

Prediction of Call Arrival Pattern

Karthi.V



Introduction

Even though there is an enormous amount written about forecasting, the number of articles about call center forecasting is not very impressive. The articles capture a reasonable amount of mathematical forecasting models, but the many undiscovered models can be applied to enhance the accuracy of forecasting. Great-scale research will be needed to broaden the knowledge about this subject. In this Document a complete literature overview of different forecasting methods is provided, with a particular focus on the forecasting methods that are expounded and utilized for predicting daily call frequencies in call centers.

Forecasting Models

In general, there are several methods to be used for forecasting time series. However, not all the forecasting methods are applied to forecast the call frequencies. In this Document, an overview of the models will be presented. For each method, the model will be given together with the accompanying literature in which the models were used.

* ARIMA Model
* Dynamic Regression Model
* Exponential Smoothing
* State Space Model
* Multiple Linear Regression Analysis.
* Winter-Holts Method

ARIMA Model

ARIMA is the abbreviation for Autoregressive Integrated Moving Average. The ARIMA model is a widely used forecasting model invented by Box and Jenkins. The basis of the ARIMA model is the ARMA model which consists of two sorts of terms: The auto regressive terms (AR) and the Moving Average (MA) terms

Autoregressive (AR) terms: The AR terms are lagged values of the dependent variable, and serve as independent variables in the model. The general autoregressive model is given by

Here, and are unknown parameters .The Process is White noise with property that =0 for all k ≥1.So the repressors are exogenous with k=1,…….As the time series is observed for t=1,……,n, the p-lagged explanatory Variable is available only from time t= onwards.This Model is Called an autoregressive model of order , also written as AR

**Moving Average (MA) terms**: The MA terms are lagged values of the errors between past actual values and their predicted values who also serve as independent variables .A general Moving Average process is given by

Where is white noise. Here and… are unknown parameters. This process is always stationary, with mean *µ=E[]=α. This model also known as a moving average model of order q, Which can be written as* ***MA(****q).*

Exponential Smoothing State Space Model

Introduction

Forecasting practitioners have long realized the potential of the **ARIMA** methodology for modeling time series data with Strong seasonal patterns. In fact, most articles on forecasting call volumes in call center refer to the ARIMA Model. J.Nijdam (*Forecasting telecommunications services using Box-Jenkins (ARIMA) models. Telecommunication Journal of Australia*) illustrated the effectiveness of ARIMA models for predicting monthly telephone traffic in the presence of a persistent seasonal Pattern. In the most recent article, W.Xu (Weidong Xu. Long range planning for call centers at FedEx. The Journal of Business Forecasting Methods and Systems, 18(4):7–11, Winter 1999/2000) a **forecasting specialist** in the Forecasting /Modeling group in Worldwide Customer Service Strategic Planning & Analysis at FedEx explained that ARIMA is one of the forecasting models used in **R**, their major forecasting software.

**Dynamic Regression Model**

The ARIMA model discussed in the previous section deals with single time series and does not allow the inclusion of other information in the models and forecasts. However, frequently other information may be used to aid in forecasting time series. Information about holidays, Marketing launches, changes in the process, or other external variables may be of use in assisting the development of more accurate forecasts.

An example of this type of useful information is the influence of Marketing/advertising campaigns. Often the effect such of information does not show up in the forecast variable () immediately, but is divided across several time periods. For instance, the effect of an Marketing/advertising campaign persists for some time after the end of the campaign.

**Exponential Smoothing**

In the late 1950s exponential smoothing methods were developed for the first time by operations researchers. Since the development of this concept, it became a widely used method, partly due to its simplicity and low costs.

There are several exponential smoothing methods. The major ones used are Single Exponential Smoothing, Holt’s Linear Model (1957) and Holt-Winters’ Trend and Seasonality Model.

**Single Exponential Smoothing**

The simplest form of exponential smoothing is single exponential smoothing, and can only be used for data **without any** **systematic trend or seasonal components**. Given such a time series, a logical approach is to take a weighted average of past values. Exponential Smoothing assigns exponentially decreasing weights as the observation get older. In other words, recent observations are given relatively more weight in forecasting than the older observations.

In the case of moving averages, the weights assigned to the observations are the same and are equal to 1/*N*. In exponential smoothing, however, there are one or more *smoothing* *parameters* to be determined (or estimated) and these choices determine the weights assigned to the observations.

