Master in Artificial Intelligence (UPC-URV-UB)

Master of Science Thesis

LIQUIDITY RISK MODELING USING ARTIFICIAL NEURAL NETWORK

Jordi Petchamé Sala

Advisor/s: Salvador Torra Porras
Advisor/s: Lluís Belanche Muñoz
Advisor/s: Maite López-Sánchez

17/01/2011

ABSTRACT

A new element of risk, liquidity risk, have flourished along this time taking importance and playing a key role in risk management tools. This has attracted the attention of the scientific community and economic and financial experts.

This thesis provides a theoretical introduction and a state of the art survey of the key elements needed to understand the complexity of the dealt issue. So it provides an study over liquidy risk and its application in market risk being included in market risk measures such as value at risk. Also an study over the behaviour of time series and it explores a relatively new alternative approach to model the liquidity risk using artificial neural networks, mainly approached in focused delay and recurrent neural networks due to their capability to work with time series .

In addition in this work have been designed and developed a methodology for the purpose of improving the way to treat time series and as resulting a simple graphical user interface with the intention of make easy the prediction.

This work has been developed on the framework Matlab Student Version version R2010a including Neural Network Toolbox 6.0.4. over a laptop with Windows Vista 32 bits, CPU: Intel(R) Core(TM)2 Duo CPU 2.20GHz and RAM: 2038 MB.

ACKNOWLEDGEMENTS

I would like to thank my supervisors Dr. Salvador Torra Porras, Dr. Lluis Belanche and Dr. Maite López-Sánchez, for their support and dedication for reviewing my master thesis. Their unvaluable advices have meant a lot to me. Particularly to Dr.Salvador for introducing to me the world of risk analysis and to Dr. Lluis Belanche for showing to me excellent ideas.

In addition, I would like to thank to Guillerm Alfaro managing partner of www.solventis.es who provide me with dataset of equity market to be studied.

I would also like to thank my family, girlfriend and close friends for their support and understanding during all the time.

TABLE OF CONTENTS

Introduction

Usually, the discussion about liquidity risk flourishes afterward stock market crash but it's a continuous problem of financial institution. Consequently asset liquidity risk is perhaps one of the prevalent treats facing today's capital markets and has played a key role in major crises during the last two decades.

Taking into account that market liquidity is an asset's ability to be sold without causing a significant movement in the price and with minimum loss of value[1], sometimes, in rare liquidity conditions, there are asset positions can often not be traded close to fair prices and many risk management systems can't predict it because they don't include market liquidity risk metrics yet.

Furthermore in today's days we are living a strangled period, the global financial crisis which began in mid-2007, where many banks are struggled to maintain adequate liquidity. Unprecedented levels of liquidity support were required from central banks in order to sustain the financial system and even with such extensive support a number of banks failed, were forced into mergers or required resolution. Thus, due to the magnitude and nature of financial risk, the Basel Committee on Banking Supervision[2] issued a package, called *Principles for Sound Liquidity Risk Management and Supervision*[3] in September 2008 belonging to the *Basel II Capital Accord*, of proposals to liquidity regulations with the goal of promoting a more resilient banking sector. These sound principles provide key elements of a robust framework for liquidity risk management at banking organisation and among them there is the use of liquidity risk management tools such as comprehensive cash flow forecasting, limits and liquidity scenario stress testing. So, it opens the need of develop new tools that manage the liquidity risk. But attempting to summarize and approaching it at the current work, one of key advancement of the *Basel II Capital Accord* is the introduction of Value at Risk (VaR, risk measure of the risk of loss on a specific portfolio of financial assets[4]) as an internal risk measure to consolidate an institution's market risk.

Looking into recent researches, I have found out that the conventional VaR approach to computing market risk of a portfolio does not explicitly consider liquidity risk, but it only assess the worst change in mark-to-market portfolio value over a given time horizon. Therefore this technique does not differentiate between market and liquidity risk and is well-know that to ignore the liquidity risk can result in significant underestimation of the VaR estimate[5].

The liquidity risk is assessed in concept of Bid-Ask spread or by adding an off-the-cuff liquidity risk multiplier to the overall market risk exposure. Thus in order to improve the VaR models including well estimate liquidity risk I've focused my work on trying to forecast the bid-ask spread (the cost to unwind the trading positions) under the assumption that it works as a time series.

---

[1] Website http://en.wikipedia.org/wiki/Market_liquidity
[2] Website http://www.bis.org/bcbs/index.htm
[3] Website http://www.bis.org/publ/bcbs144.htm
[4] Website http://en.wikipedia.org/wiki/Value_at_risk
[5] Integrating Liquidity Risk Factor into a Parametric Value-At-Risk Method. Al Janabi M.A.M (2008) pp. 76 - 87

On the other hand I have considered use artificial neural networks (ANN) to modeling the liquidity risk predicting the Bid-Ask spread so that the ANN are strongly used and many research papers justify that they are suitable in financial and economic time series ailments.

An Artificial Neural Network is an information processing paradigm which is inspired by the way biological nervous systems, such as the brain process information.

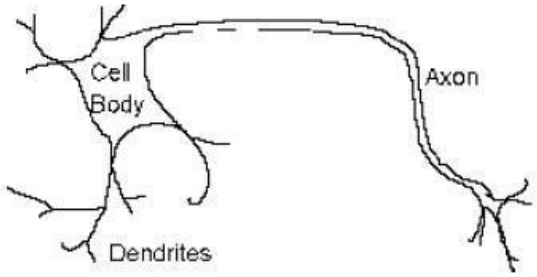


Figure 1: Biological Model [6]

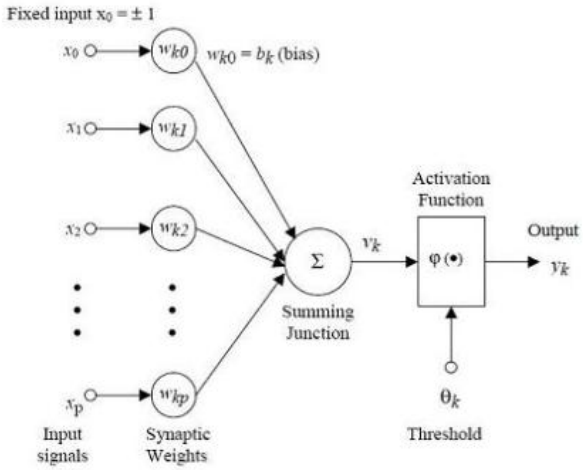


Figure 2: Mathematical Model [7]

The first attempts are concerned to McCulloch and Pitts (1943) works. They developed models of neural networks based on their understanding of neurology and they did simple models on logic functions, but were Rosenblatt (1958) who designed and developed the Perceptron. This system could learn to connect or associate a given input to a random output unit. Afterwards that increased up the number of research appearing enhancements models of networks and techniques like ADALINE (ADAptive LINear Element) which was developed by Widrow and Hoff (1960), Paul Werbos (1974) developed and used the back-propagation learning method, ART (Adaptive Resonance Theory) by Grossberg (1988), Self Organising Maps by Teuvo Kohonen (1989)...

As we can see the ANNs have an age of more 70 years old approximately and they have been rooted in many disciplines: neurosciences, mathematics, statistics, physics, computer science and engineering. Neural networks find applications in such diverse fields as modeling, time series analysis, pattern recognition, signal processing. This is consequently to the ANN capabilities provides like to learn and to generalize. Thus, the use of ANN offers useful properties: nonlinearity, Input-Output mapping, adaptivity and evidential response, very large scale integrated implementability (VLSI), uniformity of analysis and design, and neurobiological analogy.

[6] Biological Model, website http://www.learnartificialneuralnetworks.com/images/bneuron.jpg
[7] Mathematical Model, website http://www.learnartificialneuralnetworks.com/images/bneuronmodel.jpg

Furthermore according to the focus of this work the Artificial Neural Networks offer qualitative methods for business and economic systems that traditional quantitative tools in statistics and econometrics cannot quantify due to the complexity in translating the systems into precise mathematical functions[8]. Below I show the financial analysis task of which prototype neural network-based decision aids have been built[9]:

- Credit authorization screening
- Mortgage risk assessment
- Project management and bidding strategy
- Financial and economic forecasting
- Risk rating of exchange-traded, fixed income investments
- Detection of regularities in security price movements
- Prediction of default and bankruptcy

Although the conventional models are certainly suitable to some tasks involving financial forecasting which works in well-identified models, the ANNs are best applied to problem environments that are highly unstructured. As I said above the ANN are useful in time series forecasting, thus considering that the current trouble is approached to time series models, it reinforces my proposal of solving it using artificial neural networks.

In first place, I'm going to define a little bit the term time series as a collection of observations made sequentially through time[10] or as the evolution of a variable, phenomenon, over time and they can be economic, physical or social. An example is plotted below.

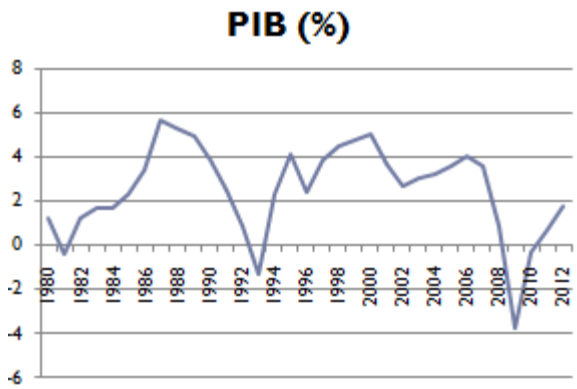


Figure 3: An example of time series: The GDP/PIB of Spain during 1980 until 2012[11]

The scope of its studies is the knowledge of its pattern's behaviour such that it let us make accurate predictions. Thus we could say that the main objectives of time-series analysis are to get a description of data usually using summary statistics and graphical models, to model it finding a suitable model which describes the data generating process (regards to ANNs techniques it is a big

[8] According to Zahedi (1993)
[9] Corresponding to Medsker (1996) list
[10] Time Series Forecasting. Chris Chatfield (2001). pp. 1
[11] The figure ilustrate the evolutio of the Spain GDP during the period of 1980-2010 more the prediction to 2012. The data has been found in econstat.com. (last visit on 3 October 2010).

disadvantage because they are considerate like black-boxes), do forecasting to estimate the future values of the series and take control which means to take actions of control from good forecasts. Of course, though the future value of a time series cannot be predicted with absolute accuracy, the result cannot be fully random such that the study takes profit.

Thereby must have some regularity in its pattern's behaviour which can be represented in models. If in the case that we could predict without any error we'd are talking of deterministic phenomenon but normally the time series have associated random events which are named stochastic phenomenon.

Nowadays there are huge amounts of literatures references to time series analysis and many researchers have come up with useful techniques based on classical model like ARIMA (autoregressive integrated moving average) or Box-Jenkins, ARMA (autoregressive moving average), GARCH (generalized autoregressive conditionally heteroscedastic model)... to forecast them but, however also there are alternative literature like Weigned and Gershenfeld (1994)[12], Stern(1996)[13], Warner and Misra(1996)[14] and Faraway and Chatfield (1998)[15] whose publications introduces ANNs in time series analysis and proof that it provides new features which are well-fitted to no-lineal and unstructured models.

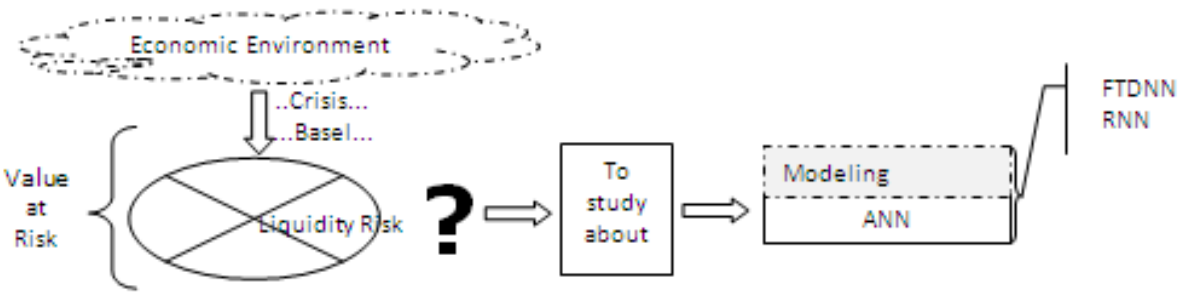
[12] Time Series Prediction. Weigned and Gershenfeld (1994).
[13] Neural Networks in applied statistics. Stern(1996).
[14] Understanding neural networks as statistical tools. Warner and Misra (1996).
[15] Time Series Forecasting with neural networks: A comparative study using the airline data. Faraway and Chatfield (1998).

Motivation

As I commented in the introduction, the banking organizations, afford to new regulations introduced by Basel II Capital Accord, have the need of using liquidity risk management tools in order to improve their analysis of the risk. So, one way is to add the liquidity risk into VaR measures so that they do not explicitly consider liquidity risk. Hence I have focused my work on attempting to forecast the Bid-Ask Spread which is considered a measure of liquidity risk.

In addition, I have used Artificial Neural Networks so that their use is strongly arising in financial and economic fields (and other approaches which are not of interesting in this work) even though they have to fight with classical statistical and econometrics models yet. Therefore, the main motivation of this work is to get optimal configuration of neuronal networks in order to proof its useful and helpful properties as forecaster (time series analysis) and consolidate them like suitable technique to solve real financial problems and underline their adaptable capabilities. For that, I've chosen a real and new ailment being this works one the first in apply NN in the liquidity risk problem.

On the other hand, is well-known that to configure the neural networks in order to reach an optimal model is bored, large and complex process. Hence I would like to attempt to introduce a simple methodology and a tool in order to satisfy the needs of time series modelling. Thus, in order to summarize and to put into context, the following figure illustrates where I come from and where I go.



From current economic environment (crisis) crop up the need of including a new element of risk such as liquidity risk into already value-at-risk measures in order to minimize portfolio's risk. Therefore, it forces to study this new component, its behaviour and to develop new techniques which let us to model and to forecast it with the purpose of minimizing its impact. So, one technique which is proposed by the market is the artificial neural networks.

Objectives

The neural networks have successfully been applied in different financial and economic fields and one of these have been to predict the behaviour of time series such as G. Peter Zhang and Douglas M. Kline(2007)[16], or Chan Man-Chung, Wong Chi-Cheong, Lam Chi-Chung[17] show. Thus, I propose to model liquidity risk using artificial neural networks such as focused delay and recurrent neural networks. Therefore the main goal will consist of predicting the variable Bid-Ask Spread at t+1 (prediction for tomorrow) of a set of assets (35 assets provided by www.solventis.es) of the Spanish equity market.

To reach it, the work has been split in two main parts:

1. From a raw dataset and a model provided by financial experts, to indentify what elements, features are important/significant to predict the bid-ask spread and what new elements can be introduced to improve its prediction.

2. To get an optimal model for each asset through a methodology developed by myself with automatizes the large steps to do in modeling time series.

As I have just commented, it also aims to make a contribution regards to time series modeling using neural networks. For that it will be developed a methodology able to adapat inputs in order to allow us to get the optimal model over any time series, of course mainly the bid-ask spread time series will be the object of this work. Therefore, it will be composed by different methods which will afford to reach relevant properties of time series such as trend and number of delays (autocorrealtion). Also, in order to improve the parameter's configuration of a network, it will be implementend few already proposed and well-known methods such as:

- golden section search in order to find optimal number of neuron for a single layer

- and a pruning algorithm to find what features improve the model.

In addition one purpose of this work is accomplish providing with a graphical user interface which allows someone to get prediction for new time series of some asset treated in this work. It will show prediction for *t+h* steps ahead where *h* will be fill in by the user.

[16] Quarterly time-series forecasting with neural networks. G. Peter Zhang and Douglas M. Kline(2007). Working paper.
[17] Financial time series forecasting by neural network using conjugate gradient learning algorithm and multiple linear regression weight initialization. Chan Man-Chung, Wong Chi-Cheong, Lam Chi-Chung. Working paper.

Related work

First of all I would like denote that this work have cropped up due to the need of the current economic and financial environment. Therefore it is not strange that I have not explicitly found any works related to the modeling of liquidity risk with neural networks. Otherwise, I have found few works talking about the need of include of liquidity risk in market risk measures (Value at Risk) and in other way some paper which apply neural networks to predict stock returns, volatility, volume traded.. which are strongly linked to liquidity risk and as the time series are treated to be used in neural networks.

Regarding to works of liquidity Risk, I have focused my efforts particulary in two recently jobs. The first has been was written by Cornelia Ernst, Sebasting Strange and Christoph Kaserer on February 3, 2009[18] which are assessed different liquidity risk measure. They implemented Bangia et al. (1999)[19], Giot and Gramming (2005)[20], Stange and Kaserer (2008) [21]and Ernst et al. (2008)[22] in a large sample of dily sotck data over 5.5 years and they used a standar Kupiec (1995)[23]-statisitic to determine if model provide precise risk forecasts on a statistically significant basis. The provided results show that available data is the main driver of the preciseness of risk forecast, so the models get accurate performance depending of the data.

Specially this work is interesting because it provides a valuable information of what, economic elements are related to liquidity risk. For instances Berkowitz (2000)[24] and Cosandey(2001)[25] developed related directly the volume or transaction with the liquitidy risk, or otherwise Francois-Heude and van Wynendaele (2001)[26] linked it with the order size by using limit order book data[27] or also find models in base of bid-ask spread as proposed by Bangia et al. (1999)[28] and Ernst et al (2008)[29].

The other one has been developed by Al Janabi, Mazin A. M on May 20, 2009 and they present a generalized theoretical modeling approach for trading and fund management portfolios in base of liquidity risk model. This reflect the importance of liquidity risk given the rising need for measuring, managing and controlling of financial risk, trading risk prediction under liquid and illiquid market condition and also as it can be included to value-at-risk measures. It provides a

[18] Measuring Market Liquidity Risk –Which model works best? Cornelia Ernst, Sebasting Strange and Christoph Kaserer(2009).
[19] Liquidity on the Outside. Risk. Bangia, A., F. X. Diebold, T. Suchermann and J.D Stroughair (1999), pp. 68 – 73.
[20] How large is liquidity risk in an automated auction market? Empirical economics. Giot, P. and J Gramming (2005), pp. 867 – 887.
[21] Why and how to integrate liquidity risk into VaR-framewok. Stange, S. and C. Kaserer (2008).
[22] Accounting for non-normality in liquidity risk. Ernst, C., S. Stange and C. Kaserer (2008).
[23] Techniques for verifying the accuracy of risk management models. Kupiec, P. (1995), pp. 73 – 84.
[24] Incorporating liquidity risk into value-at-risk models. Berkowitz (2000).
[25] Adjusting value at risk for market liquidity. Risk. Cosandey, D (2001), pp. 115 – 118.
[26] Integrating liquidity risk in a parametric intraday VaR framework. Francois-Heude, A. and P. Van Wynendaele (2001)
[27] They use the liquidity cost measure 'weighted spread', which calculates the liquidity costs compared with the fair price when liquidating a position quantity q against the limit order book.
[28] Liquidity on the outside. Riks. Bangia, A., F. X. Diebold, T. Schuermann and J.D. Stroughair (1999) pp 68 - 73 .
[29] Accounting for non-normality in liquidity risk. Ernst, C., S. Stange and C. Kaserer (2008).

methodology for evaluating trading risk whe the impact of illiquidity of specified financial products is significant.

Regarding to papers/works about neural networks used as preditor I have found many related works but however I have focused in which are strongly related with liquidity risk depedence. For instance, Mary Malliaris, Linda Salchenberger (1993)[30] show that the neural networks are more able than the classical techniques to forecast the S&P 100 using implied volatily. To choose the best set of features they is begin with a small number of variables and add new variables which improve network performance. Summarizing they included 13[31] variables where the remarkable thing is that delays were used only for volatility and neither detrend nor deseasonality the target. Their results were encouraging so that as they defend she neural network model, on the other hand, employs both short-term historical data and contemporaneous variables to forecast future implied volatility.

Paul R. Lajbcygier and Jerome t. Connor (1997)[32] concluded that somewhere between the hybrid neural network and the bootstrap predictor is best for this option pricing problem. Bootstrap methods for bias reduction was shown to give good results at the edge of input space where good extrapolation is critical.

On the other hand Cottrell, M. (1995)[33] while developing their methodology in base of combining statistical techniques of linear and nonlinear time series with the connectionist approach in order to simplify the architechture they concluded it was absolutely clear that such a multilayer perceptron cannot model a time series containing a trend. An other interesting paper which have substantained as treat of laggs of time series has been proposed by Huang Wei (2004) who provides a new methodology to seek optimal lag periods, which are more predictive and less correlated.

All of theses papers/works provide some relevants characteristics which have been consisdered along of this work.

[30] Using neural networks to forecast the S &P 100 implied volatility. Mary Malliaris, Linda Salchenberger (1996).
[31] Neurocomputing 10 (1996) . M. Malliaris, L. Salchenberger, pp 190.
[32] Improved option pricing using artificial neural networks and bootstrap methods. Paul R. Lajbcygier, Jerome t. Connor (1997).
[33] Cottrell, M., Girard, B., Girard,Y., Mangeas, M., Muller, C., 1995. Neural modeling for time series: a statistical stepwise method for weight elimination. IEEE Transactions on Neural Networks 6 (6), pp. 1355–1364.

Organization of this thesis

In this section I pretend to define the different sections of this work and as they are organized. First of all, it has the introduction where I explain and put to the reader into the context of the work and immediately come the motivation and objectives points so that it help to reach and to clear the main aims of this work.

Secondly, once detailed what about my pretensions I introduce to the reader the keys topics for reinforcing the understanding, so giving an overview of value at risk, liquidity risk, time series and artificial neural networks and as they are linked under the *State of Art* section. The *Value at risk* topic/subsection is introduced in an easy way and finalized with dump example in order to make simple the concept. I have not got into more detail because it is not the main of the work and is enough to know its principle concept and not extensions.

According with *Liquidity risk* subsection, is explained how it has got importance over time and I introduce two models where the liquidity risk is defined like the bid-ask spread and combined with VaR models to provide a more accurate evaluation of risk. Afterwards are defined the *Time series* and *Artificial neural networks* subsections. Again, as the value to predict is a time series, the *Time series* subsection pretends to introduce to the reader with an overview as they are treated and what important elements are contained in. In the last, and closing the key points is treated the *Artificial neural networks* topic where I have to denote that in this section have been emphasized the networks used or with pretension to use in this work and others have been left in second side.

In *Platform develop* section I explain in detail all the process that I followed to achieve the results detailed at the end of this work. It is divided by a main *Description* subsection where inform about the original dataset and as it composed. Afterward the *Pre-process* subsection comes *Exploring data* subsection which attempt to analyse the raw dataset looking for missing and outlier values, standardizing data and treat the trend and seasonality of time series. *Feature selection* subsection treat to find a common pattern in order to reduce or add some attributes.

One time is reached the goal of this section getting an efficient dataset the section proceed with the definition of the *Modeling* subsection, where the reader can find a global diagram of the methodology used in order to choose the optimal network and also the detail of its components. Afterward the results are studied in *Evaluating* section.

As this work does not only pretend to be an study of forecasting the liquidity risk but also make a real application, hence in *Practical application* section have been designed a prototype of a graphical user interface which try to be an easy interface and to remove all complexity so that anyone will be able use it. To finalize, this work ends with *Conclusion & Future works* section providing what about of the thesis and possible new lines of research over this theme.

State of the Art

VaR: Value at Risk

The value-at-risk (VaR) measures did not enter to financial field until early 20th century, starting with an informal capital test the New York Stock Exchange (NYSE). It developed a set of requirements that the firms ought to hold (5% of customer debits, 10% (min.) on proprietary holdings in government bonds, 30% on proprietary holdings in other liquid securities and 100% on proprietary holdings in all other securities) for setting capital requirements.

In 1975, the US Securities and Exchange Commission (SEC) established a Uniform Net Capital Rule (UNCR)[34] for US broker-dealers trading non-exempt securities. Thus, financial assets were split into 12 categories such as government debt, corporate debt, convertible securities, and preferred stock and additional haircuts were applied like any concentred position in a single asset. Although crude, the SEC's system of haircuts was a VaR measure. Later, additional regulatory VaR measures were implemented for banks or securities firms.

Also VaR measures were influenced by portfolio theory, Markowitz (1952) and others independently published VaR measures to support portfolio optimization. For instances Lietaer (1971) describes a practical VaR measure for foreign exchange risk. So, over time VaR measures were increased quickly.

In 1990 appeared a new term risk management as the use of derivatives to hedge or customize market-risk exposures. The new risk management tends to view derivatives as a problem as much as a solution. It focuses on reporting, oversight, and segregation of duties within organizations. In July 1993 was published a report, named Group of thirty 30 report, which describes then derivatives use by dealers and end-users. The heart of the study is a set of 20 recommendations, a formal model (appear the concept of Value at Risk) for the evaluation of the such market risks for portfolios and trading desks over short periods of several trading days, to help dealers and end-users manage their derivatives activities, but though the report was focused on derivatives, most of its recommendations are applicable to the risks associated with other traded instruments.

The concept of VaR was taken up by financial regulator in the 1996 Basel Accord supplement and subsequently was extended to measuring credit risks over much longer horizons. It has gone arising until to become a major player in the enterprise-wide risk management solutions appropriate to the world's financial institutions at all levels. Business activities entail a variety of risks which are distinguished between different categories of risks: market risk, credit risk, liquidity risk... although they have a strong correlation.

[34] Annual report of the SEC for the fiscal year ended June 30, 1975, pp 17.

- As market risk is defined as exposure to the uncertain market value of a portfolio35, such as we know the market value today but were are uncertain as to its market value a time ahead from today. Daily profits and losses on the contract reflect market risk.

- Credit risk is defined as an investor's risk of loss arising from a borrower who does not make payments as promised[36].

- Liquidity risk as the risk of being unable to transact a desired volume of contracts at the current market price[37].

Value-at-Risk is a category of market risk measures which let assess the potential loss of a trading or investment portfolio over some period of time. So summarizing, the notion is that losses greater than the value-at-risk are suffered only with a specified small probability. In particular, associated with each VaR measure are a probability α, or a confidence level $1-\alpha$, and a holding period, or time horizon, h. The $1-\alpha$ confidence value-at-risk is simply the loss that will be exceeded with a probability of only α percent over a holding period of length h, equivalently, the loss will be less than the VaR with probability $1-\alpha$. For instances, if h is one day, the confidence level is 95% so that α=0.05 or 5% and the value-at-risk is one million euro, then over a one day holding period the loss on the portfolio will exceed one million euro with a probability of only 5%.

Thus, value-at-risk is a particular way of summarizing and describing the magnitude of the likely losses on a portfolio. This makes it useful for measuring and comparing the market risks of different portfolios, for comparing the risk of the same portfolio at different times... But VaR's simple, summary nature is also its most important limitation. Clearly information is lost when an entire portfolio is boiled down to a single number, its value-at-risk. This limitation has led to the development of methodologies for decomposing value-at-risk to determine the contributions of the various asset classes, portfolios, and securities to the value-at-risk. In addition it helps to overcome the problems in measuring and communicating risk information. Hence, due to its usefulness and applicability, there are many variations of VaR measures. The standard value-at-risk measure would be:

VaR= (payoff in portfolio value) – k·(volatility's portfolio), where k is set depending of confidence level.

- 1-α=84% k=1
- 1-α=95% k=1.645
- 1-α=97.5% k=2

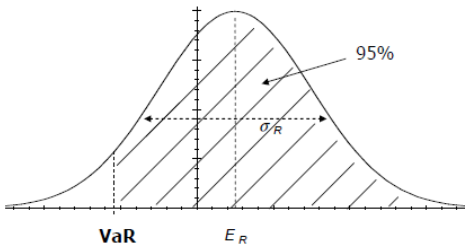


Figure 4: Normal distribution with VaR value in base of its return

[35] Value-at-Risk: theory and practice. pp. 22.
[36] Website http://en.wikipedia.org/wiki/Credit_risk
[37] Risk Management: value at risk and beyond. pp. 13

An easy example would be: X€ has been invested in an asset whose expected return is 15% with a volatility of 10%. VaR with confidence-level of 95%:

$$VaR = 15\%\text{-}1.645 \cdot 10 = \text{-}1.45.$$

Its interpretation is with a 95 % of confidence level the potential loss is 1.45%, in other words with a probability of 95%, the investor will not lose more than 1.45%.
But if we would have an upper expected return like *E*=20%, the investor wouldn't have negative return, so the VaR would be 0.

$$VaR = 20\%\text{-}1.645 \cdot 10 = +3.45 => VaR = 0.$$

Liquidity risk

Indeed the traditional VaR models is based in that the portfolio is stationary over the liquidation horizon and that the market is enoughly liquid to the market price is attainable by transactions prices. This marking-to-market approach is adequate to quantify and control sirk for an ongoing trading portfolio but may be more questionable if VaR is supposed to represent the worst loss over the unwinding period.

