DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF SOCIAL SCIENCES
DEPARTMENT OF BUSINESS ADMINISTRATION
BUSINESS INFORMATION SYSTEMS PROGRAM
MASTER THESIS

FORECAST OF WORK REQUEST FOR TELECOMMUNICATION FIELD OPERATION PLANNING

Gökhan Görkem ARPAKÇI

Supervisor
Prof. Dr. Efendi NASIBOĞLU

İZMİR - 2016

MASTER THESIS/PROJECT APPROVAL PAGE

University : Dokuz Eylül University

Graduate School : Graduate School of Social Sciences

Name and Surname : Gökhan Görkem ARPAKÇI

Title of Thesis : Forecast Of Work Request For Telecommunication Field Operation Planning

Defence Date : 19.08.2016

Supervisor : Prof.Dr.Efendi NASİBOĞLU

EXAMINING COMMITTE MEMBERS

Title, Name and Surname	University	Signature
Prof.Dr.Efendi NASİBOĞLU	DOKUZ EYLUL UNIVERSITY	
Doç.Dr.Sabri ERDEM	DOKUZ EYLUL UNIVERSITY	
Doç.Dr.Ali MERT	EGE UNIVERSITY	

Unanimity (✓)

Majority of votes ( )

The thesis titled as "**Forecast Of Work Request For Telecommunication Field Operation Planning**" prepared and presented by Gökhan Görkem ARPAKÇIis accepted and approved.

Prof.Dr.Mustafa Yaşar TINAR
Director

DECLERATION

I hereby declare that this master thesis titled as “Forecast Of Work Request For Telecommunication Field Operation Planning” has been writted by myself in accordance with the academic rules and ethical conduct. I also declare that all materials benefited in this thesis consist of the mentioned resources in the reference list. I verify all these with my honour.

../../…

Gökhan Görkem ARPAKÇI

ABSTRACT

Master Thesis

Forecast of Work Request For Telecommunication Field Operation Planning

Gökhan Görkem ARPAKÇI

Dokuz Eylul University
Institute of Social Sciences
Department of Business Administrion
Business Information Systems Program

Prediction is always being important to understand the future for all humans. Nowadays if all the firms in the industry are considered as a living organism, understanding the future is quite benefitical for reducting their cost and increasing their profits.

Latest developments on technology directly provide to reach more data day by day. If Telecom firms are considered in the industry, the main aim of Telecommunication firms are to make plan to assest their limited resoruces in more efficent way and lead these resources more optimum way with using forecast technique.

In our reseach, our goal is to use real time several service data which have been created between 2014 and 2016, of land services of a leading western Europe Telecommunication company in Italy and research on literature to find several models which are fit on these data and create a predictive model to forecast out of date week. Finally obtain lesser error rate.

Keywords: Telecommunication, prediction, forecast, Random Forest, Extreme Gradient Boosting, ARIMA, field operation

ÖZET

Yüksek Lisans Tezi

Telekomünikasyon Alan Operasyonlari İçin İş Yükü Tahminlemesi

Gökhan Görkem ARPAKÇI

Dokuz Eylül Üniversitesi
Sosyal Bilimler Enstitütsü
İngilizce İşletme Anabilim Dalı
İşletme Bilişim Sistemleri Programı

Tahminleme çağlar boyunca geleceği daha iyi anlayabilmek kaygısı ile insanlık için oldukça önemli olmuştur. Günümüzde çeşitli endüstrilerde faaliyet gösteren firmaları birer canli organizma olarak düşünürsek, yarını anlamak firmalar icin özellikle maliyetlerini azaltmak ve daha çok kar elde etmek için önemli olmaktadir.

Teknolojinin gelişmesi ile veriye olan erişim günden güne artmiş ve şirketler daha cok veriyi kontrol eder hale gelmistir. Endüstri icersinde Telekomünikasyon firmalarını baz alırsak, Telekom firmalarının amacı, kısıtlı kaynaklarını en ideal şekilde yönetebilmek için öncelikli olarak planlama yapmak ve planlama neticesinde çeşitli tahminleme yöntemlerini kullanarak elindeki kaynağı en optimum şekilde yönetmektir

Araştırmamızda batı Avrupa'nın önde gelen Telekomünikasyon şirketinin Alan Operasyonları biriminin 2014-2016 yılları arasında elde edilen gerçek zamanlı günlük iş yükü verilerini kullanarak olası iş yükünü haftalik olarak Random Forest, Extreme Gradient Boosting ve ARIMA yöntemleri ile en düşük hata payı elde edilecek şekilde tahminleme hedeflenmiş ve bu yönde literatür taranmıştır

Anahtar Kelimeler: Telekomünikasyon, Tahminleme, Random Forest, Extreme Gradient Boosting, ARIMA, Alan Operasyonları

FORECAST OF WORK REQUEST FOR TELECOMMUNICATION FIELD OPERATION PLANNING

CONTENTS

CHAPTER ONE
THEORETICAL BACKGROUND

CHAPTER TWO

DATA PREPARATION AND PROCESSING

CHAPTER THREE

DATA ANALYSIS

CHAPTER FOUR

RESULTS

ABBREVETIONS

BIP Business Integration Partners

FTE Full Time Equivalent

WR Work Request

AIA All Italy Area

AIL All Italy Locational

AIU All Italy Urban Area

MAPE Mean Absoulte Percentage Error

RAM Random Access Memory

GB Gigabyte

CPU Central Process Unit

XGB Extreme Gradient Boosting

XGBoost Extreme Gradient Boosting

AR Auto-Regression

MA Moving Average

ARMA Auto-Regressive Moving Averages

ARIMA Auto-Regressive Integrated Moving Averages

B.C Before Christ

MAE Mean Absolute Error

RMSE Root Mean Square Error

ACF Auto Correlation

PACF Patrial Auto Correalation

AIC Akeike Information Criteria

CART Classification and Regression Trees

RF Random Forest

OOB Out-of-Bag

ANN Artificial Neural Network

ID Identification

ETL Extraction, Transformation, Loading

KNIME Konstanz Information Miner

KPI Key Performance Indicator

ADSL Asymmetric Digital Subscriber Line

IQR Interquartile Range

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

INTRODUCTION

Time series forecasting is a dynamic research area which has attracted interests of researchers over last few decades. The main aim of time series forecasting is to carefully collect and absolutely study the past observations of a time series to develop a fitted forecast model which describes the innate structure of the series. This forecast model is then used to generate future values for the series, i.e. to make forecasts. “Time series forecasting thus can be termed as the act of predicting the future by understanding the past”. [1]

Forecasting is mainly using in all industries which are based on production by time. As manufacturing is uses forecasting to efficient inventory management, retail industry uses in order to sales and item prediction. Financial industry uses for understanding future trends and etc.

Forecasting in telecommunication is highly crucial, because of the fact that telecommunication is one of the fastest growing and developing industry. As it mentioned, forecasting is not only used in sales or production forecasts but also can use workload prediction in fields.

The reason of our research began in this point. Predict the time series based field workload of a Leading Western Europe Telecom Company.

Motivation of Research

In December 2015, Business Integration Partners which is a leading Milan based International Information Technology and Management Consulting firm had a project of a Leading Western Europe Telecom Company to forecast weekly work request of each location and job type and create a predictive model.

The Leading European Telecommunication firm is one of the biggest telecommunication company of Europe. They provide telephone, internet and GSM services across Italy and Europe.

Business Integration Partners (Bip) which is a leading Milan based International Information Technology and Management Consulting. Founded in 2003, Bip today employs more than 1200 professionals, who deliver management consulting and business integration services supporting companies in the research and adoption

[1] Raicharoen T , Lursinsap C. and Sanguanbhoki P., “Application of critical support vector machine to time series prediction”, **Circuits and Systems, 2003. ISCAS ’03.Proceedings of the 2003 International Symposium on Volume 5**, pages: V-741-V-744, 25-28 May, 2003,

of disruptive technological innovation. Today, Bip export our professional services, operating outside Italy for an increasing number of international clients. Strong relationships with local stakeholders and focused acquisitions have allowed us to extend our networks and establish ourselves as a trusted advisor in new target markets [2].

Since April 2015, I have been working at Bip as several positions, thanks to my managers and supervisors, they let me to assign regarding project to research.

The objective is to find the best prediction model on all job types and locations to leading European Telecommunication Company with using their data. The *work request* is daily obtained Full Time Equivalent (FTE) per any moment. 1 FTE is equal to 1 work request (WR). The data are beginning from 1st of January 2014 to 19th of May 2016.

The data consist of three data table, they are Work Request, Job Type and Location. The detailed information exists in Data Preparation and Processing chapter. In location there are three sets and they consist each other. The location data bases are:

Table 1: Locations

Location Sets	Abbreviations	Number of Locations
All Italy Area	AIA	4
All Italy Locational	AIL	27
All Italy Urban Area	AIU	74
All Macro Area	MAC	571

According to Table 1 the location is divided by 4 main area, and all of the four areas own 27 sub Location areas and all the locational areas have 74 sub urban areas. Finally all sub areas own 571 Macro areas. The object is find a predictive model according to all Italy urban area (AIU).

In Job Type data, we have 2 level of Jon type, Level I and Level II. Level I consists 3 variables, and Level II consist 13 variables.

[2] Business Integration Partners, "About Us".
https://www.businessintegrationpartners.com/about-us/ (24 July 2016)

Table 2: Level I Job Types

Level I Job Types:
Support
Distributions OL construction and transformation Retail/Wholesale
Distributions OL Permute

Table 3: Level II Job Types

Level II Job Types
Support Fibre
Support NOF Cable Fault
Support Data Products
Support Telephone Products
Support RA SLA DAY
Support RA SLA HOUR
Distributions construction and transformations fibres (with intervention)
Distributions construction and transformations Telephone /ADSL (with intervention)
Distributions Data SOL/CDN/TRANSITS
Distributions Data Products
Distributions Telephone Products
Distributions Permute
Distributions OL Permute

The last data table is work request. It consist time and location based FTE creation. In total there are 29,6 million work requests.

The data are parted into two sub datasets, training and testing. Training data are from 1st of January 2014 to 31st of December 2016. Testing data are from 1st of January 2016 to 13th of May 2016. Finally the objective data which we compare the result are from 14th of May 2016 to 19th of May 2016. Detailed Information about data consist in Data Preparation and Processing and, Data Analysis and Result chapters.

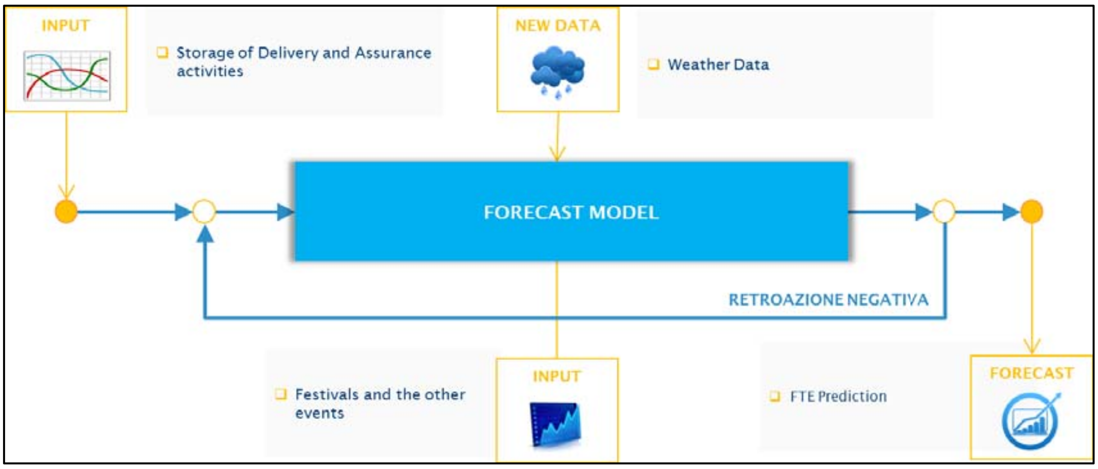


Figure 1: Schema of the Research

The Structure of the Research

Our research begins with theoretical background to understand data and type of the data, previous research on same or similar subject and analysis methods. Following chapter is data preparation and processing, how the data explore, clean and enrich, and adding in additional variables. Next chapter is data analysis. According to Theoretical background chapter we decide three analysis methodology which are Autoregressive Integrated Moving Averages, Extreme Gradient Boosting and Random Forest. In Data Analysis chapter first we show the data with time series graph we apply these three data analysis methods. Next chapter results, we obtain the results and their Mean Absolute percentage Error (MAPE) and compare with actual value. Final part is Conclusion. We interpret the result according to our objective and mention about next research. In our research we use 8 GB RAM 500 GB Hard Disk. Intel® Core™ I5-5200U CPU with 2.2Ghz Processor Microsoft Windows 7 personal computer.

CHAPTER ONE
THEORETICAL BACKGROUND

In this chapter we give an overview of context of our research. This chapter begins with descriptions of forecast and its features, forecast data types, detailed information about forecast data types of time series and several time series forecasting methods and their examples which are provided a basis of our research.

1.1 Forecasting

Forecasting can be broadly considered as a method or a technique for estimating many future aspects of business or other operation [(3)] .The goal of forecasting is to come to possible to an accurate picture of the future [(4)]

Pioneer forecasting applications could be seen on ancient civilisation as weather forecasting. 650 B.C Babylonians predicted weather from cloud patterns [(5)]. Nowadays forecasting is used not only weather forecasting but also used in other disciplines as economy and finance (Governments, businesses, policy organizations, central banks, financial services firms, and economic consulting firms around the world routinely forecast the major economic variables, such as gross domestic product as known as GDP, unemployment, consumption, investment, the price level, and interest rates), business and all its subfields as operations management and control (hiring, production, inventory, investment), marketing (pricing distributing, advertising, ...) and accounting (budgeting using revenue and expenditure forecasts)[(6)], and industry as production and manufacturing [(7)],and workload[(8)].

The appropriate forecasting methods contingent upon data availability. If the conditions of numerical information about the past is available and it is reasonable to assume that some aspects of the past patterns will continue into the future are satisfied, then quantitative forecasting methods can be chosen, if there are no data available or

[3] Pradeep K. S. and Rajesh K., "Demand Forecasting For Sales of Milk Product (Paneer) In Chhattisgarh International Journal of Inventive Engineering and Sciences" (IJIES) ISSN: 2319–9598, Volume-1, Issue-9, August 2013

[4] INC.com , "Forecasting", http://www.inc.com/encyclopedia/forecasting.html ,(01 May 2016)

[5] NASA, "Weather Forecasting Through The Ages", http://earthobservatory.nasa.gov/Features/WxForecasting/wx2.php , (01 May 2016)

[6] Francis X. Diebold, “Forecasting in Economics, Business, Finance and Beyond”, pg 4. Edition 2015 Version Monday 14th December, 2015

[7] Callegaro A, "Forecasting Methods For Spare Parts Demand",(Master Thesis). 2010

[8] Aldor-Noiman S. , Feigin P.D and Mandelbaum, “A Workload Forecasting For A Call Center: Methodology And A Case Study", 2009

if the available data are not relevant to forecasts then qualitative forecast methods must be chosen [9].

1.1.1 Qualitative Forecasting Methods

The qualitative or judgmental approach can be useful in formulating short term forecast and can be also supplement the projections based on the use of any of the quantitative methods. Those methods are [10]:

- Executive Opinions
- Delphi Technique
- Sales Force polling (Expert Opinion Polls)
- Consumer Surveys

1.1.2 Quantitative Forecasting Methods

Quantitative methods place greatest reliance on representing developments numerically. Numerical data, of many types, are useful in thinking about longer-term developments, and to a certain extent they can be useful ways of presenting foresight results, too. [11]. Quantitative methods are separated into two kind, those are associative methods and time series methods.

1.1.2.1 Associative Methods

Associative models (often called causal models) assume that the variable being forecasted is related to other variables in the environment. They try to project based upon those associations [12].

1.1.2.1 Time Series Methods

A time series is a sequence of numerical data points in successive order, usually occurring in uniform intervals. In plain English, a time series is simply a sequence of numbers collected at regular intervals over a period of time [13].

[9] Hyndman R.J, and Athanasopoulos G Forecasting: Principles And Practice,. 2016 , https://www.otexts.org/fpp/1/ , (4 May 2016)

[10] Accounting, Financial, Tax, "Qualitative Forecasting Methods And Techniques", http://accounting-financial-tax.com/2009/04/qualitative-forecasting-methods-and-techniques/ ,(4 May 2016)

[11] European Comission For Learn, "Qualitative vs Quantitative Statistics", forlearn.jrc.ec.europa.eu/guide/4_methodology/meth_quanti-quali.htm , (4 May 2016)

[12] Mighty Mechanical, "Forecasting Fundamentals", http://mech.at.ua/Forecasting.pdf ,(4 May 2016)

[13] Investopedia, “Time Series”, http://www.investopedia.com/terms/t/timeseries.asp , (4 May 2016)

1.1.2.1.1 Components of a Time Series

According to [14] any time series can contain some or all of the following components:

- Trend(T)
- Cyclical (C)
- Seasonal (S)
- Irregular(I)

These components may be combined in different ways. It is usually assumed that they are multiplied or added, i.e.

$Y_t = T * C * S * I$ (1.1)

$Y_t = T + C + S + I$ (1.2)

To correct for the trend in the first case one divides the first expression by the trend (T). In the second case it is subtracted.

Trend: The trend is the long term pattern of a time series. A trend can be positive or negative depending on whether the time series exhibits an increasing long term pattern or a decreasing long term pattern.

Cyclical: Any pattern showing an up and down movement around a given trend is identified as a cyclical pattern. The duration of a cycle depends on the type of business or industry being analysed.

Seasonal: Seasonality occurs when the time series exhibits regular fluctuations during the same month (or months) every year, or during the same quarter every year.

Irregular: This component is unpredictable. Every time series has some unpredictable component that makes it a random variable. In prediction, the objective is to “model” all the components to the point that the only component that remains unexplained is the random component.

Also according to [15] called *Residuals* instead of *Irregular* component.

[14] IHMC, "Components of a time series" http://cmapskm.ihmc.us/rid=1052458821502_1749267941_6906/components.pdf ,(4 May 2016)

[15] Hyndman R.J, Athanasopoulos G , 6.1 Time Series Components (FPP) , (4 May 2016)

1.2 Evaluate the Accuracy

1.2.1 Scale-Dependent Errors

Let y_i denote the i_{th} observation and $\hat{y}_i$, denote a forecast of y_i : (1.3)

Basically the forecast error is $e_i = y_i - \hat{y}_i$ which is on the same scale as the data. Accuracy measures that are based on e_i are therefore scale-dependent and cannot be used to make comparisons between series that are on different scales [16] (1.4)

The two most commonly used scale-dependent measures are based on the absolute errors or squared errors:

Mean absolute error:

$MAE = \frac{1}{n}\sum(|e_i|)$ (1.5)

Root mean square error:

RMSE= $\sqrt{\frac{1}{n}\sum(|e_i|)}$ (1.6)

1.2.2 Percentage errors

According to same reference with *scale-depend errors* the percentage error is given by $p_i = 100\ e_i / y_i$. Percentage errors have the advantage of being scale-independent, and so are frequently used to compare forecast performance between different data sets. The most commonly used measure is:

Mean absolute percentage error:

MAPE= $= \frac{1}{n}\sum(|p_i|)$ (1.7)

1.3 Time Series Forecasting Methods

In this section, we researched several forecasting methods which are provided a basis of our research and their examples. This section includes Auto Regressive Integrated Moving Averages (ARIMA) method, Baseline Method which is the current forecasting method of the company, Artificial Neural Network, Extreme Gradient Boosting and Random Forest. All model examples were inspected as their accuracy evaluator as MAE, RMSE and MAPE values.

1.3.1 Auto Regressive Integrated Moving Averages (ARIMA)

[16] Hyndman R.J, Athanasopoulos G 2.5 Evaluating forecast accuracy ,(4 May 2016)

The acronym ARIMA stands for *Auto-Regressive Integrated Moving Average*. Lags of the stationarized series in the forecasting equation are called "*autoregressive*" terms, lags of the forecast errors are called "*moving average*" terms, and a time series which needs to be differenced to be made stationary is said to be an "*integrated*" version of a stationary series. Random-walk and random-trend models,
Autoregressive models, and exponential smoothing models are all special cases of ARIMA models [17]

The formula of ARIMA is:

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t \text{ (1.8)}$$

Where y'_t is the differenced series if they are non-stationary. The predictors on the right hand side include both y_t and lagged errors.

A non-stationary ARIMA model is classified as an *ARIMA(p,d,q)* model where:

- *p* is number of auto-regressive terms
- *d* is the number of non-stationary differences needed for stationary
- *q* is the number of lagged forecast errors in the prediction equation

Seasonal ARIMA models are usually denoted *ARIMA(p, d, q)(P, D, Q)$_m$* ,where m refers to the number of periods in each season, and the uppercase *P, D, Q* refer to the autoregressive, differencing, and moving average terms for the seasonal part of the ARIMA model[18][19].

1.3.1.1 ARIMA Steps

In this section is to give brief information about the application the ARIMA model which is using R. These steps are [20]:

1. Plot the data. Identify any unusual observations.
2. If necessary, transform the data (using a Box-Cox transformation[21]) to stabilize the variance.
3. If the data are non-stationary: take first differences of the data until the data are stationary.
4. Examine the ACF/PACF[22]: Is an AR(*p*) or MA(*q*) model appropriate?

[17] Duke University, "Introduction to ARIMA", http://people.duke.edu/~rnau/411arim.htm ,(4 May 2016)

[18] SAS Institute, "Notation for ARIMA Models". Time Series Forecasting System. (4 May 2016)

[19] Hyndman, Rob J; Athanasopoulos, G. "8.9 Seasonal ARIMA models (FPP) (5 May 2016)

[20] Hyndman, Rob J; Athanasopoulos, G "ARIMA modelling in R" (FPP) (5 May 2016)

[21] Li P. Box-Cox transformation Box-Cox Transformations: An Overview, http://www.ime.usp.br/~abe/lista/pdfm9cJKUmFZp.pdf , (5 May 2016)

[22] Duke University, "Identifying the numbers of AR or MA terms in an ARIMA model",

5. Try your chosen model(s), and use the AICc[23] to search for a better model.

6. Check the residuals from your chosen model by plotting the ACF of the residuals, and doing a portmanteau test [24] of the residuals. If they do not look like white noise, try a modified model.
7. Once the residuals look like white noise[25], calculate forecasts.

The R Package and the model that we used as Auto.Arima[26] only takes care of steps 3–6.

As it is seen on section 4, data analysis, we chose only stationary ARIMA researches, because of the fact that our time series data are stationary. ARIMA is good at following trends but falters when switching static to dynamic time series trend[27]

One of the best example [28] researchers obtained quite good accuracy with several ARIMA methods. According to this research, researchers obtained 0,052% MAPE value with ARIMA (1,0,0) and 0,064% MAPE value with ARIMA(0,2,1). One of the reason which affected the better result could be researchers historical data were 18 years monthly data (from January 1995 to January 2013).

In recent example [29] shows us ARIMA, in spite of the fact that researchers use MAE and RMSE, they gave the actual and forecast values thus we calculated MAPE value with using these values, researchers obtained ARIMA(2,0,1). According to this research, in January 2013 researchers calculated 9298,55 unit beef import, actual value was 9111,9 units therefore its MAPE value 2,05% however in February 2013 forecasted value was 10316,18 and actual value was 7892,70 therefore its MAPE value is 30,71%. It shows us ARIMA could be varied due to stationary data.

http://people.duke.edu/~rnau/411arim3.htm , (5 May 2016)

[23] NC State University "Hu S. Akaike Information Criterion" http://www4.ncsu.edu/~shu3/Presentation/AIC.pdf , (6 May 2016)

[24] Arranz M.A, "Portmanteau Test Statistics in Time Series", http://packages.tol-project.org/docs/ndmtest.pdf , (6 May 2016)

[25] New York University, "White Noise and Moving Average Mode" http://people.stern.nyu.edu/churvich/Forecasting/Handouts/Chapt3.1.pdf , (6 May 2016)

[26] Inside R, "Auto Arima: Fit best ARIMA model to univariate time series", http://www.inside-r.org/packages/cran/forecast/docs/Auto.Arima (6 May 2016)

[27] Simmhan Y, Aman S, Kumbhare A, Liu R, Stevens S, Zhou Q and Prasanna; "Cloud-Based Software Platform For Data-Driven Smart Grid Management", V, University of Southern California, Los Angeles, USA, 2013

[28] Rotela Junior P, Salomon F.L.R, Pamplona E.O; ARIMA: An Applied Time Series Forecasting Model for the Bovespa Stock Index, Institute of Production Engineering and Management, Federal University of Itajuba, Itajuba, Brazil 2014

[29] Sánchez-López E, Pérez-Linares C, Figueroa-Saavedra F. and Barreras-Serrano A ;"A Short Term Forecast For Mexican Imports Of United States Beef Using A Univariate Time Series Model", México 2015

Another a good example [30], researchers compared with ARIMA and one of the other Method which were used our research too, random forest. According to this research, researchers divided their data set into two group and applied ARIMA and random forest. They measured the accuracy with using MSE and the results are; for first group 26,9597 and second group 28,7412 . Examination of random forest takes place in Random Forest Section.

Auto.Arima and its modified version are used in several research, for instance [31] researches used Auto.Arima model in seasonal data and obtained fairy good results. Another research [32], Auto.Arima model was compared with ARMA model and the result of ARIMA was obtained quite well than ARMA model.

1.3.2 Extreme Gradient Boosting

XGBoost or Extreme Gradient Boosting is a machine learning algorithm that is designed, and optimized for boosted (tree) algorithms. The library aims to provide a scalable, portable and accurate framework for large scale tree boosting. It is an improvement on the existing Gradient Boosting technique [33].

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of weak prediction models, typically decision trees[34]. Boosting can be interpreted as an optimization algorithm on a suitable cost function like other boosting methods, gradient boosting combines weak learners into a single strong learner, in an iterative fashion [35].

XGBoost is fitted for supervised learning problems, where is used the training data to predict a target variable. The regularization term controls the complexity of the model, which helps to hold off overfitting. XGBoost is created a Tree Ensemble model which is a set of classification and regression trees (CART). It consists of three steps:

[30] Kane M.J, price N, Scotch M and Rabinowitz P, "Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks", , BMC Bioinformatics2014, Kane et al.; licensee BioMed Central Ltd. 2014

[31] Lippi M.,Bertini M, and Frasconi P, "Short-Term Traffic Flow Forecasting: An Experimental Comparison of Time-Series Analysis and Supervised Learning" , 2012

[32] Pasapitch Chujai, Nittaya Kerdprasop, and Kittisak Kerdprasop, Time Series Analysis of Household Electric Consumption with ARIMA and ARMA Models, 2013

[33] Chen T, “Xgboost”, https://github.com/tqchen/xgboost. (6 May 2016)

[34] Friedman J.H. Stochastic gradient boosting. Computational Statistics and Data Analysis, pg 367–378, 2002

[35] Jain A., Menon M N., Chandra S. Sales Forecasting for Retail Chains 2015

- Additive Training
- Model Complexity
- Structure Score

For example (36) researchers compare several methods as “Mean of each DayOfWeek”, Linear Regression, Random Forest Regression and XGBoost. As a result of this research XGBoost method had very promising results.

The algorithm of XGBoost is

Table 4: Pseudo-Code of Extreme Gradient Boosting

```
1. Initialize the outcome
2. Iterate from 1 to total number of trees
    2.1 Update the weights for targets based on previous run (higher for the ones mis-classified)
    2.2 Fit the model on selected subsample of data
    2.3 Make predictions on the full set of observations
    2.4 Update the output with current results taking into account the learning rate
3. Return the final output.
```

XGBoost gained popularity in data science after the famous Kaggle competition called Otto Classification challenge (37). In recent Kaggle Competition (38) (detailed information about Kaggle is placed on Data Part). Researcher created an XGBoost method and compare this with several methods and advanced to the lowest error rate and gained the first place of Rossmann Sales Store Kaggle competition.

1.3.3 Random Forest

Random forests (RFs) are an ensemble learning methods for both classification and regression problems (39). RF is a collection of decision trees that grow in randomly

[36] Jain A., Menon M N., Chandra S. Sales (FFRC) 2015

[37] Kaggle, "Otto Group Product Classification Challenge", https://www.kaggle.com/c/otto-group-product-classification-challenge ,(7 May 2016)

[38] Jacobusse G. Winning Model Documentation describing my solution for the Kaggle competition “Rossmann Store Sales” https://www.kaggle.com/c/rossmann-store-sales/forums/t/18024/model-documentation-1st-place, (7 May 2016)

[39] Breiman L ,"Random Forests". Machine Learning Vol:45 (1),pp: 5–32, 2001

selected subspaces of the feature space. The working principle of RF is to combine a set of binary decision trees (Breiman's CART – Classification and Regression Trees[40]), each of which are constructed using a bootstrap sample coming from the learning sample and a subset of features (input variables or predictors) randomly chosen at each node. Thus unlike the case in CART model building strategy, an individual tree in RF is built on a subset of learning points and on subsets of features considered at each node to split on. Moreover trees in the forest are grown to maximum size and the pruning step is skipped. After individual trees in ensemble are fitted using bootstrap samples, the final decision is obtained by aggregating over the ensemble, i.e. by averaging the output for regression or by voting for classification. This procedure called bagging improves the stability and accuracy of the model, reduces variance and helps to avoid overfitting [(41)]. The bias of the bagged trees is the same as that of the individual trees, but the variance is minimized by reducing the correlation between trees [(42)]. Breiman showed that random forests do not overfit as more trees are added, but produce a limiting value of the generalization error[(43)]. The RF generalization error is estimated by an out-of-bag (OOB) error, i.e. the error for training points which are not contained in the bootstrap training sets (about one-third of the points are left out in each bootstrap training set). An OOB error estimate is almost identical to the estimate obtained by N-fold cross-validation. The main advantage of RFs is that they can be fitted in one sequence, with cross-validation being performed along the way. The training can be terminated when the OOB error stabilizes. The algorithm of RF for regression shown in Fig. x of [(44)].

