ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

**Международный институт экономики и финансов**

Ветров Алексей Андреевич

Анализ информационной эффективности российского рынка облигаций с помощью конвенциональных моделей и нейронных сетей

Analysis of Informational Efficiency of Russian Bond Market Via Conventional Models and Neural Networks

Выпускная квалификационная работа –

БАКАЛАВРСКАЯ РАБОТА

по направлению подготовки 38.03.01 «Экономика»

образовательная программа «**Программа двух дипломов по экономике НИУ ВШЭ и Лондонского университета**»

|  |  |
| --- | --- |
| Рецензент  PhD,  Boulatov Alexei  Москва 2019 | Научный руководитель  к.ф-м.н.,  Lapshin Victor  Студент  Vetrov Alexey |

**TABLE OF CONTENTS**

1. Introduction 4
2. Literature review 6 2.1 Efficient market hypothesis 6

2.2 EMH on Russian financial market 8

2.3 Recent advances in prediction models 8

1. Methodology 10

3.1 Sample formation 10

3.2 ARIMA, ARIMAX and GARCH 12

3.3 Neural networks 14

1. Empirical estimation 19

4.1 In-sample fitting 19

4.2 Out-of-sample performance 27

4.3 LSTM neural networks 30

4.4 Variance ratio test 34

5. Conclusion 35

6. Bibliography 36

7. Appendix 38

**Abstract**

This study addresses the Efficient Market Hypothesis in the context of Russian corporate bond market with assistance of autoregressive models and non-linear sequential connectionist systems, known as recurrent neural networks, in static and rolling settings.

**Аннотация**

В данной работе проводится проверка гипотезы эффективного рынка в контексте российского рынка корпоративных облигаций при использовании авторегрессионных моделей и рекуррентных нейронных сетей в статичном и динамическом экспериментальном дизайне.

1. **Introduction**

In 1960s Eugene Fama has published an influential article called “Efficient Capital Markets”. It suggested that capital markets are highly capable of correctly reflecting the value of an asset through the means of market price, hence it is not possible to consistently outperform randomly selected portfolio with any developed trading strategy (Fama, 1970)

From the point of allocation efficiency for funds in capital market the pricing was always a substantial concern and Fama’s proposal was beneficial in that regard, but this hypothesis, which acquired the name “Efficient Market Hypothesis” (EMH), was unpleasant news for fund managers and traders, since in the fields of long-term investing and trading the profits were supposed to be either unpredictable or capable of being increased only by increasing risk as well (Malkiel, 1973).

In Malkiel’s book “A Random Walk Down Wall Street”, the author implies that a blindfolded monkey throwing darts can reach the same goal as professional portfolio managers in competing with broad-based index.

However, the critics argued it was quite possible to execute a trading strategy in a short term, when the market is adjusting for news, for example. In response, Malkiel writes, that the opportunity window for news is short, markets react too quickly for a trader to consistently extract profit (Malkiel, 1973).

Nonetheless, the critique of this hypothesis, be it from behavioral finance, risk-factor models or high-frequency traders, and subsequent discussions proved to be of much value for EMH, as it was refined and its definitions improved to account for various information sets, price bubbles, new forecasting models and time horizons (Lee, Yen, 2008).

Evidence for EMH is mixed: there are studies supporting EMH in some form for some markets, there are studies that do not confirm the presence of some form of efficiency for other ones. This settles a dichotomy between EMH and forecasting models across all markets, developed and underdeveloped. The dichotomy is even more highlighted by the fact that in 2013 Nobel prize in economics was received by both Eugene Fama for EMH and Robert Shiller for his paper on long-term asset price predictability with rational and behavioral basis.

As new prediction models are developed, the issue of EMH rises every time. The recent addition to sequential forecasting in broad academic field are recurrent neural networks, used for speech recognition and text generation as well. So far, the studies have not provided a clear answer whether neural networks perform better out of sample than conventional models, there is evidence both for (e.g. Dingli, Fournier. 2017) and against it (e.g. Krauss et al, 2016). Long-short Term Memory networks have been shown by several studies to perform better than ARIMA models and this result is going to be tested while applied to bond spread forecasting,

The main goal is to give the insight into the applicability of the Efficient Market Hypothesis on Russian bond market and test whether recurrent neural networks are better at predicting bond spreads than linear autoregressive models. As these topics do not seem to be well-covered by academic researchers to this date, that is precisely the contribution this paper aims to provide.

The study sets the following milestones:

* Acquire or derive the time series for Russian corporate bonds
* Employ predictive models to test for weak-form efficiency
* Explore relevant fundamental factors for semi-strong form of efficiency
* Incorporate macroeconomic variables in order to test for semi-strong form of efficiency
* Formally test the series and derived predictions to reach the conclusion
* Ensure reproducibility of results

Content description is listed below:

In **Literature review** the evolution of EMH definitions is explored, also there is a brief review previous research of Russian markets and recent prediction models.

In **Methodology** section the approaches taken are thoroughly described, their origin, derivation and applicability analyzed.

**Empirical analysis** part offers sample description, model estimation and derived results, from which a **Conclusion** is reached.

**Bibliography** includes references to every article or book used or mentioned and **Appendix** section contains additional estimation outputs, which were not added to the text.

**Literature review**

2.1 The Efficient Market Hypothesis

Historical references indicate that in some form the hypothesis of efficient markets was expressed as early as in 1900 by French mathematician Louis Bachelier in his PhD thesis, as he proposed that stock prices move in Brownian motion.

For a time, his thesis was forgotten in financial academic field, until half a century later the idea was revived by Kendall(1953), Samuelson(1965) and Mandelbrot(1966), suggesting that asset prices follow the process of random walk and no consistent excess profit over holding strategy can be achieved. As computing power has greatly increased in the course of those years, the theory has become empirically testable, making advances in the field possible. The tests consisted mainly of autocorrelation and technical trading testing with respect to buy-and-hold strategy (Lee, Yen, 2008).

After his survey of empirical studies, the hypothesis in its general form was formulated by Fama in 1970, where he has summarized three distinct forms of market efficiency by degree of information availability to all market participants: weak form, semi-strong form and strong form.

Roberts(1967), Jensen(1978) and Malkiel(1992) have elaborated on definitions of those three types. By their definition, “a market is efficient with respect to the information set if it is impossible to make economic profits on the basis of ” (Jensen, 1978, p. 3), where index “t” denotes the time step.

The difference between forms of efficiency lies within content of the information set:

* Weak-form efficiency: includes historical data of time series in question up to time t.
* Semi-strong form of efficiency: also contains all publicly available information, including historical data for other financial assets, macroeconomic data up to time t and any other information gathered from public domain.
* Strong form of efficiency: absorbs insider information as well, making this set of information complete.

As each set of information contains more data than the previous one, it becomes apparent, that if market efficiency is violated on the basis of a smaller information set, it is violated for larger information sets as well. Since insider information is unavailable, this study is restricted to assessing weak form of efficiency and semi-strong form of efficiency.

Term “economic profit” in the definition denotes consideration for market frictions (Fama, 1970, p.389), that may occur and possibly negate profit, possible in pure type of experiment setting.

Those are:

* Transaction costs (broker fees erasing small profits)
* Information processing costs (wages of analysts and cost of research negate benefits)
* Market impact (supply of an asset changes because of participants’ actions)
* Trading restrictions (e.g. mutual funds have no right to enter a short position)

Due to sheer complexity of the task to account for every market friction for each year and bond this paper does not consider frictions.

The definition provided by Jensen, Malkiel and Roberts does not account for market bubbles, which prove to be present at times. In Black (1986) it is suggested that the price of an asset should not deviate from its intrinsic value by a factor of 2, an arbitrary figure. As this study’s concern is corporate bonds with substantial duration and no bubbles have been detected for the period of study (2011-2019), Black’s suggestion will not be used here.

There is, however, a significant improvement to the EMH definition, which does have more implications for this paper. Timmermann and Granger (2004) have proposed to enhance the definition by taking into account the timing of forecasting models. The idea was that some models are not available or are computationally hard to implement for the time periods analyzed in empirical studies, hence, while testing on historical data, these models could have provided excessive returns in retrospect, but it would have happened solely because of the fact that no other market participant could have used them.

The definition was then refined into “A market is efficient with respect to the information set , search technologies and forecasting models , if it is impossible to make economic profits by trading on the basis of a forecasting model in , selected with a search technology , based on predictor variables in ” with same distinctions for the information sets as above (Timmermann and Granger, 2004, p.12).

