This Whitepaper describes how to perform What-If analysis in Qlikview using R

Whitepaper – Linear Regression Model using R in Qlikview

**Contents**

[Introduction 2](#_Toc436930140)

[Software Requirements 2](#_Toc436930141)

[Implementation Methodology 2](#_Toc436930142)

[1. Loading Input Data: 2](#_Toc436930143)

[2. R Code in Macro 4](#_Toc436930144)

[4. Loading Output Data 6](#_Toc436930145)

[5. What-If Analysis 6](#_Toc436930146)

[Your Views on the Concept 7](#_Toc436930147)

[Attachment 7](#_Toc436930148)

# Introduction

Qlikview is a powerful visualization tool where all kinds of analysis is been made but it is not a complete or fully fledged statistical tool. Hence complex regression models could not be calculated very accurately in Qlikview. And for this very reason we integrate Qlikview BI tool with other statistical tool like SAS, R so on.

This whitepaper takes the base of an application in which Qlikview server interacts with R server to perform regression calculation. A sample data of quarterly sales has been provided along with the amount spent on advertisement.

The goal is to perform What-If analysis on sales with respect to the amount spent of advertising modes such as Newspaper, Television and Internet. With the help of regression analysis obtained from R, the sales firm could decide how much to invest on each advertising channels.

# Software Requirements

* Install StatConnDCom connector, it will be available in the below link

<http://rcom.univie.ac.at/download.html>

* Two R packages, ‘rscproxy’ and ‘rcom’ are absolute for the integration. Download the packages and save it in the below path

C:\Program Files\R\R-3.2.1\library

* Install R (Optional).

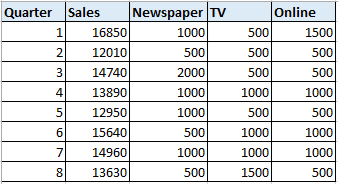
# Implementation Methodology

1. Loading Input Data:

First, create a variable in Edit script to store the source path (the path in which the application is stored/saved).

**LET** ***vSourcePath*** = Replace(DocumentPath(),'\'&DocumentName(),'\');

Now load the sample data into Qlikview, the sample data screenshot is shown below



In this sample data, the amount spent on each advertising mode has been provided separately. For analysis and to show Quarter wise sales charts we would require the advertising modes in a single column/field and the amount in another field. To achieve this, cross table is used.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Transposed the Salesforce csv data file using Cross table function and stored the data as qvd \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
SalesData:

CrossTable(Advertisement\_Channels, Amount\_Invested, 2)  
**LOAD** Quarter,   
 Sales,   
 Newspaper,   
 TV,   
 Online  
FROM  
***$(vSourcePath)***Salesforce.csv  
(txt, codepage is 1252, embedded labels, delimiter is ',', msq);  
  
**STORE** SalesData into SalesData.qvd(qvd);

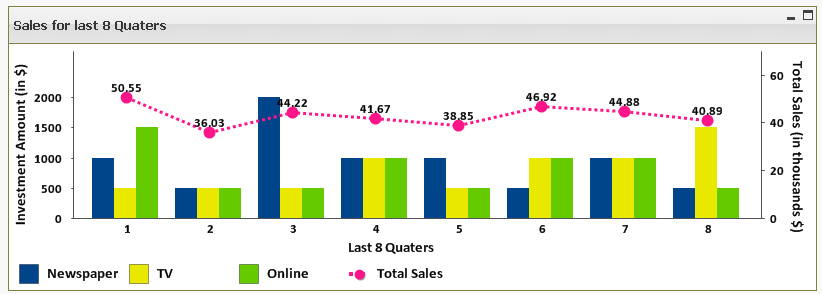
By the above script the last three columns of the data will be transposed and the fields ‘Newspaper’, ‘Tv’, and ‘Online’ becomes the record values of the field ‘Advertisement\_Channels’ and their corresponding amount values are stored in a column ‘Amount\_Invested’. This transformation is stored as a QVD and this QVD will be our input data sent to R for calculation.

**Note: The above script has to be commented after creating the QVD.**

Now loading the generated QVD back to the application the script looks as like shown below

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Transposed the Salesforce csv data file using Cross table function and stored the data as qvd \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
//SalesData:  
//CrossTable(Advertisement\_Channels, Amount\_Invested, 2)  
//LOAD Quarter,   
// Sales,   
// Newspaper,   
// TV,   
// Online  
//FROM  
//$(vSourcePath)Salesforce.csv  
//(txt, codepage is 1252, embedded labels, delimiter is ',', msq);  
//  
//STORE SalesData into SalesData.qvd(qvd);  
  
/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Importing Salesforce QVD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
SalesData:  
**LOAD** Quarter,   
 Sales,   
 Advertisement\_Channels,   
 Amount\_Invested  
FROM  
***$(vSourcePath)***SalesData.qvd  
(qvd);

The below screenshot shows the quarter wise sales



As seen, there is a constant ups and downs in the sales rather than a gradual increase in Sales. Moreover, the sales of last quarter has a considerable dip compared to the previous quarter.

2. R Code in Macro

Below is the macro that helps Qlikview and R to interact.

Qlikview is connected to R server through a connector ‘StatConnDCom Connector’. This connector should have been pre-installed in the user’s machine for this application to work.

The below macro initializes R console via the connector and the data is read by R from Qlikview and gets calculated. The output from R is then loaded in Qlikview.

