R Notebook

Project

Predicting approved conversions

```
#importing libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(car)
```

```
## Warning: package 'car' was built under R version 4.1.1

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```
df = read.csv("dataset.csv",header=TRUE)
```

```
head(df)
```

```
##    ad_id xyz_campaign_id fb_campaign_id   age gender interest Impressions
## 1 708746             916         103916 30-34      M       15        7350
## 2 708749             916         103917 30-34      M       16       17861
## 3 708771             916         103920 30-34      M       20         693
## 4 708815             916         103928 30-34      M       28        4259
## 5 708818             916         103928 30-34      M       28        4133
## 6 708820             916         103929 30-34      M       29        1915
##   Clicks Spent Total_Conversion Approved_Conversion
## 1      1  1.43                2                   1
## 2      2  1.82                2                   0
## 3      0  0.00                1                   0
## 4      1  1.25                1                   0
## 5      1  1.29                1                   1
## 6      0  0.00                1                   1
```

Checking null values

```
lapply(df,function(x) { length(which(is.na(x)))})
```

```
## $ad_id
## [1] 0
##
## $xyz_campaign_id
## [1] 0
##
## $fb_campaign_id
## [1] 0
##
## $age
## [1] 0
##
## $gender
## [1] 0
##
## $interest
## [1] 0
##
## $Impressions
## [1] 0
##
## $Clicks
## [1] 0
##
## $Spent
## [1] 0
##
## $Total_Conversion
## [1] 0
##
## $Approved_Conversion
## [1] 0
```

```
summary(df)
```

```
##      ad_id         xyz_campaign_id fb_campaign_id       age
##  Min.   : 708746   Min.   : 916    Min.   :103916   Length:1143
##  1st Qu.: 777633   1st Qu.: 936    1st Qu.:115716   Class :character
##  Median :1121185   Median :1178    Median :144549   Mode  :character
##  Mean   : 987261   Mean   :1067    Mean   :133784
##  3rd Qu.:1121805   3rd Qu.:1178    3rd Qu.:144658
##  Max.   :1314415   Max.   :1178    Max.   :179982
##     gender             interest       Impressions          Clicks
##  Length:1143        Min.   :  2.00   Min.   :     87   Min.   :  0.00
##  Class :character   1st Qu.: 16.00   1st Qu.:   6504   1st Qu.:  1.00
##  Mode  :character   Median : 25.00   Median :  51509   Median :  8.00
##                     Mean   : 32.77   Mean   : 186732   Mean   : 33.39
##                     3rd Qu.: 31.00   3rd Qu.: 221769   3rd Qu.: 37.50
##                     Max.   :114.00   Max.   :3052003   Max.   :421.00
##      Spent        Total_Conversion Approved_Conversion
##  Min.   :  0.00   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:  1.48   1st Qu.: 1.000   1st Qu.: 0.000
##  Median : 12.37   Median : 1.000   Median : 1.000
##  Mean   : 51.36   Mean   : 2.856   Mean   : 0.944
##  3rd Qu.: 60.02   3rd Qu.: 3.000   3rd Qu.: 1.000
##  Max.   :639.95   Max.   :60.000   Max.   :21.000
```

```
sp0 = df %>%
  filter(Spent == 0)
head(sp0)
```

```
##    ad_id xyz_campaign_id fb_campaign_id   age gender interest Impressions
## 1 708771             916         103920 30-34      M       20         693
## 2 708820             916         103929 30-34      M       29        1915
## 3 708979             916         103955 30-34      M       31        1224
## 4 709023             916         103962 30-34      M        7         735
## 5 709038             916         103965 30-34      M       16        5117
## 6 709040             916         103965 30-34      M       16        5120
##   Clicks Spent Total_Conversion Approved_Conversion
## 1      0     0                1                   0
## 2      0     0                1                   1
## 3      0     0                1                   0
## 4      0     0                1                   0
## 5      0     0                1                   0
## 6      0     0                1                   0
```

```
ap0 = df %>%
  filter(Approved_Conversion == 0)
head(ap0)
```

```
##    ad_id xyz_campaign_id fb_campaign_id   age gender interest Impressions
## 1 708749             916         103917 30-34      M       16       17861
## 2 708771             916         103920 30-34      M       20         693
## 3 708815             916         103928 30-34      M       28        4259
## 4 708889             916         103940 30-34      M       15       15615
## 5 708953             916         103951 30-34      M       27        2355
## 6 708958             916         103952 30-34      M       28        9502
##   Clicks Spent Total_Conversion Approved_Conversion
## 1      2  1.82                2                   0
## 2      0  0.00                1                   0
## 3      1  1.25                1                   0
## 4      3  4.77                1                   0
## 5      1  1.50                1                   0
## 6      3  3.16                1                   0
```

Initially, we thought this could not be possible. However, it turns out facebook can charge per click or per impression. Therefore, this data is not wrong.

```
length(unique(df$ad_id))
```

```
## [1] 1143
```

```
unique(df$xyz_campaign_id)
```

```
## [1]  916  936 1178
```

```
df$xyz_campaign_id = as.factor(df$xyz_campaign_id)
```

```
length(unique(df$fb_campaign_id))
```

```
## [1] 691
```

```
unique(df$age)
```

```
## [1] "30-34" "35-39" "40-44" "45-49"
```

```
unique(df$gender)
```

```
## [1] "M" "F"
```

```
unique(df$interest)
```

```
##  [1]  15  16  20  28  29  27  31   7  30  24  21  32  18  63  65  25  10  19  26
## [20]  36  23  64  22   2  66 100 101 102 103 105 107 110 111 112 113 108 109 114
## [39] 104 106
```

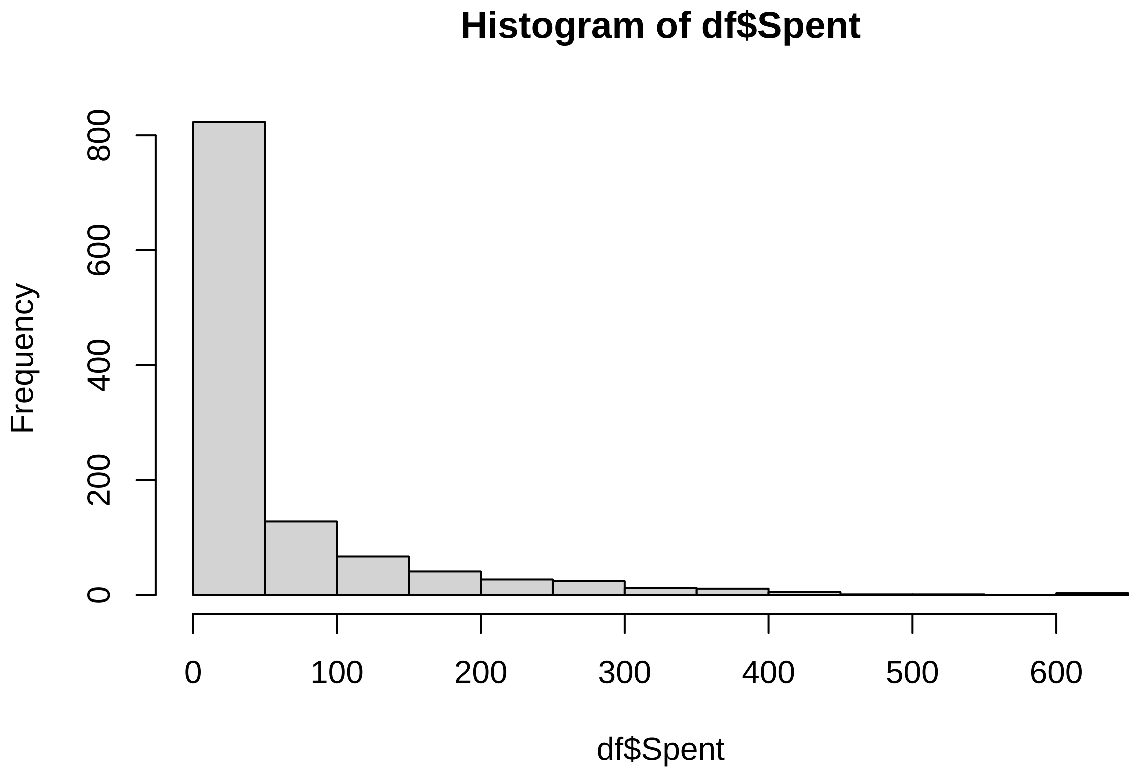
Assumption - interests will be in brackets - for example 0-25 could be tech, 26-50 could be sports, etc

```
df$interest=cut(df$interest,breaks = c(0,25,50,75,100,125),labels = c("0-25","26-50","51-75","76-100","
```