As I've explained above, in the introduction, market liquidity refers to the ability to undertake financial securities transactions in such a way as to adjust trading portfolio and risk exposure profiles without significantly disturbing prevailing market condition and underlying prices. In other words, is the risk that the liquidation value of trading assets may differ significantly form their current mark-to-market values and hence it is a function of the size of the trading positions, i.e. adverse price impac risk, and the price impact of trades, i.e. transactions cost liquidity risk[38]. Where the firs component, price impact , arises when the trader will sell out immediately by giving discount that brings the price down (the trader sells immediately at an unusually low price or buys at high price in case he had short sold assets) or however when the trader is unable to quickly sell a security at a fair price due to few people trade the given security, the second component, transactions cost liquidity risk, will be influenced.

Thus, it depends on the existence of enough number of counterparties and their readiness to trade. Therefore, it sort of risk is strongly correlated with market risk, commented in *VaR:Value at Risk* section, which is increased under critical economical conditions due to its major variation (may mesured like volatility). But also we have to realize that there are markets illiquidity by themselves like can be corporative bonds.

Thereby, an unprivileged liquidity conditions indicates a relatively small number and size of daily transactions and gives an indication of the size of a portfolio trading position than the market

[38] Asset Market Liquidity Risk Management: A Generalized Theretical Modeling Approach for Trading and Fund Management Portfolios. Al Janabi, Mazin A.M (2009). pp12.

can absorb a given level. To get efficient models of liquidity risk is convenient and useful in order to optimization' portfolio. Nowadays there are many models of liquidity risk which can be divided in two broad categories: traceable and theoretical.

A large literature has developed theoretical modeling approaches, these include more recently to Subramanian and Jarrow (2001)[39], Almegren (2003)[40], Dubil (2003)[41] and Engle and Ferstenberg (2007)[42]. These models generally use optimal trading strategies to minimize the Value-at-Risk of a position including liquidity. However, empirical estimation techniques for the large range of parameters of these models still need to be developed[43].

Regarding to empirically traceable works we find to Bangia et aL (1999)[44], Cosandey (2001)[45], Stange and Kaserer and Ernst et aL (2008)[46]. All proposed models can be grouped by the kind of required data for their estimation: bid-ask-spread models, transaction or volume data models, and models requiring limit order book data.

As the main goal of this work is the forecast of the liquidity risk in base of the bid ask spread, I'm going to explain some models focused in bid-ask spread in order to understand how ithe liquidity risk is combined with VaR, also defined as L-VaR measure (value at risk of asset i under illiquid market conditions). But previously I should define Bid and Ask: Bid price is the price that somebody will pay for a stock at a given moment, while Ask price is the price at which someone is willing to sell a stock. Hence, the impact will be higher as the spread is higher and viceversa. For instances, if a trader trade 125 time along a year and the mid bid-ask spread is 1/8 (normal conditions), hence his cost of trade will add up to 125·1/8 = 15,625 and supposing that he trades with a capital of 50 per year. It means that the cost of winding will be 15,625/50 = 31,250% over his capital each year. In other words, If the trader want to make a profit, he will make it only after overcoming this 31.25% handicap. So that bid-ask spread has to be considered as you are trading.

Bangia, Diebold, Schuermann and Stroughair (1999)[47] developed a simple liquidity adjustment of a VaR-measure based on bid-ask-spread.

$$L\text{-}Var = 1 - exp(z\partial_r) + (\mu_s + \hat{z}_s \sigma_s)$$

where ∂_r is the variance of the continuous mid-price return over the appropriate horizon and μ_s and σ_s are the mean and variance of the bid-ask spread, z is the percentile of the normal

[39] The liquidity Discount, Mathematical Finance. pp: 447-474.
[40] Optimal execution with nonlinear impact functions and trading-enhanced risk. Applied Mathematical finance, pp 11.
[41] How to include liquidity in a Market VaR Statistic. Journal of Applied Finance, pp 19-28.
[42] Execution Risk: It's the Sme as Investment Risk. Journal of Portfolio Management, pp 34-44.
[43] Cornelia Ernest, Sebastian Stange, Christoph Kaserer. Measuring Market Liquidity Risk. Which model Works best?, pp 2.
[44] Liquidity on the outside. Risk, pp 68-73.
[45] Adjusting Value at Risk for Market Liquidity. Risk, pp,115-118.
[46] Accounting for Non-normality in Liquidity Risk. CEFS working paper 2008 No 14. Available at http://ssrn.com/abstract=1316769
[47] Measuring market liquidity Risk: Which model works best?. pp 3.

distribution for the given confidence and $\widehat{z_s}$ is the empirical percentile of the spread distribution in order to account for non-normality in spreads.

Another model is proposed by Ernst, Stange and Kaserer[48] who suggest a different way to account for future time variation of prices and spreads. As it is more complex model and it is not the aim of this work, I am not going in depth.

Time Series

As I have commented above, the variation of a variable over time is called time series (a collection of random variables indexed according to the order they are obtained in time[49]). If we know its past behaviour (historical data) we could predict its possible future behaviours. Time series forecasting is a form of extrapolation in that it involves fitting a model to a set of data and them using that model outside the range of data to which it has been fitted[50] . But, it's not obviously as easy as seems such that we have to consider different assumptions like the dangers of extrapolation.

Therefore, we consider a time series such as $x_1, x_2, \ldots, x_n$ and the forecast future value such as x_n+h where h is called the forecasting horizon or lead time. For instance, if $h=1$, the forecast future value will be $n+h=n+1$ where n is the current instant. So it means we're forecasting for tomorrow. In general, a collection of random variables $\{x_t\}$, indexed by t is referred as a stochastic process. Thus, a time series is a realization or sample function from a certain stochastic process.

Notice that we are talking about forecasting, hence, we have to reference to forecasting methods which are broadly classified into three types:

- Judgemental methods regards to subjective intuition.
- Univariate methods regards to single series. The prediction is based on one time series.
- Multivariate methods regards to a predictable time series dependency. There're one or more time series variables called predictor or explanatory variables which give relevant information about the predictable variable.

Usually forecasting methods combine more than one, for instance, when univariate or multivariate methods are adjusted subjectively to take account of external information. In many cases is normally really difficult to distinct both approach and their combination get models better. For example, in particular, many macroeconomics forecasts are obtained by making adjustments to model-based forecasts, but, however it's not always clear how such adjustments are made. So the combination of judgemental and statistical approach can improve the accuracy.

[48] Accounting for Non-normality in Liquidity Risk. Ernst, C., S. Stange and C. Kaserer (2008), working paper.
[49] Time Series Analysis and its Applications. Robert H.Sumway & David S.Stoffer. pp. 9.
[50] Time Series Forecasting. Chris Chatfield (2001). pp. 7.

Coming back to the danger of extrapolation issue, we have to realize that forecasts are generally conditional statements of the form such "if this behaviour continues in the future, then..." which means that additional information in the future can produce a range of different forecasts. Even though there're lot of examples which provide forecasts of years, we never can neglect that all forecasts are led by presumptions and a sudden change in the data can produce unexpected future values, so they cannot be predicted exactly but they do contain structure which can be exploited to make better forecasts.

Basics of Time Series

A common assumption in many time series techniques is that the data are stationary. A stationary process has the property that the mean, variance and autocorrelation structure do not change over time. Stationarity can be defined in precise mathematical terms, but for our purpose we mean a flat looking series, without trend, constant variance over time, a constant autocorrelation structure over time and no periodic fluctuations. Usually, in the classical methods there are two methodologies to study the behaviour of time series: modeling by components or Box-Jenkins approaches.

<u>Modeling by component</u> consist in attempt to identify four components, they are time functions, which are not necessary exists together:

- (T)Trend, defined as the long-term change in the underlying mean level per unit time. Usually it is expressed as time function of polynomial or logarithmic type: $T=\alpha_0+\alpha_1 t+\alpha_2 t^2+\ldots$

  If the data contain a trend, we can fit some type of curve to the data and then model the residuals from that fit. Since the purpose of the fit is to simply remove long term trend, a simple fit, such as a straight line, is typically used.

  For non-constant variance, taking the logarithm or square root of the series may stabilize the variance. For negative data, you can add a suitable constant to make all the data positive before applying the transformation. This constant can then be subtracted from the model to obtain predicted (i.e., the fitted) values and forecasts for future points.

- (S)Seasonal variation is defined as the oscillation which are given or repeated in short time periods. They can be associated to dynamic factors like hotel occupancy, the sale of clothings…

  It is an important problem for forecasters. There are numerous models and many different ways to analyze and forecast seasonal time series: a easy ways would be to use for instances, the run sequence plot which is a recommended first step for analyzing any

time series. Also, although seasonality can sometimes be indicated with this plot, seasonality is shown more clearly by the seasonal subseries plot or the box plot. The seasonal subseries plot does an excellent job of showing both the seasonal differences (between group patterns) and also the within-group patterns. The box plot shows the seasonal difference (between group patterns) quite well, but it does not show within group patterns. However, for large data sets, the box plot is usually easier to read than the seasonal subseries plot. Both the seasonal subseries plot and the box plot assume that the seasonal periods are known. In most cases, the analyst will in fact know this. For example, for monthly data, the period is 12 since there are 12 months in a year. However, if the period is not known, the autocorrelation plot can help.

But all of this models are well-suitable for specifics times series such as many literature apoint that, but unfortunately no single model or modeling approach is best for all seasonal time series under different conditions as suggested by a large number of theoretical and empirical studies.

- (C)Cyclic variation is given in long time periods and normally does references to economic states. As a rule they tend to be more difficult to indentify so its period is longer becau may there are enough recollected data to identify it.

- (R)Irregular fluctuation, residue, is the random component belonging to noise.

The follow figure illustres all of this components:

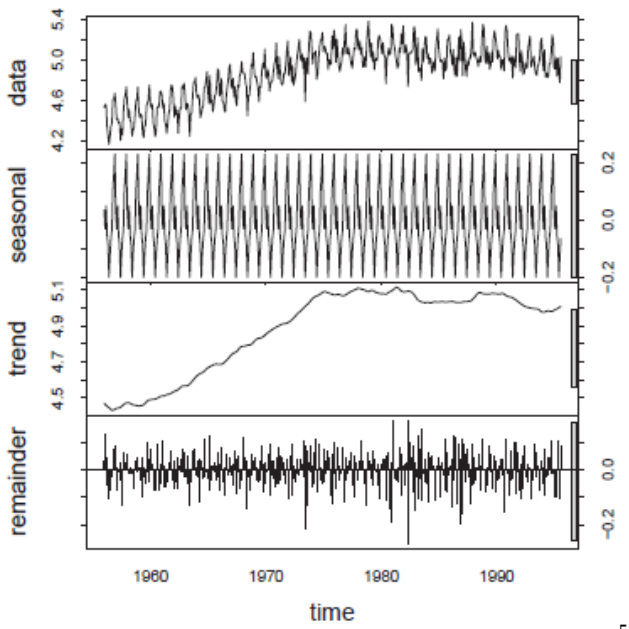


Figure 5: Performing the seasonal decomposition for monthly beer [51]

Thus, to assess the different components are used some statistical techniques such as moving average which are the most popular of them. Consequently, once identified all

[51] The components have been obtained with the command stl of cran-R using the datafile http://134.76.173.220/beer.zip

components and assuming that the irregular fluctuation is additive we have two possible models called additive and multiplicative:

- Additive Model: $Y= T + S + C + R$
- Multiplicative Model: $Y=T \times S \times C + R$

Summarizing this model we can consider if seasonal pattern is constat along time it will be an additive model, but if this going amplifying while time, it will be a multiplicative model. In addition this classification is quite restrictive so that we can find mixed models.

Regarding to Box-Jenkins approaches, is based on determining what is the probability model governing thephenomenon' behavior along the time. In other words, assuming that not ever we will be able to indentify the different components of serie, it study the randomly component (residue).

The statistical methodology used on this study is divided in 3 steps: 1)model identification, 2)parameter estimation and 3) diagnosis model.
To achieve it we have several models like moving average (MA), autoregressive(AR), integrated(I) and their combinations (ARMA y ARIMA)[52].

So far we have seen how to deal univariate time series analysis which only takes in account on past values of the same variable but perhaps theses past values depend also of other variables. Nowadays we find many examples where often the value of one variable is not only related to its predecessors in time but, in addition, it depends on past values of other variables. For instance, dealing with economic variables, household consumption expenditure may depend on variables such as income, interest rates, and investment expenditures. So if all these variables are related to the consumption expenditures it makes sense to use their possible additional information content in forecasting consumption expenditures. A general form could be:

$y_{1,t+h}=f(y_{1,t},y_{2,t},\ldots,y_{k,t},y_{1,t-1},y_{2,t-1},\ldots,y_{k,t-1},y_{1,t-2},\ldots)$ where t is discrete time and h is the forecasting horizon.

This gives rise to a new set of techniques named vector time series models where we find the previous model adapted to multivariate factors: vector AR models, vector MA models, vector ARMA models, etc…

But I just wanted to put into context the reader as is treated time series from the classic prism, hence I will not go into depth in this theme because neither is the purpose of this work. Before to finalize this section, I would like to do the definition of autocorrelation that I commented above as a technique to now the period of seasonality.

[52] For detailed treatment of ARMA and ARIMA see Time Series Analysis and its applications. Robert H Shumway and David S. Stoffer(2001). pp 89.

The autocorrelation plots are a commonly-used tool for checking randomness in a data set. This randomness is ascertained by computing autocorrelations for data values at varying time lags. If random, such autocorrelations should be near zero for any and all time-lag separations. If non-random, then one or more of the autocorrelations will be significantly non-zero. In addition, autocorrelation plots are used in the model identification stage for Box-Jenkins autoregressive, moving average time series models[53].

Artificial Neural Networks

The class of adaptive systems known as Artificial Neural Networks (ANN) were motivated by the amazing parallel processing capabilities of biological brains (especially the human brain). The main motivation, at least initially, was to re-create these abilities by constructing artificial models of the biological neuron. The actual artificial neurons --as used in the ANN paradigm-- have little in common with their biological counterpart. Rather, they are primarily used as *computational devices*, clearly intended to problem solving: optimization, approximation of functions, identification of systems, classification, time-series prediction, and others.

The power of biological neural structures stems from the enormous number of highly interconnected simple units. The simplicity comes from the fact that, once the complex electro-chemical processes are abstracted, the resulting computation turns out to be conceptually very simple. Artificial neurons are modeled to take profit of this by proposing simple computing devices that resemble the abstracted original function. However, in the ANN paradigm only very few elements are connected in most practical situations (on the order of hundreds, to say the most) and their connectivity is low. Therefore, it seems reasonable, once the ``biological origin'' is so departed, to think in compensating these low numbers by increasing the power of single units, while retaining the conceptual simplicity of seeing a neuron (a computational unit) as a em pattern recognizer: a device that integrates its incoming sources with its own local information available and outputs a single value expressing how much do they match.

Another aspect of the artificial neural networks is that there are different architectures, which consequently requires different types of algorithms, but despite to be an apparently complex system, a neural network is relatively simple. The architectures used in this work are feed-forward. A mathematical model which represents you can find in Fig.2 page 8. From the illustrated model the interval activity of the neuron can be shown to be $v_k = \sum_{j=1}^{p} w_{kj}\, x_j$.

In addition, the activation function acts as a squashing function, such that the output of a neuron in a neural network is between certain values (usually 0 and 1, or -1 and 1). In general, there are three types of activation functions, denoted by $\Phi(\cdot)$.

[53] Engineering statistics handbook: Introduction to time series analysis. Available at http://www.itl.nist.gov/div898/handbook/eda/section3/autocopl.htm

- First, there is the Threshold Function which takes on a value of 0 if the summed input is less than a certain threshold value (v), and the value 1 if the summed input is greater than or equal to the threshold value.
- Secondly, there is the Piecewise-Linear function. This function again can take on the values of 0 or 1, but can also take on values between that depending on the amplification factor in a certain region of linear operation.
- Thirdly, there is the sigmoid function. This function can range between 0 and 1, but it is also sometimes useful to use the -1 to 1 range. An example of the sigmoid function is the hyperbolic tangent function.

Learning algorithm

The property that is of primary significance for a neural network is the ability of this network to learn from its environment, and to improve its performance through learning. The improvement in performance takes place over time in accordance with some prescribed measure. A neural network learns about its environment through an interactive process of adjustments applied to its synaptic weights and bias levels. Ideally, the network becomes more knowledgeable about its environment after each iteration of the learning process.
A prescribed set of well-defined rules for the solution of a learning problem is called a learning algorithm. As one would expect, there is no unique learning algorithm for the design of neural network. Rather, we have a “kit of tools” represented by a diverse variety of learning algorithms, each of which offers advantages of its own. Basically, learning algorithms differ from each other in the way in which the adjustment to a synaptic weight of a neuron is formulated. Another factor to be considered is the manner in which a neural network made up of a set of interconnected neurons, relates to its environment.

There’re two kinds of learning: supervised and unsupervised. The current work only is focused on supervised learning. Thus, the supervised learning basically can be considered as learning with a teacher. In conceptual terms, we may think of the teacher as having knowledge of the environment, with that knowledge being represented by a set of input-output examples. Supposing that the teacher and the neural network are both exposed to a training vector, the teacher is able to provide the neural network with a desired response for that training vector. Indeed, the desired response represents the optimum action to be performed by the neural network. The network parameters are adjusted under the combined influence of the training vector and the error signal, taking account that the error signal is defined as the difference between the desired response and the actual response of the network.

One of the most well-known learning rules are the delta rule and its extension: back propagation algorithm. The delta rule, also called the Least Mean Square (LMS), was developed by Widrow and Hoff, method is one of the most commonly used learning rules. For a given input vector, the output vector is compared to the correct answer (for linear activation functions).

On the other hand for using nonlinear activation functions (tnh, log, sinh) the delta rule

have to be generalized: equations $\delta_o^p = (d_o^p - y_o^p)F_o'(s_o^p)$ and $\delta_h^p = F'(s_h^p) \sum_{o=1}^{No}(\delta_o^p w_{ho})$ give a recursive procedure for computing the δ's for all units in the network, which are then used to compute the weight changes according to equation. This procedure constitutes the generalised delta rule for a feed-forward network of non-linear units. So, we know from the delta rule that, in order to reduce an error, we have to adapt its incoming weights according to $\Delta w_{ho} = (d_o - y_o)y_h$.

But it delta rule manly is used for perceptron networks or only is able for a network with one layer. For treating with more layers was developed an extension which is called, back-propagation algorithm. So, in order to adapt the weights from input to hidden units, we again want to apply the delta rule. In this case, however, we do not have a value for δ for the hidden units. This is solved by the chain rule which does the following: distribute the error of an output unit o to all the hidden units that is it connected to, weighted by this connection. Differently put, a hidden unit h receives a delta from each output unit o equal to the delta of that output unit weighted with (= multiplied by) the weight of the connection between those units.

The application of the generalized delta rule thus involves two phases: During the first phase the input x is presented and propagated forward through the network to compute the output values yp o for each output unit. This output is compared with its desired value do, resulting in an error signal δp for each output unit. The second phase involves a backward pass through the network during which the error signal is passed to each unit in the network and appropriate weight changes are calculated.

But as I said before, nowadays there are many learning algorithms. So in this study I have chosen the Bayesian regulation back-propagation[54] which is provided by Matlab framework. It is based on updating the weight and bias values according to Levenberg-Marquardt optimization and minimizing a combination of squared errors and weights, and then determines the correct combination so as to produce a network that generalizes well.

[54] MacKay (Neural Computation, Vol. 4, No. 3, 1992, pp. 415 to 447) and Foresee and Hagan (Proceedings of the International Joint Conference on Neural Networks, June, 1997) .

Types of neural networks

A brief overview about sort of neural networks we find that there are different ones but we could make a broadly classification grouping them in 2 categories: statics and dynamics networks. Static (feed-forward) networks have no feedback elements and contain no delays; the output is calculated directly from the input through feed-forward connections. However in dynamic networks, the output depends not only on the current input to the network, but also on the current or previous inputs, outputs, or states of the network.

Add delay make flourish the apparition of the term of *time* in neural networks which is considered as an essential dimension of learning. It has been incorporated into the design of a neural network implicitly or explicitly. For instances, a straightforward method of implicit representation of time is to add a short-term memory structure in the input layer of a static neural network (e.g., multilayer perceptron). The resulting configuration is sometimes called a focused time-lagged feed-forward network (TLFN) or focused time delay network (FTDN). So we can obtain the short-term memory structure may be implemented as Tapped-Delay-Line (TDL), even though exists other ways like Gamma Memory[55] but they are not described here because they are not the focuses of this study.

Hence, the TDL is the most commonly used form of short-term memory. It consists of p unit delays with *p + 1* terminals, as shown in Fig. 6, which may be viewed as a single input–multiple output network. The unit-delay is denoted by *z-1*. The memory depth of a TDL memory is fixed at p, and its memory resolution is fixed at unity, giving a depth resolution constant of *p*.

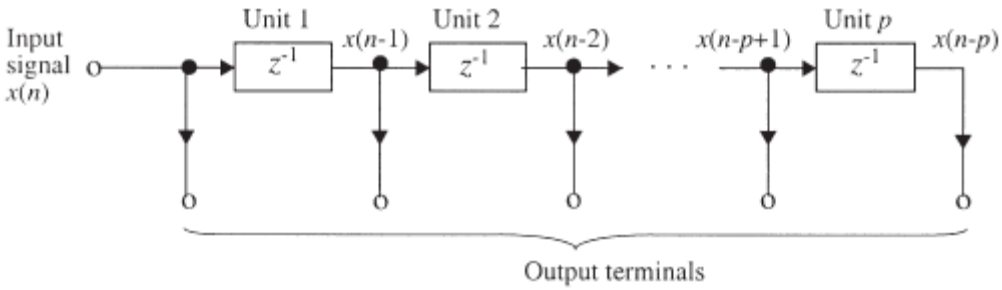


Figure 6: ordinary tapped delay line memory of order p [56]

Thus, summarizing this work is approached to use dynamic networks especially in already defined focused time-delay neural network (FTDNN) which consists in a feed-forward network with a tapped delay line at the input. The follow figure illustrates the typical network used in examples (overall in Matlab).

[55] For detailed treatment of gamma memory see: An Analysis of the Gamma Memory in Dynamic Neural Networks. Jose C. Principe and Jyh-Ming Kuo, and Same1 Celebi (1994).
[56] Feedforward Neural Networks: An introduction. Simo Hayken. pp. 12.

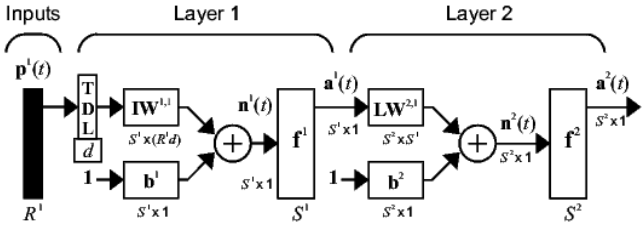


Figure 7: two-layer FTDNN [57]

On the other hand, other sort of neural network which is useful to model nonlinearity of time series is the recurrent networks. They are neural networks with one or more feedback loops. The feedback exist whenever the output of an element in the system influences in part the input applied to that particular element, thereby giving rise to one or more closed paths for the transmission of signals around the system. The next figure is an example of a recurrent network which consist of a single layer of neurons with each neuron fiding its ouput signal back to the inputs of the other neurons.

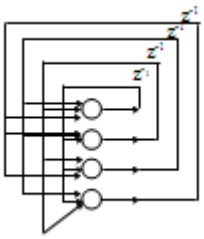


Figure 8: Single layer of neurons with each neuron fiding its ouput signal back to the inputs of the other neurons

Thus, unlike the feed-forward networks, recurrent networks offers at least one feedback loop. The presence of feedback loops has a profound impact on the learning capability of the network and on its performance. Moreover, the feedback loops involve the use of particular branches composed of unit-delay elements which result in a nonlinear dynamical behaviour, assuming that the neural network contains nonlinear units.

There are many sort of recurrent networks, but as this study is developed under Matlab framework, among the recurrent networks provided the most recommended to model time series is a nonlinear autoregressive with exogenous inputs (NARX) model, hence I'm going to explain more in detail it. A generic recurrent network that follows naturally from a multilayer perceptron would be:

[57] Two layer FTDNN, website http://www.mathworks.com/help/toolbox/nnet/dynamic3.html#34428

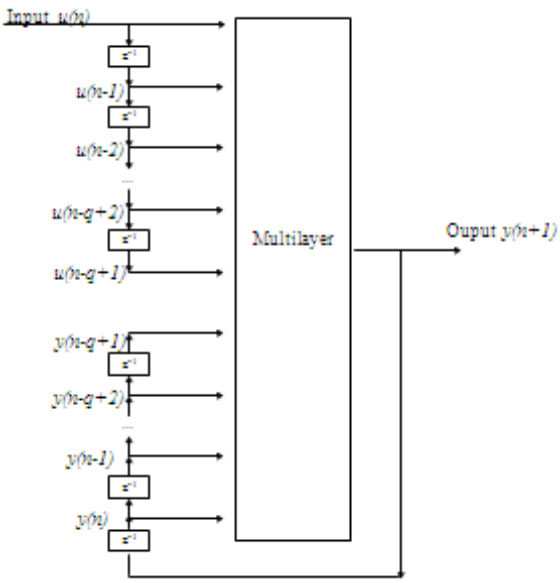


Figure 9: input that is applied to a tapped-delay-line memory of *q* units.

The model has a single input that is applied to a tapped-delay-line memory of *q* units. It has a single output that is fed back to the input via another tapped-delay-line memory also of *q* units. The contents of these two tapped-delay-line memories are used to feed the input layer of the multilayer perceptron. The present value of the model input is denoted by *u(n)*, and the corresponding value of the model output is denoted by *y(n+1)*. That is, the output is ahead of the input by one time unit. Thus, the signal vector applied to the input layer of the multilayer perceptron consists of a data window made up as follows:

- Present and past values of the input, namely *u(n), u(n-1),…,u(n-q+1)*, which represent exogenous inputs originating from outside the network.
- Delayed values of the output, namely *y(n),y(n-1),…,y(n-q+1)*, on which the model output *y(n+1)* is regressed.

Thus, this recurrent network is referred to as a nonlinear autoregressive with exogenous inputs (NARX) model. In Matlab is represented like:

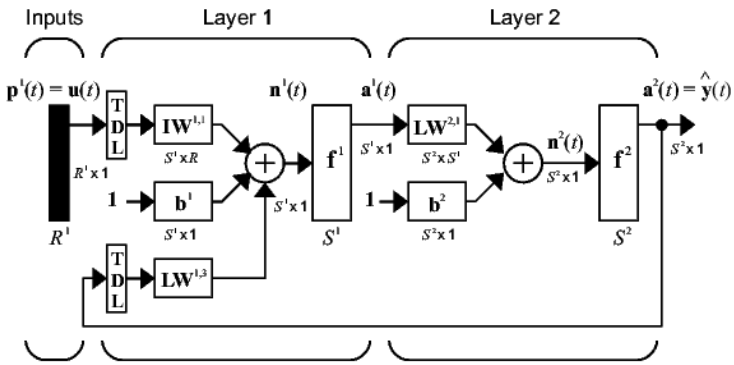


Figure 10: NARX network with two layers [58]

[58] Website http://www.mathworks.com/help/toolbox/nnet/dynamic6.html#35586

Notice this type of neural network works with two kinds of delays, the first coming from input values and the second from the output value.