Table 5: Pseudo-Code of Random Forest

```
1. For $k$ = 1 to K:
   1.1. Draw a bootstrap sample L of size N from the training data.
   1.2. Grow a random-forest tree $T_k$ to the bootstrapped data, by recursively
repeating the following steps for each node of the tree, until the minimum node size
m is reached.
```

[40] Breiman L., Friedman J.H., Olshen R.A. and Stone C.J., "Classification and Regression Trees". Chapman & Hall, 1984

[41] Dudek G. "Short-Term Load Forecasting using Random Forests, Department of Electrical Engineering", Czestochowa University of Technology, 2011

[42] Hastie T., Tibshirani R. and Friedman J."The Elements of Statistical Learning. Data Mining, Inference, and Prediction" ,Springer, 2009

[43] Breiman L,"Random Forests". Machine Learning Vol:45 (1),pp: 5–32, 2001

[44] Hastie T., Tibshirani R. and Friedman J."The Elements of Statistical Learning. Data Mining, Inference, and Prediction" ,Springer,2009

```
        1.2.1. Select F variables at random from the n variables.
        1.2.2. Pick the best variable/split-point among the F.
        1.2.3. Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_k\}_k= 1, 2, ..., K$.
To make a prediction at a new point x:
        $f(x) = \frac{1}{K}\sum_{k=1}^{K} T_k(x)$
```

The two main parameters of RF are: the number of trees in the forest, denoted by *K*, and the number of input variables randomly chosen at each split, denoted by *F*. The number of trees can be determined experimentally. During the training procedure we add the successive trees until the OOB error stabilizes. The RF procedure is not overly sensitive to the value of F. The inventors of the algorithm recommend F = n/3 for the regression RFs. Another parameter is the minimum node size m. The smaller the minimum node size, the deeper the trees. In many publications m = 5 is recommended and this is the default value in many programs which implement RFs. However, RFs show small sensitivity to this parameter. It is noteworthy that using CART model, we get a classifier or an estimate of the regression function, which is a piecewise constant function obtained by partitioning the predictor space. This is a critical limitation of CART. But building an ensemble of CART we get results which are much smoother than results from a single tree. Using RFs, we can determine the prediction strength or importance of variables which is useful for ranking the variables and their selection, to interpret data and to understand underlying phenomena. The variable importance can be estimated in RF as the increase in prediction error if the values of that variable are randomly permuted across the OOB samples. The increase in error as a result of this permuting is averaged over all trees, and divided by the standard deviation over the entire ensemble. The more the increase of OOB error is, the more important is the variable.

As a very good example, [35] researchers investigated very similar subject as ours. Also their data type is likely to ours. They applied random forest and compared several methods as Artificial Neural Network, ARIMA, etc. Artificial Neural Network (ANN) and Random Forest (RF) gave the best accuracy in this research. However according to researchers creating RF model is less complicated than ANN model.

Another example on RF researchers observed that the relationship between the input variables and the output was well defined and the prediction accuracy was quite good.

As it was mentioned before [21] ,we compared two models as ARIMA and RF. Researcher divided their dataset into two part according to first part RF accuracy was four time better than (426,6%) ARIMA model and according to second data set RF was still fairly better than ARIMA model.

CHAPTER TWO
DATA PREPARATION AND PROCESSING

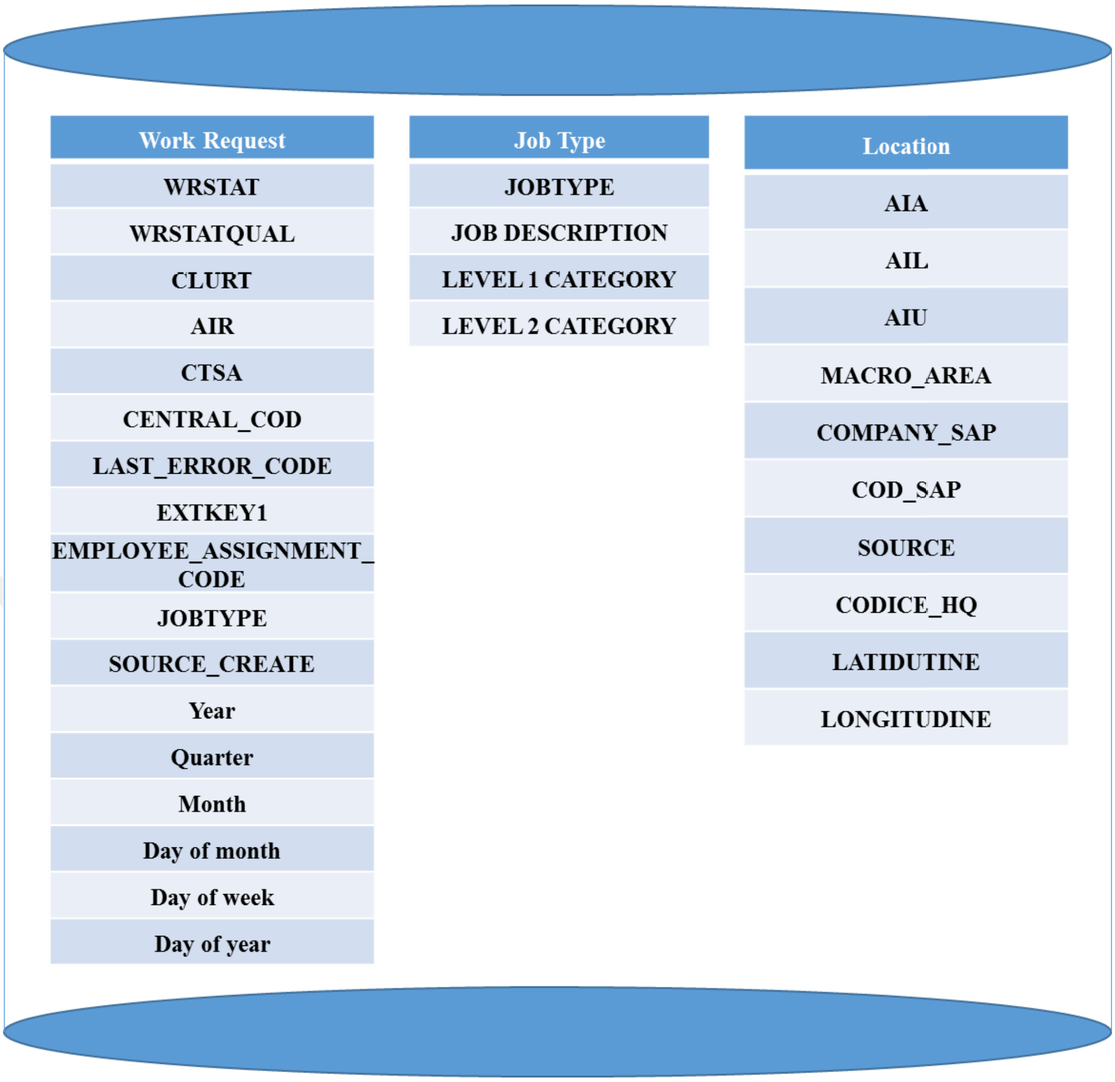
This chapter is about detailed information data processing. It includes the database of the company, data gathering approach of Business Integration Partners which it was followed by me, data transformation and data enrichment, and data feature engineering.

2.1 The Company Database

The Company has a special database for only recording and tracing Full Time Employee (FTEs) needs on its landline services. Every FTE's corresponds to a work request (WRs) which is our main data in this research. These WRs occurs after two type of work requests in all of Italy. These are Support Services and Distributions Services.

In this research, we interested The Company's three historical data tables which are about Work Request, Location and Job Type (Table 6). Time Stamp data tables consists the unique identification (ID) number of every single WR's with occurring time of regarding WR ID.

Table 6: The Company's database



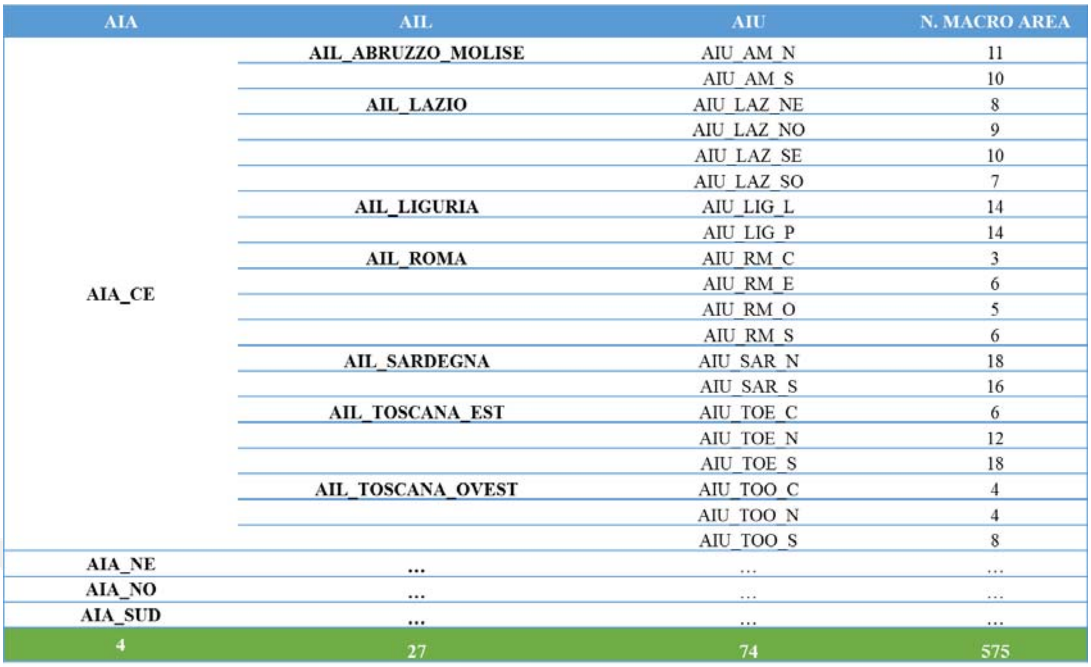
Work Request
WRSTAT
WRSTATQUAL
CLURT
AIR
CTSA
CENTRAL_COD
LAST_ERROR_CODE
EXTKEY1
EMPLOYEE_ASSIGNMENT_CODE
JOBTYPE
SOURCE_CREATE
Year
Quarter
Month
Day of month
Day of week
Day of year

Job Type
JOBTYPE
JOB DESCRIPTION
LEVEL 1 CATEGORY
LEVEL 2 CATEGORY

Location
AIA
AIL
AIU
MACRO_AREA
COMPANY_SAP
COD_SAP
SOURCE
CODICE_HQ
LATIDUTINE
LONGITUDINE

Location database, includes every unique WR's location less details to more details in Italy. In location data, the WRs which to be occurred in Italy was groped into four subgroups. Those are AIA, AIL, AIU and Macroarea. AIA is main four region of Italy. AIL has 27 subgroups (Table 7), AIU has 74 subgroups (Table 7) and Macroarea has 571 subgroups (Table 7).

Table 7: Location subgroups



AIA	AIL	AIU	N. MACRO AREA
AIA_CE	AIL_ABRUZZO_MOLISE	AIU_AM_N	11
		AIU_AM_S	10
	AIL_LAZIO	AIU_LAZ_NE	8
		AIU_LAZ_NO	9
		AIU_LAZ_SE	10
		AIU_LAZ_SO	7
	AIL_LIGURIA	AIU_LIG_L	14
		AIU_LIG_P	14
	AIL_ROMA	AIU_RM_C	3
		AIU_RM_E	6
		AIU_RM_O	5
		AIU_RM_S	6
	AIL_SARDEGNA	AIU_SAR_N	18
		AIU_SAR_S	16
	AIL_TOSCANA_EST	AIU_TOE_C	6
		AIU_TOE_N	12
		AIU_TOE_S	18
	AIL_TOSCANA_OVEST	AIU_TOO_C	4
		AIU_TOO_N	4
		AIU_TOO_S	8
AIA_NE	...	...	...
AIA_NO	...	...	...
AIA_SUD	...	...	...
4	27	74	575

Job type database consists type of work requests. They was grouped into two subgroup Category Level I, Category Level II, and number of job type. Level I Category has 3 unique variables and Level II Category has 13 unique variables and Number of Job Type has 1581 unique variables. (Table 8)

Table 8: Job Types

LEVEL 1 CATEGORY	LEVEL 2 CATEGORY	N° JOB TYPE
Support	Support Fibre	14
	Support NOF Cable Fault	8
	Support Data Products	9
	Support Telephone Products	227
	Support RA SLA DAY	78
	Support RA SLA HOUR	148
Distributions OL Cons. Trans, Retail/Wholesale	Distributions construction and transformations fibres (with intervention)	25

	Distributions construction and transformations Telephone /ADSL (with intervention)	318
	Distributions Data SOL/CDN/TRANSITS	578
	Distributions Data Products	47
	Distributions Telephone Products	78
Distributions OL Permute	Distributions Permute	36
	Distributions OL Permute	15
3	13	1581

In this research; we focused on WRSTAT and time based variables of Work Request database, AIU in Location Database and Category Level I and II of Job Type. The reason of the data selection is placed in Data Enrichment and Feature engineering part.

2.2 Data Gathering Approach

In this part, we give a brief information about the descriptions of KNIME Analytic Platform Tool and data gathering from The Company's databases with using KNIME analytic platform tool.

2.2.1 KNIME Analytic Platform Tool

KNIME, the Konstanz Information Miner, is an open source data analytics, reporting and integration platform. KNIME integrates various components for machine learning and data mining through its modular data pipelining concept. A graphical user interface allows assembly of nodes for data pre-processing (ETL: Extraction, Transformation, Loading), for modelling and data analysis and visualization [45] .In this research KNIME Analytic Platform version 2.12 was used.

2.2.2 Using KNIME Analytic Platform Tool for Data Gathering

[45] Wikipedia, "Knime Description" : https://en.wikipedia.org/wiki/KNIME, (9 May 2016)

The Company Data was stored per month in Comma Separated Values format (CSV). First of all it needed to be merged as a one data table which is consisted all months of 2014, 2015 and 2016. In that case List files and looping nodes of KNIME was used. (Figure 2)

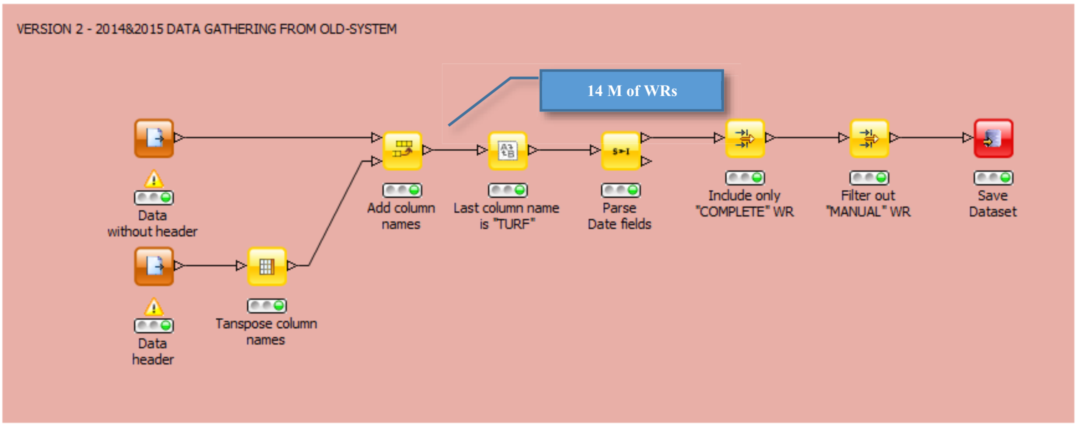


Figure 2: 2014, 2015, 2016 and Data Gathering from Old System (The Company's System)

Afterwards, three of The Company's databases was needed to be merged. KNIME join nodes (Figure 2) were chosen to manage that case. Before the join function WR data was summarized and grouped per day. Detailed information about this phase is placed in Data Enrichment part.

Finally dataset were filtered according to WRSTAT columns variable. In regarding column there were only two unique variables, they were "Complete" and "Manual". In this case "Complete" meant work request was available, and in "Manual" meant work request were not available which is seen on Figure 2 Therefore 39,6 million of rows filtered and decreased to 12,9 million rows.

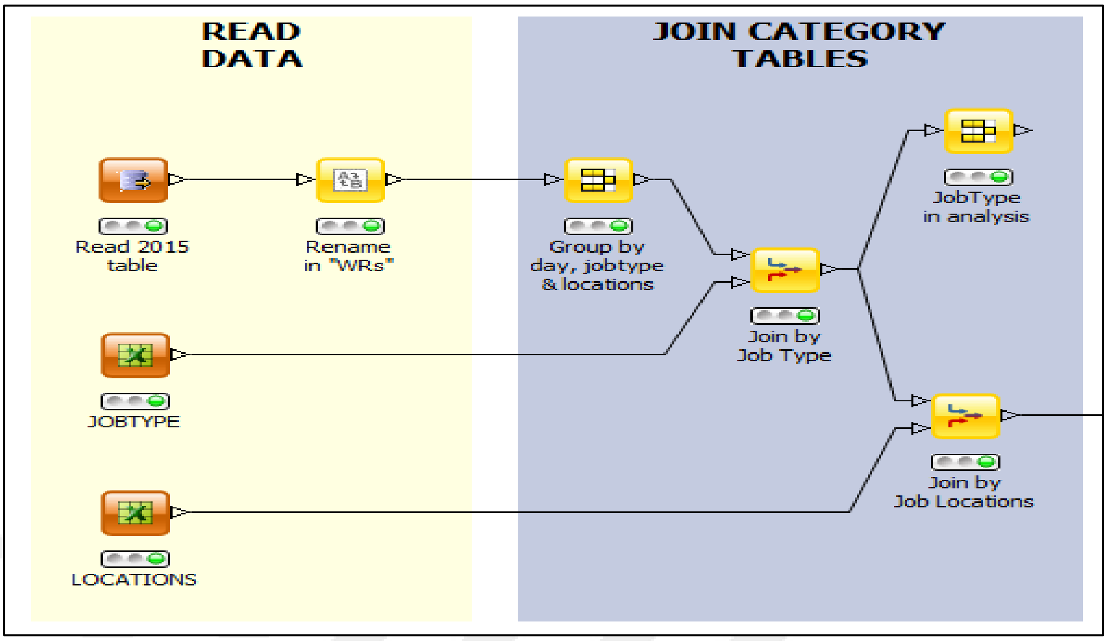


Figure 3: Data reading and Joining

2.3 Data Enrichment

This section is consisted data enrichment. Basically in data enrichment part is in order to merge the main database and grouped the data according to summarization and two data table and, filter the data due to pre-defined condition and extraction of missing values.

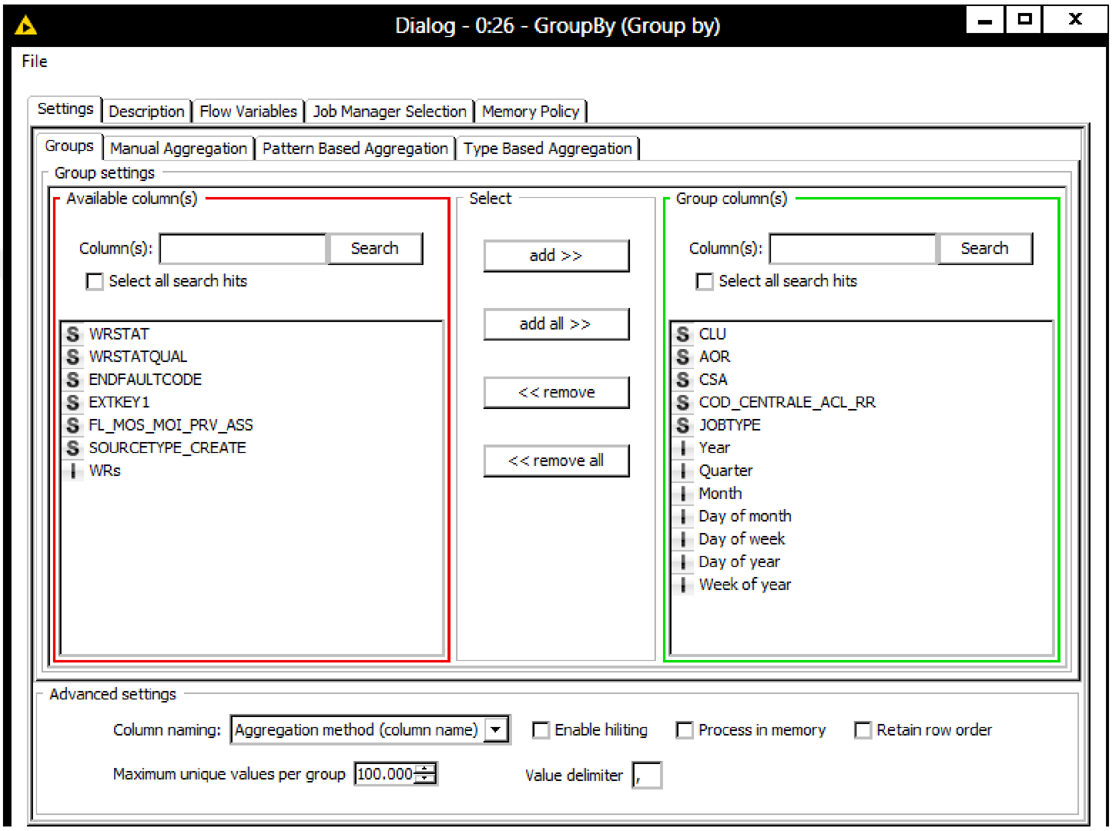
After Data Gathering, we had a database which had 14 million rows and 24 columns another word 24 different unique variables and two data tables which one of them, Job Type had 1581 rows and columns (variables) and the other one Location had 10972 rows and 12 columns (variables). These database and data tables needed to be merged with using common variables.

Before joining and concatenating process Work Request database needed to be summarised grouped. Therefore "Groupby" node of KNIME was used in order to manage regarding process. Basically it is used for grouping the rows of a table by the unique values in the selected columns. A row is created for each unique value group of the selected column(s). The remaining rows are aggregated by the defined method. The output table therefore contains one row for each existing value combination of the selected group column(s)[46]

[46] KNIME, "Knime GroupBy Node Description"

In “groupby” node, firstly WRID variables were counted per day and added a new variables as WRs which referred to daily total work request per location and per job type. Afterwards seven of variables were taken out due to aim of research (Figure 4). Hereby after “groupby” node processing 12.8 million of rows were obtained instead of 14 million rows. (Figure 4).

KNIME



Column(s):

□ Select all search hits

Available column(s): WRSTAT, WRSTATQUAL, ENDFAULTCODE, EXTKEY1, FL_MOS_MOI_PRV_ASS, SOURCETYPE_CREATE, WRs

Column(s):

□ Select all search hits

Group column(s): CLU, AOR, CSA, COD_CENTRALE_ACL_RR, JOBTYPE, Year, Quarter, Month, Day of month, Day of week, Day of year, Week of year

Column naming: Aggregation method (column name)

□ Enable hiliting □ Process in memory □ Retain row order

Maximum unique values per group 100.000

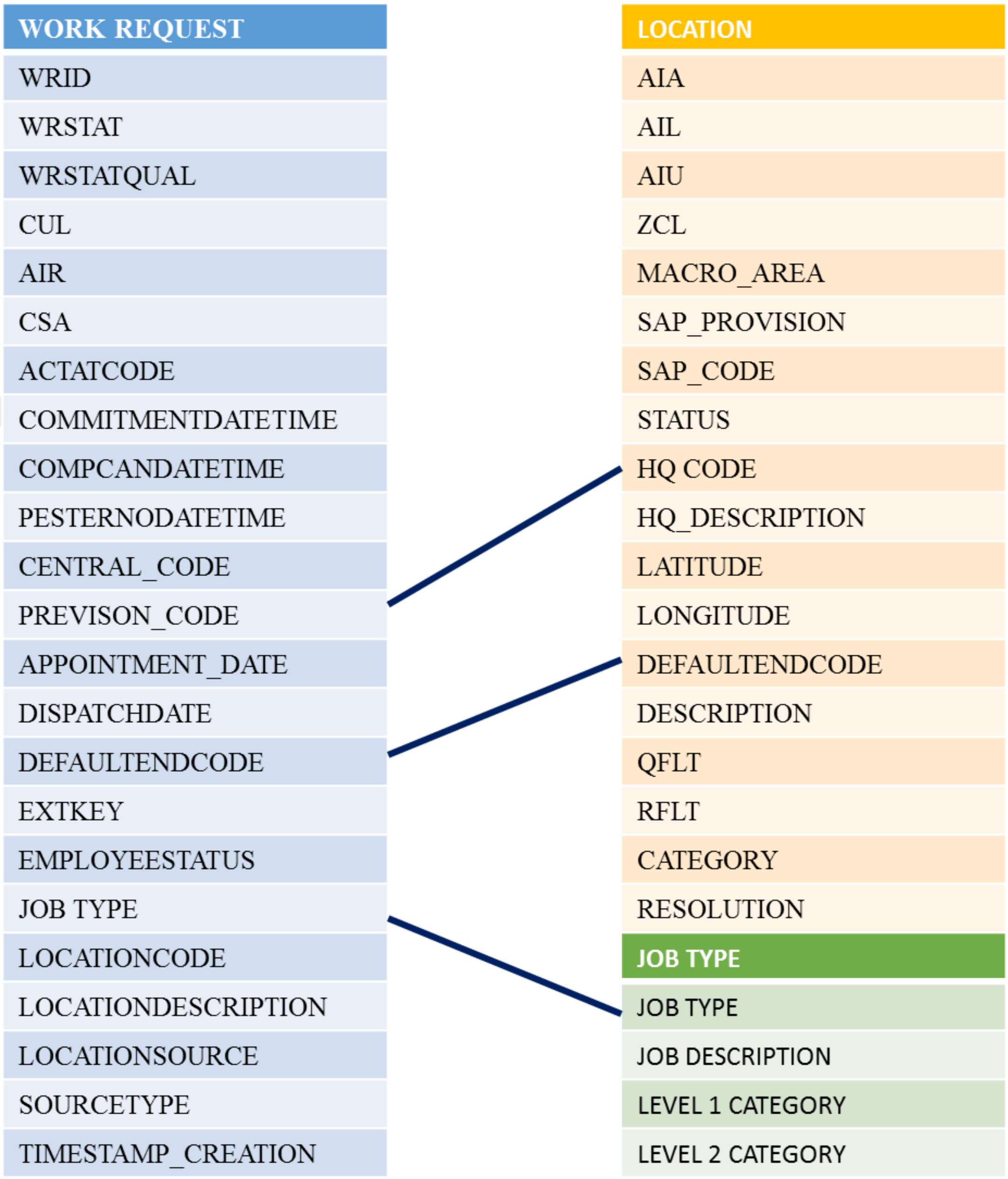
Value delimiter ,

Figure 4: Data Enriching and Grouping

As seen it the Figure 2 and 3, these database and data tables were merged with using common variables. In first merging process Work Request database and Location Data Table were joined with using common two variables PROVISION CODE – HQ CODE which are actually same variable but names are different and FAULTENDCODE- FAULTENDCODE. Therefore the new dataset was obtained with 9.8 million of rows instead of 12.8 million rows. As it is seen on Figure 4 those datasets are merged with join function of KNIME.

https://www.knime.org/files/nodedetails/_manipulation_row_row_transform_GroupBy.html,, (9 May 2016)

Table 9: Data Tables before Merging



WORK REQUEST
WRID
WRSTAT
WRSTATQUAL
CUL
AIR
CSA
ACTATCODE
COMMITMENTDATETIME
COMPCANDATETIME
PESTERNODATETIME
CENTRAL_CODE
PREVISON_CODE
APPOINTMENT_DATE
DISPATCHDATE
DEFAULTENDCODE
EXTKEY
EMPLOYEESTATUS
JOB TYPE
LOCATIONCODE
LOCATIONDESCRIPTION
LOCATIONSOURCE
SOURCETYPE
TIMESTAMP_CREATION

LOCATION
AIA
AIL
AIU
ZCL
MACRO_AREA
SAP_PROVISION
SAP_CODE
STATUS
HQ CODE
HQ_DESCRIPTION
LATITUDE
LONGITUDE
DEFAULTENDCODE
DESCRIPTION
QFLT
RFLT
CATEGORY
RESOLUTION

JOB TYPE
JOB TYPE
JOB DESCRIPTION
LEVEL 1 CATEGORY
LEVEL 2 CATEGORY

Secondly, after first joining processing, the dataset needed to be enriched with job type data table. Thus using one common variables which is JOB TYPE. After this process, number of data rows were not changed and it remained 9.7 million.

Finally dataset included a plenty of missing values, in order to avoid missing value based problem, the dataset had to be filtered out the missing values. Thus for this operation "Rule Based row splitter was used. Basically, node of Rule Based Row Filter takes a list of user-defined rules and tries to match them to each row in the input table. If the first matching rule has a TRUE outcome, the row will be selected for inclusion. Otherwise (i.e. if the first matching rule yields FALSE) it will be excluded. If no rule matches the row will be excluded. Inclusion and exclusion may be inverted.

In consequence of this operation, all the missing values were cleaned and 821280 of rows were obtained instead of 9.7 million (Figure 5).

