Use Google Analytics and R to predict Web Traffic

Juan Antonio Breña Moral
Created: 2007-05-01
Last update: 2007-05-02
Versión: 0.1

Abstract

Commercial Web sites are designed, planned and launched in many cases by Marketing Departments and Advertisement agencies. Web Traffic is a variable very important to measure the impact of different iniciatives and this measure is a mirror of economics results in many markets.

Google Analytics is a powerful tool to analyze / store Web Traffic data, but the information that you can see with the reports that it offers is poor. If you want to increase the possibilities of your Web Traffic Data, it is neccessary to use others tools to get more information. R is a wonderful free stat engine that you can use to analyze data. In this paper I will try to describe how to predict Web Traffic using R and data stored in Google Analytics. Statistical method used in this example is not unique. Data say how to predict.

The basic steps of the model-building process are:

1. Model selection
2. Model fitting
3. Model validation.

In this example I use AR models, Autoregressive Models, due to the conclusion that I have in ACF / PACF analysis. An autoregressive model is simply a linear regression of the current value of the series against one or more prior values of the series. The value of p is called the order of the AR model.

Data Sources

I started to use Google Analytics in 28th of May, 2006 in my personal web site, http://www.juanantonio.info . To start to analyze data stored in With Google Analytics, you have to export data.

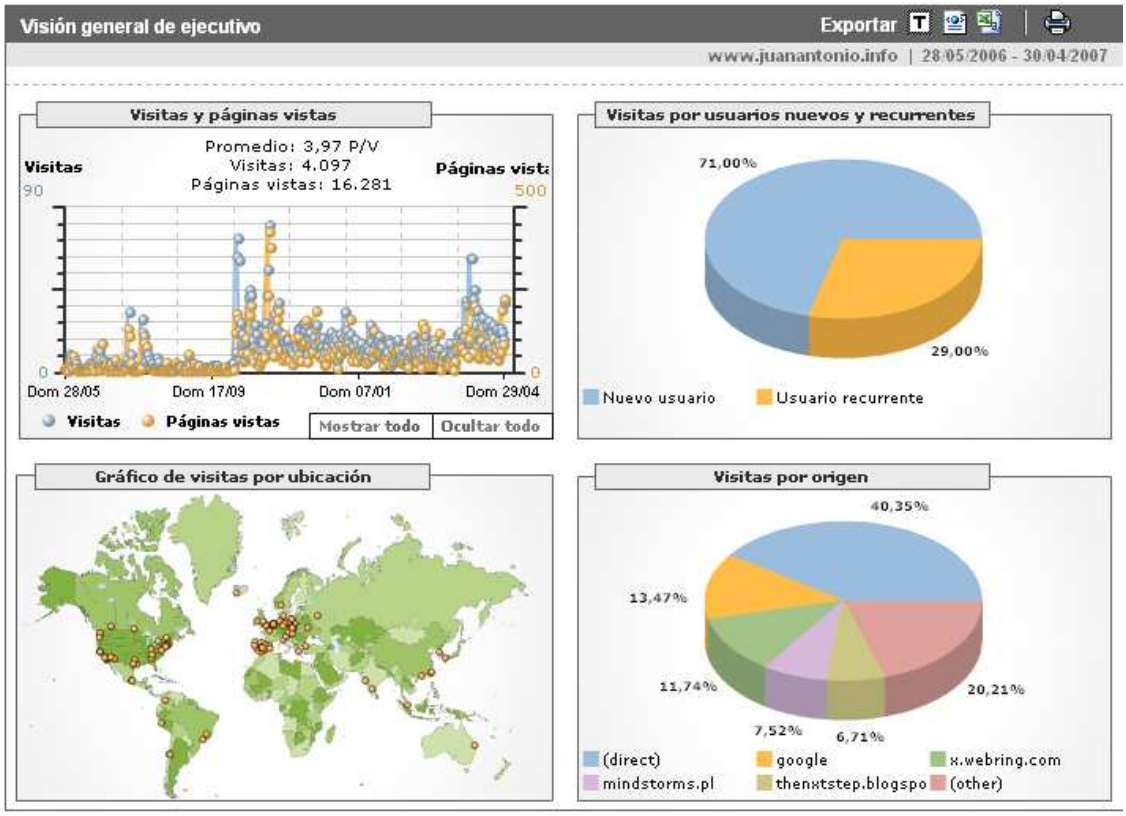

Figure 1: Main Window in Google Analytics

In main window, Google Analytics has export toolbar. It has three differents export options to export. You can export your data in CSV, XML or Excel. XML format is the best way to analyze data. Once you have download your data in your computer you can start to use R to predict your future web Traffic.

R Stat Engine

R is a excellent stat engine but it doesn't have a good GUI. There are a lot of projects developing GUI for R. I use JGR, because in 2005, it won Chambers Awards. Once you have installed JGR or Another GUI for R, woul is the moment to start to code the program to process your data.