Cross-Validation

Cross-validation is a method for estimating generalization error based on "resampling". The resulting estimates of generalization error are often used for choosing among various models, such as different network architectures. In k-fold cross-validation, you divide the data into k subsets of (approximately) equal size. You train the net k times, each time leaving out one of the subsets from training, but using only the omitted subset to compute whatever error criterion interests you. If k equals the sample size, this is called "leave-one-out" cross-validation. "Leave-v-out" is a more elaborate and expensive version of cross-validation that involves leaving out all possible subsets of v cases. In this study has been developed a cross validation algorithm due to the lack in Matlab framework.

Feature selection

When we make some work, we have to recollect information; take measures of different attributes linked to this problem and usually this process spend a lot of time and money. Every time we ask ourselves which features we could omit finding out which attributes are irrelevant or redundant in order to reduce the cost.

In the literature we can find several definitions about feature selection. Some of them are:

- Choosing a subset of the original features which will often lead to better performance.
- Attribute selection is a pre-process step needed in unsupervised knowledge discovery in order to reduce the number of irrelevant attributes that obfuscate the data.
- Also known as variable selection, feature reduction, attribute selection, variable subset selection, or dimension reduction is the technique, commonly used in machine learning, of selecting a subset of relevant features for building robust learning models.

Pre-processing the data to obtain a smaller set of representative features, retaining the optimal salient characteristics of the data, not only decreases the processing time but also leads to more compactness of the models learned and better generalization.

Others reasons for applying dimension reduction are:

- Improve learning efficiency
- Reduce the cost of measurements

- Enable better understanding of the underlying process
- Eliminate noise
- Improve the prediction performance
- Visualization

Platform Development

Description

The raw dataset is composed by 78048x9 which provides 35 assets and the values of each asset go from 03/01/2000 until 30/06/2010.

	Name	ISIN
1	ABENGOA SA	ES0105200416
2	ABERTIS INFRAESTRUCTURAS SA	ES0111845014
3	EBRO FOODS SA	ES0112501012
4	BANCO BILBAO VIZCAYA ARGENTA	ES0113211835
5	BANCO ESP CREDITO (BANESTO)	ES0113440038
6	BANKINTER SA	ES0113679I37
7	BANCO POPULAR ESPANOL	ES0113790531
8	BANCO DE SABADELL SA	ES0113860A34
9	BANCO SANTANDER SA	ES0113900J37
10	BOLSAS Y MERCADOS ESPANOLES	ES0115056139
11	GAS NATURAL SDG SA	ES0116870314
12	INDRA SISTEMAS SA	ES0118594417
13	FERROVIAL SA	ES0118900010
14	FOMENTO DE CONSTRUC Y CONTRA	ES0122060314
15	MAPFRE SA	ES0124244E34
16	ACCIONA SA	ES0125220311
17	ENDESA SA	ES0130670112
18	ENAGAS	ES0130960018
19	ACERINOX SA	ES0132105018
20	CRITERIA CAIXACORP SA	ES0140609019
21	OBRASCON HUARTE LAIN S.A.	ES0142090317
22	GAMESA CORP TECNOLOGICA SA	ES0143416115
23	IBERDROLA SA	ES0144580Y14
24	IBERIA LINEAS AER DE ESPANA	ES0147200036
25	IBERDROLA RENOVABLES SA	ES0147645016
26	INDITEX	ES0148396015
27	GESTEVISION TELECINCO SA	ES0152503035
28	ACS ACTIVIDADES CONS Y SERV	ES0167050915
29	GRIFOLS SA	ES0171996012
30	RED ELECTRICA CORPORACION SA	ES0173093115
31	REPSOL YPF SA	ES0173516115
32	TECNICAS REUNIDAS SA	ES0178165017
33	TELEFONICA SA	ES0178430E18
34	SACYR VALLEHERMOSO SA	ES0182870214
35	ARCELORMITTAL	LU0323134006

Table 1: 35 assets of dataset

TAG	Description	Type
ISIN	Asset' ID	numeric
DATE	The date of trading session	numeric
P_CLOSE	The value of price of an asset in a trading session	numeric
BID	Bid price is the price that somebody will pay for a stock at a given moment.	numeric
ASK	Ask price is the price at which someone is willing to sell a stock.	numeric
VOLUME	Number of shares traded for one asset during the trading session, excluding applications and blocks	numeric
TURN_OVER	Number of total shares traded for one asset during the trading session.	numeric
NUM_OP	Number of operations done in a trading session	numeric
CAPITAL	The capital accumulated for one asset in one trading session	numeric

Table 2: Raw dataset

This serie of features have been provided by experts, but it is easy to see as they are correlated with liquidity risk. According to the page 18, *Liquidity risk* section, the liquidity risk can be estimated in base of bid-ask-spread, transaction or volume data models. Then we have linked the bid, ask and volume features, but obviously the close price implicitly also is corresponded so that it will be the fair market price. Regarding the rest, some authors incorporate variation focusing on the use of high frequency transaction level data of stocks (see *turn_over* and *num_op*) according to their average transaction prices and capitalization. Even though the model proposed by the experts is:

TAG	Description	Type
VOLUME	Number of shares traded for one asset during the trading session, excluding applications and blocks	numeric
TURN_OVER	Number of total shares traded for one asset during the trading session.	numeric
NUM_OP	Number of operations done in a trading session	numeric
CAPITAL	The capital accumulated for one asset in one trading session	numeric
VOLATILITY	Standard deviation of the profitability since 90 days ago	Numeric
SPREAD	The difference between BID and ASK	Numeric

Table 3: Liquidity risk model provided by economic and financial experts

Where *volatility* and *spread* will be handled in *Feature selection* subsection. The *ISIN* and *date* are only informative attributes which are not used to get the model.

Pre-process

In pre-process I've noticed different tests to do in order to discard or choose an optimal characteristic/specific dataset instead of other. For instances, we can apply different methodologies to treat missing values, to normalize them or in the other hand remove tendency following the behaviour of classical models.

Exploring data

But first of all, it would be necessary get target value which is the difference between Bid and Ask. Thus, firstly I split the subset by ISIN such that each one represents a time series. So I have 35 subsets following this form:

TAG	Description	Type
P_CLOSE	The value of price of an asset in a trading session	numeric
BID	Bid price is the price that somebody will pay for a stock at a given moment.	numeric
ASK	Ask price is the price at which someone is willing to sell a stock.	numeric
VOLUME	Number of shares traded for one asset during the trading session, excluding applications and blocks	numeric
TURN_OVER	Number of total shares traded for one asset during the trading session.	numeric
NUM_OP	Number of operations done in a trading session	numeric
CAPITAL	the capital accumulated for one asset in one trading session	numeric

Table 4: Form of subsets

Before to get the spread, I checked the missing values for each subset so the next subsection treat it in deep.

Missing values

The missing values must be treated so that keeping them can be an inconvenient to apply afterward a policy optimization like standardizes inputs. Some ways to do it are:

- To remove the observation which contain it: as we are working with time series it is not good idea so that we lose information and give up the sequence of time series.

- To replace missing values by the mean what is not enough efficient so that in large period can have big variance and we can introduce noise.

- To replace missing values searching the k neighbours is the most sophisticated and efficient way to treat it so that we're searching for the nearest values of them.

I summarize how many missing values are distributed along of the raw dataset.

TOTAL Size:78048	**Attribute**	**Number of missing**	%
	P_CLOSE	2	0.0026
	BID	56	0.0718
	ASK	50	0.0641
	VOLUME	57	0.0730
	TURN_OVER	11591	**14.8511**
	CAPITAL	1868	**2.3934**
	NUM_OP	41539	**53.2224**

Table 5: Total percentages by each explicatory feature

For each missing values detail subset you can see the appendix in *Missing* d*etail* section.

In order to get the spread I have decided to replace the missing values over BID and ASK features applying the mean between the 2 nearest values in basis of Standardized Euclidean distance over the premise that they are time series with low variance in a short period. Afterwards I obtain the spreads.

From the totals, the reader can see what features have most number of missing, they are bolded. So it is alarming the number of missing values in the feature *num_op* which is greater than 50 % of the total, thus it means that ½ of assets don't have this value. So although I replace it using whichever optimal technique to I would probably provide a big error. Thus I could reformulate the initial model deleting it but previously I'm interested in knowing its weight in the model .

I have found three subsets which have no missing values in *num_op* field: ES0147645016, ES0187165017 and LU0323134006. Therefore in order to test their importance I have got correlation coefficients and their p-values so that it let me to know how much they are linearly correlated to the target.

ISIN	**Corr.**	**p-value**
ES0147645016	0.1551	7,8025e-05
ES0187165017	0.0507	0,1049
LU0323134006	0.1610	3,2439e-07

Table 6: coef. Correlation and p-values obtained of *Spread* versus *num_op* features

As above results show, all of them are insignificant producing a correlation coefficients very lowers (justified by their p-values except for ES0187165017). Therefore, we could say that there is not lineal correlation between *bid-ask spread* and *num_op*. Furthermore, I've done another experiment to contrast the relevance of *num_op*.

For each of the 3 subsets described above, I've trained a feed-forward time delay network with and without the attribute *num_op* and fixing the parameters:

Type	Train. Alg.	# Neurons	TDelay	Stand.	CV	Error
FFTD	Trainrb	30	2	Mean=0 Std=1	10	MSE NRMSE

Table 7: parameters of neural network

ISIN	Num_op	NRMSE
ES0147645016	With	1.4113
	Without	**1.3545**
ES0187165017	With	6.7169
	Without	**6.2927**
LU0323134006	With	1.8453
	Without	**1.8345**

Table 8: results of with and without *num_op* attribute

Checking the results above, we can notice that in base of the error in all cases we obtain lower error without the feature *num_op*. So, further we can add that practically don't exist lineal correlation, thus we can infer that *num_op* is irrelevant or does not provide useful information, hence I have chosen do the experiments without it.

In addition, we saw that *capital* and *turn_over* features have a large percentage overall in some specifics subsets. Thus, I have taken subsets with the treated features completely full and following the methodology used in previous case, I have performanced the experiments to attempts to find out the lineal and non-lineal correlation between these features and the target.

ISIN	Corr.	p-value
ES0113900J37	-0.0565	0.0036
ES0115056139	0.0437	0.1657
ES0171996012	0.0481	0.1195
ES0173093115	-0.0458	0.0008
ES0187165017	-0.0460	0.1419
ES0178430E18	-0.0678	0.0005

Table 9: coef. correlation and p-values obtained of Spread versus *capital* features

ISIN	Capital	NRMSE
ES0113900J37	With	2.3222
	Without	**1.8837**
ES0115056139	With	**1.2489**
	Without	1.2542
ES0171996012	With	2.0212
	Without	**1.7427**
ES0173093115	With	2.1305
	Without	**1.425**
ES0187165017	With	1.6938
	Without	**1.1433**
ES0178430E18	With	1.0711
	Without	**1.0697**

Table 10: results of with and without *capital* attribute

If we check both cases, we find out that there are cases with particular behaviour. Focusing in first set, where were analyzed 6 subsets with *capital* feature, we find that all of them have a low lineal correlation but only three of them (ES0113900J37, ES0173093115, ES0178430E18) the results are significative (p-value higher than 0.05). In other hand, looking for a non-lineal correlation, the same subsets were analyzed in base of the results from ANN. In that way, we find that 5 subsets

have not significance information to predict the spread: ES0113900J37, ES0171996012, ES0173093115, ES0187165017, ES0178430E18. Otherwise the remainder subset, ES0115056139, shows a very little improvement 1.2489 versus 1.2542 which may be due to noise component. Thus, summarizing we may infer that the *capital* feature can be remove from the original dataset.

ISIN	Corr.	p-value
ES0113860A34	-0.0091	0.6606
ES0113211835	-0.0781	0.0001
ES0113679137	0.0560	0.0040
ES0124244E34	0.0431	0.0267
ES0113790531	-0.0092	0.6359
ES0115056139	0.0833	0.0082
ES0118594417	-0.0677	0.0005
ES0118900010	-0.0033	0.8991
ES0122060314	-0.0147	0.4507
ES0171996012	0.0374	0.2263

Table 11: coef. correlation and p-values obtained of *Spread* versus *turn_over* features

ISIN	Turn_over	NRMSE
ES0113860A34	With	1.0457
	Without	**1.0408**
ES0113211835	With	1.2337
	Without	**1.2309**
ES0113679137	With	4.8211
	Without	**2.2451**
ES0124244E34	With	1.0742
	Without	**1.0755**
ES0113790531	With	2.9945
	Without	**1.3809**
ES0115056139	With	1.2181
	Without	**1.0879**
ES0118594417	With	**1.117**
	Without	1.1724
ES0118900010	With	**1.0099**
	Without	1.0135
ES0122060314	With	**1.0919**
	Without	1.1735
ES0171996012	With	1.2655
	Without	**1.1966**

Table 12: results of with and without *turn_over* attribute

Regarding to the second set, it is composed by 10 assets which have been analyzed with and without *turn_over* feature. From the results, table 11 we find out that again all of them show a low correlation even though only five (ES0113211835, ES0113679137, ES0124244E34, ES0115056139, ES0118594417) are significant. According to table 12 we find that 7 of them (ES0113860A34, ES0113211835, ES0113679137, ES0124244E34, ES0113790531, ES0115056139, ES0171996012) achieve a lower error. In other words, the *turn_over* feature don't provide useful inforfmation to the neural network . So due to almost of subsets have missing values and that the results show a better perfomance without the feature I have also decided to remove it from the original dataset.

Summarizing, from of three features to be study I have removed so that they have a high number of missing values and keep them in input set didn't enhance perfomance such as show a their really lower correlation with the feature to be predicted.

Outliers

Outliers are data values that are dramatically different from patterns in the rest of the data. They may be due to measurement error, or they may represent significant features in the data. Identifying outliers, and deciding what to do with them, depends on an understanding of the data and its source.

Thus, from the dataset without missing values are identified the possible outliers where an outlier is defined as a value that is more than three standard deviations away from the mean.

In total terms, the proportion of outliers looks no relevant but it is important how they are distributed.

	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
Total	69	70	68	1043	1084	349	1017
%	0.0884	0.0897	0.0871	1.3364	1.3889	0.4472	1.3030

Tabla 10: Outliers of raw dataset showed in number of elements and %

In mode of guide, I have commented in deep the asset ES0113860A34.

	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
Outliers	0	0	0	25	27	0	49
%	0	0	0	0.0108	0.0116	0	0.0211

Table 11: Outliers of the subset ES0113860A34

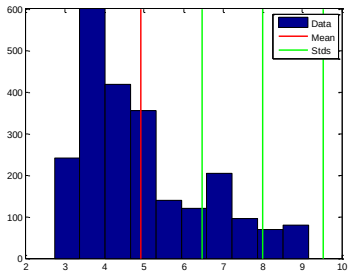


Figure 11: P_CLOSE outliers

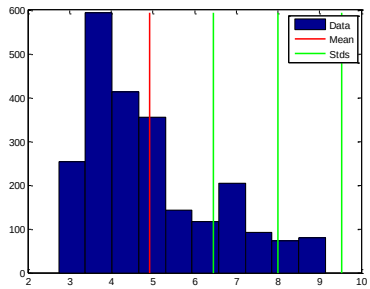


Figure 12: BID outliers

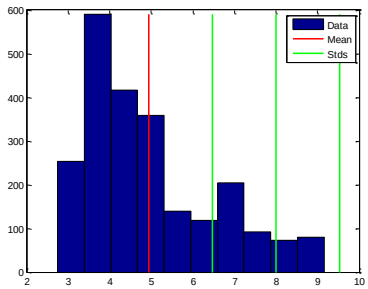


Figure 13: ASK outliers

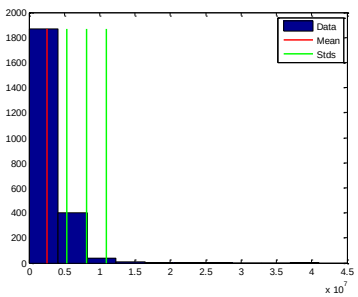


Figure 14: VOLUME outliers

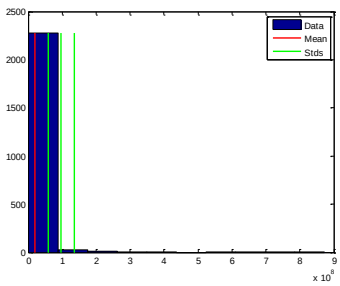


Figure 15: TURN_OVER outliers

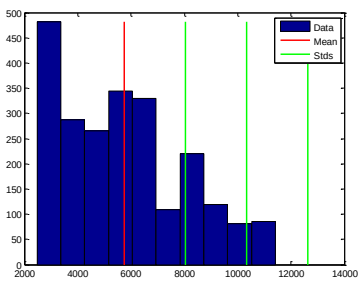


Figure 16: CAPITAL outliers

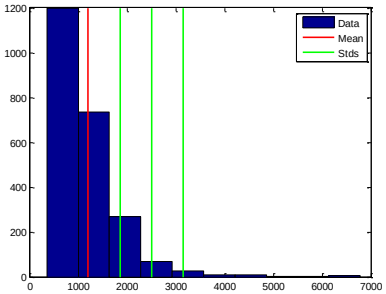


Figure 17: NUM_OP outliers

How is showed in above table and the followed figures, particularity in this case, the number of outliers in subset almost is not relevant being lower of 1% the amount of each one for each feature. So in this case, the outliers could be considerate like extreme value but not incorrect. In general cases the highest is over 2.0770% of deviation, equally it is a low percentage so the bias that can produce is minimum.

If the reader want to check the rest of outliers, he can find in *Appendix*, *Outliers Details* section.

Standardize data

Following the theory it usually advises standardize the values in order to optimize algorithms for weight determination in neural networks avoiding saturate them and as consequence they lose capacity to learn. So in order to define it we can say that normalizing, in the NN literature, mostly it refers to rescaling by the minimum and range of the vector, to make all the elements lie between 0 and 1.

In the literature we can find different cases, for example: If the input variables are combined via a distance function (such as Euclidean distance) in an RBF network, standardizing inputs can be crucial, but if the input variables are combined linearly, as in an MLP, then it is rarely strictly necessary to standardize the inputs, at least in theory. The reason is that any rescaling of an input vector can be effectively undone by changing the corresponding weights and biases, leaving you with the exact same outputs as you had before. However, there are a variety of practical reasons why standardizing the inputs can make training faster and reduce the chances of getting stuck in local optima. The main emphasis in the literature on initial values has been on the avoidance of saturation, hence the desire to use small random values. How small these random values should be depends on the scale of the inputs as well as the number of inputs and their correlations. Standardizing inputs removes the problem of scale dependence of the initial weights.

I show a little proof attempting to demonstrate it. Consider an MLP with two inputs (X and Y) and 100 hidden units and comparing the inputs over the interval [-10, 10] and the interval [-1, 1].

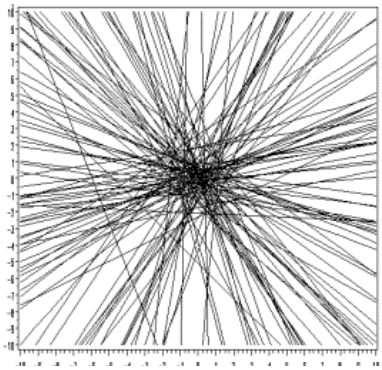


Figure 18: Hiperplanes over range [-10,10]

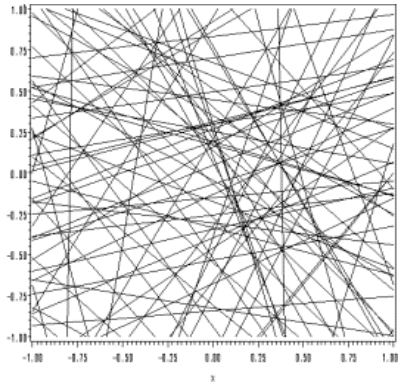


Figure 19: Hiperplanes over range [-1,1]

We can observe that in first figure where the range is over [-10, 10] the most of hyperplanes are concentrated in the centre missing the corners, so if the networks needs to learn such a hyperplane may take many random initializations before training finds a local optimum with such a hyperplane. On the other hand, if we rescale the range of input values, we find a totally different picture where we check that the hyperplane are more successfully distributed along the range easing the learning.

So normalization data is an important factor to consider, and in addition in the literature is found a close relation with transfer function in order to match the range of the outputs where logistic or hyperbolic tangent function has a typical range of [0,1] and [-1,1] respectively.

Thus, I've done a few proofs in order to evaluate which range is more efficient in the present case.I have taken the first results of *num_op* from outlier section (page 25) which are done using mean 0 and standard deviation 1 versus a range -1,+1.

Type	Train.Alg.	# Neurons	TDelay	CV	Error
FFTD	Trainrb	30	2	10	MSE NRMSE

Table 13:parameters of neural network

Stand.	ISIN	MSE	NRMSE
m=0, std=1	ES0147645016	0.00029001	1.3545
m=0, std=1	ES0187165017	0.29644	6.2927
m=0, std=1	LU0323134006	0.059013	1.8345
-1 , +1	ES0147645016	**0.00015031**	**1.1868**
-1 , +1	ES0187165017	**0.033037**	**1.1174**
-1 , +1	LU0323134006	**0.047977**	**1.2322**

Table 14: results of whit and without *num_op* attribute

The above results show working with range -1, +1 the NN obtain better results than standardizing to mean=0 and deviation 1.

Trend

The trend effect play a relevant role at the moment of modeling a time series due to its presences can produce a bad learning of the neural network, thus it should be treated. Several researches denoting that for a forecasting problem which involves continuous target values, it is reasonable to use a linear activation function for output nodes. Even so, the ANNs with linear output have the limitation that they cannot model a time series containing a trend (Cottrell, M., Girard, B., Girard,Y., Mangeas, M., Muller, C., 1995)[59]. Therefore I have done a brief study about how the trend can effects in the TS in base of 5 randomly assets.
Firstly I have indentified the trend the spread feature in ES0113860A34, ES0113900J37, ES0124244E34, ES0144580Y14, ES0105200416 subsets.

The following figure, referencing to ES0113860A34 asset:

ISIN ES0113860A34

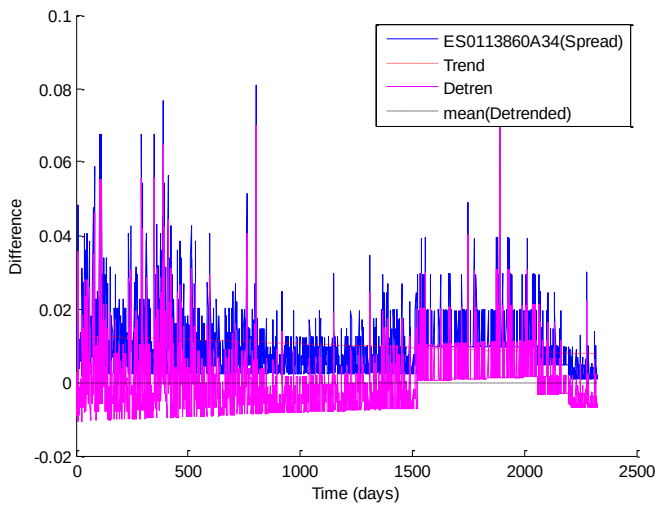


Figure 20: Analysis of trend of spread for the asset ES0113860A34 along all period

Now it is zoomed to the last year period, understanding it has 261 days (without weekends), so that being more interpretable:

[59] Cottrell, M., Girard, B., Girard,Y., Mangeas, M., Muller, C., 1995. Neural modeling for time series: a statistical stepwise method for weight elimination. IEEE Transactions on Neural Networks 6 (6), pp. 1355–1364.

ISIN ES0113860A34

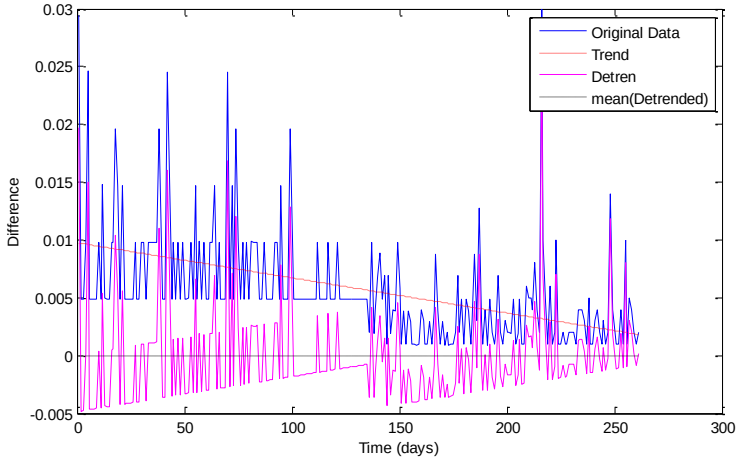


Figure 21: Analysis of trend of spread for the asset ES0113860A34 along 261 days

Now you can notice that a clear little downward trend is present. However as the spread feature derive of the difference of Bid and Ask variables, it take values near to 0 which makes almost imperceptible trend. So also it has been studied the Bid and Ask components with the purpose of getting more clear how affect the trend over them.

ISINES0113860A34 (BID)

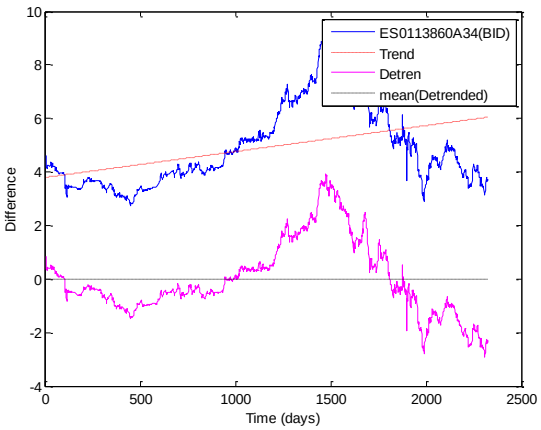


Figure 22: Analysis of trend of BID for the asset ES0113860A34

ISINES0113860A34 (ASK)

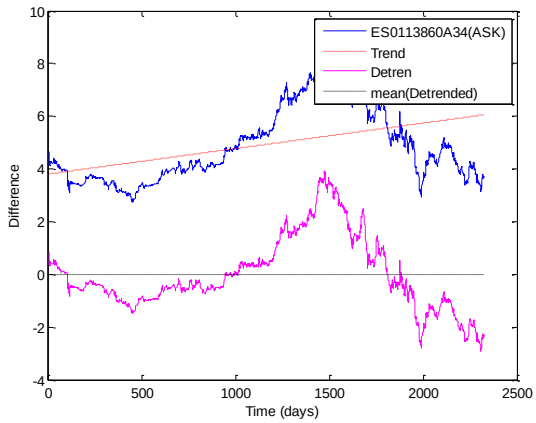


Figure 23: Analysis of trend of ASK for the asset ES0113860A34

As you can see, in both cases the trend is well denoted thus maybe I have to considerate remove it from the input. For the rest of the trends analysis you can check *Trends Detail* section of the *Appendix*.

Seasonality

As I have commented above, in *Basic of time series* subsection (*Time Series* section), the treatment of seasonality is a complex issue which doesn't has a unique solution so that it heavily depends of the nature of time series to be treated. But introducing to NNs, have been supossed that being nonlinear and data-driven in nature, may be well suited to model seasonality interacting with other components, and may relieve the practical burden of a priori model selection.

A large body of literature exist in seasonal time-series analysis and forecasting. Many recent studies fight about the importance of deseasonality the series before to be predicted. For instance we find Sharda and Patil(1992)[60] who conclude that the NNs are able to incorporate seasonality automatically, however Zhan and Qui (2001)[61]find that not only preadjustment of seasonality is important, but a combined approach of detrending and deseasonality is most effective in forecasting performance.

As there is no clear and I don't have enough experience and knowledge about it, I have decided to trust in the capability of NNss to incorporate seasonality so I have not treated it in target serie.

Feature Extraction

As I have commented above, the target value has to be calculated so it can be considerate like a derivate feature. Another one, recommended by the financial experts, is the *volatility* which is the standard deviation of the profitability. In the present case, it has been calculated for 90 days ago and the profitability has been computed over the close price, *p_close*, vector doing the difference between the $price_{t-1}$ and $price_t$ divided by $price_{t-1}$.