Let’s look at Single Exponential Smoothing first. Recall that = observed value at time ‘t’ and = forecasted value at time ‘t’ .the general expression is

Which says, forecast for the next period = \* Today value + \* Previous forecast

The forecast for the current period is a weighted average of all past observations. The weight given to past observations declines exponentially. The larger the, more weight is given to recent observations.

So you can see here that the exponentially smoothed moving average takes into account all of the previous observations, compare the moving average, above where only a few of the previous observations were taken into account. Simple exponential smoothing is an extension of weighted moving averages where the greatest weight is placed on the most recent value and then progressively smaller weights are placed on the older values.

**Double Exponential Smoothing (Holt’s Method)**

Exponential smoothing is useful when there is no trend. However if the data is trending, we need to use the Double Exponential Smoothing method. This situation can be improved by the introduction of a second equation with a second constant, **β**, the trend component, which must be chosen in conjunction with **α**, The mean component. Double exponential smoothing is defined in the following manner:

Where

This method works best when the time series has a positive or negative trend (i.e. upward or downward). After observing the value of the time series at period t (), this method computes an estimate of the base, or expected level of the time series () and the expected rate of increase or decrease per period (). It is customary to assume that = and unless told otherwise, and assume = 0.

The double exponential smoothing method is a little more complicated to build and should give us better result. What happens if the data show trend as well as seasonality? In this case double exponential smoothing will not Work. We need to use the Triple Exponential Smoothing method which will be discussed below.

**Winter-holts Trend and Seasonality**

The exponential smoothing methods examined thus far can deal with almost any type of data as long as they are non-seasonal. Nevertheless, in case of seasonality these methods are not very useful. Holt-winters’ trend and seasonality model can actually manage seasonality.

It decomposes the times series down into three components: level, trend and seasonal components. When an actual observation is divided by its corresponding seasonal factor, it is said to be “**deseasonalized**.” (i.e. the seasonal component has been removed.) This allows us to make meaningful comparisons across time periods.

Winter-Holt’s model was extended by Double Exponential Smoothing to include seasonality. The Holt-Winter’s method is based on three smoothing equations: one for the level, one for the trend, and one for seasonality. It is similar to Holt’s model. The only difference is that an equation is added to deal with seasonality. In fact there are two different Holt-Winters’ methods. It depends on whether seasonality is modeled in an additive or multiplicative way. The choice of which to use depends on the characteristics of the specific time series.

Frequency on TBATS Function

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

The relevant formulas for this method follow.

Let = Seasonal Factor for Period-‘t’

Let **c** = the number of periods in a cycle (**12** if months of year, **7** if days of week …)

where is another smoothing constant between 0 and 1. This means that Holt Winters' smoothing is similar to exponential smoothing if and =0. It will be similar to double exponential smoothing if = 0.

**Winter’s Multiplicative Model**

The below formula gives the Holt-Winter’s multiplicative function

where denotes the level of the time series at time t and represents the trend at time t. In this equation, the seasonal component is given by , with **c** as the length of the seasonal period, and the forecast for n periods ahead. For monthly data (c= 12)

**Winter’s Additive Model**

The additive model is slightly different. Instead of multiplying the Holt model by the seasonal component, the seasonal factor is added. The formula will then be given by

Besides the difference mentioned before, the smoothing equations of and also differ from those used in the multiplicative model. For perfection all the update equations are shown beneath.

In only a few articles about forecasting call volumes at call centers, the exponential smoothing methods are discussed. In the article of W. Xu (Weidong Xu. Long range planning for call centers at FedEx. The Journal of Business Forecasting Methods and Systems, 18(4):7–11, Winter 1999/2000) she declared that exponential smoothing was one of the forecasting techniques implemented in Fedex’s major forecasting software system (SAS). As described before Bianchi, Jarrett, and Hanumara (*Forecasting incoming calls to telemarketing centers. The Journal of Business Forecasting Methods & Systems, 12(2):3–12,1993*) report in their article, that AT&T Bell Laboratories used an adaptation of Holt-Winters forecasting model with its telemarketing scheduling system, called NAMES, for forecasting incoming calls to telemarketing centers. Nevertheless, in their study to evaluate the current use of the Holt-Winters model for forecasting as done by the NAMES system and indicate whether improvement is possible through the use of ARIMA time series modeling. They found **ARIMA** models with intervention analysis to perform better for the time series studied.