To process your Web Traffic Data with R, you will need to be installed into your R instance the following Libraries:

1. Normal Analysis
    a. library("nortest");
2. Time Series Analysis
    a. library("dyn");
    b. library("ArDec");
    c. library("forecast");
    d. library("fBasics");
    e. library("fCalendar");
    f. library("fSeries");
    g. library("tseries");
    h. XML Processing
3. library("XML");

The first step to analyze any data is get data in right format. The script to do it is:

```
#Set Working Directory
setwd("C:/DATOS/ESTADISTICA/R/SCRIPTS/ANALYTICS/");
#Normal Analysis library
library("nortest");
#Time Series analysis libraries
library("dyn");
library("ArDec");
library("forecast");
library("fBasics");
library("fCalendar");
library("fSeries");
library("tseries");
#XML library
library("XML");
#Loading Juan Antonio Breña Moral functions
source("http://www.juanantonio.info/p_research/statistics/r/scripts/eda/JAB.EDA.txt");
source("http://www.juanantonio.info/p_research/statistics/r/scripts/ts/JAB.TS.VIEW.txt")
;

#############
# LOAD XML #
#############

XML_DOC <-  xmlTreeParse("DATA/report.xml");
XML_DOC_ROOT <- xmlRoot(XML_DOC);
#XML_DOC_ROOT;
#Node used to analyze
DATA <- XML_DOC_ROOT[[4]];
#<date>28/05/2006 - 30/04/2007</date>
DATE_STUDY <- XML_DOC_ROOT[[3]];
#DATE_STUDY

################
# PROCESS XML #
################

#CREATE ARRAY
DATE_TRAFFIC <- c();
VISITS <- c();
PAGES <- c();
UNTIL <- xmlSize(DATA)

for (i in seq(from=6, to=UNTIL, by=1)){
    DATE_TRAFFIC <- c(DATE_TRAFFIC, xmlValue(DATA[[i]][[1]]));
    VISITS <- c(VISITS, as.numeric(xmlValue(DATA[[i]][[2]])));
    PAGES <- c(PAGES, as.numeric(xmlValue(DATA[[i]][[3]])));
}
```

Now you have in R your data in a Array Structure. It is the moment to analyze Data

Forecast your visits using R

To start any forecast process it is necessary to follow a methodology. In this case, the steps to make a predictions are:

1. Model selection
2. Model fitting
3. Model validation.

To make a model selection, I will plot data to see the form of my Web Traffic Data:

```
names(DATOS.TS) <- DATE_TRAFFIC
DATOS.TS <- ts(VISITS,start=5, frequency = frequency(VISITS), names=DATE_TRAFFIC)
TITLE <- "www.juanantonio.info visit evolution";
```

```
X_LABEL = "Days"
Y_LABEL = "Visits per day"
plot(DATOS.TS,main = TITLE, xlab=X_LABEL, ylab=Y_LABEL);
```

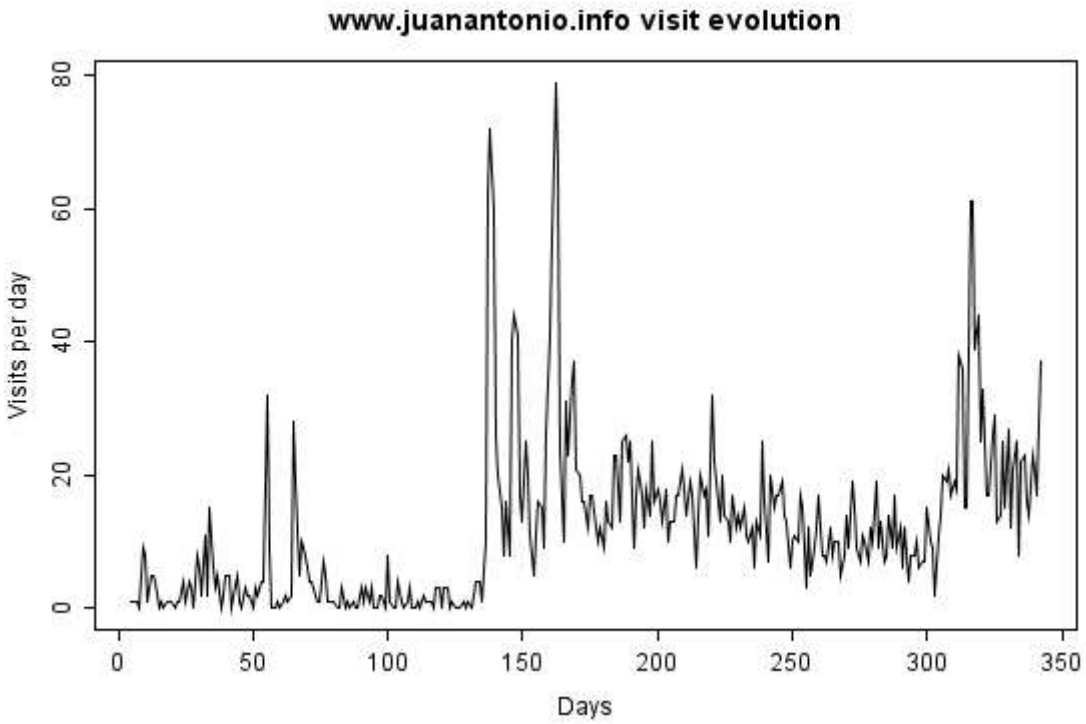

Figure 2: Web Traffic Evolution in juanantonio.info Web Site
period: 28/05/2006 - 30/04/200