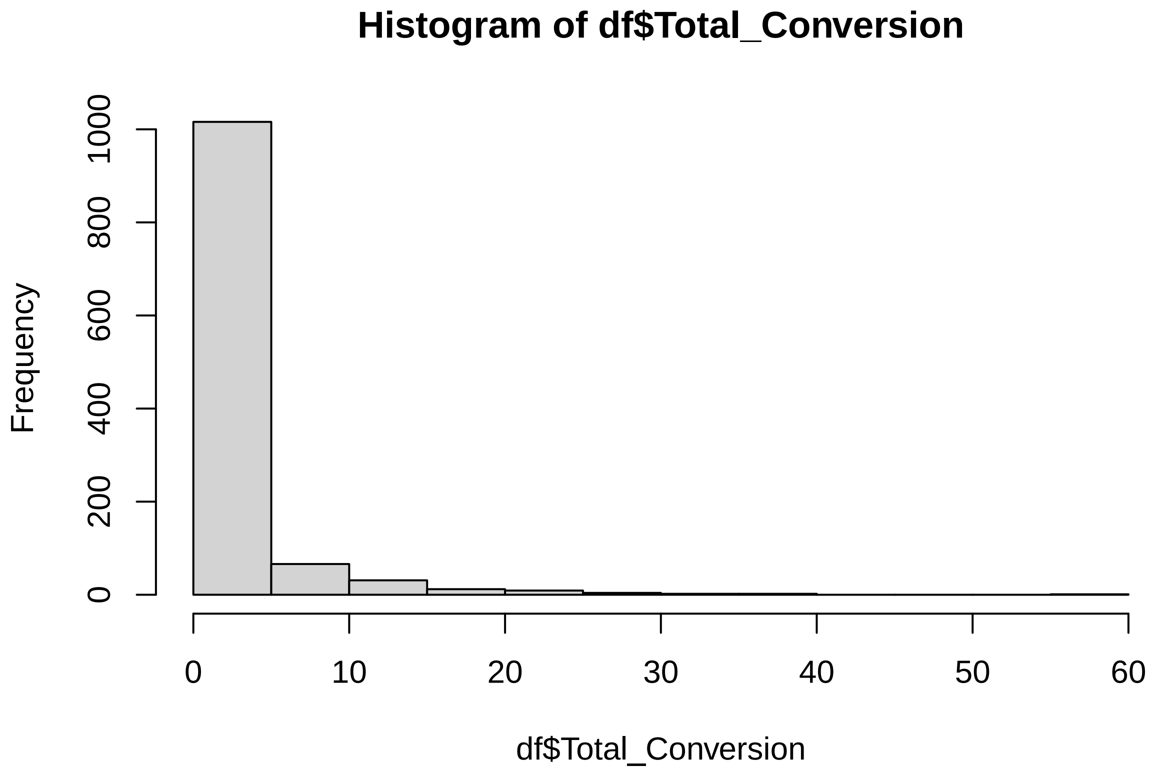
```
head(df$interest)
```

```
## [1] 0-25   0-25   0-25   26-50 26-50 26-50
## Levels: 0-25 26-50 51-75 76-100 101-125
```

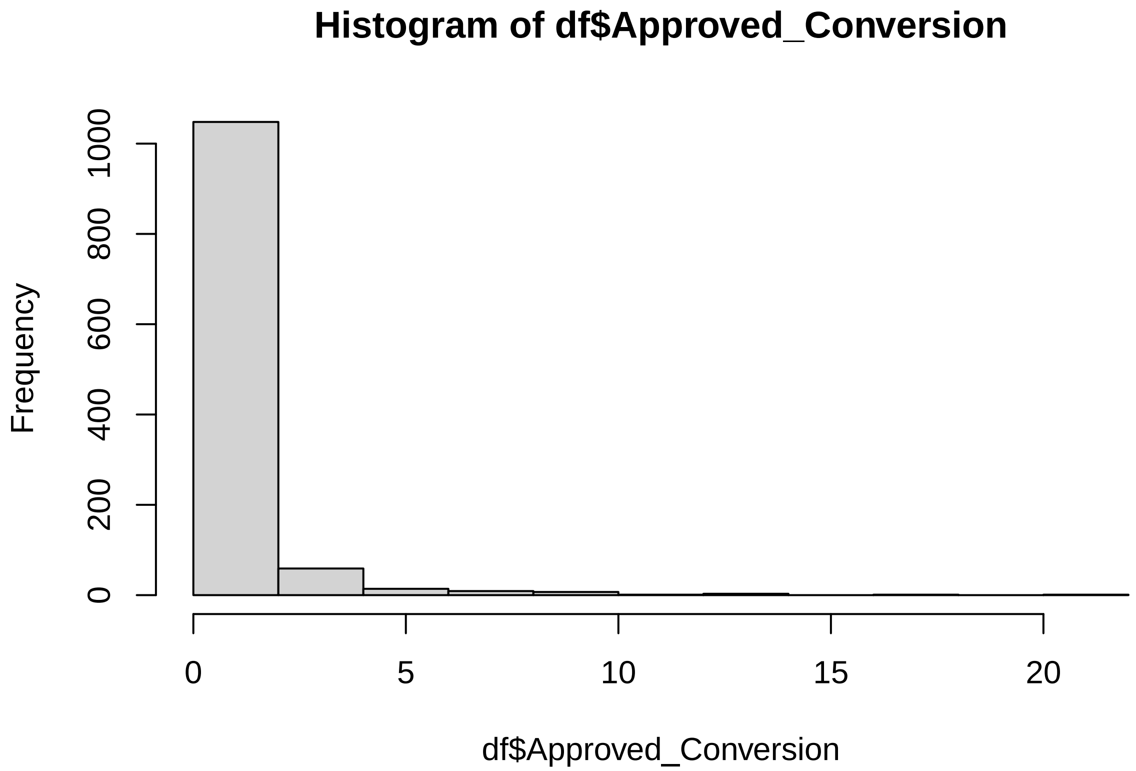
```
hist(df$Spent)
```



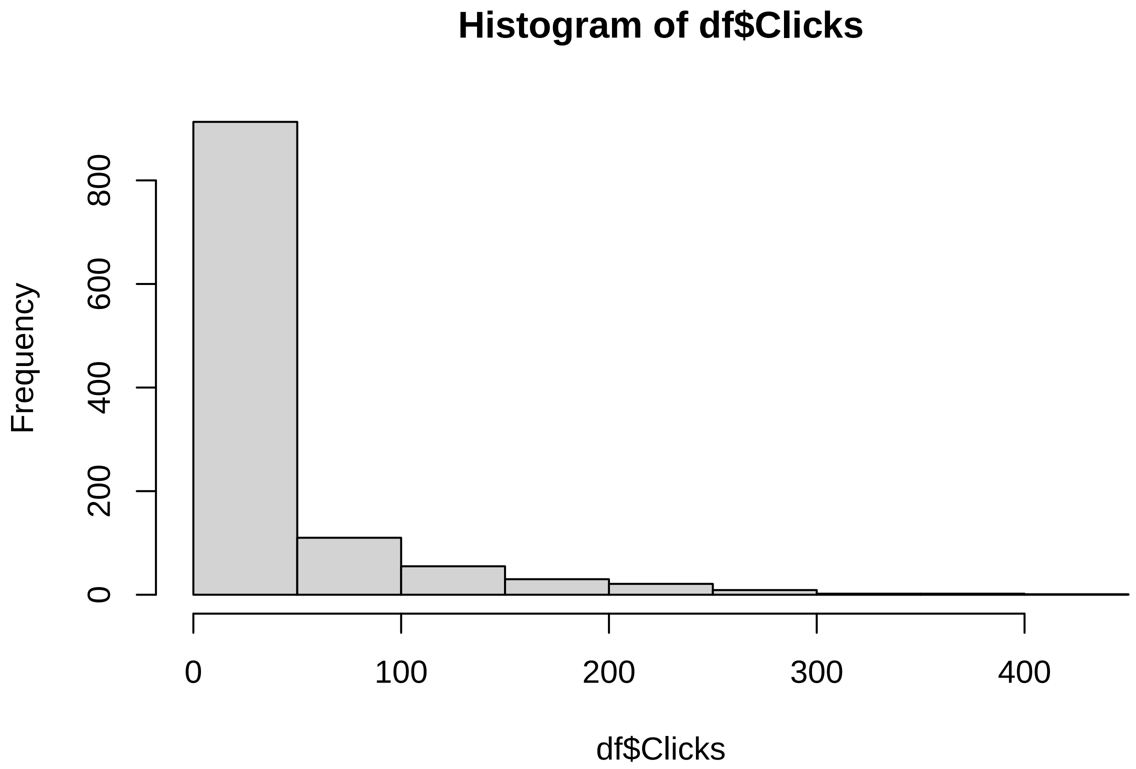
```
hist(df$Total_Conversion)
```



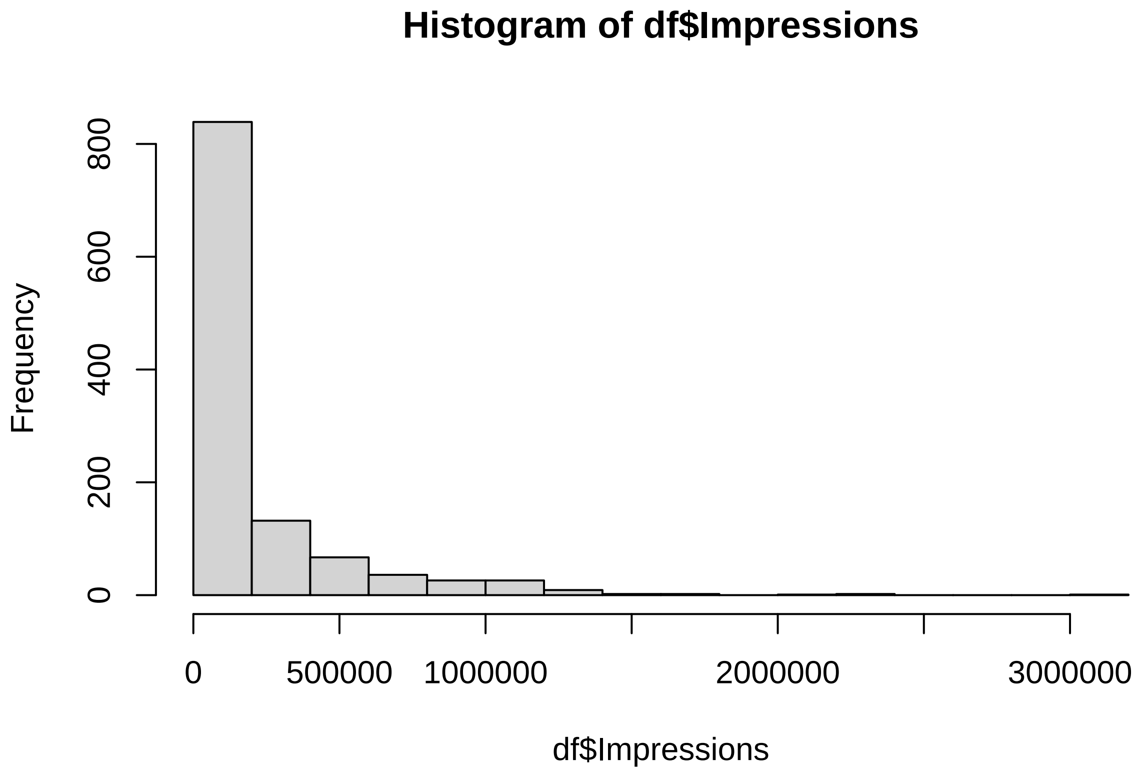
```
hist(df$Approved_Conversion)
```



```
hist(df$Clicks)
```



```
hist(df$Impressions)
```



