**Group 2**

Ankitha Abhimanyu

Goutham Chandru Manghnani

Vidhya Babu

**Website Click Stream Analytics**

**INSY 5378: Data Science - Spring 2015**

**Source:**

Data Set received from the challenge posted by Zappo.com for the advanced analytics college internship position.

**Data Set Information:**

This dataset represents a clickstream data of the customer that is visiting the any website. Clickstream analysis (sometimes called clickstream analytics) is the process of collecting, analyzing, and reporting aggregate data about which pages visitors visit in what order - which are the result of the succession of mouse clicks each visitor makes (that is, the clickstream). There are two levels of clickstream analysis, traffic analysis and e-commerce analysis.

Traffic analysis operates at the server level by collecting clickstream data related to the path the user takes when navigating through the site. Traffic analysis tracks how many pages are served to the user, how long it takes pages to load, how often the user hits the browser's back or stop button, and how much data is transmitted before a user moves on. E-commerce-based analysis uses clickstream data to determine the effectiveness of the site as a channel-to-market by quantifying the user's behavior while on the Web site. It is used to keep track of what pages the user lingers on, what the user puts in or takes out of their shopping cart, and what items the user purchases.

A large volume of data can be gathered through click stream analysis, many e-businesses rely on pre-programmed applications to help interpret the data and generate reports on specific areas of interest. Clickstream analysis is considered to be most effective when used in conjunction with other, more traditional, market evaluation resources.

**Prediction and Analysis:**

In our project we would be concentrating on E-commerce-based analysis which uses clickstream data and assess the overall performance of Business. We would be analyzing the dataset for any trends and explain any insights by exploring areas like forecasting, segmenting, regression etc. for prediction. Our dataset contains records of clickstream data for the year 2013.

The platforms considered in our dataset include: Android, BlackBerry, ChromeOS, Linux, MacOSX, Macintosh, Other, SymbianOS, Unknown, Windows, Windows Phone, iOS, iPad, and iPhone. The sites considered are: Acme, Bolty, Pinnacle, Sortly, Tabular, and Widgetry.

**Attribute Information:**

No of Attributes present: - 12

No of Instances present: - 21061

The following attributes and their description is given below –

**Data Column Description:**

· Day – The calendar day

· Site - Company site visited by users

· new\_customer - 0 = returning customer; 1 = new customer; null = neither

· Platform - The type of device used by a website visitor

· Visits - The number of distinct website visits; 1 session may have multiple visits

· distinct\_sessions - The number of distinct website visitors; 1 session may have multiple visits

· Orders - The number of website orders

· gross\_sales - The total gross sales for website orders

· bounces - The number of visits that only viewed one page

· add\_to\_cart - The number of visits that added a product to cart

· product\_page\_views - The number of product pages viewed

· search\_page\_views - The number of search pages viewed

**New attributes Included**

We included 4 new attributes to check for its significance based on the exiting independent variables. The requested Metrics Formulae for the newly added attributes are:

conversion\_rate='orders'/'visits'

bounce\_rate='bounces'/'visits'

add\_to\_cart\_rate='add\_to\_cart'/visits'

Duration = 'product\_page\_views /distinct\_sessions'

**Business Problem**

Our business problem was to analyze and assess the business performance. The dataset contained records of customer visits and views based on the site and platform they were accessing. The determining factor for predicting the number of views that got converted to potential orders is based on the platform and site. Other parameters such as conversion rate, duration a site was accessed, add to cart rate were some of the other metrics that were considered.

Further we went ahead and got a better insight about our dataset. With order as a target variable we analyzed the significance of platform and site. We analyzed the impact of an order based on the platform or site depending on the duration of access.

Forecasting was another measure that was of our interest. With our dataset comprising of clickstream data for the year 2013, we tried to forecast the number of orders in the first quarter of 2014. To get a better insight we analyzed these predictions with platform and site as measuring factors of an order conversion.

**Pre-processing Steps**

Pre-processing Step 1: Removal of missing values

* Removed records that have had visits equal to 0 (Total Number of Records removed – 2469)
* Removed the records that have neither value for new\_customer attribute. (Total Number of Records removed – 7859)
* Total Number of Instances available after the Pre-processing Step 1 – 10733

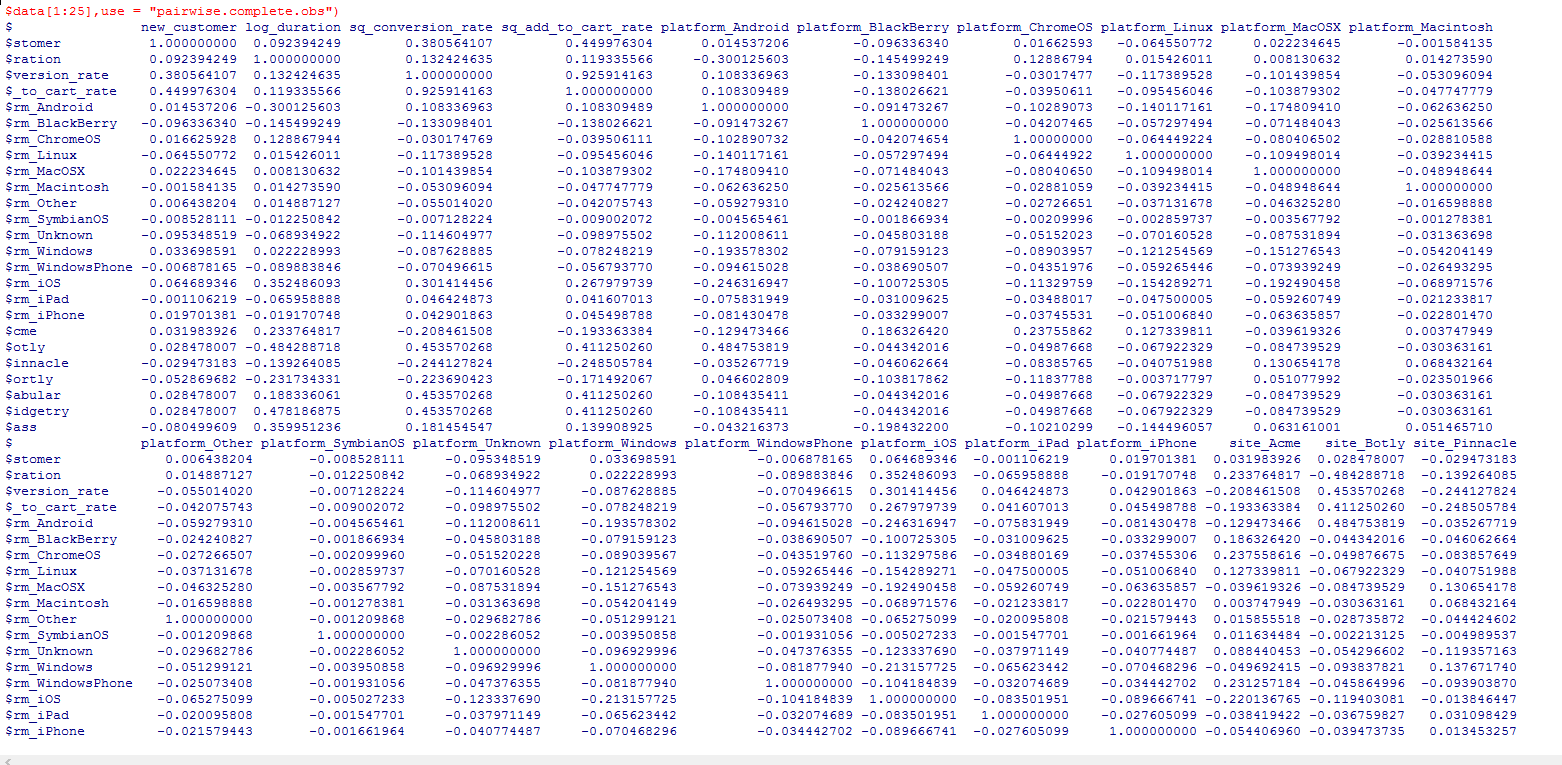
Pre-processing Step 2: Inclusion of new attributes

The following attributes were computed and included:

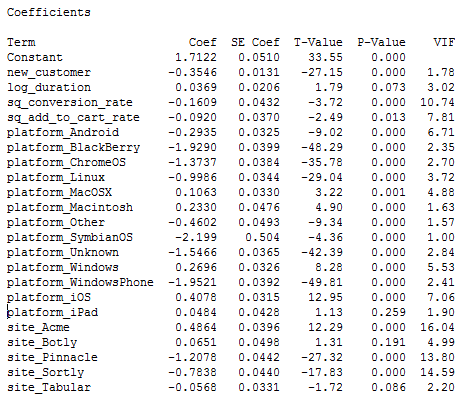
* Duration, which was computed using 'product\_page\_views /distinct\_sessions'
* Conversion rate, was computed using 'orders'/'visits'
* Add\_to\_cart\_rate, was computed using 'add\_to\_cart'/visits'
* Bounce\_rate, was computed using 'bounces'/'visits'

Pre-processing Step 3: Correlation and Multicollinearity Test

* A check for correlation to check if any of the attributes were closely related was done. Attributes that were highly correlated were removed. Visits, distinct\_sessions, gross\_sales, bounces, product\_page\_views, and search\_page\_views were the attributes that were highly correlated.
* A correlation greater than 0.8 is generally described as *strong*, whereas a correlation less than 0.5 is generally described as *weak*.
* After removing the correlated attributes the correlation matrix is as below:



Multicollinearity test was performed. Attributes with a VIF > 4 are considered to be multicollinear. Below is a screenshot of the results for a linear model.



We removed those variables which was highly correlated and multicollinear based on the p-value and VIF factor.

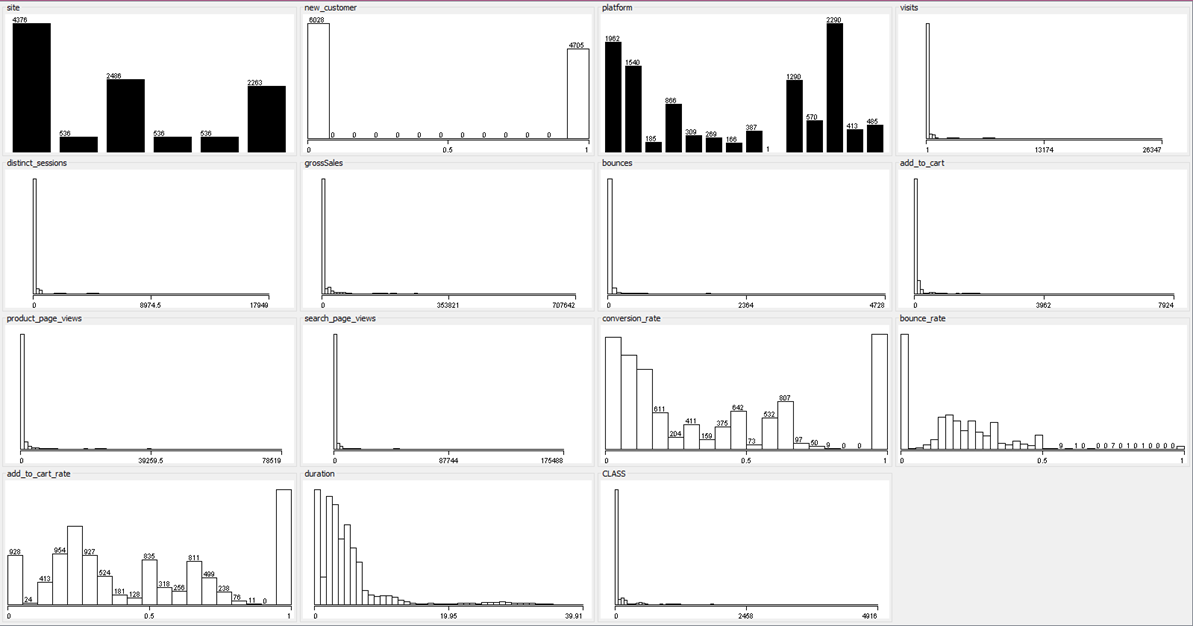
Pre-processing Step 4: Dict Vectorizer

We had 2 attributes which was non numerical (**Site:**- Acme – 1, Botly – 2, Pinnacle – 3, Sortly – 4, Tabular – 5, Widgetry – 6, **Platform :-** Blank – 1, Android – 2, BlackBerry – 3, ChromeOS – 4, iOS – 5, iPad-6, iPhone-7, Linux-8, Macintosh-9, MacOSX-10, Other – 11, SymbianOS – 12, Unknown – 13, Windows – 14,WindowsPhone – 15)

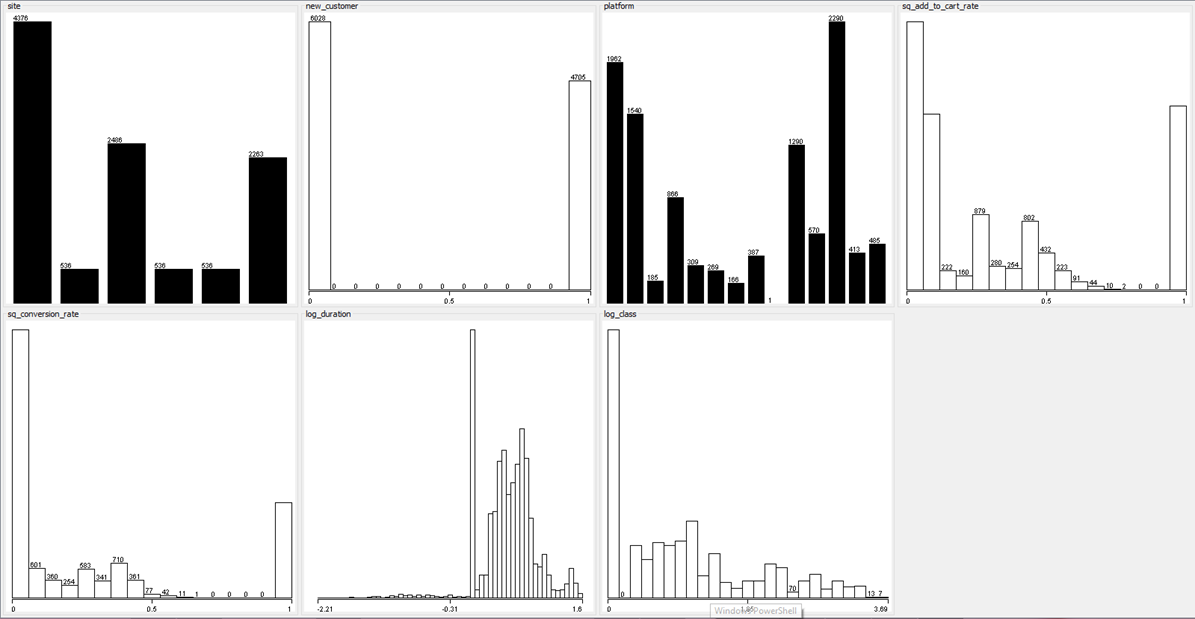
* Dict Vectorization was done to convert the categorical fields to numeric. Dict Vectorizer considers each of the categorical values as a separate column.
* 21 Dict vectorized fields were added after the platform and site attributes were vectorized.

Pre-processing Step 5: Variable transformation

* Visualizing the attributes in Weka is shown below:



* Attributes that were positively skewed or right skewed were log transformed (duration, order i.e. the class variable)
* Attributed that were fluctuating were squared (conversion\_rate, add\_to\_cart\_rate)
* Visualizing the attributes after performing transformation is shown below:



**Normality Probability Plots:**

sresid <- studres(result)

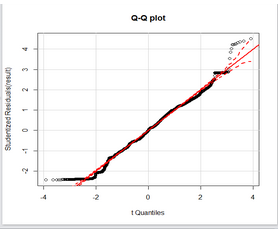
hist(sresid, freq=FALSE, main="Distribution of Studentized Residuals")

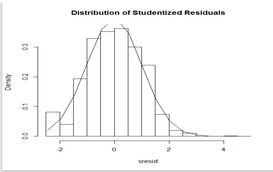
xfit<-seq(min(sresid),max(sresid),length=15)

yfit<-dnorm(xfit)

lines(xfit, yfit)

Below shown is the normality probability plot after our variable transformations:





**Regression**

Since the target variable has continuous values a Regressor was considered.