Besides I will do a Exploratory Data Analysis, EDA:

```
JAB.EDA(DATOS.TS);
```

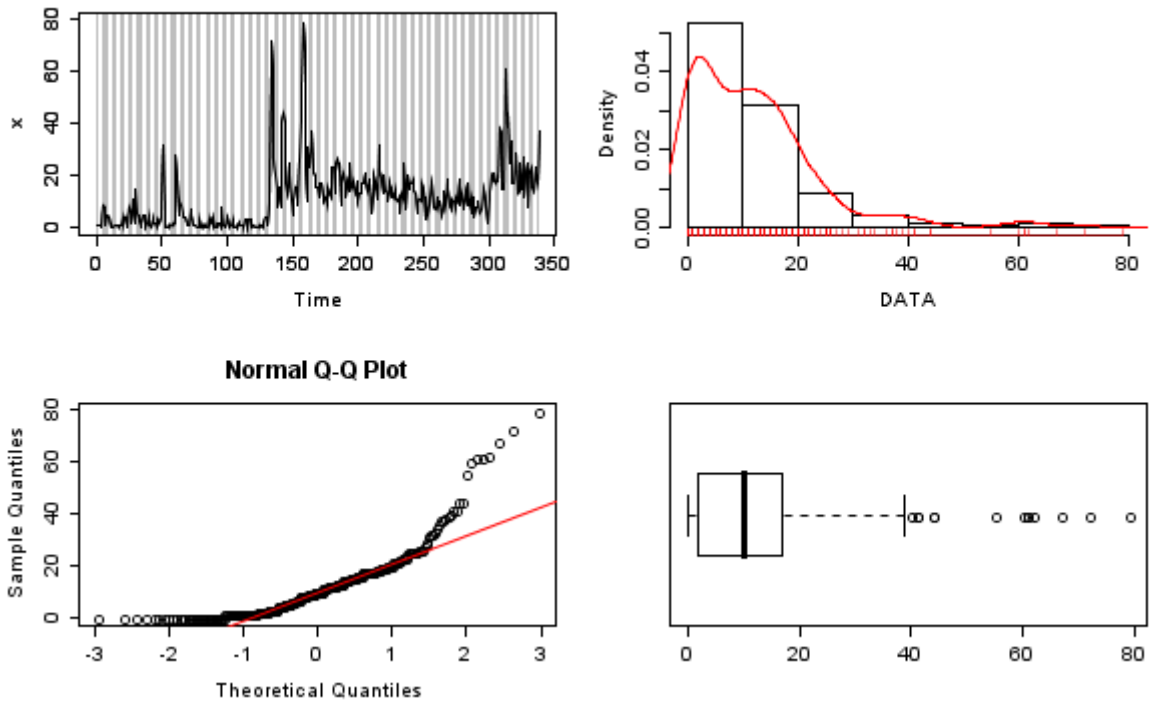

Figure 3: Web Traffic EDA

This graph show how exist in your data Outliers and Normal Analysis show that Web Traffic is not Normal.

```
> JAB.EDA(DATOS.TS);

[1] "JAB: EDA VIEW 1.1"

  The decimal point is 1 digit(s) to the right of the |

  0 | 000000000000000000000000000000000011111111111111111111111111111111+30
  0 | 555555555556666666677777777888888888888899999999999999
  1 | 000000000000001111111122222222222222333333333333444444444
  1 | 555555555556666666677777777777777777788888899999999
  2 | 0000000111111222333334
  2 | 555555556666789
  3 | 12234
  3 | 67789
  4 | 01144
  4 |
  5 |
  5 | 5
  6 | 0112
  6 | 7
  7 | 2
  7 | 9

     MEAN    MEDIAN        SD  KURTOSIS  SKEWNESS
12.121302 10.000000 12.505195  6.385821  2.114547
[1] ""
[1] "Normal Test, with p-value = 0,05"
    Shapiro-Wilk Anderson-Darling       Cramer-von          Pearson
    6.868276e-20     1.214736e-31     3.538717e-08     7.722848e-51
 Shapiro-Francia      Jarque-Bera
    2.539501e-17     0.000000e+00
[1] "X" "X" "X" "X" "X" "X"
```

Besides, it is necessary to see the periodogram:

```
JAB.TS.PERIODOGRAM(DATOS.TS);
```

