Multiple Linear Regression on Facebook Social Media Metrics

Team #4: Shengyi Liang, Kaiyi Bi, Mohao Yi, Rong Jiang

Abstract

In this project, we have created a model depending on the dataset related to the published posts on a well-known cosmetic brand during the year of 2014 on Facebook. The motivation of our research is to find the effectiveness of the advertisement on Facebook. Graphic evaluation is considered to be a good strategy to analyze data. Therefore, throughout our research, we use residual plots, histograms of residuals, normal Q-Q plots, and real-fitted value plots to evaluate the performance and the validity of the proposed model. After careful analyses, we build a linear model and derive that the performance of the posts is explicitly related with the media type and the posted month of the post, as well as whether the company has paid for promotion.

1. Introduction

Facebook, as one of the most popular social media platforms, gains overwhelming popularity among young people. The huge number of registered users are treated as potential customers by various companies. They expect that posting videos or images of the products on the main Facebook page would expose the products to a wider group of potential customers. We are curious about the important components of effectiveness of the advertisements on Facebook. In order to examine it, we construct a model by using the data of multiple attributes of posts published during the year of 2014 on the Facebook's page of a renowned cosmetics brand with R. If such approach succeeds, we could use that model to improve and predict the effectiveness of the advertisement with greatest consequent. In general, we aim to use the Multi-Linear Regression to construct a fitted model that can clearly link the relationship between the number of consumers and some features of the posts.

2. Background

Multiple linear regression is a valid model for the data and can determine whether there exists a good regression model between our predictor variables and response variable. Accordingly, we can implicitly make a series of assumptions. The data was collected randomly on posts of Facebook with 19 attributes that are related to each post. We wish to identify important and deterministic attributes that affect performance of posts with respect to commercial propagation.

Variables	**Description**
Lifetime.Post.Consumer	the number of people who clicked anywhere in the post
Type	type of content: video, link, photo, status
Category	content characterization
Page.Total.Like	number of people who liked the company's page
Post.month	the month the post was published
Post.weekday	weekday the post was published
Post.hour	hour the post was published
Paid	if the company paid to Facebook for advertising or not (yes/no)

Figure 1.Variables and Descriptions

Mohao Yi--Model Analysis, diagnostics, and editing report

Shengyi Liang--Coding, diagnostics, and editing report

Kaiyi Bi--Organizing and writing report

Rong Jiang--Trivial Tests and organizing plots

Response(Y)	**Dependent Variable(X’s)**	
Lifetime Post Consumers (people who clicked anywhere on the post, continuous)	Categorical Variables	Continuous Variables
	Type	Page Total Likes
	Category	
	Post Month	
	Post Weekday	
	Post Hour	
	Paid	

Figure 2. Potential Explanatory Variables

3. Modeling and Analysis

In this section, we will describe the key steps of building our model, and we have split the data evenly into two halves. Here, we focus on one of them to keep modifying and training our model.

3.1 Variable transformation

For a linear model, the normality assumption can be considered to be the priority, since it justifies the use of t tests and F tests. Clearly, the distribution of the response variable would be a shifted version of the distribution of the errors (if the covariates are considered to be deterministic; and for this dataset, this is reasonable since a company can always choose the environment and features of a post by itself). Therefore, before we actually begin to build the model, we had better first look at the distribution of the response variable. If it is not normal, we had better transform it in some way.