Multiple Linear regression is useful for modelling the relationship between a dependent variable(Y) and multiple explanatory or independent variables(X).

The multiple linear regression is of the form

***Y = b1X1 + b2X2 + ... + A***

Where,

*Y* is the dependent variable you are trying to predict - In our case it is Order

*X1*, *X2* and so on are the independent variables you are using to predict it. In our case the independent variables are as follows -

* sq\_add\_to\_cart\_rate
* sq\_conversion\_rate
* new\_customer
* platform\_Android
* platform\_BlackBerry
* platform\_ChromeOS
* platform\_Linux
* platform\_MacOSX
* platform\_Macintosh
* platform\_Other
* platform\_SymbianOS
* platform\_Unknown
* platform\_Windows
* platform\_WindowsPhone
* platform\_iOS
* platform\_iPad
* platform\_iPhone
* site\_Acme
* site\_Botly
* site\_Pinnacle
* site\_Sortly
* site\_Tabular
* site\_Widgetry
* log\_duartion

*b1*, *b2* and so on are the coefficients (**Intercepts**) or multipliers that describe the size of the effect the independent variables are having on your dependent variable *Y(***Order***)*, and *A* (**Intercept Coefficient**) is the value *Y*  is predicted to have when all the independent variables are equal to zero.

**Regression Diagnostics**

We ran multiple Regressor to check which model fits our dataset best. Regressors that were run are: Linear Regression, SGD Regressor with multiple penalties, LassoCV, RigdgeCV, ElasticNetCV, SVR with multiple kernel, Random forest Regressor, Extra trees Regressor, Bagging Regressor. Below is the R code for running a linear regression model on our dataset:

> result=lm(log\_class ~ new\_customer + log\_duration + sq\_conversion\_rate + sq\_add\_to\_cart\_rate + platform\_Android + platform\_BlackBerry + platform\_ChromeOS + platform\_Linux + platform\_MacOSX + platform\_Macintosh + platform\_Other + platform\_SymbianOS + platform\_Unknown + platform\_Windows + platform\_WindowsPhone + platform\_iOS + platform\_iPad + platform\_iPhone + site\_Acme + site\_Botly + site\_Pinnacle + site\_Sortly + site\_Tabular + site\_Widgetry, data = data)

> summary(result)

Call:

lm(formula = log\_class ~ new\_customer + log\_duration + sq\_conversion\_rate +

sq\_add\_to\_cart\_rate + platform\_Android + platform\_BlackBerry +

platform\_ChromeOS + platform\_Linux + platform\_MacOSX + platform\_Macintosh +

platform\_Other + platform\_SymbianOS + platform\_Unknown +

platform\_Windows + platform\_WindowsPhone + platform\_iOS +

platform\_iPad + platform\_iPhone + site\_Acme + site\_Botly +

site\_Pinnacle + site\_Sortly + site\_Tabular + site\_Widgetry,

data = mydata)

Residuals:

Min 1Q Median 3Q Max

-1.22381 -0.31166 0.00368 0.37067 2.26036

Coefficients: (2 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1.71216 0.05104 33.548 < 2e-16 \*\*\*

new\_customer -0.35462 0.01306 -27.153 < 2e-16 \*\*\*

log\_duration 0.03691 0.02060 1.792 0.073181 .

sq\_conversion\_rate -0.16089 0.04323 -3.722 0.000199 \*\*\*

sq\_add\_to\_cart\_rate -0.09204 0.03698 -2.489 0.012826 \*

platform\_Android -0.29349 0.03253 -9.022 < 2e-16 \*\*\*

platform\_BlackBerry -1.92901 0.03995 -48.291 < 2e-16 \*\*\*

platform\_ChromeOS -1.37370 0.03839 -35.784 < 2e-16 \*\*\*

platform\_Linux -0.99865 0.03439 -29.041 < 2e-16 \*\*\*

platform\_MacOSX 0.10630 0.03297 3.224 0.001266 \*\*

platform\_Macintosh 0.23302 0.04757 4.899 9.79e-07 \*\*\*

platform\_Other -0.46016 0.04927 -9.340 < 2e-16 \*\*\*

platform\_SymbianOS -2.19857 0.50398 -4.362 1.30e-05 \*\*\*

platform\_Unknown -1.54658 0.03648 -42.394 < 2e-16 \*\*\*

platform\_Windows 0.26963 0.03256 8.280 < 2e-16 \*\*\*

platform\_WindowsPhone -1.95206 0.03919 -49.812 < 2e-16 \*\*\*

platform\_iOS 0.40775 0.03148 12.952 < 2e-16 \*\*\*

platform\_iPad 0.04836 0.04283 1.129 0.258894

platform\_iPhone NA NA NA NA

site\_Acme 0.48641 0.03956 12.295 < 2e-16 \*\*\*

site\_Botly 0.06505 0.04979 1.307 0.191387

site\_Pinnacle -1.20783 0.04422 -27.316 < 2e-16 \*\*\*

site\_Sortly -0.78377 0.04396 -17.831 < 2e-16 \*\*\*

site\_Tabular -0.05681 0.03308 -1.717 0.086000 .

site\_Widgetry NA NA NA NA

---

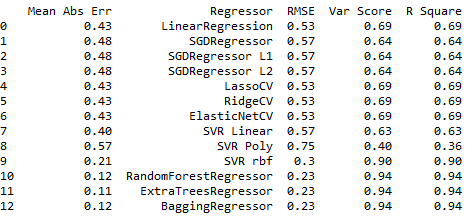
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5029 on 10710 degrees of freedom

Multiple R-squared: 0.7213, Adjusted R-squared: 0.7208

F-statistic: 1260 on 22 and 10710 DF, p-value: < 2.2e-16

**Regressor Results:**



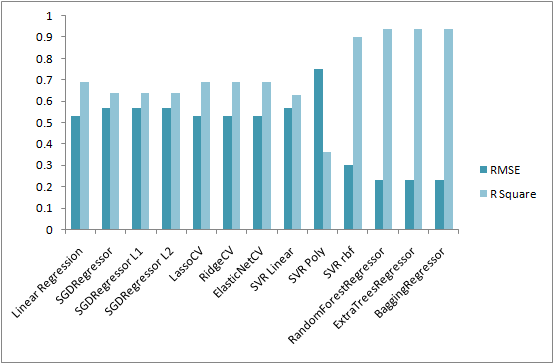
**Evaluating the Model Fit:**

In order to evaluate the model fit for our dataset, we considered the following two factors:

· R Square (Coefficient of determination) - Common statistic to evaluate model fit

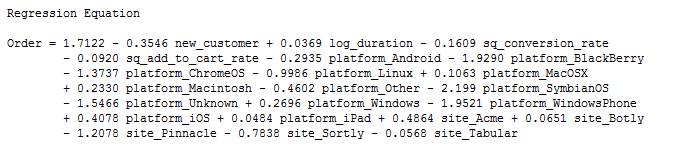
· Root Mean Square Error - Good measure of how accurately the model predicts the response

**Visualizing RMSE Vs R Square for the Regressors:**



**Best Regressors that fit our data best:**

1) Multiple linear Regression:



The weighted sum of the coefficients gives the regression equation. The prediction for the multiple linear Regressor model designed gives the following:



This represents our prediction model.

fit: vector or matrix as above

se.fit: standard error of predicted means

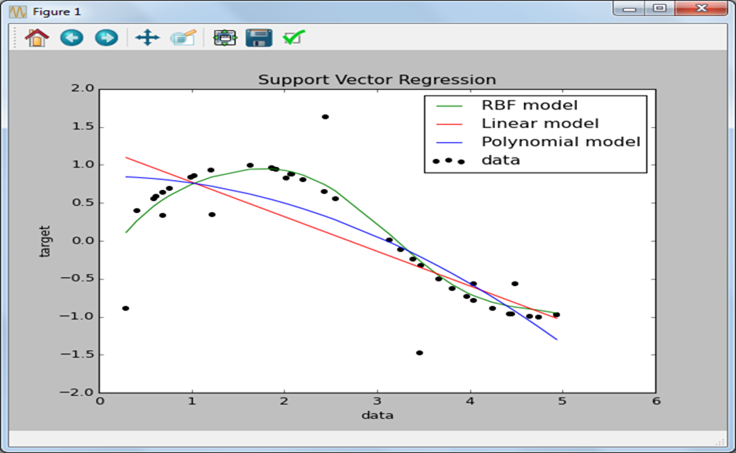
CI and PI: There is also a concept called prediction interval. Here we look at any specific value of x, x0, and find an interval around the predicted value ŷ0 for x0 such that there is a 95% probability that the real value of y (in the population) corresponding to x0 is within this interval .Prediction interval is a range that is likely to contain the response value of a single new observation given specified settings of the predictors in your model.