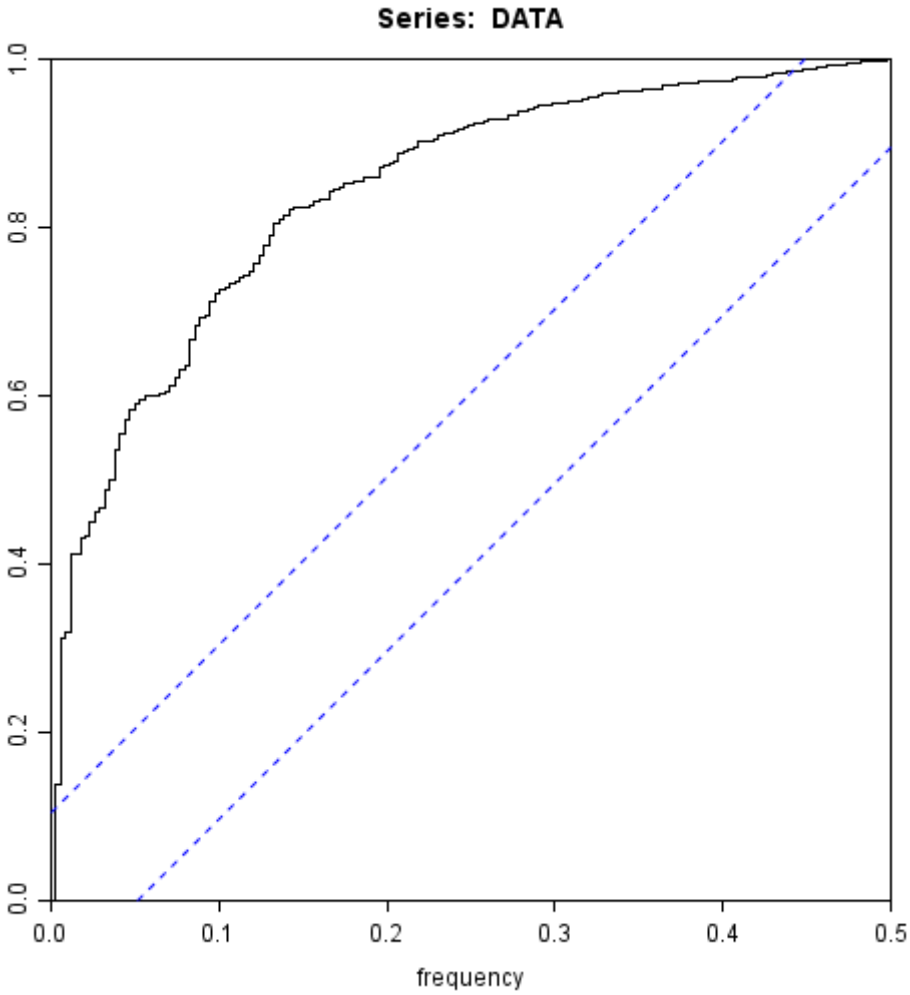

Figure 4: Periodogram Graphic

COMMENTS

If a want to model my Web Traffic to make predictions, I have to observe the following Graphics:

1. ACF Graph
2. PACF Graph

The Autocorrelation Functions
The techniques of model identification which are most commonly used were propounded originally by Box and Jenkins (1972). Their basic tools were the sample autocorrelation function and the partial autocorrelation function. We shall describe these functions and their use separately from the spectral density function which ought, perhaps, to be used more often in selecting models. The fact that spectral density function is often overlooked is probably due to an unfamiliarity with frequency-domain analysis on the part of many model builders.

Autocorrelation function (ACF).
Given a sample y0; y1; : : : ; yT¡1 of T observations, we de¯ne the sample autocorrelation function to be the sequence of values

$$r_\tau = c_\tau / c_0, \quad \tau = 0, 1, \ldots, T-1,$$

where in

$$c_\tau = \frac{1}{T} \sum_{t=\tau}^{T-1} (y_t - \bar{y})(y_{t-\tau} - \bar{y})$$

is the empirical autocovariance at lag *t* and *c0* is the sample variance. One should note that, as the value of the lag increases, the number of observations comprised in the empirical autocovariance diminishes until thenal element cT¡1 = T¡1(y0 ¡ ¹y)(yT¡1 ¡ ¹y) is reached which comprises only the rst and last mean-adjusted observations. In plotting the sequence {*rt*}, we shall omit the value of *r0* which is invariably unity. Moreover, in interpreting the plot, one should be wary of giving too much credence to the empirical autocorrelations at lag values which are signicantly high in relation to the size of the sample.

Partial autocorrelation function (PACF).

The sample partial autocorrelation *pt* at lag *t* is simply the correlation between the two sets of residuals obtained from regressing the elements yt and yt¡¿ on the set of intervening values y1; y2; : : : ; yt-1. The partial autocorrelation measures the dependence between yt and yt-1 after the effect of the intervening values has been removed. The sample partial autocorrelation pt is virtually the same quantity as the estimated coeffcient of lag *t* obtained by tting an autoregressive model of order ¿ to the data. Indeed, the diference between the two quantities vanishes as the sample size increases. The Durbin-Levinson algorithm provides an effcient way of computing the sequence *{pt}* of partial autocorrelations from the sequence of *{ct}* of autocovariances. It can be seen, in view of this algorithm, that the information in *{ct}* is equivalent to the information contained jointly in *{pt}* and c0. Therefore the sample autocorrelation function frtg and the sample partial autocorrelation function fptg are equivalent in terms of their information content.