Before **After**

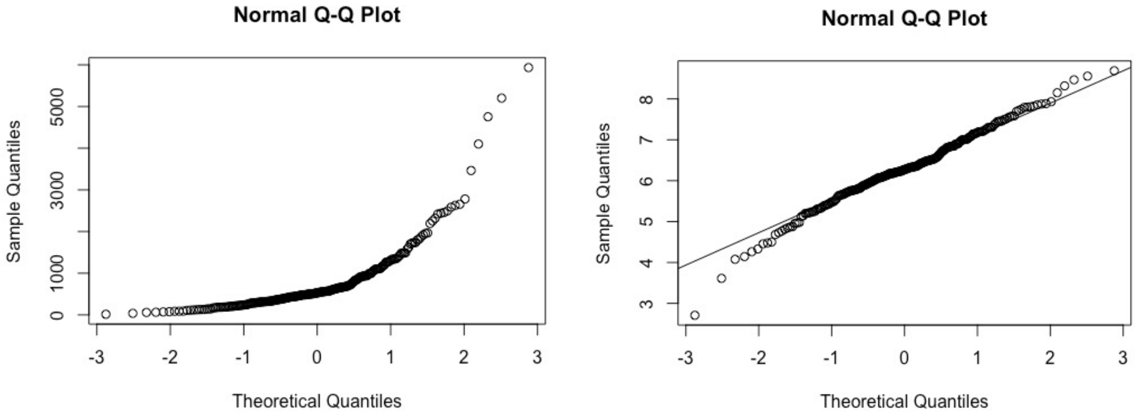


Figure 3. Q-Q plot of Lifetime Post Consumers before and after log Transformation

The left Q-Q plot in Figure 3 shows the relation between the sample quantile of our response and a theoretical population quantile of standardized normal distribution. The curving pattern provides strong evidence that the original response does not follow a normal distribution. Therefore, we perform the traditional log transformation on it to increase the normality, and the result is shown in the right plot above. We can see that the log transformation works relatively well.

3.2 Variable selection

Next, we need to decide which covariates are so important that they should be included in the model. Note that here, the p values of t tests may not be that informative, since we do not know whether the normal assumption holds. Therefore, we prefer to use an information criterion, and AICc is the one that we are going to use. The reason that we have to use the corrected version of AIC is that our dataset is small. After splitting the data, we only have 249 sample points in our training set, and 249/7 (the maximum number of potential important covariates) is less than 40, which falls into the range where AICc must be used instead of AIC.

Size 1 Model	AICc	Size 2 Model (Type+)	AICc
1	651	Post Month	539
Post Month	638	Post Weekday	605
Post Weekday	661	Post Hour	603
Post Hour	670	Category	598
Type	595	Page Total Likes	564
Category	650	Paid	592
Page Total Likes	846		
Paid	650		

Size 3 Model (Type+Post Month+)	AICc	Size 4 Model (Type+Post Month+Paid+)	AICc
Post Weekday	551	Post Weekday	550
Post Hour	561	Post Hour	562

Category	539	Category	538
Page Total Likes	541	Page Total Likes	540
Paid	538		

Figure 4. Forward AICc in Different Models

The table in Figure 4 shows the how AICc leads us to the proposed model. We use the forward selection method which requires us to include the covariate with the lowest AICc value for each iteration, until the covariates are exhausted or there is no more better covariates that can lower the AICc value. Note that in the process, the test for intercept is not always included, since we choose to use dummy variable regression to deal with the categorical covariates, and thus, the intercept is not necessarily needed in the model. And as shown above, Type, Paid, and Month are finally included in our model. As shown by Figure 5 below, within our model, none of the variable has collinearity with another variable. For actual output of linear model, please refer to Appendix 1. Note that we included Paid even if its p-value is a little bit greater than 0.05 since commercial promotion is highly related to the spread of commercial products as a relatively compulsory propagation.

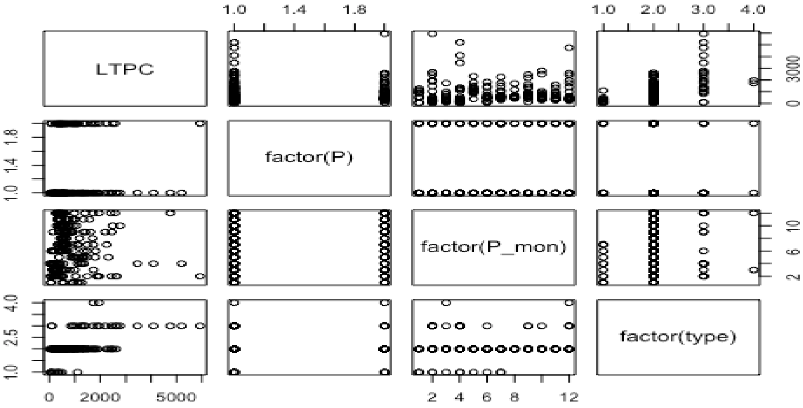


Figure 5. Scatterplot Matrix of Selected Variables

3.3 Diagnosis

Since we have found a tentative proposed model, the next step would be checking whether it satisfies the assumptions for linear models, and making some adjustments accordingly, if needed.