The prediction interval is always wider than the corresponding confidence interval of the prediction because of the added uncertainty involved in predicting a single response versus the mean response.

2) SVR with Kernel as ‘rbf’:

Support Vector Machines (SVM) are an analytical tool that can be used for both classification and regression purpose. SVR has two main strengths: good generalizability and robustness against outliers. Generalizability refers to the fact that SVR is designed in such a way that they provide the most simple solution (the most simple function that can describe the relation between the dependent and independent variables) for a given, fixed amount of (training) errors. Thus, SVR address the problem of Overfitting explicitly. Robustness of SVR is achieved by considering absolute, instead of quadratic, values of the errors. As a consequence, the influence of outliers less pronounced.

Basically there are three kernels: linear, radial basis function (rbf) and polynomial. For our dataset kernel with ‘rbf’ fits the best. A comparison of the three are graphically visualized below.



3) Bagging Regressor:

It’s a bootstrapping aggregation, a machine learning ensemble model which statistically fits a regression.

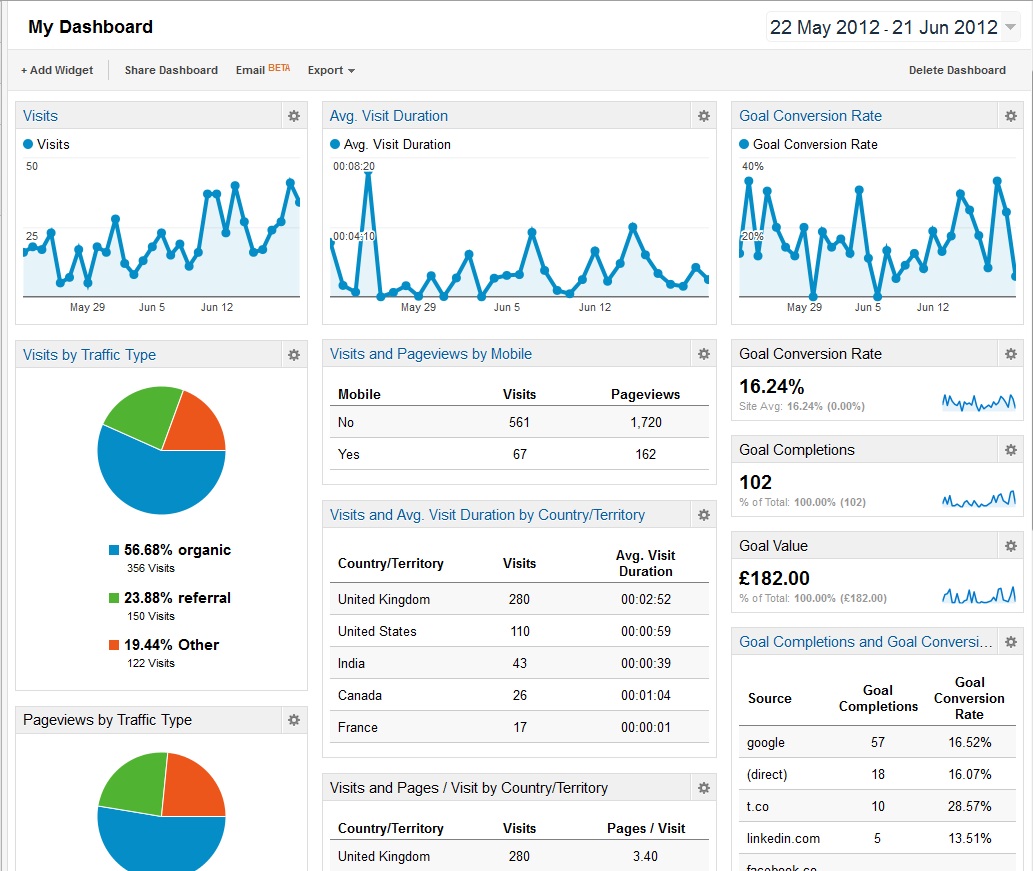
Main property of Bagging – Bagging decreases variance of the base model without changing the bias.

• Bagging typically helps – When applied with an over-fitted base model

• High dependency on actual training data

• It does not help much – High bias. When the base model is robust to the changes in the training data (due to sampling)

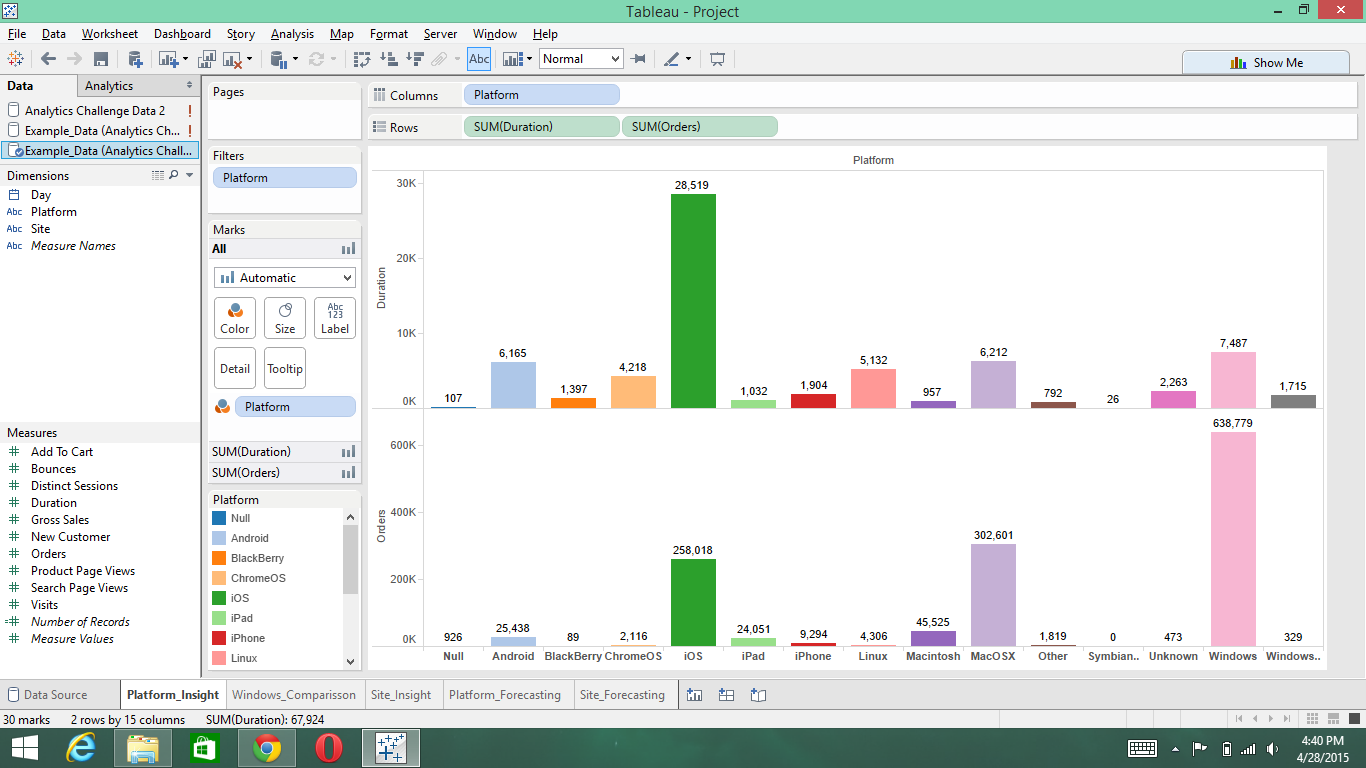
**Google Analytics Management Dashboard**



## On a Web site, click stream analysis (sometimes called clickstream analytics) is the process of collecting, analyzing, and reporting aggregate data about which pages visitors visit in what order which are the result of the succession of mouse clicks each visitor makes (that is, the clickstream).