Frequency on TBATS Function

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

**Exponential Smoothing State Space Model**

An innovations state space modeling framework is introduced for forecasting complex seasonal time series such as those with multiple seasonal periods, high frequency seasonality, non-integer seasonality and dual-calendar effects. The new approaches offer alternatives to traditional counterparts, providing several advantages and additional options. A key feature of the proposed trigonometric framework is its ability to model both linear and non-linear time series with single seasonality, multiple seasonality, high period seasonality, non-integer seasonality and dual calendar effects.

In addition, the framework consists of a new estimation procedure which is sufficiently general to be applied to any innovations state space model. By relying on **maximum likelihood estimation**, it avoids the ad hoc startup choices with unknown statistical properties commonly used with exponential smoothing. By incorporating the least-squares criterion, it streamlines the process of obtaining the maximum likelihood estimates. The applications of the proposed modeling framework to three complex seasonal time series demonstrated that the trigonometric models led to a better out of sample performance with substantially fewer values to be estimated than traditional seasonal exponential smoothing approaches The trigonometric approach was also illustrated as a means of decomposing complex seasonal time series.

A further advantage of the proposed framework is its adaptability. It can be altered to encompass various deterministic effects that are often seen in real life time series. For instance, the moving holidays such as Easter and irregular holidays can be handled by incorporating dummy variables in the models, and the varying length of months can be managed by adjusting the data for trading days before modeling. The framework can also be adapted to handle data with zero and negative values. The use of a Box-Cox transformation limits our approach to positive time series, as is often encountered in complex seasonal time series.

**Regression Analysis**

Another method used for forecasting call frequencies in call centers is regression analysis. A major advantage of the regression analysis is it is easier to understand. A regression analysis of time series is often categorized into three effects: the seasonal effect, the trend effect and the random effect.

When time series are measured per quarter, per month, or even per day, they may contain seasonal variation. The seasonal component in a time series refers to patterns that are repeated over a one-year period and that average out in the long run. The patterns that do not average out are included in the constant and trend components of the model. Whereas the trend is of dominant importance in long-term forecasting, the seasonal component is very significant in short-term forecasting as it is often the main source of short-run fluctuations. Frequently, seasonal effects can be detected from plots of the time series, and also from plots of the seasonal series that consist of the observations in the same month or quarter over different years.

The components can be added-up or multiplied. The models obtained are respectively called the additive model and the multiplicative model. Both models are stated beneath

Multiplicative Model

Additive Model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

Where denotes the factor determined by the influence of the season at time ’k’, designates the factor determined by the influence of the trend at time’ k’ and designates the factor determined by the effect of the random component at time k. It is possible to extend the equation. For instance, it is possible to include the daily effects, such as say the day of the week effect. Because time series are divided into several components, textual documentation on the subject often refers to it as time series decomposition. Of these pre-mentioned models the multiplicative model is the most widely applied, and often gives better predicting values than the additive one.

It is possible to adopt the mentioned models in combination with linear regression. Before this model is given, first the theory of linear regression will be expounded concisely. The most simple form of linear regression is simple linear regression. Simple linear regression is used in situations to evaluate the linear relationship between two variables. One example is the relationship between bank wages and education. In other words, simple linear regression is used to develop an equation by which we can predict or estimate a dependent variable (Salary), given an independent variable (education). The simple regression equation is given by

Where (Salary) as the dependent variable, and as the intercept. The variable is the slope of the line and x is the independent variable (education). Sometimes it is needed to include more than one independent variable. For example, one might find out that if the variables education and Experience are both included as independent variables, the prediction of the dependent variable bank wages is much better. The term multiple regression denotes a number of independent variables that are included. The general formula of multiple regression is as follows

As mentioned before there is a possibility to apply decomposition of time series in combination with linear regression. If an additive time series decomposition is combined with multiple regression the result is given by the next equation

Articles about the prediction of call center call volumes show that regression analysis is not frequently implemented method. In 1998, Klungle [*Call center forecasting at AAA Michigan. The Journal of Business Forecasting, 20(4):8–13, 1998*.] tried to forecast the number of incoming calls for the Emergency Road Service. They found this number to vary at different times of the day significantly during winter and spring seasons. They also tried to model the incoming calls with the Holt-Winter’s method and a neural network method, but found that the regression analysis method performed best. Antipov and Meade [*Forecasting call frequency at a financial services call centre. Journal of the Operational Research Society, 53(9):953–960, 2002.]* developed a forecasting model for the number of daily applications for loans at a financial service telephone call center. They built a regression analysis model with a dynamic level, multiplicative calendar effects and a multiplicative advertising response. It was shown to be more effective than an ARIMA model, which was used as a benchmark.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

**Solving Pain Points of Forecasting**

Everyone knows that the volume of work arriving in a call center is quite random. If we look at a history of work arriving in a typical call center, this appears to be true. The volume of work does indeed vary from one interval to the next, day to day, andweek to week – facts which may give the impression that accurately forecasting futurework is an impossible task.