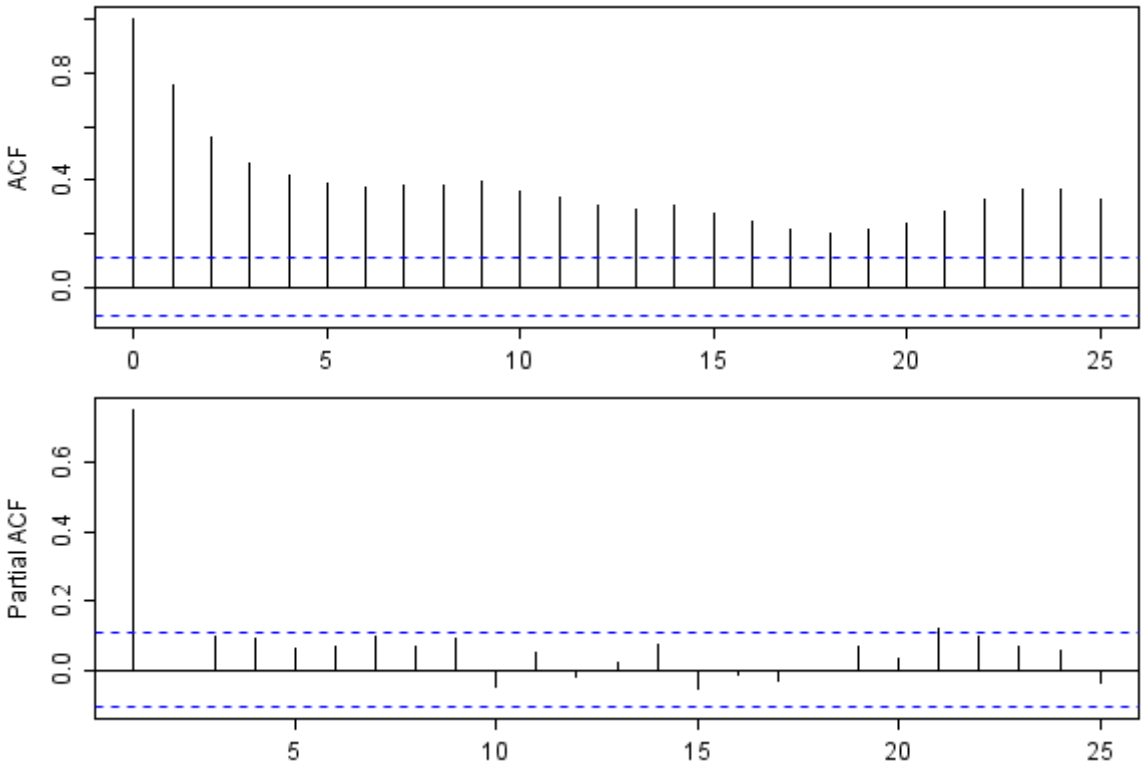

Figure 5: ACF & PACF Analysis

If you notice, data say you the right model to predict data. In this case, Web Traffic data will model using a AR model. This conclusion is due to analyze previous graphs because ACF has a lot of coeficcients not nulls but if you see PAFC you can see that only the first coeficcient is null, then the best model to forecast is AR

The following Table is very useful to reach conclusions.

MODEL	ACF	FACF
AR(p)	Many coefficients are not null	First coefficients are not null
MA(q)	First coefficients are not null	Many coefficients are not null
ARMA(p,q)	Many coefficients are not null	Many coefficients are not null

AR, Autoregressive Models

The autoregressive model is one of a group of linear prediction formulas that attempt to predict an output of a system based on the previous outputs and inputs. The autoregressive (AR) models are used in time series analysis to describe stationary time series.

The current value of the series is a linear combination of the p most recent past values of itself plus an error term, which incorporates everything new in the series at time t that is not explained by the past values. This is like a multiple regressions model but is regressed not on independent variables, but on past values; hence the term "Autoregressive" is used.

An important guide to the properties of a time series is provided by a series of quantities called sample autocorrelation coefficients or serial correlation coefficient, which measures the correlation between observations at different distances apart. These coefficients often provide insight into the probability model which generated the data. The sample autocorrelation coefficient is similar to the ordinary correlation coefficient between two variables (x) and (y), except that it is applied to a single time series to see if successive observations are correlated.

Given (N) observations on discrete time series we can form (N - 1) pairs of observations. Regarding the first observation in each pair as one variable, and the second observation as a second variable, the correlation coefficient is called autocorrelation coefficient of order one.

A useful aid in interpreting a set of autocorrelation coefficients is a graph called a correlogram, and it is plotted against the lag(k); where is the autocorrelation coefficient at lag(k). A correlogram can be used to get a general understanding on the following aspects of our time series:

1. A random series: if a time series is completely random then for Large (N), will be approximately zero for all non-zero values of (k).
2. Short-term correlation: stationary series often exhibit short-term correlation characterized by a fairly large value of 2 or 3 more correlation coefficients which, while significantly greater than zero, tend to get successively smaller.
3. Non-stationary series: If a time series contains a trend, then the values of will not come to zero except for very large values of the lag.
4. Seasonal fluctuations: Common autoregressive models with seasonal fluctuations, of periods.

AR model is defined using the following ecuations:

$$y_k = \sum_{j=1}^{p} a_j x_{k+j} \quad k = p..1$$

$$y_k = \sum_{j=1}^{p} a_j x_{k-j} \quad k = p+1..N$$

In these equations x is the data series of length N and a is the autoregressive parameter array of order p. AutoSignal uses the positive sign (linear prediction) convention for the AR coefficients. The model is defined as reverse prediction for the first p values, and forward prediction for the remaining N-p values. This definition is used for all AR fit statistics, although it is not the model fitted in any of the AR linear least-squares procedures.

Selection Criteria: Several criteria may be specified for choosing a model format, given the simple and partial autocorrelation correlogram for a series:

1. If none of the simple autocorrelations is significantly different from zero, the series is essentially a random number or white-noise series, which is not amenable to autoregressive modeling.

2. If the simple autocorrelations decrease linearly, passing through zero to become negative, or if the simple autocorrelations exhibit a wave-like cyclical pattern, passing through zero several times, the series is not stationary; it must be differenced one or more times before it may be modeled with an autoregressive process.
3. If the simple autocorrelations exhibit seasonality; i.e., there are autocorrelation peaks every dozen or so (in monthly data) lags, the series is not stationary; it must be differenced with a gap approximately equal to the seasonal interval before further modeling.
4. If the simple autocorrelations decrease exponentially but approach zero gradually, while the partial autocorrelations are significantly non-zero through some small number of lags beyond which they are not significantly different from zero, the series should be modeled with an autoregressive process.
5. If the partial autocorrelations decrease exponentially but approach zero gradually, while the simple autocorrelations are significantly non-zero through some small number of lags beyond which they are not significantly different from zero, the series should be modeled with a moving average process.
6. If the partial and simple autocorrelations both converge upon zero for successively longer lags, but neither actually reaches zero after any particular lag, the series may be modeled by a combination of autoregressive and moving average process.