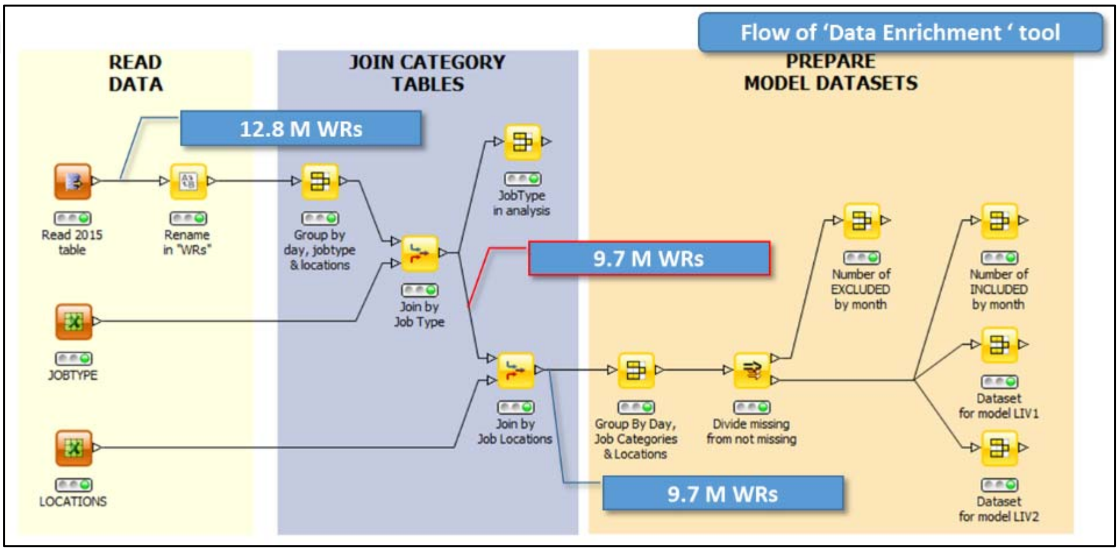


Figure 5: All Phases of Data Preparation

2.4 Data Feature Engineering

In this research we worked with daily work load per location and work type and time data which were year, month of year, week of year, day of year, day of month and day of week. Those data were enough for stochastic forecasting analysis which were Auto-regression (AR) and Auto-regressive Integrated Moving Averages (ARIMA- Box-Jenkins). However, in to be seen in analysis part predictor error of stochastic forecasting model were quite high. Therefore in this research new forecasting models were needed to select as machine learning models. Hereby, new variables were needed to create for reducing the prediction error rate. In this case data feature engineering endorsed the research. Data feature engineering; is the process of using domain knowledge of the data to create features that make machine learning algorithms work. Data feature engineering is fundamental to the application of machine learning, and is both difficult and expensive. The need for manual feature engineering can be obviated by automated feature learning [47].

In this research data feature engineering could be seen two part. Inner feature creation and outsource feature creation. In inner feature creation is to create new variables with using current data as daily number of workloads, time data etc. Outsource feature creation is to gather data from outside which relevant to current data as weather data (Daily; average, minimum and maximum temperature, mm of rainfall, humidity, etc.).

The idea behind this section comes from a forecasting competition of a web site which about statistics and machine learning, *www.kaggle.com*. Kaggle was founded as a platform for predictive modelling and analytics competitions on which companies and researchers post their data and statisticians and data miners from all over the world compete to produce the best models. This crowd sourcing approach relies on the fact that there are countless strategies that can be applied to any predictive modelling task and it is impossible to know at the outset which technique or analyst will be most effective. Kaggle also hosts recruiting competitions in which data scientists compete for a chance to interview at leading data science companies like Facebook, Winton Capital, and Walmart[48].

The regarding competition is Rossmann Store Sales competition [49]. In this competition, a competent applied the competition with a data feature engineering method [50]. Therefore in our research we followed same steps to create new features.

2.4.1 Inner Data Creation

In this section is about to create new variables whit using current data. This section is divided by two part, data are created with using R code and data are created with using Knime Modules.

[47] Wikipedia, "Data Feature Engineering Description", https://en.wikipedia.org/wiki/Feature_engineering , (9 May 2016)

[48] Wikipedia, "Kaggle.com Description" , https://en.wikipedia.org/wiki/Kaggle , (9 May 2016)

[49] Kaggle, "Rossmann Store Sales competition", https://www.kaggle.com/c/rossmann-store-sales, (10 May 2016)

[50] Kaggle, "Winning Model Documentation describing my solution for the Kaggle competition Rossmann Store Sales" : https://kaggle2.blob.core.windows.net/forum-message-attachments/102102/3454/Rossmann_nr1_doc.pdf?sv=2012-02-12&se=2016-05-17T16%3A55%3A37Z&sr=b&sp=r&sig=tU5oqVEQGyjP97%2FLupW57DIfRQu2GxdHFvOd0ewZggs%3D, (11 May 2016)

2.4.1.1 Data Creating with using R code

In this part, data are created with using R snipped node[51] .Those data are *holidays of 2014,2015 and 2016 days since last holiday, days since next holiday, holiday weeks, sum of last 30 days, sum and median of 7 days before on residuals*.

2.4.1.1.1 Holidays of 2014, 2015 and 2016 feature

This feature is about all official holidays of Italy in 2014, 2015 and 2016, basically the formula is if any day of 2014, 2015 and 2016 has official holiday, its value is 1, otherwise 0. It was created with using R. Its pseudo-code is:

Table 10: Pseudo-code of 1st feature

```
Read required data using knime.in (knime to R code module name)
1.1. Create holiday data frame
1.2. FOR date counter=1 to number of rows in holiday data frame
1.3. IF date of the month is a holiday by comparing with calendar
    1.3.1.  Assign value='1' to the date
1.4. ELSE
    1.4.1.  Assign value='0' to the date
1.5. ENDIF
1.6. ENDFOR
```

2.4.1.1.2 Days since Last Holiday Feature

This feature is about how many days past from previous official Italian holiday. It was created with using R. The regarding pseudo-code is:

Table 11: Pseudo-code of 2nd t feature

```
1.  Convert holiday data set into time series data
2.  Get the dates of the events
3.  Make an index of the events since last holiday
```

[51] KNIME, "R snipped Knime Description", https://www.knime.org/files/nodedetails/_labs_r_interactive_R_Snippet.html ,(11 May 2016)

2.4.1.1.3 Holiday Weeks Feature

This feature actually creates three variables. Those are “Number of Holidays in This Week”, “Number of Holidays in Last Week” and “Number of holidays in Next Week”. Those variables are about, how many holiday is current, last and next week. It was created with using R. The regarding pseudo-code is:

Table 12: Pseudo-code of 3rd feature

```
1. Aggregate holiday data set and KPI
2. Associate/assign the number of holidays for each week of the year
3. Combine holiday and holiday week data
```

2.4.1.1.4 Sum of Last Month

This feature calculates total number of Work Request in previous 30 days. It was created with using R. The regarding pseudo-code is:

Table 13: Pseudo-code of 4th t feature

```
1. FOR date index =1 to number of rows of data output
        1.1 IF date index<30
                1.1.1 Set residuals =NA
        1.2 ELSEIF date index>=30
                1.2.1 Set residuals =date index -30
        1.3 ENDIF
2. ENDFOR
```

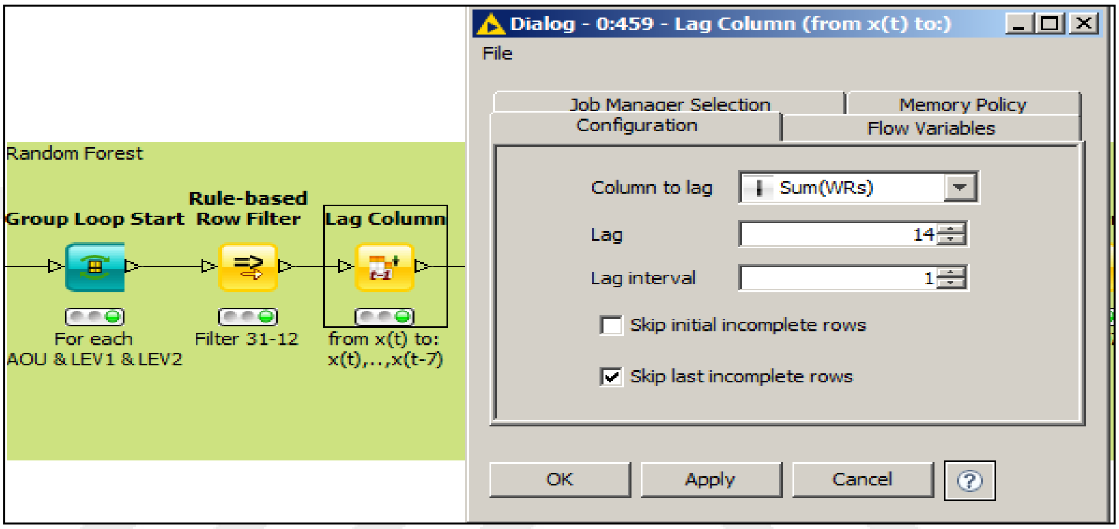
2.4.2.2 Data Creation with Knime

In this part the data which are created, 14 lags of Daily total work request, 3 lags of holidays data, Sum of work requests of Last quarter.

2.4.2.2.1 Lagged value of Daily Total Work Request

Lag model is model for time series data in which a regression equation is used to predict current values of a dependent variable based on both the current values of an explanatory variable and the lagged (past period) values of this explanatory variable (52)(53)

As it is seen on Figure 6, we chose 14 unit of lag due to our aim which explained in introduction part. In Knime lagged value is obtained with using “Lag Column” node (54).



Column to lag: Sum(WRs)

Lag: 14

Lag interval: 1

□ Skip initial incomplete rows

☑ Skip last incomplete rows

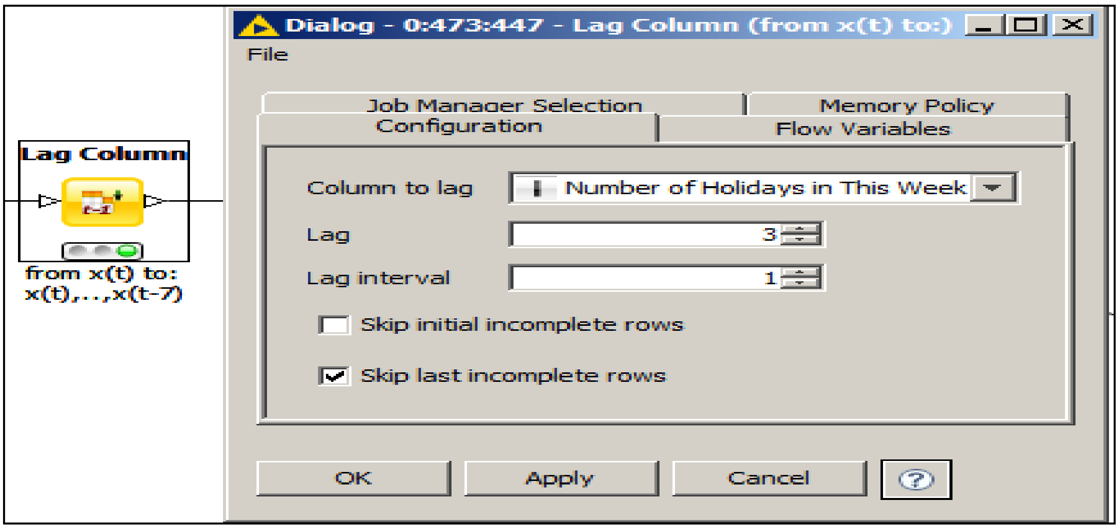
Figure 6: Lag Creation

2.4.2.2.2 Lag of Holidays

[52] Cromwell, Jeff B.; et al. . .. "Multivariate Tests For Time Series Models". SAGE Publications. ISBN

0-

8039-5440-9, 1994
[53] Judge G G, Griffiths W. E, Hill R. C; Lee T. C , "The Theory and Practice of Econometrics" New York: Wiley. pp. 637–660. ISBN 0-471-05938-2, 1980
[54] KNIME, "Lag Column Description", https://www.knime.org/files/nodedetails/_manipulation_column_column_transform_Lag_Column.html (12 May 2016)



Column to lag: Number of Holidays in This Week

Lag: 3

Lag interval: 1

□ Skip initial incomplete rows

☑ Skip last incomplete rows

Figure 7: Lag of holidays

It is created to observe the possible effect of holidays before, the idea came from Rossmann Competition [(53)]. The creation is seen on Figure 7.

2.4.3 Outsource Data Gathering and Using

In this section historical weather data of 2014, 2015 and 2016 were gathered from the web site of world weather online. The weather data were gathered per every AIU. The weather data included daily historical variables PercipitionMM rainfall and its 7 days lagged values.

Several previous research pointed importance of using weather data. In order to temperature data: Increases in temperature and higher frequency, duration, and intensity of heat waves create an additional burden on keeping equipment cool in exchanges and base stations, resulting in increased failure rates[(55)]. For precipitation: Increased precipitation (rain or snow) leads to a higher risk of flooding low-lying and underground infrastructure and facilities, as well as erosion or flood damage to transport structures, potentially exposing cables[(56)].

[55] Horrocks et al., 2010.ClimAID, 2011
[56] Horrocks et al.,2010; Ofcom, 2010

Detailed information about feature selection and re-ordering are placed on analysis part. We use also lagged value of WR as a variables, detailed information about lagged variables are placed in Data Processing and Results chapter.

Before the data processing all the data are shown Table 13:

Table 14: All Data before Data Processing

Data
Year
Week of year
LEVEL 1 CATEGORY
LEVEL 2 CATEGORY
AIU
Date and time
Quarter
Month
Day of month
Day of week
Day of year
Sum(WRs)
Sum(WRs)(-1)
Sum(WRs)(-2)
Sum(WRs)(-3)
Sum(WRs)(-4)
Sum(WRs)(-5)
Sum(WRs)(-6)
Sum(WRs)(-7)
Sum(WRs)(-8)
Sum(WRs)(-9)
Sum(WRs)(-10)
Sum(WRs)(-11)
Sum(WRs)(-12)
Sum(WRs)(-13)
Sum(WRs)(-14)
Festivity
Festivity(-1)
Festivity(-2)
Festivity(-3)
Days Since Last Holiday
SumLastMonth(WRs)
Sum(WRs) Last Quarter

Number of Holidays in This Week
Number of Holidays in Last Week
Number of Holidays in Next Week
PercipitionMM
PercipitionMM-1
PercipitionMM-2
PercipitionMM-3
PercipitionMM-4
PercipitionMM-5
PercipitionMM-6
PercipitionMM-7
Legend:
Current Data
Weather Data
Knime
R Snippet

CHAPTER THREE

DATA ANALYSIS

In this chapter we give detailed information about the distribution of our data with time stamp, the several methods which were chosen for our dataset in order to forecast better performance than The Company's current methodology, and The Company's current forecasting methodology.

Before process the data we would like to show our data in graph to understand if any business cycling, trends and/or seasonal effects exist or not.

3.1 Data Visualisation and Statistical Results

In this section we give brief information about visualisation, and advanced statistical analysis of our data. In visualisation part we put 2014-2015 data due to this amount of data are used for training as we mentioned in introduction part.

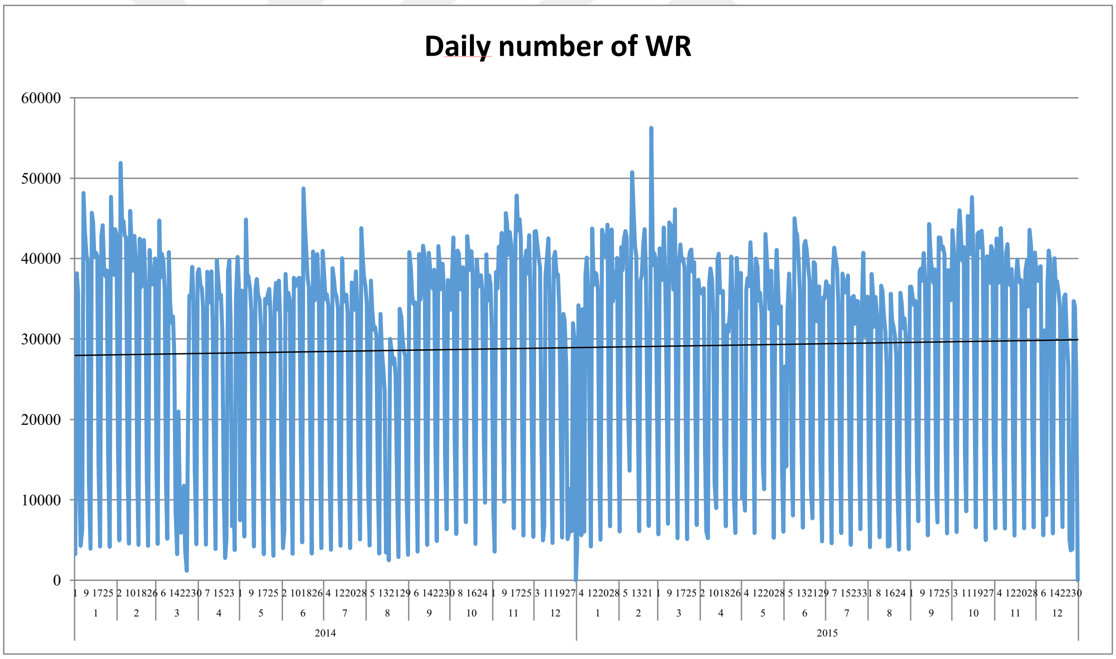


Figure 8: Graph of Daily count of total work request ın Italy

3.1.1 Data Visualisation All levels

In Figure 8 the black line shows us the mean of work request which depends on time. As it is seen of this line our data is increasing by time. Our data have several peaks as beginning of February 2014, June 2014, at the end of February 2015 and October 2015.

We can see in our data that several points are in the time work requests got down. Main reasons seems the Italian Holiday which covers many days as Christmas, Easter, so on. Also as it is seen on the graph the work request has decreasing trend on August. Its main reason is to be August is main summer vacation time in for Italy.

3.1.1.1 Data Visualisation Job Type Level I

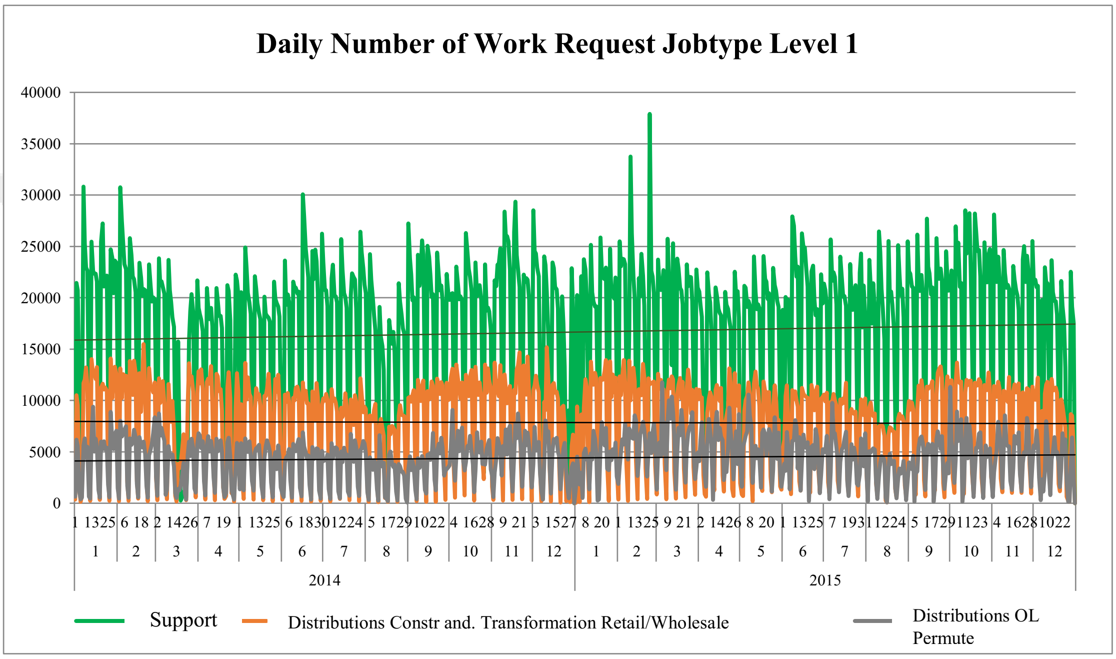


Figure 9: Graph of Daily total count of work request Job Type Level 1

In Figure 9 we see our data in divided into three sub group by Job Type Level 1. Those sub groups are Support, Distributions OL construction and transformations Retail / Wholesale (With customer premises intervention), (Abb. Distributions OL Con)and Distributions OL Permute.

As we observe that the main work request is seen on Support Job Type, as a basic view its trend line is increasing. However Distributions OL Con and Distributions OL Permute are on stable trend. Another word the relationship between three types of Level 1 work request is lower.

3.1.1.2 Data Visualisation Job Type Level II

As we mentioned in Data part Job Type Level II is detailed edition of Job Type Level I. Each level I element has their own sub groups. For example Support owns six type II level sub groups, Distributions OL Con has five type II level sub groups and Distributions OL Permute has two type II level sub groups.

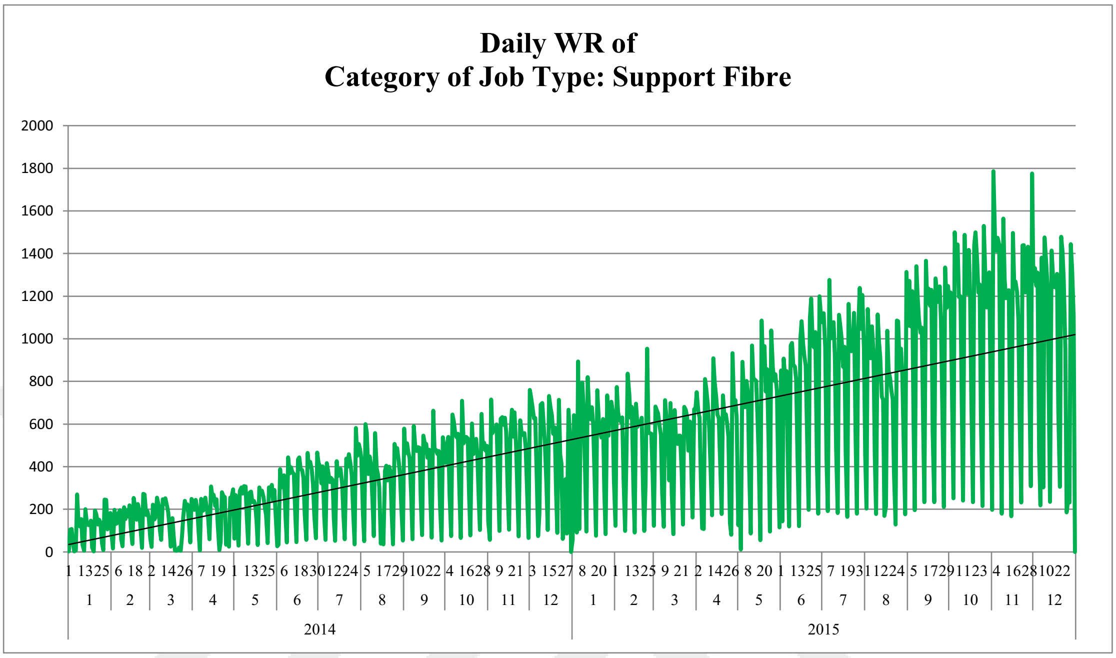


Figure 10: Graph of Daily total count of work request Support Fibre

As it is seen on Figure 10 the Trend of Support Fibre is increasing it shows us Support activity is getting increases day by day.

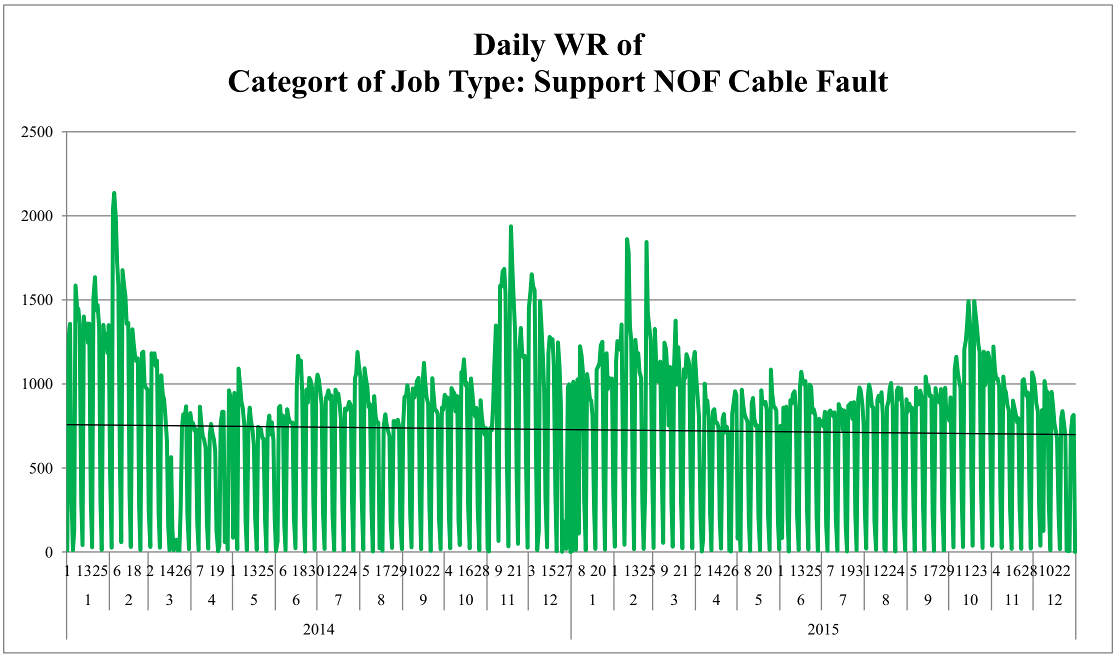


Figure 11: Graph of Daily total count of work request Support NOF Cable Fault

On the graph 3.4 we see the trend of Support NOF Cable Fault is slightly decreasing

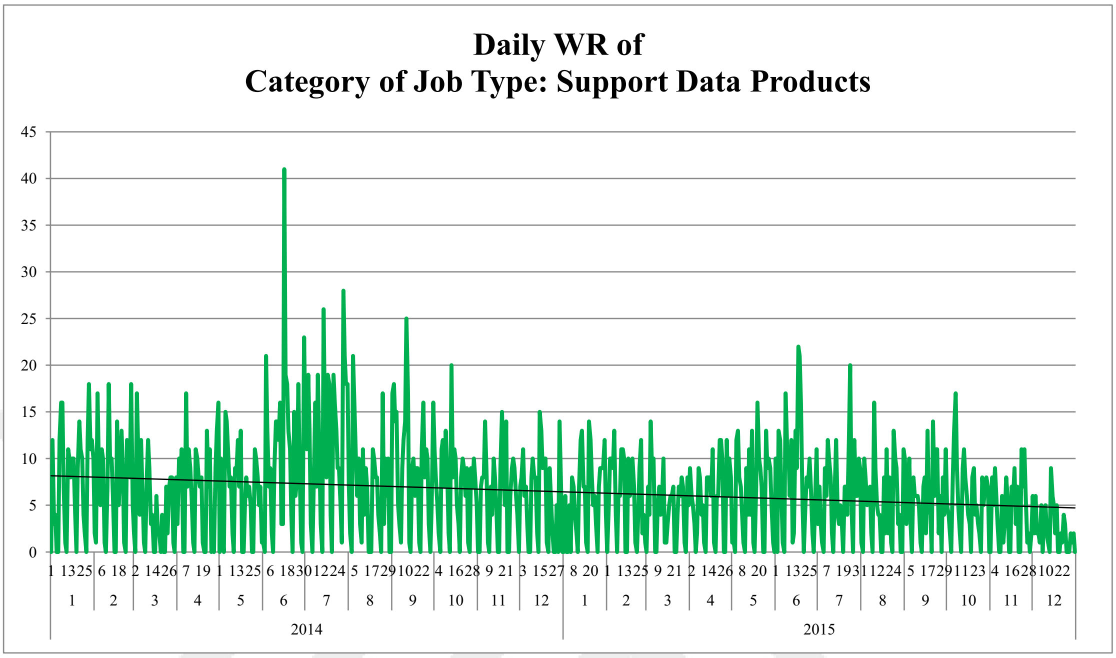


Figure 12: Graph of Daily total count of work request Support Data Products

Figure 12 refers us the trend of Support Data Products is decreasing.