Thus, summarizing we have 2 news features:

TAG	Description	Type
SPREAD	The difference between BID and ASK	Numeric
VOLATILITY	Standard deviation of the profitability since 90 days ago	Numeric

Table 15: new features

[60] R. Sharda and R. B. Patil, "Connectionist approach to time series prediction: An empirical test," J. Intell. Manuf., vol. 3, pp. 317–323, 1992.

[61] M. Qi and G. P. Zhang, "An investigation of model selection criteria for neural network time series forecasting," Eur. J. Operat. Res., vol. 132, no. 3, pp. 188–102, 2001

Modeling

Summarinzing the previous section, I have got the final model, from economic model, ready to be input's networks.

TAG	Description	Type
VOLUME	Number of shares traded for one asset during the trading session, excluding applications and blocks	numeric
VOLATILITY	Standard deviation of the profitability since 90 days ago	Numeric
SPREAD	The difference between BID and ASK	Numeric

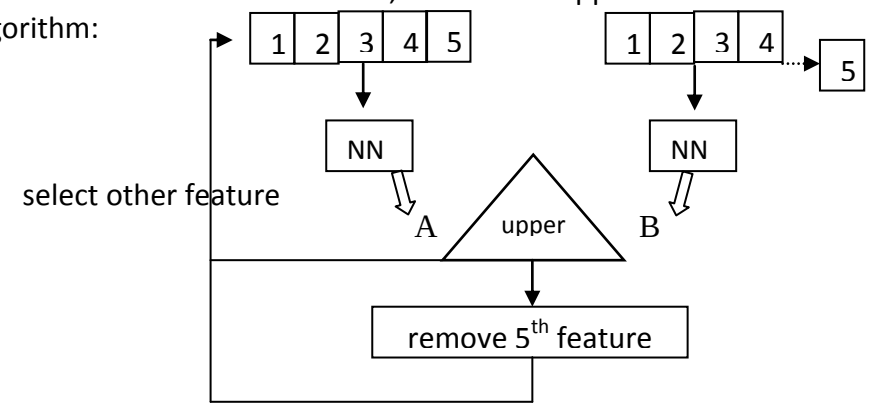
Table 16: Liquidity risk model provided by economic and financial experts without *num_op*, *capital* and *turn_over* features

AS I removed three important features following the economic model I propose add the *bid*, *ask* and *p_close* as input values attempting to provide more information. May they can be considered redundant or insignificant information (noise) so that the target, spread, is a lineal combination of *bid* and *ask* and *p_close* could be considered like a mean of both. Even so as I use a pruning algorithm (it is explained later), in the case that they provide insignificant information they should be removed. So, finalizing the input model would be:

TAG	Description	Type
P_CLOSE	The value of price of an asset in a trading session	numeric
BID	Bid price is the price that somebody will pay for a stock at a given moment.	numeric
ASK	Ask price is the price at which someone is willing to sell a stock.	numeric
VOLUME	Number of shares traded for one asset during the trading session, excluding applications and blocks	numeric
VOLATILITY	Standard deviation of the profitability since 90 days ago	Numeric
SPREAD	The difference between BID and ASK	Numeric

Table 17: final dataset after pre-processing and take into account some considerations.

On the other hand, to enhnance performance one approach would be to include all the variables in the network and perform an analysis of the connection weights or a sensitivity analysis to determine which may be eliminated without reducing predictive accuracy. An alternative is to begin with a all number of variables and remove variable by variable keeping which improve network performance. In this research, the latest approach have been used and was called pruning algorithm:



This example shows how it works for a one feature. The 5th feature is temporally removed from the original dataset. Afterward to have both performance, they are comparated and if the error performance of all dataset is upper than the other it means that to include the 5th feature provide more noise than significant information therefore it will be removed. So, once evaluated one feature, the algorithm choose the next one until it has evaluated all of them. The final set resulting would be the optimal composition (it will have the lowest error performance).

Once defined the input set and how it will be treated, I would like to comment that firstly, each subset has been split in 2 subsets: for training 2/3 and 1/3 for testing. Over the first will be applied Cross Validation of k-folds to know if the final model has learned well and do good generalization. In order to get the best model I design a model of neural networks ensemble.
Among different kind of neural networks, I have chosen use Feed-Forward-Time-Delay Neural Network (*FFTD)* and Recurrent Dynamic Neural Networks *(NARX - nonlinear autoregressive network with exogenous inputs)*, so that are well-known their capability to capture the complex underlying structure of the time series. The use of recurrent networks for prediction can be found in Gent and Sheppard (1992)[62], Connor et al.(1994)[63], Kuan and Liu (1995)[64].
Thus, the subsets are applied to 2 kinds of networks: FDNN and RNN and their combination of parameters getting the best fitted model for each one and therefore a set of 35 (=number of assets to model) models. The best fitted model will be which have the lowest validation error).

In addition, is well-known that the autocorrelation of a time series is used to determine how many steps ago are significant which means how many steps ahead can be predicts. Following this idea for all subset I have calculated, in base of the Anderson[65] studies who determine the range of no signification coefficient of correlation with a risk of 5% supposing that the time series follow $N(0;V(r_k))$, the autocorrelation function (ACF), $r_k = \frac{\widehat{y_k}}{\widehat{y_o}}$ where $\widehat{y_k} = \frac{\sum_{i=0}^{N-k}(y_i-\bar{y})(y_{i+k}-\bar{y})}{N}$ (autocovariance) and $\widehat{y_o} = \frac{\sum_{i=0}^{N}(y_i-\bar{y})^2}{N}$ (variance). Logically it will be done only over the target.

Thus once I have the correlogram, the range $\pm 2S(r_k)$, established by Anderson, will show the set of values that can take r_k for that, with a risk of 5%, it can be accepted the missing correlation between the serie' values. Therefore, it will be useful if we consider only as coefficients of autocorrelation no nulls whose estimation is out of that range. For instances the next figure shows the correlogram of one asset with the range, where we can notice that aprox. 190 values will have non-zero correlation.

---

[62] Gent, C.R., Sheppard, C.P., 1992. Predicting time series by a fully connected neural network trained by back propagation. Computing and control Engineering Journal 3 (3), May, pp.109–112
[63] Connor, J.T., Martin, R.D., Atlas, L.E., 1994. Recurrent neural networks and robust time series prediction. IEEE Transaction on Neural Networks 51 (2), pp. 240–254.
[64] Kuan, C.-M., Liu, T., 1995. Forecasting exchange rates using feedforward and recurrent neural networks. Journal of Applied Econometrics 10, pp.347–364.
[65] Time Series Analysis and Forecasting, O.D. Anderson, ed. Butterworths, 1997

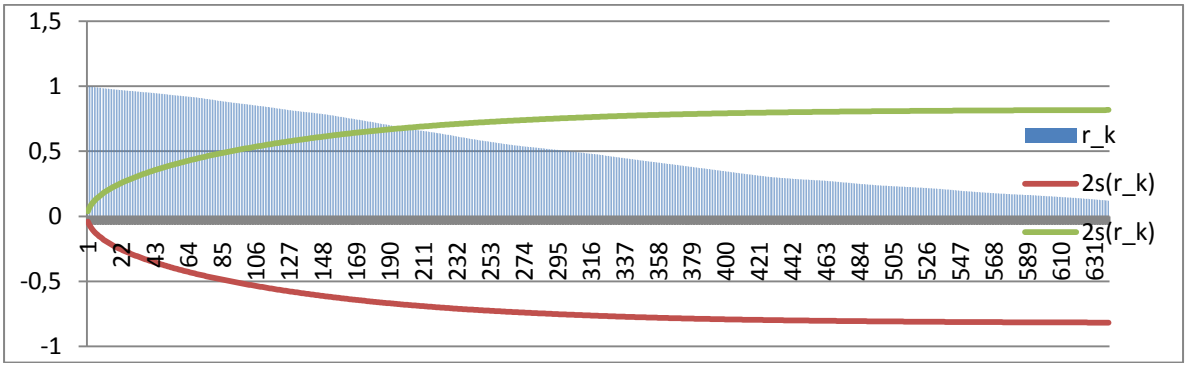


Figure 24: range $\pm\mathbf{2S}(\mathbf{r_k})$, on the correlogram of an asset

The mean over the results is 91 which mean that using the current database is admissible to do predictions to 1 steps ahead until 91 but it always will change according to the nature of time series. Certainly the results are high values a maybe it is caused a misleading selection due to the presence of distinct seasonal variations[66]. It is known that seasonal variation in the time series increases the autocorrelation of the data and produces a link that does not correspond to a dynamic connection in the data. For the rest of autocorrelations results you can check Autocorrelations Detail section of the Appendix. Otherwise, I would be naive if I don't explain that in real experiments the size of delays have been lower so that they are calculated from a 2/3 (training set) of each subsets.

An important assumption in time series forecasting is that past values in the series can be used to predict future values. Given this assumption, an important question is how many delays periods should be included in predicting the future. But, if the lag is chosen too small, there is little information contained in each subsequent datum. In contrast, if the lag is too large it will adversely affect the training time of the network, and the algorithm will likely be trapped in local optimal solutions. So the number of lags must be large enough to content relevant information and small enough to avoid the curse of dimensionality.

Some authors designed experiments to help select the number of delays while like Schwarz Information Criterion (SIC), the Akaike Information Criterion (AIC), the Hannan-Quinn Criterion (HQC), Bayesian Information Criterion (BIC), etc and others adopted some intuitive of empirical ideas.

In ANNs input variables need to be predictive, but should not be correlated. Correlated input variables are those that have essentially the same information in a slightly different form. They degrade ANN performance by interacting with each other as well as other elements to produce a biased effect. Therefore, the network gets confused and don't achieve a good generalization. Thus, in general, we could say that the continuous lag periods have higher degree of correlation.

[66] Porporato A. and Ridolfi L. (1997). Nonlinear analysis of river flow time sequences. Water Resources Research 33 (6), pp.1353-1367

Therefore, in base of this assumption I have calculated the lagged values belonging to each asset following the Autocorrelation Criterion (AC)[67], so that its implementation and understanding is easy and simple. It is based in seek periods of lag which have a high degree of correlation with y_t as possible while as low degree of correlation to the previous lag periods as possible, where y_t is the initial period.

Supposing that we have determined of lags periods $a(1),a(2),\ldots,a(m)$ If we want to add a new lag period $a(m+1)$, we wish $y_{t-a(m+1)}$ has as high degree of correlation to y_t as possible while as low degree of correlation to $y_{t-a(1)}, y_{t-a(2)},\ldots, y_{t-a(m)}$ as possible. Hence we would have:

- Step 1. Set upper limit of lag period N. Let $a(1) = 1$ and $m = 1$.
- Step 2. $a(m+1) = arg_k Max\{(|r_k|)/(\sum_{i=1}^{m} |r_{k-a(i)}|)\}$, $k=a(m)+1, a(m)+2,\ldots,N$
- Step 3. Let $m = m+1$. If $a(m)$ is less than N, go to Step 2. Otherwise, exit.

Another important point in getting the model is how many number of neurons and layers must be employed and there are several literature providing different models to approximate it but they have large dispersion, I mean if you make an exhaustive study about the proposed models quickly will notice that they accomplish opposite or sparse results so I suppose that each one must works under bias. For example, Lippmann (1987)[68] and Hecht-Nielsen (1990)[69] proposed $2n+1$, Wong (1991)[70] $2n$, Tang and Fishwick (1993)[71] n and Kang (1991)[72] $n/2$, where n is the number of input nodes. Thus avoiding to choose one of them and knowing that, generally speaking, too many nodes in a hidden layer (too many connections) produce a network that memorizes the input data and lacks the ability to generalize I have developed an algorithm in order to avoid it, where I have tested all subsets applying an increasing logarithm function (where if n is the number of neurons and each step it will be set to $n=n*2$ initializing it to $n=2$) until find an error higher than the previous.

Afterwards having a range, I have performed a search based on *golden section search* which will achieve the best approximation to the local minimums error. Under the assumption that the error got from validation set go down while is increased the number of neurons till arrive a number which produce more error due to input was too memorized. The next figure illustrates the followed steps:

[67] A general approach based on autocorrelation to determine input variables of neural networks for time series forecasting. (Vol. 17 No. 3 Journal of Systems Science and Complexity Jul., 2004)

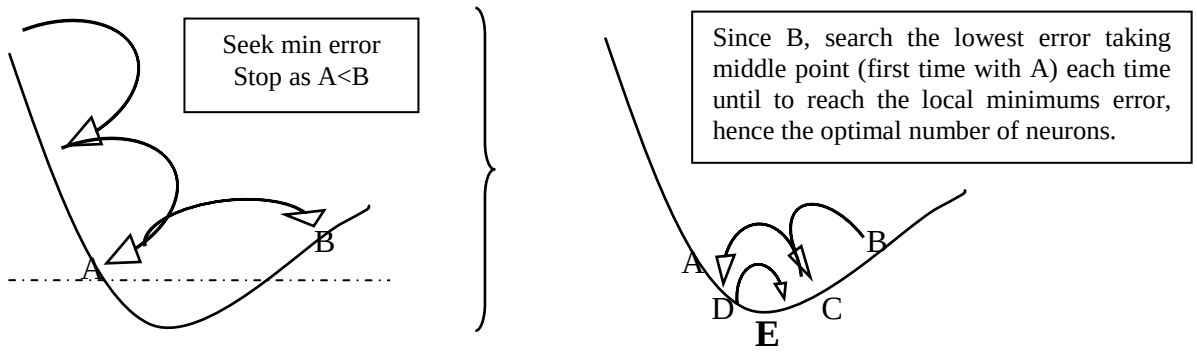
[68] An introduction to computing with neural nets, Lippmann, R.P., (1987). IEEE ASSP Magazine, April, pp. 4–22.

[69] Hecht-Nielsen, R., (1990). Neurocomputing. Addison-Wesley, Menlo Park, CA.

[70] Time series forecasting using backpropagation neural networks. Wong, F.S., (1991). Neurocomputing 2, pp. 147–159

[71] Feedforward neural nets as models for time series forecasting. Tang, Z. and Fishwick, P.A., (1993) ORSA Journal on Computing 5 4, pp. 374–385

[72] An Investigation of the Use of Feedforward Neural Networks for Forecasting. Kang, S., (1991). Ph.D. Thesis, Kent State University.



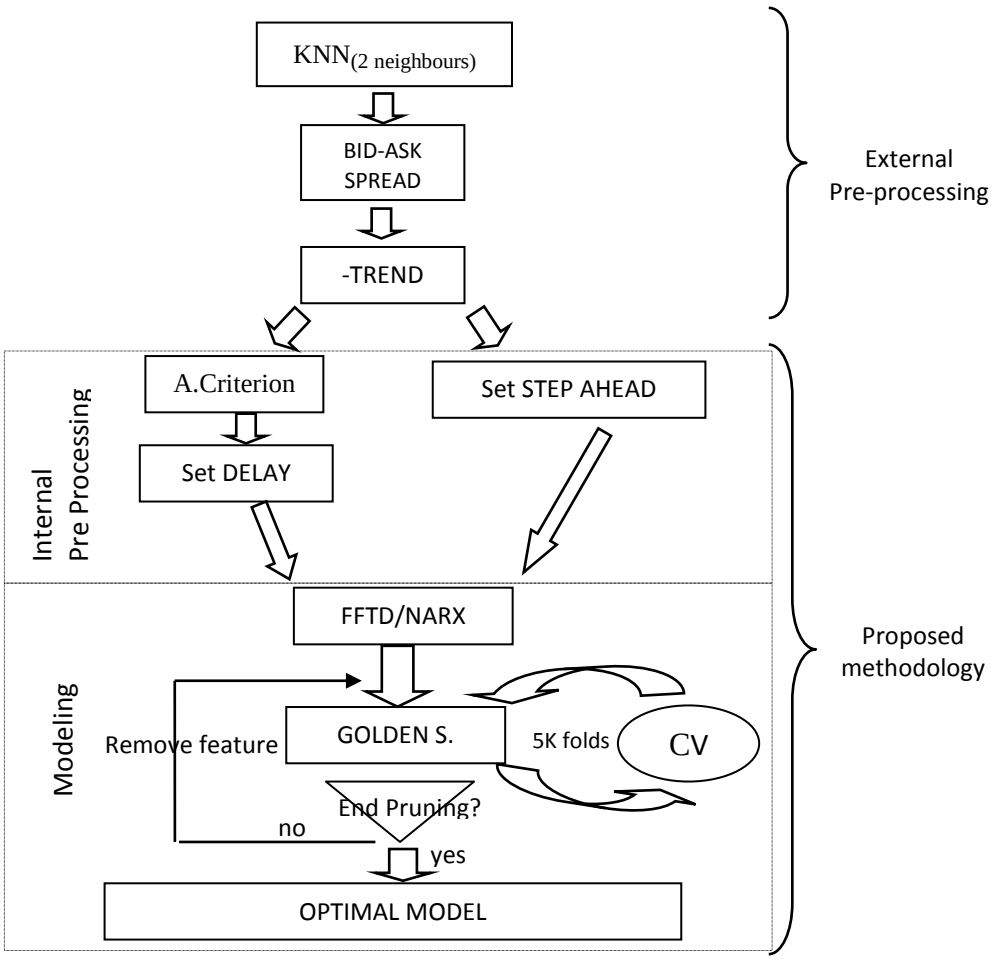
But, this assumption is theoretical so that since neural network training entails a nonlinear optimization problem, and the global optimal solution is not guaranteed by any of the currently available algorithms, a particular training run can be quite sensitive to the initial conditions of the weights in a network, so that with different starting weights may be stuck with different local minima, each of which can have quite different forecasting performance. Therefore in bases of this the algorithm will apply cross validation in order to obtain an average of error and hence an stability which let us chose the optimal model.

Otherwise, talking about the number of layers I have decided use only single hidden layer in bases to several theoretical and experimental works which show that a single hidden layer is sufficient to approximate any complex nonlinear function with any desired accuracy[73].

According to algorithm of training, the most popular is the back-propagation which is a first-order optimization method based on the steepest descent algorithm that requires a step size (or learning rate) to be specified. Because of the instability problem in the steepest descent algorithm, often a more robust training approach such as quasi-Newton, scaled conjugate gradient, Levenberg-Marquardt, and generalized reduced gradient is used. In this study, as I have commented in section learning algorithms (artificial neural networks), I have used the training algorithm, Bayesian regulation backpropagation[74].

[73] Cybenko, G., 1988. Continuous Valued Neural Networks with Two Hidden Layers are Sufficient. Technical Report, Tuft University.
[74] MacKay (Neural Computation, Vol. 4, No. 3, 1992, pp. 415 to 447) and Foresee and Hagan (Proceedings of the International Joint Conference on Neural Networks, June, 1997) .

Thus, summarizing the methodology to follow is:



It is divided by:

- KNN: replace missing values in base of the 2 nearest neighbours.
- Spread Bid-Ask: the target feature from the difference of bid and Ask attributes.
- -TREND: remove the trend to the target attribute.
- A.Criterion: obtain the optimal delay following autocorrelation criterion from target value.
- Set Delay: having set of delays update the dataset adding them.
- Set Step-Ahead: In the case of wanting to do prediction to more than 1 step ahead. The target would be updated.
- FFTD/NARX: to choose between FFTD or NARX networks and fix the networks to use Bayesian regulation back-propagation and initialization function that initializes a layer's weights and biases according to the Nguyen-Widrow initialization algorithm.
- CV: Cross-Validation split the training set in two subsets: 2/3 to training and 1/3 to validation and afterward of training the network, it will be validated over validation set, so repeatedly untill k-folds (by default 5).
- GOLDEN S.: apply golden search to get optimal network. For each step it will do a cross validation of k folds, by default 10, to achieve stability in the results.
- Pruning algorithm: this algorithm as its name say, it will go cutting the set feature by feature keeping what configuration give better performance.

This process has been iterated 35 times, one for each asset, so the result will be 35 different models which will be the optimal model for each time series.

PARAMETER	VALUES	DEFINITION
TIME DELAY	Autocorrelation Criterion	Autocorrelation is used to know how many times ago, *k*, the history is repeated. It means is significant to do predictions *t*+*k* ahead. It is sensible to the curse of dimensionality.
NUMBER OF NEURONS		Golden Search Algorithm
TRANSFER FUNCTION OF iTH LAYER	TANSIG	Hyperbolic tangent sigmoid transfer function.
TRANSFER FUNCTION OF OUTPUT LAYER	PURELIN	Linear transfer function
TRAIN ALGORITHM	TRAINBR	Bayesian regulation backpropagation
EPOCHS	500	
WEIGHTS, BIASES	INITNW	INITNW is a layer initialization function that initializes a layer's weights and biases according to the Nguyen-Widrow initialization algorithm. This algorithm chooses values in order to distribute the active region of each neuron in the layer approximately evenly across the layer's input space. The values contain a degree of randomness, so they are not the same each time this function is called.
LEARNING RATE	0.001	
GOAL	1e-5	
NORMALIZE	-1 to +1	
CROSS-VALIDATION	5 K-Folds	
ERROR VALIDATION	NRMSE	Normalized Root Mean Square Error $\text{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - \bar{t_i})^2}}$

Table 18: configuration of initial parameters

Training stops when any of these conditions occur:

1. The maximum number of EPOCHS (repetitions) is reached.
2. Performance has been minimized to the GOAL.
3. The performance gradient falls below MINGRAD.

In other hand, to get future values of steps ahead can be modelled in different ways, the most simple is use the resulting network for multistep-ahead predictions by feeding the predictions back to the input of the network and continuing to iterate. But it has problems when you use multivariate models so that we should forecast all inputs in order to feed the next predictions. Thus, this intuitively and logically produces a large error along to iterations. Another possibility is to do a set of neural networks where each one predicts one value in a instant *t*, but it

has a high computational cost, so an alternative model is set the target value such that it provide the future values ahead to be predicts formatting it in a multi-output network.

Practical Application

Design of the GUI

Once I have got the optimal models, I have designed a graphical user interface in order that anybody (not necessarily an expert user) can use it. Before to carry on, I would like comment that it is only a prototype. It is due so that the provided models are not enough good even so they have been incorporated to the application to do it more realistic. It is composed by simple windows which let to user introduce an unique asset composed by a serie of features. The asset must be one of 35 previously detailed. Afterward the user only will be able to fill in, into application *config* section, what ISIN is introduced, what columns contains the bid and ask features to calculate the target (bid-ask spread), what column contain the close price, *p_close*, which is ned to calculate the volatility, the field of step-ahead saying how many steps-ahead want to know (by default one)

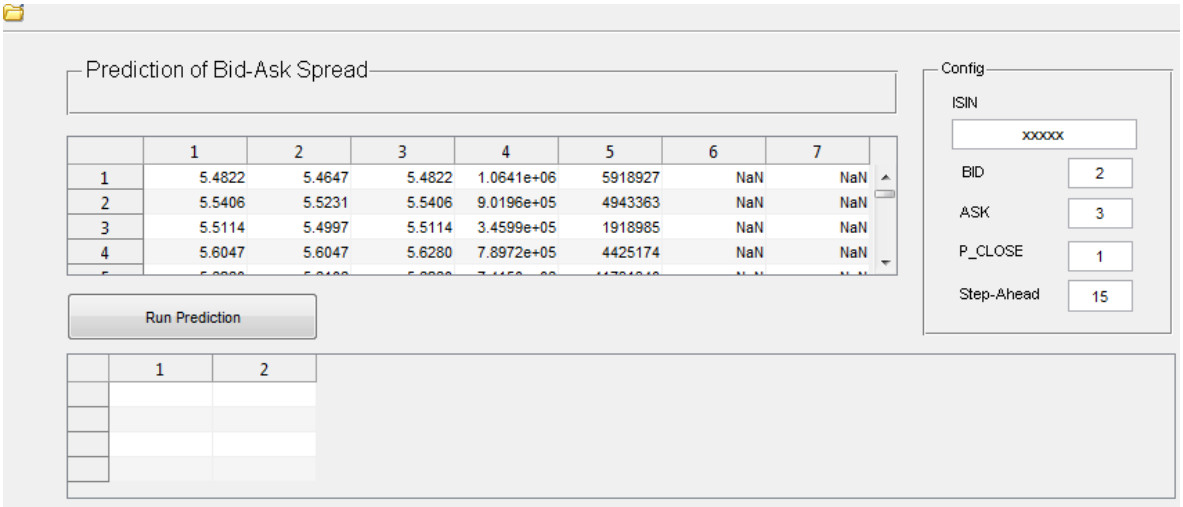


Figure 25: GUI in fill into mode

and endly to click over the *run prediction* button. From this state, the user will have not to do anything beyond to wait for results.