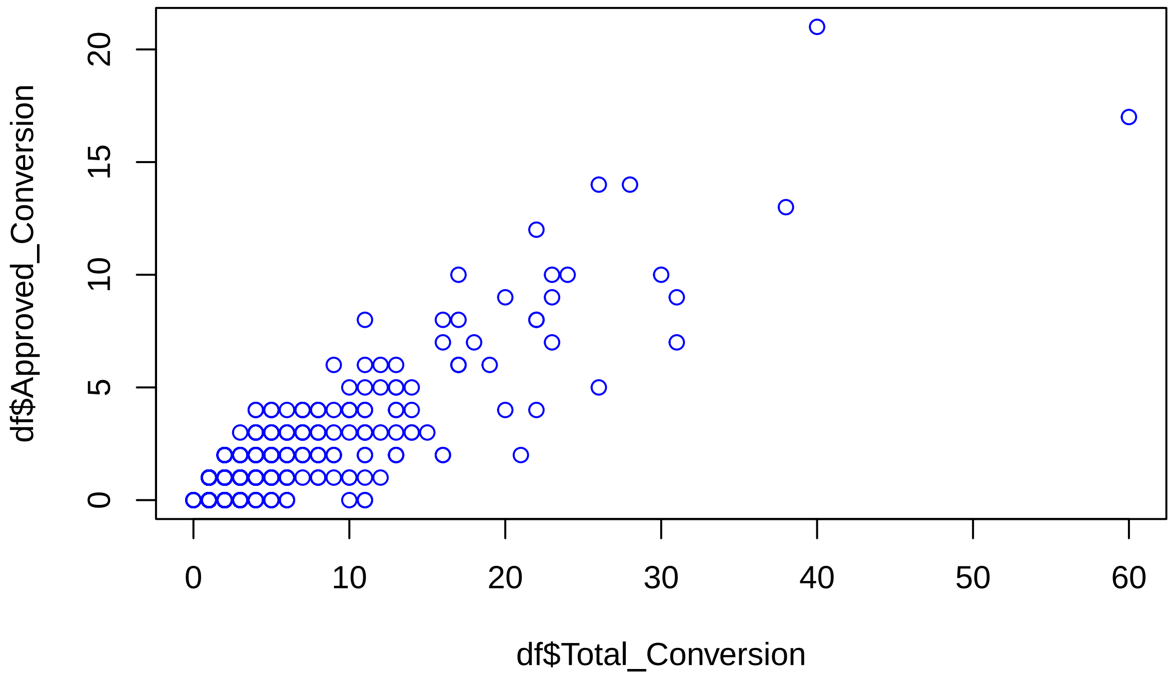
```r
head(df)
```

```
##    ad_id xyz_campaign_id fb_campaign_id   age gender interest Impressions
## 1 708746             916         103916 30-34      M     0-25        7350
## 2 708749             916         103917 30-34      M     0-25       17861
## 3 708771             916         103920 30-34      M     0-25         693
## 4 708815             916         103928 30-34      M    26-50        4259
## 5 708818             916         103928 30-34      M    26-50        4133
## 6 708820             916         103929 30-34      M    26-50        1915
##   Clicks Spent Total_Conversion Approved_Conversion
## 1      1  1.43                2                   1
## 2      2  1.82                2                   0
## 3      0  0.00                1                   0
## 4      1  1.25                1                   0
## 5      1  1.29                1                   1
## 6      0  0.00                1                   1
```

```r
# Correlation matrix!
cor(df[c(7,8,9,10,11)])
```

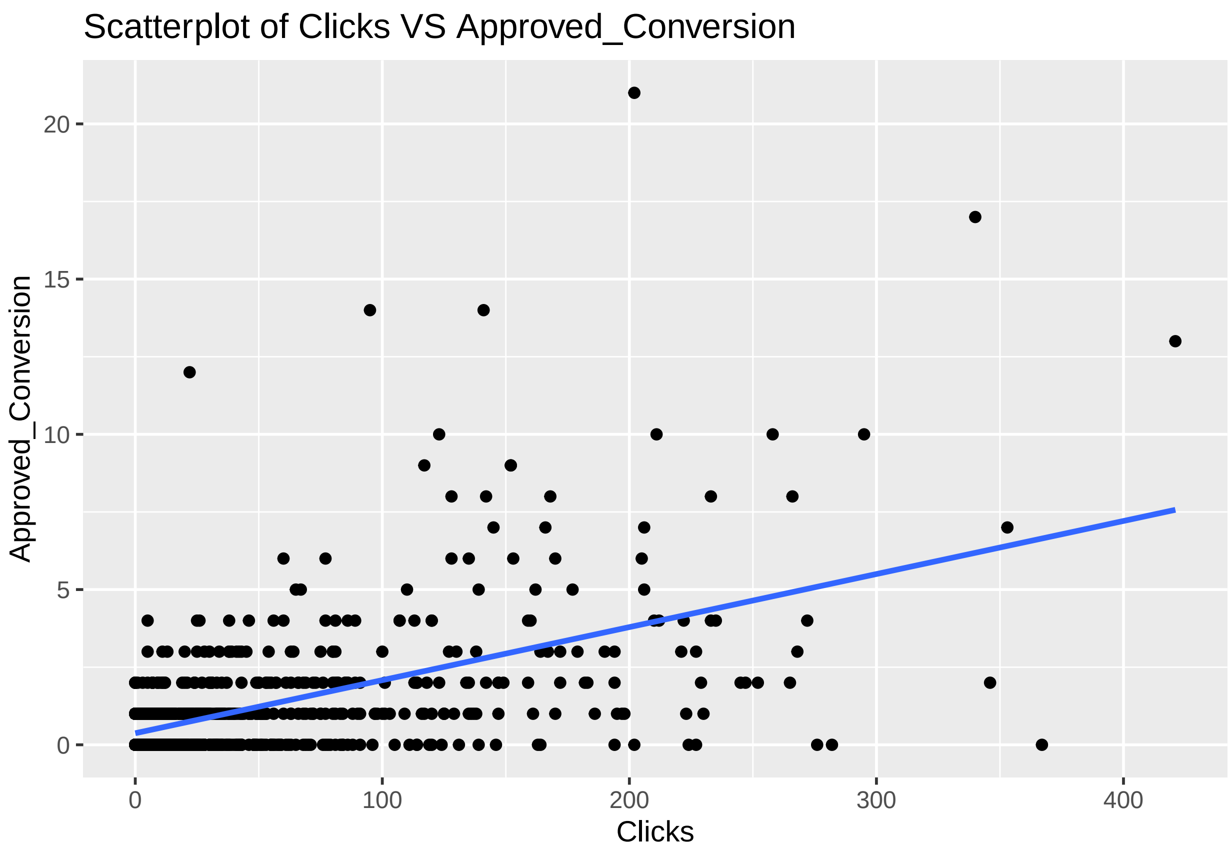
```
##                      Impressions    Clicks     Spent Total_Conversion
## Impressions            1.0000000 0.9485141 0.9703862        0.8128376
## Clicks                 0.9485141 1.0000000 0.9929063        0.6946324
## Spent                  0.9703862 0.9929063 1.0000000        0.7253794
## Total_Conversion       0.8128376 0.6946324 0.7253794        1.0000000
## Approved_Conversion   0.6842485 0.5595258 0.5931778        0.8640338
##                     Approved_Conversion
## Impressions                   0.6842485
## Clicks                        0.5595258
## Spent                         0.5931778
## Total_Conversion              0.8640338
## Approved_Conversion           1.0000000
```

```
plot(df$Approved_Conversion~df$Total_Conversion, data=df, col="blue")
```



