.7) it falls into NP case and the current year will have a positive growths. If the random number

would have been 0.8 (>0.7) the current year would be negative due to the NN case. Repeating this process the growths matrix can be easily fulfilled by MATLAB and the Markov chain process is ended with its purpose achieved.

3.2.2 Four states Markov chain

One of the advantages of these Markov chains is that it can be defined in different ways as the user desires. A more entangled way to do it is to define a four states chain instead of two. Whether it is useful or not will be tested later, but it can be of interest to see the impact of changing the Markov chain shape over the forecast results.

The process is basically the same than in the two states Markov chain. The difference is that there will be two new states. Since the growths are the interesting value, the new distribution will also spin around them. For this new chain, to reach four states, time steps are going to be considered as a pack of two, that will lead to the following states: two positive steps, positive and negative steps, negative and positive and finally two negative steps. With that, the probabilities can be seen in the transition matrix:

$$T = \begin{pmatrix} PPPP & PPPN & PPNP & PPNN \\ PNPP & PNPN & PNNP & PNNN \\ NPPP & NPPN & NPNP & NPNN \\ NNPP & NNPN & NNNP & NNNN \end{pmatrix} \tag{3.5}$$

With all rows summing up one. Figure 3.6 shows the scheme for the previous transition matrix.

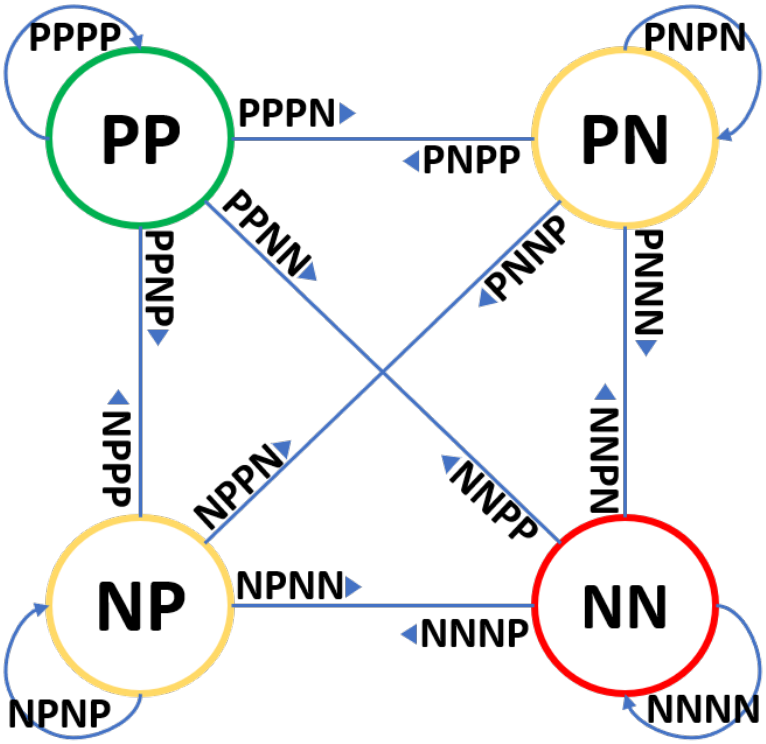


Figure 3.6: Markov chain with four states case. *Source: Self made.*

The rest of the process is very similar. The probabilities of transition matrix, T, are found from the data used, computing growths in pairs. Once the matrix is fulfilled, the growth matrix can be generated again depending on number of steps (this time it is mandatory to use an even number) and simulations, by checking the last two steps and generating a random number [0,1] that falls into one of the probabilities of each row of the matrix.

3.3 Monte Carlo experiment for the Piraeus case

Departing from the growth matrix obtained previously, a matrix which only indicates if the time step to predict has a positive or a negative growth, it is now required to transform it into a matrix with growth values, that means the predicted changes for the predicted time steps.

This is going to be achieved through the Monte Carlo experiment. For it, a probability

distribution of real data growths must be find. As it is shown in Figures 3.4a, 3.4b and 3.4c it is very tempting to choose the normal distribution for the experiment, and that can be a good first attempt.

Therefore, if the normal distribution is to be used, the mean and standard deviation have to be obtained from the growths of the original data. Once the distribution is determined, MATLAB is able to generate random values with that distribution. With the growth matrix obtained from Markov chains, MATLAB generates random growths (positives or negatives depending on what is in each position of the matrix) based on the normal distribution. For example if the growth matrix is:

$$G = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \tag{3.6}$$

Being 0's a negative growth for that time step and 1's a positive one, then the Monte Carlo simulation allows us to transform it into:

$$G' = \begin{pmatrix} 345 & 234 & 234 & -235 \\ 453 & -43 & 353 & 25 \\ -532 & 123 & -1045 & -535 \\ -12 & -534 & 124 & 657 \end{pmatrix} \tag{3.7}$$

And the only last thing to do is to transform it into the TEU matrix:

$$TEU(1,n) = Data(end) * G'(1,n) + Data(end) \tag{3.8}$$

with $n = Number\,of\,simulations$, and $Data(end) = Last\,TEU\,value\,in\,port\,data$. And:

$$TEU(t,n) = TEU(t-1,n) * G'(t,n) + TEU(t-1,n) \tag{3.9}$$

With the TEU matrix, the forecast for the time steps t for the Piraeus port is obtained. Finally the mean of all simulations is computed (mean of all rows) and the standard deviation to show the results, that will be seen in the next chapter.

3.3.1 Other probability distributions

As commented above, there are multiple options for the Monte Carlo experiment. The random value can be generated from all the probability functions existing. Nevertheless, there has to be a concordance between the data growths and the distribution, since the best the distribution fits, the best predictive model should be expect.
In order to fit a distribution for a given data, MATLAB provides a function called *Distribution Fitter* that automatically fits a variety of included distributions to the data. An example can be seen in Figure 3.7 for a given data: This is an exponen-

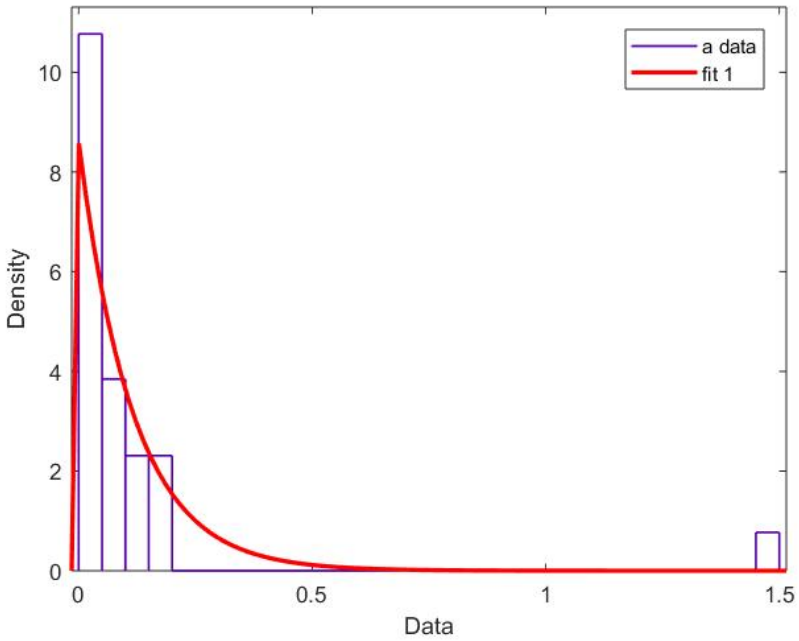


Figure 3.7: Example of a data fitting distribution in MATLAB. *Source: Self made.*

tial distribution fit to a random data. The software also returns the most important parameters. For instance:

```
Distribution:  Exponential
Log-Likelihood:  29.9637
Mean:  0.116199
Variance:  0.0135022
```

This is an interesting point since it allows to compare the fitting performance comparing the Log-Likelihood between distributions. There is also another index called *Akaike Information Criterion (AIC)* that can be obtained and combined with the Log-Likelihood for the comparison.

