International Journal of Database Theory and Application
Vol.10, No.8 (2017), pp.1-10
http://dx.doi.org/10.14257/ijdta.2017.10.8.01

A Study on the Visualizing Time Series Data Using R

Eunmi Jung[1], Andrew G. Kim[2] and Hyenki Kim[3*]

Dept. of Multimedia Engineering, Andong National University
*[1]jeilc@naver.com, [2]wgeoner@appsol.kr, [3]*hkkim@anu.ac.kr*

Abstract

With the recent increase in data volume, there is a growing interest in Big Data technology and there is also a growing interest in techniques to visualize result of big data processing. The vast majority of people accept visual information more quickly than text. Therefore, visualization is the important thing to focus on regarding big data analysis. Therefore, the study examined various visualization methods using an open source statistical analysis software R program. The study explored a method to configure data sets and a method to implement various graphs according to visualization method using R to determine patterns in data and understand the characteristics of data at a glance through visualization of data. Through this, it was possible to determine characteristics of data that were not known only through simple regression analysis and through showing that rather than interpreting data as it is, it could be visualized in various methods through conversion of data sets, it is expected that it will help users to make various decisions.

Keywords: *Big Data, R, Visualization, Time Series Data*

1. Introduction

Due to the rapid development of Internet, mobile technology, and Internet of things, the digitalization of industry overall is proceeding at a rapid pace. This change in industrial environment requires diverse and rapid response such as introduction of new business models or evolution into the digital age for the survival of companies[1].

According to Forrester Research report, predictive analytics was selected as a Top 10 Big Data technology to lead the next decade. Predictive analysis shows a trend of stable growth over the next 10 years and has been evaluated as a technology with high business value. It was evaluated as the best solution that improves business performance through company evaluation in the market and marketing optimization which also provides methods to reduce risk. It has been selected as technology that provides the highest business value as a field providing solutions providing future market forecasting models through big data analysis[2].

Over the past few decades, the amount of data has increased exponentially in many forms. This increase in data makes it difficult to find the desired information from the data[3]. Also, most people accept visual information more quickly than text. Therefore, the thing to focus on regarding big data analysis using public data is visualization[4].

Sales data is a time series data. Time-series data shows changes over time and is useful in analyzing trends. A method of visualizing by specific items through grouping cannot be expressed simply in R program. Therefore, the study examined various visualization methods using R program, an open source statistical analysis software, including methods of configuring data sets according to visualization method to help effective and quick decision-making in users and various visualization methods. Through this, it is possible to

* Corresponding Author

ISSN: 2005-4270 IJDTA

explore various characteristics of time series data and it can help users make various decisions.

2. Related Research

With the recent increase in the amount of data, there has been an increase of interest in big data and people accept visual information more quickly than text. One of the most popular big data analysis tools is open source based R. R programming language is a programming language for data analysis and is specialized for statistical processing and visualization. It provides various statistical model and processing techniques such as linear and nonlinear modeling, time series analysis, classification, and clustering. It provides a simple and effective programming language with effective data processing and storage capabilities, it can easily create charts containing mathematical symbols and formulas through graphics technology, and it has excellent expandability[5]. Because R is compatible with other development languages and can process in real time through linking with the web, it is a useful tool in developing new applications and providing Web services with economic benefits from cost savings. Open source project for R operates package distribution repository CRAN (the Comprehensive R Archive Network: http://cran.r-project.org) so that users can create application packages based on R and two support free distribution. Over 6000 packages can be downloaded through this and it is also available on Windows, Mac OS, and Lenox[6].

3. Time Series Data Visualization Method

Visualization of data helps understanding of data. Even if one is not an expert, one can have insight and make decisions more easily. The visualize data, first, data is obtained, second, data is transformed into data to be visualized, third, data is visualized, and as a last step, it is embellished using visualization option.

The data used in the study was two years of data from a restaurant in city center and as explanatory variable, only environmental factor was considered and sales was predicted using regression analysis by using newspaper articles, stock quotes, weather information, and holiday information. The basic statistics used in the experiment are shown in Table 1.

Table 1. Basic Statistic

Min	1st Qu.	Median	Mean	3rd Qu.	Max	NA's
276000	622000	764000	791800	923200	1820000	19

Figure 1 shows the actual data in a linear graph.

