

# Traffic Data Prediction

Ing. Denis Varaksin

---

Master's thesis  
2019



Tomas Bata University in Zlín  
Faculty of Applied Informatics

---

# **MASTER'S THESIS ASSIGNMENT**

(PROJECT, ARTWORK, ARTISTIC PERFORMANCE)

Degree, First Name and Surname: **Denis Varaksin**  
Personal Code: **A16802**  
Degree Programme: **N3902 Engineering Informatics**  
Degree Course: **Information Technologies**

Thesis Topic: **Traffic Data Prediction**  
Thesis Topic in Czech: **Predikce dopravních dat**

## Thesis Guidelines:

1. **At the beginning, meet and discuss with the traffic data managers from the assigned company, prepare the data and gain insights into the formulated problem.**
2. **Then, start with the data processing and simultaneously work on the next point.**
3. **Focus on the literature review of existing problems and the needed theory. Utilize gained knowledge.**
4. **Analyze the prepared data series using a selected software.**
5. **Discuss all the (even if) particular problem insights and results with the company. Provide als a discussion of applicability of the results.**
6. **Last but not least, make an overview of problems within the so-called samrt city problems where the results of the master thesis are applicable. Discuss future research challenges.**
7. **Clearly discuss the main findings of the work.**

Thesis Extent:

Appendices:

Form of Thesis Elaboration: tištěná/elektronická

Bibliography:

1. Boon, MAA (2011). Polling models : from theory to traffic intersections. Doctor of Philosophy, TUE: Department of Mathematics and Computer Science, Eindhoven. DOI: 10.6100/IR702638.
2. Montgomery DC, Jennings CL, Kulahci M (2008). Introduction to Time Series Analysis and Forecasting. Wiley Series in Probability and Statistics. ISBN: 978-1-118-74511-3.
3. De Gooijer (2017). Elements of Nonlinear Time Series Analysis and Forecasting. ISBN: 978-3-319-43251-9.
4. Adhikari R, Agrawal RK (2013). An introductory study on time series modeling and forecasting. ISBN: 978-3-659-33508-2.
5. Fam S-K, Su C-J, Nien H-T, Tsai P-F, Cheng C-Y (2018). Using machine learning and big data to predict travel time based on historical and real-time data from Taiwan electronic toll collection. Soft Computing 22, 5707-5718. DOI: 10.1007/s00500-017-2610-y
6. Raj J, Bahuleyan H, Vanajakshi LD (2016). Application of data mining techniques for traffic density estimation and prediction. Transportation Research Procedia 17, 321-33. DOI: 10.1016/j.trpro.2016.11.102.

Thesis Supervisor:

Ing. Dušan Hrabec, PhD.  
Department of Mathematics

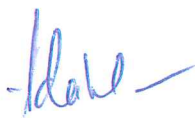
Date Assigned:

3 December 2018

Thesis Due:

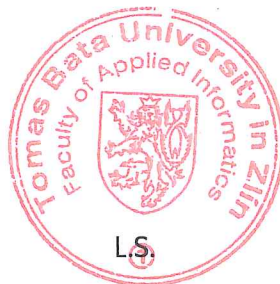
17 May 2019

Zlín, 7 December 2018



doc. Mgr. Milan Adámek, Ph.D.

Dean



prof. Mgr. Roman Jašek, Ph.D.

guarantor

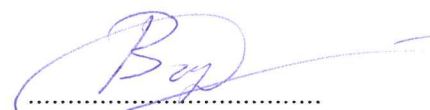
**I hereby declare that:**

- I understand that by submitting my Diploma thesis, I agree to the publication of my work according to Law No. 111/1998, Coll., On Universities and on changes and amendments to other acts (e.g. the Universities Act), as amended by subsequent legislation, without regard to the results of the defence of the thesis.
- I understand that my Diploma Thesis will be stored electronically in the university information system and be made available for on-site inspection, and that a copy of the Diploma/Thesis will be stored in the Reference Library of the Faculty of Applied Informatics, Tomas Bata University in Zlin, and that a copy shall be deposited with my Supervisor.
- I am aware of the fact that my Diploma Thesis is fully covered by Act No. 121/2000 Coll. On Copyright, and Rights Related to Copyright, as amended by some other laws (e.g. the Copyright Act), as amended by subsequent legislation; and especially, by §35, Para. 3.
- I understand that, according to §60, Para. 1 of the Copyright Act, TBU in Zlin has the right to conclude licensing agreements relating to the use of scholastic work within the full extent of §12, Para. 4, of the Copyright Act.
- I understand that, according to §60, Para. 2, and Para. 3, of the Copyright Act, I may use my work - Diploma Thesis, or grant a license for its use, only if permitted by the licensing agreement concluded between myself and Tomas Bata University in Zlin with a view to the fact that Tomas Bata University in Zlin must be compensated for any reasonable contribution to covering such expenses/costs as invested by them in the creation of the thesis (up until the full actual amount) shall also be a subject of this licensing agreement.
- I understand that, should the elaboration of the Diploma Thesis include the use of software provided by Tomas Bata University in Zlin or other such entities strictly for study and research purposes (i.e. only for non-commercial use), the results of my Diploma Thesis cannot be used for commercial purposes.
- I understand that, if the output of my Diploma Thesis is any software product(s), this/these shall equally be considered as part of the thesis, as well as any source codes, or files from which the project is composed. Not submitting any part of this/these component(s) may be a reason for the non-defence of my thesis.

**I herewith declare that:**

- I have worked on my thesis alone and duly cited any literature I have used. In the case of the publication of the results of my thesis, I shall be listed as co-author.
- That the submitted version of the thesis and its electronic version uploaded to IS/STAG are both identical.

In Zlin; dated:

  
.....  
Student's Signature

## **ABSTRACT**

Data analysis and data prediction is the field of informatics and mathematics, engaged in calculation of algorithms and mathematical models that are able to extract practical data from analyzed data. Data analysis has many aspects and approaches, covers different methods in various fields of science and everyday human life.

Data prediction and forecasting has interested people for thousands of years, with the new stage of human civilization development – expenditure of computers and different computing machines, data prediction methods and techniques tremendously change. New field of “Big Data” and machine learning, which research data sets that are too large to deal with by traditional data analysis techniques and applications are expanding. In our days “Big Data” are widely used in areas of internet search, economics, business, urban informatics and etc.

The urban informatics is one of the most interesting and applicable fields of “Big Data” usage. This field uses information and data sets in the context of smart cities and urban environments with purpose to make quality of life of pedestrians better and improve urban environment.

The aim of this project is to create a model, which would predict behavior of one of the most visible part of every urban area – crossroad. Provided information from traffic light controllers (detectors) on the crossroad at “Makro” Zlin is being registered, stored with equal periods of time and analyzed. Data analysis is implemented through usage of different statistical and computation models in a free and open-source integrated development environment “RStudio” and spreadsheet program for data storage “Microsoft Excel”. The project is aimed to predict traffic data on the crossroad.

## **ACKNOWLEDGEMENTS**

Acknowledgements, motto and a declaration of honor saying that the print version of the Bachelor's/Master's thesis and the electronic version of the thesis deposited in the IS/STAG system are identical, worded as follows:

I hereby declare that the print version of my Bachelor's/Master's thesis and the electronic version of my thesis deposited in the IS/STAG system are identical.

Foremost, I would like to express my sincere gratitude to my Supervisor Ing. Dušan Hrabec, Ph.D. for the constant support of my Master study, for his knowledge, interest in the topic, patience and persistence. His advising helped me during all the time of research and writing of the thesis.

Beside my advisor, I would like to thank representatives of “CROSS” company, product manager Ing. Ladislav Štríteský and Ing. Pavel Rychlý for provided data, support with explanations and education towards the topic.

I also would like to thank Head of Department of Informatics and Artificial Intellegence, FAI, TBU Prof. Dr. Roman Jasek, International Students Supervisor doc. Ing. Marek Kubalčík, Ph.D. and the rest of the thesis committee in particular Assoc. Prof. Ing. Jiří Vojtěšek, Ph.D, for their engagement, motivation, advises and guidance.

I thank my fellow classmates of Tomas Bata University: Vitalii Shyp, Tatsuki Monji, Kompaneev Maxim, Walid Cheikh and Hasan Jon for cooperation, discussions and teamwork towards common goal.

My sincere thanks goes to International Office of Tomas Bata University for continuous support during the whole period of studies in foreign country.

Last but not least, I would like to thank my family: My Grandmother (Mrs. Fedorova Lyudmila) and My Parents (Varaksin Pavel and Varaksina Olga) for making my Master’s education possible.

## **CONTENTS**

<b>INTRODUCTION .....</b>	<b>6</b>
<b>I THEORY .....</b>	<b>7</b>
<b>1. FORECASTING .....</b>	<b>8</b>
<b>2. DATA ANALYST FRAMEWORK .....</b>	<b>10</b>
<b>3. TIME SERIES AND PREDICTION .....</b>	<b>12</b>
3.1 Stationary Time Series and Autocorrelation .....	14
3.2 Time Series Forecasting Using Stochastic Models .....	17
3.3 Loess Decomposition Model .....	21
<b>4. SETTING APPROPRIATE MODEL (BOX-JENKINS METHODOLOGY) .....</b>	<b>24</b>
<b>5. FORECASTING ACCURACY. RESIDUALS .....</b>	<b>26</b>
<b>II ANALYSIS.....</b>	<b>29</b>
<b>6. SITUATIONAL PLAN .....</b>	<b>30</b>
<b>7. IMPLEMENTATION TECHNIQUES FOR TIME SERIES FORECASTING .....</b>	<b>32</b>
7.1 Graphical Representation .....	33
7.2 Forecasting Toolbox .....	36
<b>CONCLUSION .....</b>	<b>46</b>
<b>BIBLIOGRAPHY .....</b>	<b>47</b>
<b>LIST OF ABBREVIATIONS .....</b>	<b>49</b>
<b>LIST OF FIGURES .....</b>	<b>50</b>
<b>LIST OF TABLES .....</b>	<b>52</b>

## INTRODUCTION

In the current project traffic data from the crossroad “Makro” Zlin (76302, Czech Republic, Zlín tř. 3. května 1198) is being analyzed and forecasted. Data collected by use of “CROSS” company traffic light controller (detector), these detection inputs take value occupied by standing or moving vehicle and stored in database. Traffic light controllers save data per defined time unit – 10 minutes.

From the “CROSS” database data is being transferred to a spreadsheet platform “Microsoft Excel” and processed. Afterwards the forecast of provided data is created by use of free and open-source integrated development environment “RStudio”.

On the early stages of the forecast decisions about forecast horizon, frequency of the forecast and used methods being made. Afterwards methods applied and implemented towards used development environment. At last forecasted results compared with real values, graph plotted and conclusion about used method accuracy and accuracy of the forecast made. In theoretical part of current project all needed background for analytical part described.



## **I. THEORY**

## 1. FORECASTING

Forecasting is the task required in many situations (weather forecast for being prepared to a weather conditions, energy consumption forecast for requirements of additional power stations). Forecasts could be required for different time in future (from minutes to several years in advance).

Different events and things have different predictability. How complicated to build a forecast depends on a few things:

1. Understanding of factors that contribute to event;
2. Availability of data;
3. Affect of the forecast to an event itself (could appear in trading and stock forecasting).

Forecasting situations could widely vary in their types of provided data, time horizon which is being predict and factors which affect the event itself. Due to importance of forecasts for effective and efficient planning scientists were working towards the problem of accurate and various techniques of forecast for a long time, so forecasting methods can be simple or highly complex. In our days forecasting is an integral part of decision-making activities in the companies, depending on specific tasks modern organizations could require short-term, medium-term and long-term forecasts.

No to mention that the thing we are trying to forecast is not known, so we can represent it as a random variable. The random variable is being predicted and the accuracy of this prediction will depend on many characteristics and in order to achieve certain level of accuracy we could represent a random variable in prediction interval giving a range of values examined parameter could take. This set of values with their probabilities we call forecast distribution, however the average value of forecast distribution could be used.

By considering fact that all predicted values will be calculated by created mathematical model we could say that Thesis involves a Machine Learning part, where out time series will be a sort of “education” for a model built by machine (“R Studio”). The machine learning algorithm basically will make predictions and learn itself by improving in a lifetime (learning more by additional data). The software that is developed to make predictions for this thesis is using an algorithm of supervised learning, because and algorithm learning by using a historical data. The difference between procedural programming methods and machine learning is that machine creates an algorithm by itself, without actual use of scripts. The process of observation and learning could be represented as an vector.



*Figure 1: Process of Observation and Learning*

Not to mention a main challenge of machine learning, is to achieve a good prediction results on new data, data that is not a part of a training set. It is possible that after fitting a model the performance of it will be much worse on the new data than on a known training set, however we will try to avoid this appearance in the research.

The purpose of this Project would be creating a forecasting model that will be describing Traffic Data on the crossroad. A forecasting model is intended to show results in the specific time horizon, factors determining a forecast, appropriate methods that could be used and evaluating the accuracy of used methods would be described lately.