A function that compares every distribution for a given data has been used as well. The function takes the data as an input and shows a classification from best to worst distribution based on the combination of the Log-Likelihood (better the distribution fit as bigger it is) and the AIC (better the distribution as smaller it is) criteria. Here is an example of what the function returns when used with the same previous data:

Distribution	**Log-Likelihood**	**AIC**
Generalized Extreme Value	3.89E+01	-7.17E+01
Log-Logistic	3.86E+01	-7.32E+01
Inverse Gaussian	3.85E+01	-7.30E+01
Log-Normal	3.83E+01	-7.26E+01
Generalized Pareto	3.69E+01	-6.97E+01
Birnbaum-Saunders	3.63E+01	-6.86E+01
Weibull	3.33E+01	-6.27E+01
Gamma	3.14E+01	-5.88E+01
t-TocationScale	3.10E+01	-5.61E+01
Exponential	2.99E+01	-5.79E+01
Nakagami	2.48E+01	-4.55E+01
Logistic	1.12E+01	-1.84E+01
Normal	-3.28E+00	1.06E+01
Uniform	-9.80E+00	2.36E+01
Extreme Value	-1.74E+01	3.89E+01
Rayleigh	-2.40E+01	4.99E+01
Rician	-2.40E+01	5.19E+01

Table 3.1: Results of applying function to the previous data set. *Source: Self made.*

It can be observed that the exponential, which seemed a good distribution fit, is the 10th. So this function is useful to apply different distribution probabilities within the Monte Carlo experiment.

With this, a result comparison between normal distribution and the best fit (output from the function explained above) has been carried out in the next chapter, to see if it is worth to fit different distributions for each set of data or just a normal one.

3.4 Number of simulations

It is also advisable to define the number of simulations that MATLAB code has to deal with (this is the n dimension of most of the important matrices). For that, some convergence tests have been carried out with different data sets, comparing the predicted TEUs for same time step between code runs with different number of simulations.

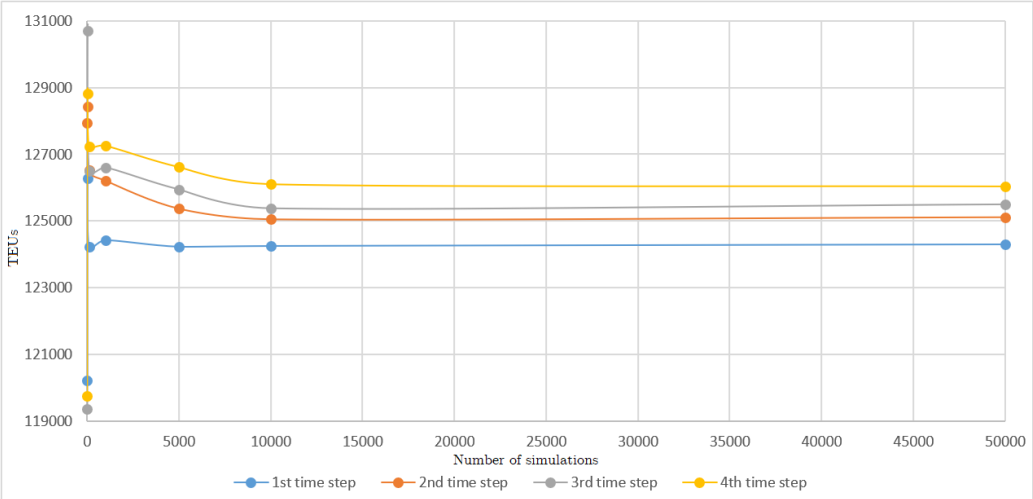


Figure 3.8: Convergence test with the same prediction and different number of simulations and time steps. *Source: Self made.*

At Figure 3.8 one of the convergence tests can be seen. Each line represents one of the four time steps predicted (in this case, four years consecutive to the port data). In the y-axis the number of TEUs predicted for different number of simulations.

The number of simulations used for the convergence test for all the four time steps have been: 1, 10, 100, 1000, $\dot{5}$000, 10000, 50000, respectively.
Each prediction is obtained from the mean of all the simulations (that means that the last value for all time steps is the mean of 50000 values). It can be seen that for each further time step, more simulations are needed in order to have a stable value. For this simulation in particular, around 8000 simulations is enough to reach the convergence.
It is important to have this analysis in mind, not to be the most precise by choosing 8000 or 9000 simulations, but to see that a convergence is reached meaning that the model is stable and does not give totally random predictions each time it changes the number of simulations, which would mean that the model would have been useless.
Finally, to make this analysis possible MATLAB is needed to define a *seed* so that the results of the analysis are the same if nothing is changed between runs (this is because of dealing with random numbers generation). For that, the function *rng* in MATLAB allows us to control the generation, extracted from MATLAB: *"Rng(seed) seeds the random number generator using the non-negative integer seed so that rand, randi, and randn produce a predictable sequence of numbers."*.

3.5 Process flow diagram

To sum up, a flow diagram with all the process is shown at Figure 3.9. It shows the steps followed to reach the final prediction with the Monte Carlo experiments and Markov chains techniques described previously.
The steps are placed in the same order as processed with MATLAB software. Diagram can be seen at Figure 3.9.

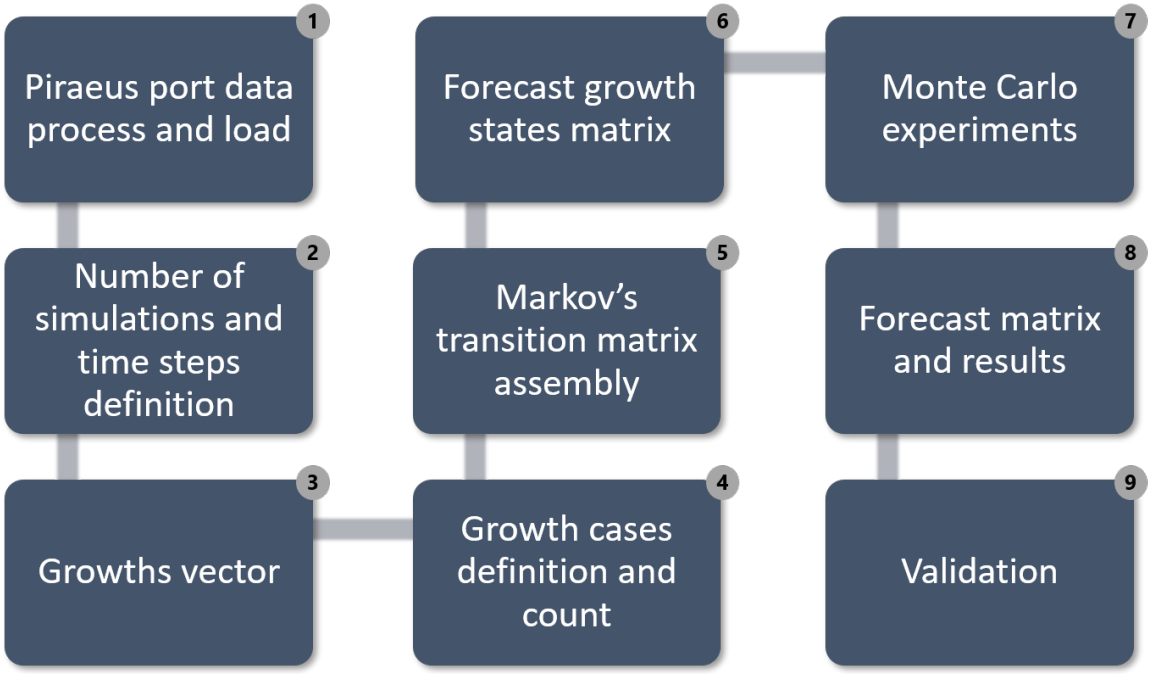


Figure 3.9: Flow diagram of the predictive model. *Source: Self made.*

Steps 1 to 3 refer to the data treatment and parameters definition. Steps 4 and 5 define the Markov chains by defining the number of growth cases desired (e.g. PP or PPPP depending on how many chains are generated) and the transition matrix from the data growth cases counting. Steps 6 and 7 generate the Monte Carlo experiments by generating a forecast growth matrix (1's and 0's for positive and negative growths respectively) based on the transition matrix probabilities and finally transforming it into real growths based on a chosen distribution (e.g. normal distribution). Finally step 8 produces the predicted TEUs matrix based on the growths previously computed at step 7, and from it the mean of all simulations and the standard deviation. Step 9 is only considered if calibrating the model with real data, validating it based on the error computation (RMSE, MAE and MAPE).