As neural networks have been reintroduced into the broad academic field not so long ago and some type of architectures have been constructed only in recent years, the availability of models has to be taken into account. Since this paper uses long-short term memory recurrent neural network (LSTM for short), the timings for data are chosen accordingly.

* 1. EMH on Russian financial markets

There are several papers assessing EMH on Russian financial markets, Russian and otherwise, but their scope is mostly limited to the stock market.

In Darushin and L’vova (2014) paper the researchers have shown that the assumption of normal distribution of stock returns was not confirmed by normality tests, neither do returns correspond to a random walk process.

Said and Harper (2015) used Ljung-Box test statistic for joint autocorrelation significance and variance ratio test for detecting a random-walk process to reject the hypothesis of market efficiency in its weak-form level for Russian stock market (period: 2003-2012).

Aristova(2010) analyses both stock and bond markets in Russia and the methodology used is test for significance for GARCH-class model coefficients (methodology described in Fedorova and Gilenko, 2008, p.32-40). It was shown that the stock market measured using MICEX index from 2003 to 2009 was weak-form efficient, but corporate bond market had been proved to be serially correlated for lags 2,3,4,5 and 10.

Overall, the results seem to be inconclusive for the stock market, while for corporate bond market they appear to be limited, which is the issue this paper aims to remedy.

* 1. Recent advances in prediction models

ARIMA models (further described in **Methodology** section) were developed by Box and Jenkins (1970) and widely used in forecasting for a long time in finance (e.g. Petrica et al, 2016) and other fields of science. However, the model does have certain limitations (A.Aryo et al, 2014), which include stationarity requirement, the assumption of conditional homoskedasticity and complexity of capturing non-linear relationships. The first issue is fixed by finding I, the order of integration, using unit-root tests. The assumption of conditional homoskedasticity could be relaxed by integrating ARIMA with ARCH or GARCH model, developed by Engle(1982) and Bollerslev(1986), respectively. The last drawback, however, remains hard to tackle, since one most likely has to identify the type of non-linearity and specify it during model construction.

Neural networks, however, are iterative models with non-linear activation function and gradient optimization, hence they are more applicable for capturing non-linearities in dependencies. They also do not require the condition of stationarity. They do have several drawbacks compared to ARIMA-GARCH, these being larger amounts of computational time, lack of interpretation for model coefficients, larger quantity of parameters (weights and biases of each neuron) and no general procedure (such as Box-Jenkins) to determine the architecture of the network best fit for the task.

Although machine-learning techniques are not a new discovery, they have been receiving increasing attention in recent years for business practices and scientific fields, including finance. Recurrent neural networks (RNN) are the type of architecture specialized for sequential data, such as speech and video recognition, text generation and time series. LSTM architecture, developed by Hochreiter and Schmidthuber (1997) overcomes the “vanishing gradient” problem associated with common RNNs, as generic RNNs value in memory only the most recent elements of a sequence, while LSTM uses a system of gates to solve the task whether to keep or forget information conveyed from previous time steps each time new data is being put into it.

There is no general conclusion whether machine learning techniques perform better in out-of-sample prediction than conventional time series models. Fischer and Krauss (2017) show that LSTM architecture outperforms memory-tree classification methods, such as random forest, fully-connected deep neural network and logistic regression classifier for S&P 500 index 1992-2015.

Another paper, Namin et al.(2018), directly compares the performance of ARIMA(5,1,0) model with LSTM in rolling setting for several stock market indices. The observed reduction in root mean square error was as large as 84 percent, however, the study did not specify the reason why that particular order of ARIMA was chosen, neither did it account for model availability at the time of the samples used (1985 – 2018), not considering the critique of such experiment design provided by Timmermann and Granger (2004). Still, this may be a significant result, hence this paper employs part of the procedure described in Namin et al.(2018) for rolling setting of out-of-sample LSTM evaluation.

**Methodology**

3.1 Sample formation

In order to get an accurate measure of Russian corporate bond market from various angles, several bond indices were downloaded from MOEX website to be transformed into time series.

The choice of time series is made in order to distinguish bonds with different durations (as bonds with lower duration could be more prone to volatility) and bonds with different rating (as the bonds with higher rating could be valued more accordingly to intrinsic value due to more plausible fundamental data). As MOEX index description considers BBB- and higher to be perceived as “investment-grade” and lower ratings “speculative”, the paper will follow that logic and set the border line there.

The derivation of index indicators is described in the file, called “metodika-index-bond.doc”, which is available on [moex.com](http://www.moex.com). For the purpose of this study one needs to extract daily data for bond yields. A translated excerpt from the file, showing the derivation for that particular indicator is shown below:

|  |  |  |
| --- | --- | --- |
|  | |  |
| Legend: | |  |
|  | * Value of yield of the index at time ; | |
|  | * Total market capitalization for the bonds included to calculation base at time *t;* | |
|  | * Mean weighted price of the bond of issue *i* at time *t*, expressed in roubles; | |
|  | * Mean weighted price of the bond of issue *i* at time *t-1*, expressed in roubles; | |
|  | * Cumulative coupon yield at time *t* for a bond of issue *i*, expressed in roubles; | |
|  | * Cumulative coupon yield at time *t-1* for a bond of issue *i*, expressed in roubles; | |
|  | * coupon yield at time *t* for a bond of issue *i*, expressed in roubles; | |
|  | * the volume of bond issue *i*, expressed in quantity of bonds. | |

The indices gathered from MOEX include:

* All corporate bonds index
* All corporate bonds with 1-year duration or lower
* All corporate bonds with duration from 1 to 3 years
* All corporate bonds with duration from 3 to 5 years
* All BBB- and higher-rated corporate bonds
* BBB- and higher-rated corporate bonds with duration from 1 to 3 years
* BBB- and higher-rated corporate bonds with duration from 3 to 5 years

However, one needs to admit that return rates of the indices is not exactly the measure to consider. After all, corporate bonds are a risky asset, hence bond yields contain both risk-free rate and risk premium. This means that one must extract the risk premia from indices by subtracting the risk-free rate.

In addition, a fitting risk-free rate must be chosen. As LIBOR and its analogues were often the risk-free rates of choice for debt market studies, for this paper the rate chosen was overnight interbank rate RUONIA, a successor of MOSIBOR after 2011. Historical data for this rate was downloaded from [ruonia.ru](http://www.ruonia.ru).

The derived bond index spreads used for time-series analysis are denoted as follows:

|  |  |
| --- | --- |
| **Notation** | **Meaning** |
| ABSPR | All corporate bonds index |
| 1YLSPR | All corporate bonds with 1-year duration or lower |
| 1T3YSPR | All corporate bonds with duration from 1 to 3 years |
| 3T5YSPR | All corporate bonds with duration from 3 to 5 years |
| BBBABSPR | All BBB- and higher-rated corporate bonds |
| BBB1T3YSPR | BBB- and higher-rated corporate bonds with duration from 1 to 3 years |
| BBB3T5YSPR | BBB- and higher-rated corporate bonds with duration from 3 to 5 years |

Table 1, Series notation (source: author)

The derived time series include daily data from 11.01.2011 to 18.01.2019.

3.2 ARIMA, ARIMAX and GARCH

Autoregressive processes *AR(p)* represent a linear regression of a dependent variable in a time series at time *t*, regressed on its own values at previous time up to lag *p*:

*Yt* = *φ*0 + *φ*1*Yt*-1 + *φ*2*Yt*-2 + … + *φpYt*-p + *εt*

*εt ~ WN(0,* *)*

Here *φ* denote coefficients for autoregression, while *εt* is white noise with zero mean.

Moving average processes *MA(q)* represent a linear regression of a dependent variable in a time series at time t, regressed on model’s own disturbance terms at previous time up to lag q:

*Yt* = *μ*0 + *εt*+ *θ*1*εt*-1 – … + *θqεt*-q

*εt ~ WN(0,* *)*

Here *θ* denote coefficients for moving average, while *εt* is white noise with zero mean and *μ* is series’ mean.

*ARIMA(p,d,q)* is a combination of autoregressive process up to lag *p* and moving average process up to lag *q*, constructed for integrated time series with degree of integration d, denoting the lowest number of differencing for time series to become covariance-stationary. For differenced series the model looks like *ARMA(p,q)* (Box, Jenkins, 1976)

*Yt* = *φ*0 + *φ*1*Yt*-1 + *φ*2*Yt*-2 + … + *φpYt*-p + *θ*1*εt*-1 + … + *θqεt*-q+ *εt*

*εt ~ WN(0,* *)*

Here *φ* denote coefficients for autoregression part, and *θ -* coefficients for moving average part, while *εt* is white noise with zero mean (Box, Jenkins, 1976).