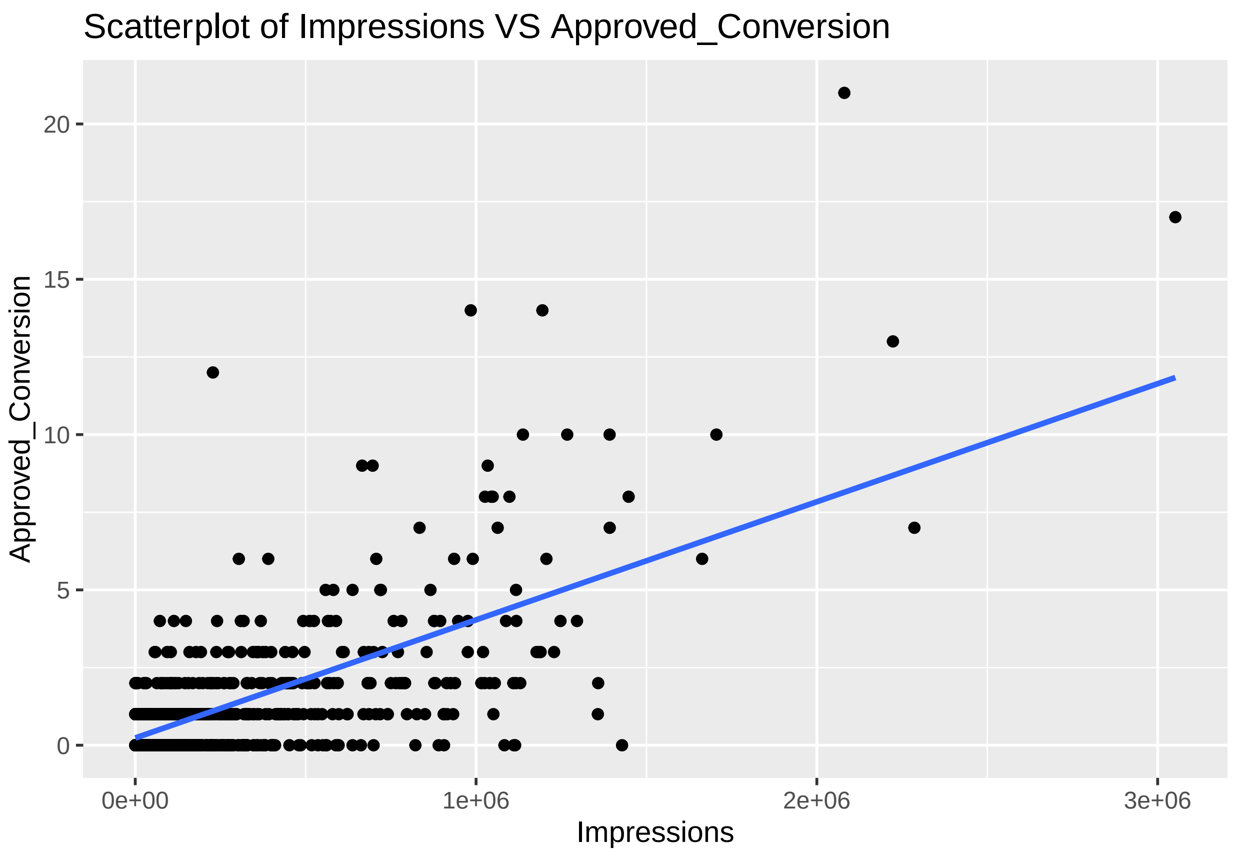
```
ggplot(df, aes(x=Clicks,y=Approved_Conversion))+geom_point()+ ggtitle("Scatterplot of Clicks VS Approve
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



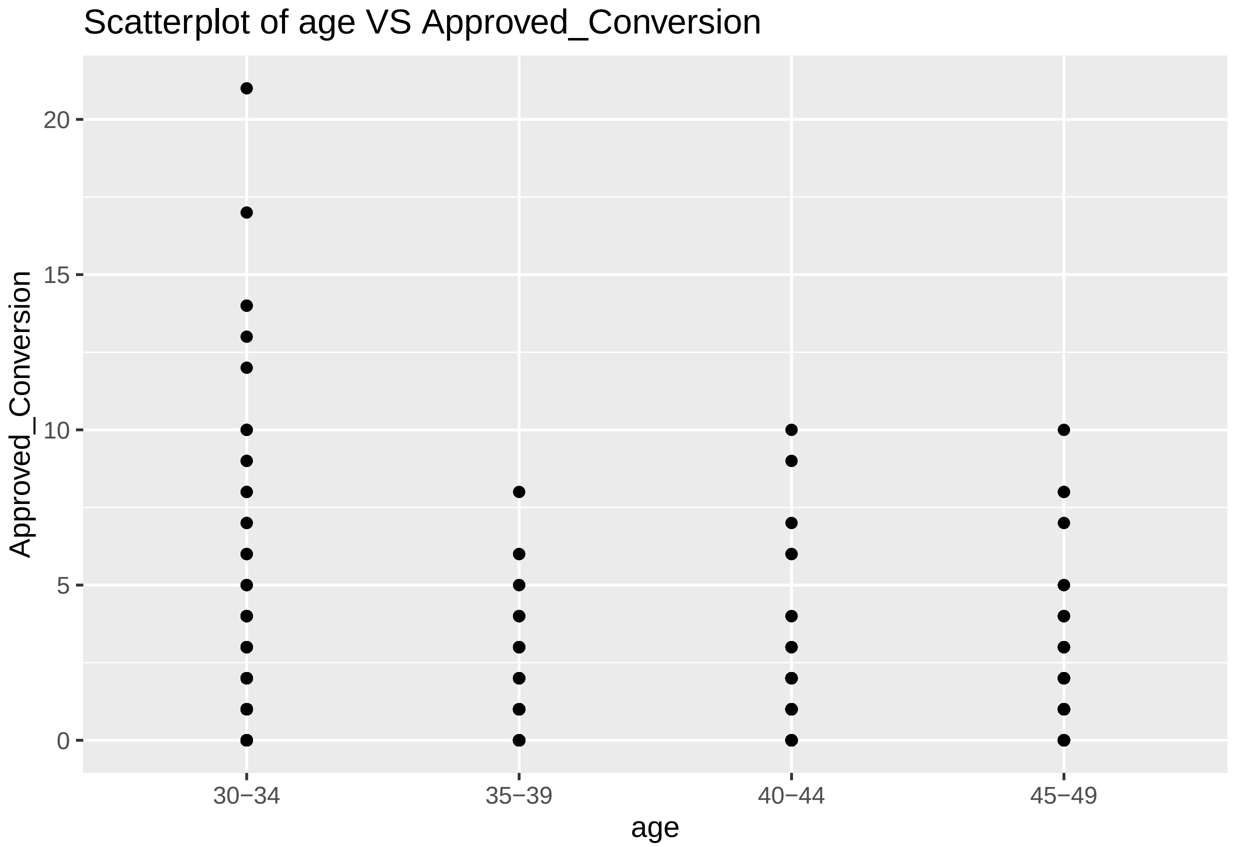
```
ggplot(df, aes(x=Impressions,y=Approved_Conversion))+geom_point()+ ggtitle("Scatterplot of Impressions V
```

```
## `geom_smooth()` using formula 'y ~ x'
```



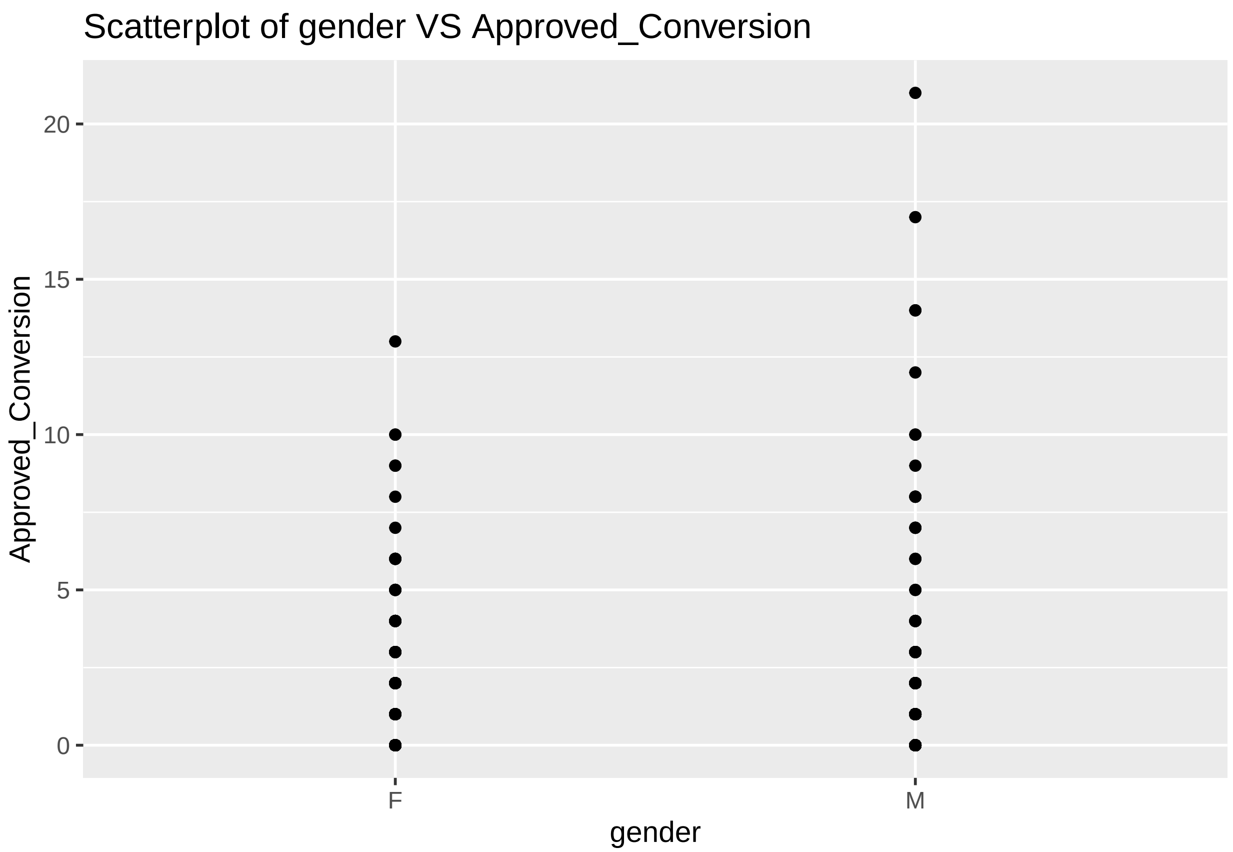
```
ggplot(df, aes(x=age,y=Approved_Conversion))+geom_point()+ ggtitle("Scatterplot of age VS Approved_Conve
```