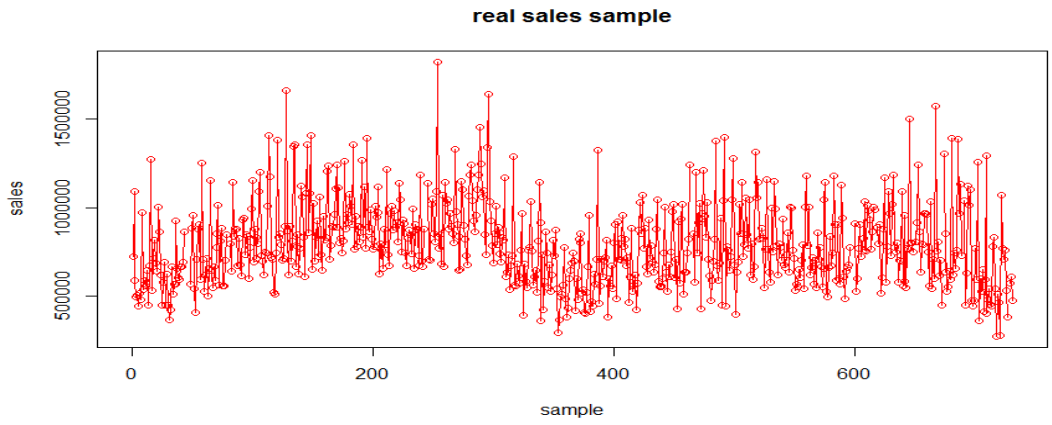


Figure 1. Real Sales Sample Plot

Figure 2 shows the predicted values obtained by regression analysis as a linear graph.

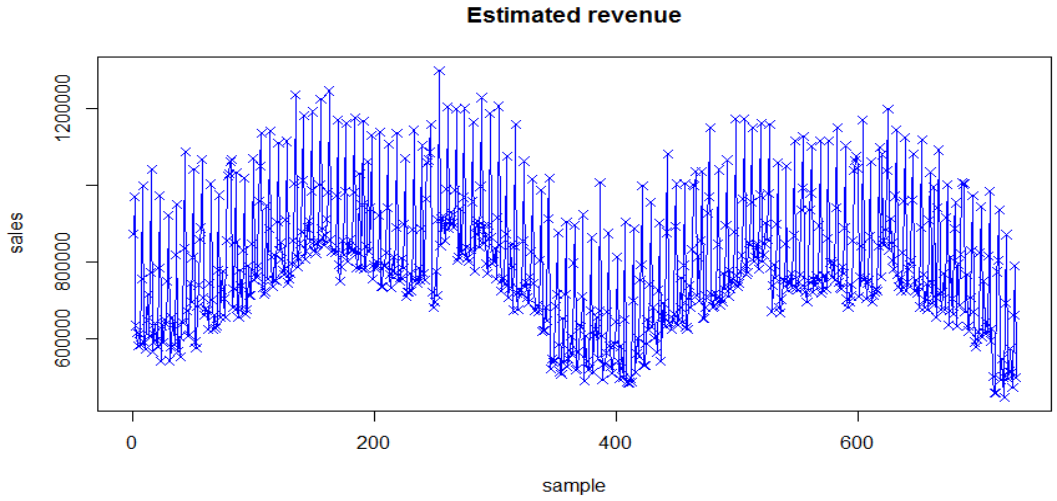


Figure 2. Estimated Revenue Plot

Figure 3 shows the overlapping of two graphs to confirm prediction accuracy.