I will calculate the parameter P to model a AR model with data

```
#############
# AR MODEL #
#############

DATOS.TS.AR <- ar(DATOS.TS, aic = TRUE, order.max=20);
DATOS.TS.AR
TITLE <- "AIC Analysis to determinate best order in AR model";
X_LABEL = "AR order";
Y_LABEL = "AIC Value";
PLOT_DATA <- DATOS.TS.AR$aic+.0001;
plot(PLOT_DATA,type="b",log="y" ,main = TITLE, xlab=X_LABEL, ylab=Y_LABEL);


Call:
ar(x = DATOS.TS, aic = TRUE, order.max = 20)

Coefficients:
      1        2        3        4        5        6        7        8
 0.7181  -0.0781   0.0202   0.0441   0.0136  -0.0028   0.0560  -0.0027
      9
 0.0928

Order selected 9  sigma^2 estimated as  66.85
```

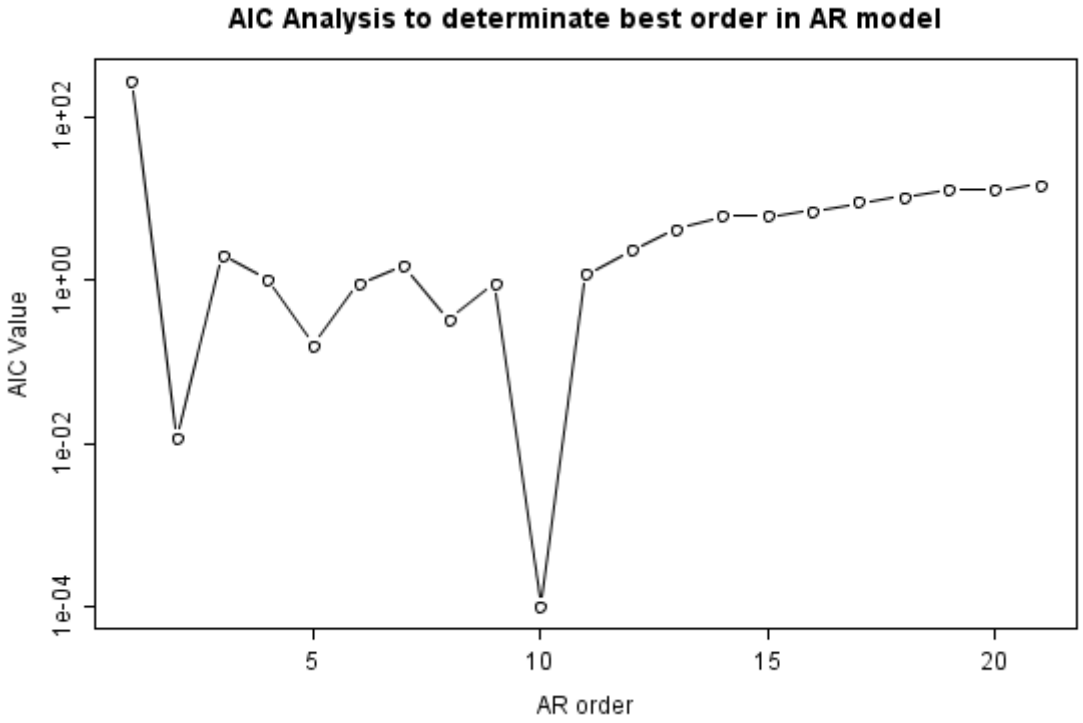

Figure 6: AIC Analysis

Using the AIC criterion, which order AR do you select for the data?
Let's look at the roots of the corresponding AR polynomial (roots of zp_(z−1) - remember the roots lie inside the unit circle if the process is stationary and causal).

```
roots<-polyroot(c(rev(-DATOS.TS.AR$ar),1))
plot(roots,xlim=c(-1.2,1.2),ylim=c(-1.2,1.2))
lines(complex(arg=seq(0,2*pi,len=300)))
```

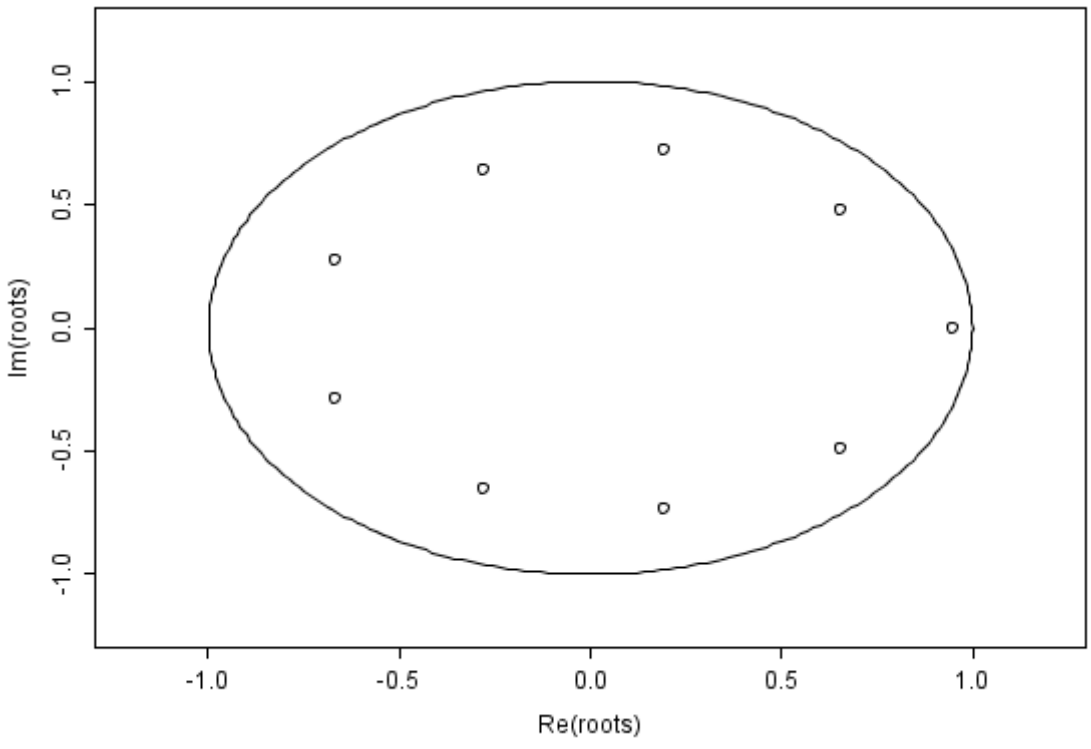

Figure 7: Uniroot Analysis

Once you have modeled your data with AR models, it is time to forecast.