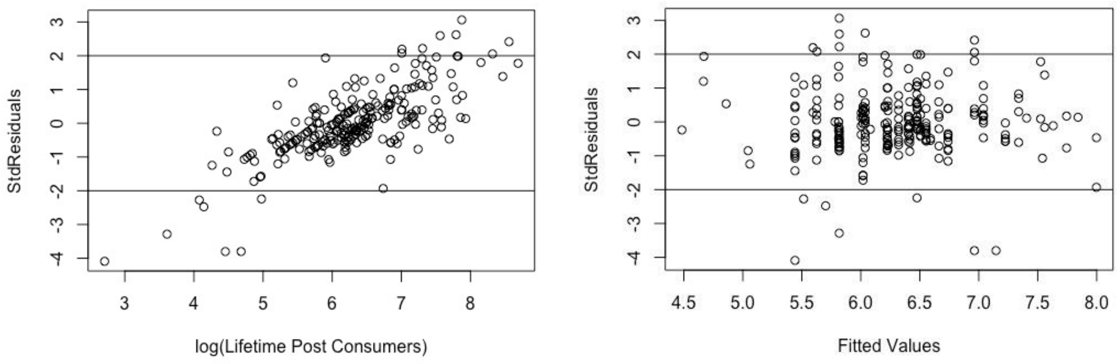


Figure 6. Comparison of Standardezed Residuals vs Fitted/Real Values

First, we use the plots in Figure 6, we can see that points are roughly centered at 0, and their absolute deviation from 0 is almost constant. In other words, the standardized residuals are evenly distributed, thereby satisfying moment assumption. (Although we can observe that the points in the left plot shows a linear pattern, the randomness of the points in the right plot indicates that our model is in good condition) Moreover, the fact that the deviations of most of these points are within [-2, 2] indicates that these noises may follow a standard normal distribution.

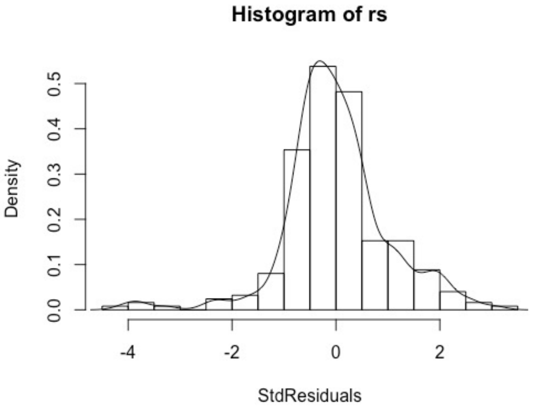


Figure 7. Histogram of standard residuals

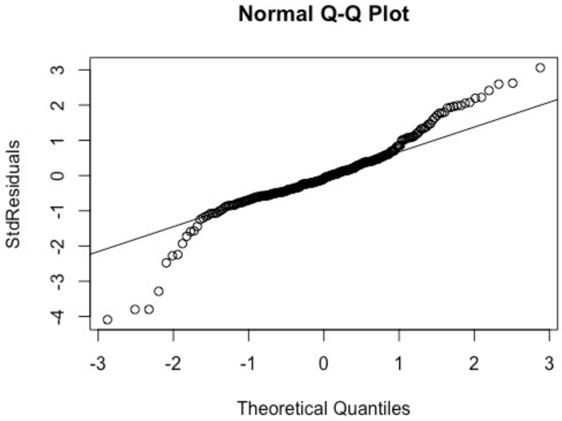


Figure 8. Normal Q-Q Plot

Next, to actually test whether the model follows the normal assumption, we can refer to Figure 7 and 8 above. In Figure 7, we can see that the density is bell shaped when the standardized residuals are around 0, while there exists right skewness.In Figure 8, similar to what we have found through Figure 7, the residuals around 0 shows strong normality, as they almost form a straight line. However, in the two tails, there exist tails that deviate from the straight line. We have to admit that the normality assumption is really important for models. However, in practice, we cannot always obtain a model that totally satisfies the normality assumption. Thus, it is reasonable to accept this model, but we still have to keep in mind that we can only accept this model to some extent.