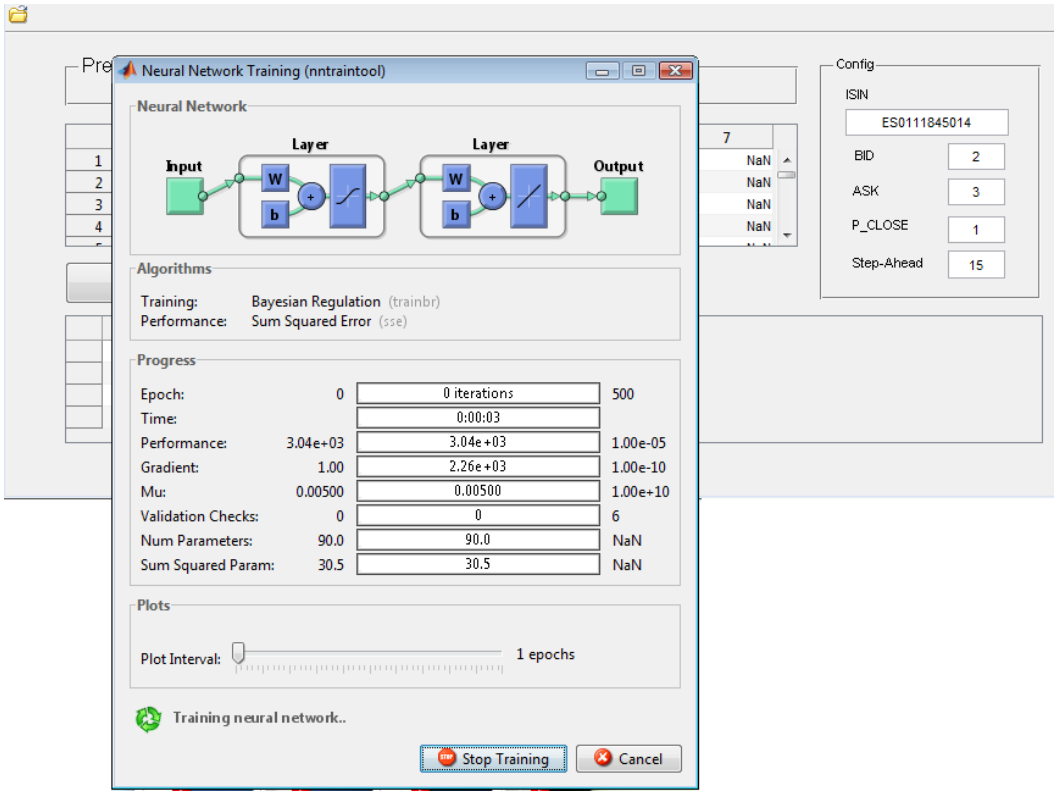
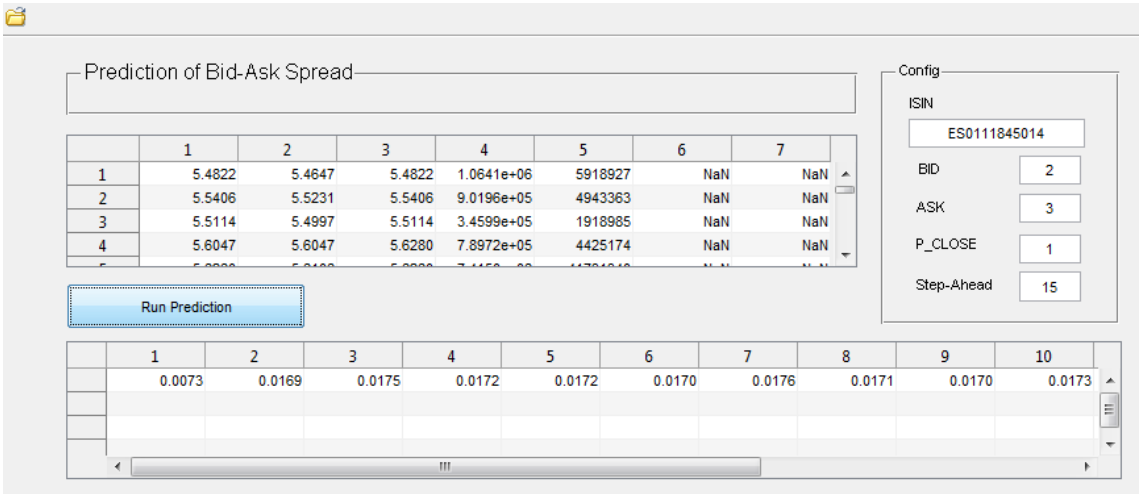


Figure 26: GUI in training-prediction mode

They will be represented in a matrix where each column will provide the prediction to *t+h* where *t* is the current time and *h* the time ahead to predict which will be corresponded to the number of column. So it means, the first column will be the prediction for *t+1*, the second column will be the prediction *n+2* ... being *h*=number of column



	1	2	3	4	5	6	7
1	5.4822	5.4647	5.4822	1.0641e+06	5918927	NaN	NaN
2	5.5406	5.5231	5.5406	9.0196e+05	4943363	NaN	NaN
3	5.5114	5.4997	5.5114	3.4599e+05	1918985	NaN	NaN
4	5.6047	5.6047	5.6280	7.8972e+05	4425174	NaN	NaN

1	2	3	4	5	6	7	8	9	10
0.0073	0.0169	0.0175	0.0172	0.0172	0.0170	0.0176	0.0171	0.0170	0.0173

Figure 27: GUI showing results

Code used

As I have commented while the work, I have developed a series of methods in order to reach my goals. So the next table list each one supplying their descriptions with the purpose to be a little guide.

path	File	Description
/LRM/bd	Set of subsets	In this directory you can find all dataset used to handle this work.
/LRM/	files_config.txt	List of dataset used.
/LRM/	acorr.m	Autocorrelation method. Algorithm based in Bartlett (ro) and Anderson (+-2S(rk)) theories. In addition, it compute the optimal time-delay in bases of ACriterion (Vol. 17 No. 3 Journal of Systems Science and Complexity Jul., 2004).
/LRM/	goldSSN.m	Golden Section Search. Algorithm based in Golden Section Search in order to achieve the optimal number of neurons for a single hidden layer. Number of neurons are increased in base of logarithm.
/LRM/	hTrend.m	To handle Trend in Time Series. Let us remove or add the trend to a time series.
/LRM/	knn.m	K nearest neighbors. Finds the nearest neighbor, near, in X(col) for each point in all X and replace it by the mean. If col=0, in all columns will be handled and replaced their missing values. Otherwise, only it will be hanled the specified column.
/LRM/	pruning.m	Pruning algorithm combing focus delay and recurrent networks in order to select the optimal configuration. Exhaustive seek including golden section search to choose number of neurons. They will be trained with trainbr algorithm (Bayesian Regulation backpropagation), standaridzed to [-1,+1] range and removed fixed rows. In additions if there are NaN's values, they will be replaced by the mean, so I recommend previously treat them.
/LRM/	sAhead.m	Set Step Ahead. In dynamic neural networks are useful for their capability to predict to more than one step ahead. This capability is given by adding steps ahead on target. So, given a dataset and array of delays, sAhead build a new dataset adding step ahead.
/LRM/	sDelay.m	Set Time Delay. In dynamic neural networks are useful for their capability to memorize value on time. This capability is given by delays. So, given a dataset and array of delays, sDelay build a new dataset adding delays.
/LRM/	trainCV.m	Cross-Validation Training. Train a network k times applying cross-validation. Return normalized root mean square.

/LRM/	volatility.m	Volatility method which will be calculated from a vector of prices.
/LRM/	mlr_v1.fig mlr_v1.m	Code which supply the graphical user interfarce explained in this work.
/LRM/	nn.m	Code which supply ANN set to the different models got in modelling process. It is used by mlr_v1.m (GUI).
/LRM/tests/	AAC.m	Test: Autocorrelations. Perfom a serie of test in order to obtain the coeficient of correlation for each time series.
/LRM/tests/	ACapital.m	Test for the analysis of capital feature over target in the subsets said in tesina. For each set will show two result. The first time will be with the capital feature and the second time without.
/LRM/tests/	ANumop.m	Test for the analysis of num_op feature versus target in the subsets said in tesina. For each set will show two result. The first time will be with the num_op feature and the second time without.
/LRM/tests/	ATurnover.m	Test for the analysis of turnover feature versus target in the subsets said in tesina. For each set will show two result. The first time will be with the turnover feature and the second time without.
/LRM/tests/	AVolatility.m	Test for the analysis of volatility feature versus target in the subsets said in tesina. It provides correlation coefficients and their p-value for each subset.
/LRM/tests/	AVolume.m	Test for the analysis of volume feature versus target in the subsets said in tesina. For each set will show two result. It provides correlation coefficients and their p-value for each subset.
/LRM/tests/	AMissing.m	Test to analyze missing values for each dataset.
/LRM/tests/	AOutlier.m	Test to analyze outliers values for each dataset. Previosly will be reemplaced the missing valued using knn.m.
/LRM/tests/	ATrend.m	Test to analyze the trend of target for each dataset. Previosly will be reemplaced the missing valued using knn.m. It shows the target' trend for each subset. Also it has been done for BID and ASK features so that the target is composed by the spread bewteen them.
/LRM/tests/	testPruning.m	Main test. This test let to get the optimal model for given the number of datasets. It has been removed turn_over,capital and num_op features. Replaced missing values and got volatility. It will be applied two kind of ANN: FFTD and NARX.

Evaluation

To evaluate the results I used normalized root mean square error measure:

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - \overline{t_i})^2}}$$

It divides the mean square error by its variance. The result provided should be lower than 1. If it would be near or equal to 1 it would mean that the provided network is giving a mean constant value which is the same set to $y_i = \overline{t_i}$. Hence, I'm willing that the results are lower.

The next tables show which features have been chosen, the errors performance and the number of neurons chosen:

ISIN	Model	ISIN	Model
ES0113860A34	P_CLOSE, BID, ASK, SPREAD,VOLATILITY	ES0130960018	P_CLOSE, ASK,VOLATILITY
ES0113900J37	P_CLOSE, BID, ASK,SPREAD	ES0132105018	P_CLOSE, BID, ASK,VOLUME, SPREAD
ES0124244E34	P_CLOSE, BID , SPREAD,VOLATILITY	ES0140609019	P_CLOSE
ES0144580Y14	P_CLOSE, ASK, VOLUME , SPREAD,VOLATILITY	ES0142090317	P_CLOSE, BID, ASK,VOLUME, SPREAD
ES0105200416	P_CLOSE, BID, ASK,VOLUME, SPREAD,VOLATILITY	ES0143416115	P_CLOSE, ASK, SPREAD,VOLATILITY
ES0111845014	VOLUME , SPREAD, VOLATILITY	ES0147200036	P_CLOSE, BID, ASK,VOLUME
ES0112501012	P_CLOSE, BID, ASK, SPREAD	ES0147645016	P_CLOSE, BID, VOLUME
ES0113211835	P_CLOSE, VOLUME	ES0148396015	BID ,VOLUME,VOLATILITY
ES0113440038	P_CLOSE, VOLUME, SPREAD, VOLATILITY	ES0152503035	VOLATILITY
ES0113679137	P_CLOSE, BID, VOLUME, SPREAD, VOLATILITY	ES0167050915	P_CLOSE, VOLUME, SPREAD, VOLATILITY
ES0113790531	VOLUME, SPREAD, VOLATILITY	ES0171996012	BID,ASK, VOLUME , SPREAD
ES0115056139	VOLUME, VOLATILITY	ES0173093115	P_CLOSE
ES0116870314	P_CLOSE, BID, VOLUME, VOLATILITY	ES0187165017	SPREAD, VOLATILITY
ES0118594417	P_CLOSE, BID, ASK	ES0178430E18	BID, SPREAD
ES0118900010	P_CLOSE, BID, ASK, SPREAD	ES0182870214	P_CLOSE, BID, ASK,VOLUME, SPREAD
ES0122060314	P_CLOSE, BID, ASK, SPREAD	LU0323134006	P_CLOSE, BID, VOLUME , SPREAD, VOLATILITY
ES0125220311	P_CLOSE, BID, ASK,VOLUME, SPREAD	ES0113900J37	BID, VOLUME , SPREAD
ES0130670112	P_CLOSE, BID, ASK,VOLUME, SPREAD		

Table 19: ISIN with its optimal input set

ISIN	#Neuron	Error	Time(h)	ISIN	#Neuron	Error	Time(h)
ES0113860A34	2	1,0076	2,165585	ES0130960018	16	1	3,93075833
ES0113900J37	2	1,3878	4,62526167	ES0132105018	2	1,0331	2,79420167
ES0124244E34	2	1,0033	2,94287333	ES0140609019	32	1,048	13,5484483
ES0144580Y14	2	1,0295	2,06323667	ES0142090317	3	1,0018	2,99165667
ES0105200416	3	1,0525	2,744095	ES0143416115	3	1,0316	2,76252667
ES0111845014	3	0,99541	3,47842833	ES0147200036	2	1,1761	2,17495833
ES0112501012	4	1,0295	3,38707667	ES0147645016	4	1,1144	1,83574333
ES0113211835	4	1,045	4,65323333	ES0148396015	3	1,0748	2,20711167
ES0113440038	2	0,98216	2,61702667	ES0152503035	8	0,99682	3,583945
ES0113679137	2	1,0926	3,94475667	ES0167050915	2	0,99716	4,03637833
ES0113790531	2	1,033	2,345905	ES0171996012	32	1,1922	22,77569
ES0115056139	18	1,0301	3,24745667	ES0173093115	8	1,0319	6,52880667
ES0116870314	2	1,0342	3,51227333	ES0187165017	3	1,0285	6,81027333
ES0118594417	8	1,0657	3,34106167	ES0178430E18	4	1,313	4,54491333
ES0118900010	2	1,0062	1,565925	ES0182870214	3	1,0509	5,140915
ES0122060314	2	1,026	2,72828833	LU0323134006	4	1,2748	2,24624667
ES0125220311	2	1,0072	2,60640333	ES0113900J37	2	1,3967	3,16109
ES0130670112	2	1,8837	6,14394833				

Table 20: results of cross-validation from optimal model for each asset

Thus the ranking is in base of configuration and inputs

Feature	Proportion	%
P_CLOSE, BID, ASK,VOLUME, SPREAD	5	14,28
P_CLOSE, BID, ASK,SPREAD	4	11,42
VOLUME , SPREAD, VOLATILITY	2	5,71
P_CLOSE, VOLUME, SPREAD, VOLATILITY	2	5,71
P_CLOSE, BID, VOLUME, SPREAD, VOLATILITY	2	5,71
P_CLOSE	2	5,71
VOLATILITY	2	5,71
P_CLOSE, BID, ASK, SPREAD,VOLATILITY	1	2,85
P_CLOSE, BID , SPREAD,VOLATILITY	1	2,85
P_CLOSE, ASK, VOLUME , SPREAD,VOLATILITY	1	2,85
P_CLOSE, BID, ASK, VOLUME , SPREAD,VOLATILITY	1	2,85
VOLUME, VOLATILITY	1	2,85
P_CLOSE, BID, VOLUME, VOLATILITY	1	2,85
P_CLOSE, BID, ASK	1	2,85
P_CLOSE, ASK,VOLATILITY	1	2,85
P_CLOSE, ASK, SPREAD,VOLATILITY	1	2,85
P_CLOSE, BID, ASK,VOLUME	1	2,85
P_CLOSE, BID, VOLUME	1	2,85
BID ,VOLUME,VOLATILITY	1	2,85
BID,ASK, VOLUME , SPREAD	1	2,85

SPREAD, VOLATILITY	1	2,85
BID, SPREAD	1	2,85
BID, VOLUME , SPREAD	1	2,85
TOTAL	35	99,85(~100)

Table 21: Ranking of set of features

The table 21 shows that the best model is composed by { P_CLOSE, BID, ASK,VOLUME, SPREAD} but only 5 models of 35 have used it. Thus, may we are interested how each feature works. The next table shows the proportion by each feature giving a significance order in the prediction of liquidity risk.

Feature	Proportion	%
P_CLOSE	26/35	74
SPREAD	24/35	68,57
BID	20/35	57,14
ASK	17/35	48,57
VOLATILITY	17/35	48,57
VOLUME	15/35	42,85

Table 22: Ranking by feature in base of its usability

Following the economic model the best features should be *volume* and *volatiliy* but, however as we notice, the most significance features are *p_close* and the target (*spread)* with 74% and 68,57% respectively. On the other hand, the worst features are *volume*, *volatility* and *ask* with 42,85%, 48,57% and 48,57%. Thus, this open again the discussion about liquidity risk modeling. In addition I supply the correlation coefficient between volatility-spread and volume-spread for each asset in order to evaluate their lineal correlation[75].

The next table show the ranking of the number of neurons.

# Neuron	Proportion	%
2	16	45,71
3	7	20
4	5	14,28
8	3	8,57
32	2	5,71
16	1	2,85
18	1	2,85
TOTAL	35	99,97(~100)

Table 23: Ranking of number of neurons

[75] The results show as for almost of asset do not exist lineal correlation neither between volatility-spread nor volume-spread. May it is due to the nature of bid-ask spread such as it is showed in *Trend detail* (Appendix).You can check correlation coefficient in *Correlation Coefficient* (Appendix).

Regarding to number of neurons, as the table 23 shows, the better number of neurons is 2 with a 45,71% got. However we can notice that the worst are 32, 16 and 18 which have not reached more than 5,71% and 2,85%.

As I commented above the performance error should be lower than 1 so that otherwise that would mean that the network give the same result as a mean value. Unfortunatelly, all of them are higher than 1, therefore none model forecast well. In order to understand this phenomen I have plotted the first 5 cross validation of the asset ISIN ES0113860A34. The next figures ilustrate them:

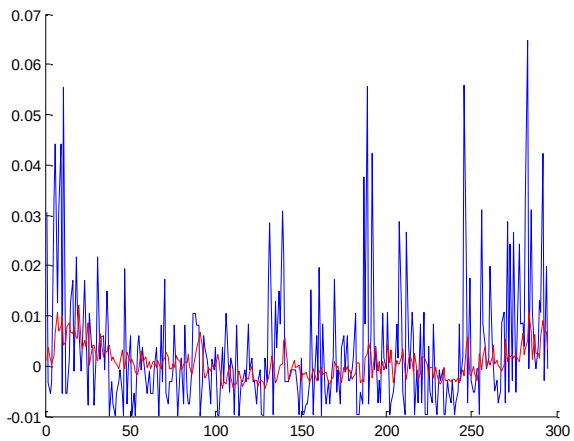


Figure 28: 1rst iteration cv (blue = target, red = prediction)

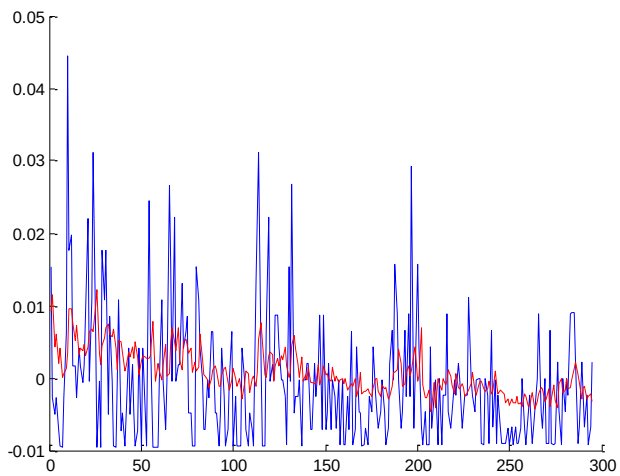


Figure 29: 2nd iteration cv (blue = target, red = prediction)

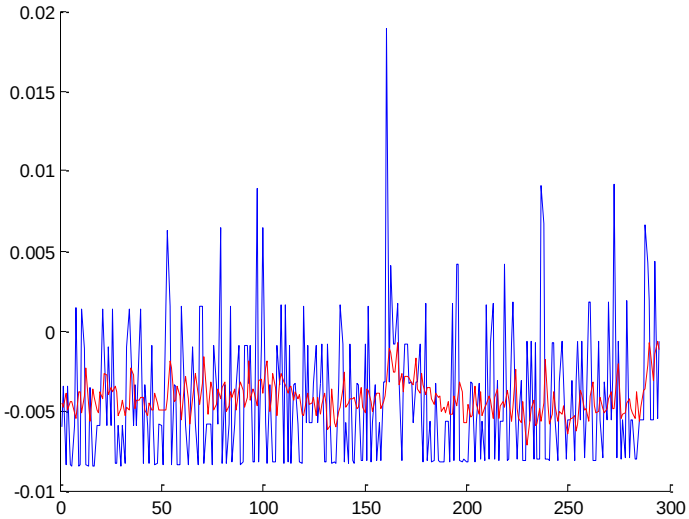


Figure 30: 3rd iteration cv (blue = target, red = prediction)

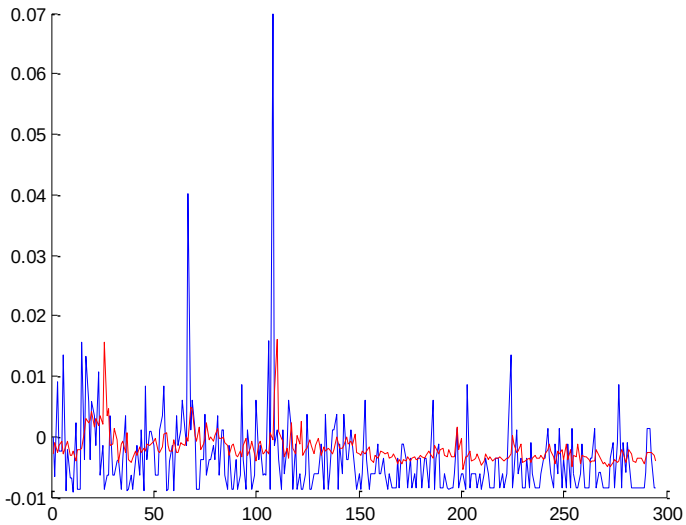


Figure 31: 4th iteration cv (blue = target, red = prediction)

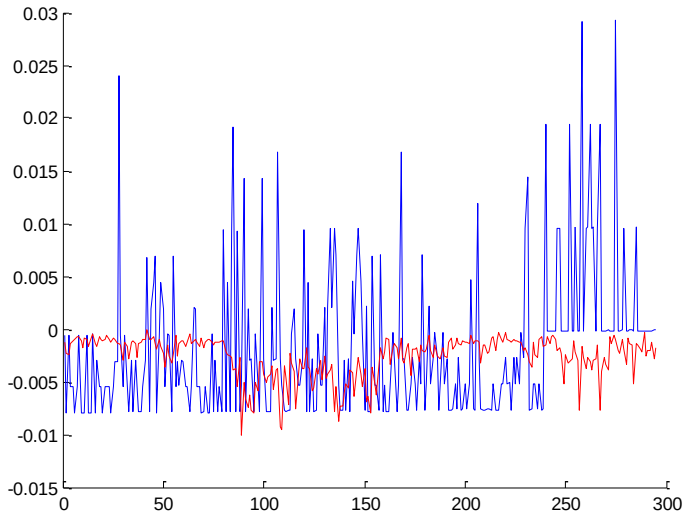


Figure 32: 5th iteration cv (blue = target, red = prediction)

Surely, checking the figures we can notice the prediction almost have the same behaviour of mean value. Therefore from these results we can learn that the method employed is not enough to treat this kind of ailment.

The table 17 only shows the final results, but how it works? So, the next table explains all process, step by step, taken on asset ES0113900J37. The labels of headers are understandable except *n.feat* which means number of feature and *r.feat* removed features.

ISIN					
#N.	**Error**	**Time(min)**			
n.feat.	**r.feat.**	**delay**	**#N.**	**Error**	**Time(min)**
ES0113900J37					
2	1,4181	6,5598			
4	1,474	9,9087			
3	1,424	11,3436			
0		**1**	**2**	**1,4181**	**27,8158**
2	1,4054	5,3077			
4	1,4181	11,3964			
3	1,4174	11,166			
6	**6**	**1**	**2**	**1,4054**	**27,8743**
2	1,4208	7,618			
4	1,4249	11,4125			
3	1,4227	10,9905			
5	**6**	**1**	**2**	**1,4208**	**30,0255**
2	1,3878	7,227			
4	1,4209	11,2693			

3	1,4832	10,4825				
4	**6 4**	**1**	**2**	**1,3878**	**28,9834**
2	1,4286	5,5404				
4	1,3956	9,8741				
8	1,4048	11,7984				
6	1,4046	11,4814				
5	1,4047	11,418				
3	**6 4**	**1**	**4**	**1,3956**	**50,1177**
2	1,5069	6,6805				
4	1,4589	10,6614				
8	1,3906	11,8147				
16	1,3904	13,5864				
32	1,4391	21,1657				
24	1,3903	15,9667				
28	1,4016	17,6445				
26	1,4032	16,8744				
25	1,3947	16,3369				
2	**6 4**	**1**	**24**	**1,3903**	**130,7372**
2	1,3882	10,768				
4	1,395	11,1473				
3	1,7538	10,1601				
1	**6 4**	**1**	**2**	**1,3882**	**32,0795**

Table 24: All process for asset ES0113900737

As we can realize, after the first cross-validation (it was 5 kfold) the method show the mean result, which would be interpreted: for 2 neurons we get an error of 1,4181 and it has spent 6,5598 minutes. Now, the number of neuron is increment logarithmically, hence, 4 neurons will be tested getting 1,474 as error performance and 9,9087 minutes. Once we have two results, and following the assumption commented above, in *Modeling* section, the error are compared. If the last one, 4 neurons, is lower than the first the algorithm will carry on finding, otherwise it will start to find the optimal number of neurons between this range. In this case is evaluated 3 neurons getting 1,424 as error performance and 11,3436 minutes. Now, finalizing the search, the lowest value will be taken. In the current case it was 2 neurons with 1,4181 error performance and the total time to get this solution was 28 minutes approximately.

Regarding to the data showed in previous table, may the reader find rows like this, but to understand it, you must see the 3rd row header which say what is each column. For instances, having:

0	1	2	1,4181	27,8158

Table 25: 3rd row from table 24

- 0 shows that has been used all dataset for test the network
- 1 as the number of delays used
- Between the 1 and 2 there is an blank space. Here will be filled up with removed features.

- 2 as the number of neurons which has been selected considering them like the optimal configuration.
- 1.4181 like the error performance.
- and 27,8158 as the total time spent in getting the optimal configuration. The reader can notice that if sum each individual time don't give the same total time, but it's due to the total time has been computed using a individual timer. However they should be very close.

So let's go with the analysis. The next step will be remove the feature 6th and to evaluate its error performance with the previous. If the new error performance is lower than the previous it will mean that the feature doesn't have significant information so it will be removed permanently. Seeing the table we can notice that were got 2 neurons, 1,4054 as error performance and 27,87 as time, this error was lower than the previous with was 1,4181 therefore the feature 6 was removed.

Now all process go iterating, so I would like to go until 6th step where we find a sample of as really works the golden search algorithm. The important range to analysis is as follows

16	**1,3904**	**13,5864**
32	1,4391	21,1657
24	1,3903	15,9667
28	1,4016	17,6445
26	1,4032	16,8744
25	1,3947	16,3369

Table 26: a piece of table 24

Having the range [16, 32] begin the golden section search. For that was taken the midpoint between them and evaluated its result and cutting the range until to find out the minimus between them. So the middle between 16 and 32 is 24 which gave 1,3903 error performance lower than 16. Hence the new range would be [24, 32]. The midpoint is 28 and its error performance was 1,4016 which is upper to 1,3903 from 24 thererfore, the new range was [24,28]. So to find the minimum error. Notice that the final result was 24 neurons for 1,3903 error performance.

At the end, the NARX network has been omitted due to they spent many hours computing. The next table shows a some experiments reached:

ISIN	**#N.**	**delay**	**Error**	**Time(min)**	**CV**
ES0113900J37	2	1	1,2141	935,094	1
ES0113900J37	4	1	1,1270	1770,188	1
....	.....	....	....	....	
ES0113860A34	2	1	1,0230	392,071404	1
ES0113860A34	4	1	1,0556	1268,28562	1
....	.....	....	....	....	

Table 27: some results from NARX networks

To work with recurrent neural networks has been imposible. As the tabla 21 shows, for the asset ES0113900J37, 2 neurons and only 1 iteration in cross-validation, the network spent almost of 15 hours to achieve the aim. Looking at asset ES0113860A34 we notice a same behaviour. Therefore, if I would have done the same test as FFTD I would need aprox. : 15hx5(kfolds)x4(mean of iterations)x6(number of features)=1800h, which is the same 75 days or probably more so that as is increased the number of neurons the time to spent is also increased.

Conclusions & Future work

This work has attempted to model the behaviour of liquidity risk in base of bid-ask spread. Firstly have been identified 35 subsets (separeted by ISIN) from a initial large dataset (provided by financial experts and composed by attributes following economic and financial theories). Afterwards I have perfomanced an exhaustive analysis of each one and haircut them in order to get better initial configurations (removing insignificant features and adding others). Once As I have had the subsets already to be used, I have developed a set of methods so that they would do more easy and dynamic the future work like: volatility algorithm, k-neighbours algorithm, autocorrelation algorithm, etc…

Finalizing the pre-processing and moving on the modelling I have decided to configure the neural networks following some theories and using already algorithms implemented by the framework (Matlab) used like: initialitation weigths and bias, training algorithm, error of training… But, however I have performed few adaptions satisfying some needs which Maltab R2010a Student did not provide. For instances, the cross-validation algorithm has been done by myself choosing 5 or 10 k folds depending of the case (it has been explained in *modelling* section) returning a mean of NRMSE measure of k times. Another aspect to denote it has been used an dynamic algorithm, based in golden section search, with the purpose of providing the best number of hidden neurons. Well, afterward of being aware of all previous consideration, I have developed an pruning algorithm which allows me to test all subsets and to keep the features which got the lowest error in cross-validation.

Unfortunately, I could only use the FTDNN due to I had not enough resource to face up to the challenge that the NARX network provided spending or wasting… many hours to get results. For illustrating it, the next table shows the timing of only 1 iteration of cross validation for an asset.

ISIN	#N.	delay	Error	Time(min)
ES0113900J37	2	1	1,2141	935,094

Regarding to FTDNN results they have not been satisfactories. Really I'm completely clueless about what is failing. Some considerations to take into account would be:

- liquidity risk modelling is a new issue and it has not a significative model yet, so the model provided in this work could be improved.
- tunning a neural network is a hard work and it increases if it handle a time series so that there is to take into account the nature of time series (trend, seasonality, delays,...).
- the dataset used supply many missing and outlier values in some features so it could make noise. Another consideration referencing at first point, the model proposed may not provide enough information to predict accurately.

Nevertheless this conclusion opens the door to future works. As is well-know the neural networks are able to predict time series better than classical techniques but it is not well-defined as the input previously due to be treated. So, in this work I have omitted study the seasonality component which may be an important element in the prediction of bid-ask spread time series. Trying to save my position about this issue, I decided to avoid it because for each time serie should be carefully analyzed the seasonality and one of my pretension along this work it has been to develop algorithms which allow me obtain dynamic results like: autocorrelation criterium, detrend and volatility algorithm which don't need an human advice. In other words, one of the aims of this work was to develop a graphical interface in order to get a useful tool. Hence the time series to be included would be different from the initials, then I was interested in methods capable of working without supervision of a human giving a dynamical configuration. However, all methods to analyze the seasonality are statics and depend of a subjective opinion.