*ARIMAX(p,d,q)* is an extension of *ARIMA* model, designed to include exogenous variables and first introduced by Box and Tiao (1975):

*Yt* = *φ*0 + *φ*1*Yt*-1 + *φ*2*Yt*-2 + … + *φpYt*-p + *θ*1*εt*-1 + … + *θqεt*-q+ + *εt*

*εt ~ WN(0,* *)*

Here *X* denotes the exogenous variables and *m* is their number used in the model.

This paper is going to employ ARIMAX as well, in order to test whether chosen macroeconomic variables have significant coefficients in-sample and whether ARIMAX has more predictive power than ARIMA out-of-sample.

*ARCH(p)*-class models (Engle, 1982) are designed to deal with heteroskedasticity up to lag *p*.



 *~ WN(0,* *)*

Here  denotes conditional variance at time *t, u* denotes squared errors of conditional mean model and *a* denotes coefficients. The coefficients are non-negative with at least one being positive and their sum has to be less than one to ensure stationarity.

Generalized *ARCH* or *GARCH(p,q)* model (Bollerslev, 1986) is a compact version of *ARCH(p)*, which is a useful parsimonious tool, especially if *ARCH* has too many lags to account for:



 *~ WN(0,* *)*

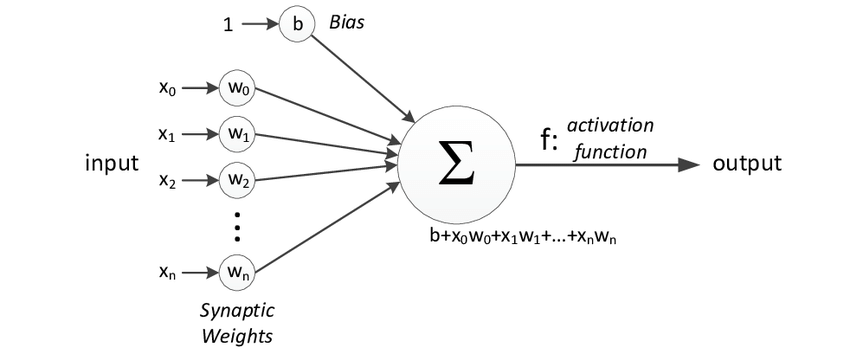
Here  denotes conditional variance at time *t, a* and *β* denote coefficients. The coefficients are non-negative with at least one being positive and their sum also has to be less than one to ensure stationarity.In practice, *q* rarely goes beyond 1, hence the model is much more compact than *ARCH*.

It is also important to note, that coefficients for models above are generally estimated using maximum likelihood, which is not the case for neural networks, where gradient descent is applied in order to find optimal weights in iterative way.

3.3 Neural networks

*Rosenblatt’s perceptron*

Neural networks consist of individual neurons, which are generally simple mathematical representations of a biological neuron. The concept of perceptron was developed by Rosenblatt (1957).



*Figure 1, Perceptron (source: Fountas, 2011)*

The model computes a weighted sum of its inputs and, if the sum passes a set threshold, the neuron activates with the output being the activation function of the weighted sum. Bias serves the same purpose as the intercept term does in regressions. Furthermore, one can simulate *ARMA(p,q)* using this perceptron with linear activation function *y = f(x) = x,* whileprevious *p* values of a series and past *q* deviations from mean serve as inputs.

*Gradient descent, a brief overview*

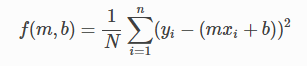
However, the perceptron does not use maximum likelihood to determine its coefficients (weights and bias). Instead, an iterative technique called “gradient descent” is used.

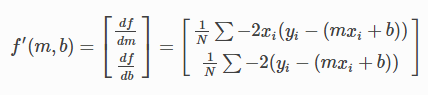
One needs to define a cost function, find its partial derivatives for each parameter and update the weights afterward with steps being proportionate to the negative of the gradient, as one is interested in finding the local minimum.

θ = θ − η⋅J(θ)

Here θ denotes the parameters and η is the learning rate, which is designed to manage how large an update will be for one iteration.

For one-input linear model *y = mx + b* with *N* observations and the cost function being Mean Square Error (MSE), the process described looks like:





*Source:* [*https://ml-cheatsheet.readthedocs.io/en/latest/gradient\_descent.html*](https://ml-cheatsheet.readthedocs.io/en/latest/gradient_descent.html)

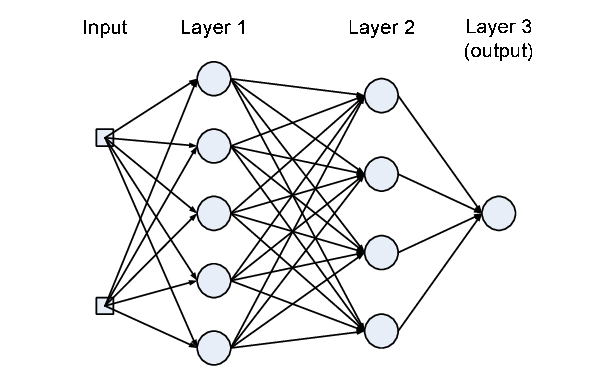
This process of gradient descent uses the whole sample to compute derivatives. One step of training is called an epoch. However, if a sample is too large, the computational time required for may be quite long. In order to tackle the problem, stochastic gradient descent was coined, which uses a random sample of given length taken from the data. If a sample is random, the technique achieves the same results with less time requirements. Epoch in this case is a learning step, when all the training examples have been used in it.

Another addition to gradient descent techniques is implementation of adaptive learning rates in order to reduce fluctuations around local minimum and accelerate the optimization towards relevant direction. Full survey of current evolutionary stage of gradient optimization techniques can be found in Ruder (2016). The most widely used technique at the time this paper was written is Adam (Kingma, Ba, 2014), which, quoting Ruder (2016, p.7), “in addition to storing an exponentially decaying average of past squared gradients, also keeps an exponentially decaying average of past gradients”. This paper uses Adam as gradient descent algorithm due to its prominence in academic field.

The tasks solved with this perceptron include linear regression and binary classification. However, the perceptron has its limitations, described in Minsky and Papert (1969), which include incapability of making global generalizations based on local data. Hence, the perceptrons were dismissed for a time, until more advanced architectures have arrived.

*Fully-connected neural networks*

Fully-connected neural networks are a system of perceptrons with non-linear activation function (usually a sigmoid), where neurons are stacked in layers with outputs of previous layer being the inputs for the next one. The name “fully-connected” originates from the fact that each neuron has inputs from all the neurons on the previous layer. The gradient descent is performed by backpropagation, computation of partial derivative of the cost function for each parameter in each neuron using chain rule, going back from the last layer to the first and updating weights.



*Figure 2, a fully-connected model (source: Aliaga (2009))*

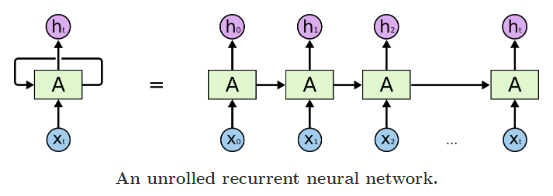
This type of neural network can deal with non-linear regression problems quite well. Due to iterative nature of parameter optimization one does not need to specify the type of non-linearity.

One can visually explore how this type of network tackles non-linear regression for simulated data at <https://playground.tensorflow.org/>.

Still, if one considers sequential data, for this design of neural network to capture effects of a certain lag one needs to add this lag to inputs in a way that was shown above for perceptron ARMA simulation. It becomes apparent, that one needs a network with memory of previous steps for this task.