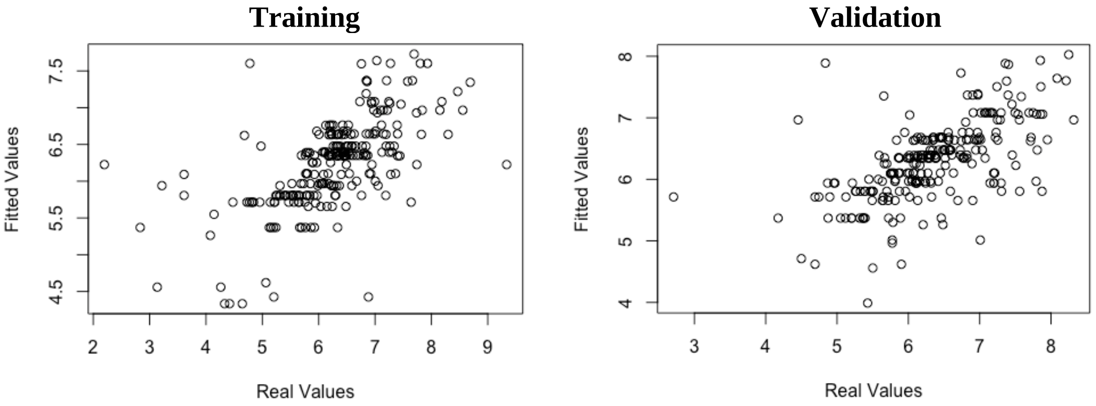
In fact, we have performed Weighted Least Squares on the model, aiming to reduce the amount of outliers shown in the residual plots of Figure 6. However, after looking at the result, we realized that improvement is only a little so we keep the original model in case the model would be overfitted. (To gain more information about our WLS, please refer to the discussion section.)

Conclusively, we finalize the model to the following form:
, this is just the model proposed by AICc

Again, for actual coefficients, please refer to Appendix 1.

4. Validation and Prediction

In the previous section, we have proposed a model that fits the training dataset well. However, it is possible that the model is not helpful for the prediction, since its good performance may result from the fact that the model is capturing the pattern of the random errors. To ensure that the proposed model is not overfitted in that way and gives us good prediction of the response, we use the remaining half of the data to validate it. The following plots of fitted values against real values visualize whether the performance of our model decays when being applied to some raw data outside the training dataset. You can see that when we shift to use our model to predict the response with the covariates in the validation set, the linear relationship between the fitted values and the real values is almost preserved. As for a good model, the fitted values that it gives must be predicted close enough to the corresponding real values. Thus, the maintenance of such a linear pattern really makes us feel more confident in our model.


Training

Validation

Figure 9. Fitted vs Real Values

However, it is not enough to just look at such plots, since they can only provide some rough feelings about the performance of the model. To justify it precisely, we need some numeric measurements. The ones that we use here are the percentage change of MSE and RMSE when shifting our model from being applied to the training dataset to the validation dataset. By definition, MSE is the average of the squared difference between the real values and the fitted values; RMSE is MSE divided by the average of squared real value. They both measure the deviation of the fitted values from the real values, while in addition, RMSE takes the size of the real values into account since it is possible that the mean error of one dataset is greater than the others, but if that dataset also has larger real values, the relative error might be close to the others. If the MSE and RMSE increase significantly when shifting from training set to the validation set, we may consider the model being tested is overfitted since it does not fit well to new, raw data. As shown in the last row of the following table, the percentage change of MSE and RMSE from training to validation is roughly 30%, which is within an acceptable range, and again, this justifies what the plots in Figure 9 shows. Therefore, we can conclude that our model is not overfitted and it passes the validation test.