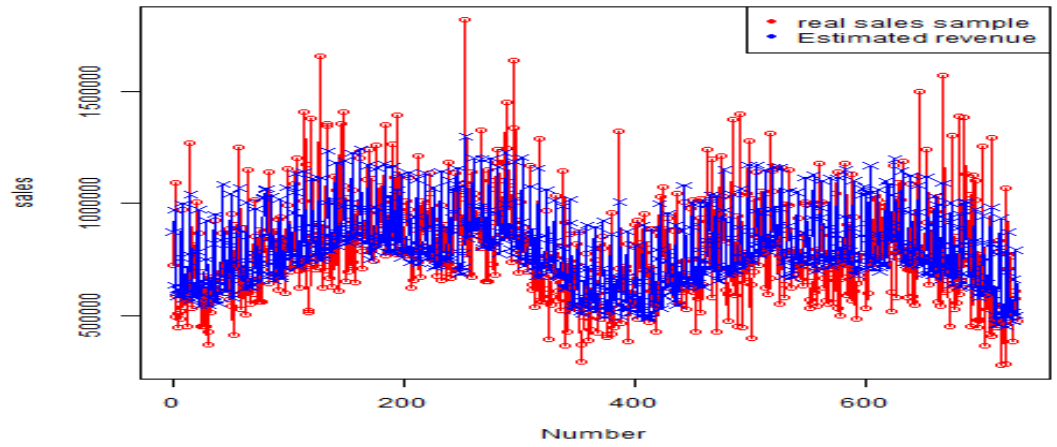


Figure 3. Plot for Overlaying the Actual Value and the Predicted Value

It can be seen that the portion where the outlier exists is not well predicted. Figures 1 through 3 can be easily expressed using the plot () function of R.

Figure 4 shows the R script code for overlaying the actual value and the predicted value. When drawing the chart by adding other data on the chart drawn using Plot () function, a series can be added using function point().

```
predict$id <- seq(1, nrow(predict), 1)
png("Plot.png")
plot(x = predict$id, y = predict$sales, ylab = "sales", xlab ="Number",type="o", cex = 1, col = "#FF0000")
points(x = predict$id, y = predict$pred_sales, type = "o", cex = 1, col = "#0000FF", pch=4)  #
legend("topright", legend=c("real sales sample", "Estimated revenue"), pch = c(20, 20), col = c("red", "blue"))
dev.off()
```

Figure 4. R Script Code for Overlaying the Actual Value and the Predicted Value

Grouping by month and days of the week using aggregate() function to draw the summarized graph, the data is summarized as shown in Figure 5. The 731 records in ① in Figure 5 has been summarized into 84 records as shown in ②. This change to the data set is required to express the graph using ggplot2 package.