Another interesting thing to work would be use other kind of networks like support vector machines or recurrent networks which do not waste so time (may avoiding Matlab platform and going to C or Python). Also It would be interesting to apply bagging or boosting methods so that minimize the variance.

Regarding the input, an alternative way do not explorated here, would be to do the prediction on bid and ask features, so that they show a smoothed behaviour, and afterward to get the spread. It could be done with a network with two outputs (Bid and ASk).

On the other hand, the graphical user interface also could be improved. Now is designed for treating only one asset so in order to improve it, it could treat multi-input assets identifying each one by their ISIN. Other aspect would be also include an multi-output where each output would have associated a row .

Thus, summarizing even though I have got poor perfomance, from my point of view I'm glad to have chosen and developed this work so that it has improved my knowledge about a useful and suitable technique as are the neural networks and also it has introduced me to the wonderful world of risk. So in next studies I'll try to reach the new goals flourished here.

References

1. *A Practical Bayesian Framework for Backpropagation Networks.* David J. C. MacKay. White paper.

2. *Annual report of the SEC for the fiscal year ended June 30, 1975.*

3. *Asset Market Liquidity Risk Management: A Generalized Theoretical Modeling Approach for Trading and Fund Management Portfolios*. Al Janabi, Mazin A.M. – UAE University (May 20, 2009). White paper.

4. *Crisis de liquidez en los mercados financieros internacionales: impacto sobre la bolsa española*. David Cano y David Fernández (2009).

5. FeedForward Neural Networks: An Introduction. Simon Haykin. White paper.

6. *Forecasting with artificial neural networks: The state of the art.* Guoqiang Zhang, B. Eddy Patuwo, Michael Y. Hu*.(31 July 1997). White paper.

7. *Measuring Market Liquidity Risk – Which Model Works Best?* Cornelia Ernst, Sebastian Stange, Cristoph Kaserer (February 3, 2009). White paper.

8. *Improved option pricing using artificial neural networks and bootstrap methods*. Paul r. Lajbcygier, Jerome t. Connor. International Journal of Neural Systems, Vol. 8, No. 4 (August, 1997) 457-471.

9. *International framework for liquidity risk measurement, standards and monitoring.* Basel Committee on Banking Supervision (December 2009).

10. *Modeling Liquidity Risk with Implications for Traditional Market Risk Measurement and Management*. University of Pennsylvania, Stern School, NYU and Oliver, Wyman & Company (1998).

11. *Neural Networks: A comprehensive foundation.* Simon Haykin (1999). ISBN: 0-13-908385-5.

12. *Neural Networks in Finance and Investing*. Robert R. Trippi and Efraim Turban (1996). ISBN: 1-55738-919-5.

13. *Neural modeling for time series: a statistical stepwise method for weight elimination.* Marie Cottrell, Bernard Girard, Yvonne Girard, Morgan Mangeas and Corinne Muller. IEEE transactions on neural networks, vol.6, n.6, (November 1995).

14. *Risk budgeting: Portfolio Problem Solving with Value-at-Risk*. Neil D. Pearson (2002). ISBN:0-471-40556-6.

15. *Risk Management: value at risk and beyond*. M.A.H Dempster, University of Cambridge (2002). ISBN: 0-521-78180-9.

16. *Recurrent Neural Networks and Robust Time Series Prediction.* Jerome T. Connor, R. Douglas Martin, Member, IEEE, and L. E. Atlas, Member IEEE (MARCH 1994). White paper.

17. *Time Series Forecasting.* Chris Chatfield (2001). ISBN:1-58488-063-5.

18. *Time Series Analysis and its Applications.* Robert H.Sumway & David S.Stoffer (2000). ISBN: 0-387-98950-1.

19. *Time Series Analysis: Univariate and Multivariate Methods.* Wei, William W. S. (1990).ISBN: 0-201-15911-2.

20. *Time Series.* Sir Maurice Kendall (1976). ISBN: 0-85264-241-5.

21. *Using neural networks to forecast the S &P 100 implied volatility*. Mary Malliaris, Linda Salchenberger (1993). Neurocomputing 10 (1996), pp. 183-195.

22. *Value-at-Risk: Theory and Practice*. Glyn A. Holton (2004). ISBN:0-12-354010-0.

Appendix

List of Acronyms

- AC – Autocorrelation Criterion
- ACF – AutoCorrelation Function
- ADALINE - ADAptive LINear Element
- AIC - Akaike Information Criterion
- ANN – Artificial Neural Network
- AR - Autoregressive
- ART – Adaptative Resonance Theory
- ARIMA – Autoregressive Integrated Moving Average
- ARMA - Autoregressive Moving Average
- BIC - Bayesian Information Criterion
- FFTD – Feed Forward Time Delay
- FDNN – Feed Forward Delay Neural Network
- FTDNN- Focused Time Delay Neural Network
- GARCH - Generalized AutoRegressive Conditionally Heteroscedastic
- GUI: Graphical User Interface
- HQC - Hannan-Quinn Criterion
- I - Integrated
- LMS - Least Mean Square
- MA – Moving Average
- MSE – Mean Square Error
- NARX - Nonlinear AutoRegressive Network with exogenous inputs
- NRMSE – Normalized Root Mean Square Error
- NYSE - New York Stock Exchange
- RMSE – Root Mean Square Error
- RNN – Recurrent Neural Network
- SEC - Securities and Exchange Commission.
- SIC - Schwarz Information
- STD – Standard Deviation
- S&P 100 – Standard and Poor's Index.
- TLFN – Time Lagged FeedForward Network
- TRAINLM – Training function Levenberg-Marquardt algorithm
- TRAINRB – Training function Resilient Back-propagation algorithm
- TRAINBFG - Training function BFGS quasi-Newton method.
- TANSIG - Hyperbolic tangent sigmoid transfer function
- UNCR - Uniform Net Capital Rule

- VaR – Value at Risk
- VLSI - very large scale integrated

Missing detail

ISIN: ES0113860A34 Size: 2324	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0430
	ASK	2	0.0861
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	5	0.2151
	NUM_OP	1533	65.9639

ISIN: ES0113900J37 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	1857	70.1284
	CAPITAL	0	0
	NUM_OP	108	4.0785

ISIN: ES0124244E34 Size=2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	3	0.1133
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	108	4.0785
	NUM_OP	1857	70.1284

ISIN: ES0144580Y14 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	1	0.0378
	VOLUME	3	0.1133
	TURN_OVER	1	0.0378
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0105200416 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	1	0.0378
	BID	1	0.0378
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	1	0.0378
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0111845014 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	0	0
	VOLUME	3	0.1133
	TURN_OVER	2	0.0755
	CAPITAL	106	4.0030
	NUM_OP	1857	70.1284

ISIN: ES0112501012 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	1	0.0378
	BID	2	0.0755
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	1	0.0378
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0113211835 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	105	3.9653
	NUM_OP	1857	70.1284

ISIN: ES0113440038 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	2	0.0755
	VOLUME	3	0.1133
	TURN_OVER	1	0.0378
	CAPITAL	108	4.0785
	NUM_OP	1875	70.1284

ISIN: ES0113679137 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	3	0.1133
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	108	4.0785
	NUM_OP	1875	70.1284

ISIN: ES0113790531 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	1	0.03878
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	108	4.0785
	NUM_OP	1875	70.1284

ISIN: ES0115056139 Size:1006	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0994
	ASK	1	0.0994
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	0	0
	NUM_OP	215	21.3718

ISIN: ES0116870314 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	1	0.0378
	VOLUME	3	0.1133
	TURN_OVER	1	0.0378
	CAPITAL	108	4.0785
	NUM_OP	1857	70.1284

ISIN: ES0118594417 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0118900010 Size:1440	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0694
	ASK	1	0.0694
	VOLUME	1	0.0694
	TURN_OVER	0	0
	CAPITAL	2	0.1389
	NUM_OP	649	45.0694

ISIN: ES0122060314 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	107	4.0408
	NUM_OP	1857	70.1284

ISIN: ES0125220311 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0130670112 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	2	0.0755
	VOLUME	4	0.1511
	TURN_OVER	1	0.0378
	CAPITAL	108	4.0785
	NUM_OP	1857	70.1284

ISIN: ES0130960018 Size:2026	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0494
	ASK	1	0.0494
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	4	0.1974
	NUM_OP	1235	60.9576

ISIN: ES0132105018 Size:2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	3	0.1133
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	1	0.0378
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0140609019 Size:689	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.1451
	ASK	1	0.1451
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	1	0.1451
	NUM_OP	0	0

ISIN: ES0142090317 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	0	0
	CAPITAL	108	4.0785
	NUM_OP	1857	70.1284

ISIN: ES0143416115 Size: 2437	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0821
	ASK	2	0.0821
	VOLUME	1	0.0410
	TURN_OVER	0	0
	CAPITAL	5	0.2052
	NUM_OP	1646	67.5421

ISIN: ES0147200036 Size: 2333	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0429
	ASK	1	0.0429
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	5	0.2143
	NUM_OP	1542	66.0952

ISIN: ES0147645016 Size: 643	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.1555
	ASK	1	0.1555
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	1	0.1555
	NUM_OP	0	0

ISIN: ES0148396015 Size: 2300	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0870
	ASK	1	0.0435
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	4	0.1739
	NUM_OP	1509	65.6087

ISIN: ES0152503035 Size: 1527	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.1310
	ASK	1	0.0655
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	2	0.1310
	NUM_OP	736	48.1991

ISIN: ES0167050915 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	3	0.1133
	VOLUME	3	0.1133
	TURN_OVER	2	0.0755
	CAPITAL	109	4.1163
	NUM_OP	1857	70.1284

ISIN: ES0171996012 Size: 1048	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0954
	ASK	1	0.0954
	VOLUME	0	0
	TURN_OVER	0	0
	CAPITAL	0	0
	NUM_OP	257	24.5229

ISIN: ES0173093115 Size: 5296	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	3	0.0566
	ASK	3	0.0566
	VOLUME	4	0.0755
	TURN_OVER	3714	70.1284
	CAPITAL	0	0
	NUM_OP	216	4.0785

ISIN: ES0187165017 Size: 1023	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0978
	ASK	1	0.0978
	VOLUME	0	0
	TURN_OVER	232	22.6784
	CAPITAL	0	0
	NUM_OP	0	0

ISIN: ES0178430E18 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	1	0.0378
	ASK	1	0.0378
	VOLUME	2	0.0755
	TURN_OVER	1857	70.1284
	CAPITAL	0	0
	NUM_OP	108	4.0785

ISIN: ES0182870214 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	1	0.0378
	VOLUME	3	0.1133
	TURN_OVER	1857	70.1284
	CAPITAL	1	0.0378
	NUM_OP	108	4.0785

ISIN: LU0323134006 Size: 996	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.2008
	ASK	3	0.3012
	VOLUME	1	0.1004
	TURN_OVER	206	20.6827
	CAPITAL	1	0.1004
	NUM_OP	0	0

ISIN: ES0113900J37 Size: 2648	Attribute	Number of missing	%
	P_CLOSE	0	0
	BID	2	0.0755
	ASK	2	0.0755
	VOLUME	2	0.0755
	TURN_OVER	1857	70.1284
	CAPITAL	0	0
	NUM_OP	108	4.0785

Outliers detail

ES0113860A34	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	25	27	0	49
%	0	0	0	1.0757	1.1618	0	2.1084

ES0113900J37	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	42	55	56	0
%	0	0	0	1.5861	2.0770	2.1148	0

ES0124244E34	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	35	45	0	49
%	0	0	0	1.3218	1.6994	0	1.8505

ES0144580Y14	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	48	59	0	64
%	0	0	0	1.8127	1.8882	0	2.4169

ES0105200416	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	1	1	1	49	54	1	47
%	0.0378	0.0378	0.0378	1.8505	2.0393	0.0378	1.7749

ES0111845014	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	40	13	0	50
%	0	0	0	1.5106	0.4909	0	1.8882

ES0112501012	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	39	14	0	50
%	0	0	0	1.4728	0.5287	0	1.8882

ES0113211835	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	41	54	0	43
%	0	0	0	1.5483	2.0393	0	1.6239

ES0113440038	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	31	50	0	40
%	0	0	0	1.1707	1.8882	0	1.5106

ES0113679137	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP
N. Outliers	0	0	0	51	27	0	50
%	0	0	0	1.9260	1.0196	0	1.8882

ES0113790531	P_CLOSE	BID	ASK	VOLUME	TURN_OVER	CAPITAL	NUM_OP

N. Outliers	0	0	0	47	58	0	52
%	0	0	0	1.7749	2.1903	0	1.9637

ES0115056139	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	7	10	0	12
%	0	0	0	0.6958	0.9940	0	1.1928

ES0116870314	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	39	21	0	41
%	0	0	0	1.4728	0.7931	0	1.5483

ES0118594417	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	46	22	0	39
%	0	0	0	1.7372	0.8308	0	1.4728

ES0118900010	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	14	11	0	23
%	0	0	0	0.9722	0.7639	0	1.5972

ES0122060314	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	13	13	0	36
%	0	0	0	0.4909	0.4909	0	1.3595

ES0125220311	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	7	7	7	40	51	8	37
%	0.2644	0.2644	0.2644	1.5106	1.9260	0.3021	1.3973

ES0130670112	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	29	21	0	0
%	0	0	0	1.0952	0.7931	0	0

ES0130960018	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	22	35	0	35
%	0	0	0	1.0859	1.7275	0	1.7275

ES0132105018	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	6	6	6	42	45	6	33
%	0.2266	0.2266	0.2266	1.5861	1.6994	0.2266	1.2462

ES0140609019	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	6	11	0	3
%	0	0	0	0.8708	1.5965	0	0.4354

ES0142090317	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**

N. Outliers	0	0	0	27	26	0	57
%	0	0	0	1.0196	0.9819	0	2.1526

ES0143416115	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	13	16	0	35
%	0	0	0	0.5334	0.6565	0	1.4362

ES0147200036	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	2	2	2	39	30	1	32
%	0.0857	0.0857	0.0857	1.6717	1.2859	0.0429	1.3716

ES0147645016	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	4	4	4	3	5	4	7
%	0.6221	0.6221	0.6221	0.4666	0.7776	0.6221	1.0886

ES0148396015	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	16	43	0	59
%	0	0	0	0.6957	1.8696	0	2.5652

ES0152503035	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	8	12	0	24
%	0	0	0	0.5239	0.7859	0	1.5717

ES0167050915	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	27	44	0	18
%	0	0	0	1.0196	1.6616	0	0.6798

ES0171996012	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	5	13	0	16
%	0	0	0	0.4771	1.2405	0	1.5267

ES0173093115	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	60	57	90	0
%	0	0	0	1.1329	1.0763	1.6994	0

ES0187165017	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	7	17	7	0
%	0	0	0	0.6843	1.6618	0.6843	0

ES0178430E18	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	6	6	6	34	13	58	0
%	0.2266	0.2266	0.2266	1.2840	0.4909	2.1903	0

ES0182870214	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**

N. Outliers	43	44	42	37	38	46	16
%	1.6239	1.6616	1.5861	1.3973	1.8127	1.7372	0.6042

LU0323134006	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	19	18	16	0
%	0	0	0	1.9076	1.8072	1.6064	0

ES0113900J37	**P_CLOSE**	**BID**	**ASK**	**VOLUME**	**TURN_OVER**	**CAPITAL**	**NUM_OP**
N. Outliers	0	0	0	45	55	56	0
%	0	0	0	1.5861	2.0770	2.1148	0

Autocorrelations detail

ISIN: ID of Asset.

K: Coefficient of autocorrelation in base of +-2S(rk) criterion.

LAC: Length of values of Autocorrelation Criterion.

AC: Values of Autocorrelation Criterion.

Mean: Accumulative Mean.

ISIN	**K**	**LAC**	**AC**	**Mean**
ES0113860A34	138	22	1 3 4 8 10 12 15 32 38 51 58 77 78 83 104 125 132 133 134 136 137 138	138
ES0113900J37	127	22	1 2 6 7 24 42 44 45 71 72 73 101 106 109 110 113 115 116 121 123 126 127	133
ES0124244E34	137	13	1 3 5 7 8 10 12 39 44 51 122 126 137	134
ES0144580Y14	143	12	1 6 39 74 119 124 125 131 132 137 142 143	136
ES0105200416	50	14	1 3 9 16 18 23 24 25 30 42 46 47 49 50	119
ES0111845014	37	6	1 2 24 31 36 37	105
ES0112501012	138	16	1 3 5 14 18 25 39 69 73 77 80 82 93 114 128 138	110
ES0113211835	178	12	1 14 56 87 100 106 142 146 175 176 177 178	119
ES0113440038	153	35	1 2 3 4 5 6 7 8 9 13 16 18 20 32 34 35 36 37 38 41 45 50 51 52 104 105 124 125 136 137 142 147 150 152 153	122
ES0113679137	87	14	1 2 4 9 15 33 44 46 63 71 72 74 85 87	119
ES0113790531	104	10	1 15 20 75 88 93 95 97 99 104	117
ES0115056139	22	8	1 3 6 7 18 19 21 22	110
ES0116870314	131	17	1 2 4 5 6 13 17 20 39 72 111 112 127 128 129 130 131	111
ES0118594417	79	7	1 21 32 59 71 78 79	109
ES0118900010	2	2	1 2	102

ES0122060314	150	12	1 8 16 39 45 57 111 114 132 145 149 150	105
ES0125220311	40	10	1 2 4 5 17 27 31 34 35 40	101
ES0130670112	78	13	1 20 35 39 43 44 48 54 57 62 72 76 78	100
ES0130960018	21	4	1 11 20 21	96
ES0132105018	95	11	1 2 4 9 12 63 66 76 77 94 95	96
ES0140609019	64	7	1 20 41 51 57 63 64	94
ES0142090317	144	17	1 2 10 21 31 49 62 67 68 81 86 95 123 130 136 143 144	96
ES0143416115	104	13	1 7 11 14 35 56 65 83 91 97 100 102 104	97
ES0147200036	220	13	1 16 23 38 39 53 65 117 121 131 171 219 220	102
ES0147645016	51	5	1 6 41 46 51	100
ES0148396015	112	9	1 2 27 86 92 95 109 111 112	100
ES0152503035	8	4	1 4 7 8	97
ES0167050915	22	9	1 2 3 8 9 18 19 21 22	94
ES0171996012	21	4	1 16 20 21	92
ES0173093115	225	32	1 7 8 13 19 20 31 38 57 74 80 86 93 104 111 112 125 130 132 133 134 143 149 171 185 186 210 213 216 220 222 225	96
ES0187165017	22	6	1 2 3 11 12 22	94
ES0178430E18	86	23	1 5 7 14 19 20 44 46 50 52 53 57 59 63 67 72 75 77 81 82 83 85 86	93
ES0182870214	74	17	1 7 9 14 15 26 29 31 34 36 44 55 59 63 72 73 74	93
LU0323134006	12	5	1 2 4 7 12	90
ES0113900J37	127	22	1 2 6 7 24 42 44 45 71 72 73 101 106 109 110 113 115 116 121 123 126 127	91

Trend detail

Spread

ISIN ES0113860A34

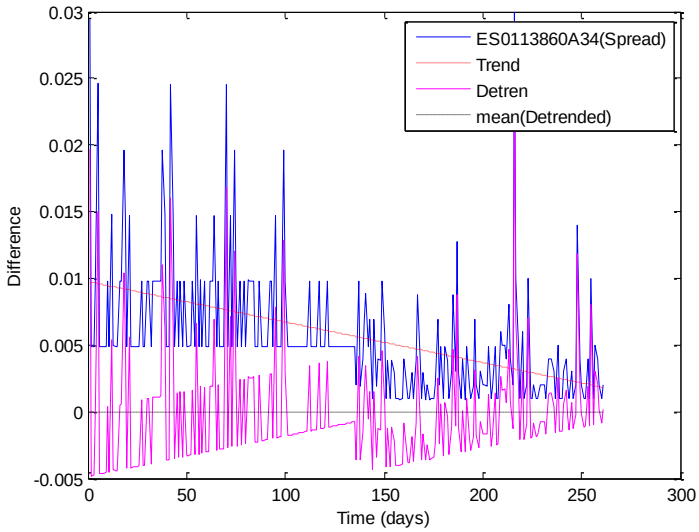


Figure 33: Analysis of trend of spread for the asset ES0113860A34 along 261 days

ISIN ES0113900J37

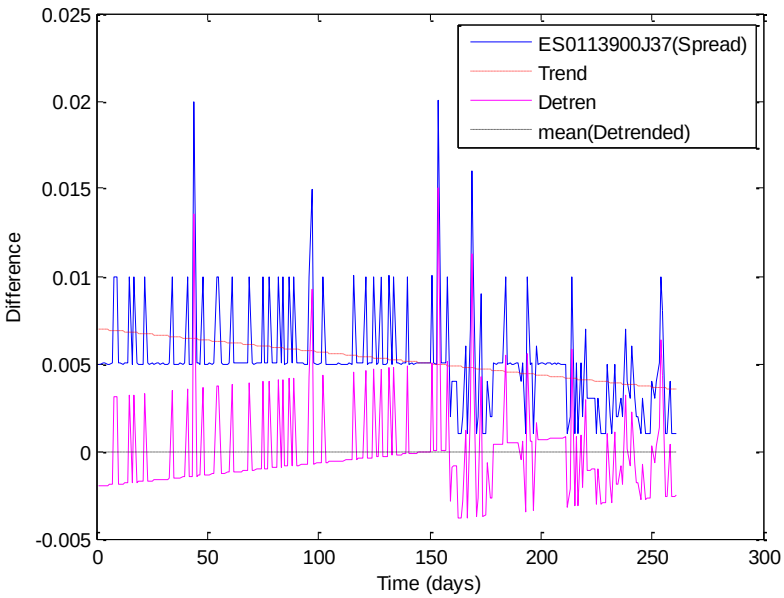


Figure 34: Analysis of trend of spread for the asset ES0113900J37 along 261 days

ISIN ES0124244E34

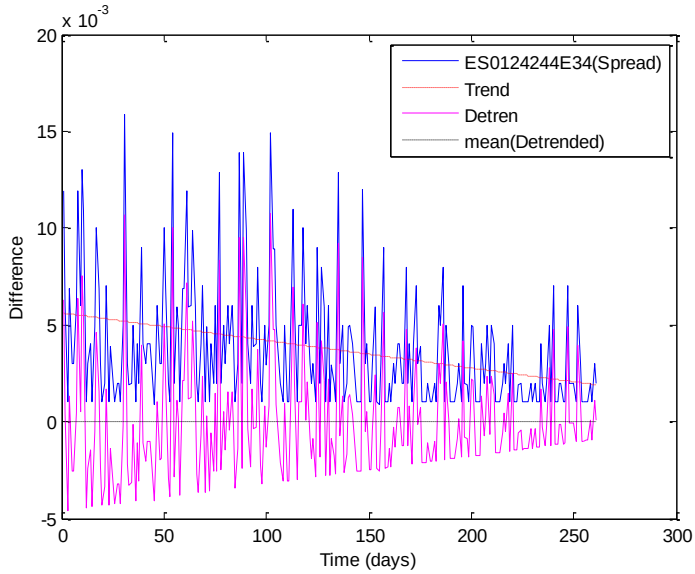


Figure 35: Analysis of trend of spread for the asset ES0124244E34 along 261 days

ISIN ES0144580Y14

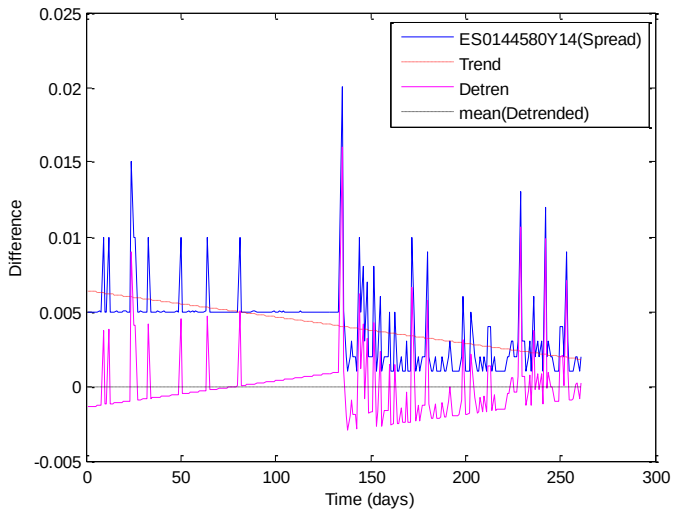


Figure 36: Analysis of trend of spread for the asset ES0144580Y14 along 261 days

ISIN ES0105200416

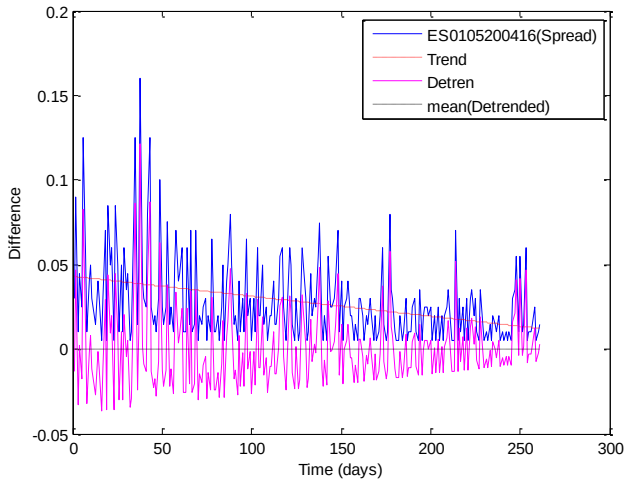
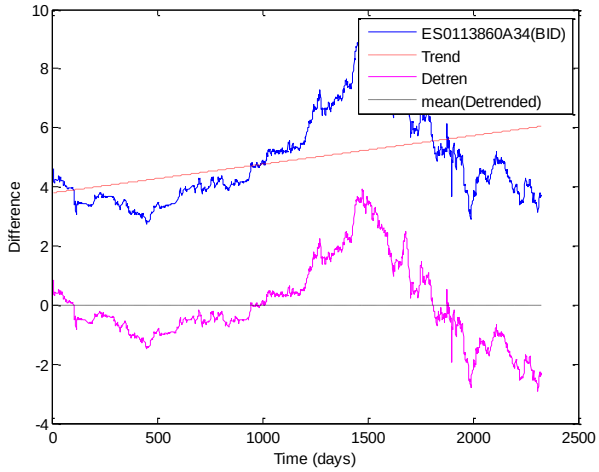