## 2 DATA ANALYST FRAMEWORK

A data analysis is the main concept of data prediction, in order to systematically analyze data it is necessary to understand a data analytics framework. Figure number 2 shows a simple flow in such a framework (Nabati & Thoben, 2017). It represents information in form of visualization and reporting tools.

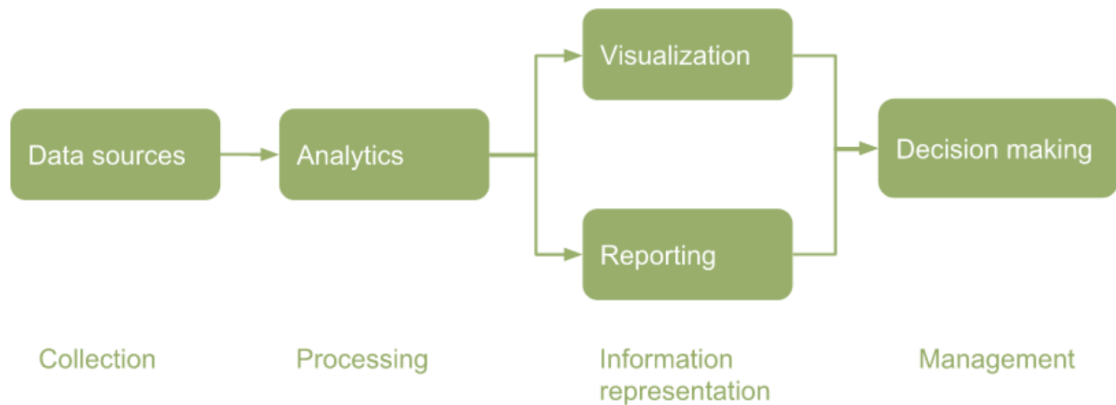


Figure 2: Data Analysis Flow

Data collection is the starting point of building successful data prediction model. Methods, equipment and data sources can vary. Real-life data streams usually taken from physical devices such as sensors. The used sensors can be infrared detectors, camera systems, ultrasonic sensors and others, equipment installed in special places for data monitoring. Another type of devices for measuring could be any kind of CPSs unit complex systems. Devices work with technique of Extract, Transform and Load – which helps to receive data in needed format. Later, collected data serve as a basis for further data analysis and to eliminate uncertainties in decision making process.

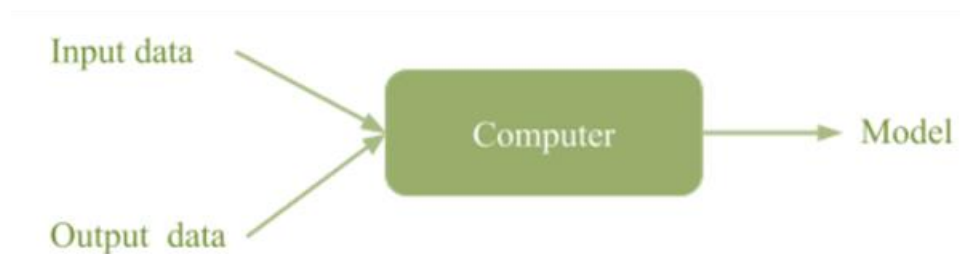
Data processing is important to get rid of redundant or event inconsistent data, because for quality data analysis it is necessary to have qualitative data in huge quantities. Consequently in order achieve high quality of data analysis it is necessary to filter and change data on the way from data collection. Well-known data preprocessing techniques are eliminating redundancies, handling of missing values and errors, removing irrelevant data and more complicated different data mining algorithms. Whatsoever full implementation of data filtering can significantly reduce the amount of data it leads to a better quality of data analysis.

Data storage is important in order to archive data for longer period of time. One of the up-to-date solutions for low cost archiving systems is Cloud Computing, it makes possible

to store data in distributed manner. Data gets stored on different servers and can be accessed from any remote access device. From disadvantages we could mention data security vulnerability.

Information representation is important in order to support decision making process. It makes it more convenient and understandable to present the results of data analysis in comprehensible format. Special diagrams, charts and graphs allow represent data in human readable format.

Last step – “Decision making” based on actual Data Prediction or data-based decision making. Data prediction can be important for understanding trends of data in future. This can help companies to make correct decision towards processes or/and company actions.



*Figure 3: Model Creation Flow*

The goal of the project is basically, by having input data (and output data in case of filtering) from traffic data sensors – build a model which allow to gain a prediction results within different time frames.

### 3 TIME SERIES AND DATA PREDICTION

Within Thesis and a task of Traffic Data Prediction we are using concept of Time Series and represent given data with dependence by time as analyzed Time Series. Anything that is observed sequentially over time (hourly, daily, weekly etc.) is a time series.

In this section of work we will describe theoretical part of a Time Series, from what it consists and modelling (prognoses) made with using Time Series.

By Time Series we implied sequence of data (one specific indicator) measured with specific time cuts – sequential set of data points in time. Mathematically we can basically represent a Time Series by formula:  $x(t)$  where  $t$  represents the time and  $x$  is a random variable. Within a time series model we know variable from time 0 to  $t$ , everything after  $t - (t+d)$  is future values of investigated variable.

$$x_0, x_1 \dots x_t \dots - \text{time series } x_i \in R$$

$$\hat{x}_{t+d}(w) = f_t(x_0, \dots, x_t; w) - \text{time series model}$$

$$d = 1, \dots, D, \text{ where } D \text{ is forecast horizon}$$

$$w - \text{vector of model parameters}$$

A time series could be discrete (variable measured at discrete points of time, usually with equal time cuts – hourly, daily, monthly) and continuous (variable measured at every instance of time). When time series data is being forecasted, the aim is to estimate how the sequence of data will continue in future.

In Thesis we will use forecasting methods that use only information about variable to be forecast (ignoring all other information), therefore we could extrapolate 4 main Time Series components, which are:

- Trend (general tendency of a time series (increasing, decreasing, or stagnate over time period) is termed a Trend);
- Seasonal (changes in a time series, which cause seasonal variations, if they occur - displayed in Seasonal factor);
- Cyclical (dependence that occurs in day-time or other cyclical dependence (caused by circumstances, which repeat in cycles) called Cyclical);
- Irregular (unpredictable influences, which don't have any particular pattern, they could be caused by any unusual event, like a hockey match will affect loads of roads and crossroads on the way to a stadium).

Considering the effect of these components, two models are usually used for time series: Multiplicative and Additive.

$$Y(t) = T(t) \times S(t) \times C(t) \times I(t)$$

$$Y(t) = T(t) + S(t) + C(t) + I(t)$$

Where in upper formulas  $Y(t)$  is the observation and  $T(t)$ ,  $S(t)$ ,  $C(t)$ ,  $I(t)$  are trend, seasonal, cyclical and irregular variation at time  $t$ .

Consequently we can describe the main goal in time series research as finding and calculating all of the four components, to use the obtained information for data prediction. A main stages for time series analysis would be:

- 1) Gathering information (in our case data were collected from crossroads using detectors);
- 2) Graphical representation of a time series and exploratory analysis which includes determination of trend, seasonality, cycles;
- 3) Choosing and fitting models. In this stage we are choosing the best model that would fit given time series, it is common to compare potential models among each other. In theoretical part of current Thesis we will describe possible models to be used;
- 4) Examination and evaluating of built model. Predicted data is being compared with Test Set (real values of data traffic in our case), assessing the accuracy of forecasts.

In forecasting theory models that use only information about forecasted variable are exponential smoothing, decomposition and ARIMA models. In current project “mixed models” are not considered to be used (that include dependency of different variables which affect the system), because we assume that a system of data traffic is not fully described by multiple parameters and it is extremely difficult to measure the relationship among those parameters affecting a data. Secondly, the main concern for us is to predict what exactly will happened and not why it will happens. And finally, with time series model we might get a more accurate forecast.

To conclude, in current Thesis built model would be fitted to a given time series. In forecasting of time series, past observations are collected and analyzed to, as a main goal – develop a suitable mathematical model, the future data are then predicted using built model.

### 3.1 Stationary time series and autocorrelation

For building a time series model that is useful for future forecasting the necessary condition is stationarity of a time series. A stationary time series means that properties do not depend on the time at which a series is observed.

A process is Strongly Stationary if the joint probability distribution function of  $\{x_{t-s}, x_{t-s+1}, \dots, x_t, \dots, x_{t+s-1}, x_{t+s}\}$  is independent of  $t$  for all  $s$  – for strongly stationary process joint distribution of any possible set of random variables from the process is independent of time. For weakly stationary of order  $k$  – implies that statistical characteristics of a time series (variance, covariance) are dependent only (up to that order  $k$ ) on time differences of the data being used to estimate the moments. In simple words – a distribution of a Strongly Stationary process is not time dependent (Trend and Seasonal component leads to non-stationarity, Cyclical component is not affecting stationarity). The way to make non-stationary time series stationary – is to use differencing (difference between sequential observations) it can help to stabilize a mean of time series, in order to stabilize a variance of time series – logarithmic transformation could be used.

One of the logarithmic transformations is a Box Cox transformation that may apply. A Box Cox transformation intend to transform non-normal dependent variables into a normal shape. At the core of the Box Cox transformation is an exponent, lambda ( $\lambda$ ), which lays in range from -5 to 5, Box Cox transformation has the form:

$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda} \\ \log y, \text{ if } \lambda = 0 \end{cases}$$

It could also appear that differenced data will not appear to be stationary and it needs a second-order differencing (rarely appears in practice), it is calculated with the formula:

$$y_t'' = y_t' - y_{t-1}' = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2})$$

Sometimes it could be useful to use a seasonal differencing, when we take the difference between an observation and the observation from the same season appeared previously, it is calculated with the formula, where  $m$  is the number of seasons:

$$y_t' = y_t - y_{t-m}$$

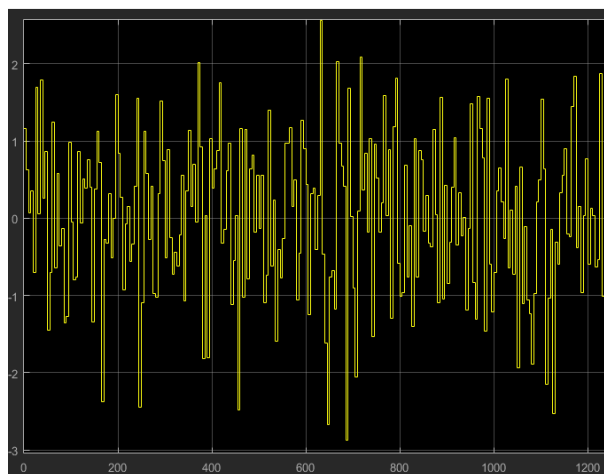


Next important parameter one should know while analyzing a time series is Autocorrelation. Autocorrelation measures internal correlation within a time series, it explains internal association between observations in a time series.

The way to tell if differencing is required for time series is to use a statistical hypothesis tests of stationary - unit root test (designed to determine whether differencing is required). One of the most useful unit root tests is KPSS (Kwiatkowski–Phillips–Schmidt–Shin) test. It is used to test that a null hypothesis that an observable time series is stationary around a deterministic trend against the alternative of a unit root. After the test result we could indicate if a null-hypothesis were rejected (test statistic should be significantly bigger than 1%) or approved (test statistics is close to 1%). Afterwards if the hypothesis about stationary were rejected we difference the data and apply the test again.

Autocorrelation determines internal association: assigning a value from +1 (positive linear dependence) to -1 (negative linear dependence), 0 value will show no association. Strong autocorrelation will indicate that it is possible to predict variable values in future by observation of variable values in past.