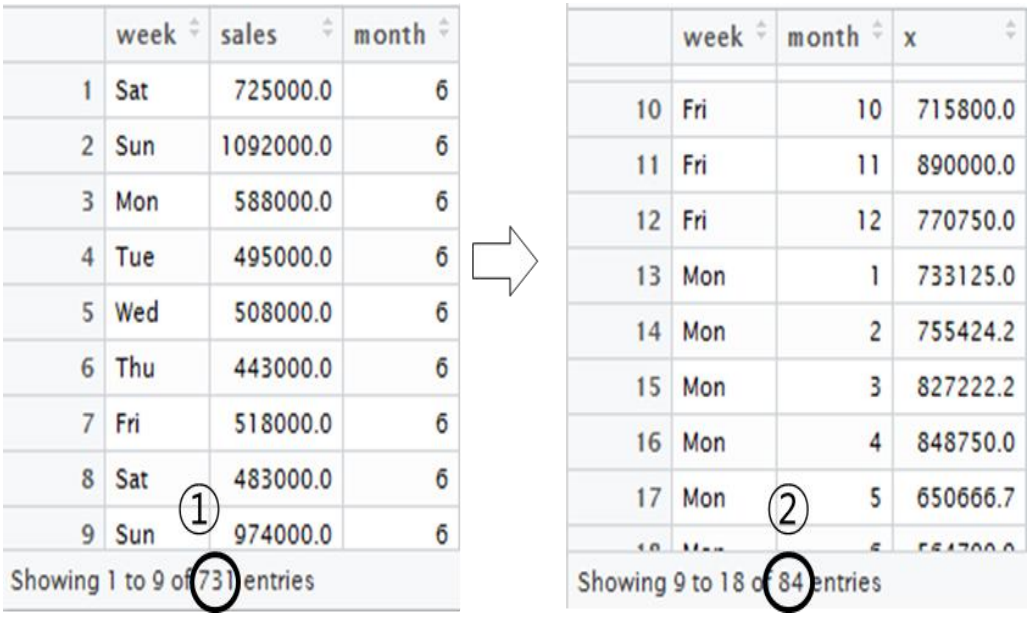


Figure 5. Change Dataset for Summary Graph

Figure 6 is a flowchart showing data averages by month and days of the week as a stacked bar graph. By performing average of data by month and days of the week, sales information by month and days of the week is determined and a technique to look at monthly sales trend is explored through the visualization process. First, to make aggregation by groups, when reading time series data, the field with the character values is read using the 'stringsAsFactors=FALSE' option to recognize it as a string vector instead of a factor. The next step is to extract only the fields needed to aggregate by group the read data to configure as a subset. Because statistical functions cannot be used if missing values exist in the subset, there needs to be processing on missing values. Missing value data is replaced by the average of the entire data excluding the missing data. After loading 'plyr' library to use the function for group aggregation, aggregate() function is used to obtain averages by month and days of the week. After calculating the average for each group, sorting is done by month and days of the week. 'ggplot2' package necessary for drawing charts is installed, 'ggplot2' library is loaded, and visualization is completed using 'ggplot' function.

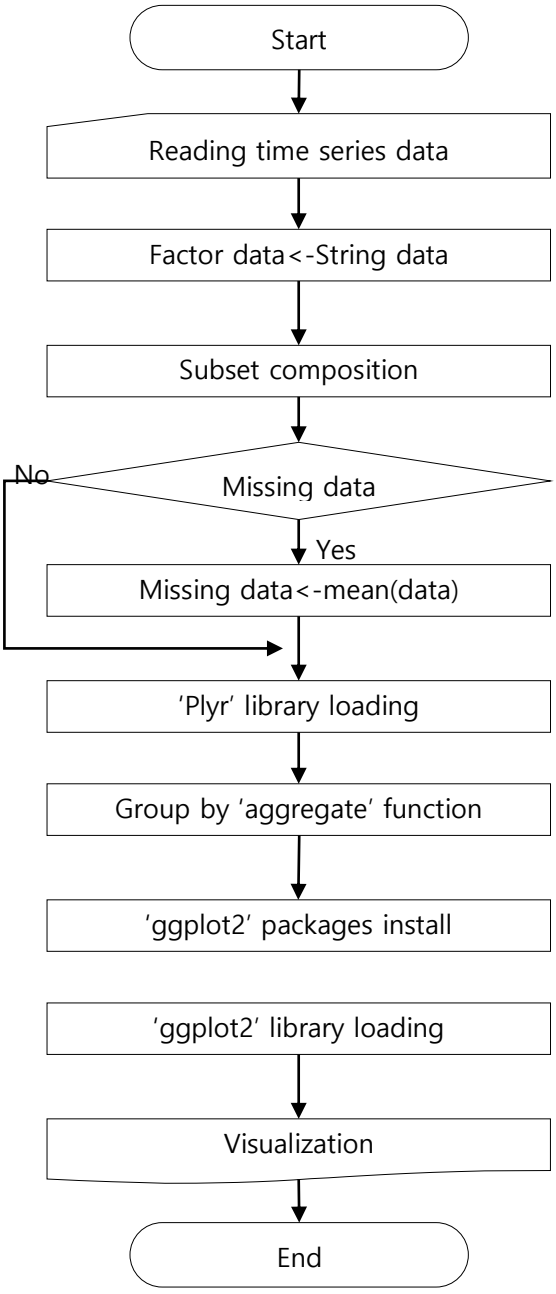


Figure 6. Flowchart for Drawing a Summary Graph

Figure 7 shows the cumulative graph of the average of data by month and days of the week. Through this graph, the monthly earnings trend can be seen at a glance. It can be seen that there is poor performance between May and August and good performance between March and April, and October and December.