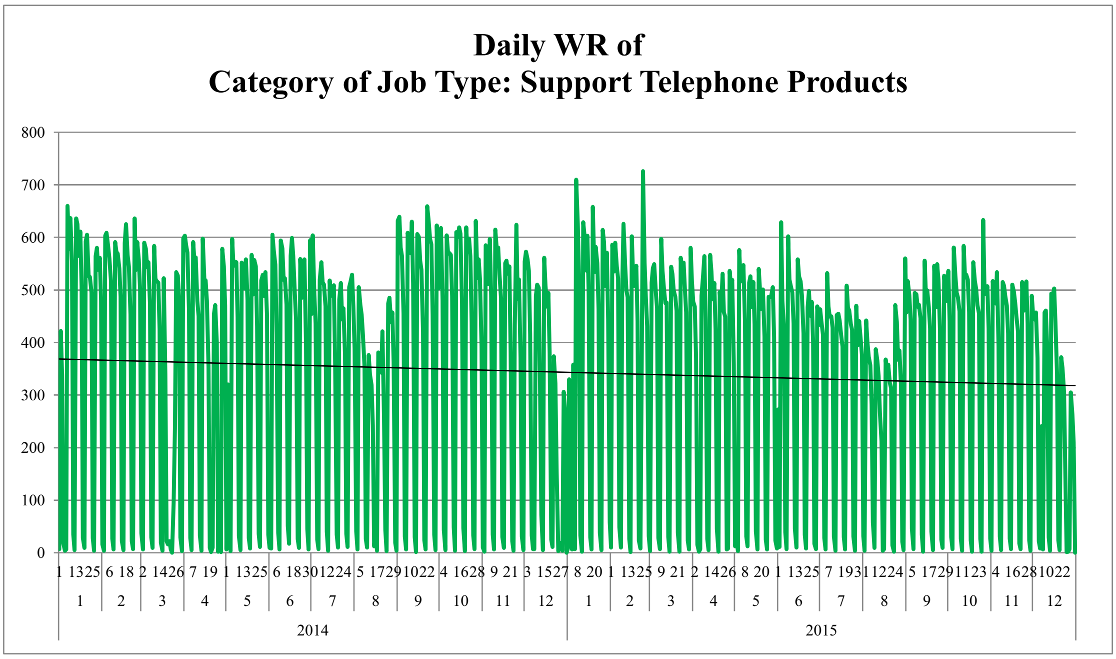


Figure 13: Graph of Daily total count of work request Support Telephone Products

The trend of Support Telephone Products is decreasing (Figure 13) as Support Data Products.

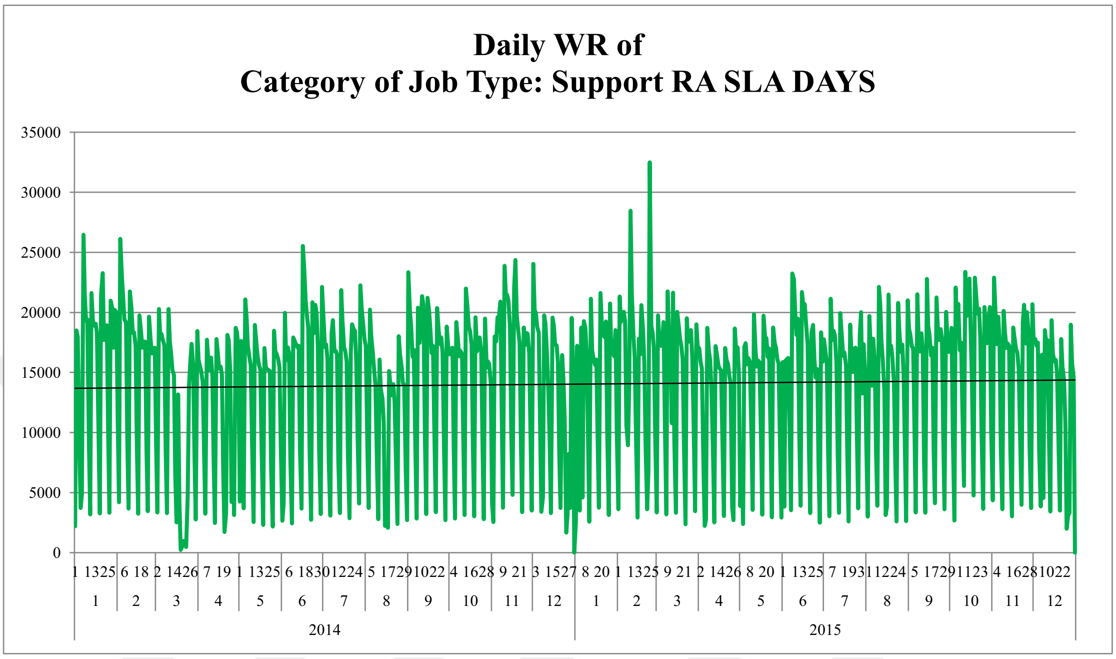


Figure 14: Graph of Daily total count of work request Support RA SLA DAY

The Figure 14 shows us trend of Support RA SLA DAY is slightly increasing.

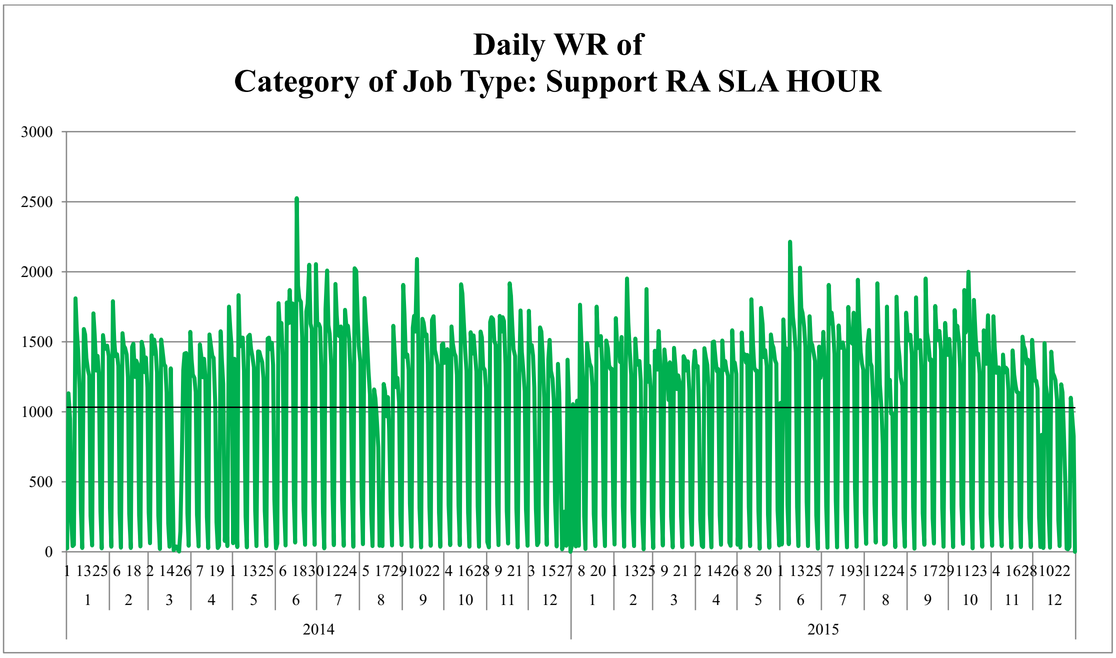


Figure 15: Graph of Daily total count of work request Support RA SLA HOUR

As we see on the Figure 15, the trend of Job Type Level II Support RA SLA HOUR is stable.

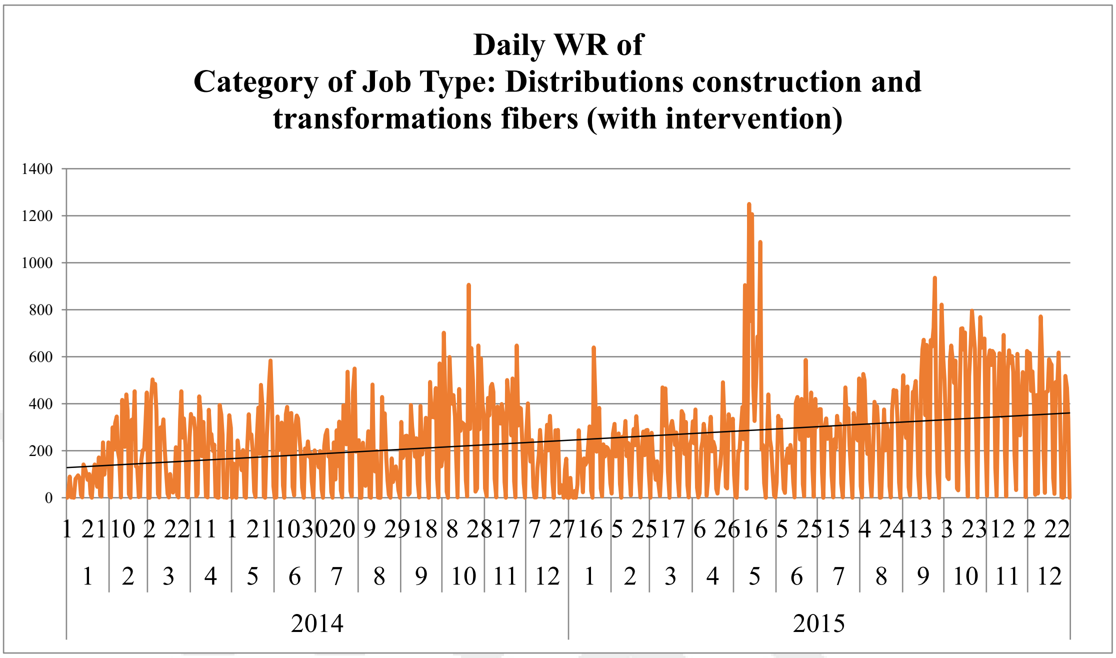


Figure 16 Graph of Daily total count of work request Distributions construction and transformations fibres (with intervention)

In Figure 16 Distributions construction and transformations fibres (with intervention) has quite increasing trend.

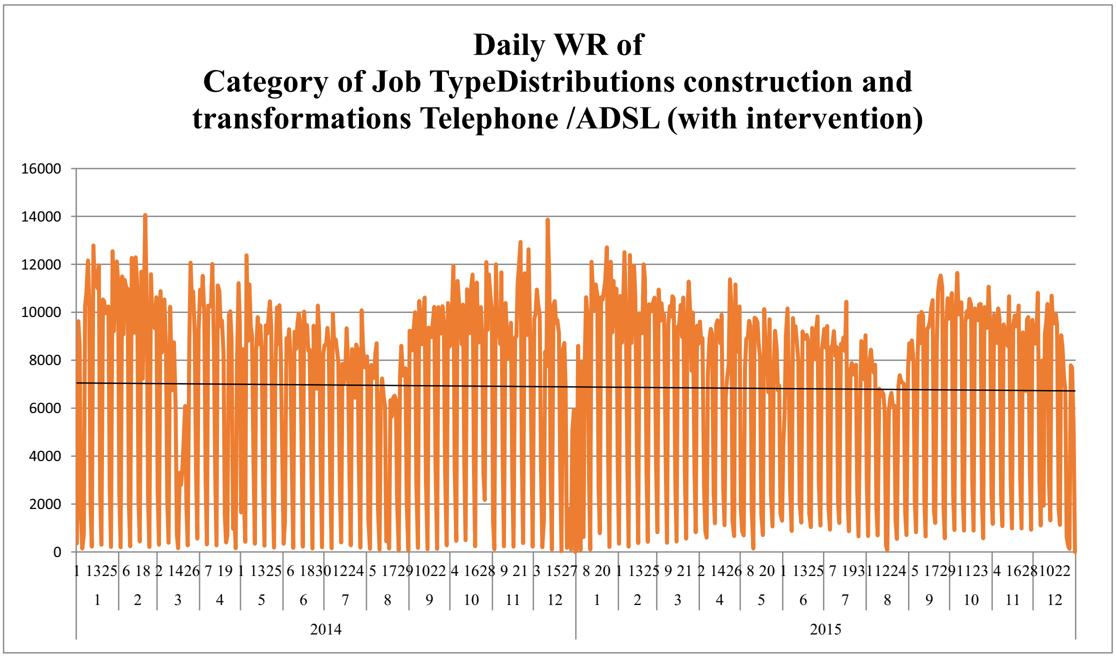


Figure 17: Graph of Daily total count of work request Distributions construction and transformations Telephone /ADSL (with intervention)

According to Figure 17 Distributions construction and transformations Telephone /ADSL (with intervention) is slightly decreasing.

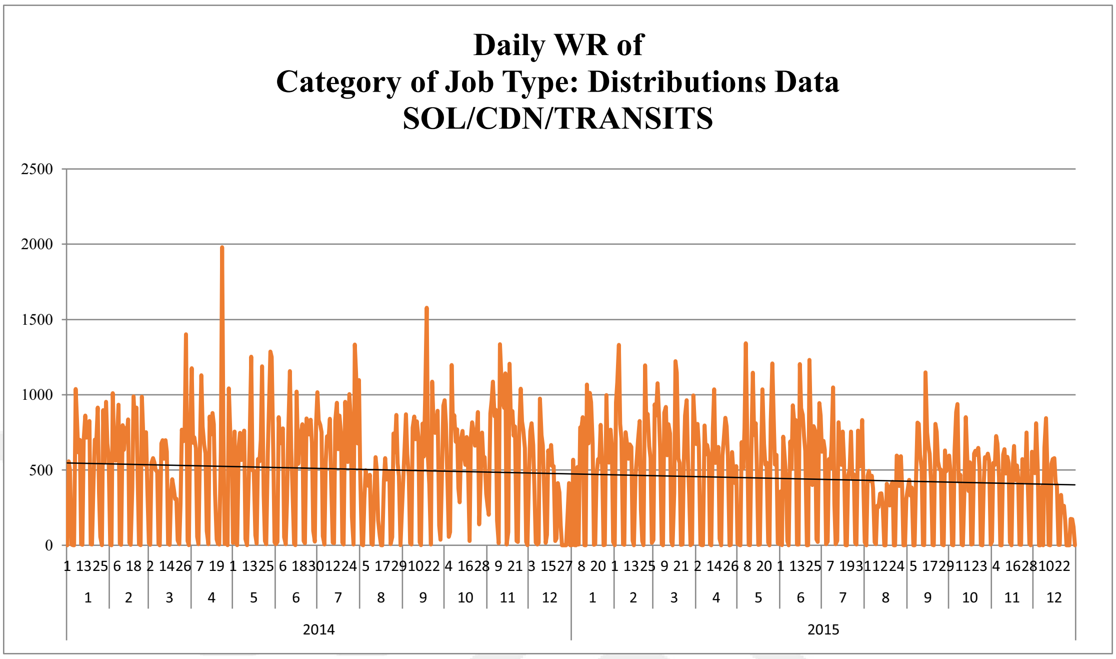


Figure 18: Graph of Daily total count of work request Distributions Data SOL/CDN/TRANSITS

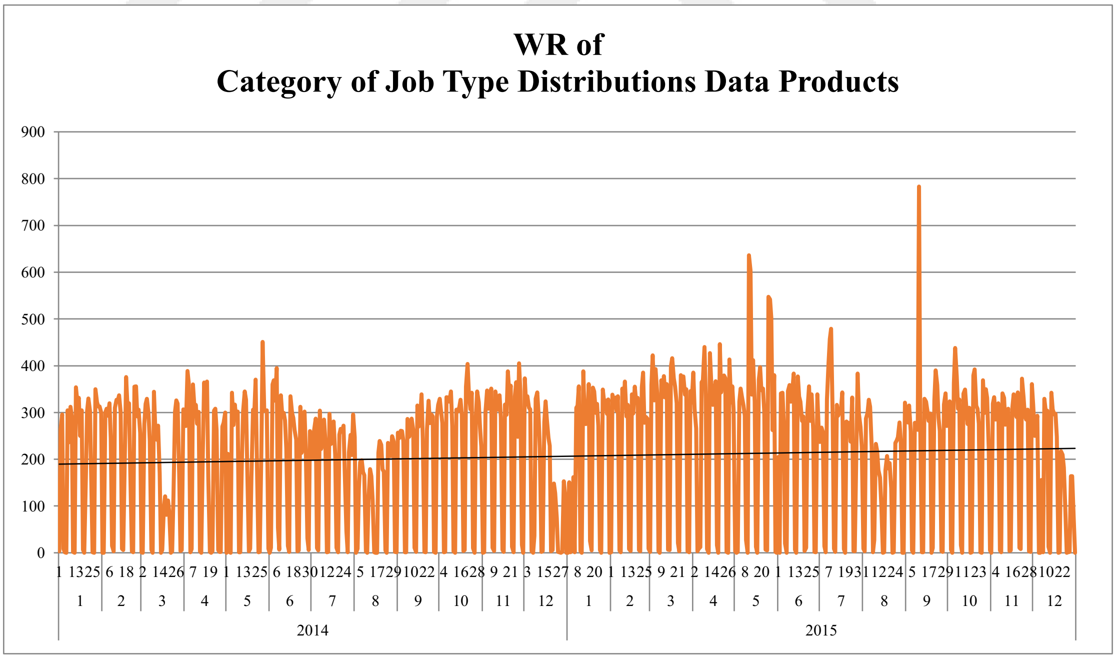


Figure 19: Graph of Daily total count of work request Distributions Data Products

Trend of Distributions Data SOL/CDN/TRANSITS job type (Figure 18) is decreasing. According to Figure 19 trend of Distributions Data Products is slightly increasing.

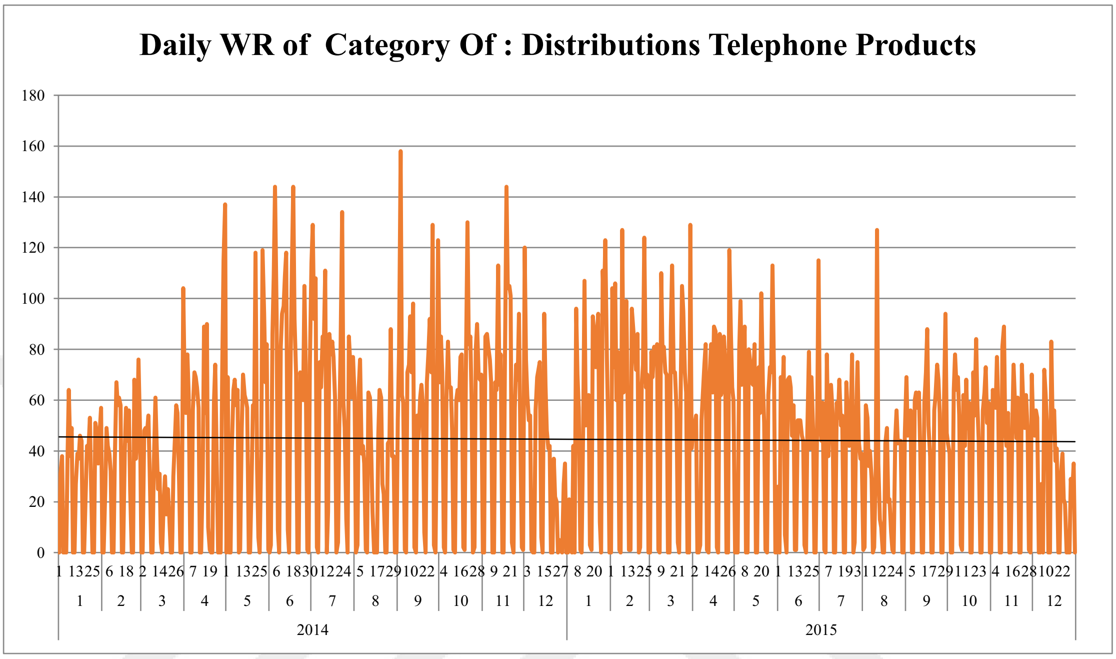


Figure 20 Graph of Distributions Telephone Products

In Figure 20 trend of Distributions Telephone Products is quite stable.

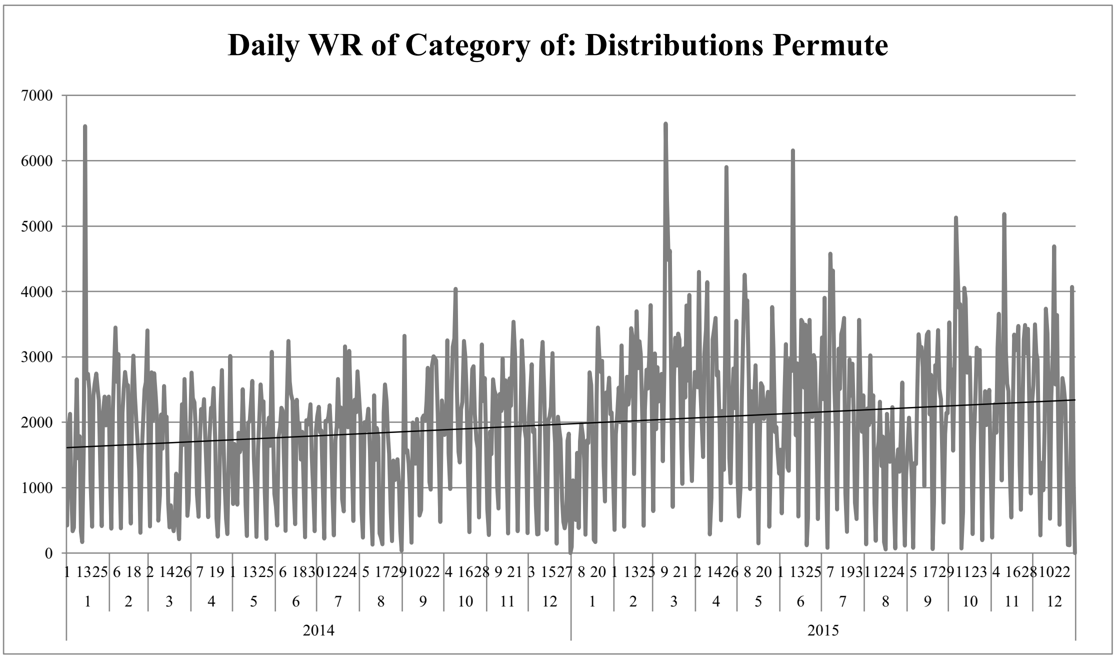


Figure 21: Graph of Distributions Permute

As it is seen on Figure 21 trend of Distributions Permute increases.

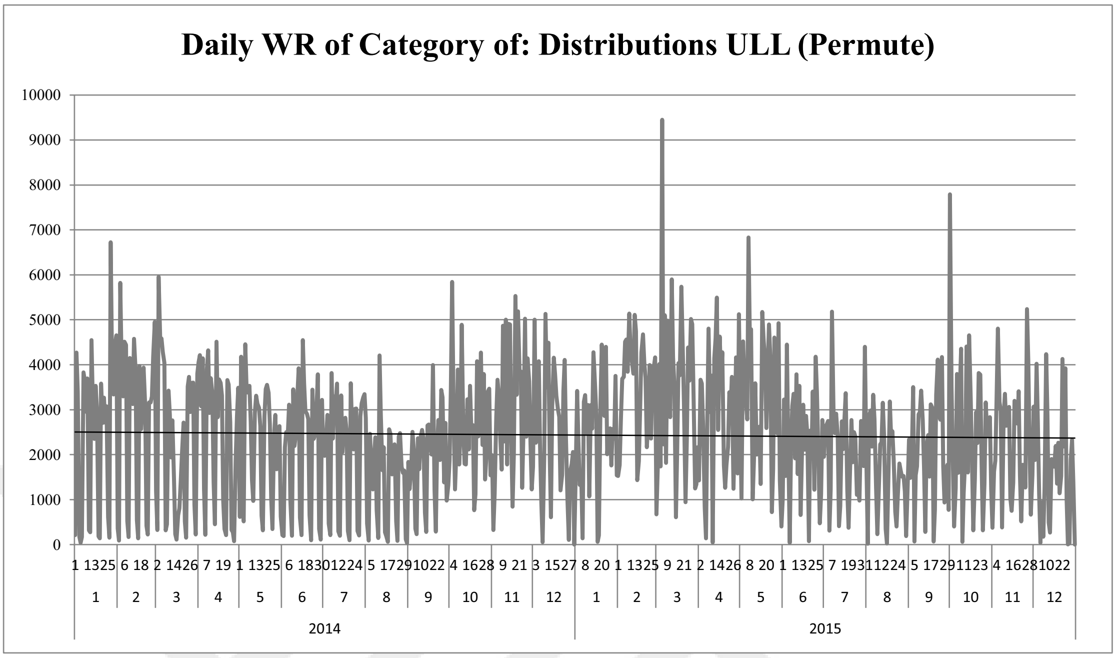


Figure 22 Graph of Distributions ULL (Permute)

As it is seen on Figure 22 trend of Distributions ULL Permute is slightly decreasing.

3.2 Statistical Analysis of Our Dataset

In this section we give a brief statistical information of our dataset by all job types and days of week. We use Knime for demonstrate this statistical information. In statistical information we mentioned median of all series by job types and days of week. Interquartile Range (IQR) and their outliers, and demonstrated this information with graphs.

3.2.1 Statistical Analysis of Job Types All Level and All Levels by Days of Week

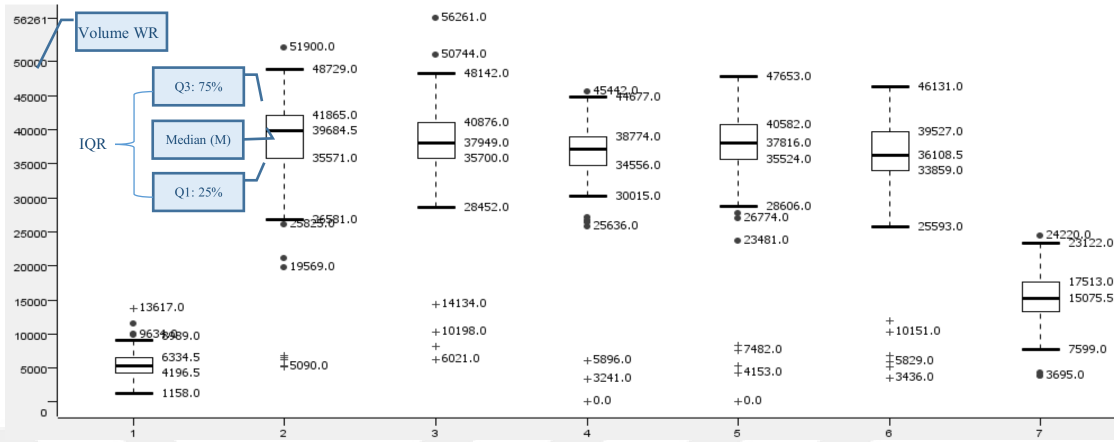


Figure 23: Graph of Statistical analysis of job Types all level by day of week

In Figure 23 demonstrated us day of week distributions of work request in all job types. Axis of this graph is refer to day of a week, for instance 1 is Sunday, 2 is Monday, 3 is Tuesday, 4 is Wednesday, 5 is Thursday, 6 is Friday and 7 is Saturday.

Also we observe thanks to Figure 23 that the highest count of anomaly (outliers) is seen on Tuesday and the lowest count of anomaly is seen on Sunday. The reason behind this anomaly is to be Sunday is absolute holiday therefore count of work request is observed less than the other and the chance that be seen anomaly is lower too.

3.2.2 Statistical Analysis of All Level Job Type and AIL Level Location by Days of Week

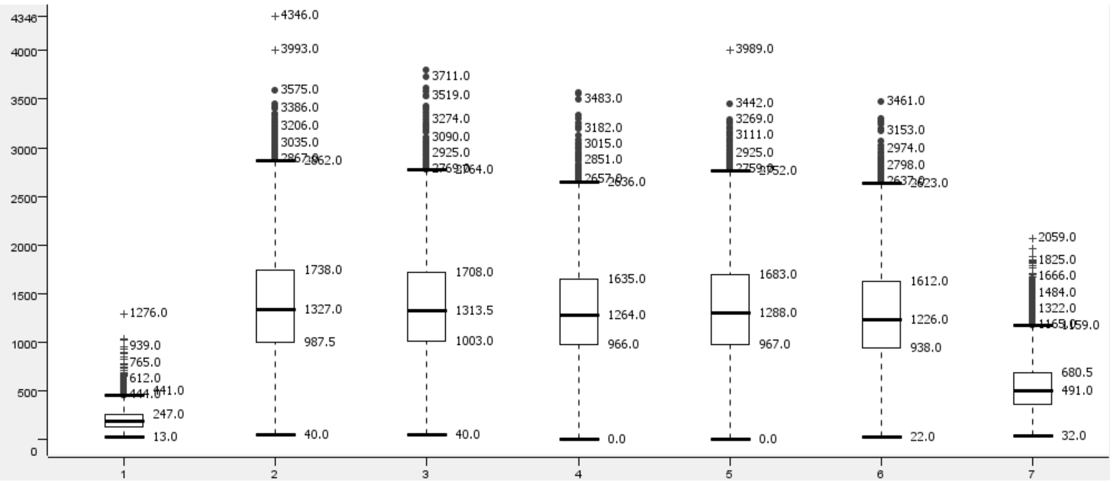


Figure 24: Graph of AIL Location and all Job type level WR by days of week

As it mentioned on Data Chapter AIL is Mid-level locations of Italy, according to graph 24 all days of week owns upper outliers.

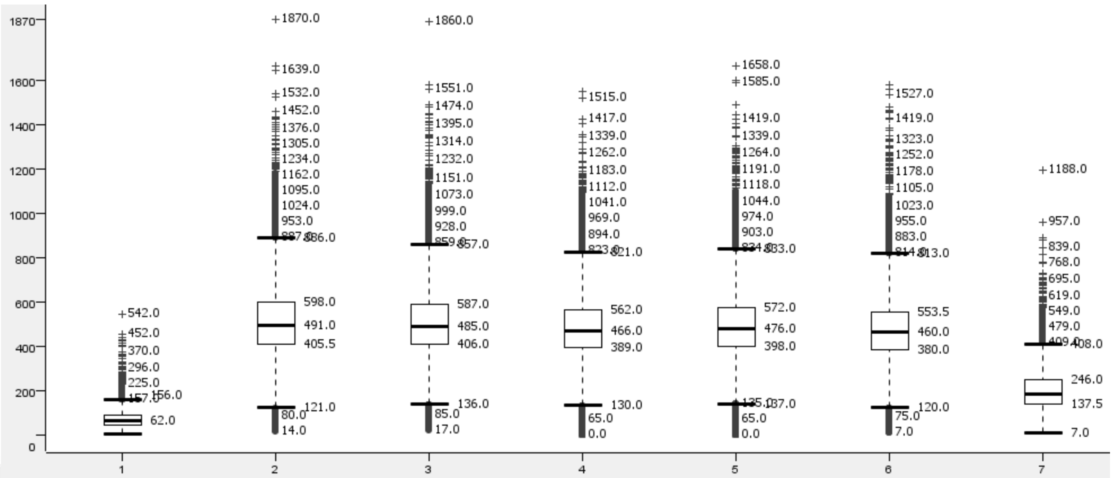


Figure 25: Graph of AIU location and all job type level WR by days of week

As we mentioned in Data Chapter AIU is detailed location part of our dataset and our main location level. According to Figure 25 there are plenty of outliers. Its reason is work request loads are not shared equally to all AIU levels in Italy.