Figure 37: Analysis of trend of spread for the asset ES0105200416 along 261 days

BID

ISIN ES0113860A34



Analysis of trend of BID for the asset ES0113860A34

ISIN ES0113900J37

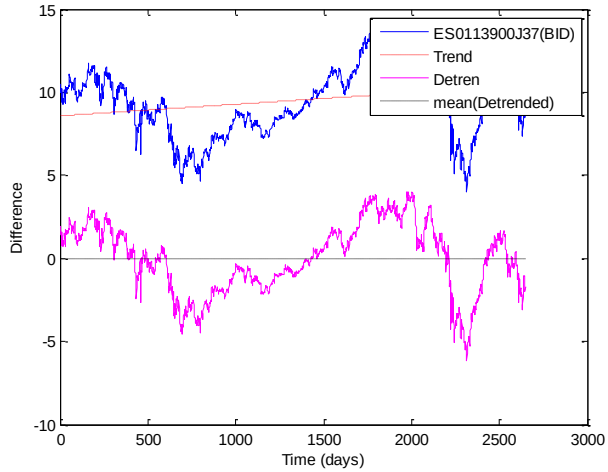


Figure 38: Analysis of trend of BID for the asset ES0113900J37

ISIN ES0124244E34

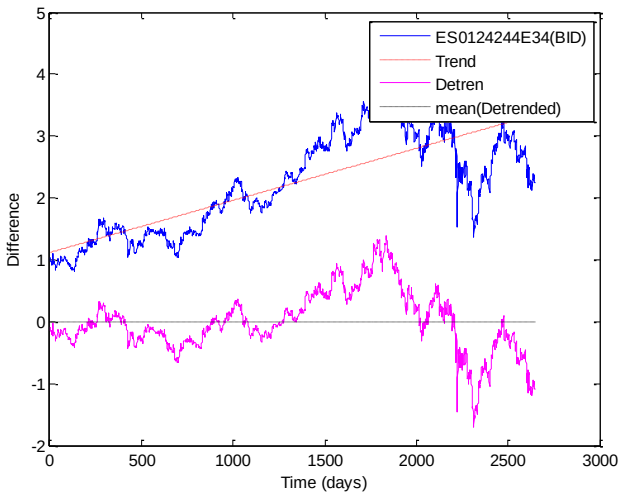


Figure 39: Analysis of trend of BID for the asset ES0124244E34

ISIN ES0144580Y14

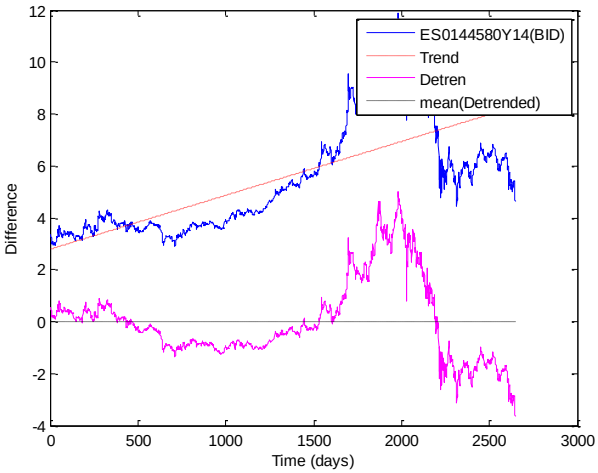


Figure 40: Analysis of trend of BID for the asset ES0144580Y14

ISIN ES0105200416

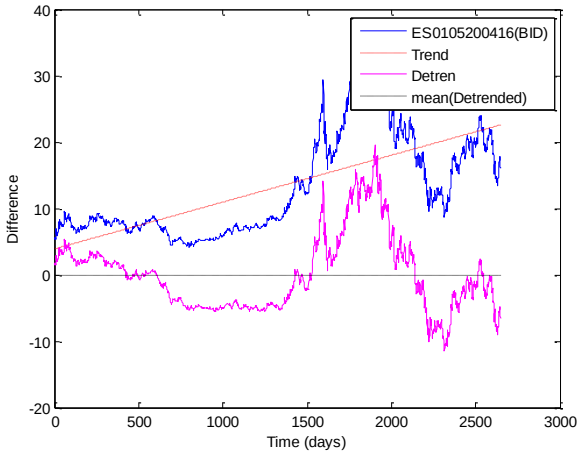
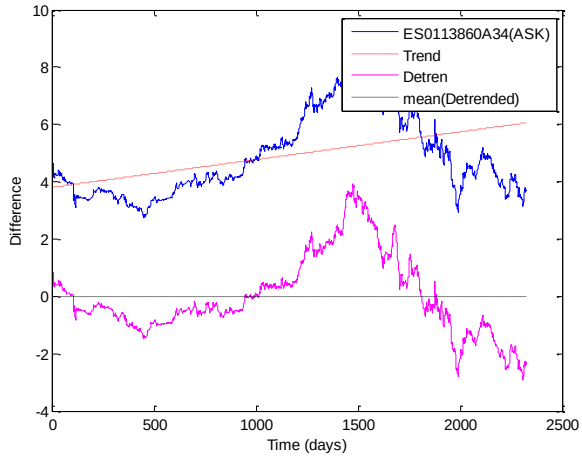


Figure 41: Analysis of trend of BID for the asset ES0105200416

ASK

ISIN ES0113860A34



Analysis of trend of ASK for the asset ES0113860A34

ISIN ES0113900J37

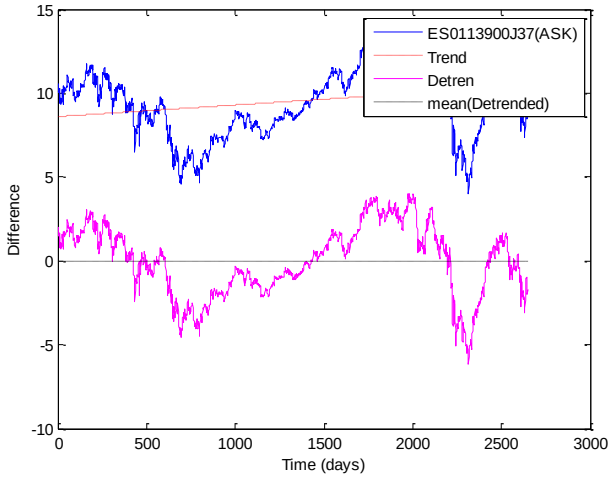


Figure 42: Analysis of trend of ASK for the asset ES0113900J37

ISIN ES0124244E34

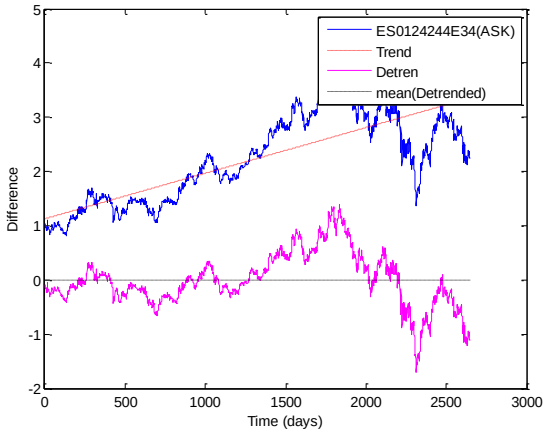


Figure 43: Analysis of trend of ASK for the asset ES0124244E34

ISIN ES0144580Y14

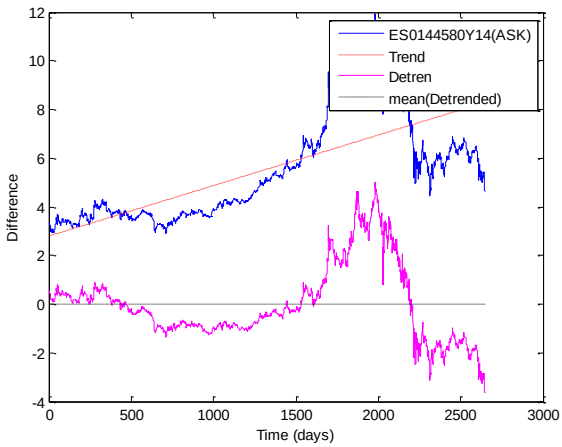


Figure 44: Analysis of trend of ASK for the asset ES0144580Y14

ISIN ES0105200416

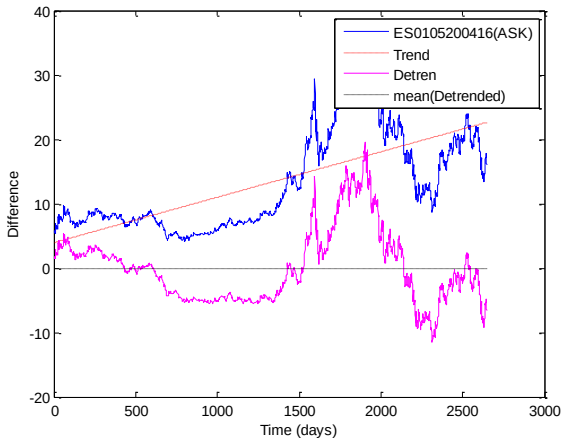


Figure 45: Analysis of trend of ASK for the asset ES0105200416

Results detail

#N : Number of neurons.

n.feat.: Non feature. Evaluated without the chosen feature {p_close=1, Bid=2, Ask=3, Volume=4, Spread=5, Volatility=6, 0=none remove feature}.

r.feat.: Fixed set of removed features which are chosen in base of performance error. {p_close=1, Bid=2, Ask=3, Volume=4, Spread=5, Volatility=6}.

delay: Set of delays which are used (max. 5).

Time: Time spent in minutes.

Error: Nrmse performance.

ISIN					
#N.	**Error**	**Time(min)**			
n.feat.	**r.feat.**	**delay**	**#N.**	**Error**	**Time(min)**
ES0113860A34					
2	1,0141	6,7676			
4	1,0468	10,4619			
3	1,0517	8,2818			
0		1 3 4 8 13	2	1,0141	25,535
2	1,016	2,4539			
4	1,0568	9,8967			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
3	1,0432	7,7759			
6		1 3 4 8 13	2	1,016	20,1311
2	1,0518	2,97			
4	1,1389	6,005			
3	1,0898	3,1803			
5		1 3 4 8 13	2	1,0518	12,1595
2	1,0076	5,6904			
4	1,0493	8,4313			
3	1,049	4,2456			
4	4	1 3 4 8 13	2	1,0076	18,372
2	1,0155	3,2742			
4	1,0557	7,5174			
3	1,0521	6,3797			
3	4	1 3 4 8 13	2	1,0155	17,1755
2	1,0155	3,4318			
4	1,0535	9,5595			
3	1,0539	5,2351			
2	4	1 3 4 8 13	2	1,0155	18,2311
2	1,0157	5,498			
4	1,0794	7,9506			
3	1,0501	4,8779			
1	4	1 3 4 8 13	2	1,0157	18,3309
ES0113900J37					
2	1,4181	6,5598			
4	1,474	9,9087			
3	1,424	11,3436			
0		1	2	1,4181	27,8158
2	1,4054	5,3077			
4	1,4181	11,3964			
3	1,4174	11,166			
6	6	1	2	1,4054	27,8743
2	1,4208	7,618			
4	1,4249	11,4125			
3	1,4227	10,9905			
5	6	1	2	1,4208	30,0255
2	1,3878	7,227			
4	1,4209	11,2693			
3	1,4832	10,4825			
4	6 4	1	2	1,3878	28,9834

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
2	1,4286	5,5404			
4	1,3956	9,8741			
8	1,4048	11,7984			
6	1,4046	11,4814			
5	1,4047	11,418			
3	6 4	1	4	1,3956	50,1177
2	1,5069	6,6805			
4	1,4589	10,6614			
8	1,3906	11,8147			
16	1,3904	13,5864			
32	1,4391	21,1657			
24	1,3903	15,9667			
28	1,4016	17,6445			
26	1,4032	16,8744			
25	1,3947	16,3369			
2	6 4	1	24	1,3903	130,7372
2	1,3882	10,768			
4	1,395	11,1473			
3	1,7538	10,1601			
1	6 4	1	2	1,3882	32,0795
ES0124244E34					
2	1,0197	7,1077			
4	1,2104	11,7383			
3	1,1349	12,6135			
0		1 3 5 7 8	2	1,0197	31,4644
2	1,0422	3,0836			
4	1,1896	8,2317			
3	1,0242	10,4463			
6		1 3 5 7 8	3	1,0242	21,766
2	1,0968	6,7351			
4	1,2013	13,1322			
3	1,1011	10,1172			
5		1 3 5 7 8	2	1,0968	29,9889
2	1,0092	7,2405			
4	1,3263	11,2563			
3	1,4871	7,1074			
4	4	1 3 5 7 8	2	1,0092	25,6086
2	1,0033	5,8225			
4	1,3543	9,1833			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
3	1,1447	8,4589			
3	4 3	1 3 5 7 8	2	1,0033	23,4691
2	1,0354	4,9793			
4	1,1787	7,2634			
3	1,1254	6,8934			
2	4 3	1 3 5 7 8	2	1,0354	19,1403
2	1,0233	7,3666			
4	1,2268	10,1926			
3	1,1395	7,572			
1	4 3	1 3 5 7 8	2	1,0233	25,1351
ES0144580Y14					
2	1,0505	1,546			
4	1,0959	6,506			
3	1,0788	3,6704			
0		1 39 52	2	1,0505	11,7263
2	1,0963	2,0662			
4	1,0952	3,9538			
8	1,1033	9,6			
6	1,1037	7,6217			
5	1,0961	5,7611			
6		1 39 52	4	1,0952	29,0076
2	1,8611	2,6861			
4	1,1285	6,1459			
8	1,1631	11,0826			
6	1,1569	8,4319			
5	1,1457	7,8258			
5		1 39 52	4	1,1285	36,1769
2	1,0637	2,5699			
4	1,1441	7,0424			
3	1,0718	5,1197			
4		1 39 52	2	1,0637	14,7361
2	1,0506	1,5457			
4	1,1367	4,9848			
3	1,0773	3,788			
3		1 39 52	2	1,0506	10,3223
2	1,0295	1,5689			
4	1,137	4,901			
3	1,0758	3,3207			
2	2	1 39 52	2	1,0295	9,795

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
2	1,0512	1,5715			
4	1,0932	5,6332			
3	1,0761	4,8212			
1	2	1 39 52	2	1,0512	12,03
ES0105200416					
2	1,0599	4,6054			
4	1,2932	9,8197			
3	1,0525	8,1102			
0		1 3 9 17 28	3	1,0525	22,5398
2	1,0635	6,888			
4	1,0798	12,9528			
3	1,1132	6,2276			
6		1 3 9 17 28	2	1,0635	26,0724
2	1,1236	2,6635			
4	1,1601	8,1464			
3	1,1417	3,2999			
5		1 3 9 17 28	2	1,1236	14,1142
2	1,0762	5,0295			
4	1,1151	14,7905			
3	1,2568	7,9129			
4		1 3 9 17 28	2	1,0762	27,7372
2	1,058	4,1624			
4	1,2932	12,9312			
3	1,0579	6,1152			
3		1 3 9 17 28	3	1,0579	23,2129
2	1,0833	6,0237			
4	1,2927	15,2806			
3	1,0603	7,4324			
2		1 3 9 17 28	3	1,0603	28,7411
2	1,0578	4,1343			
4	1,2948	12,3805			
3	1,0725	5,7091			
1		1 3 9 17 28	2	1,0578	22,2281
ES0111845014					
2	1,0062	2,9003			
4	1,02	7,5985			
3	1,0142	6,0981			
0		1 2 4 6	2	1,0062	16,6011
2	1,0146	2,2368			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	1,0155	3,5846			
3	1,0157	2,8455			
6		1 2 4 6	2	1,0146	8,6713
2	1,0248	3,9551			
4	1,0247	9,995			
8	1,0245	11,5103			
16	1,0277	24,5717			
12	1,0242	21,2502			
14	1,0241	24,2834			
15	1,0241	29,6251			
5		1 2 4 6	15	1,0241	125,1965
2	1,017	4,3282			
4	1,0182	6,9678			
3	1,0163	4,7526			
4		1 2 4 6	3	1,0163	16,0525
2	1,0053	2,5956			
4	1,0176	7,3219			
3	1,0141	6,0942			
3	3	1 2 4 6	2	1,0053	16,0156
2	1,0045	2,4386			
4	1,0161	6,1212			
3	1,0098	6,0705			
2	3 2	1 2 4 6	2	1,0045	14,6345
2	0,99865	3,1029			
4	1,0034	3,8249			
3	0,99541	4,6023			
1	3 2 1	1 2 4 6	3	0,99541	11,5342
ES0112501012					
2	1,0825	7,1157			
4	1,0857	8,5946			
3	1,1318	8,9513			
0		1 3 5 38 39	2	1,0825	24,6661
2	1,0655	3,6694			
4	1,0583	5,7512			
8	1,3452	19,5974			
6	1,1992	14,8131			
5	1,2885	11,66			
6	6	1 3 5 38 39	4	1,0583	55,4968
2	1,1125	6,3365			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	1,1948	7,7129			
3	1,1094	4,1566			
5	6	1 3 5 38 39	3	1,1094	18,2108
2	1,0324	3,5053			
4	1,0295	8,5946			
8	1,3191	14,7616			
6	1,0784	9,7914			
5	1,0666	9,6549			
4	6 4	1 3 5 38 39	4	1,0295	46,3131
2	1,0296	3,0597			
4	1,0536	7,9324			
3	1,0406	8,0005			
3	6 4	1 3 5 38 39	2	1,0296	18,997
2	1,0303	2,5899			
4	1,1299	8,949			
3	1,0413	7,246			
2	6 4	1 3 5 38 39	2	1,0303	18,789
2	1,0298	4,7315			
4	1,078	7,9236			
3	1,0326	8,0923			
1	6 4	1 3 5 38 39	2	1,0298	20,7518
ES0113211835					
2	1,0797	7,2444			
4	1,6392	8,9441			
3	1,2141	7,8591			
0		1 4 6 7	2	1,0797	24,0515
2	1,0577	3,062			
4	1,6588	7,8081			
3	1,0558	4,8459			
6	6	1 4 6 7	3	1,0558	15,7202
2	1,0517	3,5326			
4	1,0476	4,2986			
8	1,0471	10,9544			
16	1,0569	20,8696			
12	1,047	19,2635			
14	1,0563	21,218			
13	1,0473	19,5207			
5	6 5	1 4 6 7	12	1,047	99,6631
2	1,06	4,1184			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	1,1093	6,6805			
3	1,0813	5,3378			
4	6 5	1 4 6 7	2	1,06	16,1406
2	1,0603	1,8343			
4	1,047	7,6713			
8	1,0471	13,3123			
6	1,0466	10,0701			
7	1,0466	10,7794			
3	6 5 3	1 4 6 7	7	1,0466	43,6725
2	1,0457	4,1444			
4	1,045	8,5977			
8	1,0453	11,8121			
6	1,053	11,8228			
5	1,0453	9,9921			
2	6 5 3 2	1 4 6 7	4	1,045	46,3736
2	1,1543	2,1006			
4	1,1542	11,1296			
8	1,1637	6,0534			
6	1,154	7,695			
7	1,1633	6,5895			
1	6 5 3 2	1 4 6 7	6	1,154	33,5725
ES0113440038					
2	0,98381	5,9533			
4	3,6651	13,6246			
3	2,7526	8,3645			
0		1 2 3 4 5	2	0,98381	27,9471
2	1,0769	4,7554			
4	1,3003	7,4886			
3	1,1213	7,4366			
6		1 2 3 4 5	2	1,0769	19,6852
2	1,2968	6,271			
4	1,5265	7,3276			
3	1,8796	11,2539			
5		1 2 3 4 5	2	1,2968	24,8567
2	0,98414	4,1951			
4	2,3298	10,2929			
3	3,0168	8,2278			
4		1 2 3 4 5	2	0,98414	22,72
2	0,98347	5,4286			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	3,6603	7,7694			
3	2,4911	7,3198			
3	3	1 2 3 4 5	2	0,98347	20,5217
2	0,98216	5,6449			
4	2,7672	8,7782			
3	2,4481	8,253			
2	3 2	1 2 3 4 5	2	0,98216	22,6805
2	0,98648	4,4276			
4	1,2079	6,3386			
3	1,1388	7,8401			
1	3 2	1 2 3 4 5	2	0,98648	18,6104
ES0113679137					
2	1,093	3,9882			
4	1,5743	12,5248			
3	2,672	5,947			
0		1 2 5 23 25	2	1,093	22,4644
2	1,118	5,1153			
4	1,1721	8,6276			
3	1,1068	10,2701			
6		1 2 5 23 25	3	1,1068	24,0174
2	1,1834	5,3836			
4	1,14	9,8439			
8	2,2849	19,5753			
6	2,2232	10,1536			
5	1,5651	15,0099			
5		1 2 5 23 25	4	1,14	59,972
2	1,3493	4,9091			
4	1,1721	11,8709			
8	1,7707	16,6404			
6	1,2237	11,86			
5	1,2877	12,2255			
4		1 2 5 23 25	4	1,1721	57,5105
2	1,0926	3,9078			
4	1,1662	14,4075			
3	1,2596	6,3802			
3	3	1 2 5 23 25	2	1,0926	24,6999
2	1,1658	4,9998			
4	1,2748	9,5735			
3	1,2399	8,6607			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
2	3	1 2 5 23 25	2	1,1658	23,2384
2	1,174	5,9052			
4	2,2971	10,4544			
3	1,2755	8,4191			
1	3	1 2 5 23 25	2	1,174	24,7828
ES0113790531					
2	1,1758	2,82			
4	1,3503	9,0509			
3	1,6035	7,4007			
0		1 15	2	1,1758	19,2757
2	1,4633	1,9602			
4	1,6507	5,9397			
3	1,7765	4,6701			
6		1 15	2	1,4633	12,5739
2	1,3052	5,653			
4	1,5628	10,3257			
3	1,4345	6,532			
5		1 15	2	1,3052	22,5146
2	1,2604	2,1739			
4	1,3657	7,6843			
3	1,3963	4,8925			
4		1 15	2	1,2604	14,7554
2	1,1093	2,5095			
4	1,0532	8,1482			
8	1,3536	14,1028			
6	1,1942	10,6798			
5	1,2524	8,7376			
3	3	1 15	4	1,0532	44,1826
2	1,0355	3,2394			
4	1,0496	5,9839			
3	1,0429	4,8506			
2	3 2	1 15	2	1,0355	14,0781
2	1,033	1,8356			
4	1,0568	6,6667			
3	1,0505	4,867			
1	3 2 1	1 15	2	1,033	13,374
ES0115056139					
2	1,0858	1,0299			
4	1,0702	6,4158			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
8	1,0715	8,5741			
6	1,0713	7,4394			
5	1,0705	6,916			
0		1	4	1,0702	30,3794
2	1,104	2,1606			
4	1,1597	4,2375			
3	1,2476	4,8077			
6		1	2	1,104	11,21
2	1,0888	1,0665			
4	1,0665	4,5929			
8	1,0665	5,98			
6	1,0667	5,5084			
5	1,0663	5,1837			
5	5	1	5	1,0663	22,3375
2	1,49	2,9814			
4	1,4174	2,8691			
8	6,0824	6,0008			
6	1,3059	3,6382			
7	1,3282	5,0609			
4	5	1	6	1,3059	20,5573
2	1,076	0,95473			
4	1,0605	3,3304			
8	1,0608	4,8636			
6	1,0612	4,1566			
5	1,0607	3,97			
3	5 3	1	4	1,0605	17,2808
2	1,0631	0,87439			
4	1,054	3,0659			
8	1,0515	4,1351			
16	1,0531	4,7427			
12	1,053	4,4268			
10	1,0541	4,1855			
9	1,0515	4,2372			
2	5 3 2	1	8	1,0515	25,6757
2	1,039	0,81017			
4	1,0353	4,5698			
8	1,0349	7,2673			
16	1,034	8,3931			
32	1,0358	10,8545			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
24	1,036	9,3983			
20	1,0361	8,807			
18	1,0301	8,5884			
19	1,0354	8,7116			
1	5 3 2 1	1	18	1,0301	67,4067
ES0116870314					
2	1,0624	8,293			
4	1,4641	11,5646			
3	2,4142	13,0565			
0		1 2 5 6 9	2	1,0624	32,9186
2	1,0955	5,3989			
4	1,4622	11,1213			
3	1,1658	8,1001			
6		1 2 5 6 9	2	1,0955	24,625
2	1,2063	10,2342			
4	1,0364	5,0186			
8	2,0245	22,383			
6	1,4311	15,7337			
5	1,2889	12,5409			
5	5	1 2 5 6 9	4	1,0364	65,9156
2	1,0716	4,9406			
4	1,3793	11,5774			
3	1,2842	9,1435			
4	5	1 2 5 6 9	2	1,0716	25,6661
2	1,0342	4,9294			
4	1,3537	8,6474			
3	1,1695	8,5455			
3	5 3	1 2 5 6 9	2	1,0342	22,1264
2	1,0427	4,4174			
4	1,5094	8,1906			
3	1,1785	4,5732			
2	5 3	1 2 5 6 9	2	1,0427	17,1851
2	1,0424	9,0608			
4	1,489	7,6342			
3	1,1744	5,6007			
1	5 3	1 2 5 6 9	2	1,0424	22,2996
ES0118594417					
2	3,3923	3,6754			
4	1,2222	7,0754			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
8	1,3437	10,9765			
6	1,3321	9,4758			
5	1,3884	6,6033			
0		1 4 6	4	1,2222	37,8113
2	1,0718	4,074			
4	1,0919	4,5755			
3	1,0762	3,5105			
6	6	1 4 6	2	1,0718	12,1644
2	1,0732	3,1954			
4	1,0969	3,3101			
3	1,0661	3,0267			
5	6 5	1 4 6	3	1,0661	9,5366
2	1,0833	4,022			
4	1,0802	5,4996			
8	1,0657	9,4248			
16	1,111	17,3689			
12	1,0952	14,6475			
10	1,0871	13,147			
9	1,0826	10,5256			
4	6 5 4	1 4 6	8	1,0657	74,6405
2	1,0813	5,8025			
4	1,0921	5,5484			
3	1,0959	3,0925			
3	6 5 4	1 4 6	2	1,0813	14,4473
2	1,0816	4,4814			
4	1,0971	5,4457			
3	1,0818	5,913			
2	6 5 4	1 4 6	2	1,0816	15,8437
2	1,0818	4,0586			
4	1,0809	7,7075			
8	1,0875	11,0875			
6	1,0853	7,1279			
5	1,0943	6,0331			
1	6 5 4	1 4 6	4	1,0809	36,0199
ES0118900010					
2	1,0381	3,502			
4	1,0395	6,0055			
3	1,039	4,6291			
0		1 2	2	1,0381	14,1407