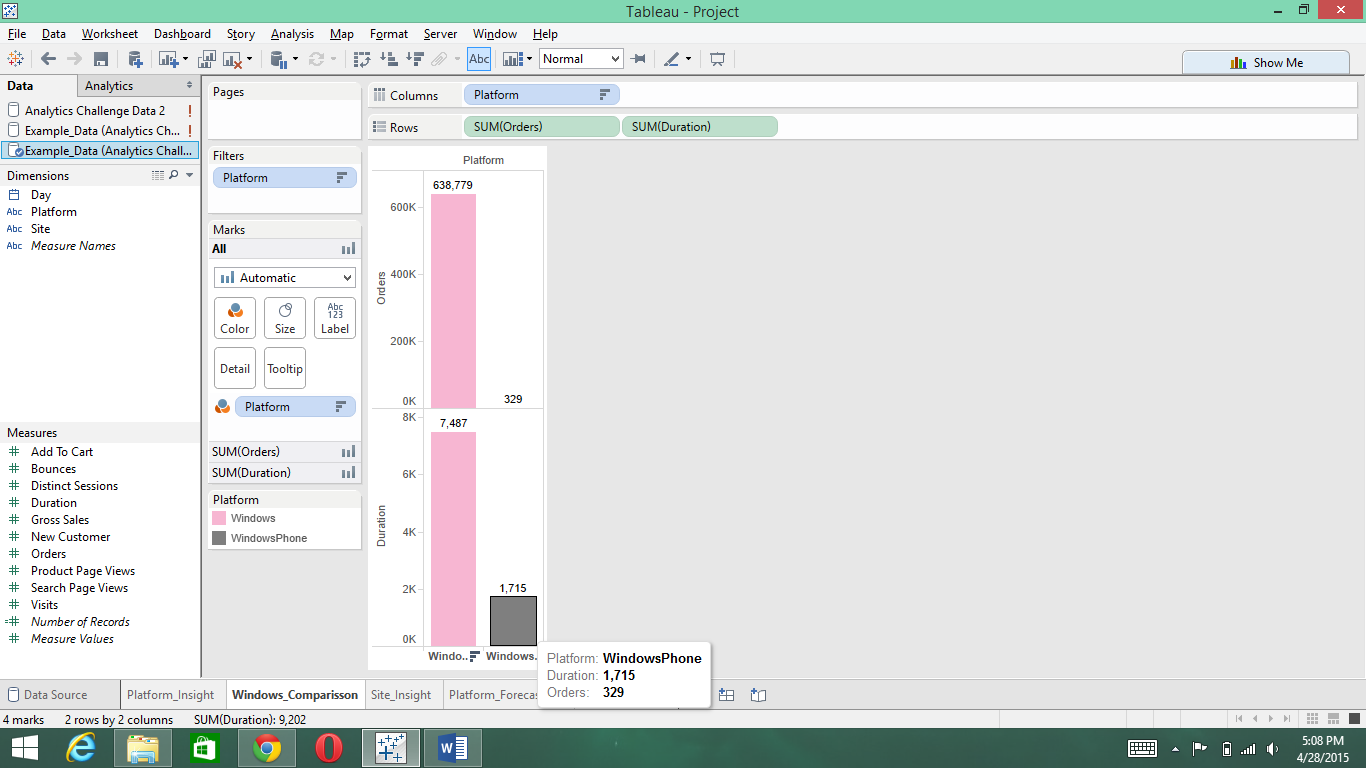
The above view represents a generic view of the Google analytics dashboard that would display a default Dashboard (called “My Dashboard”) that is pre-populated with a number of widgets showing the site’s traffic as measured via certain key metrics and dimensions: a timeline for number of users, a table of sessions by platforms, timelines for bounce rate and goal conversions, etc.

**Platform Insights**



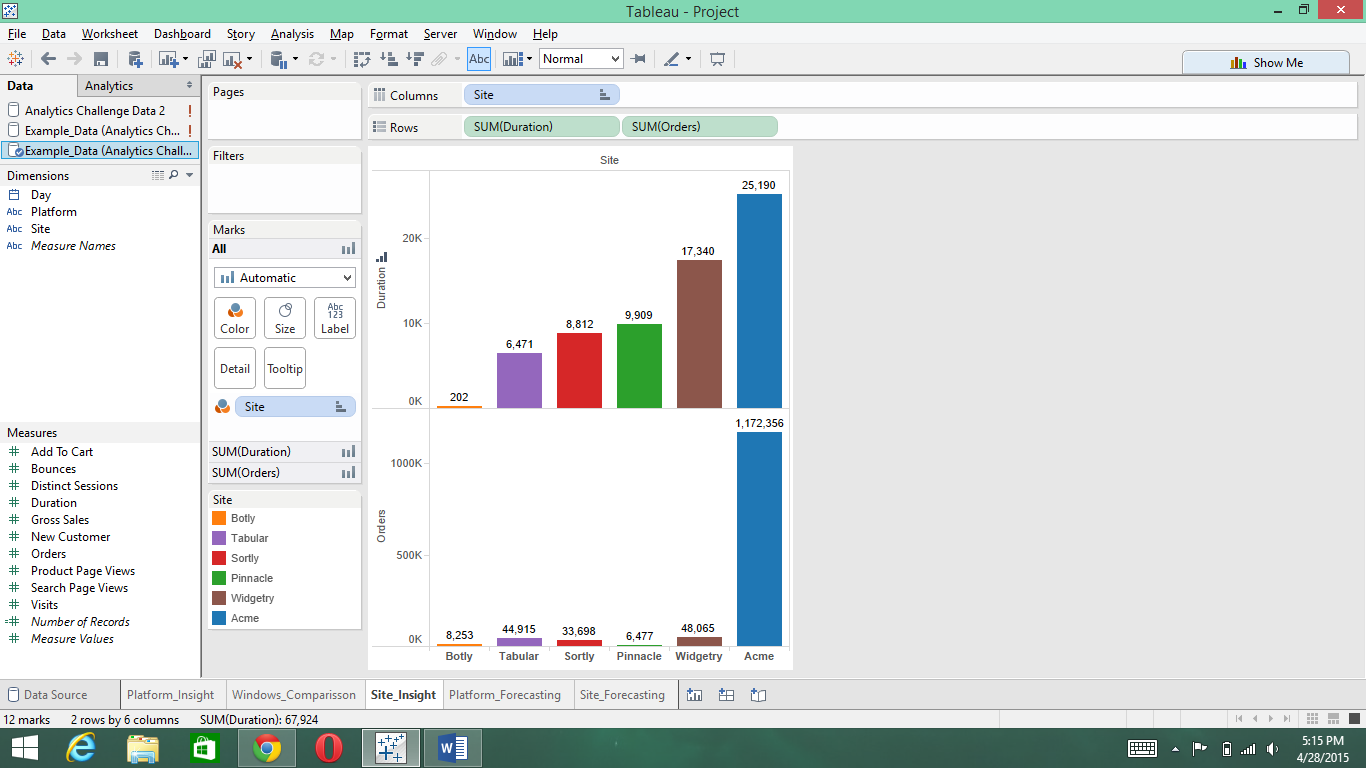
The above bar chart plots the Platform vs. Orders & Platform vs. Duration. The reports aggregates the metrics orders & duration for the whole year based on the dimension platform. This would give an insight as to which particular platform was responsible for increased orders and which particular platform the user spent most of the time, so that the company can target these platform users in future. Here, durations is a new attribute introduced which was calculated by dividing the product page views count with the distinct sessions count. According to our report, Windows OS was the platform that contributed to the maximum orders and maximum time spent by the customers.

**Windows OS vs. Windows Phone**



The above bar chart plots the platform Windows OS and Windows Phone with Orders & Platform vs. Duration. The reports aggregates the metrics orders & duration for the whole year based on the dimension platform for Windows OS and Windows Phone. Here, we can see that the people prefer using Windows desktop devices rather than using their Windows Phone to make an order. The amount of time spent on the Windows phone is very small when compared to the amount of time spent on the Windows desktop. So this can be the reason why people would prefer the Windows desktop to make an order.