As an example of calculating autocorrelation function we would like to give a samples from signal-processing theory (however it could be input and output of any certain model) below showed an example of 250 samples with sampling time 5 seconds (the whole simulation time 1250 seconds). After adjustments we are obtaining an input and output signal graphics:



*Figure 4: Input Signal*

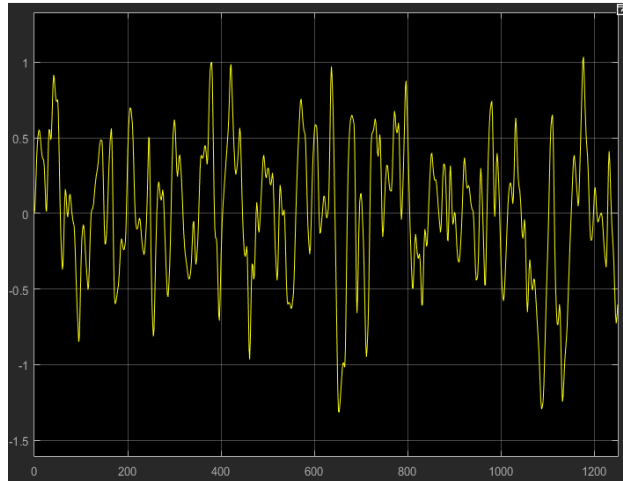


Figure 5: Output Signal

Autocorrelation function is computed as a mean value of heterochronic values of one signal:

$$\hat{R}_{uu}(i) = \frac{1}{N-i} \sum_{k=1}^{N-i} u(k) \cdot u(k+i) \quad i = 0, 1, K, m$$

$$\hat{R}_{yy}(i) = \frac{1}{N-i} \sum_{k=1}^{N-i} y(k) \cdot y(k+i) \quad i = 0, 1, K, m$$

Autocorrelation function for input and output (The interval of the time shift is gradually increasing from 1 sampling interval to 25 sampling intervals:  $0,1 \cdot 250$ ):

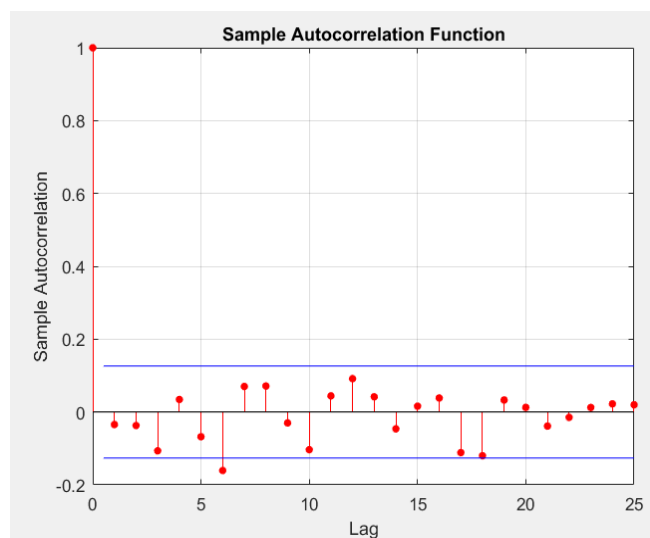


Figure 6: Input ACF

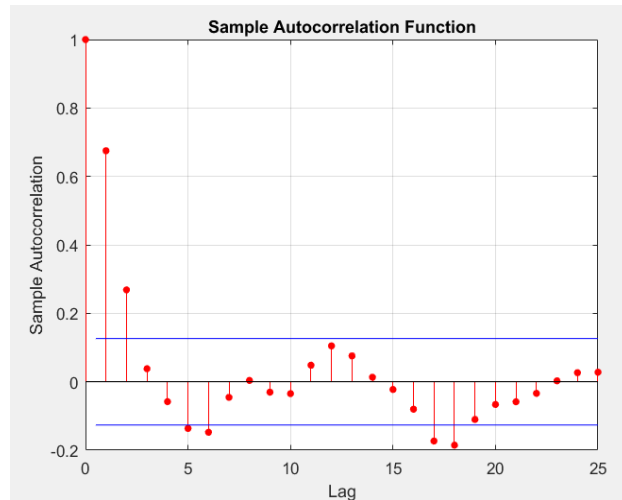


Figure 7: Output ACF

Computation of the autocorrelation function – mean value of a multiplication of time shifted values. Autocorrelation coefficients one by one form an autocorrelation function. In both input and output functions we could see the similar trend – important task is to determine the trend and create a Mathematical Model. Which would be describing the process.

Further in theoretical part of the Thesis the application of the autocorrelation and stationarity will be explained.

### 3.2 Time series forecasting using Stochastic Models

In general processes models for time series data can have many forms and represent different stochastic processes. Two mostly used linear time-series models are Autoregressive (AR), Moving Average (MA) and combining this two, the Autoregressive Moving Average (ARIMA) models. ARIMA models and its different variations are based on the Box-Jenkins principle and basically known as Box-Jenkins models (looking for the best fit of time-series model to past values of a time-series).

Linear models are easier to implement due to their relative simplicity in understanding. Whatsoever in practice many time series models shows non-linear patterns. Most well-known non-linear models described in literature are Autoregressive Conditional Heteroscedasticity (ARCH) model and its variations.

In this chapter important linear and non-linear stochastic time-series models with their different properties will be discussed. This chapter will provide us with theoretical background for choosing appropriate model for researched process of crossroad traffic.

The Autoregressive Moving Average (ARMA(p, q)) is combined of two models AR(p) and MA(q) and suitable for one variate time series modelling. Mathematical expression of the AR(p) model, where predicted value of a variable is calculated through linear combination of p past observations and a random error together with a constant term:

$$y_t = c + \sum_{i=1}^p \varphi_i y_{t-i} + \varepsilon_t = c + \varphi_1 y_{t-1} + \dots + \varphi_p y_{t-p} + \varepsilon_t$$

Where  $y_t$  represents actual value and  $\varepsilon$  represents a random error at time period t with integer constant p representing an order of the model,  $\varphi_i (i = 1, 2, \dots, p)$  are model parameters and c is a constant.

In MA(q) model uses past errors as the explanatory variables:

$$y_t = \mu + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \varepsilon_t = \mu + \theta_1 \varepsilon_{t-1} + \dots + \theta_p \varepsilon_{t-p} + \varepsilon_t$$

Where  $\mu$  is the mean of the time series,  $\theta_i (i = 1, 2, \dots, q)$  are the model parameters and q is the order of the model. A noise in current model assumed to be a white noise, which has a zero mean value and constant variance.

Previously in Thesis the time series were represented as a summary of seasonal, trend, cycle (usually trend and cycle combined together) and irregular components. Moving averages is one of the first steps in classical decomposition to estimate a trend-cycle. Usually after applying MA formula the trend-cycle becomes smoother than the original time series and shows main movement of the graph without shifts. The higher the order of applied moving average the smoother data curve becomes, typically the MA have an odd-order – so they are symmetric. In case we need to have an even order of MA and still keep it symmetric the moving average is being applied to a moving average (i.e. MA of an order 4 could be taken and then MA of an order 2 applied to the results).

For a time series with seasonal period and if we assume that seasonal component is constant every year – classical decomposition could be used, which contains the next pattern (for additive decomposition):

1. If seasonal period is an even number, a trend-cycle component calculated using a 2 order MA applied to an seasonal period number MA, if its an odd – seasonal period number MA is used;
2. Series:  $y_t - \hat{T}_t$  calculated

3. Then to estimate the seasonal component for each season, values from step 2 for that season average;
4. The irregular component calculated:  $\widehat{R}_t = y_t - \widehat{T}_t - \widehat{S}_t$

Besides classical decomposition there is several modern and better methods that could be used, however their explanation goes beyond the scope of current Thesis.

If only one model would be taken time series most likely will fit to an AR model because in MA the random errors are not foreseeable. The combination of both autoregressive (AR) and moving average (MA) models can be effectively combined together in ARMA models, represent with mathematical equation:

$$y_t = c + \varepsilon_t + \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \sum_{i=1}^p \varphi_i y_{t-i}$$

The term autoregression (AR) shows that there is a regression of the variable against itself. If we use a formula for AR component it will include a summary of a lagged values of analyzed variable in the past plus white noise:

$$y_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} + \varepsilon_t$$

The term moving average (MA) uses past forecast errors in regression-like model. By using MA method each value of variable can be represented as weighted moving average of the past few forecasted errors. If we use a formula of MA component it will include a summary of forecast errors:

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}$$

The constant  $c$  and the value of  $d$  has an important effect on the long-term forecasts using ARIMA models:

- If  $c=0$  and  $d=0$ , the long-term will go to zero;
- If  $c=0$  and  $d=1$ , the long term will go to a constant;
- If  $c=0$  and  $d=2$ , the long term forecast will follow a straight line;
- If  $c \neq 0$  and  $d=0$ , the long-term forecast will go to the mean of the data;
- If  $c \neq 0$  and  $d=1$ , the long-term forecast will go to a straight line;
- If  $c \neq 0$  and  $d=2$ , the long-term forecast will go to a quadratic trend.

Also important parameter for ARMA model representation is lag operator. The lag of backshift operator is defined by  $Ly_t = y_{t-1}$ , it represent ARMA models as follows:  $\varphi(L)y_t = \theta(L)\varepsilon_t$ . ARMA and ARIMA models basically intend to describe the autocorrelations in the data.

Data set from crossroad have strong seasonal character, so basic ARIMA model will not describe time series precisely in a long term. In this case Seasonal ARIMA model are capable of modelling a wide range of seasonal data. Additional seasonal terms are included in the ARIMA models: non-seasonal part (p, d, q) and seasonal part of the model (P, D, Q). The seasonal part is similar to non-seasonal but involves backshifts of the seasonal period.

Important part of creating a data prediction model, based on past values of the researched variable is calculating autocorrelation (ACF) and partial autocorrelation (PACF) function (they will allow us to determine a values of p and q and will allow us to see seasonal lags on the graph). These statistical measures reflect how the observations in a time series are related to each other. For modelling purposes it is important to plot both functions in certain time lags. Mathematical definition for a time series  $\{x(t), t = 0, 1, 2, \dots\}$ , the autocovariance at lag k is defined as:

$$y_k = Cov(x_t, x_{t+k}) = E[(x_t - \mu)(x_{t+k} - \mu)]$$

The Autocorrelation Coefficient at lag k is calculated as:

$$\rho_k = \frac{\gamma_k}{\gamma_0}$$

Where  $\mu$  is the mean of the time series, autocorrelation coefficient belongs to interval from -1 to 1 and closer coefficient moves to an edges of the interval – the stronger linear dependence it shows.

In case if autocorrelation of observation needs to be measured on specific lags – partial autocorrelation function (PACF) is used.

In case of time-series data shows non-stationary behavior (when seasonal patterns and trends occurs) used ARIMA model (ARMA model could only be used for and stationary time series data).

In ARIMA models a non-stationary time series is made stationary by applying finite differencing of the data points. The mathematical formulation of the ARIMA(p,d,q) model is shown below:

$$\varphi(L)(1-L)^d y_t = \theta(L)\varepsilon_t$$

$$\left(1 - \sum_{i=1}^p \varphi_i L^i\right) (1-L)^d y_t = \left(1 + \sum_{j=1}^q \theta_j L^j\right) \varepsilon_t$$

- Where:  $p$ ,  $d$  and  $q$  are integers that shows the order of the autoregressive, integrated and moving average parts of the model respectively;
- The integer  $d$  shows the order of differencing. Generally 1st order is enough for most of the cases, it is important not to over differentiate the model, or it could loose connection with the real (not differentiate model). When  $d$  is equal to 0, then it is ARMA( $p,q$ ) model;
- We could represent ARIMA( $p,0,0$ ) as the AR( $p$ ) model and ARIMA( $0,0,q$ ) as the MA( $q$ ) model.

Selecting of suitable and accurate values of  $p$ ,  $d$  and  $q$  could be difficult task, however modern tools (“R Studio” – practical part described later on) used in the Thesis will allow us to calculate it automatically.

Once the order of the model has been chosen ( $p$ ,  $d$  and  $q$ ), we need to choose a constant parameter  $c$  in our Thesis for parameters estimation maximum likelihood estimation (MLE) is used.

After describing different time series model we would be determining which model data traffic from crossroad could be better implemented.

### 3.3 Loess decomposition model

The time series of traffic data on the crossroad have strong seasonal factor. Different and one of suitable methods for seasonal time series is a seasonal-trend decomposition based on Loess (“locally-weighted scatterplot smoothing”). This model decompose time series into three components: trend, seasonal and remainder.

$$Y(t) = T(t) + S(t) + R(t)$$

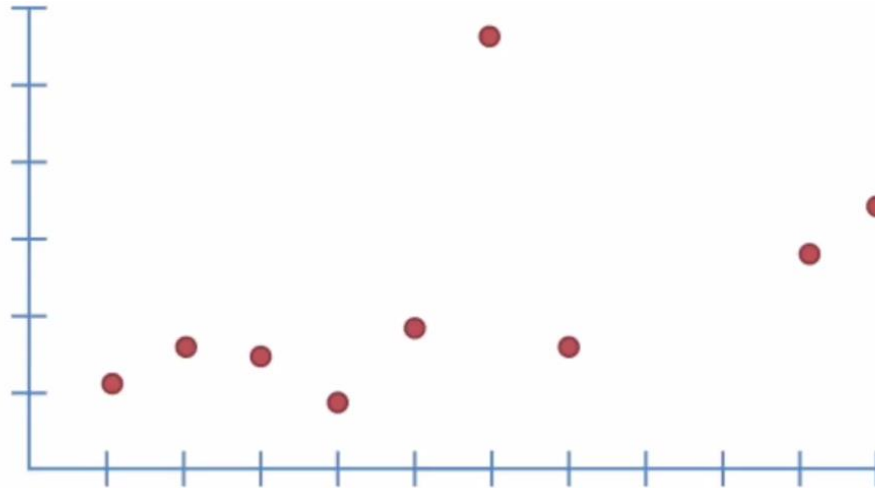
Loess method also known as locally weighted polynomial regression. At each point in the data set a low-degree polynomial is fit to a subset of the data. The polynomial is fit using weighted least squares, with more weight to points near the point. Basically the Loess fit is complete after regression function values have been computed for each of the data point.