*Recurrent neural networks (RNNs)*

RNNs are designed with that specific goal in mind. Their architecture includes loops, which allows the network to feed information from previous time steps into itself at the next step:

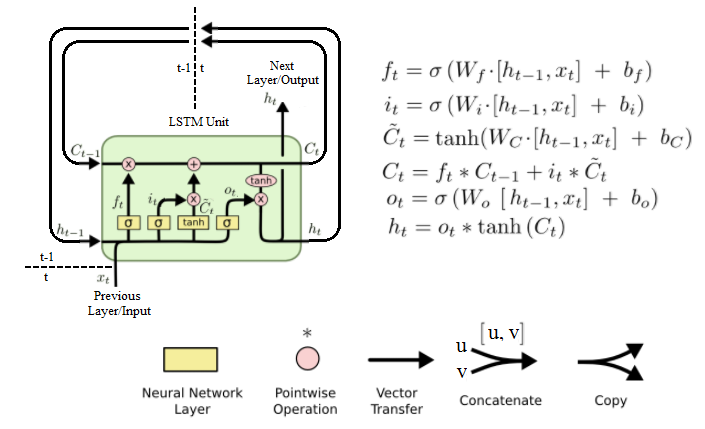


*Figure 3, RNN (source: http://colah.github.io)*

Despite RNN being fit for sequential data, it was unable to effectively capture long-term dependencies, as described in Bengio et al (1994). That happens due to vanishing gradient problem, as inputs used later in time come from earlier time points and, if for some lags gradients are small, the weights will cease to update properly, hence the model will be inefficient, if a series has long-term dependencies (e.g. if a word in the beginning of a sentence determines the end of this sentence).

*Long-short term memory neural networks (LSTM)*

LSTM is a type of RNN, which remedies the problem of vanishing gradients by using a system of gates within a cell. Developed by Hochreiter and Schmidthuber (1997), it contains a system, which allows it to forget or to pass on previous inputs, using gradient descent optimization for each gate. The structure is depicted below:



*Figure 4, LSTM structure ()*

Here is the cell state at time *t*, is the hidden state, is the function for the “forget” gate, assigning importance to each input using sigmoid function, which takes values from 0 to 1, is input gate layer, which determines how important the current input is, using sigmoid function; denotes a vector of candidate values of current input, transformed with hyperbolic tangent function (which is scaled by afterwards). After new information is added to the previous cell state, which has passed through results of forget gate, the values are passed on to the next time period of the cell state, but for the hidden state the values are copied, squeezed between -1 and 1 by *tanh* and scaled according to derived importance of current inputs. Hence, the problem of vanishing gradients is eliminated by using a “forget” gate and determining the importance of each input at every time step, using backpropagation through time.

Full graphical walkthrough of LSTM is available at <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>.

For the goals of this paper LSTM is going to be used for estimating out-of-sample values of bond spreads.

**Empirical estimation**

4.1 In-sample fitting

*Procedure and environment*

In this section two forms of market efficiency and performance of several models are put to empirical test.

For the prediction models the paper considers both **static** and **rolling** setting:

* Static setting: the data is split into training and testing portions, the models are being trained on a training set. Their hyperparameters and their coefficients become fixed. Afterwards, they are being fed testing data on day-to-day basis for one-step out-of-sample prediction. The predictions are then evaluated.
* Rolling setting: the data is split into training and testing portions, the models are being trained on a training set. Their hyperparameters become fixed, **but** their coefficients are constantly updated to account for the data added. Afterwards, they are being fed testing data on a day-to-day basis for one-step out-of-sample prediction. The predictions are then evaluated.

Rolling setting and neural networks show the need for programming environment.

The integrated development environment of choice was python-based Jupyter notebook, as it has all the tools necessary to assess both models. For LSTM Keras library is used, as it proves to be the most convenient one for academic studies. For conventional models statsmodels package, native to R as well, is used.

For reproducibility of results full code with commentary is available at: <https://github.com/alexeyavetrov/emh_ru_cb>

*Data processing*

Recalling the notations for bond time series:

|  |  |  |
| --- | --- | --- |
| **Series №** | **Notation** | **Meaning** |
| 1 | ABSPR | All corporate bonds index |
| 2 | 1YLSPR | All corporate bonds with 1-year duration or lower |
| 3 | 1T3YSPR | All corporate bonds with duration from 1 to 3 years |
| 4 | 3T5YSPR | All corporate bonds with duration of 3 to 5 years |
| 5 | BBBABSPR | All BBB- and higher-rated corporate bonds |
| 6 | BBB1T3YSPR | BBB- and higher-rated corporate bonds with duration from 1 to 3 years |
| 7 | BBB3T5YSPR | BBB- and higher-rated corporate bonds with duration from 3 to 5 years |

*Table 1.1, Series notation (source: author)*

The series derived correspond to Russian corporate bond spreads and contain daily data from 11.01.2011 to 18.01.2019. All the time series are expressed in basis points (equivalent to 0,01 percent or 0.0001). The descriptive statistics is shown below:



*Table 2, original series descriptive statistics (source: author’s calculations)*

The data is balanced, but the minimum statistic shows a value much below zero. This seems suspicious, since investors should be compensated for holding a risky asset. It would be of importance to take a look at the graphs.

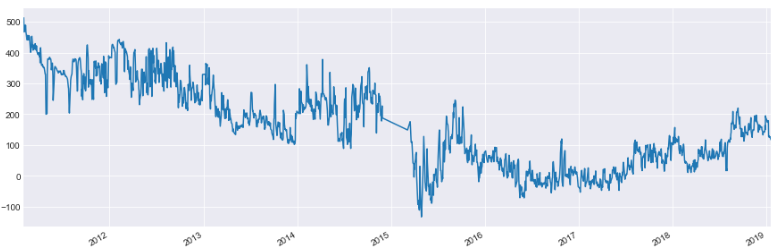
The graphs for all the time series reveal a sharp drop below zero for a time in December 2014:



*Figure 5, All-bonds index spread graph (source: author)*

It seems, some outliers are present because of regulatory measures implemented during the Black Tuesday on 16th December 2014, when the Central Bank of Russia has set the key interest rate to 17%, a 6.5% increase in a single day, following the currency crisis.

As regulatory measures are not considered to be an intrinsic property of the time series, the outliers above 3 standard deviations (an arbitrary choice) from the mean are excluded from analysis.

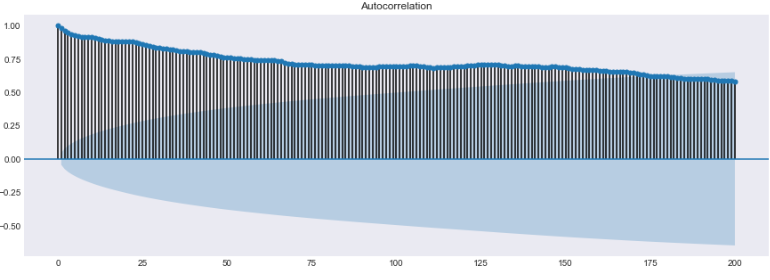


*Figure 6, All-bonds index with outliers removed (source: author’s calculations)*

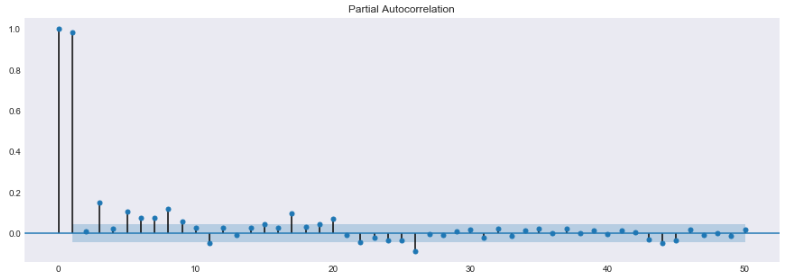


*Table 3, series’ descriptive statistics, outliers removed (source: author’s calculations)*

Afterwards, the paper follows Box and Jenkins procedure to determine the best fitting ARIMA model.



*Figure 7, autocorrelation plot for ABSPR (source: author’s calculations)*



*Figure 8, partial autocorrelation plot for ABSPR (source: author’s calculations)*

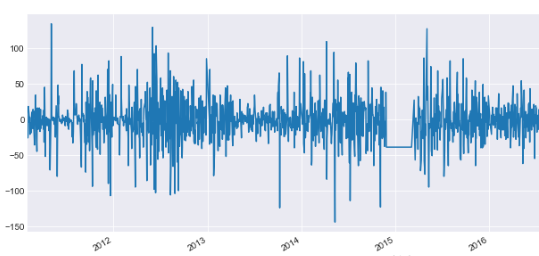
The autocorrelation plots suggest the presence of both AR and MA processes. However, one has to account for stationarity and train-test data split.

*Stationarity check*

As the models are fit on the training set of data by experiment design, one needs to define train-test split first. Here, 70% of data (01.2011 - 09.2016) is used for training purposes and 30% (10.2016 – 01.2019) is used for out-of-sample evaluation. That way one ensures, that all models are available and widely used by the time of the beginning of the test period.