	MSE	RMSE
Training	0.4380	0.0696
Validation	0.5817	0.0922
Change (%)	32.81	32.47

Figure 10. Table of MSE and RMSE

Since the validation tells us that our model is suitable for prediction, it is legitimate to talk about predicting the real values. Again, according to the MSE and RMSE for the validation dataset, the average error is about √0.6 = 0.77 and the relative error with respect to the real value is about 0.3. These indicate that on average, the fitted value of our model roughly deviates from the real value by ±0.77, and 0.77 is roughly 30% of the real value. After getting the prediction we want, we should be aware that the fitted value is likely to deviate from the real value by that much. Also, recall that when building the model, we transformed the response, Lifetime.post.consumers, using the log() function.

Therefore, to get the actual desired result, we need to use the exponential function to transform our predicted value back.

5. Discussion

Obviously, media type and posted month of posts along with paid promotion are the most deterministic and essential attributes of posts and affect future commercial effectiveness of propagational posts. Among them, status posts with promotion that are posted in April usually attract more viewers to click on them. Thus, for companies who would like to promote and spread their products, they should post more status posts and pay for Facebook promotions while avoiding posting too many posts in Oct, Nov, and Dec since those months have relatively lower coefficients.

Summary of problems:

1. Small sample size

We are given 500 data and each of them have 19 attributes. For modeling, 500 is relatively small to make assumptions such as normality. Moreover, due to operation of training and validation, 250 is a small dataset to cover all combinations of categorical variables so that some predictions may not be surjective for all posts on Facebook.

2. Unclear lm() function generates confusion

While using linear model to predict on validation dataset, all vectors must be re-assigned; otherwise, data from training dataset may be re-used again thereby causing incorrect output such as abnormal values of MSE.

3. Extend the model to non-linear types

According to linearity of points in plot of standard residuals versus log transformed Lifetime Post Consumers, we are still missing terms in our model. Thus, it is possible that the actual model of prediction is not linear. For example, since advertising through media depend on its rate of spreading, consumers who have clicked anywhere on the post should be part of their derivative as well since the more people who have clicked on the post, the faster the post spread. Therefore, Lifetime Post Consumers may be exponentially distributed and it corresponds the Q-Q plot of real values of Lifetime Post Consumers.