The main steps of Loess decomposition are:

1. A window of specified step-size is placed over a data (the wider the window, the smoother the resulting loess curve);
2. A regression curve is fitted to the observations that locate within the window using a least squares method, the points at the window are being weighted (closer to the evaluated at the moment point, greater the weight);

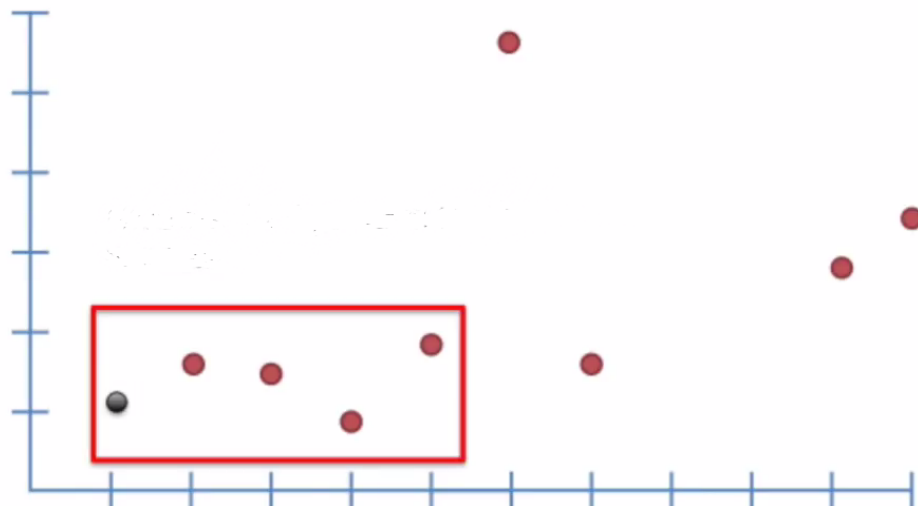
3. The process of moving the window further and weights recalculation repeated several times;
4. The points of the regression line obtained and connected. Each point of the resulting loess curve is the intersection of a regression line and a vertical line at the center of such window.

A simple example of using a Loess method, fitting a curve to a data with size 5 is given below:



*Figure 8: Data set for Loess Decomposition*

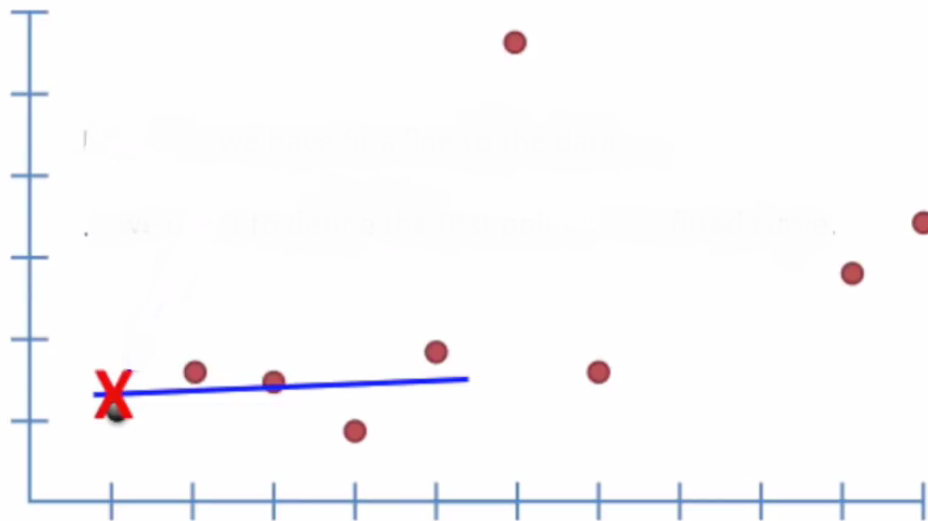
In current example we use window size 5:



*Figure 9: Window in Data Set for Loess Decomposition*

Afterwards a fitted line created by considering weights of the closest point and actual first point of the fitted curve determined:





*Figure 10: Fitted Curve for Loess Decomposition*

Afterwards a next points are evaluated with window moving further during the process. When the fitted curve is created – it is actually an regression function which could predict values in a future.

Advantages of the Loess method are no need of function specification for fitting a model, Loess is very flexible and convenient to use with seasonal time series. However, the down sides of the Loess is that it requires large, densely sampled data, and it does not produce and actual regression function that represented by mathematical formula and might be applied in current project.

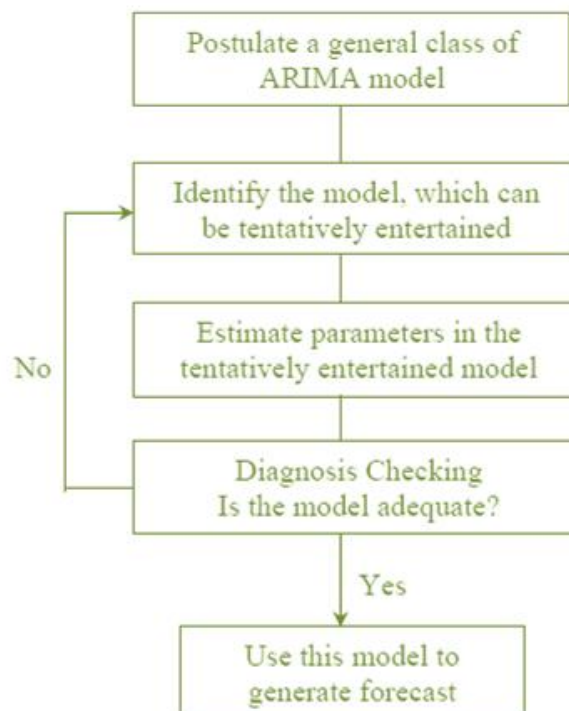
## 4 SETTING APPROPRIATE MODEL (BOX-JENKINS METHODOLOGY)

On the first stage of data prediction based on historical data – is to actually select appropriate model and optimal model orders that can produce accurate forecast of analyzed variable. One of the approaches were built by statisticians George Box and Gwilym Jenkins, which allows to find a best fit of parameters to build ARIMA model.

Box-Jenkins method uses, so-called: three step iterative approach. It includes:

- model identification;
- parameter estimation;
- diagnostic checking.

This process is being repeated several times, until we are able to build a model which satisfy the accuracy. Then model could be used for future estimations and forecasting. Box-Jenkins methodology graphically represented below:



*Figure 11: Box-Jenkins Methodology*

One of the most complex steps in appropriate model selection is to calculate estimated model parameters.

One of the most important criteria's to indicate a model accuracy (could be used to any model) is Akaike's Information Criterion (AIC), it could be written as:

$$AIC = -2 \log(L) + 2(p + q + k + 1)$$

Where L is a likelihood of the data. The formula is given for ARIMA models, however criteria could be used to any model. Good models are obtained by minimizing AIC.

## 5 FORECASTING ACCURACY. RESIDUALS.

After creating and fitting a model it is important to measure an accuracy, and in the end - the “residuals” are what is left over. The residuals are equal to the difference between the real values and the values obtained through built model:

$$e^t = y_t - \hat{y}_t$$

After calculating residuals they are useful in checking if a model has fully captured the information in the provided data. A residuals of a good forecasting model will have following properties:

1. There is no correlation (close to zero) between residuals. If correlation appears it means that there is information left in the residuals itself, however this information needs to be used in calculating forecast;
2. There is a zero mean (residuals mean close to zero). If the mean is differs from zero, it means that the forecasts are biased;
3. The residuals variance are constant;
4. The residuals have normal distribution.

Forecasting methods that doesn't satisfy first and second criteria's can be improved and modified to give better forecasts. Whatsoever is it possible that it will be several forecasting methods that satisfy the criteria's and still could be improved. Forecasting methods that do not match third and fourth criteria's not necessarily could be improved.

In addition, looking at the residuals autocorrelation plots from statistical point there is a more formal tests, which consider the whole set of residuals as a group, those type of tests called “portmanteau” test and include “Box-Pierce” test, “Ljung-Box” test and others.

To conclude the role of residuals in forecast model accuracy – we expect the residuals and autocorrelations of residuals to look more like a white-noise process.

To calculate residuals while evaluating forecasts accuracy - time series data will be separated in training and test data set, where the training data is used to create a forecasting model itself and the test data is used to compare predicted and real values.

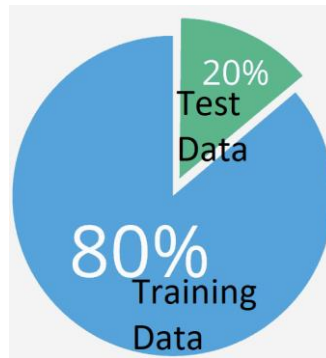


Figure 12: Data Set Separation

The size of the test data set is taken approximately 20% of the total data sample. Below important characteristics of built mathematical model are given:

1. If model fits the training data set not necessarily leads to an accurate forecast;
2. Creating model with enough parameters leads to a well-fit of the model;
3. Over-fitting a model leads to an additional errors.

While working on forecasting accuracy it is also important to work on forecast “error”, where “error” means an unpredictable part of an observation, it is calculated with formula:

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}$$

Important to mention that errors are different from residuals:

1. Errors are calculated on the test data set, while residuals are calculated on the training data set;
2. Errors could be based on multi-step forecast, while residuals always based on one-step forecast.

Additionally forecast error could transformed in scale-dependent errors – calculated as a mean absolute error:

$$MAE = mean(|e_t|)$$

Could be used while comparing forecast methods applied to a single time series, or to several time series with the same units. Another error representation is percentage error, it is calculated as:

$$MAPE = mean(|p_t|)$$

And it is unit-free, it is convenient to use while comparing forecast performances between data sets.

Forecast values could also be presented in prediction interval, within which we expect predicted values to lie with a specific probability. If assumption that forecast errors are normally distributed will be taken – then we can conclude then 95% of values in prediction interval for the  $h$ -step forecast is laying down in interval of  $\hat{y} \pm 1.96\sigma_h$ , where  $\sigma_h$  is a standard deviation of the forecast distribution, 1.96 – is a multiplier and it is depends on the coverage probability (when percentage is lower– lower a multiplier). Basically the main purpose of the prediction intervals is that they show the uncertainty of the forecasts – giving not only point value, but a range of values that event could obtain in future.

Most precise prediction intervals obtained while using one-step prediction intervals, the standard deviation of the residuals will be the same with the standard deviation of forecast distribution, so interval could be easily calculated by using the values of the multiplier (dependent on percentage accuracy). In case of multi-step prediction intervals – the length if the forecast horizon will increase. Simply saying – the further ahead we forecast the more uncertainty will appear in forecast and larger the prediction interval will become. To calculate the actual interval used a “benchmark methods”, which assumes that residuals are uncorrelated.

Another more complicated case, when forecast errors does not have a normal distribution, in this case used a bootstrapping, which assumes forecast errors are uncorrelated. Bootstrapping method workflow implements adding an errors one by one to an calculated data set. While we do this repeatedly – we obtain future values and calculate an forecast horizon, by calculating percentiles for each forecast horizon.

When it goes to an actual forecast calculation we need to go back to ARIMA models in the theoretical part, and calculate an actual forecast using the following steps:

- Expand the ARIMA equation, to move  $y_t$  on the left side;
- Rewrite the equation with replace of  $t$  to  $T + h$ ;
- On the right side of the equation replace the future observations with their calculated forecasts, future errors will be equal to 0 and past errors equal to residuals.

To conclude, to obtain high forecast accuracy and have understanding of possible values of forecasted values – residuals analysis with prediction intervals are used.