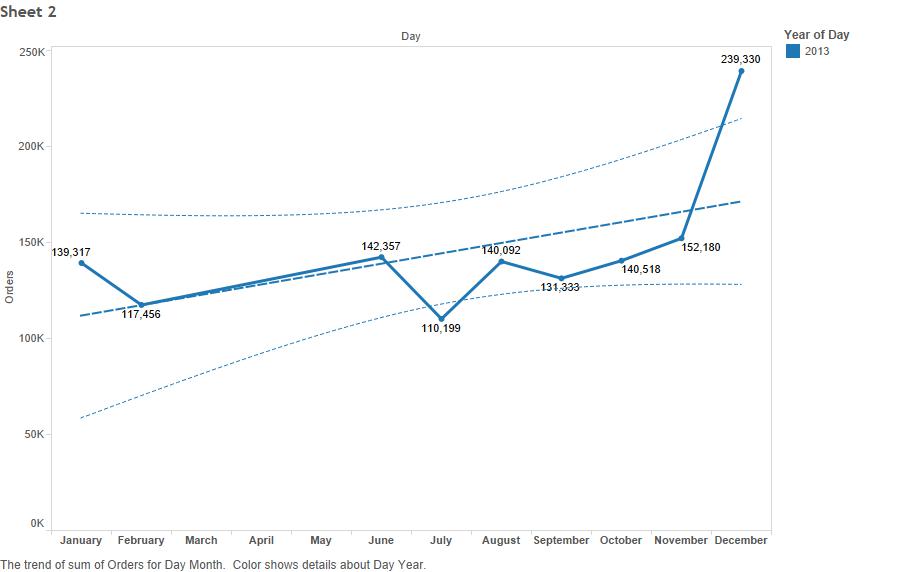
**Site Insights**



The above bar chart plots the Site vs. Orders & Site vs. Duration. The reports aggregates the metrics orders & duration for the whole year based on the dimension Site. This would give an insight as to which particular Site was responsible for increased orders and which particular Site the user spent most of the time, so that the company can target these Site users in future. Here, durations is a new attribute introduced which was calculated by dividing the product page views count with the distinct sessions count. According to our report, Acme was the site that contributed to the maximum orders and maximum time spent by the customers.

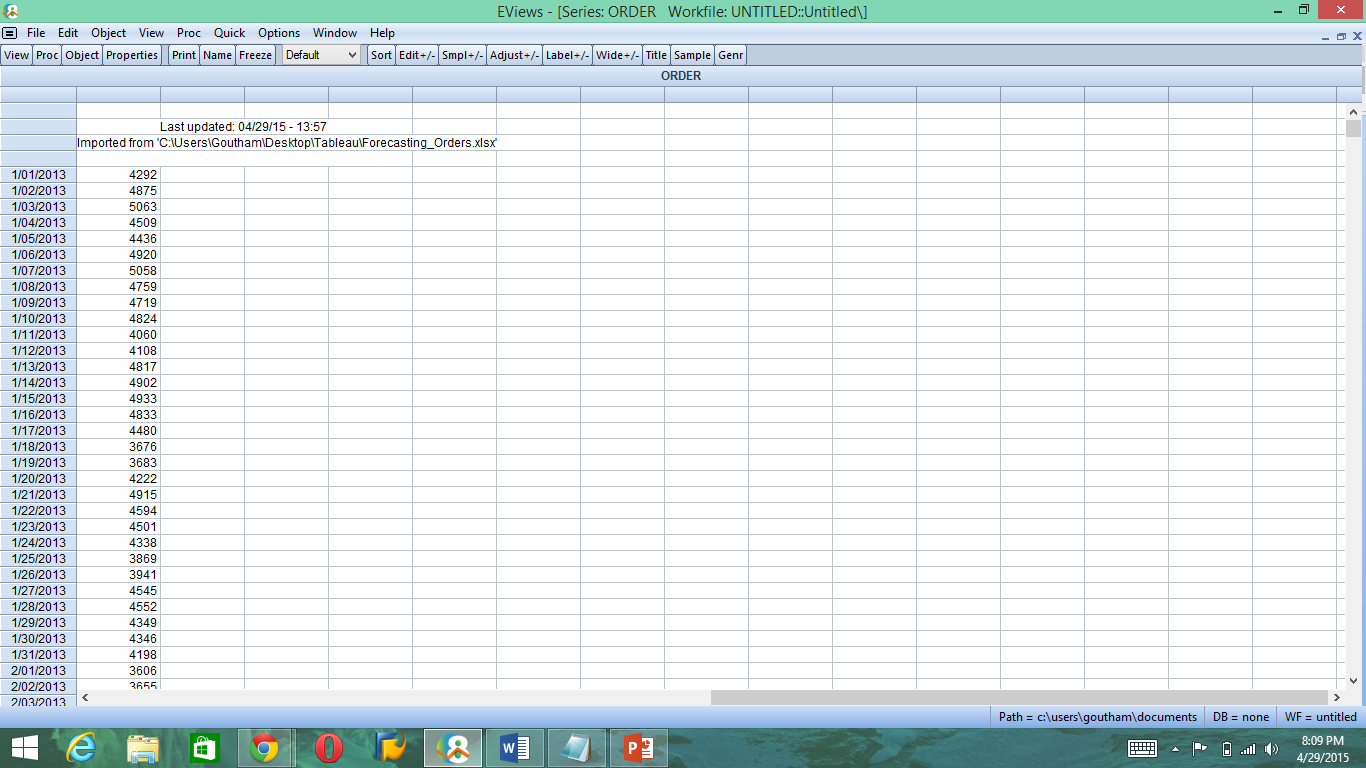
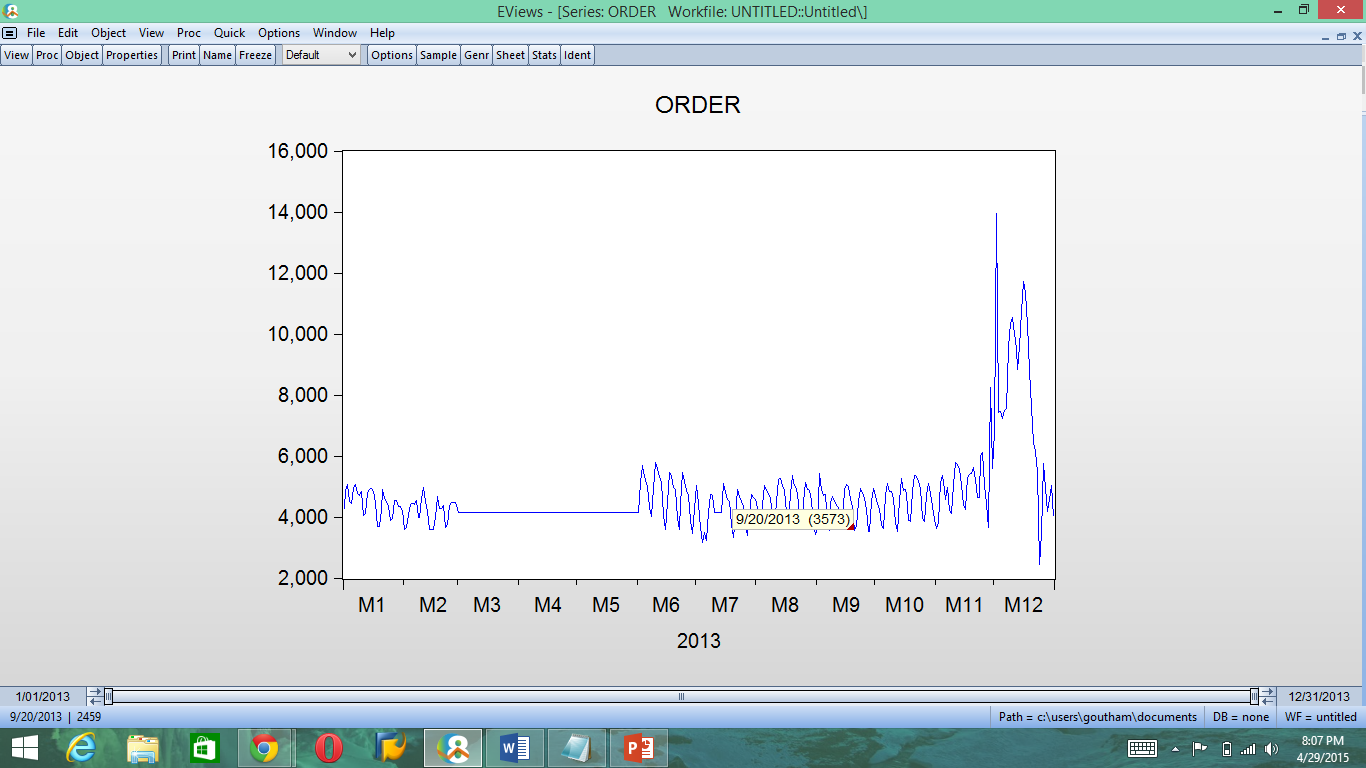
**Total Orders for every month in the year 2013 - With Trend Lines**