4. WLS

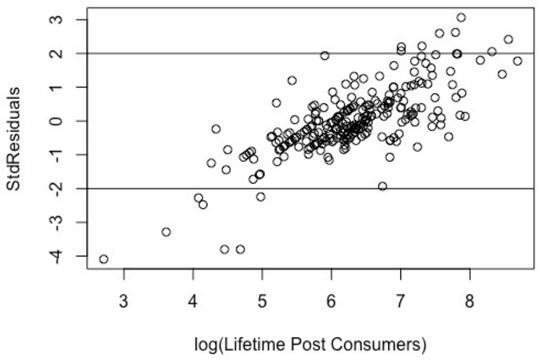


Figure 11. without WLS

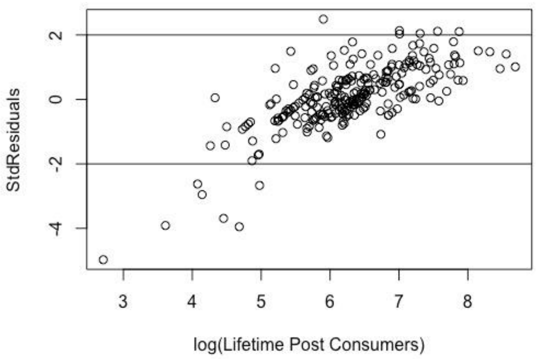
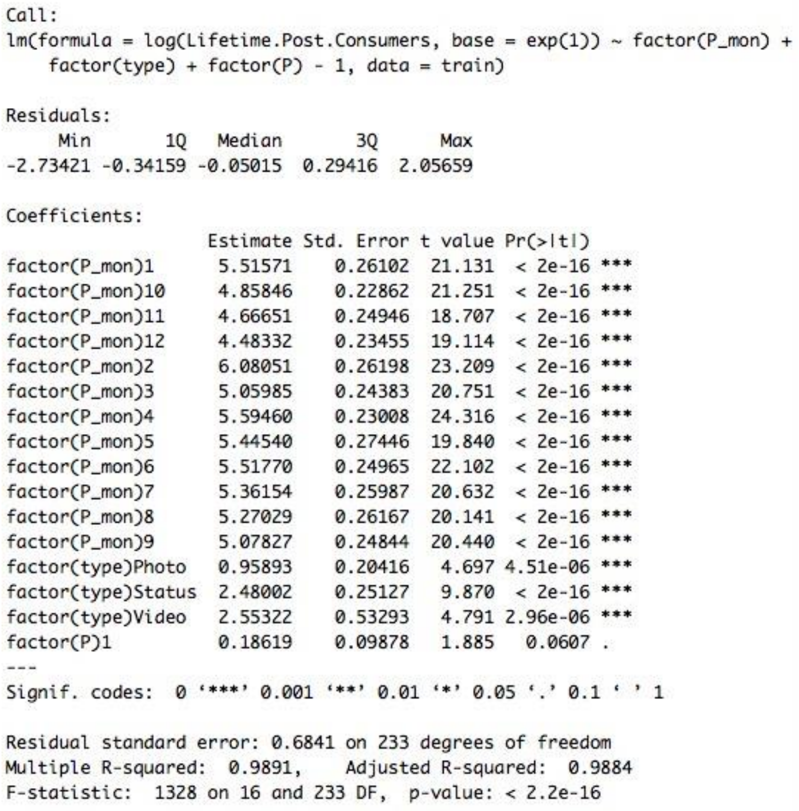


Figure 12. with WLS

We tried to implement the Weighted Least Square to improve performance of the model. In doing so, we hope to make our standardized residuals look more normal and condensed around y=0. However, after comparing plots of standard residuals with and without WLS, we found that WLS only improve the distribution of standard residuals slightly. Thus, with respect to reliability of the model, we keep the model without WLS.

6 Appendix

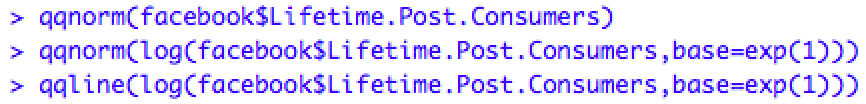
1:



```
Call:
lm(formula = log(Lifetime.Post.Consumers, base = exp(1)) ~ factor(P_mon) +
    factor(type) + factor(P) - 1, data = train)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.73421 -0.34159 -0.05015  0.29416  2.05659 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
factor(P_mon)1      5.51571    0.26102  21.131  < 2e-16 ***
factor(P_mon)10     4.85846    0.22862  21.251  < 2e-16 ***
factor(P_mon)11     4.66651    0.24946  18.707  < 2e-16 ***
factor(P_mon)12     4.48332    0.23455  19.114  < 2e-16 ***
factor(P_mon)2      6.08051    0.26198  23.209  < 2e-16 ***
factor(P_mon)3      5.05985    0.24383  20.751  < 2e-16 ***
factor(P_mon)4      5.59460    0.23008  24.316  < 2e-16 ***
factor(P_mon)5      5.44540    0.27446  19.840  < 2e-16 ***
factor(P_mon)6      5.51770    0.24965  22.102  < 2e-16 ***
factor(P_mon)7      5.36154    0.25987  20.632  < 2e-16 ***
factor(P_mon)8      5.27029    0.26167  20.141  < 2e-16 ***
factor(P_mon)9      5.07827    0.24844  20.440  < 2e-16 ***
factor(type)Photo   0.95893    0.20416   4.697 4.51e-06 ***
factor(type)Status  2.48002    0.25127   9.870  < 2e-16 ***
factor(type)Video   2.55322    0.53293   4.791 2.96e-06 ***
factor(P)1          0.18619    0.09878   1.885   0.0607 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6841 on 233 degrees of freedom
Multiple R-squared:  0.9891,	Adjusted R-squared:  0.9884 
F-statistic:  1328 on 16 and 233 DF,  p-value: < 2.2e-16
```