In order to test the time series for stationarity, Augmented Dickey-Fuller test (Fuller, 1976) is implemented with H0 being the presence of a unit root.

The results show, that the presence of a unit root is not rejected for all the time series at 5% significance level (see Appendix for exact results). All time series in question seem to be difference stationary, indicating that integration term for ARIMA is 1.



*Figure 9, differenced ABSPR-train (source: author’s calculations)*

*(Note: the straight line on the graph is not interpolation within the data itself, it is only on the graph to keep the date intervals regular)*

*ARIMA fitting*

In Namin et al.(2018) the authors did not specify the reasons why the order of ARIMA(5,1,0) was picked. In this paper it is chosen to select the model with the lowest information criterion in order to reduce the possibility of overparametrization. The Schwarz information criterion (Schwarz, Gideon, (1978)) or Bayesian information criterion seems to be the one of choice, since it penalizes the quantity of parameters more, than Akaike information criterion and there is evidence the models chosen with it perform better out of sample (Medel, Salgado, 2012). The program iterates the models for each time series and finds the order with the lowest BIC value. The results for ARIMA order are as follows:



*Figure 10, ARIMA order for every time series in the training set (source: author’s calculations)*

After the chosen ARIMAs were fit, a residual check was in order. The residuals for every series are stationary (ADF-test), appear to be centered around 0 and are bell-shaped with a higher kurtosis than one of Gaussian normal distribution, a common occurrence for financial time series. Additionally, the residuals will be analyzed for autocorrelation presence in GARCH section.

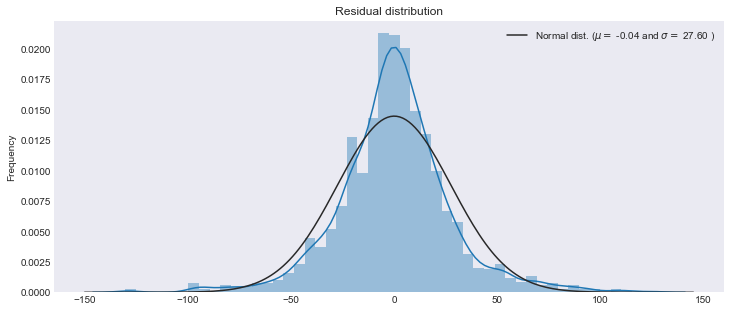


Figure 11, residual distribution for ARIMA(2,1,2) fit on ABSPR-train (source: author’s calculations)

*ARIMAX fitting*

The procedure for ARIMAX fitting is the same, with addition of exogenous explanatory variables.

This is done in order to test for semi-strong type of efficiency and, possibly, add significant predictor variables. Diebold et al. (2008) uses the measures of inflation and business cycle in order to construct global yield curve, hence the measures of these, namely Confidence Leading Indicator and Consumer Price Index, were retrieved from Thomson Reuters Eikon database.

As these indicators are monthly data, they have been interpolated into daily time series with value for the first day of the month being the value for every day in the month in order not to use future data for interpolation, as it would have distorted the validity of the analysis.

After the same procedure was used to choose order of ARIMAX, the results are:



*Figure 12, ARIMAX order for every time series in the training set (source: author’s calculations)*

The result seems to be the same as the one for ARIMA, possibly because of information criterion choice. As for residuals, they are also stationary for all series and keep the same properties as ones of ARIMA fitting. Model summaries indicate, that coefficient for CPI is statistically significant for every series.

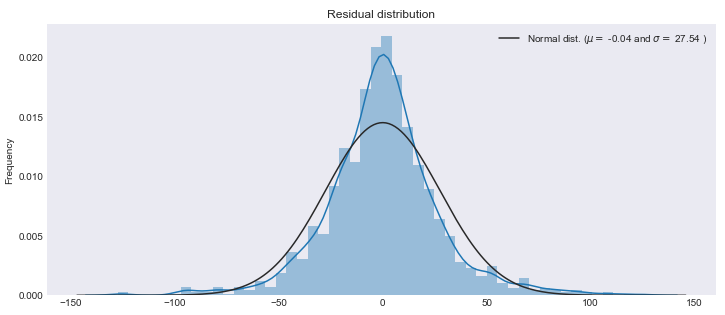


Figure 13, residual distribution for ARIMAX(2,1,2) fit on ABSPR-train (source: author’s calculations)

*Conditional variance*

For estimating conditional variance, the residuals of ARIMA and ARIMAX need to be checked for the presence of ARCH-effects. Ljung-Box test is applied to testing joint autocorrelation up to the lag of 10 (see Appendix for p-values) and the results show, that for significance level of 0.05 only 1YLSPR does not show significant ARCH effects, both for ARIMA and ARIMAX residuals. Hence, the work proceeds to GARCH estimation.

GARCH models are evaluated using the same process, which was described earlier, on the time series of squared residuals. The model with the lowest BIC is picked.



*Figure 14, GARCH order for every time series with ARIMA residuals (source: author’s calculations)*

**

*Figure 15, GARCH order for every time series with ARIMAX residuals (source: author)*

The results seem to be in accordance with literature, as GARCH(1,1) is rarely surpassed by performance. The only exception in this case is GARCH(1,2) for ARIMAX(1,1,2) residuals of BBB1T3SPR time series. GARCH coefficients are significant for 6 out of 7 series, confirming findings of Aristova (2010).

Summarized results of train-data model fitting are:

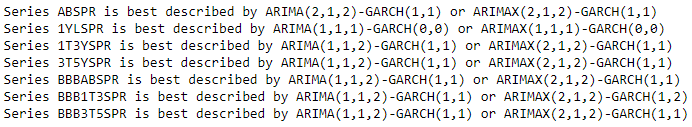


Figure 16, conditional mean and variance model fitting hyperparameters (source: author’s calculations)

*Bond spreads, CDS and Granger causality*

One of important issues when considering the debt market is the link between credit-default swaps and bonds. O’Kane (2012) showed a link between CDS prices and sovereign bond spreads for several EU countries, where for some countries one-sided Granger causality was detected, with both directions in effect, while for some two-sided Granger causality was present.

For this paper the possibility of a link between CDS and bond may provide a new predictor variable, if CDS prices Granger-cause bond spreads. Semi-strong form of efficiency would be violated in that case. Hence, daily data for CDS price for 1Y, 3Y and 5Y was extracted from Thomson Reuters Eikon database.

Every time series was shown to be difference-stationary with help of ADF-test (see Appendix), hence the data was differenced to achieve covariance stationarity required for the test. Afterwards, the Granger-causality test (Granger, 1980) was implemented for lags up to 10. The results are below:

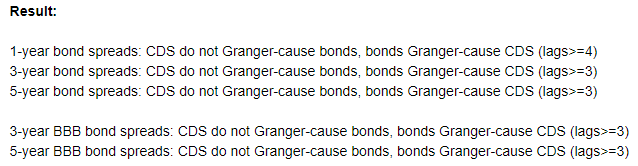


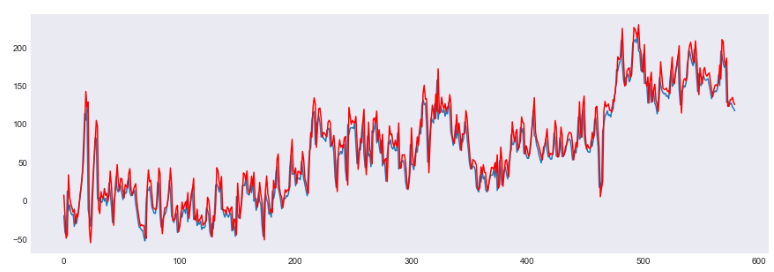
Figure 17, Granger-causality test results for bond spreads and CDS price (source: author’s calculations)

The result indicates, that no previous CDS spreads can help in predicting bond spreads.

4.2 Out-of-sample performance

*Static ARIMA*

This is the setting, where both hyperparameters and coefficients are fixed and 1-step ahead predictions are evaluated. The metrics used are MSE and RMSE, which is root of mean-square error, designed to both penalize larger errors and be expressed in same units as predicted variable.