3.2.3 Statistical Analysis of Job Type Level I by Days of Week

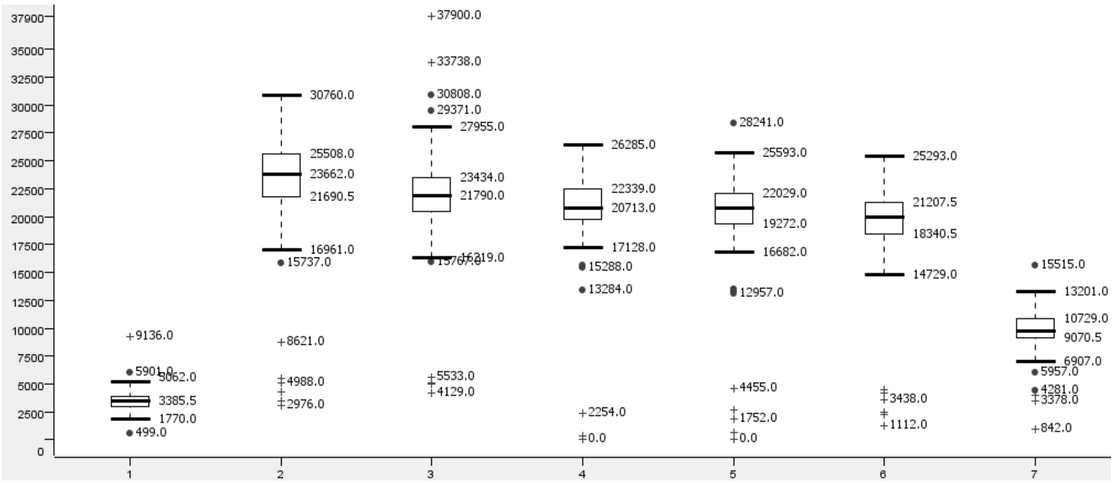


Figure 26: Graph of Statistical analysis of job type level I: Support

According to Figure 26 Tuesday owned the higher count of upper outliers. However, Monday owned highest count of lower outliers. Another point Saturday has also a plenty of count of lower outliers. It shows us Support level I has irregular data distribution by days of week.

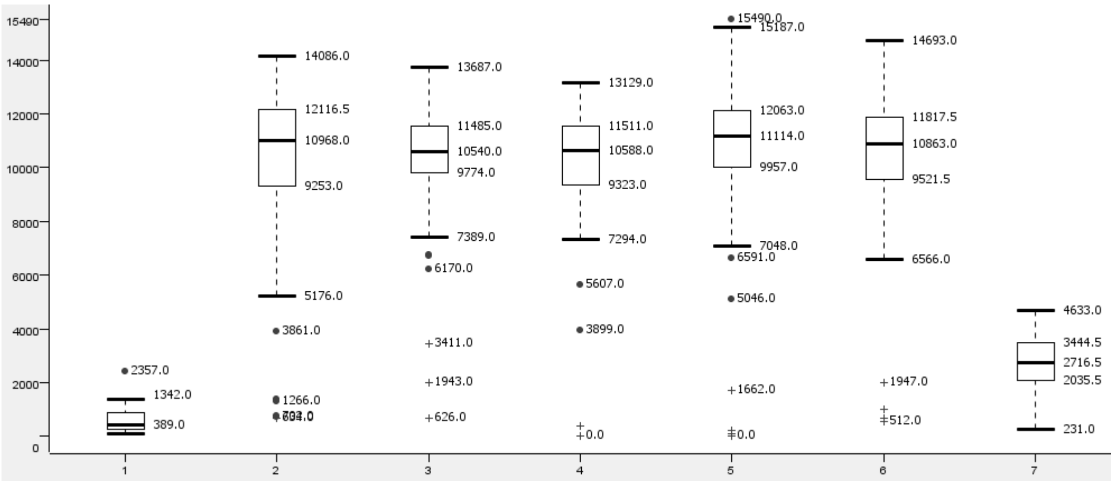


Figure 27 Graph of Statistical analysis of job type level I: Distributions OL construction and transformations Retail / Wholesale (With customer premises intervention)

According to Figure 27 in Distributions OL construction and transformations Retail / Wholesale (With customer premises intervention) level I job type Sunday and Thursday own upper outliers. However, all week days own lower outliers.

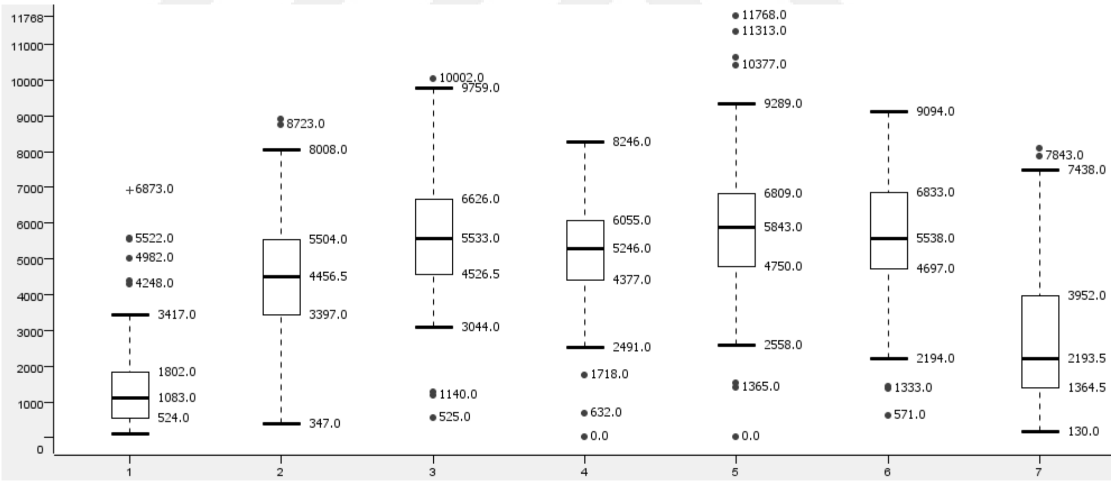


Figure 28 Graph of Statistical analysis of job type level I: Distributions OL: Permute

In Figure 28 the work request distributions of Distributions OL: Permute seems similar to Figure 27, in spite of range of the data. Sunday and Tuesday own higher number of upper outliers and higher number of lower outliers are observed on Wednesday.

3.3 Time Series Data Analysis

In this section, methodologies which are mentioned in Theoretical Background are applied on our dataset and The Company's current methodology. As it mentioned in Introduction Chapter the dataset is from 1st of January 2014 to 31st of December 2015 used as training to our models and testing dataset is from 1st of January 2016 to 12th of May 2016, and finally the goal is forecast a week of May 2016 which is from 13th of May to 19th of May precisely and obtain better result than The Company's current methodology. The results compares with real data of regarding week of May 2016 and accuracy of all methodology measure with MAPE.

3.3.1 ARIMA Application

As we mentioned in *Theoretical Background Chapter*, we follow these steps [18]. Steps begin with stationary test. For this test KNIME is used.

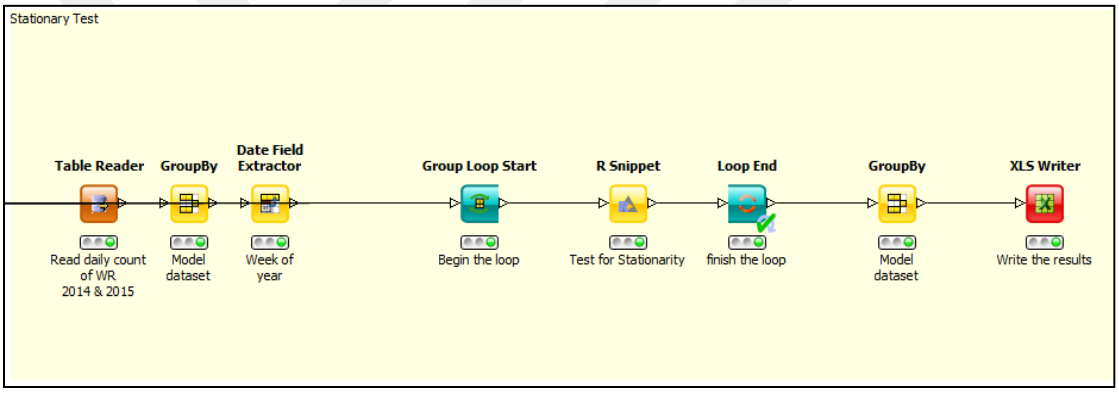


Figure 29: Stationary Test KNIME flow

In Figure 29 stationary flow application on KNIME. It begins with read the current data with Table Reader node, continuous with Groupby node to group the data by Job Type and Location, Date Field Extractor node creates week of year time variable. In second phase begins with begin the loop node; it separates all combination of job type and location variables which have 13 level II job type and 74 locations, in total 953 of 962 combinations entered the analysis, 9 of 962 combinations is not entered into analyse due to their all values are zero and executes them one by one.

Test for stationary R node calculates two test as Kwiatkowski-Phillips-Schmidt-Shin[57] (KPSS) for unit root and Ljung-Box test for stationary and if Ljung-

[57] RTMath, "Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test" https://www.rtmath.net/help/html/695835bf-570e-411f-9d76-05ee2570d0d7.htm , (12 May 2016)

Box test result is bigger than zero that means regarding variable is non-stationary, then code execute to differentiate the variable to convert the its value into stationary. The following node as Finish the Loop is to finish the combination creation and execution and write the result and Model Dataset Node is to summarize the result, and the final node XLS Writer to write the results on excel.

According to results our data is non-stationary and it is not being needed to differentiate. Next step is to apply Auto.Arima R package on our data, in this section ARIMA analysis is applied on only our lean data which only includes time stamps as year, month, week and day, and our main variables which is Work Request.

3.3.1.1 Auto.Arima Application

The Auto.Arima[58] R package is chosen because as it is mentioned in *introduction* and *Stationary* part we have dataset 963 different combinations and to define different model to every dataset combination is not efficient and economical. Therefore a random data combination is chosen, the data combination is Job Type Level II: Support RA SLA HOUR. Location Level AIU; AIU_TOVA_A

Step 1 Plot the data. Identify any unusual observations.

As it seen on Figure 30 the time plot shows some sudden changes, particularly the big drop in 2014 December due to Christmas 2014. Otherwise there is nothing unusual about the time plot and there appears to be no need to do any data adjustments.

[58] Hyndman R. J, Khandakar Y. "Automatic Time Series Forecasting: The forecast Package for R", **Journal of Statistical Software**, July 2008, Volume 27, Issue 3.

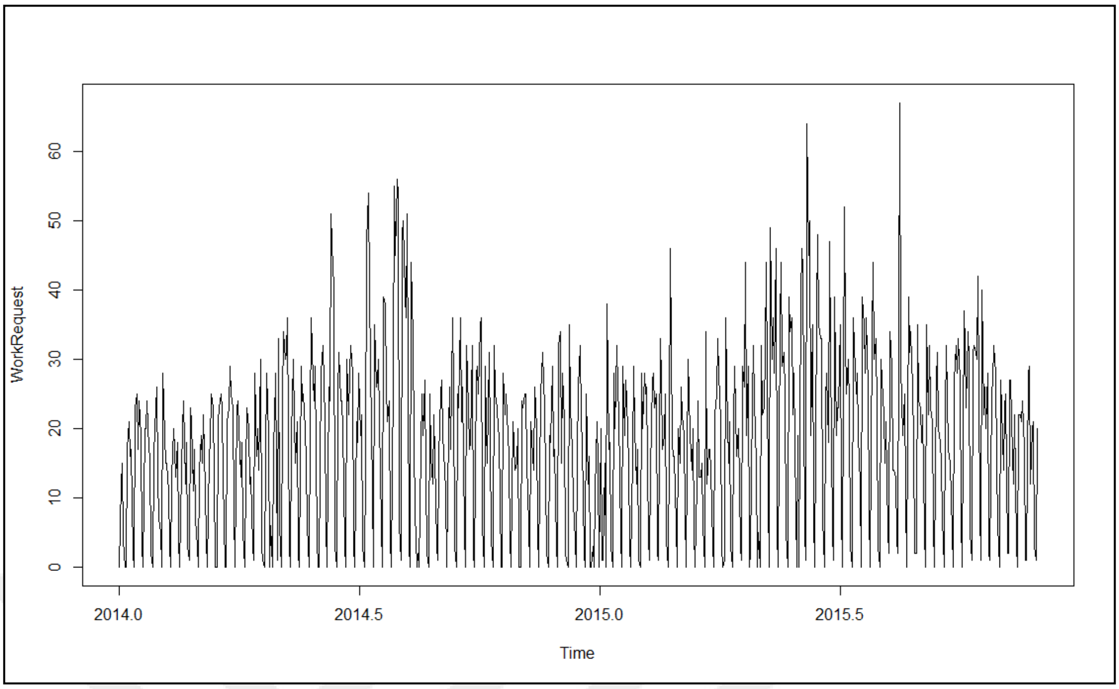


Figure 30: Graph of Support RA SLA HOUR, AIU_TOVA_A

Step 2: If necessary, transform the data (using a Box-Cox transformation) to stabilize the variance.

As it is seen on graph there is no seasonal effect in this data, peak and deep points are not similar. Also we only have two years of data therefore it is not effective to see any seasonal effects on the data.

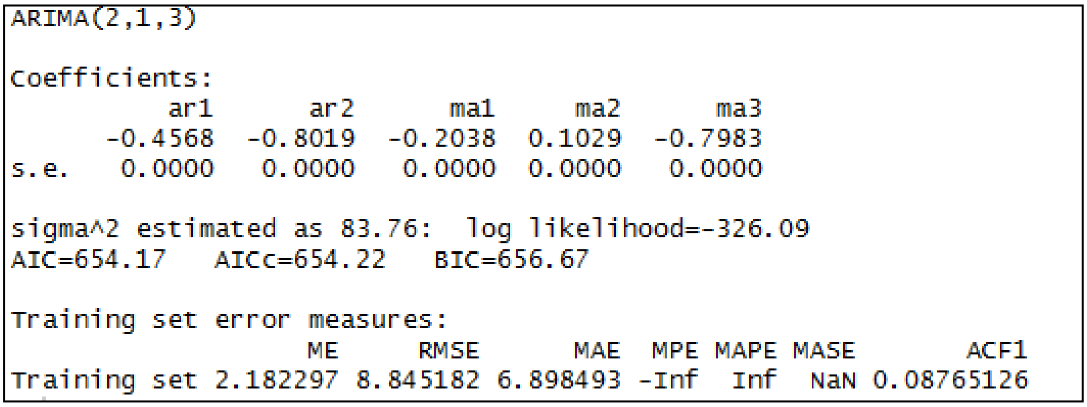
Step3-5 is skipped due to use Auto.Arima because using Auto.Arima provides to find the best ARIMA model for corresponding time series (59)

Step 6: Check the residuals

In this section the result of Box-Ljung test result is checked and according to sample, ARIMA result is:

[59] Hyndman R.J. Athanasopoulos G. (FPP) https://www.otexts.org/fpp/8/7 (13 May 2016)

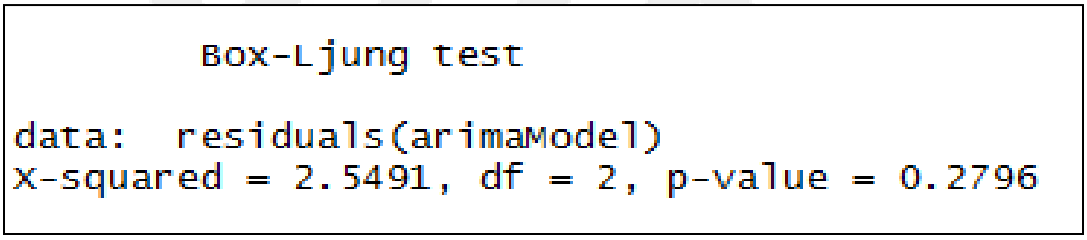
Table 15: ARIMA result of Support RA SLA HOUR, AIU_TOVA_A



```
ARIMA(2,1,3)

Coefficients:
          ar1      ar2      ma1     ma2      ma3
      -0.4568  -0.8019  -0.2038  0.1029  -0.7983
s.e.   0.0000   0.0000   0.0000  0.0000   0.0000

sigma^2 estimated as 83.76:  log likelihood=-326.09
AIC=654.17   AICc=654.22   BIC=656.67

Training set error measures:
                   ME     RMSE      MAE  MPE MAPE MASE       ACF1
Training set 2.182297 8.845182 6.898493 -Inf  Inf  NaN 0.08765126
```

As it is seen on Table 14 Auto.Arima choose the optimum ARIMA model with using AICc value here is 654,22 means after this ARIMA(2,1,3) any of chosen model's AICc value is higher than corresponding one.

Table 16: Ljung - Box Test



```
        Box-Ljung test

data:  residuals(arimaModel)
X-squared = 2.5491, df = 2, p-value = 0.2796
```

As it seen on Table 15 p-value<0.05 The Ljung-Box test examines whether there is significant evidence for non-zero correlations at lags 1-20. Small p-values (i.e., less than 0.05) suggest that the series is stationary. Therefore this sample is not stationary and needs a differentiation.

Step7 Forecast

Forecast from chosen model are show in graph 4.23

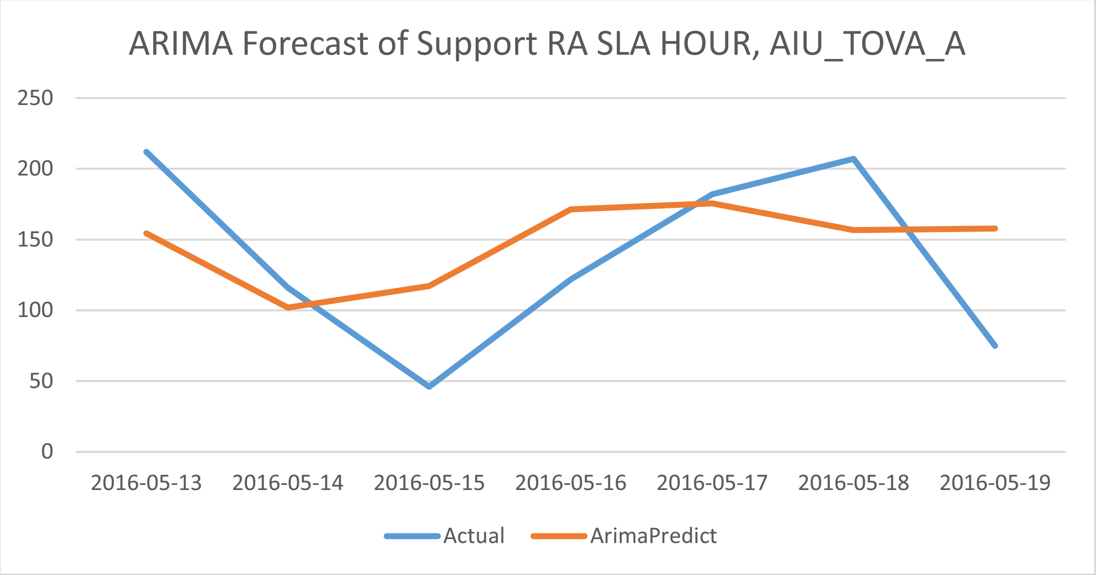


Figure 31: Graph of ARIMA Forecast of Support RA SLA HOUR, AIU_TOVA_A

All Data Validation in ARIMA Model

As it is mentioned in Introduction part Out-of-The sample validation method is used in our researched. The data are separated in two parts; training and testing. The partition is; 1st day 2014 to the end of day of 2015 for training, and 1st day of 2016 to 31st of March 2016 for test (12% of All data)

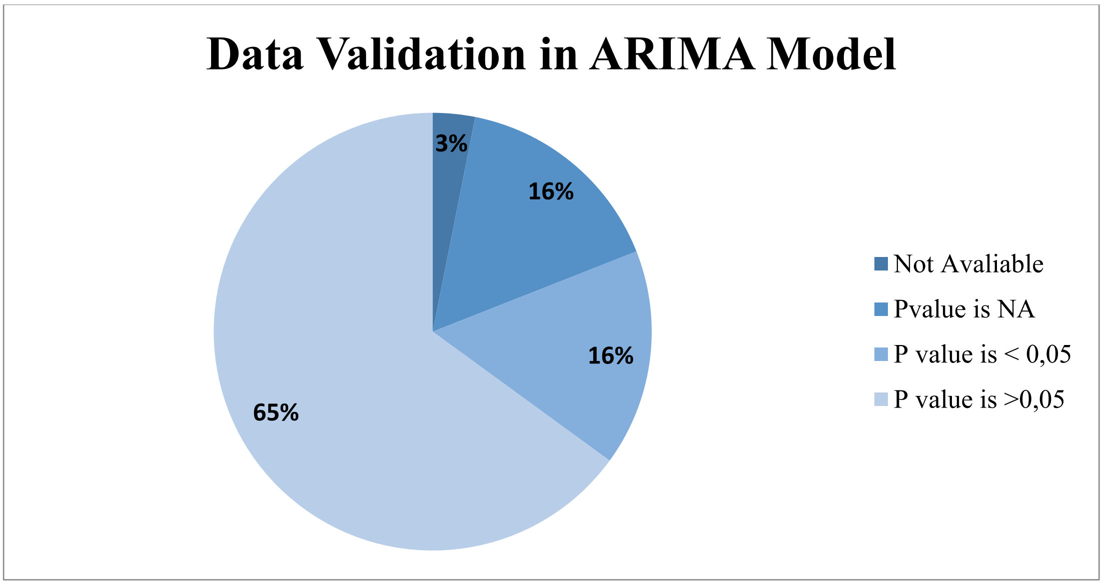


Figure 32 Graph of ARIMA model Compliance according to Ljung-Box test on all combinations

In Figure 32 is shown that 65% of data is non stationary and needs to be stationary, 16% of the data are already stationary and Auto.Arima model only uses

ARMA features for these model. After Auto.Arima model application all the available data is managed to convert into stationary data.

If it is considered only the available data, eliminated 183 combinations which are included NA and irrelevant values. Then 779 combinations out of 962 is obtained.

3.3.2 Random Forest Application

In random forest application we followed the steps that are referred on theoretical background. In additional we add several data which are own importance to affect our prediction result. Also we demonstrate the variable importance in Importance section.

Model:

In model section we followed randomForest R package [60]. Our model is:

Random Forest model = randomForest(Sum_WRs(Null variable, aim is to predict this variable) ~.(all variables that we mentioned in Data Chapter), data= "Training Data",ntree=500, Importance= True)

Description of Model

ntree: Number of trees to grow. This should not be set to too small a number, to ensure that every input row gets predicted at least a few times.

Importance: Should importance of predictors be assessed? If it is yes. Print Variables importance

The data separated into two pieces as ARIMA model with same date.

Variable Importance

Definitions of the variable importance measures, the first measure is computed from permuting OOB data: For each tree, the prediction error on the out-of-bag portion of the data is recorded (error rate for classification, MSE for regression). Then the same is done after permuting each predictor variable. The difference between the two are then averaged over all trees, and normalized by the standard deviation of the differences. If the standard deviation of the differences is equal to 0 for a variable, the division is not done (but the average is almost always equal to 0 in that case) [58].

Table 17: Pseudo-Code of Random Forest Importance

[60] Breiman L and Cutler A,CRAN-R,"Package 'randomForest' " https://cran.r-project.org/web/packages/randomForest/randomForest.pdf , (13 May 2016)

```
In R code, Random forest importance is:
1. Compute model MSE
2. For each variable in the model:
      2.1 Permute variable
      2.2 Calculate new model MSE according to variable permutation
      2.3 Take the difference between model MSE and new model MSE
3. Collect the results in a list.
4. Rank variables' importance according to the value of the %IncMSE. The
   greater the value the better.
```

Variable Importance

The second measure is the total decrease in node impurities from splitting on the variable, averaged over all trees. For classification, the node impurity is measured by the Gini index. We selected only %IncMSE as an Importance accurator. For regression, it is measured by residual sum of squares, and the chosen combination Importance calculated in R with randomForest package (importance) [(61)] and its graph is like:

[61] Breiman L and Cutler A, Inside R,"importance {randomForest} Extract variable importance measure" http://www.inside-r.org/packages/cran/randomforest/docs/importance , (17 May 2016)

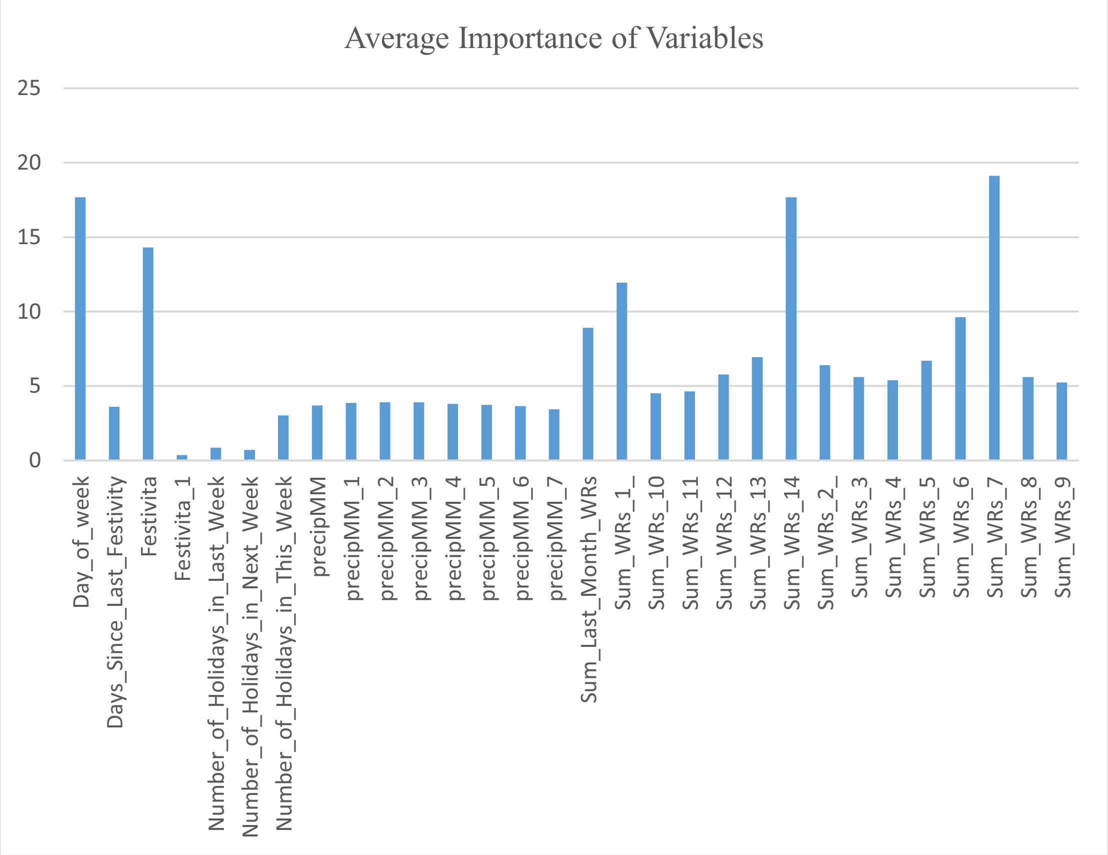


Figure 33: Graph of Variable Importance in Random Forest application

It is seen on Figure 33 the variable which has the greatest importance is 7th lagged value of Sum of Work Request variable.

In order to obtain the result of Random Forest application the Prediction function of randomForest[62]. The prediction formula is:

predict ("Random Forest model", test dataset)

Description:

Random Forest Model: an object of class randomForest, as that created by the function randomForest

Test: a data frame or matrix containing new data.

[62] Breiman L and Cutler, "A{predict.randomForest} Prediction of test data using random forest." https://cran.r-project.org/web/packages/randomForest/randomForest.pdf June 2016

3.3.3 Extreme Gradient Boosting Application

XGBoost algorithm is one of the popular winning recipe of data science competition, it is pretty new algorithm whose R implementation was launched August 2015. In this research we used “xgboost” R package to analyse our dataset. Although there are very rare article mentioned Extreme Gradient Boosting. This algorithm mostly are used in Time Series forecasting base data analysis competition. Therefore we decided to try Extreme Gradient Boosting methods on our dataset.

Our model is:

Xgboost(Data= “Train Data”, Max.depth = 2, eta =1, nthread=4, nround=100, verbouse= 0). In order to explain the model;

Max.Depth: Parameters, maximum depth of the tree

Eta: control the learning rate: scale the contribution of each tree by a factor of 0 < eta < 1 when it is added to the current approximation. Used to prevent overfitting by making the boosting process more conservative. Lower value for eta implies larger value for nrounds: low eta value means model more robust to overfitting but slower to compute. Default: 0-3(63)

nthread: number of thread used in training, if not set, all threads are used

nround: the max number of iterations

Verbouse: Boolean, print the statistics during the process

Prediction: We use same prediction function as Random Forest it is: predict (“XGBoost Model”, test dataset)

Variable Importance

In this section choose same variables as Random Forest to compare two accuracy and efficiency of both models.