2:

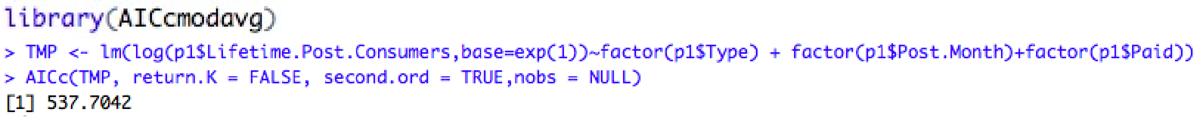
Figure 3:



```
> qqnorm(facebook$Lifetime.Post.Consumers)
> qqnorm(log(facebook$Lifetime.Post.Consumers,base=exp(1)))
> qqline(log(facebook$Lifetime.Post.Consumers,base=exp(1)))
```

3:

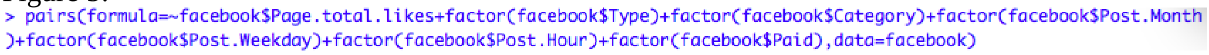
Figure 4 (AICc):



```
library(AICcmodavg)
> TMP <- lm(log(p1$Lifetime.Post.Consumers,base=exp(1))~factor(p1$Type) + factor(p1$Post.Month)+factor(p1$Paid))
> AICc(TMP, return.K = FALSE, second.ord = TRUE,nobs = NULL)
[1] 537.7042
```

4:

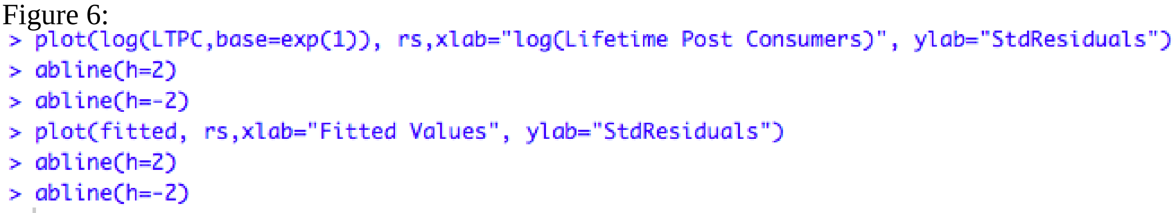
Figure 5:



```
> pairs(formula=~facebook$Page.total.likes+factor(facebook$Type)+factor(facebook$Category)+factor(facebook$Post.Month
)+factor(facebook$Post.Weekday)+factor(facebook$Post.Hour)+factor(facebook$Paid),data=facebook)
```

5:

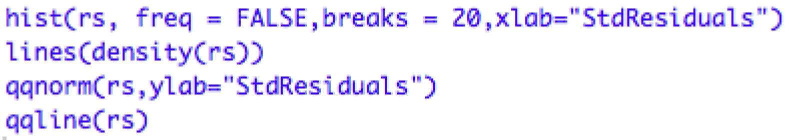
Figure 6:



```
> plot(log(LTPC,base=exp(1)), rs,xlab="log(Lifetime Post Consumers)", ylab="StdResiduals")
> abline(h=2)
> abline(h=-2)
> plot(fitted, rs,xlab="Fitted Values", ylab="StdResiduals")
> abline(h=2)
> abline(h=-2)
```

6:

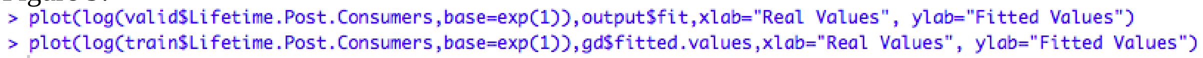
Figure 7 & 8:



```
hist(rs, freq = FALSE,breaks = 20,xlab="StdResiduals")
lines(density(rs))
qqnorm(rs,ylab="StdResiduals")
qqline(rs)
```