There are two simple effects of getting the forecast wrong and both cost you money.

***1. Forecasting too high***

Overstaffing creates a scenario for idle, unproductive agents to suffer from low morale or become bored or distracted and not focused on customer service.

***2. Forecasting too low***

Understaffing can result in lost customers due to abandoned calls or poor customer service. It should also be remembered that both of the above could occur on the same day if the forecasted distribution of work is incorrect.

**Four common problems involved with producing an accurate forecast**

**Using averages**:Forecasting an average is a safe bet (and the easiest to perform) but is unlikely to be the most accurate. Very little that happens in a contact center is an average. What happens next week is unlikely to be an average of what happened over the last few weeks.

**The forecasting tool lacks enough data**:Generally speaking, the more data the forecasting tool has to work with, the greater chance of producing an accurate forecast. If the forecasting tool cannot process more than a few weeks of data, its accuracy will be compromised. A good rule of thumb is *the more data, the* *better*.

**Having unrealistic expectations**:The forecasting tool’s predictions can be based only on what has happened historically and on what it is told will happen in the future. It can never know more than this! This may sound obvious, but don’t expect an accurate forecast for the coming year if you have only a few weeks’ data to forecast from.

**Not understanding how your forecasting tool works regarding:**

* How much data it can store/use
* If it takes into account inflation due to abandoned calls or any other factors
* If it recognizes seasonal and growth trends
* If special event information can be input and correlation factors applied
* How all of this is accomplished

A poor forecast can result in high staffing costs and lost customer revenue, but forecast accuracy depends on many factors. The key is ensuring that the forecasting tool has as much information about what happened in the past and what you expect in the future, and that it will allow you to properly utilize it.

An additional consideration for ensuring the forecaster has sufficient information is the quality of that information. The nature of the data is important. Validate your historical data by comparing new incoming data against a previously validated set of historical data.

**Critical components for accurate forecasting**

Ensuring your forecast takes these bullet points into consideration can help solve your forecasting pain points.

* The amount of historical data available
* The nature of the data
* The forecasting period
* Algorithms that reflect real life customer behavior
* Special events are treated differently, i.e., mail drops, campaigns, and special promotions can be quantified

**Correlating events**

In the example of mail or catalog drops, similar events may occur on several occasions, but will affect work differently based on number of letters delivered. The system must have the capacity to identify and appropriately weight these events to plan for future occurrences.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

**R-Codes**

**Seasonal ARIMA Model**

**Package required:** forecast,fpp, moments,nortest, e1071

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

final\_data=ts(data$**Volume**,frequency=Days)

**# Identifying Seasonality**

library(fpp)

fit=tbats(final\_data)

seasonal=!is.null(fit$seasonal)

seasonal

**# Identifying Number of differences required for a stationary series**

library(forecast)

nsdiffs(final\_data, m=frequency(final\_data), test=c("ocsb","ch"), max.D=6)

**# Build The ARIMA Model**

library(forecast)

model=auto.arima(final\_data,seasonal=TRUE,stepwise=TRUE,approximation=FALSE, trace=TRUE,D=1,max.P = 6, max.Q = 6)

**# Diagnosing Check**

tsdiag(model)

tsdisplay(residuals(model))

Box.test(residuals(model), lag=36, fitdf=8, type="Ljung")

**# Normality Test**

qqnorm(residuals(model))

qqline(residuals(model))

**# Histogram**

colors=c("red","yellow","green","violet","orange","blue","pink","cyan")

h=hist(residuals(model), density=90, col=colors, xlab="Residuals", main="Histogram of Residuals")

xfit=seq(min(residuals(model)),max(residuals(model)),length=50)

yfit=dnorm(xfit,mean=mean(residuals(model)),sd=sd(residuals(model)))

yfit=yfit\*diff(h$mids[1:2])\*length(residuals(model))

lines(xfit, yfit, col="blue", lwd=2)