## **II. ANALYSIS**

## 6 SITUATIONAL PLAN

Below presented a situational plan from Satellite obtained from “Google Maps” of investigated intersection exit out of “Makro” (address tř. 3. května 1198, 763 02 Zlín, Czech Republic) provided below:



*Figure 13: Satellite Situational Plan*



Situational plan with traffic data lights and detectors description (provided by “CROSS” company) showed below:

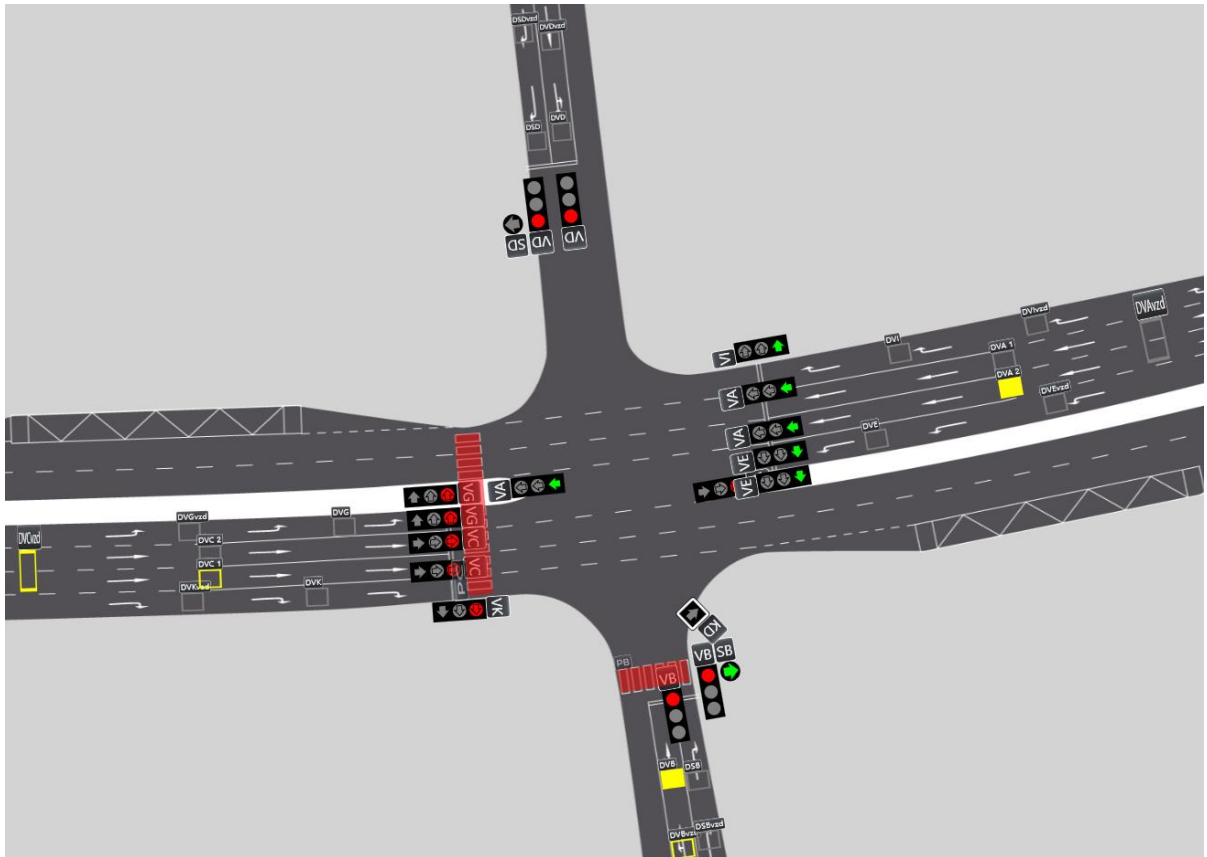


Figure 14: Situational Plan

Table that represents main Data Detectors showed below:

DVA1	Straight from right to left
DVA2	Straight from right to left
DVE	Left turn from right
DVC1	Straight from left to right
DVC2	Straight from left to right
DVK	Right turn from left
DVB	Straight from down
DSB	Right turn from down

Table 1: Data Detectors

Data (amount of cars passed through detector), taken from interval (1.05.2018 – 30.06.2018) and analyzed, predicted values are being created by analysis of mentioned time period. Data in the future is provided for a test data set (cross-validation and comparison real and predicted values).

## 7 IMPLEMENTATION OF TECHNIQUES FOR TIME SERIES FORECASTING

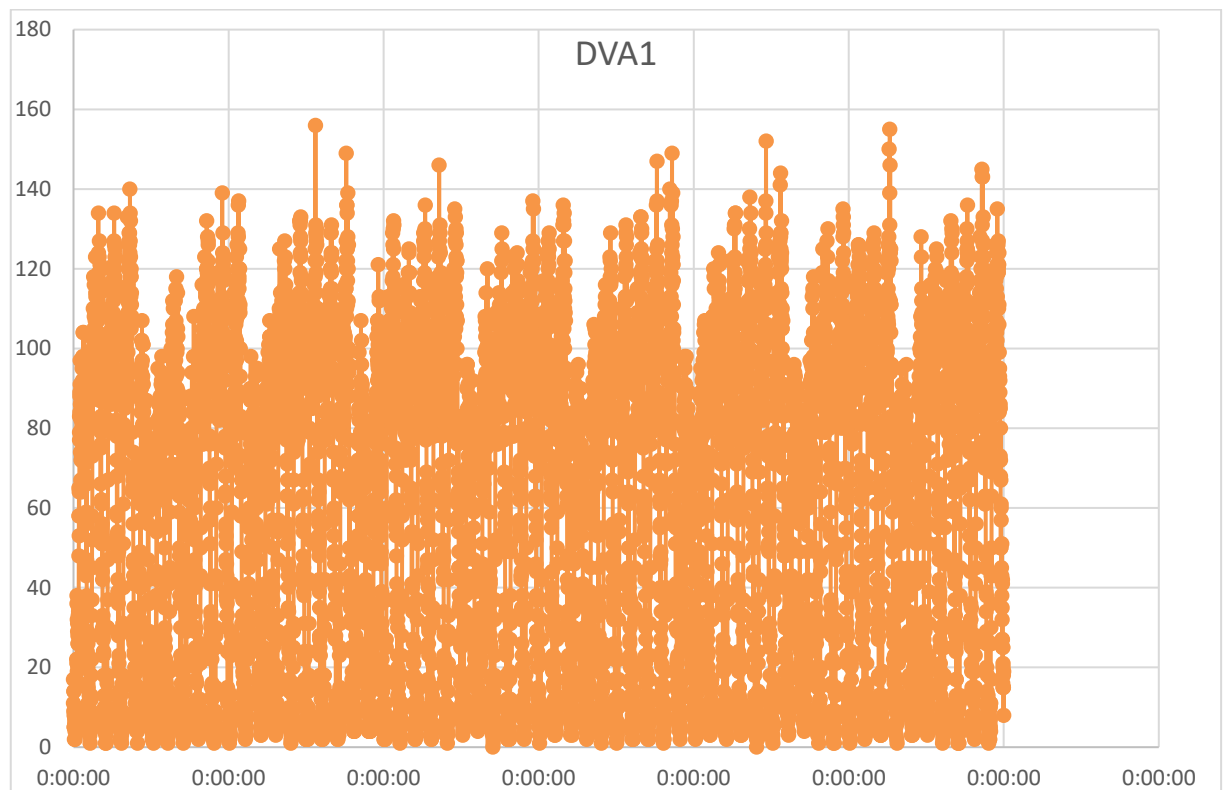
In the previous parts of Thesis, we have described various theoretical parts and popular techniques for time series forecasting. The next important step is - implementation, i.e. to apply these methods for generating forecasts. To conclude we are going to list each steps for actual Forecasting:

1. When applying a particular model to simulated or real time series, first the raw data is divided into two parts, the Training Set and Test Set (data is usually divided 80% to 20% for sets).
2. The observations in the training set are used for constructing the model itself. Validation Set – is a small part of a Training Set which is kept for validation purposes.
3. Visual analysis of a time series.
4. Next step is preprocessing it could be done by various techniques, it is done by normalizing the data or taking logarithmic, differencing (d, D parameters) or other transforms like a Box-Cox Transformation.
5. Afterwards ACF and PACF graphs are being created and analyzed, parameters p, q, P, Q could be set up.
6. Models are being created (different parameters could be used for testing of a most suitable model). AIC of models are calculated and used to compare the accuracy of models (lower AIC – better it should describe a process).
7. Residuals of created model and real values are being calculated and compared. Residuals are being analyzed (graph should be close to a white noise which will indicate that residuals do not contain valid for the model data).

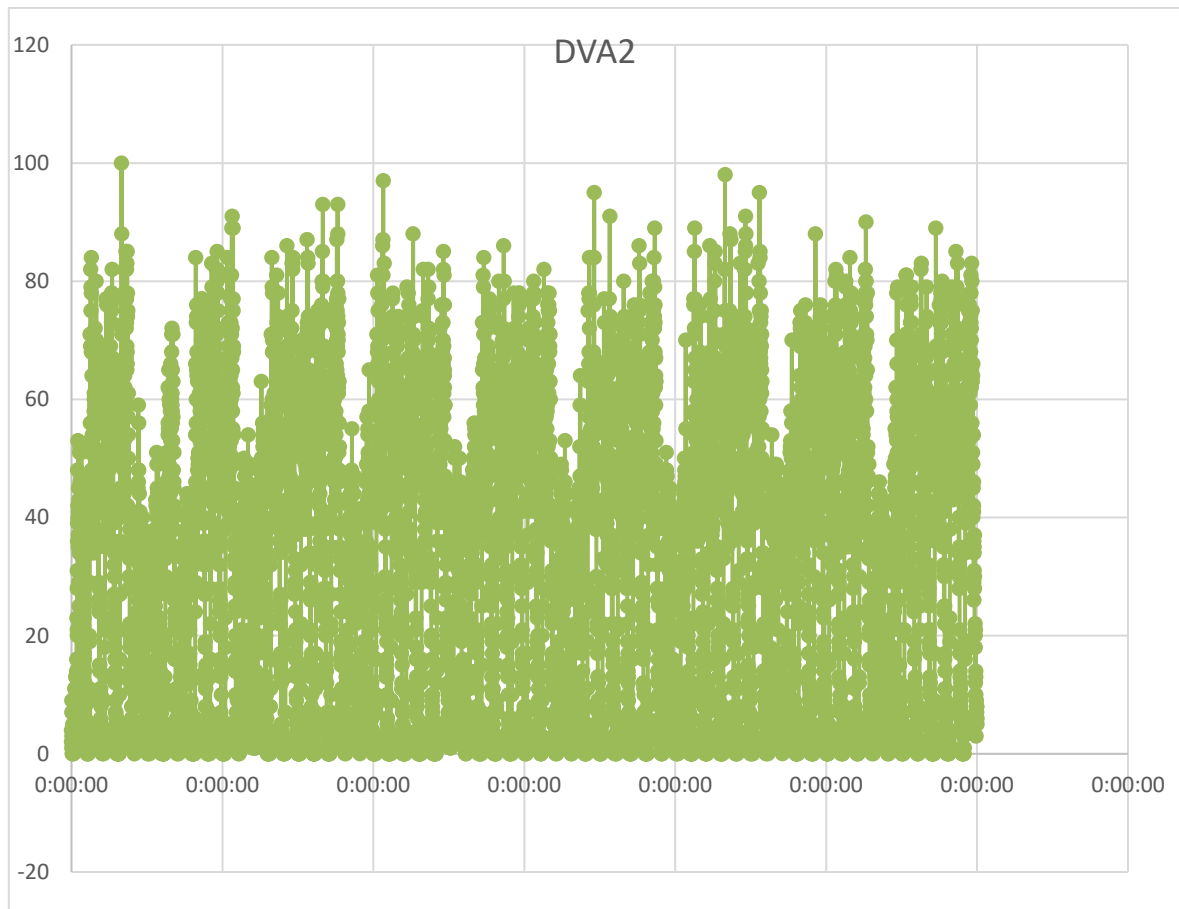
## 7.1 Graphical Representation

For time series data first graph we built is a time plot. The observation of data traffic on a crossroad are plotted against the time (with 10 minutes lag).

In our research we are provided with data from 23 “CROSS” radars. A detailed analysis and graph representation of a time interval will be shown from DVA1 and DVA2 radars (direct lines going from right to left side of the situational plane) 1.05.2018 to 29.06.2018 showed below:



*Figure 15: DVA1 Time Series*



*Figure 16: DVA2 Time Series*

The time plot reveals some interesting features in the graphs:

- We don't see long-term increase or decrease, so could make conclusion about no trend in time series;
- Clear seasonal factor, which occurs in peaks within a "rush hour" during the day and weekly-dependency, which shows higher peaks within a week days and lower peaks during the weekend.

For precise data forecasting we are dividing days within given time period into groups by day of the week from Monday to Sunday.