Chapter 4

Results of the predictive model

Through this chapter the results of the predictive model will be shown. All the methodology applied to reach the results is explained in the previous chapter. Monthly and yearly data has been used to produce them, and the different methods explained above such as different probability functions or different Markov chains, just to find the best forecast possible.
In order to compare between methods and to choose the best options, the main error techniques have been used (refer to Chapter 2 for the basic knowledge).

4.1 Benchmark model

The first result comes with the simplest methodology described in the previous chapter for the yearly data. The traffic values used come from Figure 3.1 as total values. The prediction is made under a two states Markov chain and normal distribution for random values generation in the Monte Carlo simulation. The result gives an idea of how the predictive model is working and if the results are coherent.
This result is going to be produced from ten simulations so that all the simulations can properly be seen.

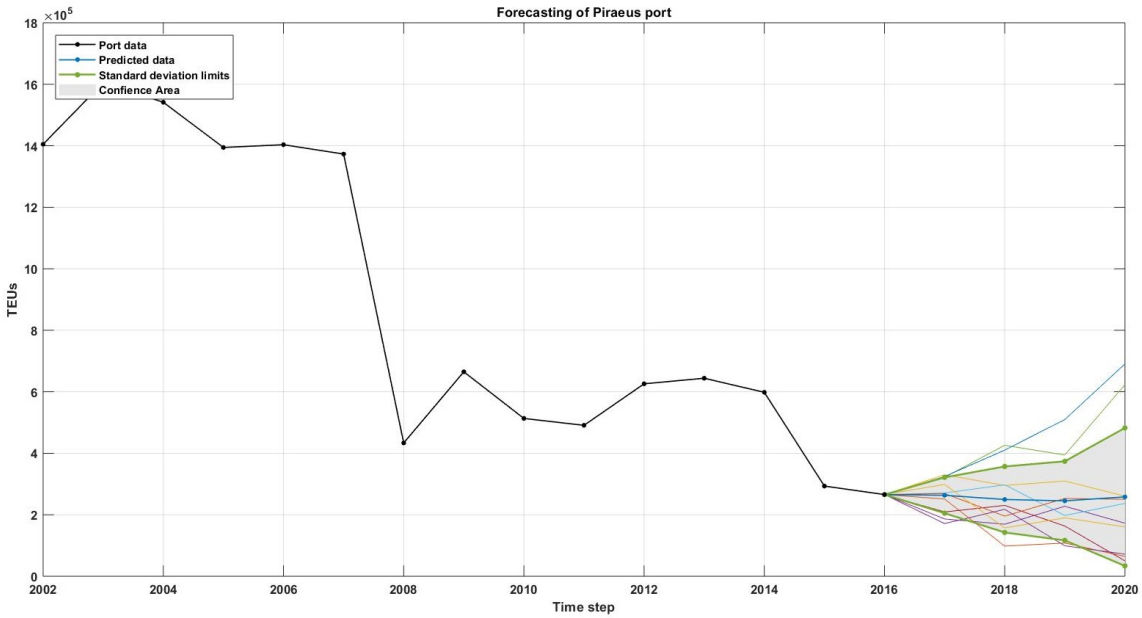
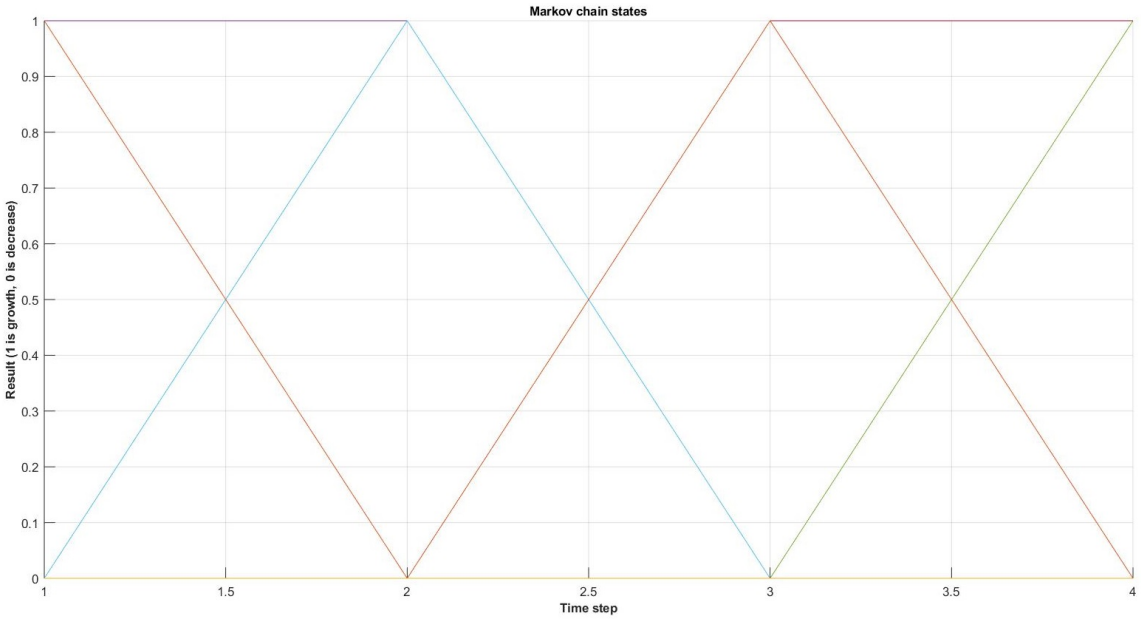


Figure 4.1: First prediction over next four years. *Source: Self made.*

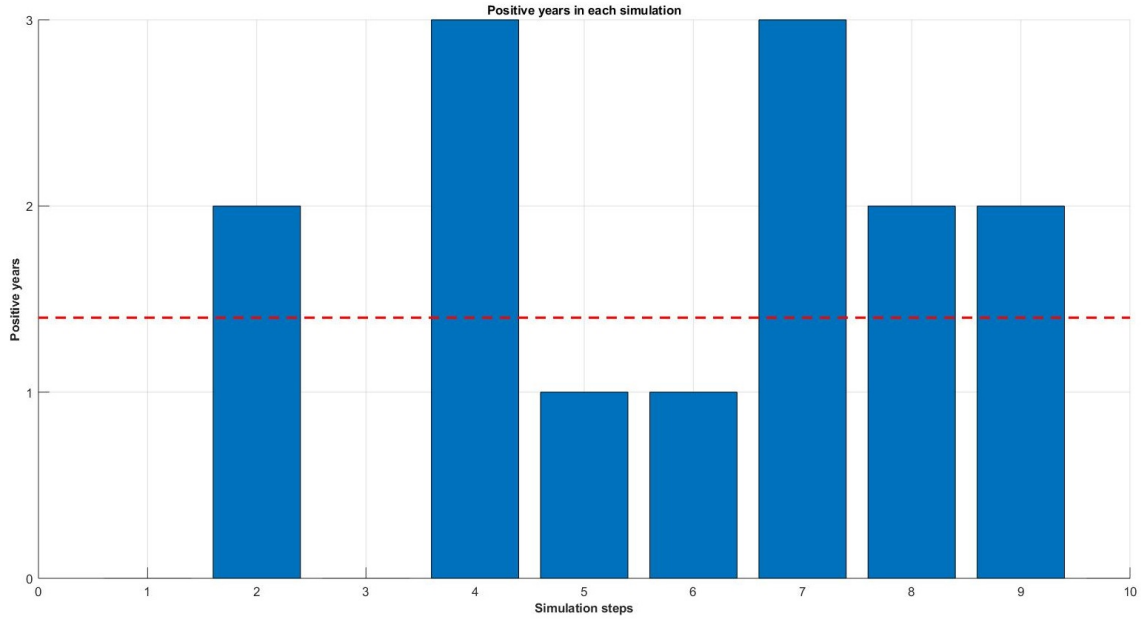
The graph shows the real data provided by the port for the period 2002-2016 and the four years prediction. The prediction is carried out from the 10 simulations, and all of them can be seen as different colour lines. The mean (which is the final prediction is the blue thicker line) and the standard deviation (green thicker lines wrapping up the confidence area in gray) are also plotted.