For example if you want to forecast next quart, 90 days:

```
#May 31
#June 30
#July 31
plot(forecast(DATOS.TS.AR,92,conf=c(80,95)));
```

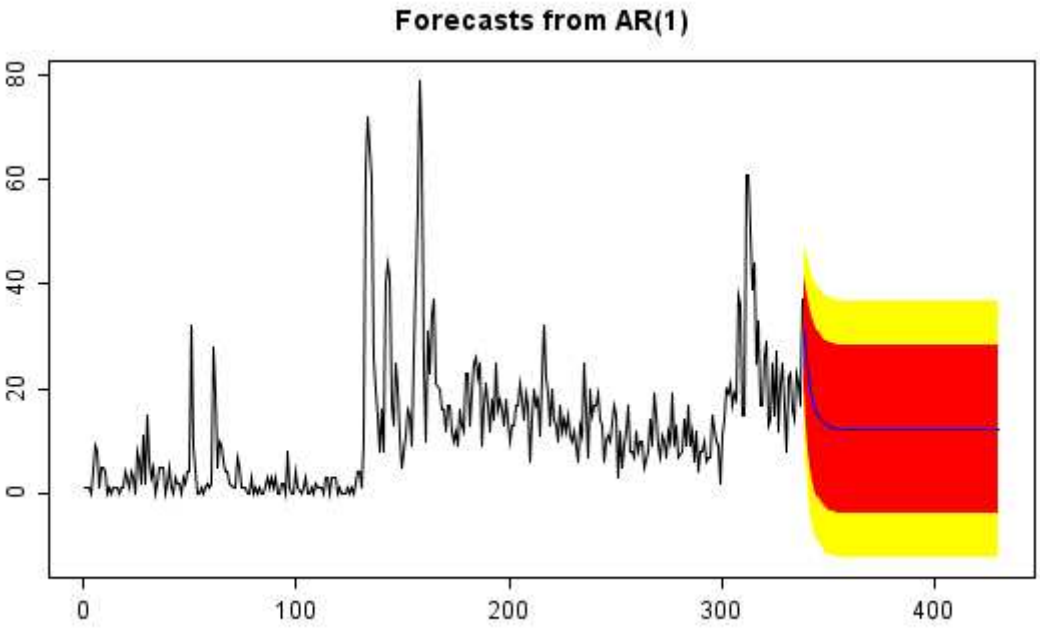

Figure 8: Forecast next Quarter

Now I know the forecast for next quarter, but I want to know the rates. To know the rates, I have to isolate information per month in a array to calculate that information:

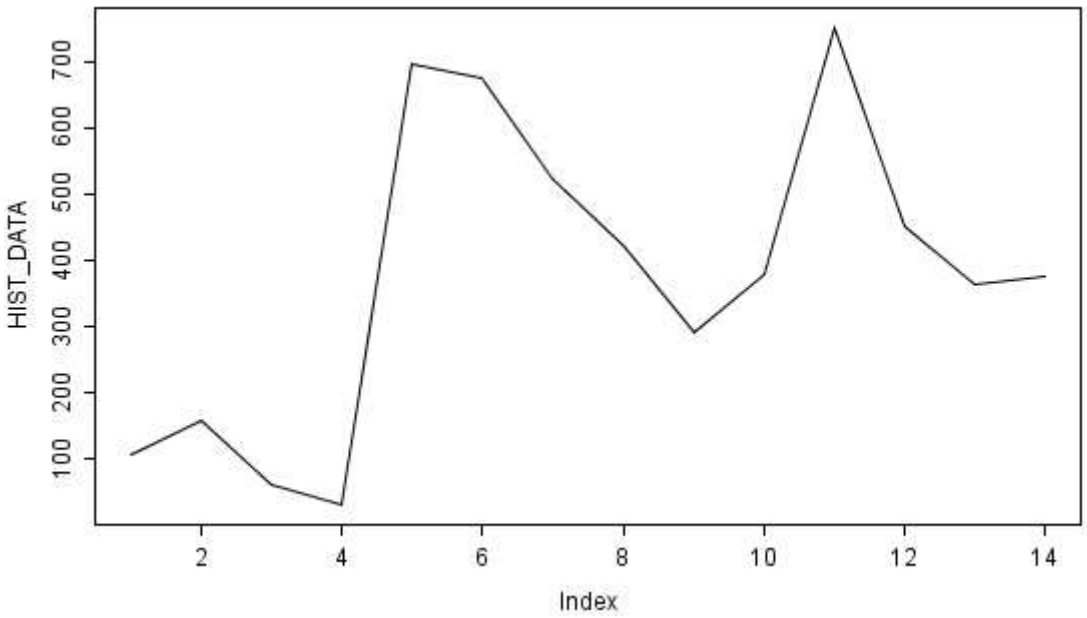

Figure 9: Web Traffic with Monh detail

```
SPLIT_DATA <- PAGES

#Create a histogram with agregate data
HIST_DATA <- c()
INIT <-5
LIMIT <- 5+30-1;
#LIMIT
JUN_2006 <- sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
#DATE_TRAFFIC[INIT:LIMIT];
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
#DATE_TRAFFIC[INIT:LIMIT];
JUL_2006 <-  sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
#DATE_TRAFFIC[INIT:LIMIT];
AGO_2006 <- sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 30 -1
#DATE_TRAFFIC[INIT:LIMIT];
SEP_2006 <- SPLIT_DATA[INIT:LIMIT];
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
#DATE_TRAFFIC[INIT:LIMIT];
OCT_2006 <-  sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 30 -1
#DATE_TRAFFIC[INIT:LIMIT];
NOV_2006 <- sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
DATE_TRAFFIC[INIT:LIMIT];
DEC_2006 <-  sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
```

```
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
#DATE_TRAFFIC[INIT:LIMIT];
JUN_2007 <-  sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 28 -1
#DATE_TRAFFIC[INIT:LIMIT];
FEB_2007 <- sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 31 -1
#DATE_TRAFFIC[INIT:LIMIT];
MAR_2007 <- SPLIT_DATA[INIT:LIMIT];
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
INIT <- LIMIT +1
LIMIT <- INIT + 30 -1
#DATE_TRAFFIC[INIT:LIMIT];
APR_2007 <-  sum(SPLIT_DATA[INIT:LIMIT]);
MONTH <- sum(SPLIT_DATA[INIT:LIMIT]);
HIST_DATA <- c(HIST_DATA,MONTH)
#FORECAST
MAY_2007 <- sum(DATOS.TS.AR.PREDICTION_92$pred[1:31])
JUN_2007 <-  sum(DATOS.TS.AR.PREDICTION_92$pred[32:61])
JUL_2007 <-  sum(DATOS.TS.AR.PREDICTION_92$pred[62:92])
MAY_2007 <- as.integer(MAY_2007)
JUN_2007 <- as.integer(JUN_2007)
JUL_2007 <- as.integer(JUL_2007)
HIST_DATA <- c(HIST_DATA,MAY_2007)
HIST_DATA <- c(HIST_DATA,JUN_2007)
HIST_DATA <- c(HIST_DATA,JUL_2007)
#HIST_DATA
#HIST_DATA.TS <- ts(HIST_DATA)

plot(HIST_DATA, type="l")

RATE_JUN <- (JUN_2006/ JUN_2007) * 100;
RATE_JUN
RATE_JUL <- (JUL_2006/ JUL_2007) * 100;
RATE_JUL
```

The results says me that next June I will increase my web traffic in 29.75% and July in 42.4%

This results show that this web site has good health in relation to web Traffic.

DOUBT: Why if i have aggregated data, my results are normal?

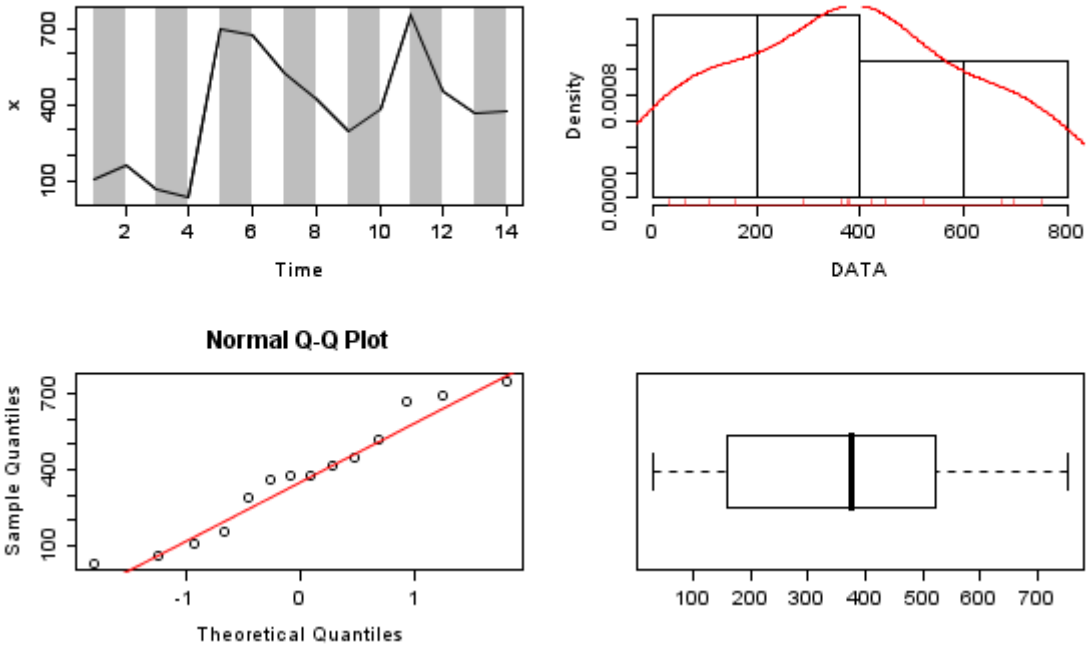

Figure 10: EDA with Month Data

```
[1] "JAB: EDA VIEW 1.1"

  The decimal point is 2 digit(s) to the right of the |

  0 | 3616
  2 | 9688
  4 | 252
  6 | 705

       MEAN       MEDIAN           SD     KURTOSIS     SKEWNESS
377.28571429 377.00000000 233.04586164  -1.30104094   0.06103328
[1] ""
[1] "Normal Test, with p-value = 0,05"
    Shapiro-Wilk Anderson-Darling       Cramer-von          Pearson
       0.5185340        0.6090312        0.7019398        0.5152632
 Shapiro-Francia      Jarque-Bera
       0.6935137        0.7300644
[1] "OK" "OK" "OK" "OK" "OK" "OK"
```