Sub RScore  
  
 set R = createobject("StatConnectorSrv.StatConnector")  
 R.Init "R"  
 R.EvaluateNoReturn "Input\_Data=read.csv ('C:/Users/379817/Desktop/RegressionModel/Salesforce.csv',header=TRUE)"

R.EvaluateNoReturn "attach(Input\_Data)"  
 R.EvaluateNoReturn "LinearModel=lm(Sales ~ Newspaper + TV + Online)"  
 R.EvaluateNoReturn "Coeff=coef(LinearModel)"  
   
 R.EvaluateNoReturn "Sales\_without\_ad=Coeff[1]"  
 R.EvaluateNoReturn "Newspaper=Coeff[2]"  
 R.EvaluateNoReturn "TV=Coeff[3]"  
 R.EvaluateNoReturn "Online=Coeff[4]"  
 R.EvaluateNoReturn "values = data.frame(Sales\_without\_ad,Newspaper, TV, Online, row.names='Sales\_per\_1$')"  
   
 R.EvaluateNoReturn "Transpose=t(values)"  
 R.EvaluateNoReturn "write.csv(Transpose,'D:/05.R Integration/sample.csv')"  
 R.EvaluateNoReturn "Output=read.csv('D:/05.R Integration/sample.csv',header= TRUE)"

R.EvaluateNoReturn "write.csv(Output,'D:/05.R Integration/Output.csv')"

ActiveDocument.DoReload 2, false, false   
   
R.close

end sub  
   
3. Macro Brief

* The code *R.Init "R"* initializes R console.
* *R.EvaluateNoReturn "Input\_Data=read.csv ('C:/Users/379817/Desktop/ RegressionModel/Salesforce.csv',header=TRUE)"*

The above code reads the csv file in the given path and stores it in the variable Input\_Data.

* *R.EvaluateNoReturn "attach(Input\_Data)"*

The attach() function in R can be used to make objects within dataframes accessible in R

* *R.EvaluateNoReturn "LinearModel=lm(Sales ~ Newspaper + TV + Online)"*

The R function ‘lm’ represents ‘Linear Model’ and the code ‘lm(Sales ~ Newspaper + TV + Online)’ calculates linear regression for the advertising channels and it gives the per $ sales obtained from each advertising channels.

* *R.EvaluateNoReturn "Coeff=coef(LinearModel)"*

The coefficients or the values obtained after the linear model calculation (Per $ Sales) is then

Stored in the variable Coeff with the help of R function coef ()

* *R.EvaluateNoReturn "Coeff=coef(LinearModel)"*

By default the first constant or the first coefficient gives the per $ sales obtained without any advertisement. And the next consecutive coefficients are for Newspaper, TV and Online as per the coding order lm(Sales ~ Newspaper + TV + Online).

* *R.EvaluateNoReturn "Sales\_without\_ad=Coeff[1]"  
  R.EvaluateNoReturn "Newspaper=Coeff[2]"  
  R.EvaluateNoReturn "TV=Coeff[3]"  
  R.EvaluateNoReturn "Online=Coeff[4]"*

Each per $ sales of advertising channels are stored in separate variables.

* *R.EvaluateNoReturn"values=data.frame(Sales\_without\_ad,Newspaper,TV,Online, row.names='Sales\_per\_1$')"*

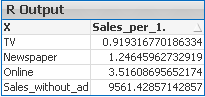
*R.EvaluateNoReturn "Transpose=t(values)"*

Data.frame is function in R used to create a table structure. In this case the values of sales\_without\_ad, Newspaper, TV and Online are stored under a column ‘Sales\_per\_1$’.

The output from R is stored in a csv file. The file will be stored in the source path and this can be loaded into Qlikview.

4. Loading Output Data

/\*\*\*\*\*\*\* Importing Coefficient values into Qlikview from R through csv file \*\*\*\*\*\*\*/  
  
OutPut:  
  
**LOAD**   
\*  
FROM  
D:\05.R Integration\Output.csv  
(txt, codepage is 1252, embedded labels, delimiter is ',', msq, filters(  
Remove(Col,Pos(Top,1))));

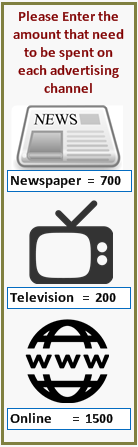


5. What-If Analysis

The sales per $ that is obtained from R will be the constant parameter and the amount that’s about to be spent on each advertising channel will be the variable parameter.

Input boxes for each advertising channels needs to be created so that the amount can be given as input to see the predicted sales.

In the sample Qlikview application, the input boxes are created as shown below

In here, the amount that is planned to spend on each advertising channels such as Newspaper, Television and in Internet can be entered in their respective Input boxes as shown in the screenshot on the left side.

The logic goes with the simple mathematical linear equation y=a+bx.

Where, y is the predicted sales

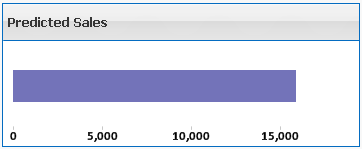
a is the sales obtained without advertisement

b is the constant value or the coefficients of all three advertising channels obtained from the R

x is the variable or the amount planned to spend on each channels.

As all the parameters are available, the expression written for expected sales is

=(sum({<X={'Newspaper'}>}Sales\_per\_1.)\****vNewspaper***)+(sum({<X={'TV'}>}Sales\_per\_1.)\****vTV***)+(sum({<X={'Online'}>}Sales\_per\_1.)\****vOnline***)+sum({<X={'Sales\_without\_ad'}>} Sales\_per\_1.)



# Your Views on the Concept

This method will greatly help every business firm to take a decision on how much to invest upon each products. And which product will give high profit when invested in large sum. This will help in reduce unwanted losses.

# Attachment