7:

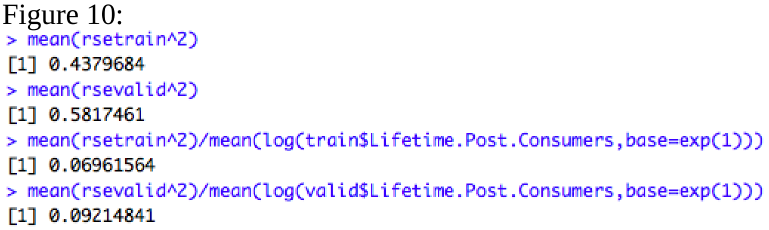
Figure 9:



```
> plot(log(valid$Lifetime.Post.Consumers,base=exp(1)),output$fit,xlab="Real Values", ylab="Fitted Values")
> plot(log(train$Lifetime.Post.Consumers,base=exp(1)),gd$fitted.values,xlab="Real Values", ylab="Fitted Values")
```

8:

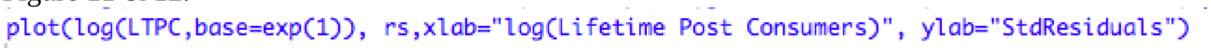
Figure 10:



```
> mean(rsetrain^2)
[1] 0.4379684
> mean(rsevalid^2)
[1] 0.5817461
> mean(rsetrain^2)/mean(log(train$Lifetime.Post.Consumers,base=exp(1)))
[1] 0.06961564
> mean(rsevalid^2)/mean(log(valid$Lifetime.Post.Consumers,base=exp(1)))
[1] 0.09214841
```

9:

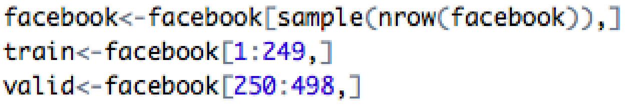
Figure 11 & 12:



```
plot(log(LTPC,base=exp(1)), rs,xlab="log(Lifetime Post Consumers)", ylab="StdResiduals")
```

10:

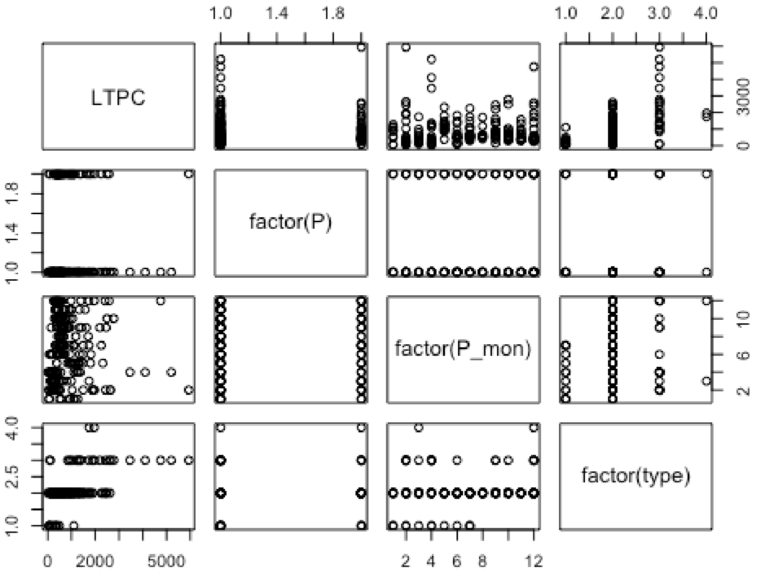
Spliting Data into Training and Validation:



```
facebook<-facebook[sample(nrow(facebook)),]
train<-facebook[1:249,]
valid<-facebook[250:498,]
```

11:

ScatterplotMatrix of final model:


LTPC
factor(P)
factor(P_mon)
factor(type)