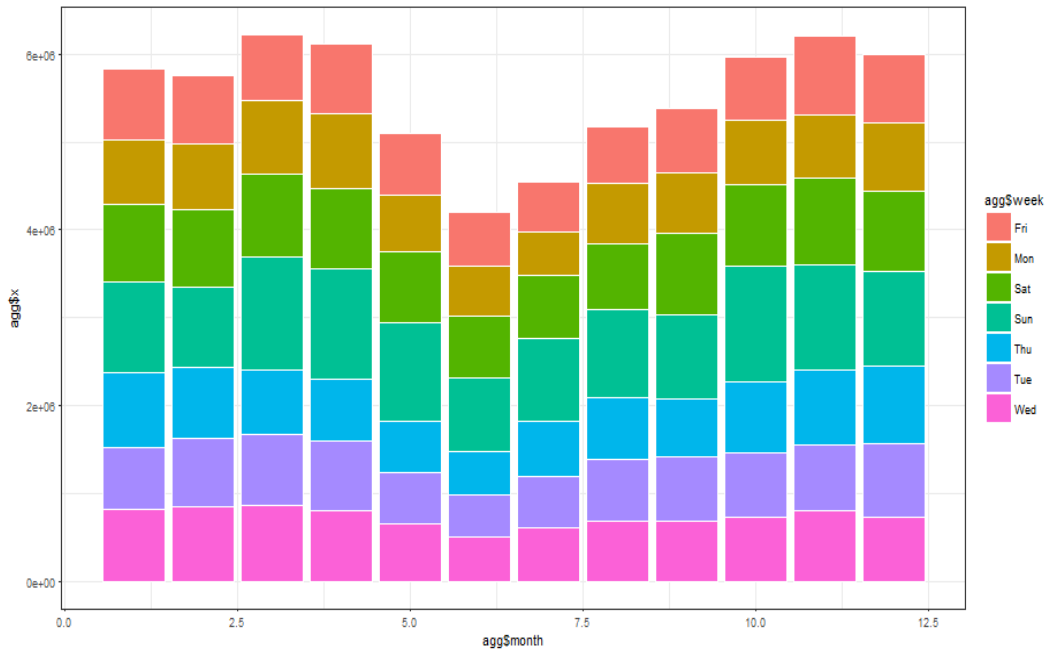


Figure 7. Average Cumulative Graph by Summary

Figure 8 arranged side-by-side the stacked bar graph by month and days of the week by month to visualize it to make it possible to easily determine which days of the week had good performance by month. Through visualization, it can be easily determined which days of the week had good performance along with monthly earnings trend.

Figure 8. Arranged Side-by-side the Stacked Bar Graph

Figure 9 shows the R script code for visualizing the sales average by month and days of the week. aggregate() function was used to calculate the sales average by month and days of the week. Among the function arguments, FUN=mean represents using averages for aggregation by group. Through arrange() function, the results obtained through

aggregate() was sorted by month and days of the week. To visualize this, ggplot2 package was used. ggplot is an expanded version of plot which can express data more aesthetically and with more detail. geo_bar() function is a function to create graphs with bar graphs and color="white" specifies the border color of the bar. Bar graphs can be drawn side-by-side using option position="dodge".

```
#group by
sales_Info <- read.csv("dst_data_frame.csv",stringsAsFactors = FALSE)
select.column <- c('week','sales','month')
sub_data<-subset(sales_Info,select = select.column,Na.rm=TRUE)
#NA
sub_data$sales[is.na(sub_data$sales)]<-mean(sub_data$sales,na.rm = TRUE)#NA
library(plyr)
agg<-aggregate(sub_data$sales,by=list(week=sub_data$week,month=sub_data$month),FUN=mean)
#sort
agg<-arrange(agg,agg$week,agg$month)

install.packages("ggplot2")
library(ggplot2)
#-----------------------------------------------------
plt<-ggplot(agg,aes(x=agg$month,y=agg$x,fill=agg$week))
plt+theme_bw()+geom_bar(stat='identity',colour='white')
   +scale_x_discrete('month',labels=agg$month)+ylab('sales')+scale_fill_discrete(name=c("week"))
write.csv(agg,"agg.csv")
#-----------------------------------------------------------
pt<-ggplot(agg,aes(x=agg$month,y=agg$x,fill=agg$week))
pt+geom_bar(stat='identity',colour='white',position="dodge")+scale_x_discrete('month',labels=agg$month)
      +ylab('sales')+scale_fill_discrete(name=c("week"))
```

Figure 9. R Script Code for Summary Visualization

Existing data set cannot be used directly to compare actual and predicted values by item. By running BarPlot using existing data set, the actual value and the predicted value are displayed in divided areas as shown in Figure 10. When it is outputted like this, actual value and predictive value by date cannot be compared. Therefore, the data set must be changed. The data set data frame must be changed into a matrix, changed to a transposed matrix, and then change the form back into a matrix. A graph as shown in Figure 10 is shown when BarPlot is executed with data set shown in ① of Figure 12, and when BarPlot is executed but transposing to data set shown in ② of Figure 12, it can be outputted like Figure 11 so that the two items can be compared. To draw the graph, only 10% of the data is extracted and used without using the entire data. Because items can be compared by date, production results can be seen at a glance.