*Figure 16, Static ARIMA one-step ahead predictions for ABSPR-test (source: author’s calculations)*

*Note: predictions are plotted using red color*

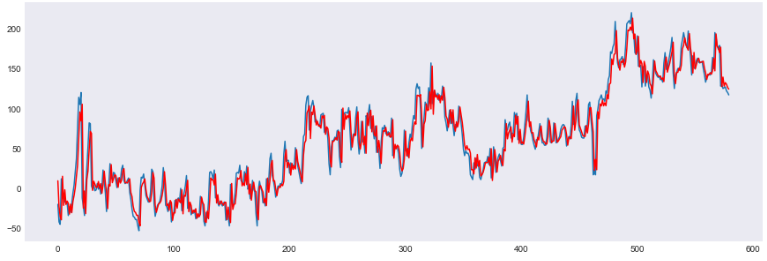
The results are depicted below (rounded to 2nd decimal):

|  |  |  |
| --- | --- | --- |
| **Series name** | **MSE** | **RMSE** |
| ABSPR | 438.48 | 20.94 |
| 1YLSPR | 2496 | 49.96 |
| 1T3YSPR | 520.75 | 22.82 |
| 3T5YSPR | 381.03 | 19.52 |
| BBBABSPR | 471,76 | 21.72 |
| BBB1T3YSPR | 633.53 | 25.17 |
| BBB3T5YSPR | 528.54 | 22.99 |

*Table 4, Static ARIMA OOS metrics, (source: author’s calculations)*

*Rolling ARIMA*

This is the setting, where hyperparameters are fixed, but the coefficients are reevaluated for each time step in test sample. One-step ahead predictions are evaluated. The metrics used are MSE and RMSE.



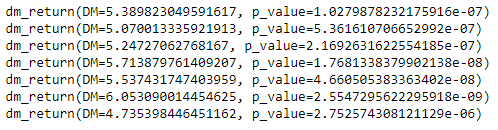
*Figure 16, Rolling ARIMA one-step ahead predictions for ABSPR-test (source: author’s calculations)*

*Note: the predicted values are plotted red*

|  |  |  |
| --- | --- | --- |
| **Series name** | **MSE** | **RMSE** |
| ABSPR | 311.52 | 17.65 |
| 1YLSPR | 1142.44 | 33.8 |
| 1T3YSPR | 369.4 | 19.22 |
| 3T5YSPR | 274.9 | 16.58 |
| BBBABSPR | 330.51 | 18.18 |
| BBB1T3YSPR | 423.54 | 20.58 |
| BBB3T5YSPR | 384.16 | 19.6 |

*Table 5, Rolling ARIMA OOS metrics, (source: author’s calculations)*

One can conclude, that rolling ARIMA is of superior performance compared to static one. The formal test to determine whether this is a valid assumption is Diebold-Mariano test (Diebold, Mariano (1994)).

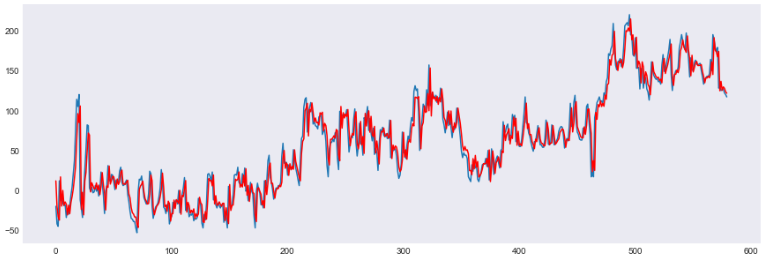


*Figure 17, DM-test Rolling ARIMA vs Static ARIMA (source: author’s calculations)*

The results are conclusive for every reasonable significance level, rolling ARIMA performs better out of sample, than the static one.

*Rolling ARIMAX*

Same rolling experiment design used with addition of exogenous predictors.



*Figure 16, Rolling ARIMAX one-step ahead predictions for ABSPR-test*

*(source: author’s calculations)*

|  |  |  |
| --- | --- | --- |
| **Series name** | **MSE** | **RMSE** |
| ABSPR | 309.76 | 17.6 |
| 1YLSPR | 1141.76 | 33.79 |
| 1T3YSPR | 369.02 | 19.21 |
| 3T5YSPR | 274.89 | 16.58 |
| BBBABSPR | 327.97 | 18.11 |
| BBB1T3YSPR | 423.54 | 20.58 |
| BBB3T5YSPR | 382.2 | 19.55 |

*Table 6, Rolling ARIMAX OOS metrics, (source: author’s calculations)*

It seems, not much has been altered in the resulting metrics. Diebold-Mariano test confirms this with p-values above 0.5 (see Appendix). Although the coefficient for CPI is significant for every time series, ARIMAX does not perform much better out-of-sample.

4.3 LSTM neural networks

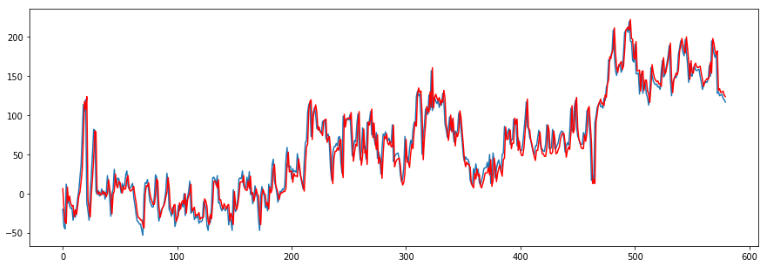
One of the drawback neural networks have is that one does not have a straightforward methodological procedure in order to find the best architecture and hyperparameters and has to repeat the search, often computationally costly, for each task. For this paper 17 hours of computation time on google cluster with GPU acceleration were needed to determine the architecture used, which may not prove to be the best one in existence. Additional material used for model construction was J.Brownlee(2017).

Testing on mock time series (trend-stationary series and sine wave with trend) proved to be a valuable tool. For mock series rolling model has vastly outperformed static one.

However, for bond spreads the results were more intriguing.

*Static LSTM*

Weights and biases are being trained in an iterative process, fixed after training on 70% of the data. Then one-step ahead predictions are acquired and evaluated. Number of LSTM cells is set to 50, the layers are 1 LSTM and 1 output, number of epochs is set to be 200, activation function is ReLU (*f(x) = max(0, x)*), optimization technique is “adam”.

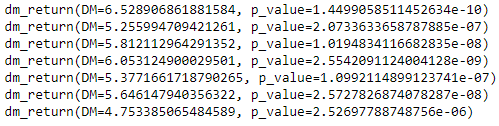


*Figure 17, Static LSTM one-step ahead predictions for ABSPR-test (source: author’s calculations)*

|  |  |  |
| --- | --- | --- |
| **Series name** | **MSE** | **RMSE** |
| ABSPR | 361.68 | 18.5 |
| 1YLSPR | 1249.62 | 35.35 |
| 1T3YSPR | 406.43 | 20.16 |
| 3T5YSPR | 308 | 17.55 |
| BBBABSPR | 362.52 | 19.04 |
| BBB1T3YSPR | 460.53 | 21.46 |
| BBB3T5YSPR | 402.4 | 20.06 |

*Table 7, Static LSTM OOS metrics, (source: author’s calculations)*

Static LSTM seems to perform better than static ARIMA in out-of-sample setting. DM-test confirms this:

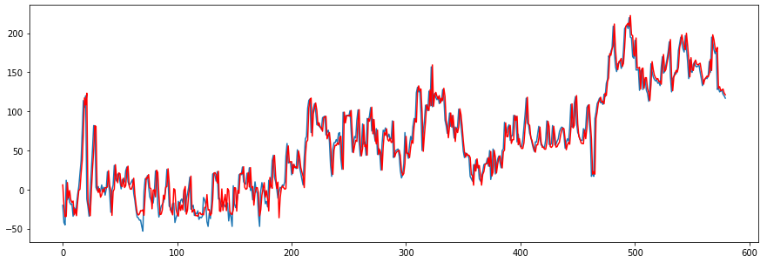


*Figure 17, DM-test Static LSTM vs Static ARIMA (source: author’s calculations)*

*Rolling LSTM*

The setting here is similar to the one used in Namin et al.(2018). However, their finding of unimportance of number of epochs for out-of-sample performance was found to be not applicable to these particular time series.

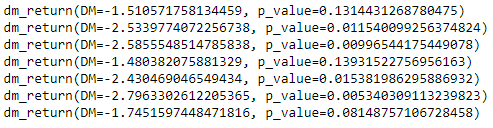
Weights and biases are being trained in an iterative process for 200 epochs each time step after new data is fed into the network. Then for each time step one-step ahead predictions are acquired and evaluated. Number of LSTM cells is set to 50, the layers are 1 LSTM and 1 output, number of epochs is set to be 200, activation function is ReLU, optimization technique is “adam”.