```
## `geom_smooth()` using formula 'y ~ x'
```



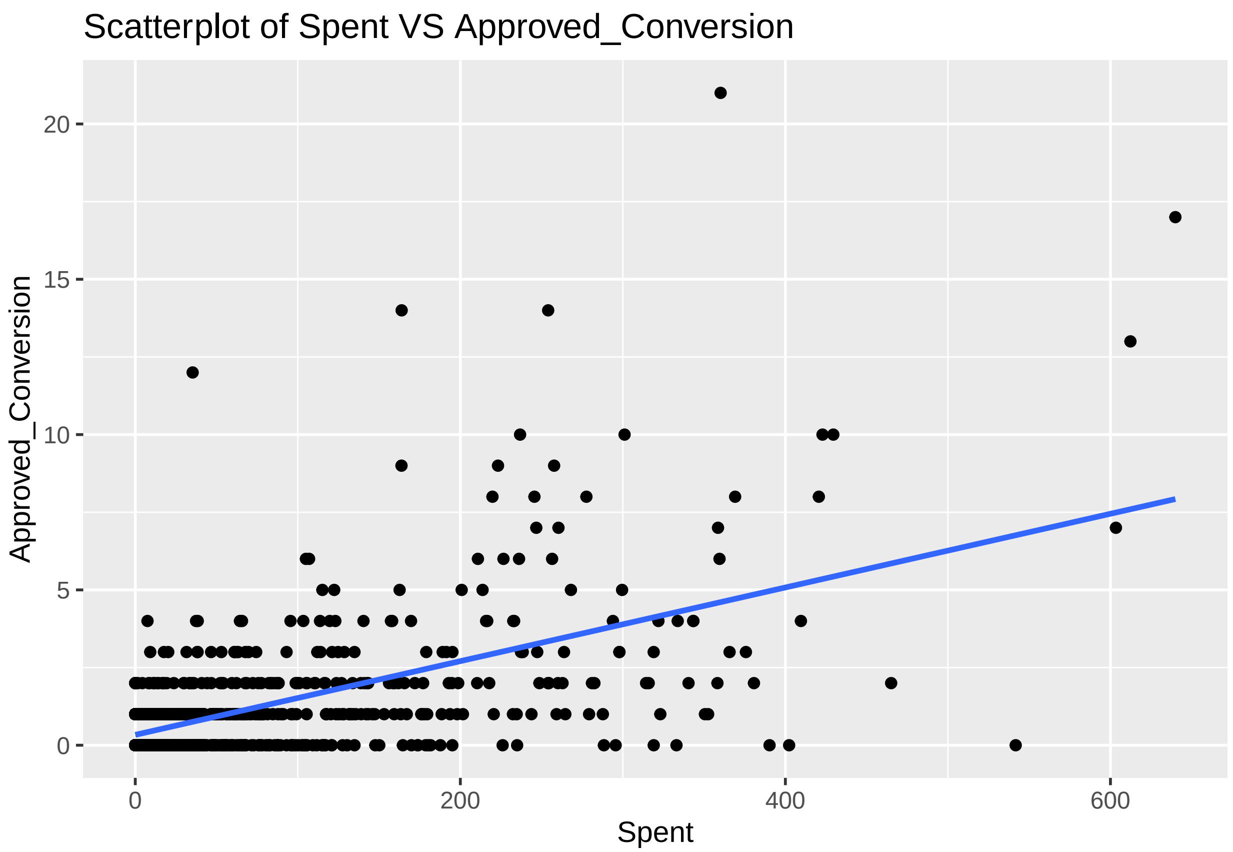
```
ggplot(df, aes(x=gender,y=Approved_Conversion))+geom_point()+ ggtitle("Scatterplot of gender VS Approved
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



```
ggplot(df, aes(x=Spent,y=Approved_Conversion))+geom_point()+ ggtitle("Scatterplot of Spent VS Approved_C
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Predicting variables to play with - xyz_campaign_id, age, gender,interest, Impressions, Clicks, Spent, Total Conversion

Response - Approved Conversion

```
ndf = df[,c(2,4,5,6,7,8,9,10,11)]
head(ndf)
```

```
##   xyz_campaign_id   age gender interest Impressions Clicks Spent
## 1             916 30-34      M     0-25        7350      1  1.43
## 2             916 30-34      M     0-25       17861      2  1.82
## 3             916 30-34      M     0-25         693      0  0.00
## 4             916 30-34      M    26-50        4259      1  1.25
## 5             916 30-34      M    26-50        4133      1  1.29
## 6             916 30-34      M    26-50        1915      0  0.00
##   Total_Conversion Approved_Conversion
## 1                2                   1
## 2                2                   0
## 3                1                   0
## 4                1                   0
## 5                1                   1
## 6                1                   1
```

```r
row.cnt = nrow(ndf)

# Split the data into training and testing sets
dftest = ndf[(row.cnt-230):row.cnt,]
dftrain = ndf[1:(row.cnt-231),]
row.cnt
```

```
## [1] 1143
```

```r
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 4.1.2
```

```r
#training and testing data 0.25 and 0.75
set.seed(100)
split = sample.split(ndf, SplitRatio = 0.75)
dftrain = subset(ndf, split == TRUE)
dftest = subset(ndf, split == FALSE)
```

```r
unique(dftrain$interest)
```

```
## [1] 0-25    26-50   51-75   76-100  101-125
## Levels: 0-25 26-50 51-75 76-100 101-125
```

Full model

```r
model = lm(Approved_Conversion ~., data = dftrain)
summary(model)
```

```
## 
## Call:
## lm(formula = Approved_Conversion ~ ., data = dftrain)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -4.5897 -0.4420 -0.2049  0.5501  5.0085 
## 
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)  
## (Intercept)           7.095e-03  1.506e-01   0.047  0.96244  
## xyz_campaign_id936  -1.208e-02  1.468e-01  -0.082  0.93446  
## xyz_campaign_id1178  2.381e-02  1.502e-01   0.159  0.87406  
## age35-39            -4.531e-02  8.194e-02  -0.553  0.58045  
## age40-44            -5.516e-02  8.895e-02  -0.620  0.53539  
## age45-49            -1.232e-01  9.023e-02  -1.365  0.17258  
## genderM              1.170e-01  6.686e-02   1.750  0.08061 .
## interest26-50        8.292e-02  7.037e-02   1.178  0.23902  
## interest51-75        1.208e-01  1.000e-01   1.208  0.22732  
## interest76-100      -9.354e-01  3.750e-01  -2.495  0.01282 *
## interest101-125     -3.551e-01  1.310e-01  -2.710  0.00688 **
## Impressions          1.093e-06  6.402e-07   1.707  0.08830 .
## Clicks               4.675e-03  5.536e-03   0.845  0.39865
## Spent               -7.372e-03  4.970e-03  -1.483  0.13839
## Total_Conversion     3.282e-01  1.247e-02  26.326  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8262 on 747 degrees of freedom
## Multiple R-squared:  0.7738, Adjusted R-squared:  0.7696
## F-statistic: 182.5 on 14 and 747 DF,  p-value: < 2.2e-16
```

```
vif(model)
```

```
##                        GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id    1.526112  2        1.111467
## age                1.301976  3        1.044962
## gender             1.246257  1        1.116359
## interest           1.359903  4        1.039174
## Impressions       47.662090  1        6.903774
## Clicks           117.168960  1       10.824461
## Spent            221.954684  1       14.898144
## Total_Conversion   3.719045  1        1.928483
```

```
max(10, 1/(1-summary(model)$r.squared))
```

```
## [1] 10
```

take out clicks first start with the one with not significant given others

No Clicks

```
modelb = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + interest + Impressions + Spent + Tota
summary(modelb)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     interest + Impressions + Spent + Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5947 -0.4357 -0.2120  0.5567  5.0727
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          2.302e-02  1.494e-01   0.154   0.8776
## xyz_campaign_id936  -1.650e-02  1.467e-01  -0.112   0.9105
## xyz_campaign_id1178  2.165e-02  1.501e-01   0.144   0.8854
## age35-39            -5.084e-02  8.167e-02  -0.623   0.5338
## age40-44            -6.043e-02  8.871e-02  -0.681   0.4960
## age45-49            -1.319e-01  8.962e-02  -1.472   0.1415
## genderM              1.005e-01  6.394e-02   1.572   0.1164
## interest26-50        8.307e-02  7.036e-02   1.181   0.2381
## interest51-75        1.219e-01  9.999e-02   1.219   0.2232
## interest76-100      -9.141e-01  3.740e-01  -2.444   0.0148 *
## interest101-125     -3.221e-01  1.250e-01  -2.576   0.0102 *
## Impressions          8.398e-07  5.658e-07   1.484   0.1382
## Spent               -3.462e-03  1.806e-03  -1.917   0.0556 .
## Total_Conversion     3.278e-01  1.246e-02  26.317   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8261 on 748 degrees of freedom
## Multiple R-squared:  0.7736, Adjusted R-squared:  0.7696
## F-statistic: 196.6 on 13 and 748 DF,  p-value: < 2.2e-16
```