It is hard to predict an exact number of TEUs traffic for the next years (blue line), but the confidence area gives us an interval of the most probable TEUs traffic prediction. The more years predicted, the wider the confident area is, this is completely logic when thinking that each simulation can draw away more and more from the mean trend.

It is also interesting to see some other middle points inside the code that have been explained in the last chapter. Figure 4.2a in the next page shows the Markov chain for each time step predicted over the 10 simulations (where 1 is positive growth and 0 is negative one). Figure 4.2b shows the amount of positive growths in each of the ten simulations, and the mean of positive growing predicted years (somewhere around 1.4 positive growing years are predicted).



(a) Markov chain states for each simulation.



(b) Amount of positive years in each simulation.

Figure 4.2: Markov chains and growth predictions for each simulation. *Source: Self made.*

It is also possible to see the transition matrix generated for this two states Markov chain:

$$T = \begin{pmatrix} PP & PN \\ NP & NN \end{pmatrix} = \begin{pmatrix} 0.2 & 0.8 \\ 0.375 & 0.625 \end{pmatrix} \tag{4.1}$$

And finally, to know that the Monte Carlo simulation has been performed over a normal distribution random values with the parameters: $\mu = 0.19$ *and* $\sigma = 0.22$.
The monthly data is also tested, in order to predict the year 2016 (12 months). The data used is the period 2011-2015 (Figure 3.3b since there is a break and big alterations before it.

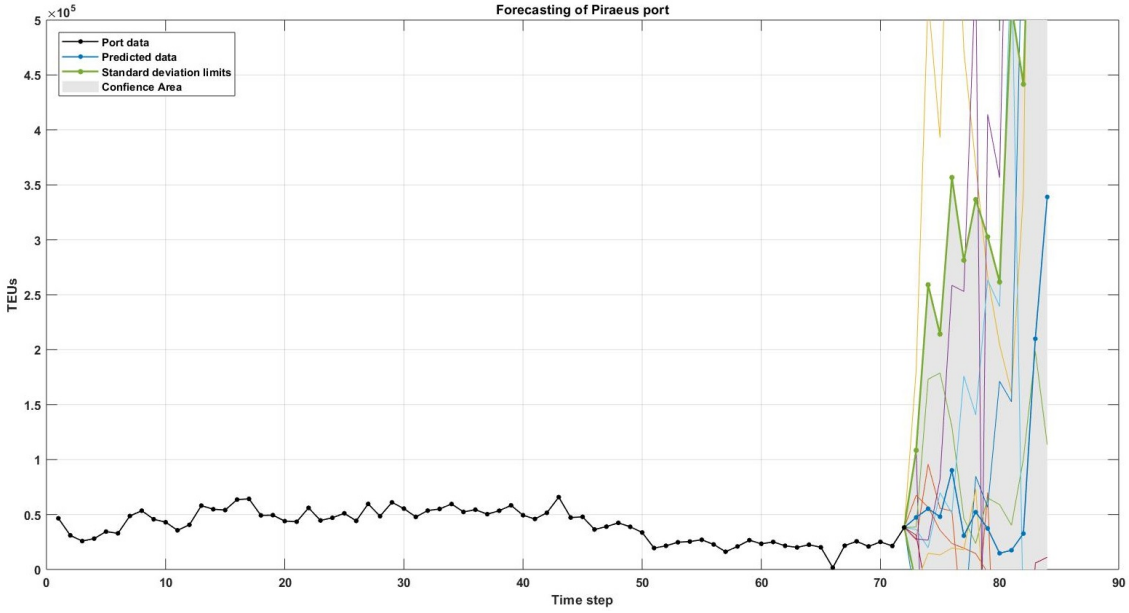


Figure 4.3: First prediction over 2016 monthly. *Source: Self made.*

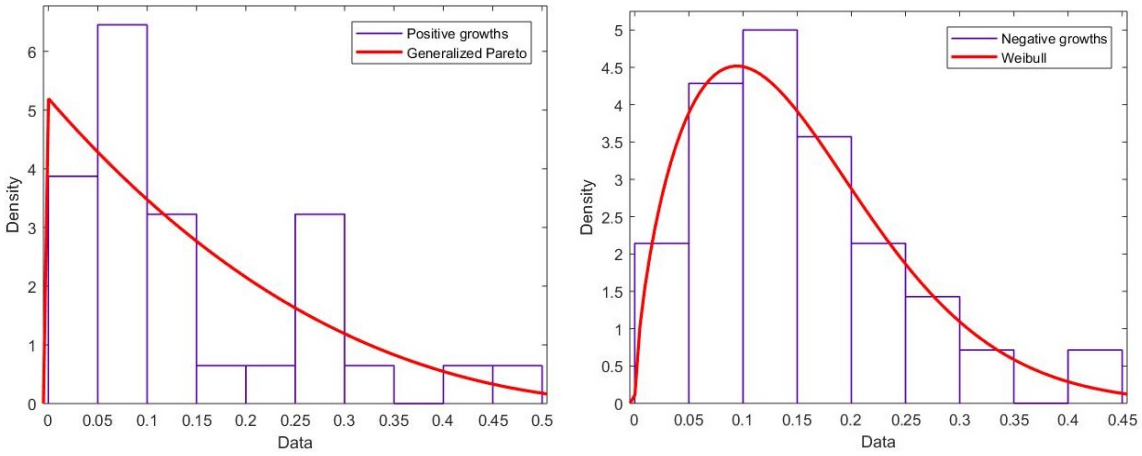
This time, the prediction is going on twelve time steps, and that is making the last steps to have a useless confident area, since it is telling that nearly everything can happen. The first months are in the line with the previous prediction. Also, this time the data is closer to 0 and some of the prediction simulations which are getting more negative growths than positive ones are falling into the negative zone, this can happens as there is no limit established (which could be), so they are considered as 0 TEUs traffic.

The monthly data is more difficult to use as a prediction since it has much more variability and it is needed to predict 12 times more steps per year. But this data can be useful for the calibration and comparison of methods, since it has much more volume of TEUs traffic that are useful to fit probability distribution and for the accuracy of Markov chains probabilities.
Once it is shown that the prelusive predictive model is working, it is time to try different methods and to calibrate them in order to find the best and final predictive model.

4.2 Probability distribution calibration

In this section, different probability distributions are tested. As explained in Chapter 3, MATLAB provides us with a distribution fitting tool, and with the combined use of an automatic testing of all the distributions over any data set it is possible to find the best fit for any given data set.
For this calibrations, monthly data is used as discussed above, since the amount of data is twelve times bigger than the yearly data. Therefore, the first step is to check the best fit for the 2011-2016 data set, but this time saving the last year (2016) for result comparison and error computation. Also to be mentioned that data has to be split between negative and positive growths, without this step only normal distribution can be fitted, so it loses all the interest.
Also, the normal distribution is going to be computed for the same amount of simulation steps, since the objective of this calibration is to compare whether the normal distribution or the best fit distribution (regarding MATLAB function) is the best forecast.

Figure 4.4 shows the distribution fitting for positive and negative growths over the 2011-2015 monthly data:



(a) Positive growths Generalized Pareto fitting. (b) Negative growths Weibull fitting.

Figure 4.4: Best fitting distributions for 2011-2015 monthly data. *Source: Self made.*

As it is shown in Figure 4.4, the best fits for the positive growths and negative ones are a Generalized Pareto distribution and a Weibull one respectively. Also the normal distribution is shown:

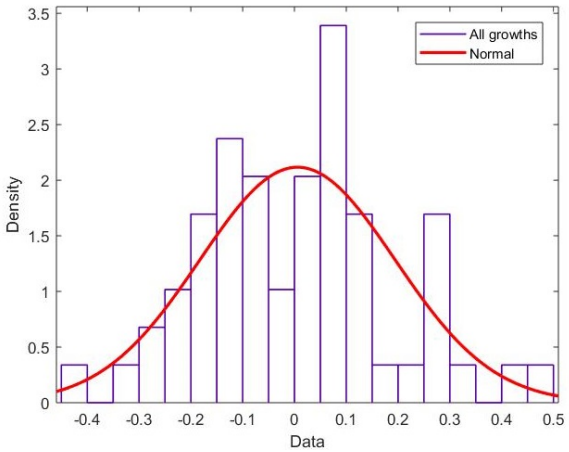


Figure 4.5: Normal distribution over all growths data. *Source: Self made.*

The parameters obtained from this fittings can be seen in Table 4.1. Parameters refer to the distribution ones, for example in the normal distribution parameter 1 refers to the mean and parameter 2 to the standard deviation. As it can be seen, the optimal

fits given by MATLAB have a clearly better fitting values (log-likelihood and AIC) becoming around twice better in both cases compared to the normal distribution.