**# Statistical Test for Normality**

library(moments)

library(nortest)

library(e1071)

skewness(residuals(model))

kurtosis(residuals(model))

shapiro.test(residuals(model))

ks.test(residuals(model),"pnorm",mean(residuals(model)),sqrt(var(residuals(model))))

ad.test(residuals(model))

**# Forecasting using Best-ARIMA Model without drift**

forecast=predict(model,n.ahead=**156**) forecat\_values=data.frame(cbind(forecast$pred))

forecat\_values

**# Forecasting using Best-ARIMA Model with drift**

forecast=predict(model,n.ahead=**156**,newxreg=(noobs+1):(noobs+156))

forecat\_values=data.frame(cbind(forecast$pred))

forecat\_values

**# graphical view of Volume vs Forecast**

plot(final\_data,pch=21,col="blue",lty=3,type="o",xlim=c(0,(length(final\_data)/3)),ylim=c((min(final\_data)),(max(forecat\_values))))

lines(forecast$pred,type="o",col="red",lty=3)

title("Actual vs Forecast")

legend('topleft',c("Actual","Forecast"),col=c('blue','red'), lty=3, bty='n', cex=1)

**Exponential Smoothing State Space Model**

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data with Multiple Seasonality**

library(forecast)

Days=**7**

Year=**365**

Week=round(Year/Days)

Seasonal\_list=c(Days,Year)

final\_data=msts(data$Volume, seasonal.periods= Seasonal\_list)

**# Build TBATS Model**

library(forecast)

fit <- tbats(final\_data,seasonal.periods= Seasonal\_list)

components <- tbats.components(fit)

plot(components,main="Components of The Data")

**# Forecast**

forecast=forecast(fit,h=**62**)

plot(forecast,main="Forecasting using Exponential Smoothing State Space Model")

**Seasonal Frequency**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Types** | **Frequency** | | | | |
| **Minute** | **hour** | **day** | **week** | **year** |
| Daily |  |  |  | **7** | **365\*** |
| Hourly |  |  | **24** | **168** | **8766** |
| Half-​​​​hourly |  |  | **48** | **336** | **17532** |
| Minutes |  | **60** | **1440** | **10080** | **525960** |
| Seconds | **60** | **3600** | **86400** | **604800** | **31557600** |

\*\* The time series data of which one of years is a leap year, then use 365.25.the difference between leap and non-leap years is so small it shouldn’t matter much.

**Winter-Holts Trend and Seasonality**

**Package Reqiured :** forecast,ggplot2,reshape

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

ts\_data=ts(data$Volume,frequency=Days)

**# Displaying The Time series Data**

view\_data=tsdisplay(ts\_data,main="Daily Offered Volumes",pch=1,cex=0.7)

**# Identifing The Seasonal Pattern based on Accuracy**

fit1=HoltWinters(ts\_data,seasonal="additive")

forecast1=forecast.HoltWinters(fit1, h=95)

accuracy(forecast1)

fit2=HoltWinters(ts\_data,seasonal="multiplicative")

forecast2=forecast.HoltWinters(fit2, h=95)

accuracy(forecast2)

**# Identifing The Seasonal Pattern based on Graphical\_View**

aust=window(ts\_data)

fit3=hw(aust,seasonal="additive",h=95)

fit4=hw(aust,seasonal="multiplicative",h=95)

plot(fit4,ylab="Daily Offered Volums",type="l", fcol="white", xlab="")

lines(fitted(fit3),col="red")

lines(fitted(fit4),col="green")

lines(fit3$mean, type="o", col="red")

lines(fit4$mean, type="o", col="green")

legend("topleft",lty=1, pch=1, col=1:3,

c("data","Winters-Holts Additive","Winters-Holts-Multiplicative"))

**# View the Components of the Data**

states=cbind(fit3$model$states[,1:3],fit4$model$states[,1:3])

colnames(states)=c("level","slope","seasonal","level","slope","seasonal")

plot(states)

fit3$model$state[,1:3]

**# Build The Winter-Holts Model**

require(graphics)

library(ggplot2)

library(reshape)

seasonal1=("multiplicative")

Winter\_Holts<-function(ts\_object, n.ahead=4, CI=.95, error.ribbon='green', line.size=1)