Data in form of time series for Mondays only for DVA1 and DVA2 data detectors showed below:

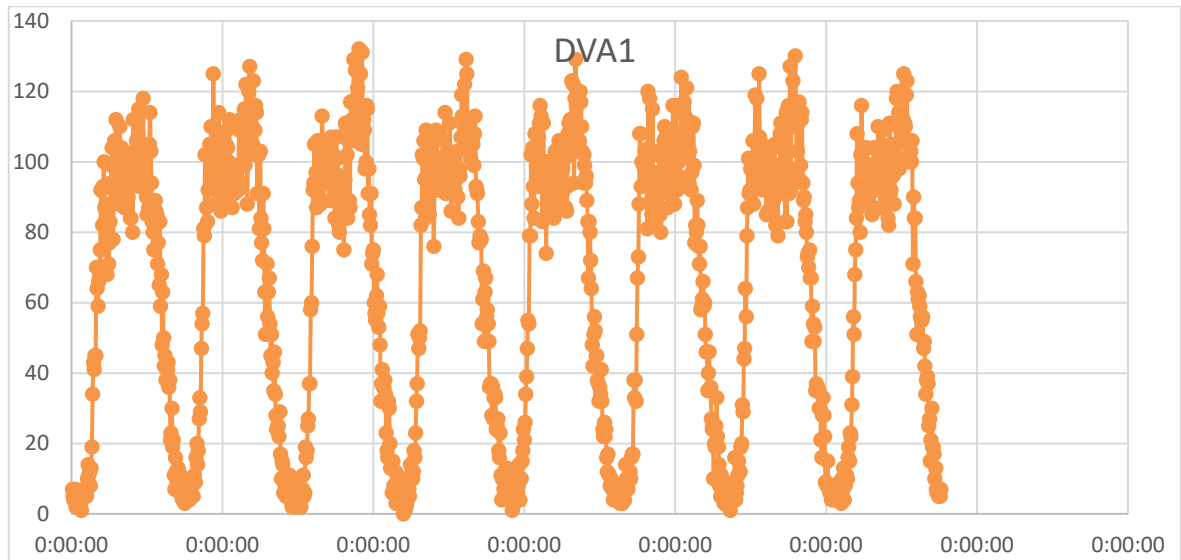


Figure 17: DVA1 Time Series for Mondays

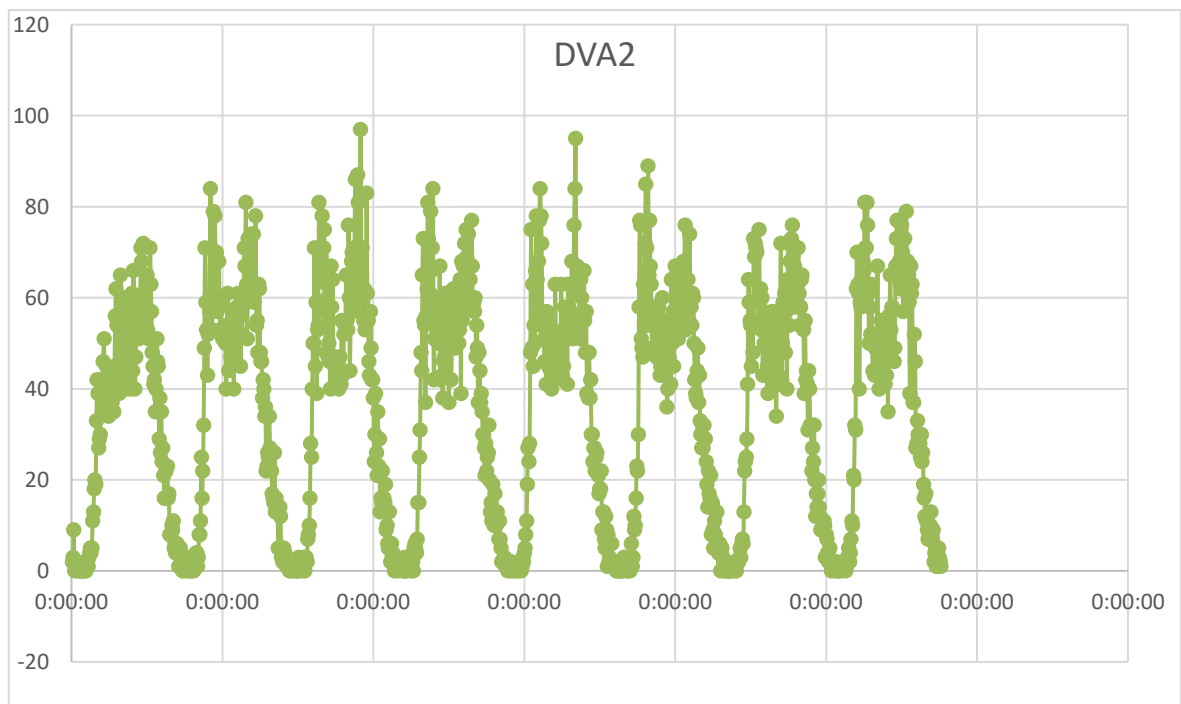


Figure 18: DVA2 Time Series for Mondays

Data analysis and actual forecasting using appropriate statistical model described later on in the project.

## 7.2 Forecasting Toolbox

As a main tool for calculation in the Thesis program – “R” were used. “RStudio” is a free software and programming language mainly created for statistical computing, data analysis and graphics developed by “R Core Team”.

First of all in order to work and analyze time series data additional packages needs to be installed at “RStudio”:

```
install.packages("forecast")
```

```
install.packages("seasonal")
```

```
install.packages("fpp2")
```

We write this code first in the command line before actual program implementation in order to complete installation of needed additional packages.

Afterwards analyzed data (in first example is data from detector DVA1 on Mondays) is being stored in the .txt file at the same directory with .r file and Present Working Directory is being set up in “R” and all data from the .txt file is being stored to a variable dataIn with the commands:

```
setwd('E:\\Data')
```

```
dataIn <- read.table("dva1mon.txt", header=TRUE, sep=";")
```

Time series itself is being stored into special “ts()” object in order to use all needed function for time series analytics. In order to store time series in variable tsData with specified frequency it is needed to be calculated. Originally frequency for time series data is set up as 1 year, in our case data is being stored every 10 minutes, so we need to calculate it as multiplication of hourly (measurement 6 times per hour), daily (24 hours) and amount of days per year  $6*24*365=52560$ . The start of measurement is 7.05.2018, which will be 126 days from the beginning of 2018, or 2018.345. In our case “RStudio” code shown below:

```
tsData <- ts(dataIn, start=2018.345, freq=6*24*365)
```

Time series is being stored in “tsData” variable as a “ts” object which takes data from already stored numerical vector “dataIn”. To build a graph of created time series we use function:

```
autoplot(tsData) + ggtitle("DVA1 detectors on Monday") + xlab("Days") +
  ylab("Cars")
```

Function `autoplot` will be used multiple times lately in the research in order to obtain time series plots, in our case built time plot showed below:

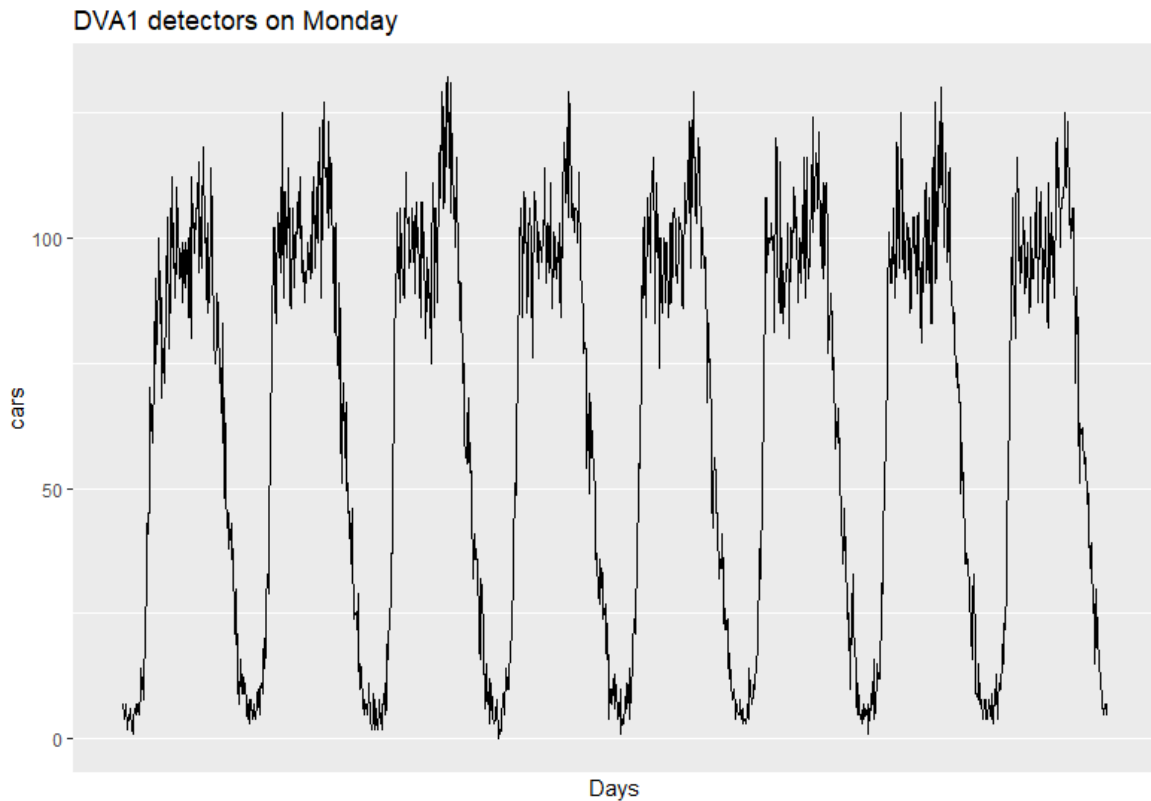


Figure 19: DVA1 Detectors On Monday in R

All steps described at “Implementation of Techniques for Time Series Forecasting” chapter are automatically implemented in used `auto.arima()` “RStudio” toolbox. `Auto.arima()` returns best ARIMA model according to their AIC value. The full function set in “RStudio” provided below:

```
auto.arima(y, d = NA, D = NA, max.p = 5, max.q = 5, max.P = 2,
  max.Q = 2, max.order = 5, max.d = 2, max.D = 1, start.p = 2,
  start.q = 2, start.P = 1, start.Q = 1, stationary = FALSE,
  seasonal = TRUE, ic = c("aicc", "aic", "bic"), stepwise = TRUE,
  nmodels = 94, trace = FALSE, approximation = (length(x) > 150 |
  frequency(x) > 12), method = NULL, truncate = NULL, xreg = NULL,
```

```

test = c("kpss", "adf", "pp"), test.args = list(),
seasonal.test = c("seas", "ocsb", "hegy", "ch"),
seasonal.test.args = list(), allowdrift = TRUE, allowmean = TRUE,
lambda = NULL, biasadj = FALSE, parallel = FALSE, num.cores = 2,
x = y, ...)

```

One last thing needed to be calculated before using `auto.arima()` is lambda function. In our case it could be done by using Box Cox transformation parameter. It is implemented by another function in “RStudio” – `BoxCox.Lambda()`, where inside an actual function we insert calculated time series.

*BoxCox.lambda(tsData)*

Calculated result of  $\lambda$  by “RStudio” in case of working with DVA1 data detector for Monday is:  $\lambda=1$ , now the result could be used in `auto.arima()` function. First ARIMA model will be implemented without seasonality, in short term it will be giving an actual prediction, however in a long term forecast function will go to a straight line. The implementation showed below:

```

fit1 <- auto.arima(tsData, stepwise=FALSE, seasonal=FALSE, trace=TRUE,
lambda=1)

```

Checked ARIMA models parameters are set up by AIC test result and presented in the table below:

Model	AIC result
ARIMA(0,0,0)	with zero mean : 13261.01
ARIMA(0,0,0)	with non-zero mean : 11789.85
ARIMA(0,0,1)	with zero mean: 11889.64
ARIMA(0,0,1)	with non-zero mean : 10630.48
ARIMA(0,0,2)	with zero mean: 11053.96
ARIMA(0,0,2)	with non-zero mean : 10039.64
ARIMA(0,0,3)	with zero mean: 10451.65
ARIMA(0,0,3)	with non-zero mean : 9619.931
ARIMA(0,0,4)	with zero mean: 10097.21
ARIMA(0,0,4)	with non-zero mean : 9389.293
ARIMA(1,0,0)	with zero mean : 9772.648
ARIMA(1,0,0)	with non-zero mean : 9207.024
ARIMA(1,0,1)	with zero mean: Inf
ARIMA(1,0,1)	with non-zero mean : 8598.968



ARIMA(1,0,2)	with zero mean: Inf
ARIMA(1,0,2)	with non-zero mean : 8455.746
ARIMA(1,0,3)	with zero mean: Inf
ARIMA(1,0,3)	with non-zero mean : 8445.655
ARIMA(1,0,4)	with zero mean: Inf
ARIMA(1,0,4)	with non-zero mean : 8437.215
ARIMA(2,0,0)	with zero mean: Inf
ARIMA(2,0,0)	with non-zero mean : 8469.767
ARIMA(2,0,1)	with zero mean: Inf
ARIMA(2,0,1)	with non-zero mean : 8456.11
ARIMA(2,0,2)	with zero mean: Inf
ARIMA(2,0,2)	with non-zero mean : 8455.227
ARIMA(2,0,3)	with zero mean: Inf
ARIMA(2,0,3)	with non-zero mean : 8446.945
ARIMA(3,0,0)	with zero mean: Inf
ARIMA(3,0,0)	with non-zero mean : 8452.337
ARIMA(3,0,1)	with zero mean: Inf
ARIMA(3,0,1)	with non-zero mean : 8453.706
ARIMA(3,0,2)	with zero mean: Inf
ARIMA(3,0,2)	with non-zero mean : 8449.757
ARIMA(4,0,0)	with zero mean: Inf
ARIMA(4,0,0)	with non-zero mean : 8454.877
ARIMA(4,0,1)	with zero mean: Inf
ARIMA(4,0,1)	with non-zero mean : 8450.418
ARIMA(5,0,0)	with zero mean: Inf
ARIMA(5,0,0)	with non-zero mean : 8457.901

Table 2: ARIMA models

After AIC test we could determine a best-fit model for a short-term prediction, as ARIMA(1,0,4) with non-zero mean. To build a graph using “RStudio” we use a following code:

```
fcast <- forecast(fit1, h = 500)
plot(fcast)
```

To indicate seasonality the ACF and PACF function is being created in “RStudio” and correlogram graphs are plotted. Correlogram is always starting from 1, so the started lag is being shifted. Implemented code and correlograms showed below:

```
#ACF and PACF correlograms starting from lag 1
Acf(tsData, lag = length(tsData)-1)
```