Distribution	**Parameter 1**	**Parameter 2**	**Log-Likelihood**	**AIC**
Generalized Pareto	-2.845e-01	1.924e-01	2.891e+01	-5.383e+01
Weibull	1.696e-01	1.625e+00	2.901e+01	-5.401e+01
Normal	5.863e-03	1.869e-01	1.525e+01	-2.650e+01

Table 4.1: Results of the curve fitting in MATLAB. *Source: Self made.*

Results of the forecast with Generalized Pareto and Weibull distribution look like this:

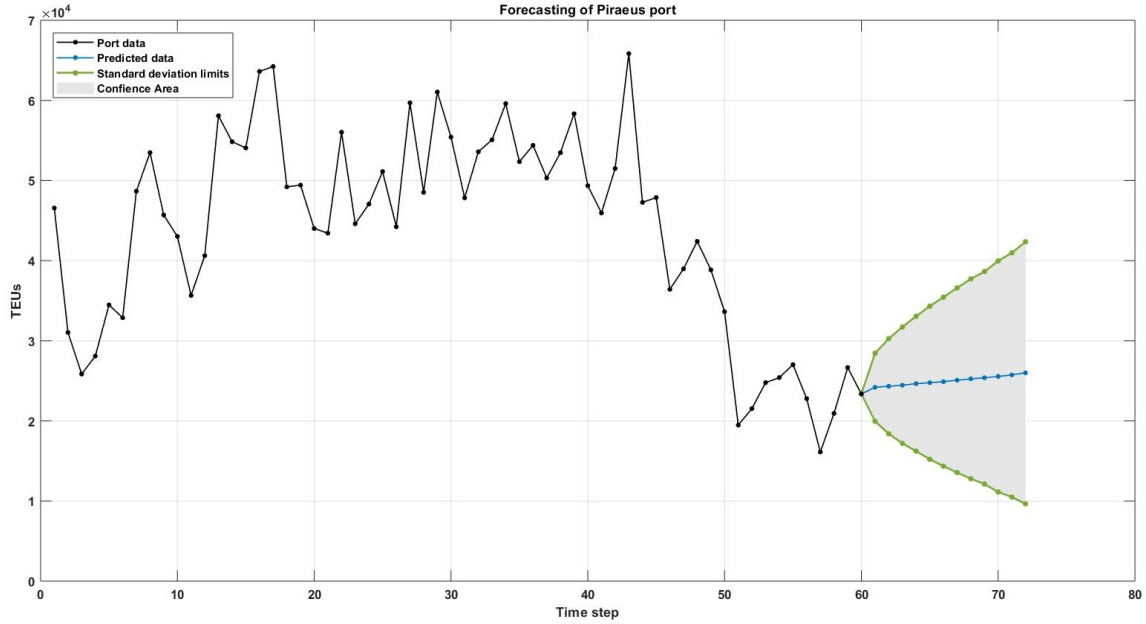


Figure 4.6: Forecast with Generalized Pareto and Weibull distributions. *Source: Self made.*

The mean shows a very stable solution, and the confidence area is small enough to be useful even at twelve steps prediction. The normal distribution shows a similar graph so it wont be plotted.

The next step is to compare the results and compute the errors in order to determine the best option for this case. Results can be seen in Table 4.2.

Month	Real (2016)	Predicted Normal	Predicted GP&W	Confidence area Normal	Confidence area GP&W
January	25046	**24195**	24194	4354	4247
February	21557	**24225**	24322	6077	5936
March	20078	**24384**	24456	7445	7266
April	22526	**24566**	24643	8673	8416
May	20170	**24719**	24765	9888	9554
June	1656	**24845**	24894	10891	10547
July	21690	25077	**25075**	11841	11521
August	25610	25222	**25253**	12781	12469
September	20971	**25263**	25379	13676	13258
October	25083	**25357**	25546	14746	14419
November	21418	**25578**	25738	15756	15241
December	38189	25791	**26003**	16823	16351

Table 4.2: Results (in TEUs per time step) comparison between Normal and Generalized Pareto plus Weibull distributions. *Source: Self made.*

Results in bold are the closer predictions to real value.

Confidence area values represent $\pm$ *std deviation* and are colored in green if real value falls into it, both methods are producing the same results in terms of it (which means the prediction is acceptable). Only the 2016 June collapse is out and unpredictable by the methods. Also RMSE, MAE and MAPE errors (explained in Chapter 2) are computed:

Error	Normal	GP&W
MAE	5208.50	5255.33
MAPE	129.98	130.45
RMSE	8114.31	8121.63

Table 4.3: Compared RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage) errors between Normal and Generalized Pareto plus Weibull distributions. *Source: Self made.*

As it is shown, the Normal probability distribution is giving a better performance regarding all three error computation (the lower value the better for all three errors). Despite the other two distributions are better fits for the growth data, normal distribution reaches a higher level of adaptation for the forecast model in this case.

4.2.1 2003-2007 distribution calibration

Now the same procedure is followed with the intention of having a second view of the calibration. The data set goes from 2003 to 2007 and the last year is used again as real data for comparison. This time, however, the prediction starts at November 2006 in order to avoid the collapse that can be seen in Figure 3.2 since it induces the predictive model to fail.

As for the comparison, again normal distribution is used, and the best fit for both positive and negative growths data is a Generalized Extreme Value distribution regarding Log-Likelihood and AIC criteria.

Month	Real (2007)	Predicted Normal	Predicted GeV	Confidence area Normal	Confidence area GeV
January	94586	90272	**90857**	22566	212780
February	82258	92357	**87306**	32805	552547
March	97305	**94290**	89953	41260	591217
April	98249	**96268**	92124	49047	637584
May	120303	**98637**	94947	57335	783376
June	110793	**100542**	97923	64429	817163
July	112635	103216	**105721**	72207	1036561
August	122744	105034	**122648**	79827	1821837
September	132857	106300	**128843**	86887	2133636
October	143082	108147	**134196**	95687	2058440
November	126466	110829	**140333**	106338	2096966
December	127380	**113673**	150907	116716	2277423

Table 4.4: Results (in TEUs per time step) comparison between Normal and Generalized Extreme Value distributions. *Source: Self made.*

Error	Normal	GeV
MAE	14107.58	9815.33
MAPE	11.63	8.45
RMSE	17001.47	12343.08

Table 4.5: Compared RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage) errors between Normal and Generalized Extreme Value distributions. *Source: Self made.*

This time, the Generalized Extreme Value distribution has a better error performance.

Despite that, the standard deviation (which defines the confidence area) is way too much bigger than normal distribution. For example, in December 2007 the standard deviation with the normal distribution is 116716 and for the generalized extreme value 2277423 which is a totally useless confidence area. So again, the normal distribution should be used in this case.

4.3 Markov chain states calibration

In this section, the method described in Chapter 3 is tested, in order to figure out the impact generated due to different Markov chain definitions. This test runs with a four states Markov chain (PP, PN, NP and NN) regarding positive or negative growths. The test is carried out for the same data set (predicting 2016 monthly) and with the same number of simulations (10000).

Also, the calibrations are made with the results coming out from the normal distribution and the generalized Pareto plus Weibull (best distribution fits for positive and negative growths data respectively) ones in order to compare all of them.

Graphical results are very similar to the previous section so only the numbers are going to be shown, which is the interesting part for comparative purposes.

As shown, the results are also very acceptable with small confidence areas which give the most valuable point to the prediction. Also, the errors for the exact (mean) predictions are computed as usual:

Again normal distribution is giving best results in terms of error (despite that Generalized Pareto plus Weibull is not giving much worse results). Compared to a two states Markov chain, the errors are very similar, with the four states chain giving best results in all of the three errors, which is indicating that the increase of the states has improved the prediction.