{

hw\_object<-HoltWinters(ts\_object,seasonal=seasonal1)

forecast<-predict(hw\_object,n.ahead=n.ahead,prediction.interval=T,level=CI)

for\_values<-data.frame(time=round(time(forecast), 3), value\_forecast=as.data.frame(forecast)$fit, dev=as.data.frame(forecast)$upr-as.data.frame(forecast)$fit)

fitted\_values<-data.frame(time=round(time(hw\_object$fitted), 3), value\_fitted=as.data.frame(hw\_object$fitted)$xhat)

actual\_values<-data.frame(time=round(time(hw\_object$x), 3), Actual=c(hw\_object$x))

graphset<-merge(actual\_values, fitted\_values, by='time', all=TRUE)

graphset<-merge(graphset, for\_values, all=TRUE, by='time')

graphset[is.na(graphset$dev), ]$dev<-0

graphset$Fitted<-c(rep(NA, NROW(graphset)-(NROW(for\_values) + NROW(fitted\_values))), fitted\_values$value\_fitted, for\_values$value\_forecast)

graphset.melt<-melt(graphset[, c('time', 'Actual', 'Fitted')], id='time')

p<-ggplot(graphset.melt, aes(x=time, y=value)) + geom\_ribbon(data=graphset, aes(x=time, y=Fitted, ymin=Fitted-dev, ymax=Fitted + dev), alpha=.2, fill=error.ribbon) + geom\_line(aes(colour=variable), size=line.size) + geom\_vline(x=max(actual\_values$time), lty=2) + xlab('Time') + ylab('Value') + theme(legend.position='bottom') + scale\_colour\_hue('')

return(p)

}

**# Forecasting Using Winter-Holts Method**

graph=Winter\_Holts(ts\_data,n.ahead=**93**,CI=.95,error.ribbon='red',line.size=1)

graph

forecast

**Dynamic Linear Regression Model**

**Package Reqiured : dynlm**

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

ts\_data=ts(data$Volume,frequency=Days)

**# Build Dynamic Regression Model**

library(dynlm)

Model=dynlm(final\_data ~ trend(final\_data) + season(final\_data))

summary(Model)

plot(final\_data, type = "l",col="red",lty=4,lwd=1.5)

lines(fitted(Model),col = 3,lty=2,lwd=1)

title("Actual vs Fit")

legend('topleft',c("Actual","Fit"),col=c(2,3), lty=c(3,3), bty='n', cex=0.9,lwd=c(2,2))

**Artificial Neural Network**

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

final\_data=ts(data$**Volume**,frequency=Days)

**# Forecasting using Neural Network**

fit <- nnetar(final\_data)

str(fit)

str(fit$model[[1]])

summary( fit$model[[1]] )

fcast <- forecast(fit,h=156)

plot(fcast)

**Winter-holts Model with Seasonal Adjustment**

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

final\_data=ts(data$Volume,frequency=Days)

**# Forecasts from Holt's method with seasonal adjustment**

data.stl <- stl(final\_data,"periodic")

data.sa <- seasadj(data.stl)

data.fcast <- holt(uk.sa,156)

seasf <- sindexf(data.stl,156)

data.fcast$mean <- uk.fcast$mean + seasf

data.fcast$lower <- uk.fcast$lower + cbind(seasf,seasf)

data.fcast$upper <- uk.fcast$upper + cbind(seasf,seasf)

data.fcast$x <- final\_data

plot(data.fcast,main="Forecasts from Holt's method with seasonal adjustment")

**Seasonal Naïve Method**

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

final\_data=ts(data$Volume,frequency=Days)

**# Naive forecasts**

library(forecast)

forecast=snaive(final\_data,h=**156**)

forecast

plot(forecast)

**Outlier –Detection**

**Package Reqiured :** tsoutliers,expsmooth,fma

**# Reading the Data from External Files**

data=read.csv("**C:\\Users\\vkarth2\\Desktop\\To Send\\Rupesh-Spotify\\US\\Final-9-Dec\\till July'14.csv",**header=TRUE,sep=",")

data

**# Creating Time series Data**

library(forecast)

Days=**7**

final\_data=ts(data$Volume,frequency=Days)

**# Identifying The Outliers**

library(tsoutliers)

library(expsmooth)

library(fma)

outlier=tsoutliers::tso(final\_data,types = c("AO","LS","TC"),maxit.iloop=10)

outlier

plot(outlier)