*Figure 18, Rolling LSTM one-step ahead predictions for ABSPR-test (source: author’s calculations)*

|  |  |  |
| --- | --- | --- |
| **Series name** | **MSE** | **RMSE** |
| ABSPR | 344.1 | 18.55 |
| 1YLSPR | 1357.19 | 36.84 |
| 1T3YSPR | 409.65 | 20.24 |
| 3T5YSPR | 321.48 | 17.93 |
| BBBABSPR | 428.9 | 20.71 |
| BBB1T3YSPR | 498.18 | 22.32 |
| BBB3T5YSPR | 475.68 | 21.81 |

*Table 8, Rolling LSTM OOS metrics, (source: author’s calculations)*



*Figure 19, DM-test Static LSTM vs Static ARIMA (source: author’s calculations)*

A curious result was acquired: Rolling LSTM did actually perform worse than Rolling ARIMA for 4 out of 7 time series, namely 1YLSPR, 1T3YSPR, BBBABSPR and BBB1T3SPR, which correspond to bonds with <1-year duration, bonds with 1-3 year duration, all BBB- and higher bonds and BBB- and higher bonds with 1 to 3 year duration.

Rolling LSTM also generated lower quality predictions compared to Static LSTM, which suggests a substantial degree of overfitting, since there are a lot of parameters to be estimated. However, the standard remedies, such as including dropout, have not produced a substantial effect.

Either the right model architecture has not been found yet, or the whole class of LSTM models is underperforming compared to ARIMA-GARCH in case of Russian corporate bond market.

Another explanation might be that one or several non-linear dependencies exist in the time series, which are better captured by static LSTM and changing coefficients of rolling ARIMA, but not the static ARIMA.

The paper can be extended and improved further, using the provided code.

4.4 Variance ratio test

Variance Ratio test, with its variations described in Charles, Darné (2009) is used extensively in EMH literature.

For the market to be efficient in a weak form, the difference of spreads must follow a random walk with zero mean, which is the null hypothesis for the test employed.

|  |  |  |
| --- | --- | --- |
| **Series name** | **Test statistic** | **p-value** |
| ABSPR | -15.757 | 0.000 |
| 1YLSPR | -14.502 | 0.000 |
| 1T3YSPR | -15.957 | 0.000 |
| 3T5YSPR | -15.613 | 0.000 |
| BBBABSPR | -15.902 | 0.000 |
| BBB1T3YSPR | -16.701 | 0.000 |
| BBB3T5YSPR | -15.509 | 0.000 |

*Table 9, Variance Ratio test results (source: author’s calculations)*

The results of Variance Ratio test have p-value very close to zero for every series in consideration, hence rejecting weak-form of efficiency for corporate bonds market in Russia.

**Conclusion**

In the course of this paper two classes of prediction models were analyzed with measurement of prediction efficiency judged by results of Diebold-Mariano test.

A program was developed, which can be used to enhance this study.

Although LSTM neural network used in this paper performed significantly better in static experiment setting, in rolling setting ARIMA has proved to be a clear winner.

In addition, Ljung-Box test has indicated that serial correlation is present and it was determined that each series is best described by combined ARIMA-GARCH model, hence, the claim in Aristova (2010) has gained further support.

The variance test has rejected the hypothesis of series following a process of random walk as well, providing additional evidence.

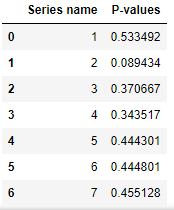
That being said, it becomes apparent that weak-form efficiency of Russian corporate bond market is rejected and, subsequently, other forms of efficiency are rejected as well.

**BIBLIOGRAPHY**

1. Е.В.Аристова, Анализ эффективности финансового рынка в России, 2010, *Финансы и кредит*
2. И.А. Дарушин, Н.А. Львова, Оценка эффективности российского финансового рынка непараметрическим методом, 2014, *Финансы и кредит*
3. Федорова Е.А., Гиленко Е.В. Методология оценки изменения информационной эффективности фондового рынка / *Финансы и кредит. 2008. № 33. С. 32–40*.
4. О.Шади, Анализ эффективности российского фондового рынка, 2017, *Финансы, денежное обращение и кредит*
5. A.Ariyo, A.O.Adewumi, C.K.Ayo, Stock price prediction using the ARIMA model, 2014, *16th international conference on computer modelling and simulation*.
6. Y.Bengio, P.Simard, P.Frasconi, Learning Long-Term Dependencies with Gradient Descent is Difficult, 1994
7. Black, Fischer, 1986, Noise, *Journal of Finance*
8. Bollerslev T., (1987), Generalized Autoregressive Conditional Heteroskedasticity, *Journal of Econometrics, 31, pp.307-327*
9. Box, G. and Jenkins, G. (1970) Time Series Analysis: Forecasting and Control. *Holden-Day, San Francisco*
10. Box, G. E. P and Jenkins, G.M., (1976). “Time series analysis: „Forecasting and control,” *Holden-Day, San Francisco*
11. Box, G.E.P. and Tiao, G.C. (1975) Intervention Analysis with Applications to Economic and Environmental Problems. *Journal of the American Statistical Association, 70, 70-79*
12. J.Brownlee, 2017, Deep Learning for Time Series Forecasting
13. Amélie Charles, Olivier Darné. Variance ratio tests of random walk: An overview. *Journal of Economic Surveys, Wiley, 2009, 23 (3), pp.503-527*
14. Francis X. Diebold & Robert S. Mariano, 1994. "Comparing Predictive Accuracy", *NBER Technical Working Papers 0169, National Bureau of Economic Research, Inc*
15. F.X.Diebold, Global yield curve dynamics and interactions: A dynamic Nelson-Siegel approach, 2008, *Journal of Econometrics*
16. A.Dingli and K.S.Fournier, 2017, *International Journal of Machine Learning and Computing*
17. Engle R.F. (1982) Autoregressive conditional heteroskedasticity with Estimates of the Variance of UK inflation, Econometrica, 50
18. Eugene F.Fama, Efficient Capital Markets: A review of Theory and Empirical Work, 1970, *Journal of Finance, Volume 25 Issue 2*
19. T.Fischer, C.Krauss, Deep learning with long short-term memory networks for financial market predictions, 2017, *Econstor*
20. Z.Fountas, Spiking Neural Networks for Avatar Control in a Simulated Environment
21. Fuller, W. A. (1976). Introduction to Statistical Time Series. New York: *John Wiley and Sons*
22. Granger, C. W. J. 1980 Testing for causality: A personal viewpoint. Journal of Economic Dynamics and Control 2, 329-352
23. S.Hochreiter, J.Schmidthubder, Long short-term memory, 1997, *Neural computation 9*
24. Jensen, Some anomalous evidence regarding market efficiency, 1978, *Journal of Financial Economics, 6, 95-101*
25. Kendall, M. (1953) The Analysis of Economic Time Series, Part I: Prices. *Journal of the Royal Statistical Society, 96, 1953*
26. D.P.Kingma, J.L.Ba, 2015, Adam: a Method for Stochastic Optimization
27. Christopher Krauss, Xuan Anh Do, Nicolas Huck, Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500, *European Journal of Operational Research* (2016)
28. B.G.Malkiel, A Random Walk Down Wall Street, 1973, *W.W. Norton and Company, Inc*
29. Malkiel, B. 1992, Efficient market hypothesis, in P. Newman, M. Milgate and J. Eatwell (eds.), *New Palgrave Dictionary of Money and Finance, Macmillan, London*
30. B.G.Malkiel, The Efficient Market Hypothesis and Its Critics, 2003, *Journal of Economic Perspectives*
31. Mandelbrot, B (1966) Forecasts of Future prices, unbiased markets and martingale models, *Journal of Business, 39*
32. C.Medel, S.Saldago, Does the BIC estimate and forecast better that AIC?
33. M.Minsky, S.A.Papert, 1969, Perceptrons: An Introduction to Computational Geometry
34. A.S.Namin, S.Namini, N.Tavakoli, A Comparison of ARIMA and LSTM in Forecasting Time Series, 2018
35. O’Kane, The Link between Eurozone sovereign debt and CDS prices, *EDHEC Risk Institute*, 2012
36. A.E.Petrica, S.Stancu, A.Tindeche, *Theoretical and Applied Economics Volume XXIII (2016), No. 4(609), Winter, pp. 19-42*
37. Roberts, H., 1967, Statistical versus clinical prediction of the stock market
38. F.Rosenblatt, 1957, The perceptron: a probabilistic model for information storage and organization in the brain, *Psychological Review*
39. S.Ruder, An overview of gradient descent optimization algorithms, 2017
40. Samuelson P.A, Proof that properly anticipated prices fluctuate randomly, 1965, *Industrial Management Review, 6*
41. Schwarz, Gideon E. (1978), "Estimating the dimension of a model", *Annals of Statistics, 6 (2): 461–464*
42. Timmermann, A. and Granger, C. W. J. (2004), Efficient market hypothesis and forecasting, *International Journal of Forecasting 20(1), 15–27*
43. G.Yen, C.Lee, Efficient Market Hypothesis (EMH): Past, Present and Future, 2008, *Review of Pacific Basin Financial Markets and Policies*
44. <https://github.com/alexeyavetrov/emh_ru_cb>
45. <http://colah.github.io>
46. <https://playground.tensorflow.org/>
47. <https://www.rbc.ru/finances/16/12/2014/548f58d72ae5966d31a64d76>
48. [moex.com](http://www.moex.com)
49. ruonia.ru
50. [*https://ml-cheatsheet.readthedocs.io/en/latest/gradient\_descent.html*](https://ml-cheatsheet.readthedocs.io/en/latest/gradient_descent.html)