```
vif(modelb)
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id   1.522922  2        1.110886
## age               1.282940  3        1.042400
## gender            1.140234  1        1.067817
## interest          1.231834  4        1.026406
## Impressions      37.239681  1        6.102432
## Spent            29.331675  1        5.415873
## Total_Conversion  3.714759  1        1.927371
```

No clicks and impressions

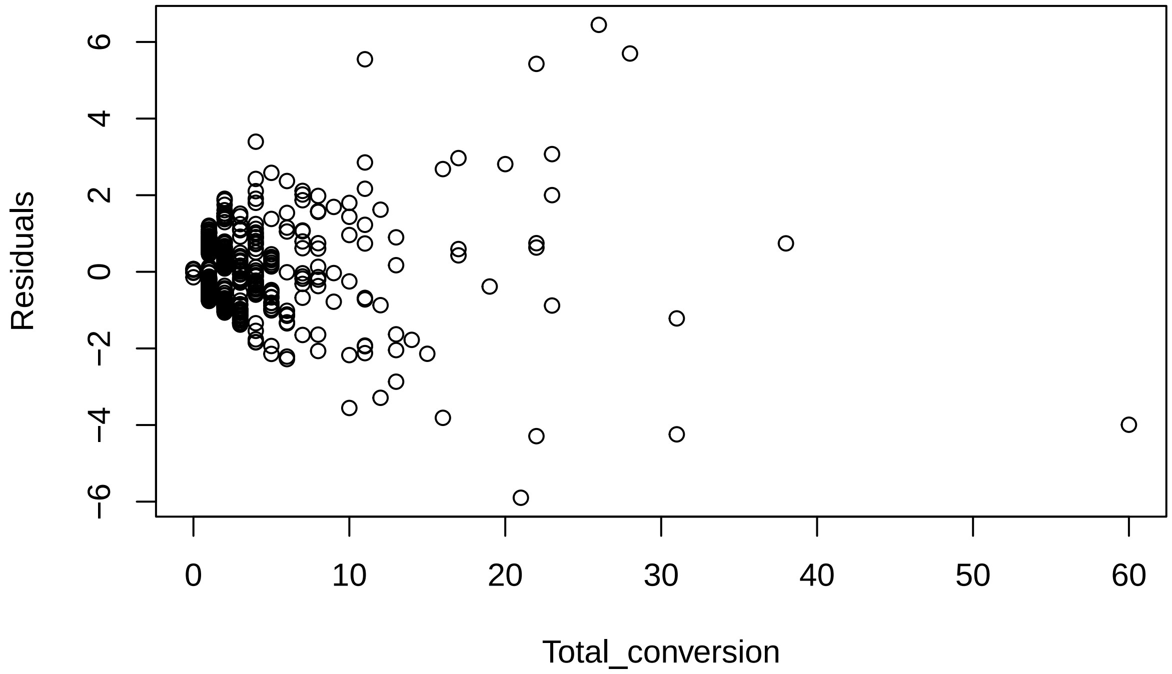
```
modelc = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + interest + Spent + Total_Conversion,
summary(modelc)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     interest + Spent + Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7581 -0.4481 -0.2086  0.5415  5.1795
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.0084351  0.1491998   0.057   0.9549
## xyz_campaign_id936  -0.0084156  0.1467304  -0.057   0.9543
## xyz_campaign_id1178  0.0420798  0.1496032   0.281   0.7786
## age35-39            -0.0593319  0.0815315  -0.728   0.4670
## age40-44            -0.0784839  0.0879462  -0.892   0.3725
## age45-49            -0.1648435  0.0868948  -1.897   0.0582 .
## genderM              0.1229708  0.0621760   1.978   0.0483 *
## interest26-50        0.0700540  0.0698633   1.003   0.3163
## interest51-75        0.1142805  0.0999351   1.144   0.2532
## interest76-100      -0.9239827  0.3742771  -2.469   0.0138 *
## interest101-125     -0.2964605  0.1239418  -2.392   0.0170 *
## Spent               -0.0009213  0.0005767  -1.598   0.1106
## Total_Conversion     0.3383856  0.0102186  33.115   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8268 on 749 degrees of freedom
## Multiple R-squared:  0.7729, Adjusted R-squared:  0.7693
## F-statistic: 212.4 on 12 and 749 DF,  p-value: < 2.2e-16
```

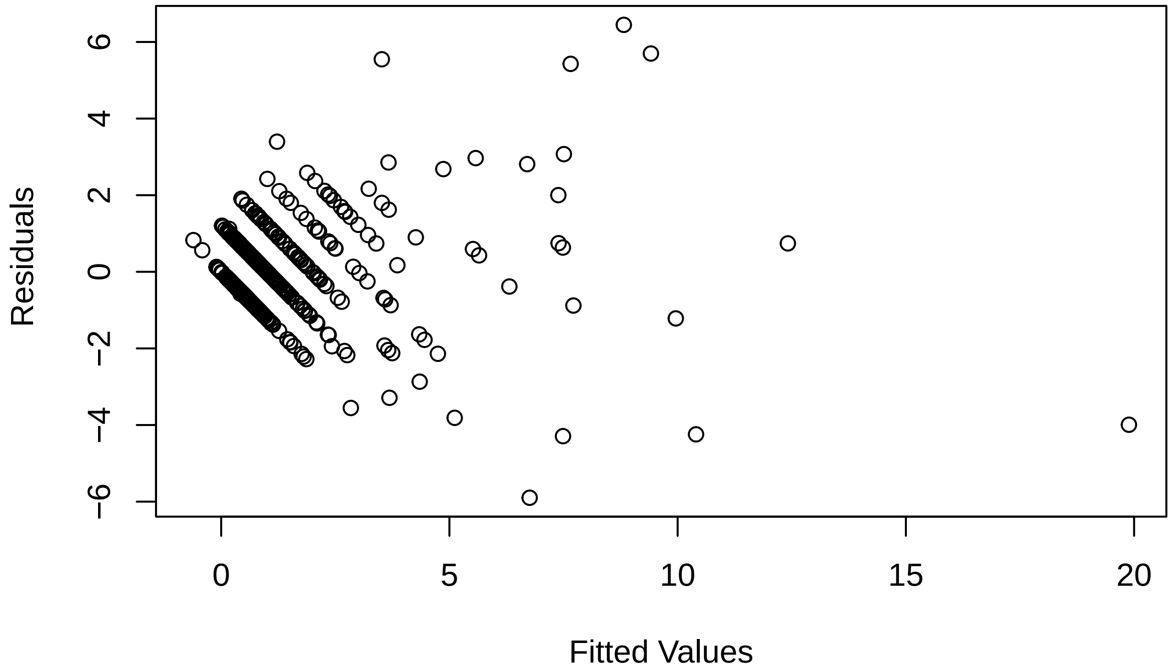
```
vif(modelc)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id  1.498480  2        1.106401
## age              1.200192  3        1.030881
## gender           1.076364  1        1.037480
## interest         1.177295  4        1.020612
## Spent            2.985244  1        1.727786
## Total_Conversion 2.496214  1        1.579941
```

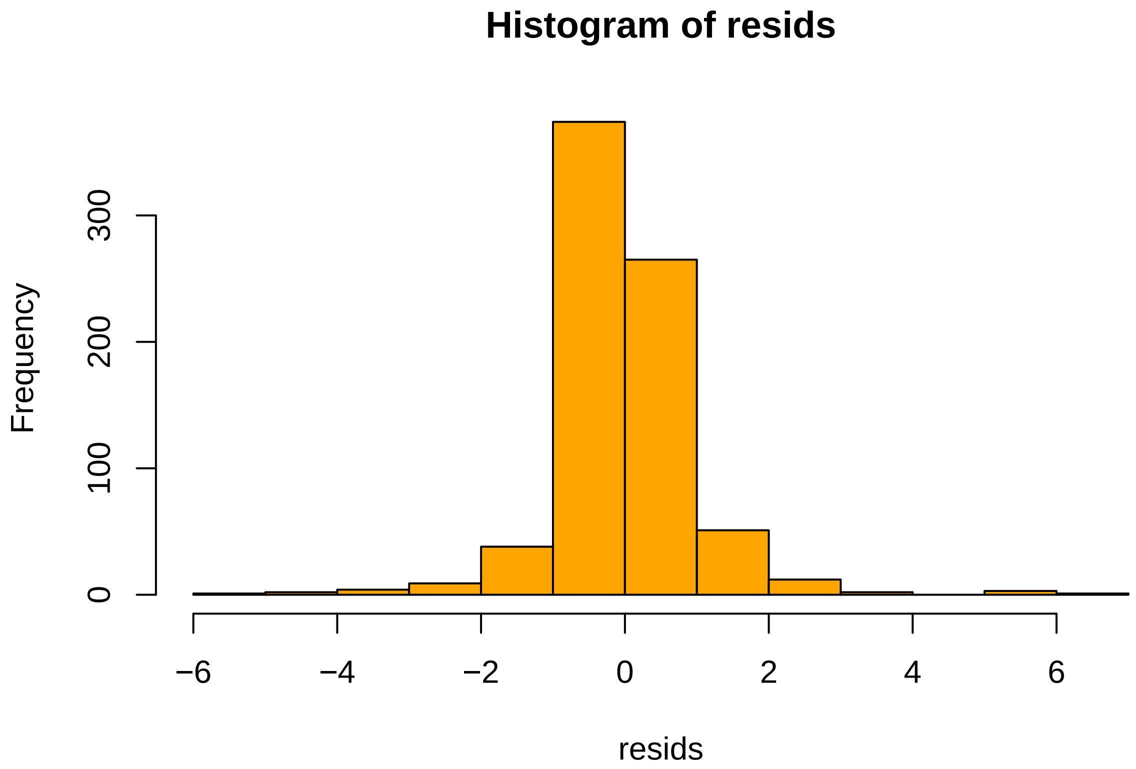
```
resids =rstandard(modelc)