Month	Real (2016)	Predicted 4 states Normal	Predicted 4 states GP&W	Confidence area Normal	Confidence area GP&W
January	25046	23771	**23782**	4383	4249
February	21557	**24036**	24042	5855	5734
March	20078	**23926**	24014	6577	6566
April	22526	**24276**	24380	7524	7441
May	20170	**24124**	24220	8331	8269
June	1656	**24676**	24772	9373	9195
July	21690	**24559**	24746	10087	9966
August	25610	25008	**25196**	10987	10797
September	20971	**24822**	25061	11458	11380
October	25083	**25261**	25568	12218	12204
November	21418	**25186**	25407	12964	12759
December	38189	**25663**	25968	13956	13789

Table 4.6: Results (in TEUs per time step) comparison between Normal and Generalized Pareto plus Weibull distributions with four states Markov chain. *Source: Self made.*

Error	Normal 4 states	GP&W 4 states
MAE	5010.00	5080.00
MAPE	128.19	129.01
RMSE	7988.27	8006.21

Table 4.7: RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage) errors between Normal and Generalized Pareto plus Weibull distributions with four states Markov chain. *Source: Self made.*

4.3.1 2003-2007 Markov chains calibration

Again, the analysis is repeated for the 2003-2007 data set, which is less stable and harder to predict, but which can help giving an overview of performance in terms of comparison between methods.

Results and error computation are shown in Tables 4.8 and :

The generalized extreme value (best fit for positive and negative growths) is giving again better results in terms of errors, despite that again the confidence area is really useless as a prediction. For that, it can be said again that the normal prediction should be considered since the confidence area, which is still big, can be acceptable.

Error	Normal 4 states	GeV 4 states
MAE	10943.92	8158.33
MAPE	9.15	7.35
RMSE	13824.62	9906.94

Table 4.8: RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage) errors between Normal and Generalized Extreme Value distributions with four states Markov chain. *Source: Self made.*

Month	Real (2007)	Predicted 4 states Normal	Predicted 4 states GeV	Confidence area Normal	Confidence area GeV
January	94586	**95904**	96488	24261	315862
February	82258	**97403**	100199	37226	369659
March	97305	**96616**	93911	44211	344545
April	98249	**100019**	101171	51335	520114
May	120303	**102727**	99903	61063	493933
June	110793	**105790**	103429	70418	555306
July	112635	**106438**	104351	76609	578316
August	122744	109667	**117042**	85534	1255555
September	132857	110824	**124483**	92586	1379541
October	143082	114205	**131350**	102543	1485248
November	126466	115394	**129982**	110173	1552599
December	127380	**118810**	133749	118766	1634193

Table 4.9: Results (in TEUs per time step) comparison between Normal and Generalized Extreme Value distributions with four states Markov chain. *Source: Self made.*

Again all error terms have been reduced compared to the two states Markov chain 2003-2007 results, which reinforces the previous analysis results.

4.4 Calibration with a different data set

Last tests will be run over data. As it is shown in 3.2 transhipment data induces a instability for the whole. It might be interesting for the prediction to test it over import+export data only, since it is more stable and as explained, time series methods are specially recommendable with this kind of distributions. All the methods are tested on different data (i/e instead of t+i/e), and the results are shown all together:

Month	Real	Predicted Normal	Predicted GP&t-Loc	Predicted 4 states Normal	Predicted 4 states GP&t-Loc
January	4359	5441	5450	**5256**	5271
February	5946	5465	5515	5532	**5547**
March	6994	5550	**5603**	5563	5573
April	7628	5647	5720	5721	**5752**
May	4702	5735	**5814**	5768	5842
June	356	5798	**5913**	5921	6007
July	5555	5910	**6024**	6006	6124
August	5133	5975	**6134**	6130	6276
September	4418	6011	**6232**	6183	6361
October	5185	6076	**6329**	6302	6522
November	5192	6181	**6439**	6393	6626
December	6512	6277	6566	**6515**	6800

Table 4.10: Results (in TEUs per time step) of imports and exports predictions for all the methods (2016). *Source: Self made.*

Month	Predicted Normal	Predicted GP&t-Loc	Predicted 4 states Normal	Predicted 4 states GP&t-Loc
January	1344	1320	1324	1298
February	1872	1902	1760	1698
March	2330	2326	2164	2112
April	2759	2805	2552	2490
May	3202	3240	2859	2849
June	3557	3686	3281	3213
July	3938	4061	3672	3569
August	4261	4616	3989	3985
September	4588	5119	4284	4365
October	5004	5440	4597	4836
November	5449	5818	4939	5286
December	5884	6290	5293	5624

Table 4.11: Results for confidence areas of all methods for imports and exports (2016). *Source: Self made.*

Error	Predicted Normal	Predicted GP&t-Loc	Predicted 4 stats Normal	Predicted 4 stats GP&t-Loc
MAE	1364.00	1434.92	1401.17	1509.42
MAPE	144.07	148.14	147.59	151.54
RMSE	1899.43	1967.39	1952.96	2022.69

Table 4.12: Error comparison between all methods for imports and exports (2016). *Source: Self made.*

For this data set, the best distribution fits were found to be Generalized Pareto (for positive growths) and t-Location (for negative growths).

As expected, the prediction is working way better with these data sets (only import+export container traffic) and it can be seen just having a look at the errors. The confidence areas are behaving well, with acceptable values, only miss-predicting the 2016 June collapse.

The normal prediction, but for two-states Markov chain, is the best prediction in terms of errors this time. Regarding exact prediction (means), the generalized Pareto plus t-Location with two states chain is giving the most precise results, but checking confidence areas, the normal prediction has a narrower one and hence a more useful one.

4.4.1 2003-2007 data calibration

Same analysis is carried out for the 2007 monthly prediction but with only imports and exports traffics. With this data set, the best distribution for the positive growths is a Log Logistic distribution and for the negative ones the Generalized Extreme Value one.

Month	Real	Predicted Normal	Predicted LogL&GeV	Predicted 4 states Normal	Predicted 4 states LogL&GeV
January	85803	**64363**	63358	61457	59247
February	65202	66074	63624	66546	**64457**
March	72754	**68086**	64548	67013	62555
April	69644	**70280**	64994	71231	65886
May	71632	72485	65007	**71533**	63720
June	69041	74729	65995	75980	**67637**
July	68936	77230	**67466**	76221	66831
August	70595	79429	**71858**	80668	72826
September	72152	81200	73760	80731	**71209**
October	76687	83543	74408	85185	**75663**
November	66646	86464	75472	85513	**74682**
December	68175	89140	**77441**	90501	79017

Table 4.13: Results (in TEUs per time step) of imports and exports predictions for all the methods (2007). *Source: Self made.*

Month	Predicted Normal	Predicted LogL&GeV	Predicted 4 states Normal	Predicted 4 states LogL&GeV
January	16615	49428	13976	57516
February	24767	89164	24462	71495
March	31846	106651	29436	71061
April	38297	123787	37456	98059
May	45353	136047	42240	88524
June	51393	154356	51272	110391
July	57644	180461	56521	120836
August	64010	371162	65568	240364
September	71028	474484	68631	223790
October	80307	467577	76348	250830
November	91351	477867	82455	249459
December	100180	504584	91397	276782

Table 4.14: Results for confidence areas of all methods for imports and exports (2007). *Source: Self made.*

Error	Predicted Normal	Predicted LogL&GeV	Predicted 4 stats Normal	Predicted 4 stats LogL&GeV
MAE	8997.67	5938.50	9640.33	6312.92
MAPE	12.47	8.01	13.30	8.45
RMSE	11632.19	8284.98	12358.99	9495.45

Table 4.15: Error comparison between all methods for imports and exports (2007). RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage).*Source: Self made.*

This time the best error performing prediction is the one with two chain states and the log logistic plus generalized extreme value prediction. The comparison between these results and the 2016 data set results is very similar to the previous comparison with transhipment traffic included: this prediction is worse since the data is more unstable.

Regarding to confidence areas, again the only useful ones are the ones produced by the normal distribution, and the two states normal method seems to be a bit better than the four states one.