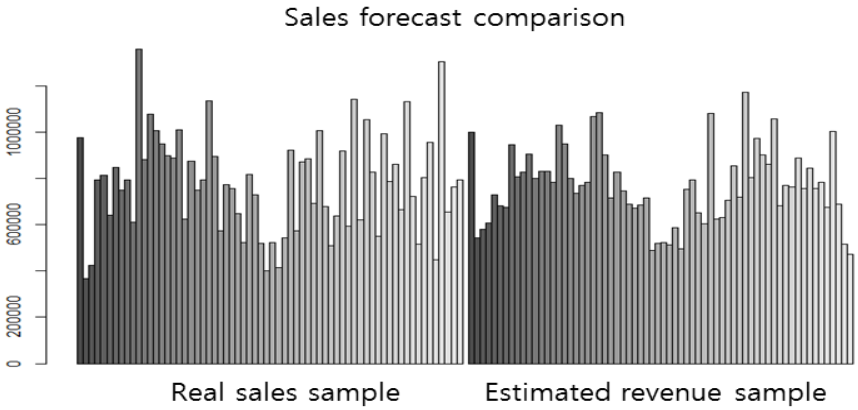


Figure 10. Graph before Data Set Change

Figure 11 shows a bar graph to compare actual and predicted values. To express like Figure 11, the data set must be changed like ② in Figure 12.

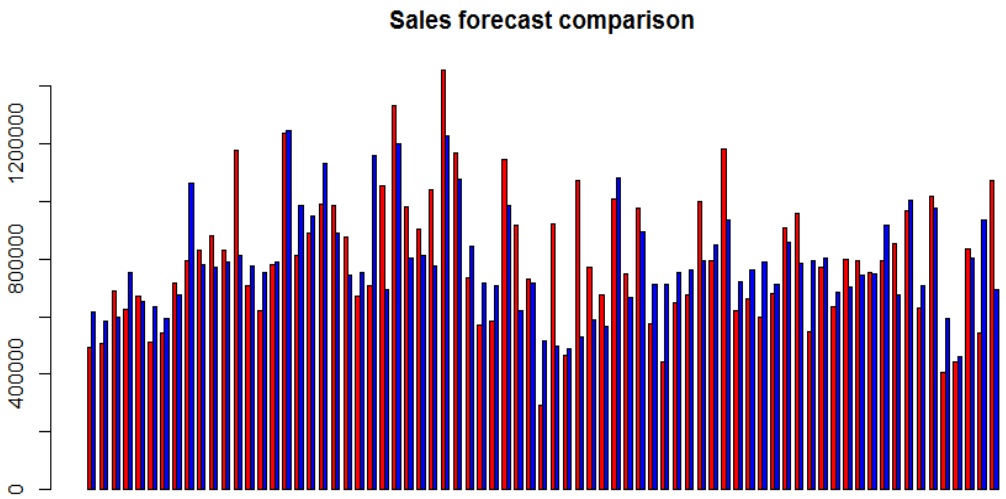


Figure 11. Bar Graph for Comparing Actual and Predicted Values

Figure 12 shows newly processed data set to compare by item as shown in the chart in Figure 11. ① In Figure 12 shows 10% of data frame changed to matrix and ② of Figure 12 represents checking of the status of data frame before the change into matrix. ③ Of Figure 12 shows a data set reconfigured into matrix through transposing ①.