plot(dftrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



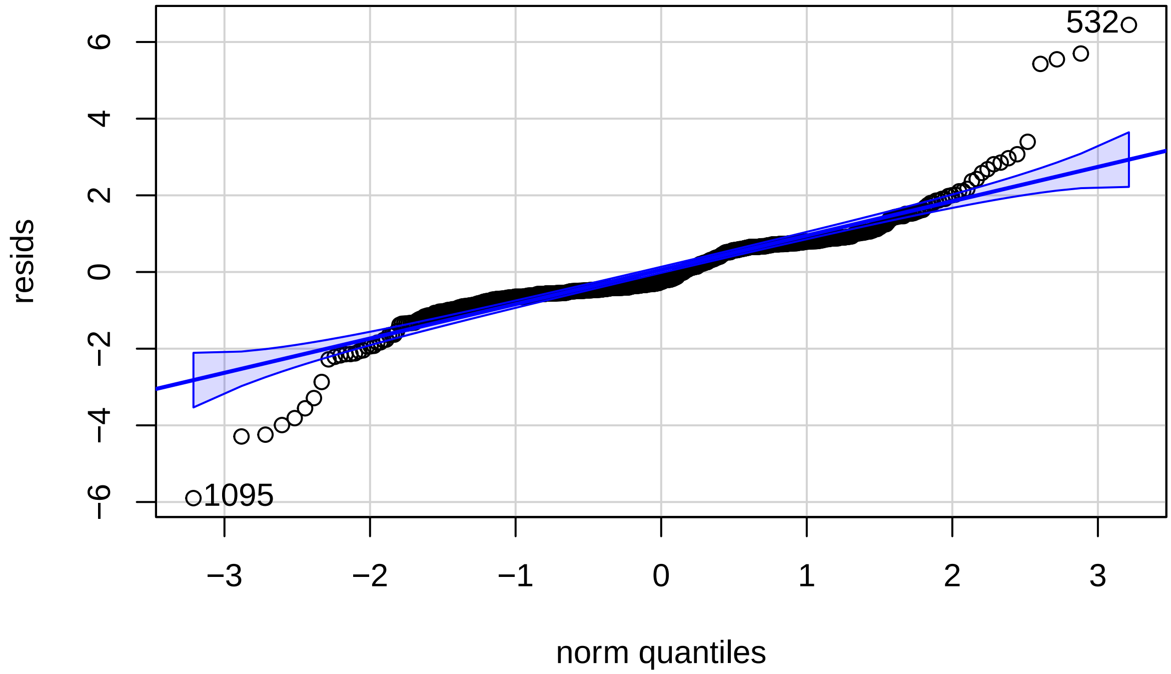
```
plot(modelc$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```



```
##  532 1095
##  355  731
```

No clicks and Spent

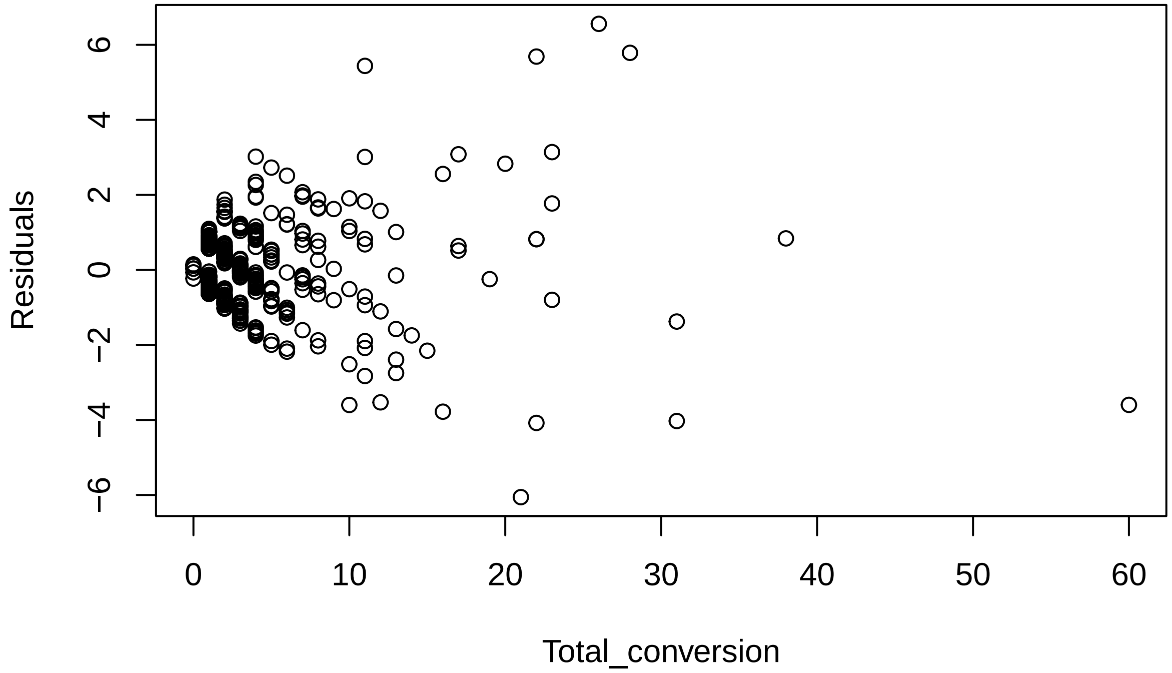
```
modeld = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + Impressions + Total_Conversion, data
summary(modeld)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     Impressions + Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9240 -0.4234 -0.2075  0.5533  5.3431
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            4.034e-02  1.490e-01   0.271  0.78671
## xyz_campaign_id936     1.149e-02  1.477e-01   0.078  0.93800
## xyz_campaign_id1178 -5.652e-03  1.495e-01  -0.038  0.96985
## age35-39             -8.037e-02  8.185e-02  -0.982  0.32650
## age40-44             -1.179e-01  8.759e-02  -1.345  0.17887
## age45-49             -2.194e-01  8.459e-02  -2.593  0.00969 **
## genderM               1.452e-01  6.151e-02   2.360  0.01851 *
## Impressions          -1.204e-07  1.798e-07  -0.670  0.50330
## Total_Conversion      3.306e-01  1.160e-02  28.501  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8334 on 753 degrees of freedom
## Multiple R-squared:  0.768,  Adjusted R-squared:  0.7656
## F-statistic: 311.6 on 8 and 753 DF,  p-value: < 2.2e-16
```

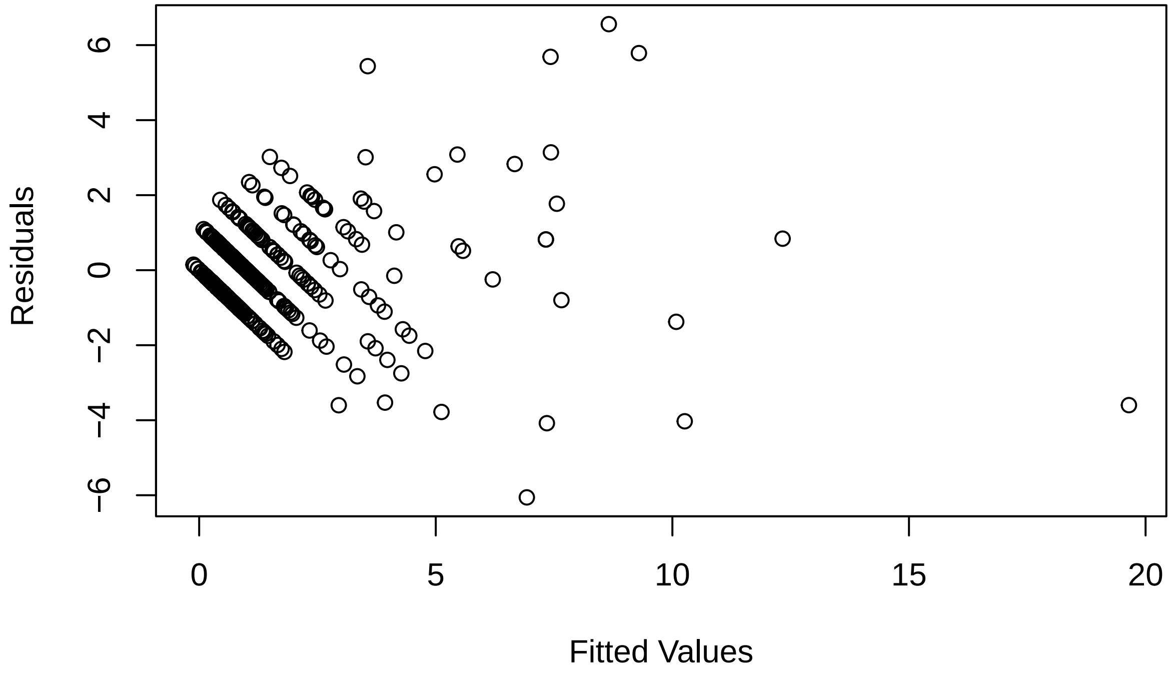
```
vif(modeld)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id  1.387202  2        1.085263
## age              1.111531  3        1.017779
## gender           1.036757  1        1.018213
## Impressions      3.693672  1        1.921893
## Total_Conversion 3.165620  1        1.779219
```

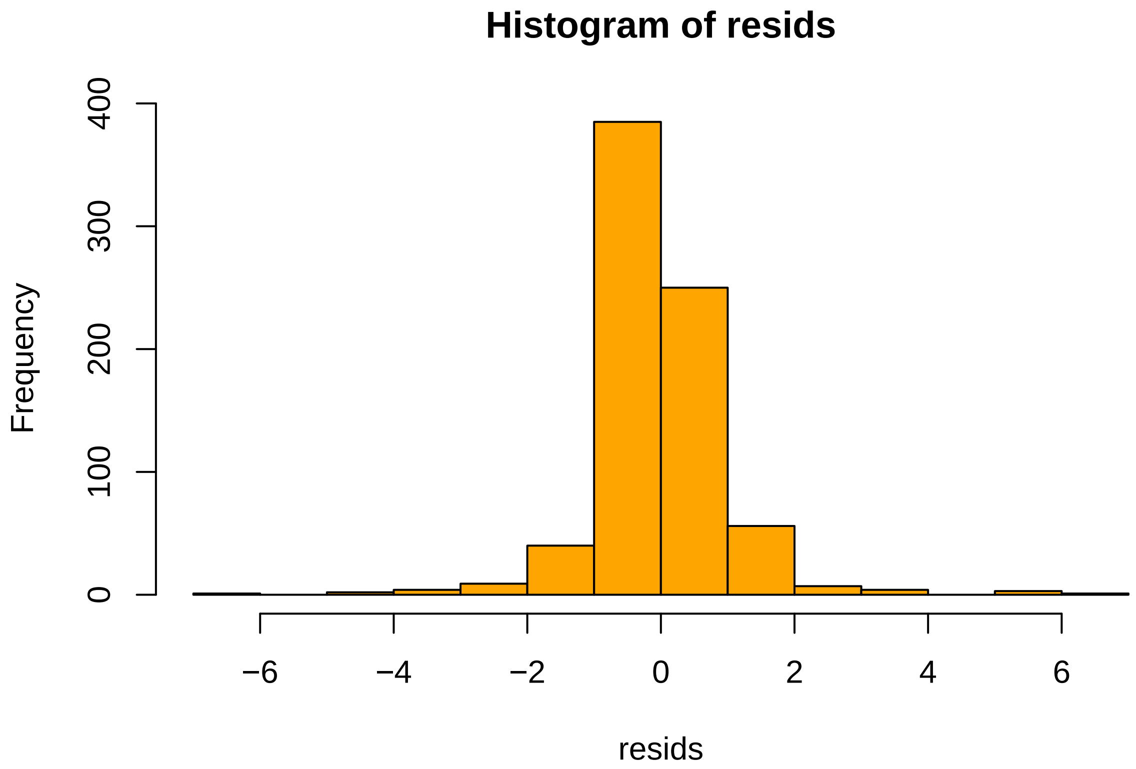
```
resids =rstandard(modeld)