3.3.4 Knime Application of Random Forest and Extreme Gradient Boosting

In Data Preparation Processing part we mentioned how we enrich the data. In this section we demonstrate KNIME application which runs the Random Forest Model.

[63] Chen T, CRAN-R "Package ‘xgboost’ " https://cran.r-project.org/web/packages/xgboost/xgboost.pdf , (18 May 2016)

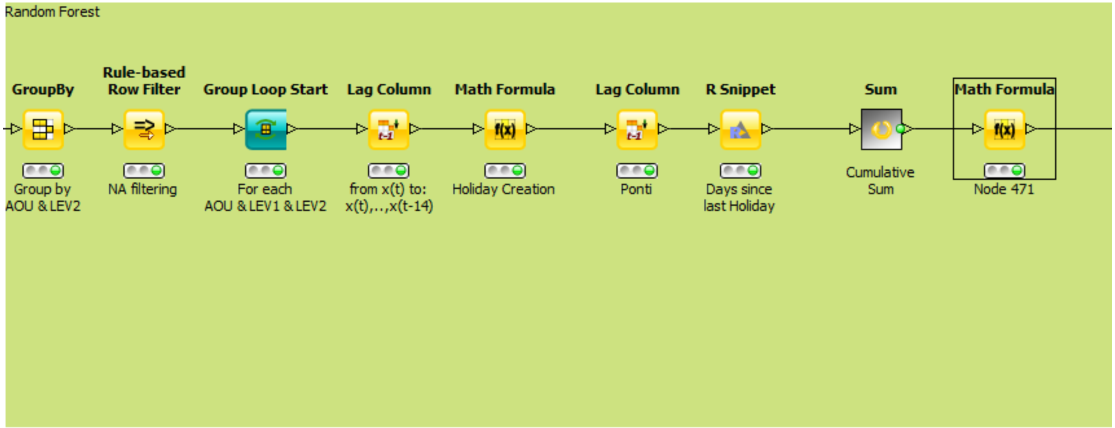
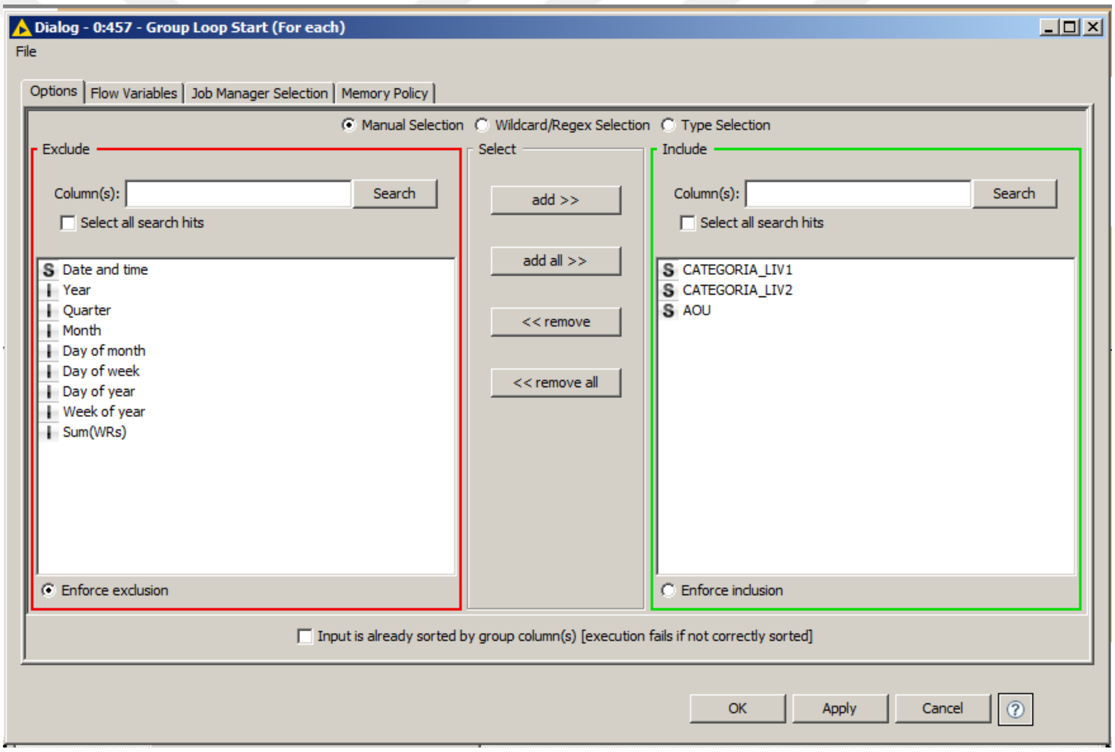


Figure 34: Data entry to Random Forest Application flow

In Figure 34, data enter the flow with passing Loop node [64]



● Manual Selection ○ Wildcard/Regex Selection ○ Type Selection

Exclude

Column(s):

□ Select all search hits

- S Date and time
- I Year
- I Quarter
- I Month
- I Day of month
- I Day of week
- I Day of year
- I Week of year
- I Sum(WRs)

● Enforce exclusion

Select

Include

Column(s):

□ Select all search hits

- S CATEGORIA_LIV1
- S CATEGORIA_LIV2
- S AOU

○ Enforce inclusion

□ Input is already sorted by group column(s) [execution fails if not correctly sorted]

Figure 35: Inside of Loop node.

As it is seen on Figure 35 Random Forest Application flow loops the all flow according to three variables; LEVEL 1 CATEGORY as known as Job Type Level 1, LEVEL 2 CATEGORY as known as Job Type Level 2 and AIU which is referred to each location which data are created.

[64] KNIME, "Loop and flow variable", https://tech.knime.org/forum/knime-users/loop-and-flow-variable , (20 May 2016)

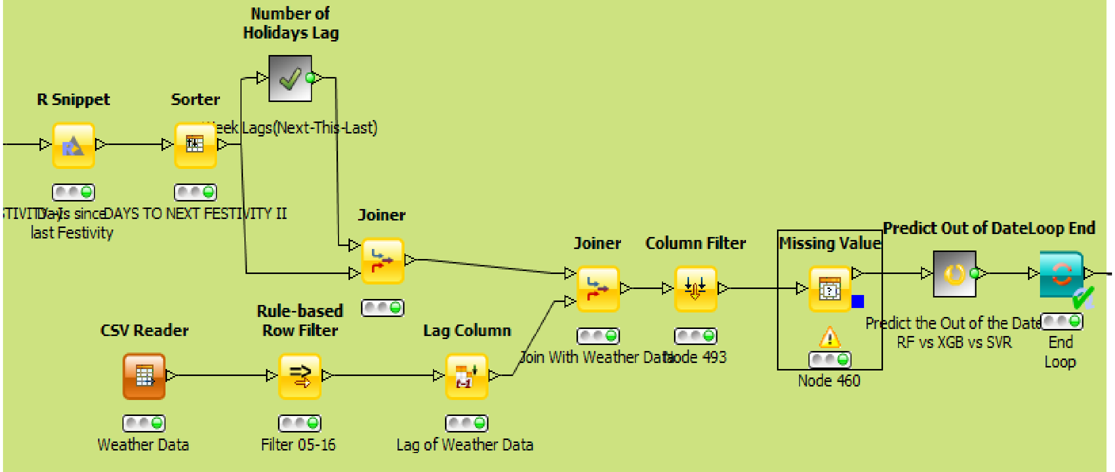
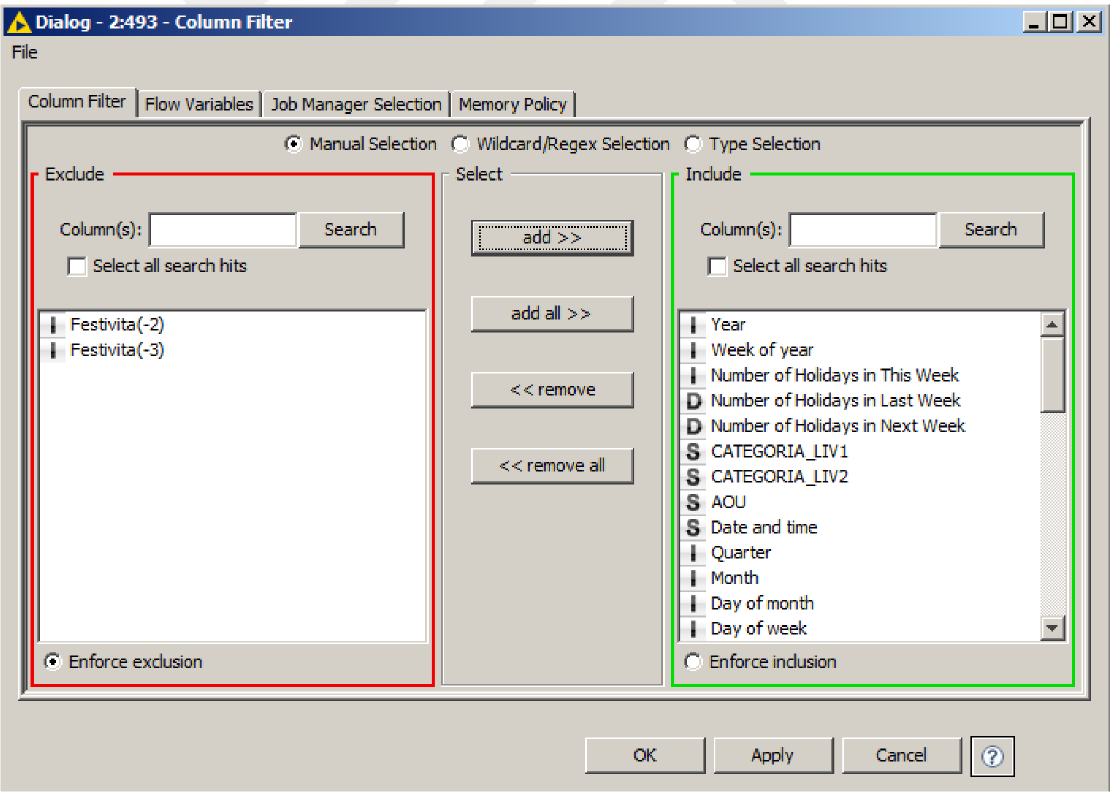


Figure 36: Joining with Featured variables, Random Forest and XGB applications, and end of the loop

In Figure 36 shows joining with featured data which are created by us and weather data. Then Column filter node filtered out the data that are not to be needed in the model.


Dialog - 2:493 - Column Filter

File

Column Filter | Flow Variables | Job Manager Selection | Memory Policy

● Manual Selection ○ Wildcard/Regex Selection ○ Type Selection

Exclude

Column(s): [ ] Search

□ Select all search hits

- Festivita(-2)
- Festivita(-3)

● Enforce exclusion

Select

add >>

add all >>

<< remove

<< remove all

Include

Column(s): [ ] Search

□ Select all search hits

- I Year
- I Week of year
- I Number of Holidays in This Week
- D Number of Holidays in Last Week
- D Number of Holidays in Next Week
- S CATEGORIA_LIV1
- S CATEGORIA_LIV2
- S AOU
- S Date and time
- I Quarter
- I Month
- I Day of month
- I Day of week

○ Enforce inclusion

OK | Apply | Cancel | ?

Figure 37: Column Filter node, filtering out not to be needed variables.

In Figure 37 the variables which are not be needed, be taken out from model. We decided to take out these variables our first Random Forest Attempt.

According to our first random forest attempt. The importance graph is:

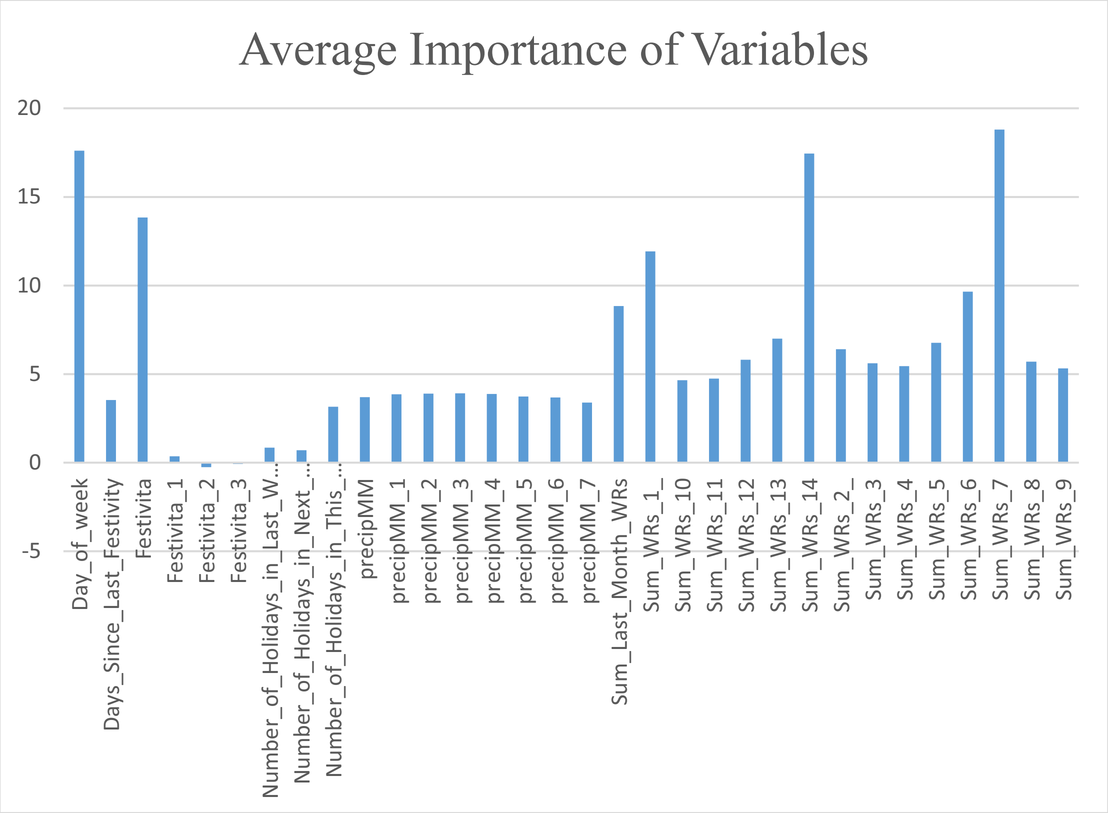


Figure 38: Graph of former Random Forest Importance of Variables

As it is seen on Figure 38 importance of Festivity-2 (the second lag of Holidays) and Festivity-3 (the third lag of Holidays) are negative. Therefore those variables are taken out from model with using column filter node.

Inside of Predict Out of the Date Meta Node, Random Forest and Extreme Gradient Boosting model are evaluated for each location and job type.

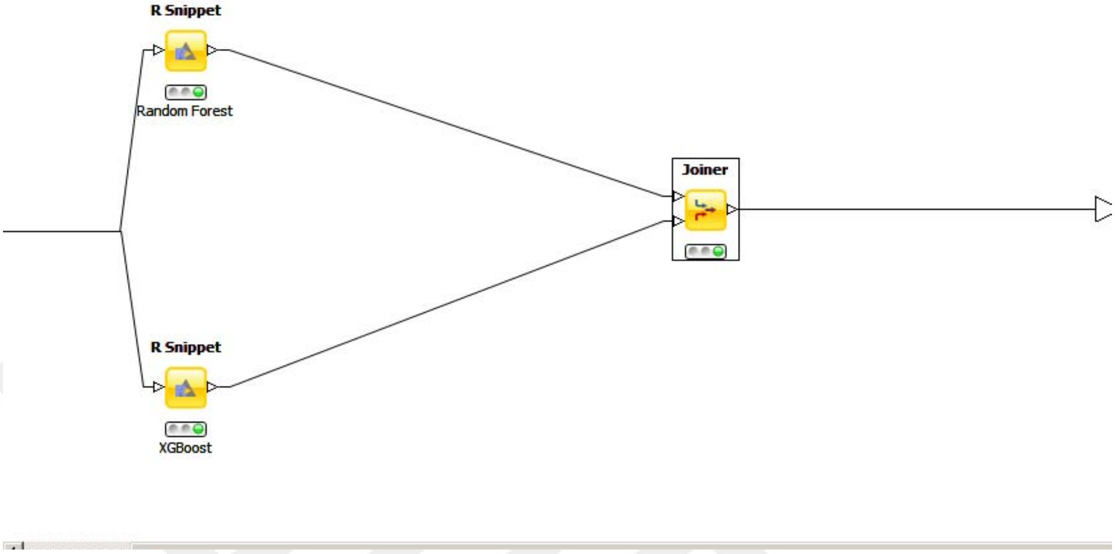


Figure 39: Random Forest and XGB application

In Figure 39 we apply Random Forest and Extreme Gradient Boosting at the same time and join the result to show them in same table but separately.

Finally all flow in once is shown Figure 40

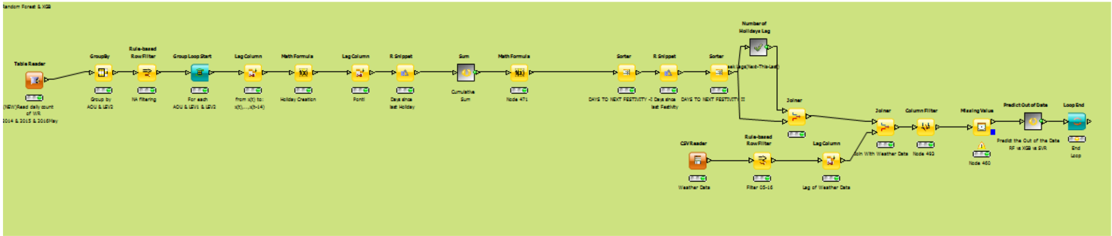


Figure 40: All flow of Random Forest and Extreme Gradient Boosting

CHAPTER FOUR

RESULTS

In this chapter the results are evaluated with several data level and days of week. The data levels are type of Work Requests as Level I and Level II types. We decide to inspect weekdays to understand days of week effect.

4.1 Results According to Data Levels

In this section the results are inspected according to types level I and II.

4.1.1 Results According to Data Type Level I

As we mentioned in Data Chapter Level II is Sub group of Level I. The work request distribution is:

Table 18: Work Request Distribution According to Level I

Type	Weight
Support	54,1%
Distributions	30,8%
Distributions Permute	15,1%
Total	100,0%

Inspect to the result with using MAPE accuracy method:

Table 19: Result matrix according to Data Type Level I

Levels/Methods	Random Forest	XGB	Auto.Arima
Support	0,1256	0,0802	0,0567
Distributions OL Cons. Trans, Retail/Wholesale	0,1555	0,2397	0,0926
Distributions OL Permute	0,1304	0,2382	0,0565

Result Graphs of Level II Job Type

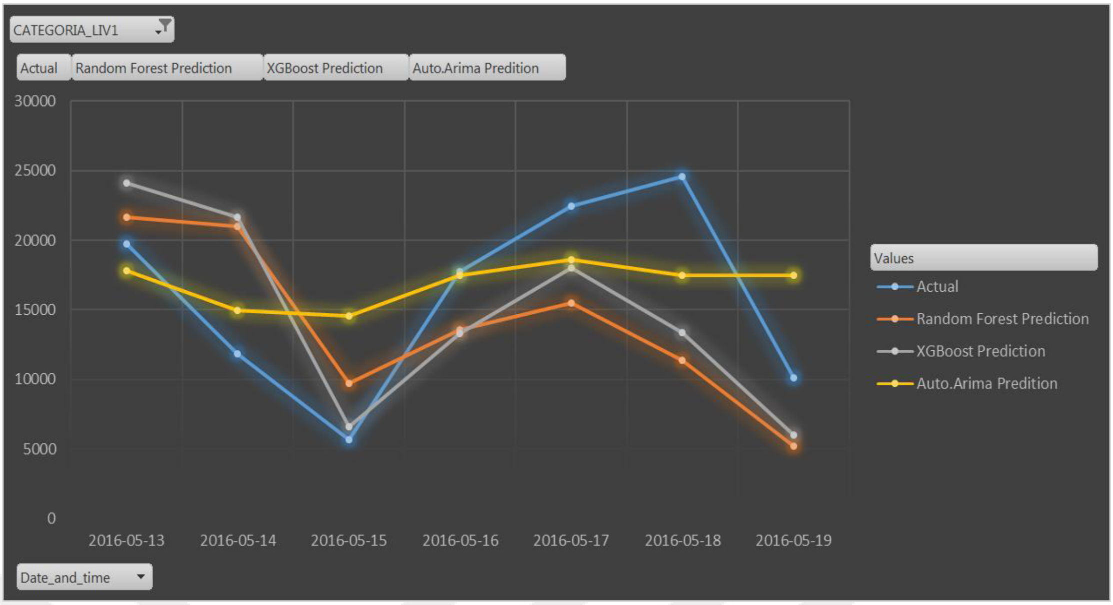


Figure 41: Graph of Results of Support

As it is seen on Figure 41 XGBoost and Random Forest models are more descriptive than ARIMA for only weekends in Job Type Level I Support. Rest of the result ARIMA provides better results.

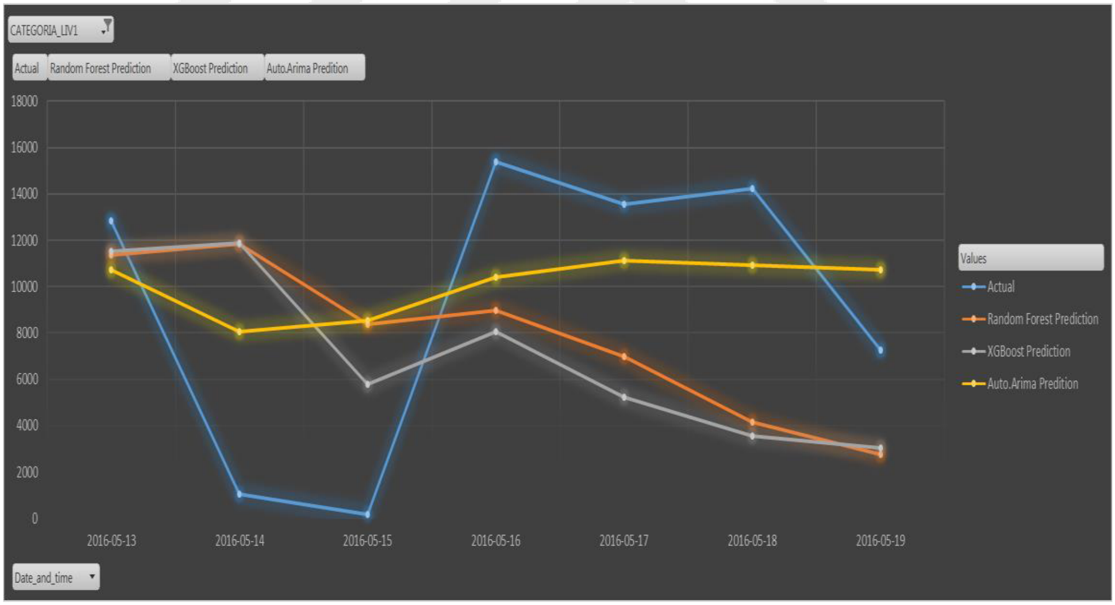


Figure 42: Graph of Results of Distributions OL Cons. Trans, Retail/Wholesale

According to Job type Level I Distributions OL Cons. Trans, Retail/Wholesale ARIMA model reaches the lowest error rate (Figure 42).

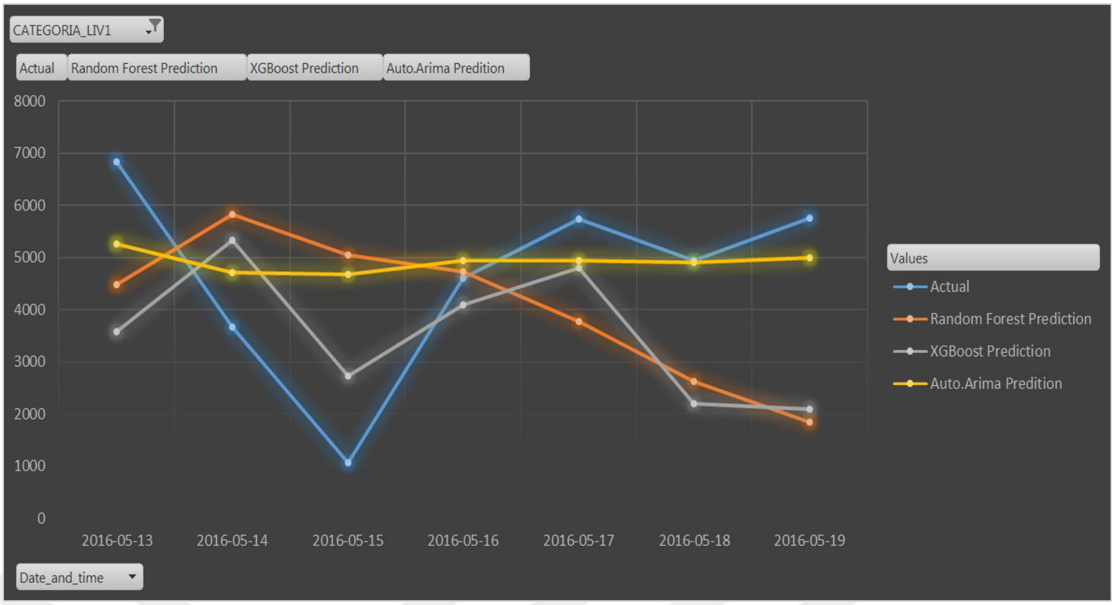


Figure 43: Graph of Results of Distributions OL Permute

According to Figure 43 XGBoosting model is more descriptive especially on weekends

4.1.2 Results According to Data Type Level II

Data type Level II includes more detailed information if it is compared with Level I. Work request distribution of Level II is:

Table 20: Result matrix according to Data Type Level II Support

Type of Work Request	Weight
Support Fibre	2,12%
Support NOF G.	2,31%
Support Data Products	0,02%
Support Telephone Products	0,90%
Support RA SLA DAY.	45,37%
Support RA SLA HOUR	3,35%
Distributions Data S/C/T	1,57%
Distributions Data Products	0,70%
Distributions Telephone Products	0,14%
Distributions ULL (Permute)	8,00%
Distributions Cons & Tans	5,47%
Distributions Cons &ADSL	22,90%
Distributions Permute	7,12%
Total	100%

As it is seen on Table 20 almost half of all work requests are created by Support RA SLA DAY.

Level II Support

Table 21: Result matrix according to Data Type Level II Support

Levels/Methods	Random Forest	XGB	Auto.Arima
Support Fibre	0,2753	0,2969	0,0999
Support NOF G.	0,0147	0,0465	0,0717
Support Data Products	0,0675	0,4432	0,1080
Support Telephone Products	0,1774	0,1480	0,0015
Support RA SLA G.	0,1100	0,0499	0,0534
Support RA SLA HOUR	0,2223	0,2839	0,0558

According to Table 20 and 21, when total work request by job type is decreased, XGBoosting and Random Forest models provide better results than ARIMA models. This interpretation can be seen well on following Figures in this chapter.

Result Graphs of Level II Job Type

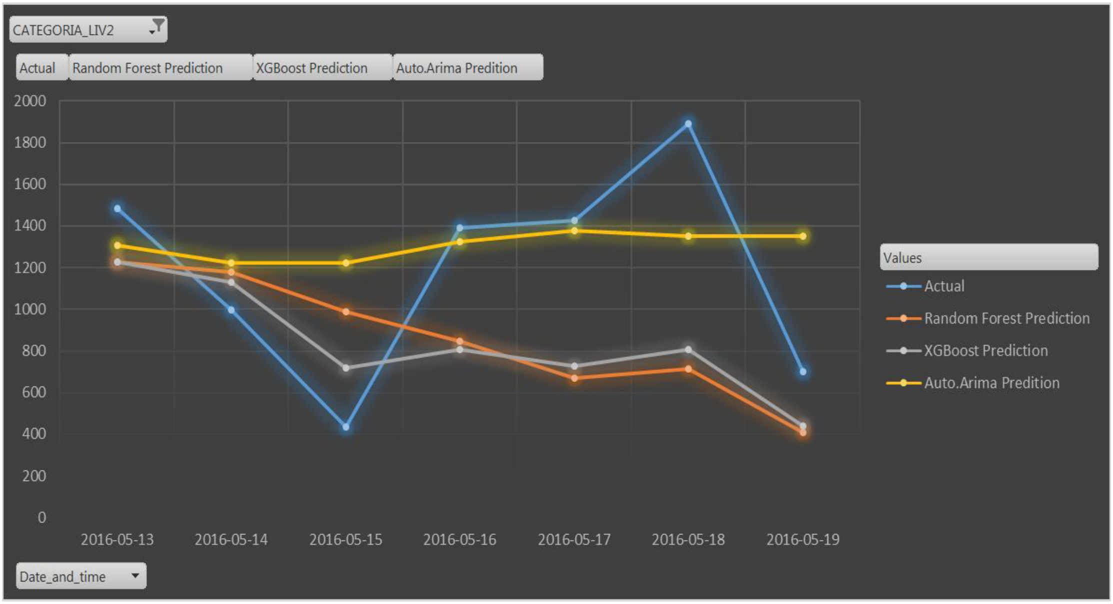


Figure 44: Graph of Results of Support Fibre

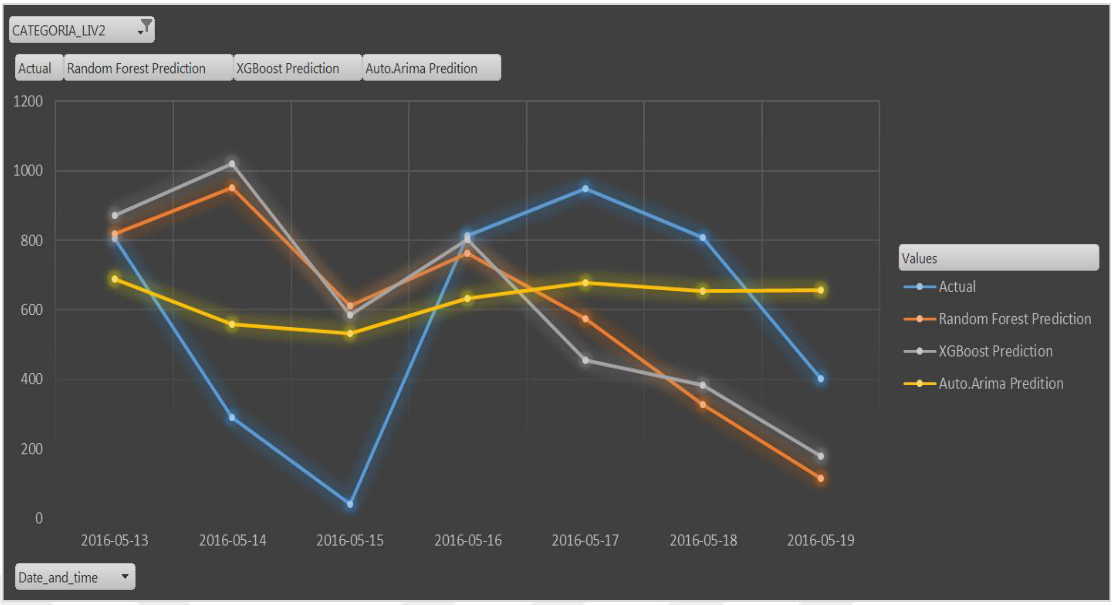


Figure 45: Graph of Results of Support NOF G.