The following line graph gives us an insight of the aggregated orders for every month in the year 2013. Here, we get the understanding that the number of orders reaches its peak of nearly 240K during the December month and it may be because of the holiday season like Thanksgiving weekend.



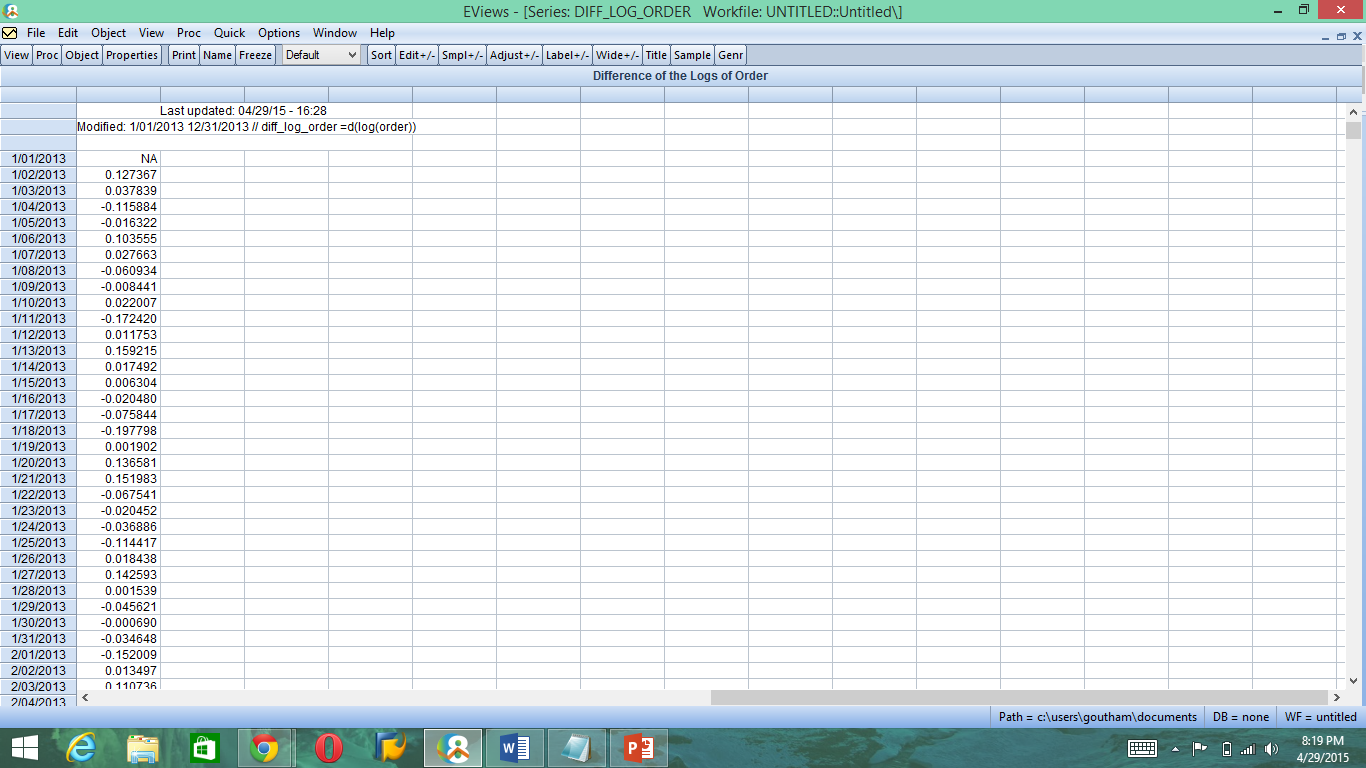
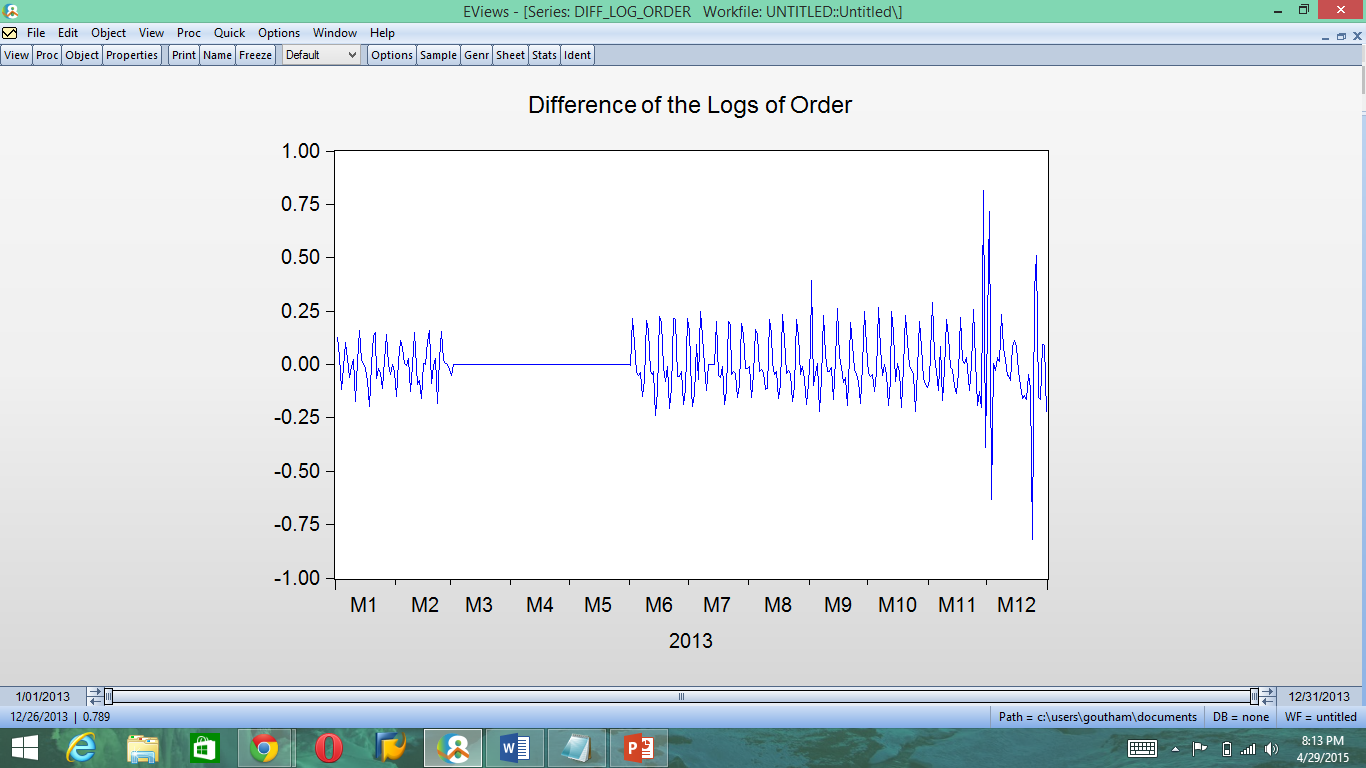
**Line graph for the orders on daily basis for the year 2013**