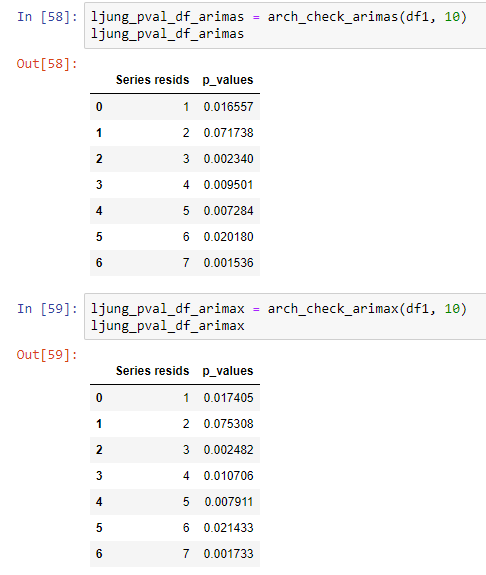
**Appendix**

*Additional estimation outputs*



*Appendix: Figure 1, ADF-test results for bond spread training time series*

*(Source: author’s calculations)*



*Appendix: Figure 2, Ljung-Box test results for joint autocorrelation up to lag 10 for residuals of ARIMA and ARIMAX (source: author’s calculations)*

CDS1Y

ADF Statistic: -2.249418

p-value: 0.188801

Non-stationary, H0 is not rejected

CDS3Y

ADF Statistic: -2.544467

p-value: 0.105026

Non-stationary, H0 is not rejected

CDS5Y

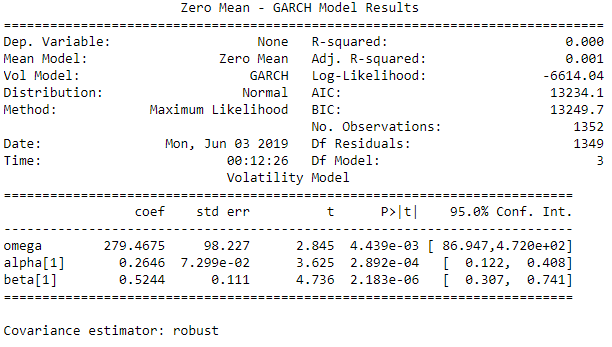
ADF Statistic: -2.504283

p-value: 0.114436

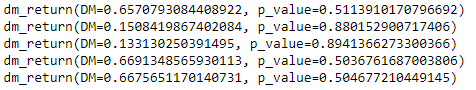
Non-stationary, H0 is not rejected

*Appendix: Figure 3, ADF-test results for CDS series*

*(source: author’s calculations)*

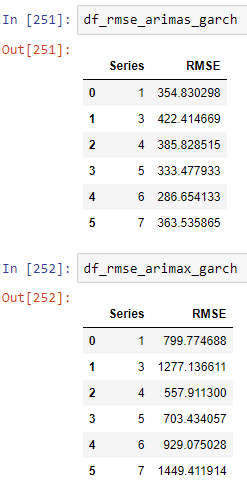


*Appendix: Figure 4, GARCH model results and coefficient significance for ABSPR training series (source: author’s calculations)*

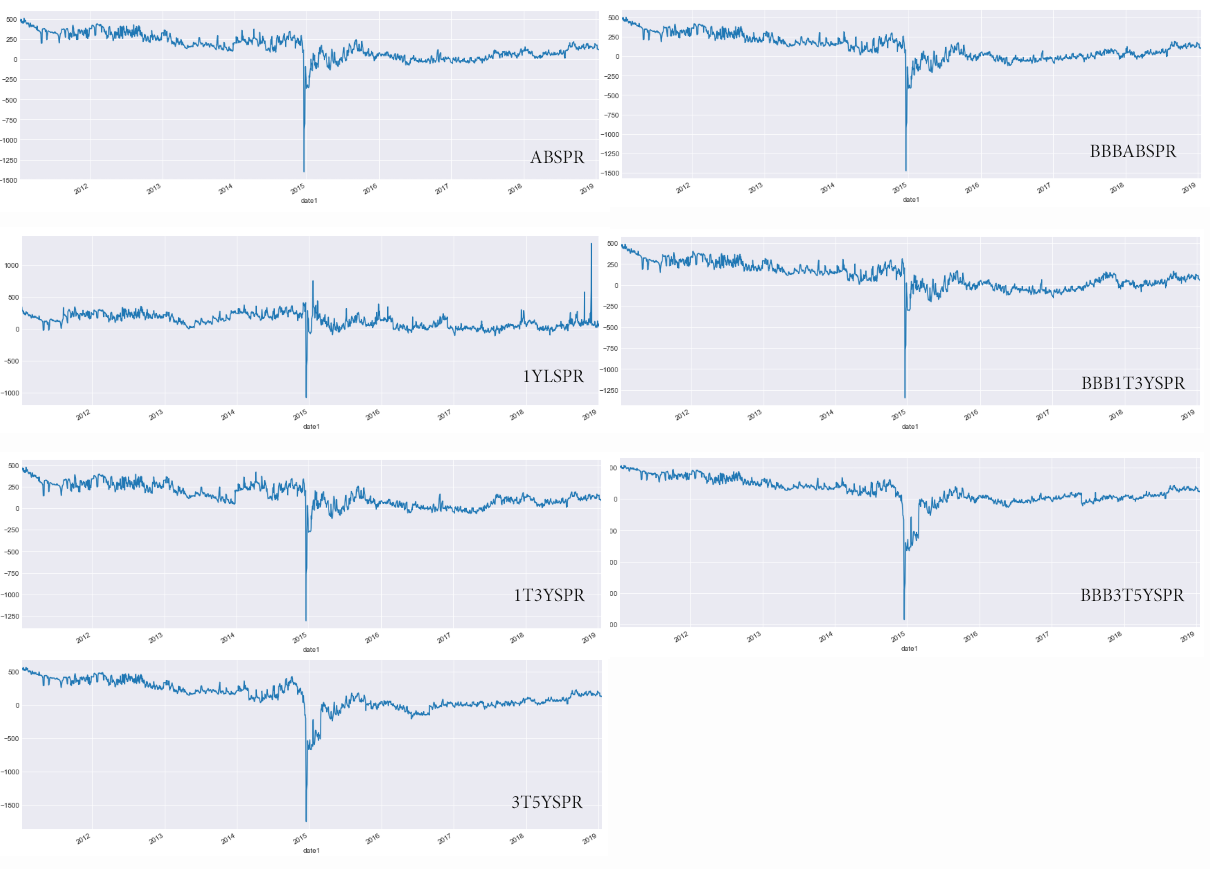


*Appendix: Figure 5, DM-test for prediction performance of ARIMAX over ARIMA*

*(source: author’s calculations)*



*Appendix: Figure 6, prediction evaluation for Rolling GARCH for residuals of ARIMA and ARIMAX (source: author’s calculations)*



*Appendix: figure 7, original time series for bond spreads (source: author’s calculations)*