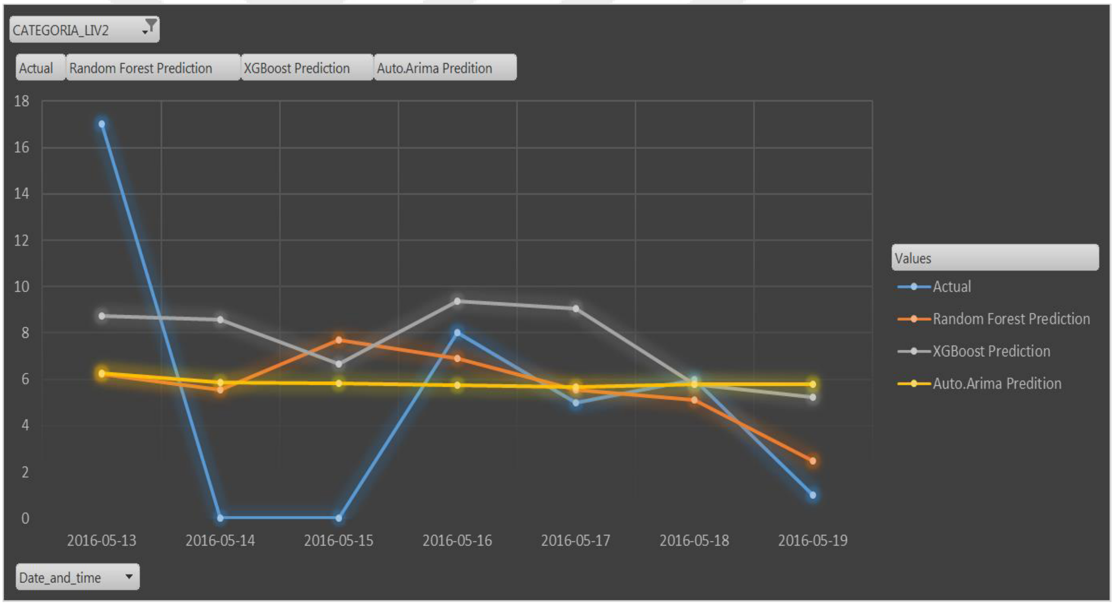


Figure 46: Graph of Results of Support Data Products

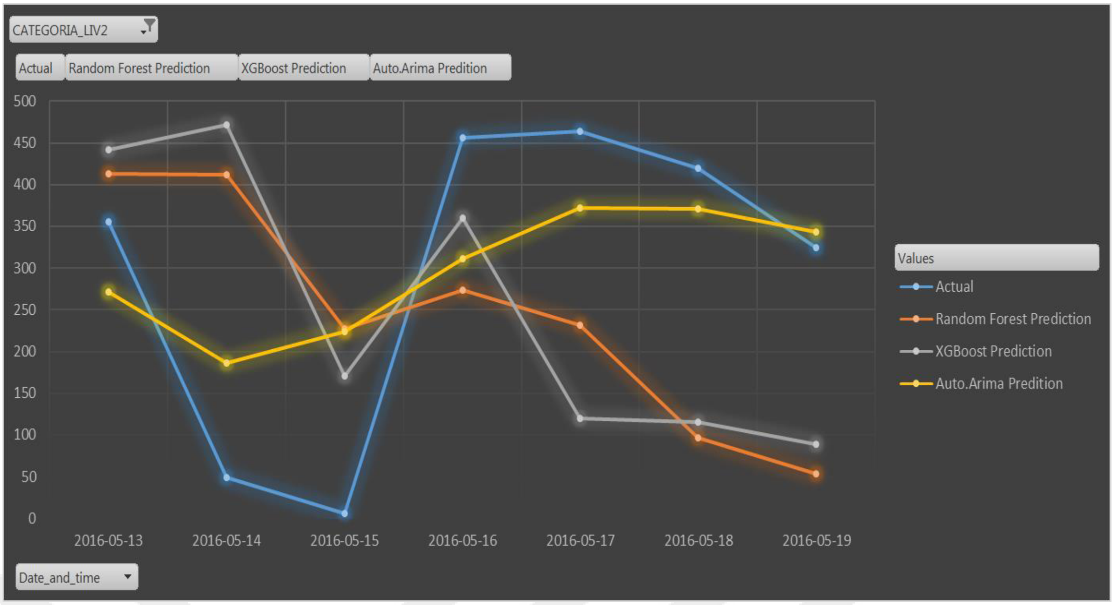


Figure 47: Graph of Results of Support Telephone Products

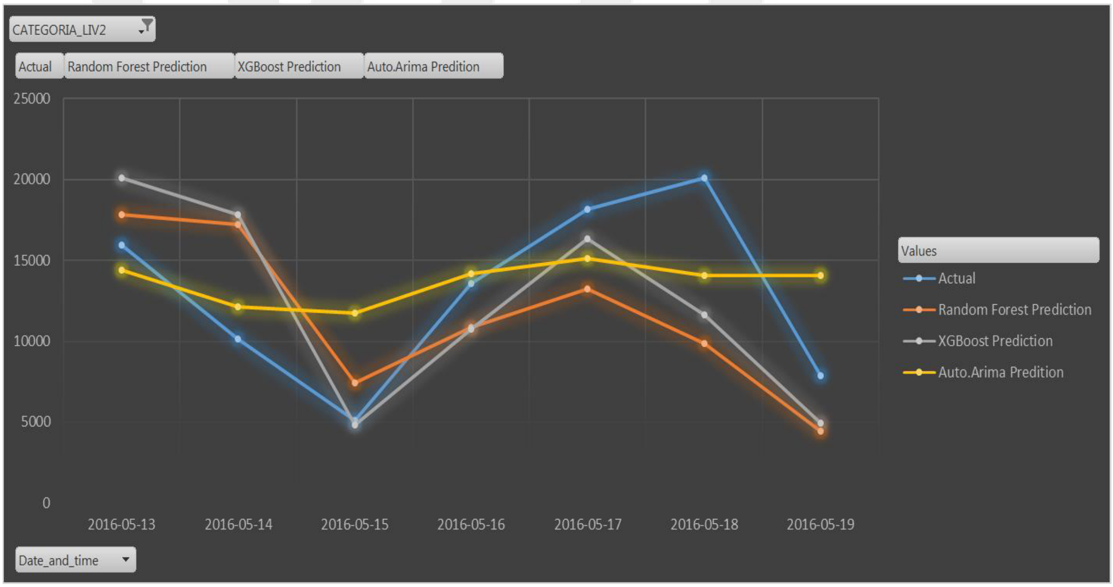


Figure 48: Graph of Results of Support RA SLA G.

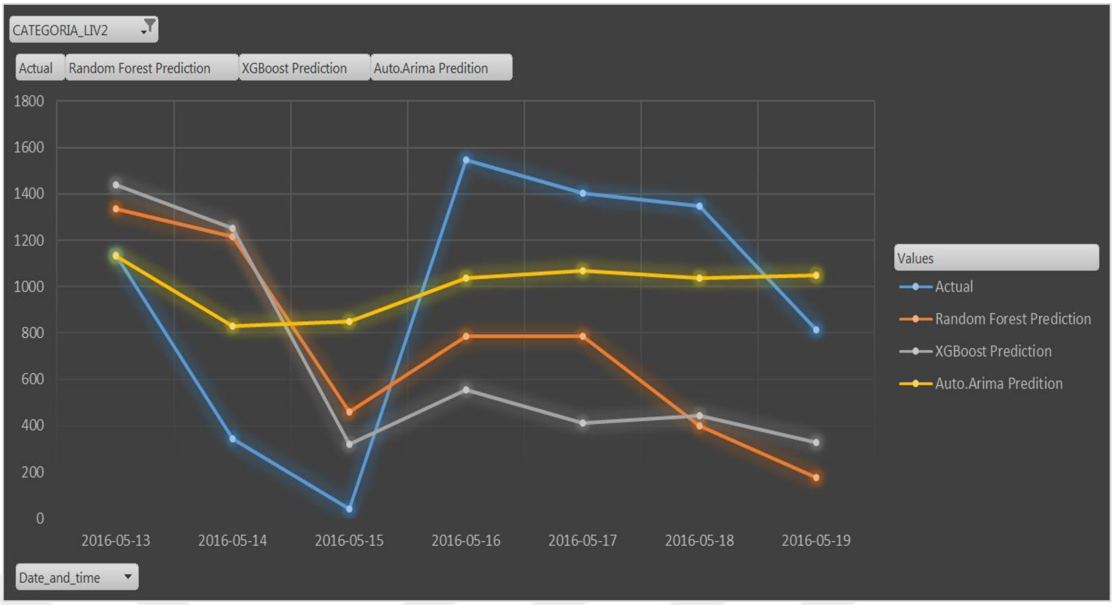


Figure 49: Graph of Results of Support RA SLA HOUR

Level II Distributions OL Cons. Trans, Retail/Wholesale

Table 22: Result matrix according to Data Type Level II Distributions

Levels/Methods	Random Forest	XGB	Auto.Arima
Distributions Data S/C/T	0,0017	0,1299	0,0829
Distributions Data Products	0,2330	0,2461	0,1136
Distributions Telephone Products	0,1766	0,1540	0,0469
Distributions Cons & Tans	0,2692	0,3314	0,1531
Distributions Cons &ADSL	0,1197	0,2126	0,0704

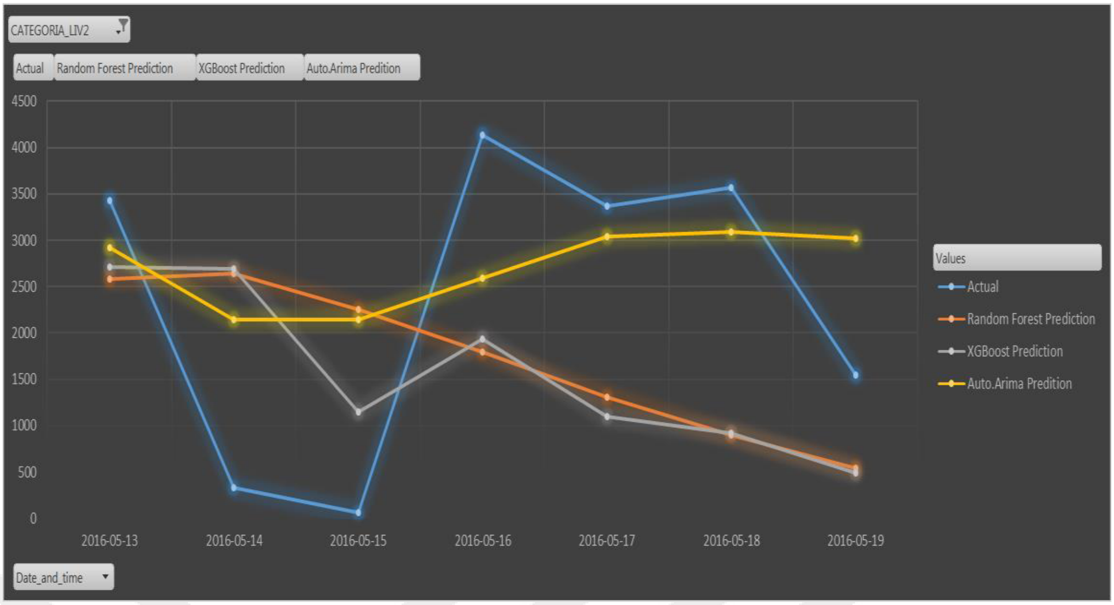


Figure 50: Graph of Results of Distributions Data S/C/T

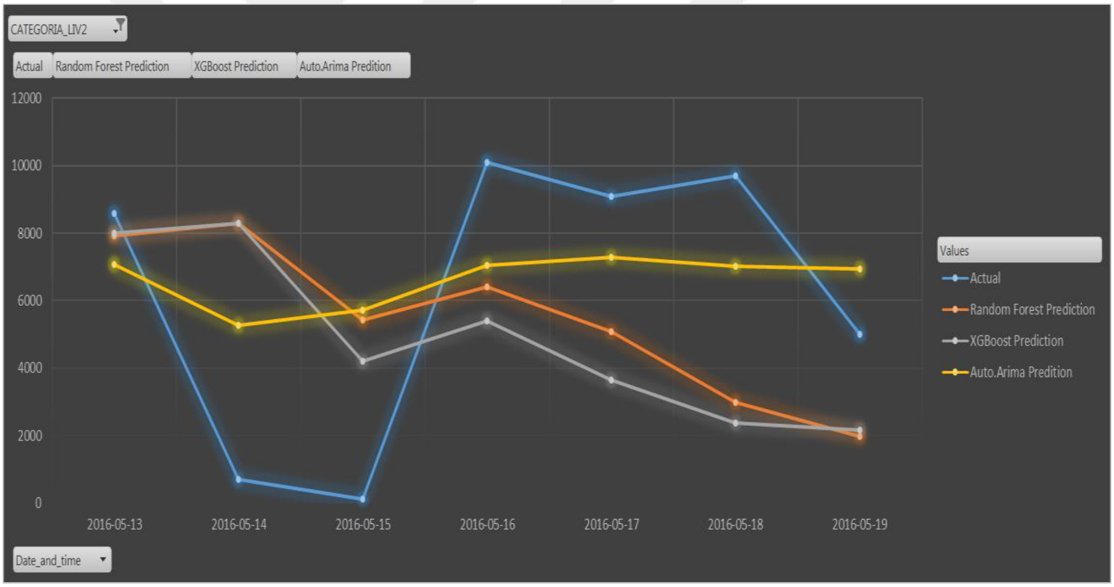


Figure 51: Graph of Results of Distributions Data Products

Although the actual value of Distributions Data S/C/T and Distributions Data Product are similar, the predictive models do not seem significant to obtain the lowest MAPE value. Also those predict models results seem different per each job types.

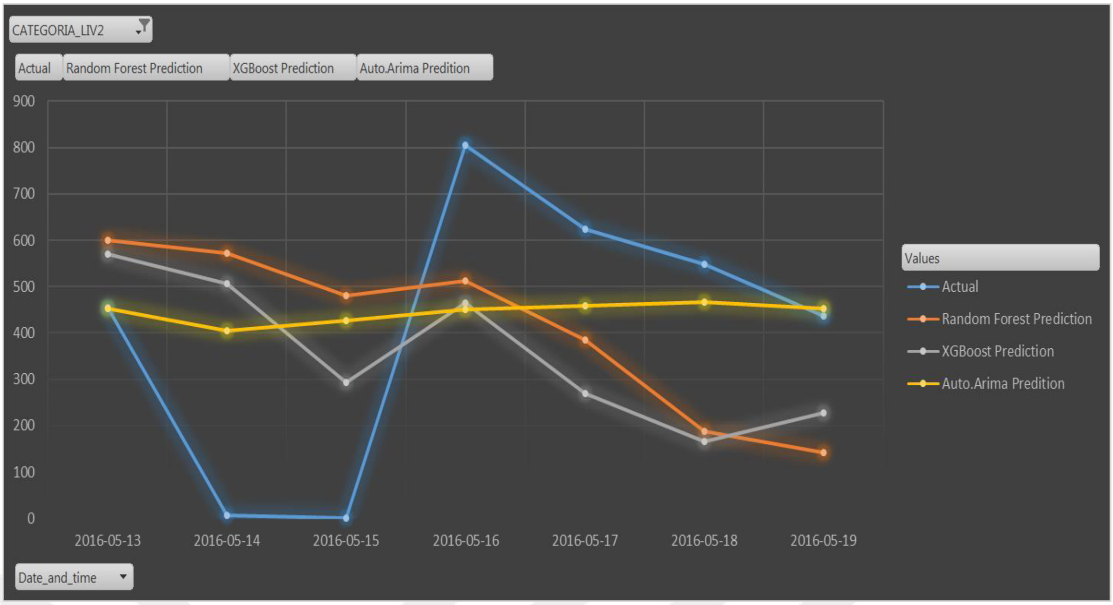


Figure 52: Graph of Result of Distributions Telephone Products

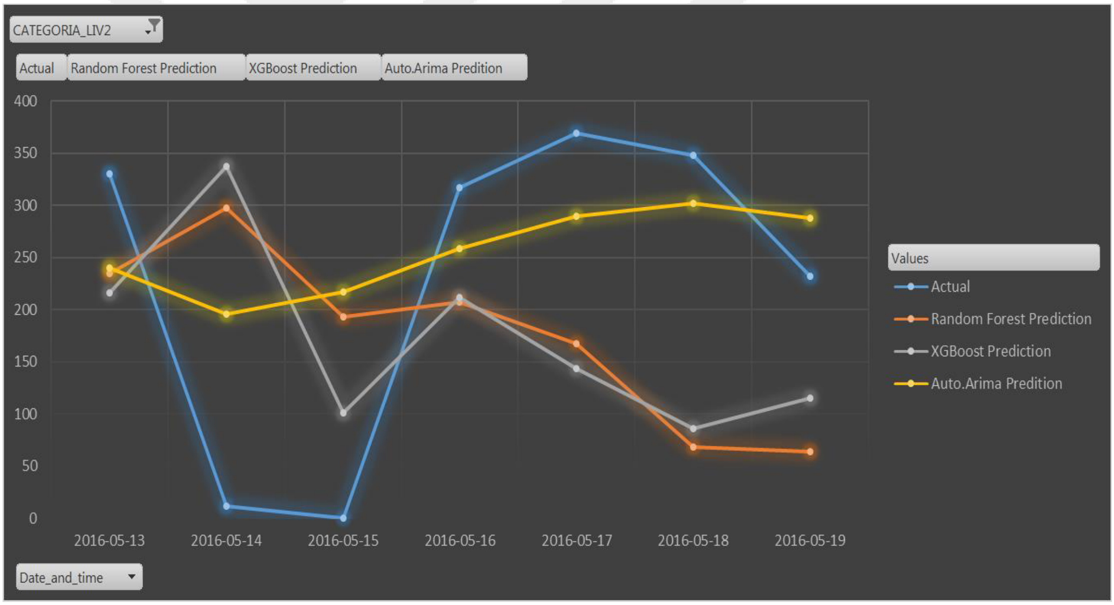


Figure 53: Graph of Result of Distributions Cons & Tans

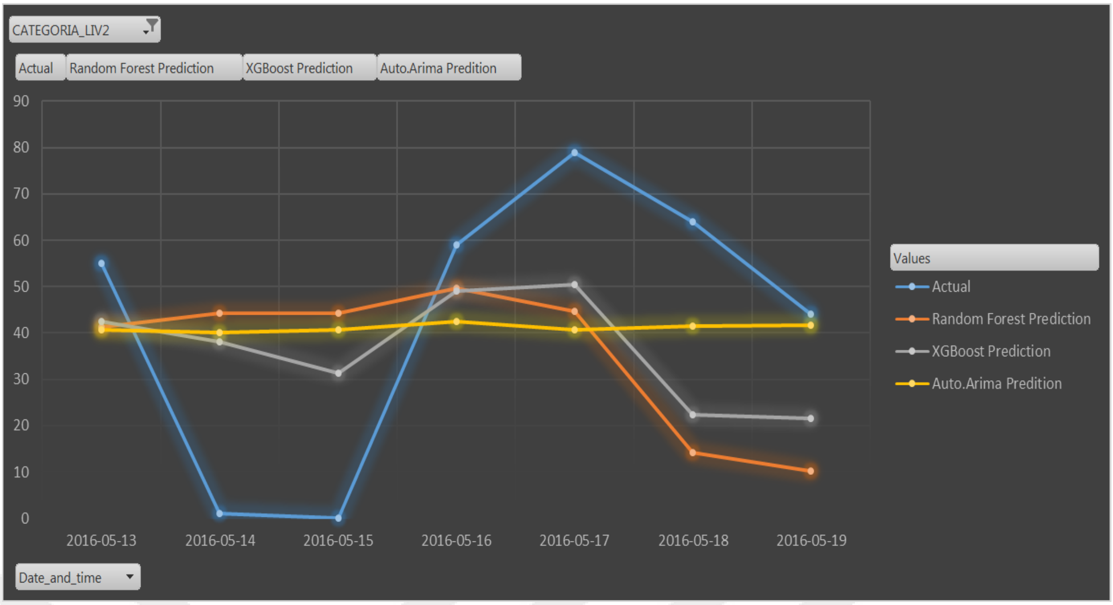


Figure 54: Graph of Results of Distributions Cons &ADSL

Level II Distributions OL Permute

Table 23: Result matrix according to Data Type Level II Distributions II

Levels/Methods	Random Forest	XGB	Auto.Arima
Distributions Permute	0,2124	0,2425	0,0619
Distributions ULL (Permute)	0,0319	0,2330	0,1988

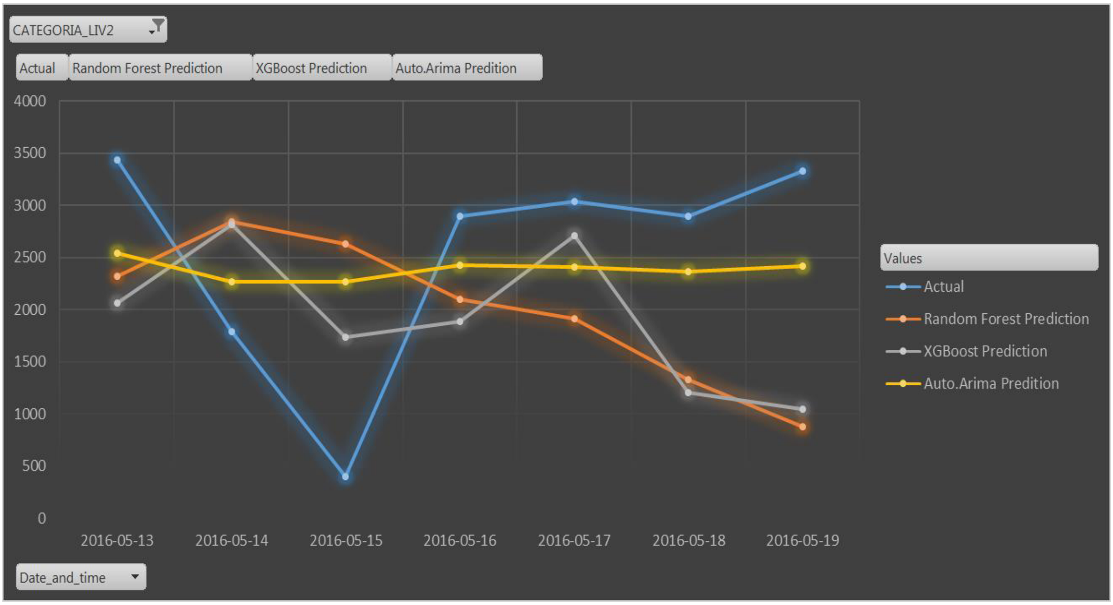


Figure 55: Graph of Results of Distributions Permute

As it is seen on Figure 55 XGB responds the beginning of Weekend trend, however after Sunday it fails to predict Monday well.

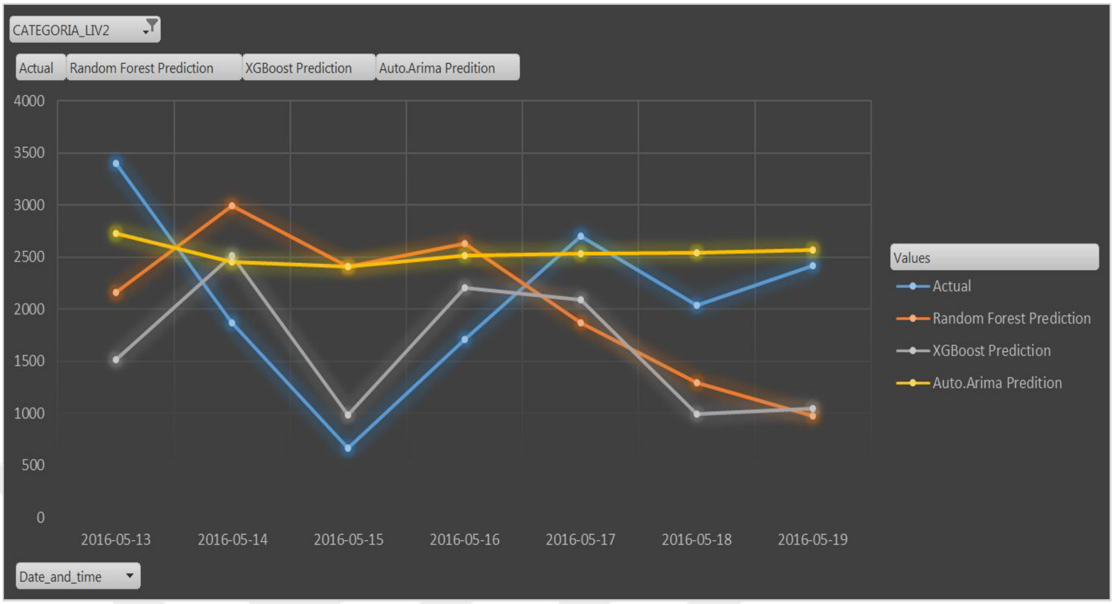


Figure 56: Graph of Results of Distributions ULL (Permute)

In Figure 56, XGB model provides the lowest MAPE value for Sunday.

4.2 Results According to Days of Week

To know the weights of the each days of week is quite important to understand the results. The weight distribution of days of week are:

Table 24: Work Request Distribution according to days of week

Days Of Week	Weight
Sunday	2%
Monday	17,1%
Tuesday	18,5%
Wednesday	18,4%
Thursday	18,4%
Friday	17,6%
Saturday	7,7%
Total:	100%

According to Table 24, 90,1% of work request happens on weekdays, Sundays have the least count of work requests. In Table 25 while we consider the days of weekend the MAPE value of Auto.Arima is dramatically increasing. However lower MAPE value is obtained with using XGB for the days of weekend.