4.5 Final forecast approach

The objective of this section is to sum up all the calibration results obtained through the previous ones, and with them be able to take a decision on which method can fit the yearly data sets (transhipment included and without it). Once the decision is taken, provide with the two final predictions for the Piraeus port case as the final result of this study.

All the previous results have been condensed in the form of the three errors that have been computed for each method. Regarding *Peng (2009)* and other forecasting articles, the errors are the decision maker over which method is best for each case. Despite that, other elements can obviously taken into account since the performance of confidence areas.

	Method	MAE	MAPE	RMSE
2016 T+I/E	**Normal**	5208.50	129.98	8114.31
	GP&W	5255.33	130.45	8121.63
	4 states Normal	5010.00	128.19	7988.27
	4 states GP&W	5080.00	129.01	8006.21
2016 I/E	**Normal**	**1364.00**	144.07	**1899.43**
	GP&t-Loc	1434.92	148.14	1967.39
	4 states Normal	1401.17	147.59	1952.96
	4 states GP&t-Loc	1509.42	151.54	2022.69
2007 T+I/E	**Normal**	14107.58	11.63	17001.47
	GeV	9815.33	8.45	12343.08
	4 states Normal	10943.92	9.15	13824.62
	4 states GeV	8158.33	**7.35**	9906.94
2007 I/E	**Normal**	8997.67	12.47	11632.19
	LogL&GeV	5938.50	8.01	8284.98
	4 states Normal	9640.33	13.30	12358.99
	4 states LogL&GeV	6312.92	8.45	9495.45

Table 4.16: Comparison between all methods tested performances based in errors for Transhipment (T), Import (I) and Export (E) in different periods. RMSE (in TEU^2), MAE (in TEU) and MAPE (percentage). *Source: Self made.*

From here, the results for all methods for yearly data sets (2002-2016 with and without transhipments) are summarized in Tables 4.17 and 4.18. Having a look at them and taking into account Table 4.16, the decision can be made. Analysis are also a 10000 simulation steps.

Mean predicted (TEUs)	**2017**	**2018**	**2019**	**2020**
Predicted Normal	251486	231172	213319	198198
Predicted GeV&IG	7.67e+31	3.05e+39	2.93e+39	3.11e+39
Predicted Normal 4 states	318019	298000	237897	232144
Predicted GeV&IG 4 states	6.86e+32	3.24e+41	3.11e+41	2.72e+41

Standard deviations	**2017**	**2018**	**2019**	**2020**
Predicted Normal	79605	102685	117964	129856
Predicted GeV&IG	7.57e+33	3.05e+41	2.93e+41	3.10e+41
Predicted Normal 4 states	58332	110652	115865	125108
Predicted GeV&IG 4 states	5.66e+34	3.24e+43	3.11e+43	2.64e+43

Table 4.17: Results for the four year prediction (including transhipments) of all methods. *Source: Self made.*

Mean predicted (TEUs)	**2017**	**2018**	**2019**	**2020**
Predicted Normal	37746	35066	31207	28449
Predicted GP&Beta	38496	36472	33165	30764
Predicted Normal 4 states	39085	36564	32645	29704
Predicted GP&Beta 4 states	39897	38045	34572	32240

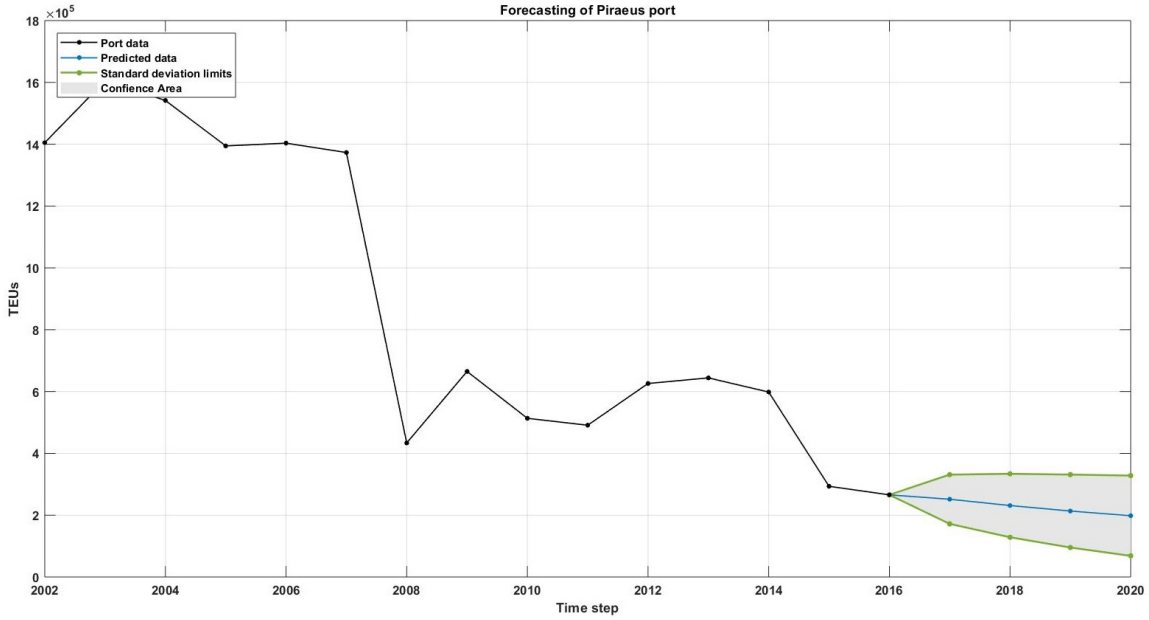
Standard deviations	**2017**	**2018**	**2019**	**2020**
Predicted Normal	14960	18399	20322	21926
Predicted GP&Beta	14517	18127	20485	22195
Predicted Normal 4 states	15427	17252	20127	20645
Predicted GP&Beta 4 states	14862	16772	19717	20237

Table 4.18: Results for the four year prediction (without transhipments) of all methods. *Source: Self made.*

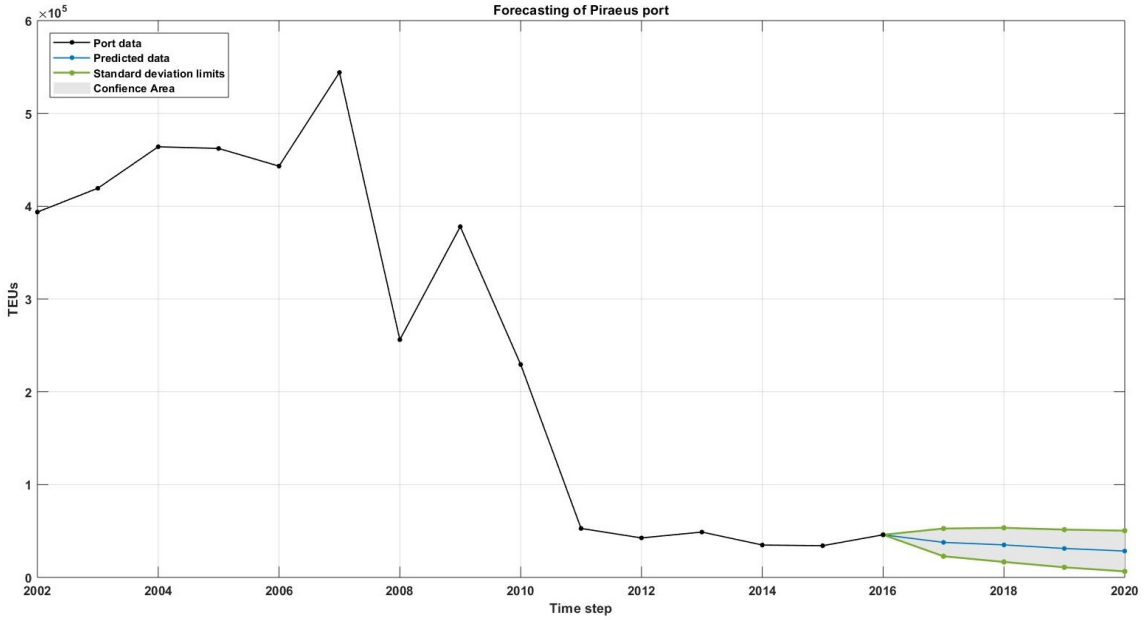
With all the information, seems that the clear option is to use a normal distribution (for its performance in error terms and confidence areas), with a very similar yield between a two states and a four states Markov chain.