①

	sales	pred_sales
1	495000.0	615686.3
2	508000.0	582533.1
3	689000.0	597016.4
4	625000.0	750762.5
5	669000.0	654500.1
6	512000.0	634882.0
7	541000.0	591622.0
8	714000.0	673700.6
9	791817.4	1060312.3
10	828000.0	777816.6

②

```
> str(df)
'data.frame':   75 obs. of  2 variables:
 $ sales     : num  495000 508000 689000 625000 669000 ...
 $ pred_sales: num  615686 582533 597016 750763 654500 ...
```

③

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11
sales	495000.0	508000.0	689000.0	625000.0	669000.0	512000	541000	714000.0	791817.4	828000.0	881000.0
pred_sales	615686.3	582533.1	597016.4	750762.5	654500.1	634882	591622	673700.6	1060312.3	777816.6	769288.8

Figure 12. Change Data Set for Bar Graph to Compare Actual and Predicted Value

Figure 13 shows the R script code for representing the data set configuration and visualization to compare the actual value with the predicted value after extracting 10% of the data.


```
#bar plot
predict <- read.csv("predict.csv")
predict$sales[is.na(predict$sales)]<-mean(predict$sales,na.rm = TRUE)
index <- sample(2, nrow(predict), replace = TRUE, prob = c(0.9, 0.1))
predict.sample <- predict[index==2,]
#x<-predict$X
sales<-predict.sample$sales
pred_sales<-predict.sample$pred_sales
str(sales)
df<-data.frame(sales=sales,pred_sales=pred_sales)
str(df)
f<-as.matrix(df)
f<-t(f)#전치행렬
f<-as.matrix(f)

barplot(f,main="Sales forecast comparison",beside=T,col=c("red","blue"))
```

Figure 13. R Script Code for Representing the Data Set Configuration and Visualization

4. Conclusion

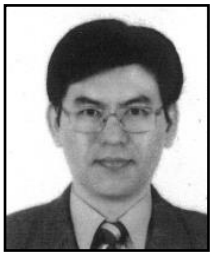
The study explored methods of configuring data sets according to visualization method and implemented various graphs using R to enable seeing the characteristics of data at a glance through determining patterns through visualization of data. The study was able to determine characteristics of data that were not seen through simple regression analysis results, and explored drawing graphs in various method through conversion of data sets rather than only interpreting data as it is. Therefore, through data set configuration through visualization method and various visualization methods, the study expects that it will help effective visualization and user decision-making.

References

[1] E. Jung and H. Kim, "A Study On The Factors To Influence The Estimated Outcome", Far East Journal of Electronics and Communications, vol. 2, **(2016)**, pp. 19-24.
[2] K-ICT Big Data Center Editor, "BigData BiMonthly", National Information Society Agency Publishers, Korea, vol. 21, **(2016)**.
[3] K. S. Lee, S. Lee, S. Kang, C. Park and J. Kim, "Design and Implementation of Text Visualization Tools for Analyzing Big Data in Logistics Industry", Journal of Information Technology and Architecture, vol. 13, no. 2, **(2016)**, pp. 355-365.
[4] G.-S. Choe, Y.-G. Ham and S.-H. Kim, "Big Data Visualization", Journal of The Korean Society of Computer and Information, vol. 21, no. 1, **(2013)**, pp. 33-43.
[5] K.-S. Kim and K.-W. Lee, "A Web Application for Open Data Visualization Using R", Journal of Korean Association of Geographic Information Studies, vol. 17, no. 3, **(2014)**, pp. 72-81.
[6] J. H. Lee and H.-K. Lee, "A Study on unstructured text mining algorithm through R programming based on data dictionary", Journal of the Korea Industrial Information System Research, vol. 20, no. 2, **(2015)**, pp. 113-124.

Authors

EunMi Jung, February 2009, Andong National University, Computer Science, Master of Engineering. February 2009, Andong National University, Information and Communication Technology, Dr. completion. March 2009~Now, Andong National University, Multimedia Engineering, Foreign instructors. Interests: Big Data, Image Processing, Natural Language Processing



Andrew G. Kim, he received the B.S. degree in business administration from Yeungnam University, Korea, 1996. He joined Appsol.kr inc. in 2013, where he is currently a CEO at Appsol.kr inc. in Korea. His research interests include mobile application and Multimedia app.

Hyenki Kim, he received the B.S. and M.S. degree in electronics engineering from Kyungpook National University, Korea, 1986 and 1988 respectively. He received Ph. D. in electronics engineering from Kyungpook National University, Korea, 2000. He joined Andong National University in 2002, where he is currently a professor at Dept. of multimedia engineering in Korea. His research interests include multimedia system, mobile app. and Multimedia communication.