Table 25: Result Distribution according to days of week

Days Of Week	Random Forest	XGB	Auto.Arima
Sunday	236,10%	119,25%	302,87%
Monday	27,88%	32,68%	13,10%
Tuesday	37,08%	32,74%	16,83%
Wednesday	58,45%	56,23%	23,83%
Thursday	57,62%	51,91%	43,82%
Friday	4,83%	0,51%	14,21%
Saturday	133,78%	135,02%	67,33%

4.3 Results According to Location

Break down of location is:

Table 26: Result Breakdown according to days of week

AIU	Weight
AIU_AM_N	1,240%
AIU_AM_S	1,247%
AIU_BAS_BE	1,102%
AIU_BAS_BO	1,076%
AIU_CAL_N	1,441%
AIU_CAL_S	1,351%
AIU_CAM_C	2,893%
AIU_CAM_S	2,284%
AIU_EMO_M	1,268%
AIU_EMO_P	1,162%
AIU_EMO_R	1,006%
AIU_ER_B	1,563%
AIU_ER_F	1,155%
AIU_ER_R	1,229%
AIU_FVG_B	0,762%
AIU_FVG_P	0,809%
AIU_FVG_TS	0,877%
AIU_FVG_TV	1,011%
AIU_LAZ_NE	1,534%
AIU_LAZ_NO	1,628%
AIU_LAZ_SE	1,705%
AIU_LAZ_SO	1,809%
AIU_LCE_B	1,314%
AIU_LCE_BN	1,216%
AIU_LCE_BS	1,138%
AIU_LCE_M	1,201%
AIU_LIG_L	1,504%
AIU_LIG_P	1,291%
AIU_LN_C	1,567%
AIU_LN_M	1,359%
AIU_LO_C	1,665%

AIU_LO_V	1,195%
AIU_MAR_A	1,201%
AIU_MAR_M	1,231%
AIU_MI_MC	0,900%
AIU_MI_ML	1,153%
AIU_MI_MS	1,449%
AIU_MI_MT	1,011%
AIU_NA_C	1,905%
AIU_NA_E	2,039%
AIU_NA_N	2,101%
AIU_PIE_A	0,837%
AIU_PIE_C	1,052%
AIU_PIE_N	1,041%
AIU_PUG_B	1,492%
AIU_PUG_F	1,340%
AIU_PUG_S	1,496%
AIU_RM_C	1,420%
AIU_RM_E	1,860%
AIU_RM_O	1,579%
AIU_RM_S	1,781%
AIU_SAR_N	1,128%
AIU_SAR_S	1,162%
AIU_SIE_C	1,876%
AIU_SIE_M	1,080%
AIU_SIE_R	1,150%
AIU_SIO_A	1,298%
AIU_SIO_PE	1,415%
AIU_SIO_SO	1,562%
AIU_TAA_B	1,332%
AIU_TAA_V	1,296%
AIU_TOE_C	1,255%
AIU_TOE_N	1,506%
AIU_TOE_S	1,172%
AIU_TOO_C	1,281%
AIU_TOO_N	1,110%
AIU_TOO_S	1,158%
AIU_TOVA_A	1,124%
AIU_TOVA_TC	1,270%
AIU_TOVA_TS	1,344%
AIU_UMB	1,197%
AIU_VE_P	1,255%
AIU_VE_VE	1,450%
AIU_VE_VI	1,095%
TOTAL	100%

The result of analysis by location:

Table 27: Result Distribution according to locations

AIU	Random Forest	XGB	Auto.Arima
AIU_AM_N	0,1517	0,1165	0,0544
AIU_AM_S	0,1473	0,2813	0,0593
AIU_BAS_BE	0,1049	0,0975	0,1103
AIU_BAS_BO	0,2445	0,3733	0,0033
AIU_CAL_N	0,0131	0,1005	0,2053
AIU_CAL_S	0,0644	0,1366	0,1222
AIU_CAM_C	0,0809	0,1559	0,0112
AIU_CAM_S	0,0197	0,1010	0,0698
AIU_EMO_M	0,2340	0,2486	0,0689
AIU_EMO_P	0,1431	0,1656	0,0567
AIU_EMO_R	0,2543	0,2838	0,0419
AIU_ER_B	0,1448	0,0886	0,0777
AIU_ER_F	0,1976	0,1260	0,0354
AIU_ER_R	0,2771	0,3726	0,0083
AIU_FVG_B	0,1840	0,1980	0,0811
AIU_FVG_P	0,2898	0,2275	0,0085
AIU_FVG_TS	0,1251	0,0139	0,0998
AIU_FVG_TV	0,2318	0,2593	0,0225
AIU_LAZ_NE	0,1967	0,1539	0,0137
AIU_LAZ_NO	0,1197	0,0551	0,0254
AIU_LAZ_SE	0,0887	0,1355	0,1274
AIU_LAZ_SO	0,1658	0,0872	0,0676
AIU_LCE_B	0,1231	0,1779	0,1165
AIU_LCE_BN	0,1880	0,2156	0,0164
AIU_LCE_BS	0,1361	0,0367	0,0412
AIU_LCE_M	0,1674	0,1884	0,0759
AIU_LIG_L	0,1201	0,1345	0,1043
AIU_LIG_P	0,1565	0,1456	0,0909
AIU_LN_C	0,0349	0,0000	0,1988
AIU_LN_M	0,1432	0,2362	0,1131
AIU_LO_C	0,0330	0,1316	0,2368
AIU_LO_V	0,1339	0,1533	0,1686
AIU_MAR_A	0,1044	0,1690	0,0646
AIU_MAR_M	0,1023	0,0863	0,1211
AIU_MI_MC	0,2620	0,3716	0,0613
AIU_MI_ML	0,0796	0,0603	0,1474
AIU_MI_MS	0,0516	0,1750	0,1634
AIU_MI_MT	0,3651	0,3138	0,0758

AIU_NA_C	0,1418	0,1287	0,0042
AIU_NA_E	0,0603	0,0910	0,0577
AIU_NA_N	0,1956	0,1076	0,0043
AIU_PIE_A	0,2577	0,1934	0,0203
AIU_PIE_C	0,2324	0,2979	0,0541
AIU_PIE_N	0,0867	0,0344	0,1516
AIU_PUG_B	0,1109	0,3230	0,0548
AIU_PUG_F	0,0873	0,1271	0,1027
AIU_PUG_S	0,0160	0,0026	0,1603
AIU_RM_C	0,1079	0,0050	0,0940
AIU_RM_E	0,0780	0,2053	0,1855
AIU_RM_O	0,1981	0,3253	0,1996
AIU_RM_S	0,0839	0,1983	0,1098
AIU_SAR_N	0,0806	0,1018	0,1226
AIU_SAR_S	0,2164	0,2828	0,0583
AIU_SIE_C	0,0062	0,0678	0,1011
AIU_SIE_M	0,0030	0,1444	0,2198
AIU_SIE_R	0,0082	0,0409	0,1350
AIU_SIO_A	0,0308	0,0745	0,1573
AIU_SIO_PE	0,1766	0,2777	0,1058
AIU_SIO_SO	0,0665	0,1720	0,1216
AIU_TAA_B	0,0907	0,1975	0,1526
AIU_TAA_V	0,3367	0,2867	0,0885
AIU_TOE_C	0,1405	0,1377	0,1404
AIU_TOE_N	0,1987	0,2367	0,0075
AIU_TOE_S	0,1140	0,1416	0,0541
AIU_TOO_C	0,1739	0,1388	0,0425
AIU_TOO_N	0,1388	0,2592	0,0353
AIU_TOO_S	0,2553	0,2662	0,0381
AIU_TOVA_A	0,1194	0,0584	0,1364
AIU_TOVA_TC	0,1322	0,2986	0,1767
AIU_TOVA_TS	0,0761	0,1531	0,0825
AIU_UMB	0,0373	0,0112	0,0873
AIU_VE_P	0,3045	0,2899	0,0987
AIU_VE_VE	0,1770	0,1652	0,0378
AIU_VE_VI	0,1997	0,2054	0,1134

According to Table 26, all locations weight are similar to each other. As it is seen on Table 27 Auto.Arima model provides better results.

CONCLUSION

Conclusion

Before to interpret the results, the demonstration of all results according to MAPE are:

Table 28: Result Distribution according to Total

MAPE	Random Forest	XGB	Auto.Arima
Total	0,1356	0,1540	0,0677

As it is seen on Table 28 the lowest MAPE value is observed on Auto.Arima model which is 6,77% . However Auto.Arima is not always owned the lowest MAPE value in all cases. It only is observed always owned the lower MAPE value in Level I job type. When it is drilled down in job category level II. 5 times the lowest MAPE value is obtained with using Random Forest model and 1 time the lowest MAPE value is obtained with using Extreme Gradient Boosting model.

When the results are inspected as days of week perspective. In some cases the MAPE values are seen dramatically high. When it is drilled down the reasons of these dramatically high MAPE values are those days were weekend days, Saturdays and Sundays. As we mentioned Table 25 Saturdays weight or another word work request owns only 7,7% and Sundays is only 2%.

In Table 25 we understand that week days own the main work request creation (90,3% of all). According to weekdays result as it is seen in Table 26 Extreme Gradient Boosting is obtained two times the lowest MAPE value and Auto.Arima is obtained 5 times.

In Location, according to 74 different locations in Italy we obtain the lowest MAPE value with Extreme Gradient Boosting 13 times, Random Forest 19 times and Auto.Arima 42 times.

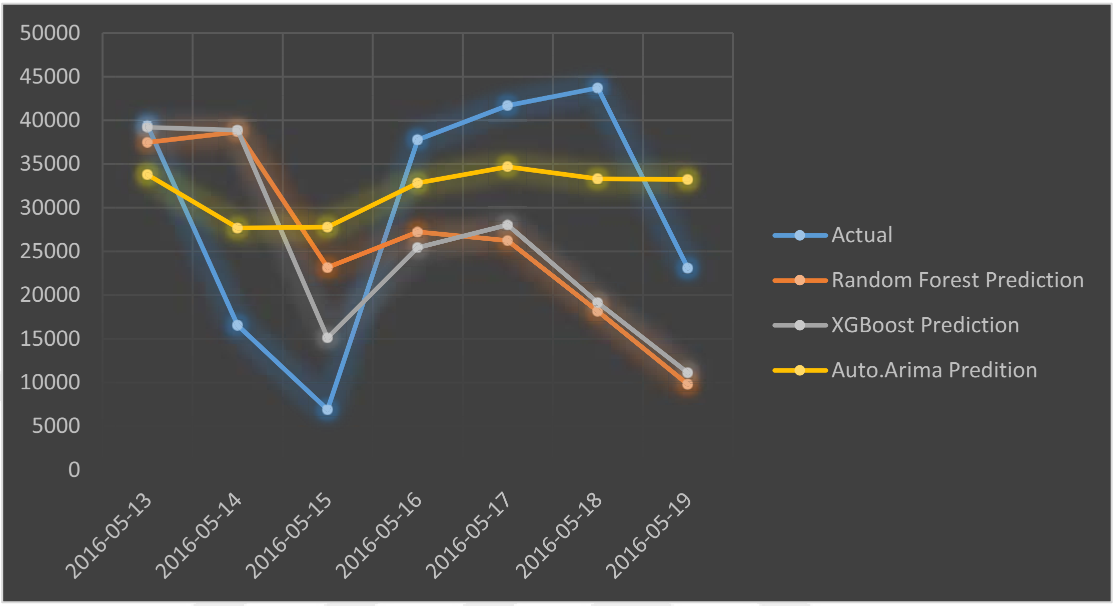


Figure 57: Graph of Result Distribution according to Total

As it is seen in graph 4.1 Auto.Arima model follows the actual closer than other models; however Extreme Gradient Boosting and Random Forest model respond the time based increasing and decreasing better than Auto.Arima model.

Future Works

In our research we learn according to MAPE Auto.Arima model gives lower value if the data set are stationary and decision tree typed regression machine learning methods as Extreme Gradient Boosting and Random Forest own better respond for time based changes like increasing and decreasing. Therefore in future works we would like to research Hyndman's TBATS with regressors [65] as a variable and to assemble Tbats and the best fitted Decision Tree typed regression models, likely to Random Forest to manage weekend days effect better.

[65] Hydnman R. "TBATS with regressors" http://robjhyndman.com/hyndsight/tbats-with-regressors/ ,(20 June 2016)

REFERENCES

Accounting, Financial, Tax, "Qualitative Forecasting Methods And Techniques", http://accounting-financial-tax.com/2009/04/qualitative-forecasting-methods-and-techniques/ (4 May 2016)

Aldor-Noiman S. , Feigin P.D and Mandelbaum A,"Workload Forecasting For A Call Center: Methodology and A Case Study", 2009

Arranz M.A, "Portmanteau Test Statistics in Time Series", http://packages.tol-project.org/docs/ndmtest.pdf , (6 May 2016)

Breiman L,"Random Forests". Machine Learning Vol: 45 (1), pp: 5–32 (2001).

Breiman L and Cutler A, Inside R,"importance {randomForest} Extract variable importance measure" http://www.inside-r.org/packages/cran/randomforest/docs/importance (17 May 2016)

Breiman L and Cutler A,CRAN-R,"Package 'randomForest' " https://cran.r-project.org/web/packages/randomForest/randomForest.pdf (13 May 2016)

Breiman L., Friedman J.H., Olshen R.A. and Stone C.J., "Classification and Regression Trees". Chapman & Hall (1984)

Breiman L.: "Random Forests". Machine Learning Vol: 45 (1), pp: 5–32 (2001).

Business Integration Partners, About Us". https://www.businessintegrationpartners.com/about-us/ (24 July 2016)

Callegaro A, "Forecasting Methods for Spare Parts Demand", (Master Thesis). 2010

Chen T, "Xgboost" https://github.com/tqchen/xgboost. (7 May 2016)

Chen T, CRAN-R "Package 'xgboost' " https://cran.r-project.org/web/packages/xgboost/xgboost.pdf (18 May 2016)

Cromwell, Jeff B.; et al. . . "Multivariate Tests for Time Series Models". SAGE Publications. ISBN 0-8039-5440-9. (1994)

Dudek G. "Short-Term Load Forecasting using Random Forests, Department of Electrical Engineering", Czestochowa University of Technology 2011

Duke University, ""Identifying the numbers of AR or MA terms in an ARIMA model"", http://people.duke.edu/~rnau/411arim3.htm , (5 May 2016)

Duke University, "Introduction to ARIMA", http://people.duke.edu/~rnau/411arim.htm , (4 May 2016)

Sánchez-López E, Pérez-Linares C., Figueroa-Saavedra F and Barreras-Serrano A ,"A Short Term Forecast For Mexican Imports Of United States Beef Using A Univariate Time Series Model", México 2015

European Commission For Learn, "Qualitative vs Quantitative Statistics", forlearn.jrc.ec.europa.eu/guide/4_methodology/meth_quanti-quali.htm , (4 May 2016)

Francis X. Diebold, "Forecasting in Economics", Business, Finance and Beyond, pg 4. Edition 2015 Version Monday 14th December, 2015

Friedman J.H. "Stochastic gradient boosting", Computational Statistics and Data Analysis, pg 367–378, 2002

Hastie T., "Tibshirani R. and Friedman J."The Elements of Statistical Learning. Data Mining, Inference, and Prediction",Springer, 2009

Horrocks et al., 2010.ClimAID, 2011

Horrocks et al.,2010; Ofcom, 2010

Hydnman R. "TBATS with regressors" http://robjhyndman.com/hyndsight/tbats-with-regressors/ , (20 June 2016)

Hyndman R. J, Khandakar Y. "Automatic Time Series Forecasting: The forecast Package for R", Journal of Statistical Software, July 2008, Volume 27, Issue 3.

Hyndman R.J, Athanasopoulos G, "Forecasting: Principles And Practice", 2016, https://www.otexts.org/fpp/1/ (4 May 2016)

"IHMC, "Components of a time series"
http://cmapskm.ihmc.us/rid=1052458821502_1749267941_6906/components.pdf
,(4 May 2016)
"

INC.com,"Forecasting", http://www.inc.com/encyclopedia/forecasting.html, (01 May 2016)

Inside R, "Auto Arima: Fit best ARIMA model to univariate time series", http://www.inside-r.org/packages/cran/forecast/docs/Auto.Arima , (6 May 2016)

Jacobusse G., "Winning Model Documentation describing my solution for the Kaggle competition: Rossmann Store Sales" https://www.kaggle.com/c/rossmann-store-sales/forums/t/18024/model-documentation-1st-place ,(7 May 2016)

Jain A., Menon M N. and Chandra, "S. Sales Forecasting for Retail Chains", 2015

Judge G G, Griffiths W. E, Hill R. C; Lee T. C, "The Theory and Practice of Econometrics" New York: Wiley. pp. 637–660. ISBN 0-471-05938-2, 1980

Kaggle, "Otto Group Product Classification Challenge", https://www.kaggle.com/c/otto-group-product-classification-challenge ,(7 May 2016)

Kaggle, "Rossmann Store Sales competition" https://www.kaggle.com/c/rossmann-store-sales, (10 May 2016)

Kaggle, "Winning Model Documentation describing my solution for the Kaggle competition Rossmann Store Sales" : https://kaggle2.blob.core.windows.net/forum-message-attachments/102102/3454/Rossmann_nr1_doc.pdf?sv=2012-02-12&se=2016-05-17T16%3A55%3A37Z&sr=b&sp=r&sig=tU5oqVEQGyjP97%2FLupW57DIfRQu2GxdHFvOd0ewZggs%3D, (11 May 2016)

Kane M.J, price N, Scotch M and Rabinowitz P, "Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks", , BMC Bioinformatics2014, Kane et al.; licensee BioMed Central Ltd. 2014

KNIME, "Knime GroupBy Node Description" https://www.knime.org/files/nodedetails/_manipulation_row_row_transform_GroupBy.html , (9 May 2016)

KNIME, "Lag Column Description", https://www.knime.org/files/nodedetails/_manipulation_column_column_transform_Lag_Column.html , (12 May 2016)

KNIME, "Loop and flow variable", https://tech.knime.org/forum/knime-users/loop-and-flow-variable , (20 May 2016)

KNIME, ""R snipped Knime Description", https://www.knime.org/files/nodedetails/_labs_r_interactive_R_Snippet.html ,(11 May 2016)

Li P. Box-Cox transformation Box-Cox Transformations: An Overview, http://www.ime.usp.br/~abe/lista/pdfm9cJKUmFZp.pdf , (5 May 2016)

Lippi M.,Bertini M, and Frasconi P, "Short-Term Traffic Flow Forecasting: An Experimental Comparison of Time-Series Analysis and Supervised Learning" , 2012
Mighty Mechanical, "Forecasting Fundamentals", http://mech.at.ua/Forecasting.pdf ,(4 May 2016)

NASA, "Weather Forecasting Through The Ages", http://earthobservatory.nasa.gov/Features/WxForecasting/wx2.php , (01 May 2016)

NC State University, "Hu S. Akaike Information Criterion" http://www4.ncsu.edu/~shu3/Presentation/AIC.pdf , (6 May 2016)

New York University, ""White Noise and Moving Average Mode" http://people.stern.nyu.edu/churvich/Forecasting/Handouts/Chapt3.1.pdf , (6 May 2016)

Pasapitch C. P, Kerdprasop N, and Kerdprasop K, "Time Series Analysis of Household Electric Consumption with ARIMA and ARMA Models", (2013)

Pradeep K. S. and Rajesh K., "Demand Forecasting For Sales of Milk Product (Paneer) In Chhattisgarh International Journal of Inventive Engineering and Sciences" (IJIES) ISSN: 2319–9598, Volume-1, Issue-9, (August 2013)

Raicharoen T, Lursinsap C. and Sanguanbhoki P., “Application of critical support vector machine to time series prediction”, Circuits and Systems, 2003. **ISCAS ’03.Proceedings of the 2003 International Symposium on Volume** 5, 25-28 May, 2003, pp: V-741-V-744"

Rotela Junior P, Salomon F.L.R, Pamplona E.O; ARIMA: An Applied Time Series Forecasting Model for the Bovespa Stock Index, Institute of Production Engineering and Management, Federal University of Itajuba, Itajuba, Brazil (2014)

RTMath, ""Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test"" https://www.rtmath.net/help/html/695835bf-570e-411f-9d76-05ee2570d0d7.htm (12 May 2016)

SAS Institute, "Notation for ARIMA Models". Time Series Forecasting System. (4 May 2016)

Simmhan Y, Aman S, Kumbhare A, Liu R, Stevens S, Zhou Q and Prasanna; "Cloud-Based Software Platform For Data-Driven Smart Grid Management", V, University of Southern California, Los Angeles, USA, (2013)

Time Series, http://www.investopedia.com/terms/t/timeseries.asp , (5 May 2016)

Wikipedia, "Data Feature Engineering Description", https://en.wikipedia.org/wiki/Feature_engineering, (9 May 2016)

Wikipedia, "Kaggle.com Description”, https://en.wikipedia.org/wiki/Kaggle , (9 May 2016)

Wikipedia, "Knime Description", https://en.wikipedia.org/wiki/KNIME (9 May 2016)

APPENDICES

APPENDIX 1: Auto.Arima R code sample

```
#Load the Packages
library(zoo)
library(forecast)
library(fpp)

datAIUtput <- knime.in


# CREATE Time Series
datAIUtput$"dateString" <- as.Date(datAIUtput$"dateString", "%Y-%m-%d",
tz="UTC")
date.min <- as.Date("2014-01-01", "%Y-%m-%d",  tz="UTC")
date.max <- as.Date("2015-12-31", "%Y-%m-%d", tz="UTC")

date.min2<-as.Date("2016-01-01", "%Y-%m-%d",  tz="UTC")
date.max2<-as.Date("2016-05-12", "%Y-%m-%d",  tz="UTC")
dtest <- datAIUtput[datAIUtput$"dateString" >= as.Date("2016-05-13"),]
datAIUtput$"Sum(WRs)"[datAIUtput$"Sum(WRs)"<0] <- 0

y_inb <- msts(datAIUtput[datAIUtput$"dateString">=date.min &
datAIUtput$"dateString" <= date.max,ncol(datAIUtput)],
seasonal.periods=c(365.25), start=c(2014,1,1),ts.frequency=365.25)
d_inb<-msts(datAIUtput[datAIUtput$"dateString">=date.min2 &
datAIUtput$"dateString" <= date.max2,ncol(datAIUtput)],
seasonal.periods=c(365.25), start=c(2014,1,1),ts.frequency=365.25)

#Evaluate the ARIMA Model for every combiantion
arimaModel <- Auto.Arima(y_inb)
arimaModel <- Arima(d_inb,model=arimaModel)

resultArima <- forecast(arimaModel, h=7)
a<-substr(resultArima$"method",7,7)#taken the AR value
AR<- as.numeric(a)#Convert AR value string to Integer
BoxTest<-Box.test(residuals(arimaModel), lag=AR, fitdf=0, type="Ljung")
#Evaluate Ljung-BoxTest
BoxPvalue<- replicate(7,BoxTest$p.value)#replicate Plvalue 30 times to fit
dim(BoxPvalue)
dtest$"BoxTestPvalue"<- BoxPvalue
method<- replicate(7,resultArima$"method",)
ArimaMean <- resultArima$"mean"

if (is.na(BoxTest$p.value)==TRUE){
  dtest$"Warnings" <-replicate(7,"variables is not good for forecast")
}else if (BoxTest$p.value<0.05 )
{
  dtest$"Warnings"<- replicate(7,"p value is lower H0 is not acceptable, change the model")

}else
{
  dtest$"Warnings"<-replicate(7,"p value is ok")
}
dtest$"ArimaPredict" <- ArimaMean
dtest$"Arima_Method"<-method
dtest<-dtest[,1:16]

dtest$"dateString"<-as.character(as.Date(dtest$"dateString"))

knime.out <-dtest #Print the Results
```

APPENDIX 2: Random Forest R Code Sample:

```
#Load Packages
library(randomForest)

set.seed(123)

#Create Time Series

knime.in$"Date and time" <- as.Date(knime.in$"Date and time", format = "
%Y-%m-%d")
knime.in$"Number of Holidays in Next Week"[is.na(knime.in$"Number of
Holidays in Next Week")] <- 0
#knime.in[362:364,4]<-0
dtrain <- knime.in[knime.in$"Date and time" < as.Date("2016-01-01"),]
dtest <- knime.in[knime.in$"Date and time" < as.Date("2016-05-13") &
knime.in$"Date and time" > as.Date("2015-12-31"),]
dfinal <-knime.in[knime.in$"Date and time" >= as.Date("2016-05-13"),]


#Change the variables name
names(dtrain)<-
c("Year","Week_of_year","Number_of_Holidays_in_This_Week","Number_
of_Holidays_in_Last_Week","Number_of_Holidays_in_Next_Week","LEVE
L 1 CATEGORY","LEVEL 2
CATEGORY","AIU","Date_and_time","Quarter","Month","Day_of_month",
"Day_of_week","Day_of_year","Sum_WRs_","Sum_WRs_1_","Sum_WRs_
2_","Sum_WRs_3","Sum_WRs_4","Sum_WRs_5","Sum_WRs_6","Sum_W
Rs_7","Sum_WRs_8","Sum_WRs_9","Sum_WRs_10","Sum_WRs_11","Su
m_WRs_12","Sum_WRs_13","Sum_WRs_14" , "Festivity" , "Festivita_1" ,
"Festivita_2","Festivita_3" , "Days_Since_Last_Festivity" ,
"Sum_Last_Month_WRs","precipMM","precipMM_1","precipMM_2","preci
pMM_3","precipMM_4","precipMM_5","precipMM_6","precipMM_7")
names(dtest)<-
c("Year","Week_of_year","Number_of_Holidays_in_This_Week","Number_
of_Holidays_in_Last_Week","Number_of_Holidays_in_Next_Week","LEVE
L 1 CATEGORY","LEVEL 2
CATEGORY","AIU","Date_and_time","Quarter","Month","Day_of_month",
"Day_of_week","Day_of_year","Sum_WRs_","Sum_WRs_1_","Sum_WRs_
2_","Sum_WRs_3","Sum_WRs_4","Sum_WRs_5","Sum_WRs_6","Sum_W
Rs_7","Sum_WRs_8","Sum_WRs_9","Sum_WRs_10","Sum_WRs_11","Su
m_WRs_12","Sum_WRs_13","Sum_WRs_14" , "Festivity" , "Festivita_1" ,
"Festivita_2","Festivita_3" , "Days_Since_Last_Festivity" ,
"Sum_Last_Month_WRs","precipMM","precipMM_1","precipMM_2","preci
pMM_3","precipMM_4","precipMM_5","precipMM_6","precipMM_7")
names(dfinal)<-
c("Year","Week_of_year","Number_of_Holidays_in_This_Week","Number_
of_Holidays_in_Last_Week","Number_of_Holidays_in_Next_Week","LEVE
L 1 CATEGORY","LEVEL 2
CATEGORY","AIU","Date_and_time","Quarter","Month","Day_of_month",
"Day_of_week","Day_of_year","Sum_WRs_","Sum_WRs_1_","Sum_WRs_
2_","Sum_WRs_3","Sum_WRs_4","Sum_WRs_5","Sum_WRs_6","Sum_W
Rs_7","Sum_WRs_8","Sum_WRs_9","Sum_WRs_10","Sum_WRs_11","Su
m_WRs_12","Sum_WRs_13","Sum_WRs_14" , "Festivity" , "Festivita_1" ,
"Festivita_2","Festivita_3" , "Days_Since_Last_Festivity" ,
"Sum_Last_Month_WRs","precipMM","precipMM_1","precipMM_2","preci
pMM_3","precipMM_4","precipMM_5","precipMM_6","precipMM_7")

#Apply Random Forest Model

rf <- randomForest(Sum_WRs_ ~ Sum_WRs_1_ + Sum_WRs_2_ +
          Sum_WRs_3 + Sum_WRs_4 +
          Sum_WRs_5 + Sum_WRs_6 +
          Sum_WRs_7 + Sum_WRs_8 + Sum_WRs_9 +
          Sum_WRs_10 + Sum_WRs_11+ Sum_WRs_12 +
          Sum_WRs_13 + Sum_WRs_14 + Day_of_week +
Days_Since_Last_Festivity +
          Festivity + Festivita_1 + Festivita_2 + Festivita_3 +
Sum_Last_Month_WRs + Number_of_Holidays_in_This_Week +
 Number_of_Holidays_in_Last_Week +
Number_of_Holidays_in_Next_Week+precipMM +
+precipMM_1+precipMM_2+precipMM_3+precipMM_4+precipMM_5+pre
cipMM_6+precipMM_7,
          data = dtrain, importance = TRUE)


dfinal$"Prediction_Sum_WRs_Random_Forest_November" <-
data.frame(predict(rf,dtest))[1:7,]

#Print the Results
knime.out <- dfinal
```

APPENDIX 3: Extreme Gradient Boosting R Code Sample:

```
#Load the Packages

library(xgboost)

set.seed(123)


#Create Time Series
knime.in$"Date and time" <- as.Date(knime.in$"Date and time", format =
"%Y-%m-%d")
knime.in$"Number of Holidays in Next Week"[is.na(knime.in$"Number of
Holidays in Next Week")] <- 0

dtrain <- knime.in[knime.in$"Date and time" < as.Date("2016-01-01"),]
dtest <- knime.in[knime.in$"Date and time" < as.Date("2016-05-13") &
knime.in$"Date and time" > as.Date("2016-01-01"),]
dfinal <-knime.in[knime.in$"Date and time" >= as.Date("2016-05-13"),]


Date_Time<-dfinal$"Date and time"

#Delete the string and irrelevant variables
dtrain$"Week of year" <- NULL
dtrain$"LEVEL 1 CATEGORY" <- NULL
dtrain$"LEVEL 2 CATEGORY"  <- NULL
dtrain$"AIU" <- NULL
dtrain$"Date and time" <- NULL
dtrain$"Year" <- NULL
dtrain$"Quarter" <- NULL
dtrain$"Month" <- NULL
dtrain$"Day of month" <- NULL
dtrain$"Day of year" <- NULL

dtest$"Week of year" <- NULL
dtest$"LEVEL 1 CATEGORY" <- NULL
dtest$"LEVEL 2 CATEGORY"  <- NULL
dtest$"AIU" <- NULL
dtest$"Date and time" <- NULL
dtest$"Year" <- NULL
dtest$"Quarter" <- NULL
dtest$"Month" <- NULL
dtest$"Day of month" <- NULL
dtest$"Day of year" <- NULL

dfinal$"Week of year" <- NULL
dfinal$"LEVEL 1 CATEGORY" <- NULL
```

```
dfinal$"LEVEL 2 CATEGORY"  <- NULL
dfinal$"AIU" <- NULL
dfinal$"Date and time" <- NULL
dfinal$"Year" <- NULL
dfinal$"Quarter" <- NULL
dfinal$"Month" <- NULL
dfinal$"Day of month" <- NULL
dfinal$"Day of year" <- NULL


#Create XGB Model
dtrainmatrix <- xgb.DMatrix(data = as.matrix(dtrain[, !(colnames(dtrain) ==
"Sum(WRs)")]), label = dtrain $"Sum(WRs)")#updated part
bst <- xgboost(data = dtrainmatrix, max.depth = 2, eta = 1, nthread = 4, nround
= 100, verbose = 0)
test.pred.XBG <- predict(bst,as.matrix(dtest[, !(colnames(dtest) ==
"Sum(WRs)")]))#updated part

dummy<-data.frame(test.pred.XBG)[1:7,]

dfinal$"Prediction (Sum(WRs))XGBoost November" <- dummy

#Print the Model
knime.out <- data.frame(dfinal,Date_Time)
```