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
2	1,0142	1,4862			
4	1,0159	5,8365			
3	1,0171	5,4158			
6	6	1 2	2	1,0142	12,7429
2	1,0171	1,5754			
4	1,0183	5,1254			
3	1,0184	4,1522			
5	6	1 2	2	1,0171	10,8577
2	1,0062	0,86919			
4	1,0073	7,3814			
3	1,0069	4,6956			
4	6 4	1 2	2	1,0062	12,9502
2	1,0063	0,83929			
4	1,0071	9,2147			
3	1,0069	4,6707			
3	6 4	1 2	2	1,0063	14,7288
2	1,0062	0,91027			
4	1,0097	9,1469			
3	1,0069	4,7279			
2	6 4	1 2	2	1,0062	14,7889
2	1,0062	0,99217			
4	1,0072	8,56			
3	1,0068	4,1894			
1	6 4	1 2	2	1,0062	13,7463
ES0122060314					
2	1,0672	3,8275			
4	1,1556	9,3346			
3	1,1172	8,5455			
0		1 8 16 41 45	2	1,0672	21,7117
2	1,0411	2,9133			
4	1,0771	6,4257			
3	1,1018	9,5538			
6	6	1 8 16 41 45	2	1,0411	18,8969
2	1,0759	2,1211			
4	1,1514	9,6307			
3	1,1238	3,92			
5	6	1 8 16 41 45	2	1,0759	15,6763
2	1,026	1,9716			
4	1,057	8,3799			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
3	1,0639	4,0708			
4	6 4	1 8 16 41 45	2	1,026	14,4265
2	1,0593	1,8408			
4	1,0679	7,4935			
3	1,0636	5,94			
3	6 4	1 8 16 41 45	2	1,0593	15,2785
2	1,059	4,0168			
4	1,0557	7,2657			
8	1,287	11,1785			
6	1,5282	10,6939			
5	1,0642	8,4176			
2	6 4	1 8 16 41 45	4	1,0557	41,5776
2	1,0585	1,8767			
4	1,043	6,9459			
8	1,2924	11,1718			
6	1,1926	7,8123			
5	1,2072	8,3185			
1	6 4	1 8 16 41 45	4	1,043	36,1298
ES0125220311					
2	1,0137	1,6625			
4	1,0124	6,8217			
8	1,0233	14,752			
6	1,0155	11,5412			
5	1,0132	9,6375			
0		1 2 5 11	4	1,0124	44,4205
2	1,0072	1,5454			
4	1,0138	8,1056			
3	1,0087	4,6558			
6	6	1 2 5 11	2	1,0072	14,311
2	1,0171	1,8252			
4	1,0234	9,151			
3	1,0234	6,7933			
5	6	1 2 5 11	2	1,0171	17,7737
2	1,0088	1,534			
4	1,0123	6,2125			
3	1,0106	4,8784			
4	6	1 2 5 11	2	1,0088	12,6288
2	1,0268	1,6232			
4	1,0175	7,754			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
8	1,0254	12,2357			
6	1,0181	9,6814			
5	1,013	9,3341			
3	6	1 2 5 11	5	1,013	40,6336
2	1,0072	1,4817			
4	1,0174	7,6916			
3	1,0139	4,4252			
2	6	1 2 5 11	2	1,0072	13,6033
2	1,0074	1,6089			
4	1,0173	7,0369			
3	1,0108	4,3631			
1	6	1 2 5 11	2	1,0074	13,0133
ES0130670112					
2	40,7309	10,5098			
4	14,41	10,0537			
8	17,312	23,5135			
6	17,0953	17,6474			
5	5,1529	13,5996			
0		1 23 35 44 45	5	5,1529	75,3295
2	1,8837	6,8274			
4	1,9132	11,4086			
3	1,9048	11,1845			
6	6	1 23 35 44 45	2	1,8837	29,4252
2	2,5694	4,1501			
4	14,1056	7,7215			
3	6,292	9,2176			
5	6	1 23 35 44 45	2	2,5694	21,0942
2	2,6562	5,2346			
4	4,0539	11,7661			
3	4,0245	8,7969			
4	6	1 23 35 44 45	2	2,6562	25,8026
2	1,9051	8,1903			
4	1,9269	14,5819			
3	1,9315	12,4819			
3	6	1 23 35 44 45	2	1,9051	35,2583
2	2,0235	6,6935			
4	1,8935	13,4863			
8	1,8884	17,3429			
16	2,0399	24,3551			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
12	2,0312	19,0329			
10	1,9334	20,3527			
9	2,0888	15,2977			
2	6	1 23 35 44 45	8	1,8884	116,5668
2	1,9118	8,2259			
4	1,9083	12,5292			
8	1,9168	16,7613			
6	1,9346	14,2959			
5	1,9043	13,3428			
1	6	1 23 35 44 45	5	1,9043	65,1603
ES0130960018					
2	1,0065	1,2628			
4	1,0044	5,6506			
8	1,0044	8,9355			
16	1,0045	12,867			
12	1,0044	10,0394			
14	1,0046	11,829			
13	1,0045	11,5719			
0		1	12	1,0044	62,1625
2	1,0062	1,1952			
4	1,0082	4,9042			
3	1,0085	4,1605			
6		1	2	1,0062	10,2646
2	1,0051	1,0631			
4	1,002	5,8259			
8	1,0033	8,5728			
6	1,0022	7,7543			
5	1,0021	7,4704			
5	5	1	4	1,002	30,6914
2	1,0013	1,4459			
4	1,0001	4,9608			
8	1,0003	7,8981			
6	1,0001	6,2577			
7	1,0002	8,3159			
4	5 4	1	6	1,0001	28,8851
2	1,0039	1,0213			
4	1,0003	4,3875			
8	1,0008	4,7918			
6	1,0005	4,341			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
5	1,0009	4,5495			
3	5 4	1	4	1,0003	19,0961
2	1,0039	1,5816			
4	1,0005	4,5513			
8	1,0003	4,2362			
16	1	7,2803			
32	1,0007	15,4329			
24	1,0007	10,469			
20	1,0044	7,9334			
18	1,0009	8,5106			
17	1,0002	7,4038			
2	5 4 2	1	16	1	67,406
2	1,0229	1,288			
4	1,0022	4,1408			
8	1,0031	3,9611			
6	1,0026	3,9356			
5	1,0025	4,0087			
1	5 4 2	1	4	1,0022	17,3398
ES0132105018					
2	2,4852	7,6227			
4	1,3338	6,1969			
8	1,9323	10,1707			
6	1,833	9,3699			
5	1,7393	9,2901			
0		1 2	4	1,3338	42,6556
2	1,0422	4,7404			
4	1,216	4,8342			
3	1,0485	5,6072			
6	6	1 2	2	1,0422	15,1862
2	1,0622	10,159			
4	1,2172	9,8075			
3	1,162	7,8339			
5	6	1 2	2	1,0622	27,8041
2	1,0331	7,0939			
4	1,1542	7,032			
3	1,1388	7,4659			
4	6 4	1 2	2	1,0331	21,5963
2	1,0574	7,4379			
4	1,1553	6,8536			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
3	1,1979	5,367			
3	6 4	1 2	2	1,0574	19,6629
2	1,0558	7,2356			
4	1,1326	7,1883			
3	1,1046	6,5528			
2	6 4	1 2	2	1,0558	20,9814
2	1,0541	8,8676			
4	1,1633	5,7934			
3	1,2048	5,1005			
1	6 4	1 2	2	1,0541	19,7656
ES0140609019					
2	1,3591	2,9019			
4	1,3451	4,5487			
8	1,3439	5,465			
16	1,3422	9,3504			
32	1,2986	27,5594			
64	1,3073	97,4629			
48	1,3026	55,5912			
40	1,3413	40,2542			
36	1,3452	33,5571			
34	1,3414	30,1742			
33	1,3462	28,7523			
0		1	32	1,2986	335,6455
2	1,2186	2,2818			
4	1,2163	4,5058			
8	1,2156	5,6254			
16	1,3259	8,2207			
12	1,2587	6,8222			
10	1,2164	7,0193			
9	1,2162	6,3622			
6	6	1	8	1,2156	40,8429
2	1,2124	1,6237			
4	1,2105	4,1148			
8	1,2102	7,0486			
16	1,21	7,7514			
32	1,2239	13,589			
24	1,2103	10,5506			
20	1,2094	9,0382			
22	1,2097	9,716			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
21	1,226	7,2829			
5	6 5	1	20	1,2094	70,7225
2	1,1448	2,0023			
4	1,1384	1,884			
8	1,1323	2,4341			
16	1,1377	5,7466			
12	1,1376	3,8639			
10	1,1378	3,9336			
9	1,1386	3,5059			
4	6 5 4	1	8	1,1323	23,378
2	1,135	2,2248			
4	1,1343	3,0964			
8	1,1331	3,9512			
16	1,1323	5,5812			
32	1,132	9,1089			
64	1,1316	22,0396			
96	1,1318	43,0901			
80	1,1305	31,4264			
88	1,1304	36,9663			
92	1,1317	39,098			
90	1,1317	37,3365			
89	1,1304	36,1085			
3	6 5 4 3	1	89	1,1304	270,0367
2	1,1532	0,81667			
4	1,1303	3,0321			
8	1,1301	3,9304			
16	1,1295	5,7593			
32	1,048	6,0508			
64	1,5799	12,9112			
48	1,0837	8,742			
40	1,1206	6,4969			
36	1,0701	5,2011			
34	1,0732	6,272			
33	1,0646	5,9691			
2	6 5 4 3 2	1	32	1,048	65,1892
2	1,1046	1,275			
4	1,1094	3,2204			
3	1,1045	2,5933			
1	6 5 4 3 2	1	3	1,1045	7,0921

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
ES0142090317					
2	1,0239	2,6411			
4	1,1774	5,6888			
3	1,1018	5,2835			
0		1 205 222 226	2	1,0239	13,6184
2	1,0123	2,7482			
4	1,0264	7,7977			
3	1,0367	6,5728			
6	6	1 205 222 226	2	1,0123	17,1232
2	1,035	1,8525			
4	1,0335	4,4156			
8	1,0548	10,1289			
6	1,0532	11,6884			
5	1,0653	9,3905			
5	6	1 205 222 226	4	1,0335	37,4805
2	1,0032	2,2534			
4	1,0073	6,4925			
3	1,0018	4,4426			
4	6 4	1 205 222 226	3	1,0018	13,193
2	1,004	2,2061			
4	1,0019	8,9857			
8	1,0209	10,6715			
6	1,0093	10,1427			
5	1,0124	8,8			
3	6 4	1 205 222 226	4	1,0019	40,8106
2	1,0022	4,3295			
4	1,0021	8,9747			
8	1,0265	10,7659			
6	1,0193	10,704			
5	1,0057	8,3913			
2	6 4	1 205 222 226	4	1,0021	43,1699
2	1,0022	1,8005			
4	1,0057	7,6147			
3	1,004	4,6842			
1	6 4	1 205 222 226	2	1,0022	14,1038
ES0143416115					
2	1,0852	8,2777			
4	1,0898	6,0372			
3	1,0406	4,6967			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
0		1 7 8 18	3	1,0406	19,0157
2	1,098	4,2557			
4	1,154	5,6249			
3	1,1677	4,4619			
6		1 7 8 18	2	1,098	14,3472
2	1,0428	2,593			
4	1,0776	7,4158			
3	1,1544	4,2918			
5		1 7 8 18	2	1,0428	14,3061
2	1,0354	7,4009			
4	1,0797	8,877			
3	1,0393	5,6017			
4	4	1 7 8 18	2	1,0354	21,8841
2	1,0388	6,3004			
4	1,0384	8,2829			
8	1,0554	12,3779			
6	1,0985	10,5665			
5	1,0877	7,2928			
3	4	1 7 8 18	4	1,0384	44,8253
2	1,0476	7,6019			
4	1,0867	8,9321			
3	1,0316	8,6648			
2	4 2	1 7 8 18	3	1,0316	25,2027
2	1,0333	8,3591			
4	1,0612	6,3615			
3	1,0353	11,4455			
1	4 2	1 7 8 18	2	1,0333	26,1705
ES0147200036					
2	1,2193	6,3516			
4	1,3552	9,9375			
3	1,6167	6,6584			
0		1 8 9	2	1,2193	22,9522
2	1,2175	6,3089			
4	1,3748	8,0044			
3	1,7497	7,0596			
6	6	1 8 9	2	1,2175	21,3781
2	1,1761	2,4843			
4	1,8596	6,8807			
3	1,2647	5,8045			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
5	6 5	1 8 9	2	1,1761	15,1737
2	1,2965	5,2645			
4	1,4372	8,6276			
3	1,4356	7,344			
4	6 5	1 8 9	2	1,2965	21,2401
2	1,1992	4,9226			
4	1,568	6,0695			
3	1,8107	6,0544			
3	6 5	1 8 9	2	1,1992	17,0504
2	1,4525	4,4122			
4	1,5752	8,4103			
3	1,4243	5,2253			
2	6 5	1 8 9	3	1,4243	18,0522
2	1,1977	4,7222			
4	1,6696	5,3274			
3	1,5744	4,5971			
1	6 5	1 8 9	2	1,1977	14,6508
ES0147645016					
2	2,6097	2,9619			
4	2,834	5,3095			
3	2,107	3,6078			
0		1	3	2,107	11,8829
2	1,2	2,8189			
4	1,2292	4,0344			
3	1,1473	5,3792			
6	6	1	3	1,1473	12,2365
2	1,1404	2,1713			
4	1,4985	2,9697			
3	1,1296	3,0022			
5	6 5	1	3	1,1296	8,1487
2	1,1671	2,7017			
4	1,2799	2,7981			
3	1,2804	1,5587			
4	6 5	1	2	1,1671	7,0635
2	1,1602	2,3814			
4	1,1144	5,2736			
8	1,2234	6,9537			
6	1,2627	6,2049			
5	1,2348	6,1841			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
3	6 5 3	1	4	1,1144	27,0022
2	1,5453	2,5925			
4	1,1792	4,5986			
8	1,3027	5,8971			
6	1,883	4,939			
5	1,3247	5,6465			
2	6 5 3	1	4	1,1792	23,6802
2	1,1519	1,8094			
4	1,1333	3,9034			
8	1,4859	5,5232			
6	1,2834	4,7066			
5	1,1197	4,1834			
1	6 5 3	1	5	1,1197	20,1306
ES0148396015					
2	1,1314	2,6762			
4	1,2989	6,441			
3	1,1965	6,4826			
0		1 2 3 4 60	2	1,1314	15,6048
2	1,1961	3,0818			
4	1,1805	5,8909			
8	1,3135	18,4552			
6	1,2008	8,819			
5	1,1929	9,769			
6		1 2 3 4 60	4	1,1805	46,0211
2	1,0974	2,0379			
4	1,1769	3,3582			
3	1,1933	5,205			
5	5	1 2 3 4 60	2	1,0974	10,6055
2	1,1281	4,5724			
4	1,1607	4,971			
3	1,134	3,5984			
4	5	1 2 3 4 60	2	1,1281	13,1462
2	1,1508	3,0607			
4	1,1956	4,8007			
3	1,0949	3,7383			
3	5 3	1 2 3 4 60	3	1,0949	11,6041
2	1,1459	3,9429			
4	1,0979	4,0394			
8	1,9883	6,9719			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
6	1,2976	5,2692			
5	1,276	4,7721			
2	5 3	1 2 3 4 60	4	1,0979	25,0015
2	1,0896	2,9141			
4	1,1217	4,6894			
3	1,0748	2,8353			
1	5 3 1	1 2 3 4 60	3	1,0748	10,4435
ES0152503035					
2	1,0012	6,9511			
4	1,0013	8,8993			
3	1,0013	8,2964			
0		1	2	1,0012	24,1521
2	1,0025	5,3066			
4	1,0025	5,0932			
3	1,0026	4,8646			
6		1	2	1,0025	15,2702
2	0,99985	9,3978			
4	1,001	9,7831			
3	1,0008	9,704			
5	5	1	2	0,99985	28,8888
2	0,99907	5,4143			
4	0,99904	6,0266			
8	0,99904	6,3804			
16	0,99904	7,7566			
32	0,99904	13,031			
24	0,99904	9,9037			
20	0,99904	8,3926			
18	0,99904	7,7894			
17	1,0097	5,8173			
4	5 4	1	16	0,99904	70,5197
2	1,0063	5,7887			
4	1,0137	6,0362			
3	0,99897	5,2125			
3	5 4 3	1	3	0,99897	17,0415
2	0,99882	5,2822			
4	1,0657	4,5763			
3	0,99887	6,479			
2	5 4 3 2	1	2	0,99882	16,3421
2	0,99726	4,0667			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	0,99685	4,9632			
8	0,99682	6,8313			
16	0,99684	7,2065			
12	0,99686	5,3623			
10	0,99759	7,4082			
9	0,99683	6,9779			
1	5 4 3 2 1	1	8	0,99682	42,8223
ES0167050915					
2	1,0083	2,5761			
4	1,0057	9,7995			
8	1,0165	21,2837			
6	1,0098	14,9511			
5	1,0048	11,6517			
0		1 2 3 8 11	5	1,0048	60,2673
2	1,0104	1,8928			
4	1,0097	2,9866			
8	1,0119	10,6216			
6	1,0129	8,82			
5	1,0129	8,9186			
6		1 2 3 8 11	4	1,0097	33,2449
2	1,0152	2,9167			
4	1,0399	8,6253			
3	1,0305	3,4281			
5		1 2 3 8 11	2	1,0152	14,9745
2	1,0167	2,5995			
4	1,0153	11,5399			
8	1,0193	19,7869			
6	1,0187	13,5068			
5	1,0156	12,4455			
4		1 2 3 8 11	4	1,0153	59,8836
2	1,006	2,2373			
4	1,0052	7,3027			
8	1,0057	14,8157			
6	1,0082	11,6208			
5	1,0047	8,7496			
3	3	1 2 3 8 11	5	1,0047	44,7312
2	0,99716	2,8951			
4	1,0042	9,8203			
3	1,0024	5,284			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
2	3 2	1 2 3 8 11	2	0,99716	18,0043
2	1,0295	3,3618			
4	1,0992	4,1566			
3	1,0867	3,5542			
1	3 2	1 2 3 8 11	2	1,0295	11,0769
ES0171996012					
2	1,3145	4,6007			
4	1,2833	7,8352			
8	1,2827	10,0046			
16	1,4474	14,707			
12	1,2823	12,0802			
14	1,2821	13,6979			
15	1,2821	14,1113			
0		1	15	1,2821	77,0445
2	1,31	5,1852			
4	1,3065	6,3945			
8	1,3046	8,2473			
16	1,3034	13,4834			
32	1,303	28,6462			
64	1,3448	93,7966			
48	1,723	54,124			
40	1,3033	40,7186			
36	1,303	34,4736			
38	1,195	37,7678			
39	1,3028	38,428			
6	6	1	38	1,195	361,2767
2	1,2998	5,2864			
4	1,2954	6,0365			
8	1,2933	9,7579			
16	1,2922	12,4411			
32	1,2909	23,1388			
64	1,2935	67,4187			
48	1,291	40,1453			
40	1,2917	30,6794			
36	1,2917	26,4754			
34	1,2919	24,8944			
33	1,2909	23,7369			
5	6	1	32	1,2909	270,019
2	1,2918	7,5437			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	1,2883	7,9145			
8	1,2849	7,2314			
16	1,283	12,4562			
32	1,2813	23,2935			
64	1,2836	66,3254			
48	1,281	40,7922			
56	1,2824	52,5144			
52	1,2821	46,2262			
50	1,2822	43,3386			
49	1,2817	41,0805			
4	6	1	48	1,281	348,7272
2	1,2865	3,3278			
4	1,288	4,0789			
3	1,2891	3,9081			
3	6	1	2	1,2865	11,3194
2	1,281	3,5238			
4	1,2786	4,5261			
8	1,2917	9,2665			
6	1,2922	6,6592			
5	1,278	6,5502			
2	6	1	5	1,278	30,531
2	1,2956	3,3317			
4	1,292	5,0495			
8	1,291	10,2974			
16	1,2906	12,4608			
32	1,1922	23,1492			
64	1,309	67,5253			
48	1,2901	39,4729			
40	1,1933	30,8029			
36	1,1928	26,6715			
34	1,1922	25,0379			
33	1,2909	23,8154			
1	6 1	1	32	1,1922	267,6236
ES0173093115					
2	12,3443	8,5322			
4	10,4708	14,9912			
8	48,67	20,0347			
6	21,076	18,7703			
5	8,2027	15,7288			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
0		1	5	8,2027	78,0621
2	1,2111	10,9656			
4	1,0959	15,579			
8	1,1984	19,4614			
6	1,1503	18,5646			
5	1,1524	18,1021			
6	6	1	4	1,0959	82,6774
2	1,0447	5,2162			
4	1,0875	10,0888			
3	1,2674	8,2036			
5	6 5	1	2	1,0447	23,5125
2	1,0395	9,8			
4	1,0717	17,3062			
3	1,3446	17,0766			
4	6 5 4	1	2	1,0395	44,1868
2	1,0566	12,4125			
4	1,0363	7,6578			
8	1,042	14,7736			
6	1,0341	8,0018			
7	1,0341	9,574			
3	6 5 4 3	1	6	1,0341	52,4249
2	1,2509	10,1876			
4	1,0321	7,1391			
8	1,0319	8,325			
16	1,045	15,3346			
12	1,0668	13,9899			
10	1,2877	14,155			
9	1,0347	11,4864			
2	6 5 4 3 2	1	8	1,0319	80,6234
2	1,0464	7,4191			
4	1,0999	9,4836			
3	1,0442	13,3347			
1	6 5 4 3 2	1	3	1,0442	30,2413
ES0187165017					
2	7,2559	4,2669			
4	5,4184	7,4173			
8	2,5655	10,2238			
16	2,5162	14,6051			
32	2,6379	36,6779			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
24	3,7069	24,1318			
20	2,5221	19,6949			
18	3,473	16,4077			
17	4,9144	15,5273			
0		1	16	2,5162	148,9596
2	7,3468	2,6707			
4	5,5274	6,9332			
8	4,2405	8,5951			
16	2,9668	12,8893			
32	3,5555	28,9338			
24	2,8757	20,3571			
28	3,3337	24,1536			
26	3,1543	22,2507			
25	3,0836	21,1966			
6		1	24	2,8757	147,989
2	16,3191	2,2956			
4	8,1363	4,459			
8	3,9866	9,0111			
16	7,0363	13,4569			
12	4,3262	11,7126			
10	4,6083	10,4879			
9	4,3611	9,9183			
5		1	8	3,9866	61,3487
2	1,1023	1,3356			
4	1,1341	6,076			
3	1,0991	2,8785			
4	4	1	3	1,0991	10,2955
2	1,0981	1,2212			
4	1,1373	5,1535			
3	1,136	3,4565			
3	4 3	1	2	1,0981	9,8374
2	1,0904	1,2035			
4	1,1413	6,0807			
3	1,1391	3,6073			
2	4 3 2	1	2	1,0904	10,8969
2	1,0286	4,2547			
4	1,0287	8,2223			
3	1,0285	6,8076			
1	4 3 2 1	1	3	1,0285	19,2893

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
ES0178430E18					
2	5,4148	5,752			
4	5,4833	11,8324			
3	1,7491	11,2087			
0		1	3	1,7491	28,7975
2	1,7217	3,1229			
4	1,7195	9,606			
8	10,1494	12,672			
6	5,7607	12,2058			
5	1,7229	11,2474			
6	6	1	4	1,7195	48,8587
2	1,7233	3,9843			
4	1,7237	8,1784			
3	1,7277	7,4519			
5	6	1	2	1,7233	19,6189
2	1,316	9,1276			
4	1,3186	11,6398			
3	1,3164	11,4396			
4	6 4	1	2	1,316	32,2111
2	1,3141	10,3923			
4	1,3141	11,4931			
3	1,3145	11,3028			
3	6 4 3	1	2	1,3141	33,1926
2	1,3168	9,3525			
4	1,3166	11,3938			
8	1,3484	11,8943			
6	1,3159	11,647			
7	1,3375	9,5639			
2	6 4 3	1	6	1,3159	53,8567
2	1,314	10,0628			
4	1,313	11,4097			
8	1,315	11,8974			
6	1,3293	11,6574			
5	1,4433	11,1263			
1	6 4 3 1	1	4	1,313	56,1593
ES0182870214					
2	1,885	9,0171			
4	1,3278	8,143			
8	1,3544	20,7765			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
6	1,2801	11,7237			
7	1,5867	19,393			
0		1 26 27 29 32	6	1,2801	69,0588
2	1,2204	5,8805			
4	1,4578	7,0281			
3	1,0509	9,0673			
6	6	1 26 27 29 32	3	1,0509	21,98
2	2,1876	3,4814			
4	1,5279	6,0596			
8	1,3332	10,3829			
16	2,3532	31,3596			
12	1,578	20,3399			
10	1,6295	16,2257			
9	1,5665	13,427			
5	6	1 26 27 29 32	8	1,3332	101,2816
2	2,1134	4,4728			
4	2,0223	5,4262			
8	2,1804	10,3072			
6	2,1736	8,8544			
5	2,151	4,1759			
4	6	1 26 27 29 32	4	2,0223	33,2423
2	1,152	6,2611			
4	1,3816	9,0878			
3	1,3176	4,1681			
3	6	1 26 27 29 32	2	1,152	19,5212
2	1,1575	6,3313			
4	1,39	7,6934			
3	2,4047	6,981			
2	6	1 26 27 29 32	2	1,1575	21,0102
2	2,0329	7,1274			
4	1,5083	7,0515			
8	1,7702	13,2754			
6	1,4106	6,8435			
7	1,18	8,058			
1	6	1 26 27 29 32	7	1,18	42,3608
LU0323134006					
2	1,4497	2,0681			
4	1,6058	4,7853			
3	1,6997	5,2554			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
0		1 3 4 7 10	2	1,4497	12,1148
2	1,8329	6,3711			
4	2,8131	4,1725			
3	1,7013	4,069			
6		1 3 4 7 10	3	1,7013	14,6165
2	1,4651	1,67			
4	4,0843	7,4763			
3	3,2514	7,4901			
5		1 3 4 7 10	2	1,4651	16,6422
2	2,246	5,6574			
4	1,814	4,4447			
8	2,2965	11,5953			
6	1,53	6,4408			
7	1,8838	8,8234			
4		1 3 4 7 10	6	1,53	36,9686
2	1,4381	2,3229			
4	1,2748	5,4177			
8	3,3756	12,3269			
6	2,6898	6,9592			
5	1,8026	6,285			
3	3	1 3 4 7 10	4	1,2748	33,3174
2	1,549	2,4367			
4	2,0044	4,2073			
3	1,6465	3,6374			
2	3	1 3 4 7 10	2	1,549	10,287
2	1,6128	2,5979			
4	2,104	4,7042			
3	1,4282	3,5207			
1	3	1 3 4 7 10	3	1,4282	10,8283
ES0113900J37					
2	1,4174	6,5926			
4	1,4251	11,7404			
3	1,4245	9,1976			
0		1	2	1,4174	27,5344
2	1,4054	5,6051			
4	1,4178	11,7269			
3	1,4178	11,5917			
6	6	1	2	1,4054	28,928
2	1,4211	7,3367			

ISIN					
#N.	Error	Time(min)			
n.feat.	r.feat.	delay	#N.	Error	Time(min)
4	1,4235	11,6988			
3	1,4243	9,0878			
5	6	1	2	1,4211	28,1275
2	1,4068	9,047			
4	1,4504	11,236			
3	1,4814	11,4502			
4	6	1	2	1,4068	31,7374
2	1,4037	5,154			
4	1,4184	11,7102			
3	1,4185	10,0735			
3	6 3	1	2	1,4037	26,9416
2	1,4041	4,8524			
4	1,4054	9,5358			
3	1,4137	9,1141			
2	6 3	1	2	1,4041	23,5065
2	1,3967	3,9967			
4	1,4078	9,339			
3	1,4029	9,5504			
1	6 3 1	1	2	1,3967	22,89

Correlation coefficient

Volatility - Spread

ISIN	Cor.Coefficient	p-value
ES0113860A34	0,13388	2,11E-10
ES0113900J37	0,039551	0,045481
ES0124244E34	0,13372	1,12E-11
ES0144580Y14	0,19651	1,10E-23
ES0105200416	0,007962	0,68732
ES0111845014	0,047585	0,01609
ES0112501012	0,17357	9,47E-19
ES0113211835	0,024715	0,21145
ES0113440038	-0,01652	0,40362
ES0113679137	0,15695	1,42E-15
ES0113790531	0,065792	0,00086992
ES0115056139	0,070498	0,032892
ES0116870314	0,076107	0,00011669

ES0118594417	0,29809	1,19E-53
ES0118900010	0,018481	0,49748
ES0122060314	0,073317	0,00020621
ES0125220311	0,095577	1,28E-06
ES0130670112	0,075024	0,0001459
ES0130960018	0,040787	0,072776
ES0132105018	0,070837	0,00033646
ES0140609019	0,2021	6,12E-07
ES0142090317	0,038291	0,052822
ES0143416115	0,054041	0,0088295
ES0147200036	0,11312	7,80E-08
ES0147645016	0,29278	2,16E-12
ES0148396015	0,16937	1,09E-15
ES0152503035	-0,066887	0,011207
ES0167050915	0,03979	0,044193
ES0171996012	0,041006	0,20477
ES0173093115	0,08125	4,34E-09
ES0187165017	-0,025588	0,435
ES0178430E18	0,28555	3,41E-49
ES0182870214	0,11053	2,09E-08
LU0323134006	0,012176	0,71437
ES0113900J37	0,039551	0,045481

Volume - Spread

ISIN	Cor.Coefficient	p-value
ES0113860A34	-0,00052456	0,97984
ES0113900J37	-0,040417	0,037553
ES0124244E34	0,078864	4,85E-05
ES0144580Y14	0,061733	0,0014817
ES0105200416	0,002844	0,8837
ES0111845014	-0,031296	0,10738
ES0112501012	-0,19148	2,77E-23
ES0113211835	-0,072576	0,0001856
ES0113440038	-0,21751	1,00E-29
ES0113679137	0,083466	1,70E-05
ES0113790531	-0,0009653	0,9604
ES0115056139	-0,017841	0,57192
ES0116870314	-0,050605	0,0092011
ES0118594417	-0,062547	0,001281
ES0118900010	-0,0063826	0,80878
ES0122060314	-0,022411	0,24897
ES0125220311	0,020507	0,29148
ES0130670112	-0,0093064	0,63217

ES0130960018	-0,02226	0,31661
ES0132105018	-0,047253	0,015024
ES0140609019	0,058664	0,12396
ES0142090317	0,02407	0,21564
ES0143416115	-0,03762	0,063329
ES0147200036	0,045423	0,028239
ES0147645016	0,15232	0,00010555
ES0148396015	-0,030708	0,14096
ES0152503035	-0,028158	0,27148
ES0167050915	-0,017874	0,35787
ES0171996012	0,01916	0,53554
ES0173093115	-0,054214	7,90E-05
ES0187165017	-0,054882	0,079337
ES0178430E18	-0,11911	7,84E-10
ES0182870214	-0,048866	0,011906
LU0323134006	-0,15234	1,37E-06
ES0113900J37	-0,040417	0,037553