$Pacf(tsData, lag = length(tsData)-1)$

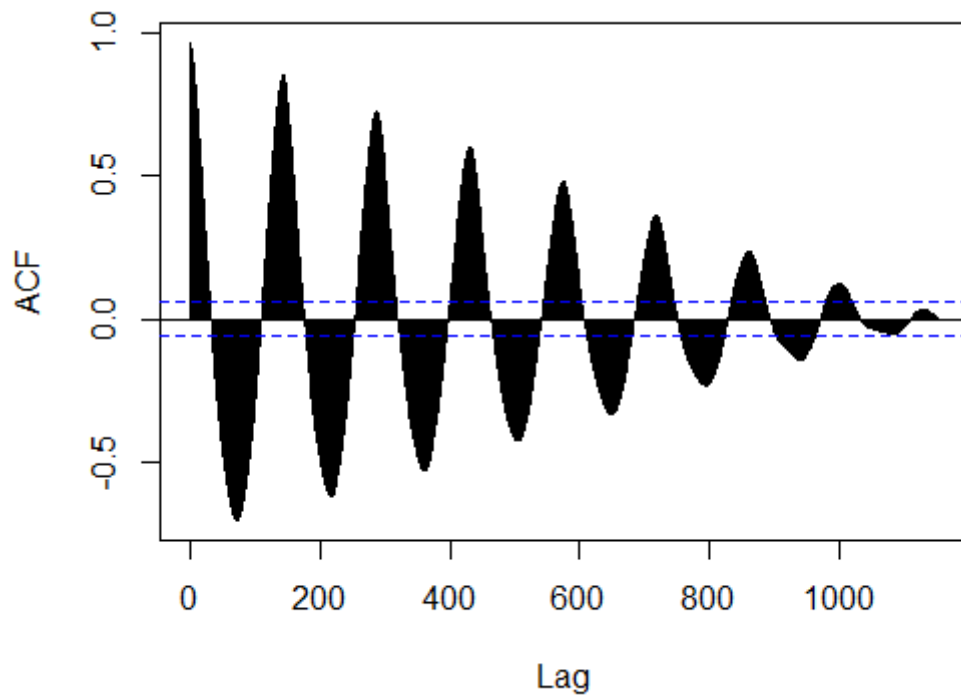


Figure 20: ACF

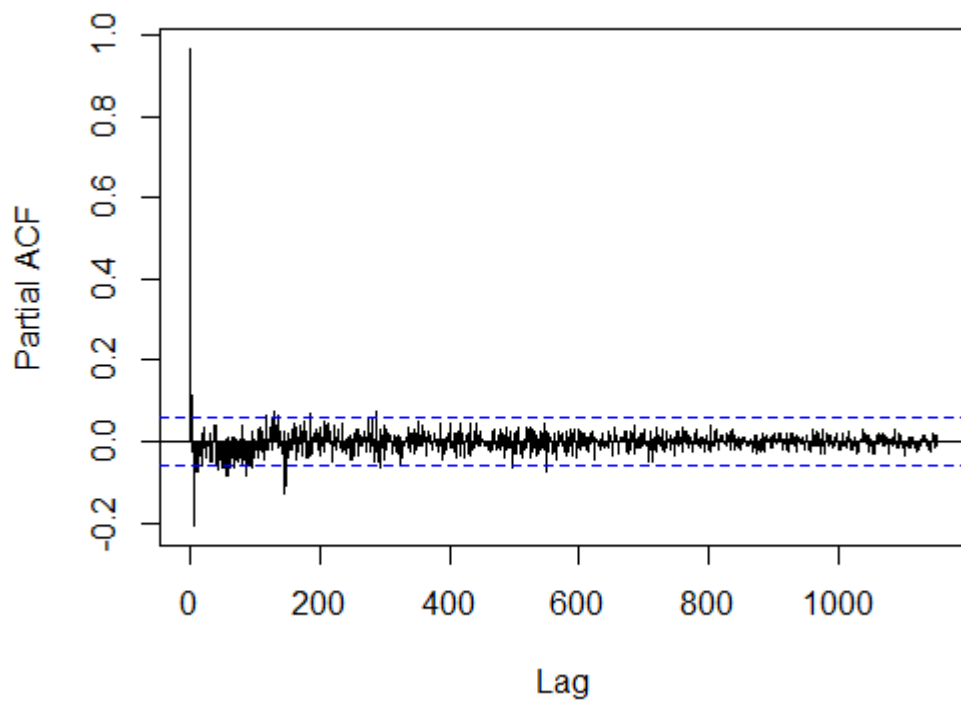
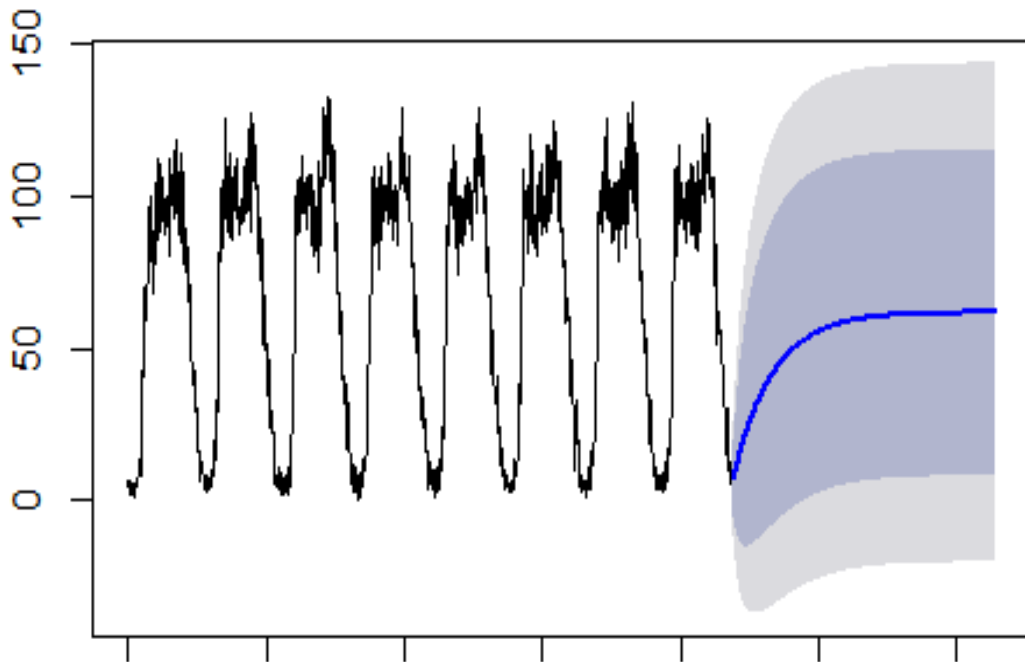


Figure 21: PACF

Graph for 500 steps is plotted by previous code and showed below. We could see that in a long term function starts to get closer to a straight line so our suggestion is being approved. Talking about closest forecast the value of the first five predicted steps are: 7.309917, 8.078554, 8.662148, 9.457279, 10.157212.



*Figure 22: Long-Term Prediction for DVA1 Mondays*

Now the predicted values and actual values will be compared and residuals between them will be calculated, meanwhile the model itself will be constantly expanded by adding actual values to the created ARIMA model. In additional .txt file added data from Monday (02.07.2018), so the time series graph represented 9 days showed below:

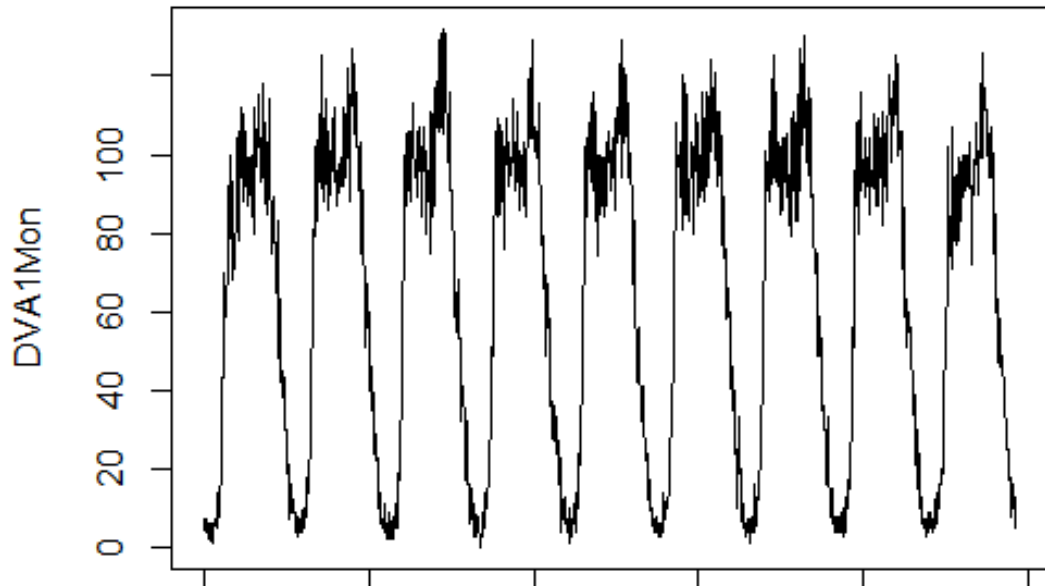


Figure 23: DVA1 for Mondays with Additional Day

In implemented “Rstudio” code data from .txt file is being imported. Afterwards time series (ts) object is being created with start and end value, to obtain new values the end value is being consistently changed to end value plus one in order to “teach” ARIMA model and the predicted values (10 values in our case) are generated and stored to “Excel” file as predicted values. Implemented programming code showed below:

```
library(seasonal) # install.packages("seasonal")

library(forecast) # install.packages("forecast")

library(fpp2) # install.packages("fpp2")

setwd('E:\\Data')

dataIn3 <- read.table("dva1monPredict.txt", header=TRUE, sep=";")

tsData4 <- ts(dataIn3, start = 1, end = 1155)

fit4 <- auto.arima(tsData4, seasonal.test = "seas", stepwise=FALSE, seasonal=TRUE, trace=TRUE, lambda=1)
```

```
fcast4 <- forecast(fit4, h = 10)
```

```
fcast4
```

Programming code could be improved in our case by adding a loop, that will have an end value of analyzed time series as an variable which changes every interaction to end value plus one. In provided example implemented code will return 10 forecasted values from lag 1165 to lag 1175, the values are being transferred to integer values, also “RStudio” returns values for 95 and 80 percent confidence interval:

```
for (var in c(1165:1175)){
  tsData4 <- ts(dataIn3, start = 1, end = var)
  fit4 <- auto.arima(tsData4, seasonal.test = "seas", stepwise=FALSE, seasonal=TRUE, lambda=1)
  fcast4 <- forecast(fit4, h = 1)
  print(fcast4)
}
```

Real data to forecasted values and differences between them (residuals) for 02.07.2018 showed below in graphical representation, blue line represents an actual data from DVA1 data detector and red line represents predicted values with ARIMA (1, 0, 4) model:

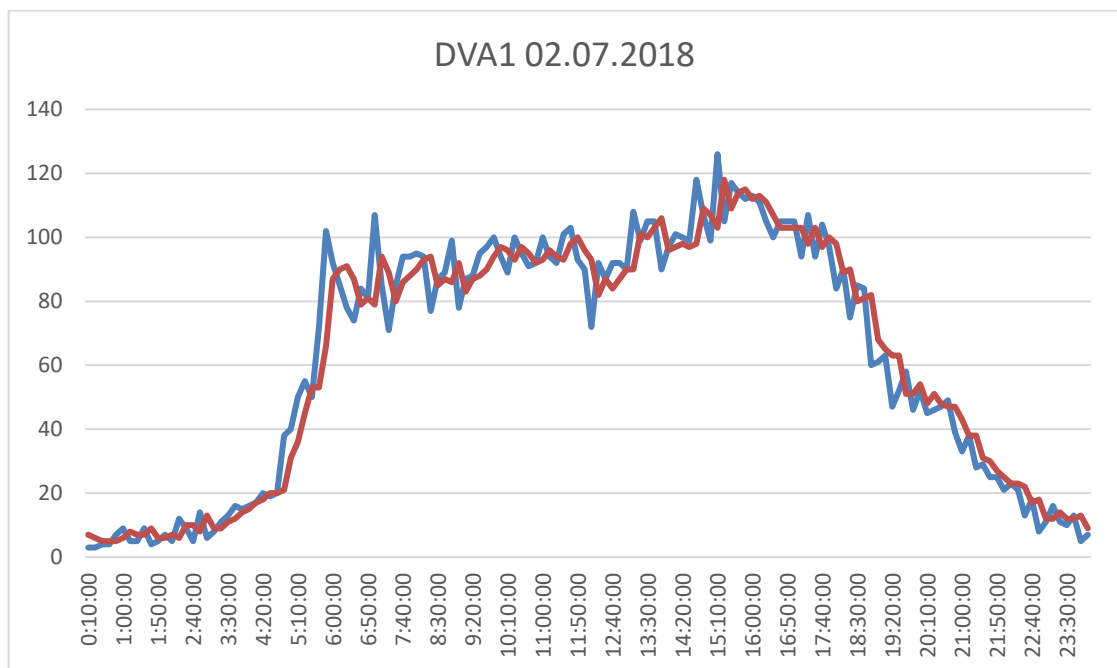


Figure 24: DVA1 Real and Forecasted Values

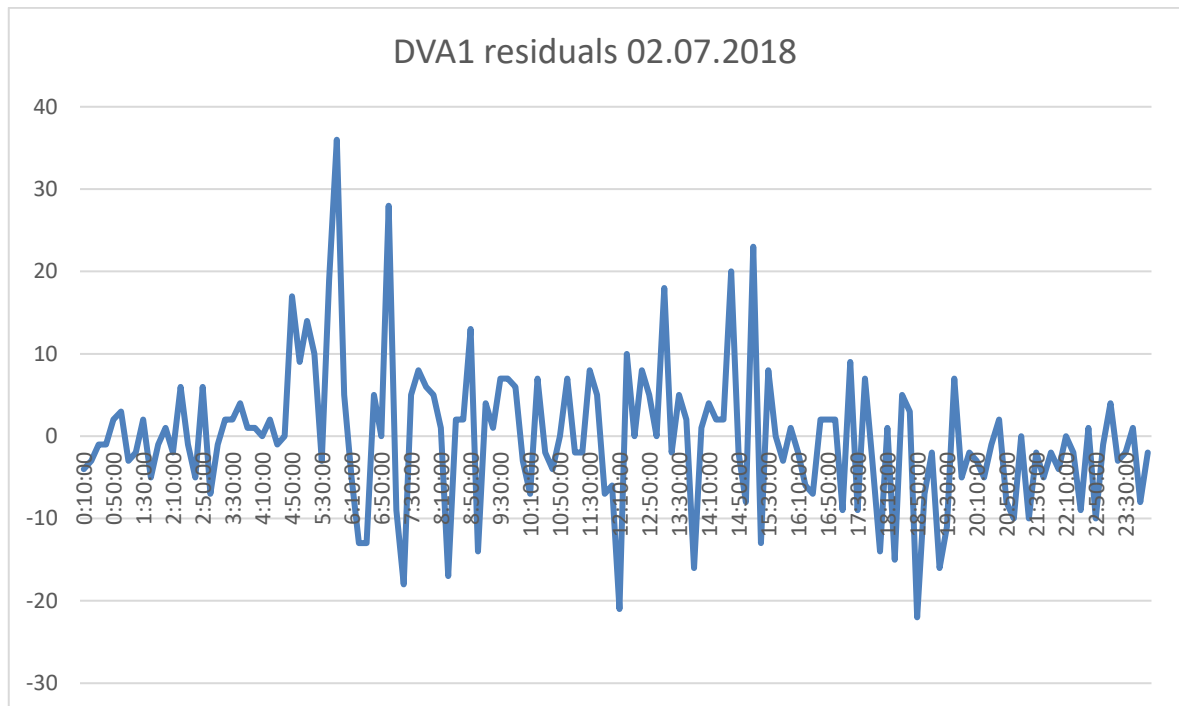


Figure 25: DVA1 Residuals

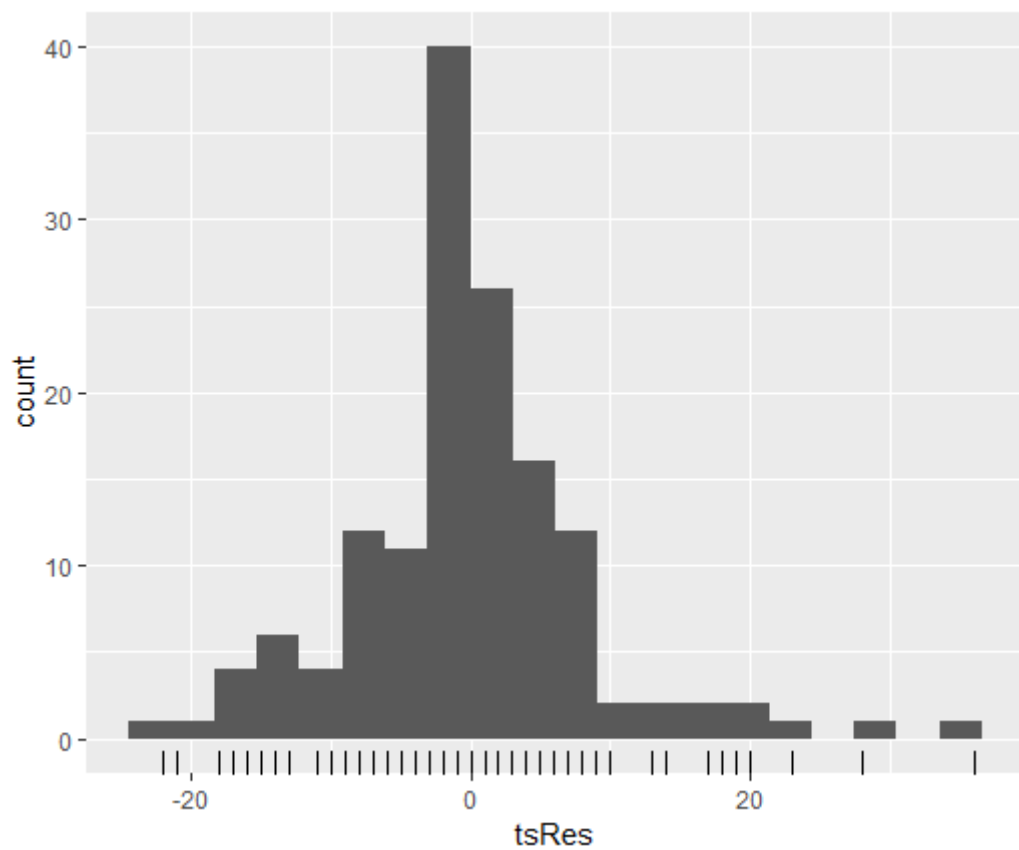
By analyzing residuals and comparing both: real and predicted data graphs we can say that, beside sharp peak values in real data graph, forecast is accurate. Graph of residuals looks like a white noise, which approved by normal distribution function of residuals (built in “RStudio” with code showed below, firstly residuals stored in .txt object in the working directory):

```
#Distribution histogram of residuals
```

```
residuals <- read.table("dva1monResiduals.txt", header=TRUE, sep=";")
```

```
tsRes <- ts(residuals)
```

```
gghistogram(tsRes)
```



*Figure 26: Residuals Distribution*

The right part of residuals distribution with 0 mean is looks a little longer than left part, however limited amount of residuals data (144) allows to make a conclusion about normal distribution of residuals consequently created with ARIMA(1,0,4) is precise for the forecasts on the investigated time series.

## CONCLUSION

In general data prediction is complicated and not ambiguous task, however in our days by means of up to date technologies and programs it is possible to precisely forecast data sets. Software environments contain complicated statistical calculations and algorithms and allow users to tremendously simplify the process of data analysis and data prediction. Not to mention that field is rapidly developing and new, more powerful software is being continuously developed.

Whatsoever, during the process of forecast development it is necessary to follow restrictions and certain flow in order to create a model that will describe process precise and actual data will fit predicted values.

From the inception of data prediction and data analysis it is one of the most important and progressive fields that could be implemented in all parts of human life, from urbanistic and ecological to economical and financial. While creating accurate data forecast for the future let humanity know the best behavior towards the process which is described by analyzed data. All that gives us incredible advantage in sense of knowing the future, by knowing future events and processes incomes.

A project helps to understand theoretical and practical components of modern data analysis and data prediction, beside that gives practical illustration of forecast usage in real urban crossroad.



**BIBLIOGRAPHY**

- [1] Hyndman, J. R., Athanasopoulos, G. (2018). Forecasting: principles and practice, 2<sup>nd</sup> edition. [online] Available at: <https://otexts.com/fpp2/>
- [2] Dalinina, R., (2017). Introduction to forecasting with ARIMA in R. [online] Available at: <https://www.datascience.com/blog/introduction-to-forecasting-with-arma-in-r-learn-data-science-tutorials>
- [3] Torres-Reyna, O., (2013). Introduction to RStudio, v1.3. [online] Available at: <https://dss.princeton.edu/training/RStudio101.pdf>
- [4] Mahendra, R. G., (n.d). Forecasting techniques. Retrieved 17 April 2019 from: <http://nsdl.niscair.res.in/jspui/bitstream/123456789/829/1/CHAPTER-6%20FORECASTING%20TECHNIQUES-%20Formatted.pdf>
- [5] Wu, M., (2018). PwC Approach – The Data and Analytics Framework. [online] Available at: <https://medium.com/next-thoughts/pwc-approach-the-data-and-analytics-framework-6ff5c8a72dd9>
- [6] Brockwell, P. J., Davis, R. A., (1991). Time Series: Theory and Methods, second edition.
- [7] Alonso, C., (n.d.). Econometrics. Topic 8: Autocorrelation. [online] Available at: [http://ocw.uc3m.es/economia/econometrics/lecture-notes-1/Tema8Autocorrelacionlogo\\_Eng.pdf](http://ocw.uc3m.es/economia/econometrics/lecture-notes-1/Tema8Autocorrelacionlogo_Eng.pdf)
- [8] Galrinho, M., (2016). Least Squares Methods for System Identification of Structured Models. [online] Available at: <https://www.diva-portal.org/smash/get/diva2:953835/FULLTEXT01.pdf>
- [9] Geiss, C., (2011). Stochastic Modeling (March 4,2011). [online] Available at: <http://users.jyu.fi/~geiss/lectures/models.pdf>
- [10] Taylor, M. T., Samuel, K., (1998). An Introduction To Stochastic Modeling 3<sup>rd</sup> edition. Academic Press. [online] Available at: <https://www.ime.usp.br/~fmachado/MAE5709/KarlinTaylorIntrodStochModeling.pdf>
- [11] Jones, C., (2014). LOESS Seasonal Decomposition as a Forecasting Tool (April 16, 2014). [online] Available at: <http://businessforecastblog.com/loess-seasonal-decomposition-as-a-forecasting-tool/>

- [12] Hyndman, R., (n.d.). Forecasting: Principles and Practice. 5. Time series decomposition and cross-validation. [online] Available at: <https://robjhyndman.com/uwafiles/5-Cross-validation.pdf>
- [13] Google Maps, (2019). MAKRO Cash & Carry, 1:1.500. Google Maps [online] Retrieved 25 April 2019 from: <https://www.google.ru/maps/place/>
- [14] Pelgrin, F., (2011). Lecture 5 Box-Jenkins methodology (Sept. 2011- Dec. 2011) [online] Available at: [https://math.unice.fr/~frapetti/CorsoP/Chapitre\\_5\\_IMEA\\_1.pdf](https://math.unice.fr/~frapetti/CorsoP/Chapitre_5_IMEA_1.pdf)

**LIST OF ABBREVIATIONS**

ARIMA	Autoregressive Integrated Moving Average
AR	Autoregressive
MA	Moving Avarage
KPSS	Kwiatkowski–Phillips–Schmidt–Shin
ACF	Autocorrelation Function
PACF	Partial Autocorrelation Function
TS	Time Series
ARCH	Autoregressive Conditional Heteroscedasticity
LOESS	Locally-Weighted Scatterplot Smoothing
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
AIC	Akaike’s Information Criterion
CPS	Computer Power System
STL	Seasonal and Trend decomposition using Loess
MLE	Maximum Likelihood Estimation

**LIST OF FIGURES**

Figure 1: Process of Observation and Learning.....	9
Figure 2: Data Analysis Flow.....	10
Figure 3: Model Creation Flow.....	11
Figure 4: Input Signal .....	15
Figure 5: Output Signal.....	16
Figure 6: Input ACF.....	16
Figure 7: Output ACF.....	17
Figure 8: Data set for Loess Decomposition.....	22
Figure 9: Window in Data Set for Loess Decomposition.....	22
Figure 10: Fitted Curve for Loess Decomposition .....	23
Figure 11: Box-Jenkins Methodology.....	24
Figure 12: Data Set Separation.....	27
Figure 13: Satellite Situational Plan.....	30
Figure 14: Situational Plan.....	31
Figure 15: DVA1 Time Series.....	33
Figure 16: DVA2 Time Series.....	34
Figure 17: DVA1 Time Series for Mondays.....	35
Figure 18: DVA2 Time Series for Mondays.....	35
Figure 19: DVA1 Detectors On Monday in R.....	37
Table 2: ARIMA models.....	38
Figure 20: ACF.....	40
Figure 21: PACF.....	41
Figure 22: Long-Term Prediction for DVA1 Mondays.....	41
Figure 23: DVA1 for Mondays with Additional Day.....	42
Figure 24: DVA1 Real and Forecasted Values.....	43

---

Figure 25: DVA1 Residuals.....44

Figure 26: Residuals Distribution.....45

**LIST OF TABLES**

Table 1: Data Detectors ..... 31

Table 2: ARIMA models ..... 38