The following Line graphs displays the orders on daily basis for the year 2013. Here, we replace the missing values with the median. Also, there is a huge spike in the orders in the month of December because of the holiday season like Thanksgiving weekend (Black Friday and Cyber Monday)



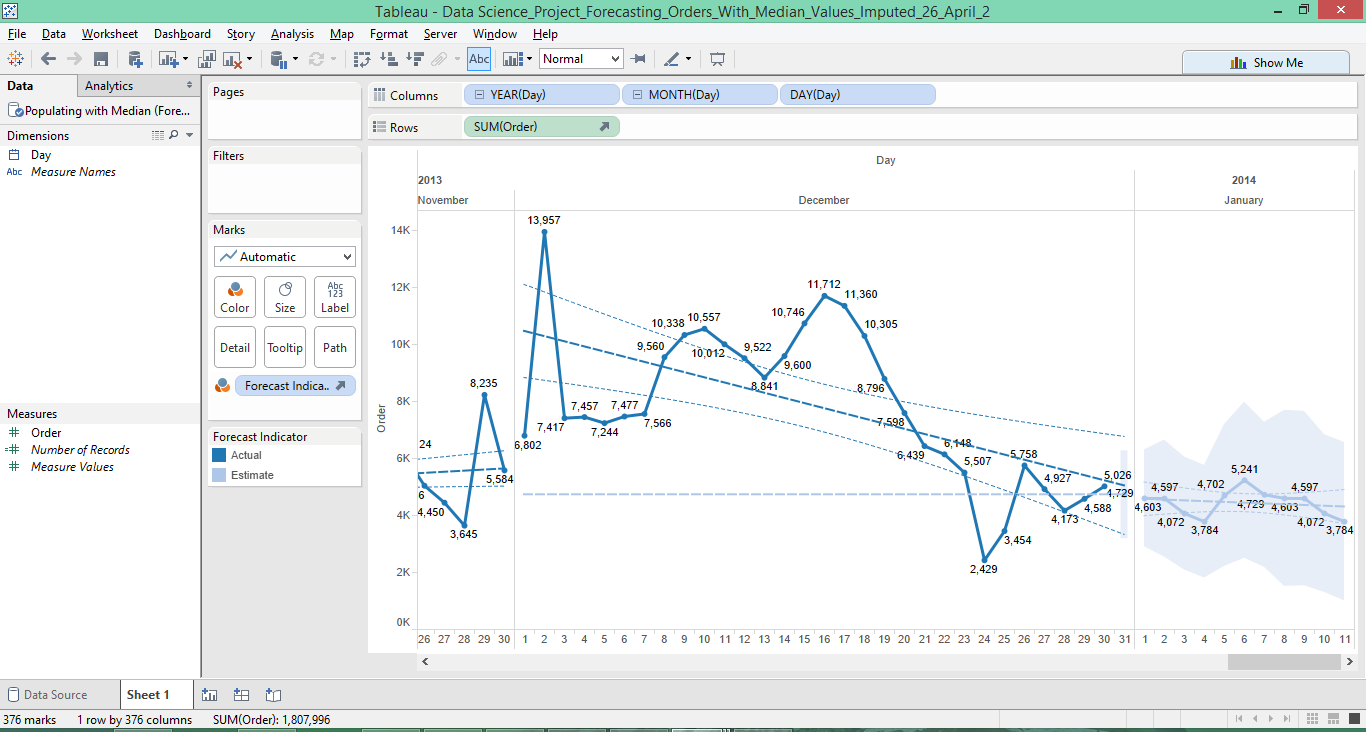
**Line graph for the difference of the logs of orders on daily basis for the year 2013**

The following Line graphs displays the difference of logs of orders on daily basis for the year 2013. Here, we see that the values oscillates about the mean on both the positive and negative direction. This would mitigate any significant trend if present.

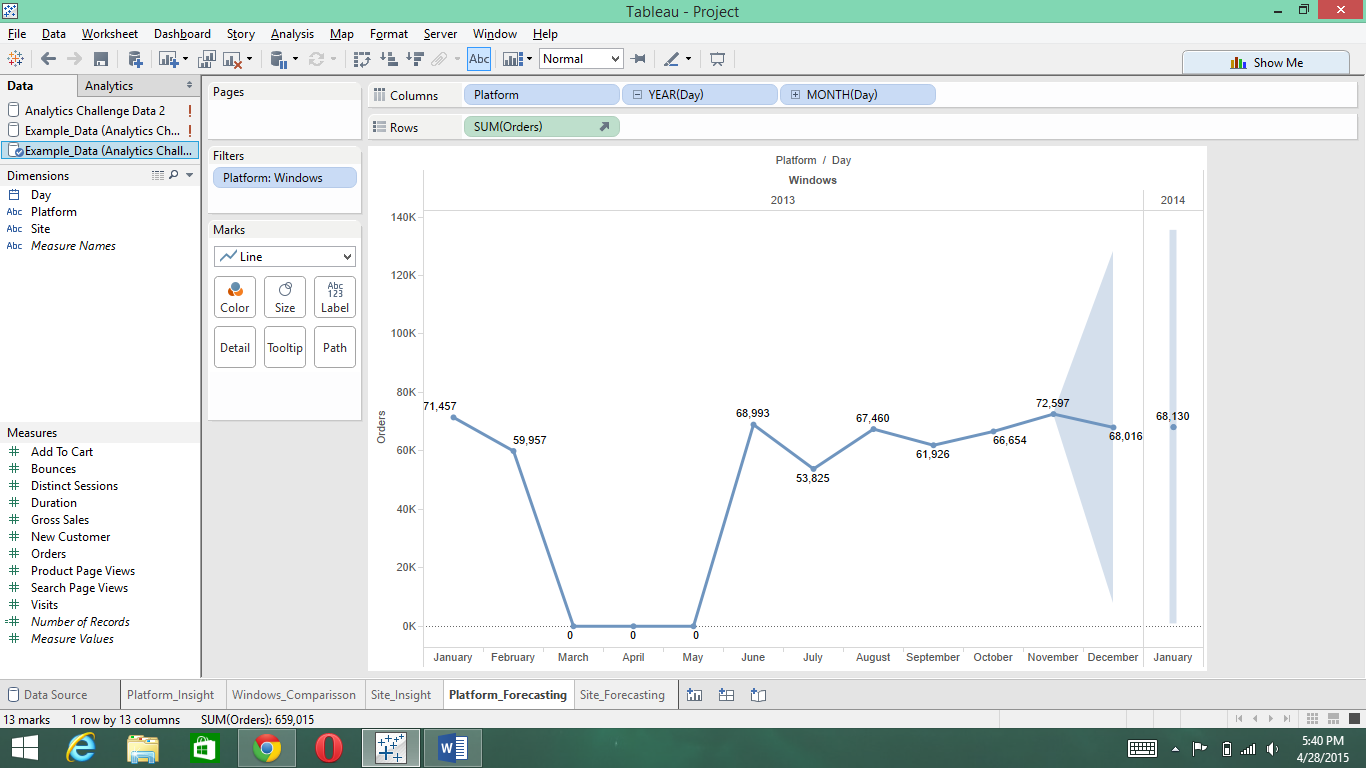


**Checking for Trend and Seasonality**

After taking the difference of log it would mitigate any significant trend if available. We observe that there is significant seasonality observed. The model that we chose was SARMA (4, 1) X (3, 0) S=7 with seasonality observed at the 7th, 14th and 21st lag. With the median values imputed for the missing values and applying the model on the data set, the orders were forecasted for the first 11 days of January. It was in the range of 3700 and 5200.



The following graph forecasts the value of orders for Windows phone for the month of January 2014 with no values imputed for the missing values. Approximately, around 68k would the orders count for the month of January.



To conclude, we were able to use various models on our data set and come up with forecasting results for Orders for the next year 2014. By analyzing these clickstream analytics report, it would help any product company to target segments of customers based upon metrics like platform, site, time etc. to maximize their gross sales and in turn profits.

References:

<http://mathbits.com/MathBits/TISection/Statistics2/correlation.htm>

<http://www.statmethods.net/stats/rdiagnostics.html>

<http://scikit-learn.org/0.5/auto_examples/svm/plot_svm_regression.html>