Since data is made out of yearly information and it only consists in 15 real values, **the choice in this case is the prediction with a two states Markov chain and normal distribution**. The reason to pick this between the two and four states chain is that the four states chain reduces the growth probability information to a half of the two states chain, and it is better to exploit the few data that is available this time.

Final predictions for Transhipment+Import+Export and Import+Export traffic up to four years from last available data are plotted in Figures 4.7a and 4.7b.



(a) Prediction for T+I/E data for the next four years.



(b) Prediction for I/E data for the next four years.

Figure 4.7: Four years prediction for the Piraeus port. *Source: Self made.*

Chapter 5

Conclusions and future work

The objectives of the study have been accomplished in order to define the most suitable predictive model for the Piraeus port case. The model, a time-series analysis based one, has been successfully adapted to produce results supported on the combination of Markov chains and Monte Carlo experiment. The decision on which method to use falls into the comparison between techniques briefly described in the Chapter 2 which indicates that the time-series is the best analysis for this data.

Data provided by the port shows, as expected, that the transhipment traffic induces a volatile behavior over the total traffic which is translated in an increase of the overall error computation, since volatility worsens the performance of a time-series based model.

Results given can be considered useful as a help on decision making for the Piraeus case. This can be confirmed by two factors. One fact is that the model validation based on the errors computation is proving that it is working properly according to other studies validations (Peng et al, 2009; Grifoll, 2018) values. For example, Peng et al compare different forecasting methods and the best fit is giving a 5442.80 TEU MAE 3.09% MAPE and 6029.89 TEU^2 RMSE, which are higher than some of this study predictions. The other fact is that the confidence areas in most of the calibrations are good enough (narrow enough and with the real solution inside it) to be considered for decision making.

Time-series forecast models rely directly on the past data, and from two main difficulties have arisen. In one hand the lack of a broader yearly data set forced the method to be calibrated under monthly data, which is a solution that worked but not the desired one since monthly data induces a lot more instability, seasonality and irregularities than the yearly data, which is a decisive factor for a method such as time-series that increases drastically its performance without these difficulties. In this particular model, the Markov chains probabilities are more accurate as the data available gets larger, and Monte Carlo experiment distributions are more reliable as well. On the other hand, the irregularities produced even in yearly data due to the global crisis in 2008, and even the lack of monthly data for that year are considerably damaging the efficiency of the model.

Despite that, the results were pretty acceptable and other conclusions can be highlighted from the calibration tests. First, the increase in Markov chain states seemed to be a good improve for the model, this can be due to the increase of the transition matrix size and the consequent larger distribution for probabilities (more cases and its probabilities induced more accuracy). Nonetheless it is required a broad data set in order to reach a decent level of precision in each case probabilities, and for this reason it was decided not to be used for the final yearly prediction which relies on a shorter data set than the monthly one. Second, the chose of distributions for the Monte Carlo experiment showed that the normal distribution had a better performance overall. The need of splitting the data into negative and positive values in order to test distribution fittings reduced the data pool, and with it the distribution fits loses reliability, which sometimes lead to nonsense results. The normal distribution in this case showed to be more regular and stable in model results performance. Despite this, this is an element that depends entirely on the data, so it needs to be decided and tested for each case.

In terms of simulations, the increase of it to produce a better mean and confidence areas (standard deviation) show a convergence after a determined number. This is a requirement in order to qualify this method as a working one, and also it useful to

know the number of simulations needed to reach stability and to invest the minimum computational cost.

It has also been checked that the increase in steps to be predicted is reducing the reliability of the solution. This refers to the quick growth of the confidence area when the time interval for the prediction is increased. It is another decision to make where the frontier between useful and useless predictions lays, for example when the sum of the mean prediction and the standard deviation is bigger than a determined percentage of the mean itself, and this should be based on experience.

The utility of this time-series based method, allows the user to obtain more valuable results beyond the prediction itself, and those are probability distributions for the predicted times, histograms and other interesting statistical profitable benefits that have not been analyzed for the purposes of this study.

The final conclusion is that the normal distribution combined with a two states Markov chain for smaller data amounts and the normal distribution with a four states Markov chain for larger data sets are the best forecast models produced in the study in terms of error performance (all results shown at Table 4.16. For the final prediction (yearly and therefore small data set) a normal distribution two states chain has been used.

This conclusion is supported by Table 4.16 which summarizes the error analysis for all the tested methods (for import and export, and for transhipment also, for both predictions 2007 and 2016).

In the line of future research for this predictive model, there are points that could be improved. First of all, it would be very profitable to study if there is any relation between the optimum data amount (or minimum) and the performance of Markov chains (probabilities reliability) and the adaptation of Monte Carlo experiment distributions. Specially for the probabilities, it would be interesting a convergence analysis regarding the increase of data amounts.

Finally it would be a promising improvement to combine this method with other variables. Two of them can be highlighted: the use of the GDP method in order to

implement the economy effect over the traffic (which is very determinant) and the use of a decomposition technique of the data in different curves (the trend and the irregularities, seasonality and cyclical components) which is a common used technique in the time-series methods.

Bibliography

[1] George EP Box et al. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[2] PORTOPIA Collaborative project. "Port Traffic Forecasting Tool". In: *Towards a competitive and resource efficient port transport system* (2015).

[3] *Container port traffic (GDP growth, annual percentage)*. URL: `https://data.worldbank.org/indicator/IS.SHP.GOOD.TU?end=2016&start=2000&view=chart`.

[4] Alexandru Cotorcea, Filip Nistor, and Catalin Popa. "KEY TRENDS IN THE GLOBAL PORT DUE TO TRAFFIC VOLUMES". In: *Scientific Bulletin" Mircea cel Batran" Naval Academy* 20.1 (2017), p. 136.

[5] Persi Diaconis. "The markov chain monte carlo revolution". In: *Bulletin of the American Mathematical Society* 46.2 (2009), pp. 179–205.

[6] Manel Grifoll. "A statistical forecasting model applied to container throughput in a multi-port gateway system: the Barcelona-Tarragona-Valencia case". In: *Int J. Shipping and Transport Logistics* (2018).

[7] Alen Jugović, Svjetlana Hess, and Tanja Poletan Jugović. "Traffic demand forecasting for port services". In: *Promet-Traffic&Transportation* 23.1 (2011), pp. 59–69.

[8] William HK Lam et al. "Forecasts and reliability analysis of port cargo throughput in Hong Kong". In: *Journal of urban Planning and Development* 130.3 (2004), pp. 133–144.

[9] Olaf Merk. "The competitiveness of global port-cities: synthesis report". In: (2013).

[10] Pamela Paxton et al. "Monte Carlo experiments: Design and implementation". In: *Structural Equation Modeling* 8.2 (2001), pp. 287–312.

[11] Wen-Yi Peng and Ching-Wu Chu. "A comparison of univariate methods for forecasting container throughput volumes". In: *Mathematical and Computer Modelling* 50.7-8 (2009), pp. 1045–1057.

[12] *Piraeus Port Authority, S.A.* URL: `http://www.olp.gr/en/`.

[13] Olcay Polat and Hans-Otto Günther. "The impact of seasonal demand fluctuations on service network design of container feeder lines". In: *Journal of Transportation and Logistics* 1.1 (2016).

[14] Harilaos N Psaraftis and Athanasios A Pallis. "Concession of the Piraeus container terminal: turbulent times and the quest for competitiveness". In: *Maritime Policy & Management* 39.1 (2012), pp. 27–43.

[15] Athena Roumboutsos et al. "COST Action TU1001 Public–Private Partnerships in Transport: Trends and Theory (P3T3)". In: *COST Action TU1001: 2013 Discussion Papers: Part II Case Studies* (2013).

[16] RREEF. *Global ports: Trends and opportunities*. 2009.

[17] ANDERS TOLVER. "AN INTRODUCTION TO MARKOV CHAINS". In: (Unknown).

[18] ELISABETH Woschnagg and J Cipan. "Evaluating forecast accuracy". In: *University of Vienna, Department of Economics* (2004).