plot(dftrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



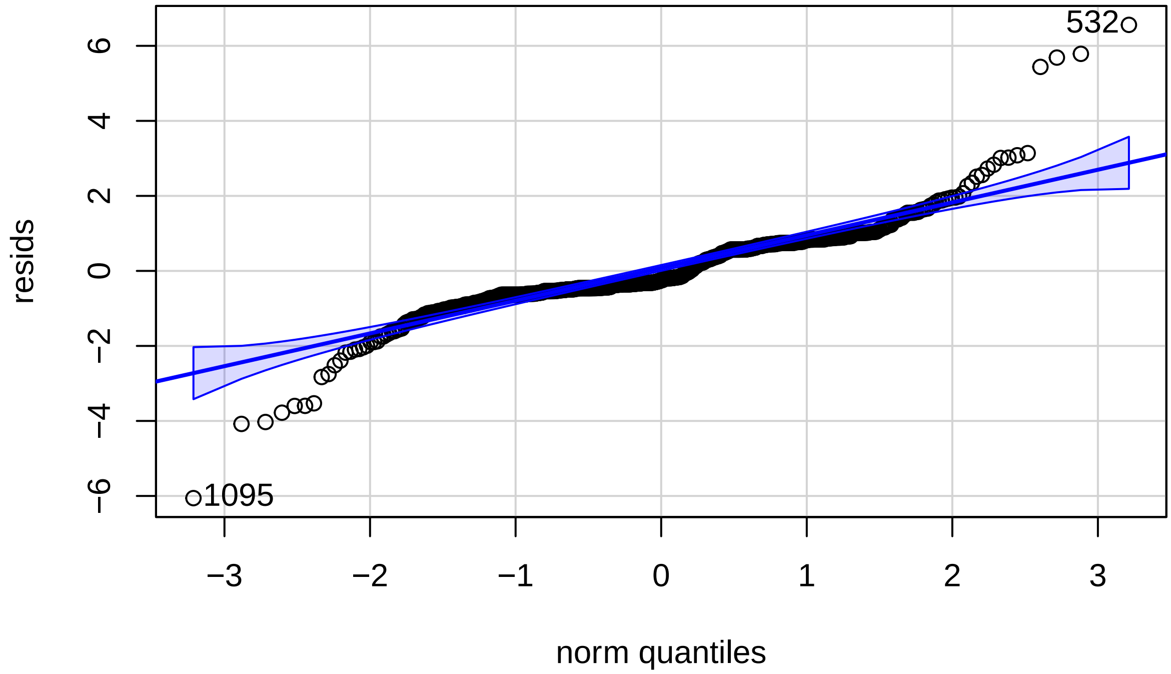
```
plot(modeld$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```



```
##  532 1095
##  355  731
```

No clicks, impressions and spent

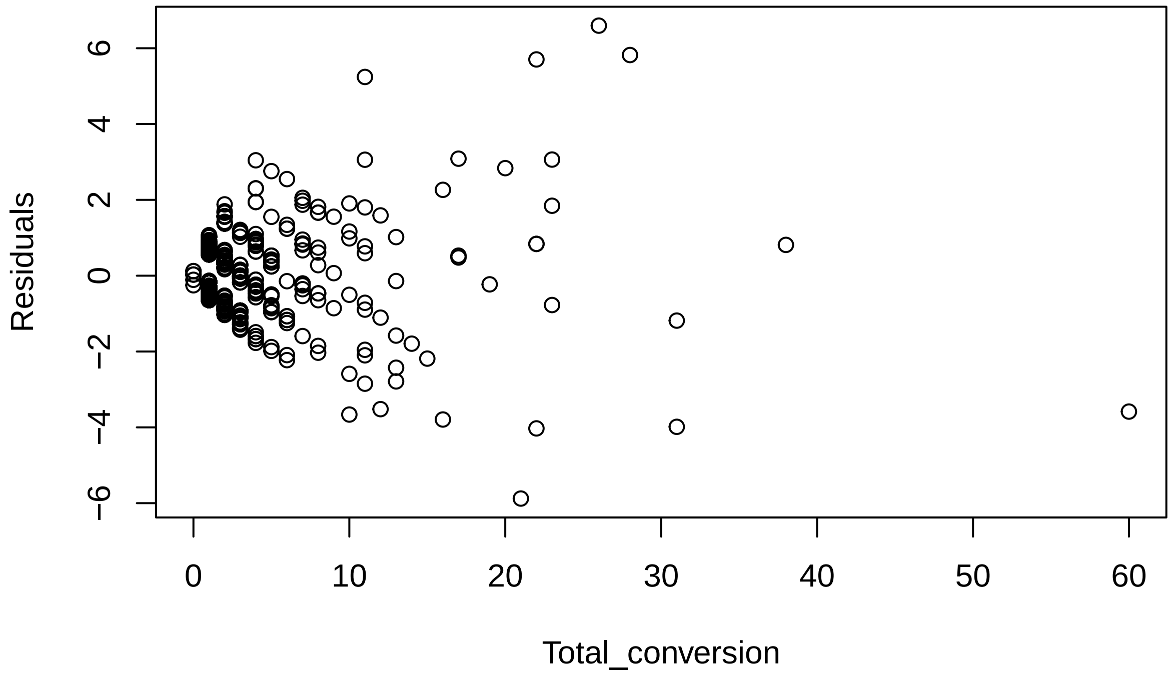
```
modele = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + Total_Conversion, data = dftrain)
summary(modele)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8369 -0.4141 -0.2242  0.5504  5.3915
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.046266   0.148704   0.311  0.75579
## xyz_campaign_id936     0.013625   0.147571   0.092  0.92646
## xyz_campaign_id1178 -0.021898   0.147458  -0.149  0.88199
## age35-39            -0.084230   0.081621  -1.032  0.30242
## age40-44            -0.124609   0.086977  -1.433  0.15237
## age45-49            -0.233563   0.081866  -2.853  0.00445 **
## genderM              0.149553   0.061140   2.446  0.01467 *
## Total_Conversion     0.324407   0.006999  46.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8331 on 754 degrees of freedom
## Multiple R-squared:  0.7679, Adjusted R-squared:  0.7657
## F-statistic: 356.4 on 7 and 754 DF,  p-value: < 2.2e-16
```

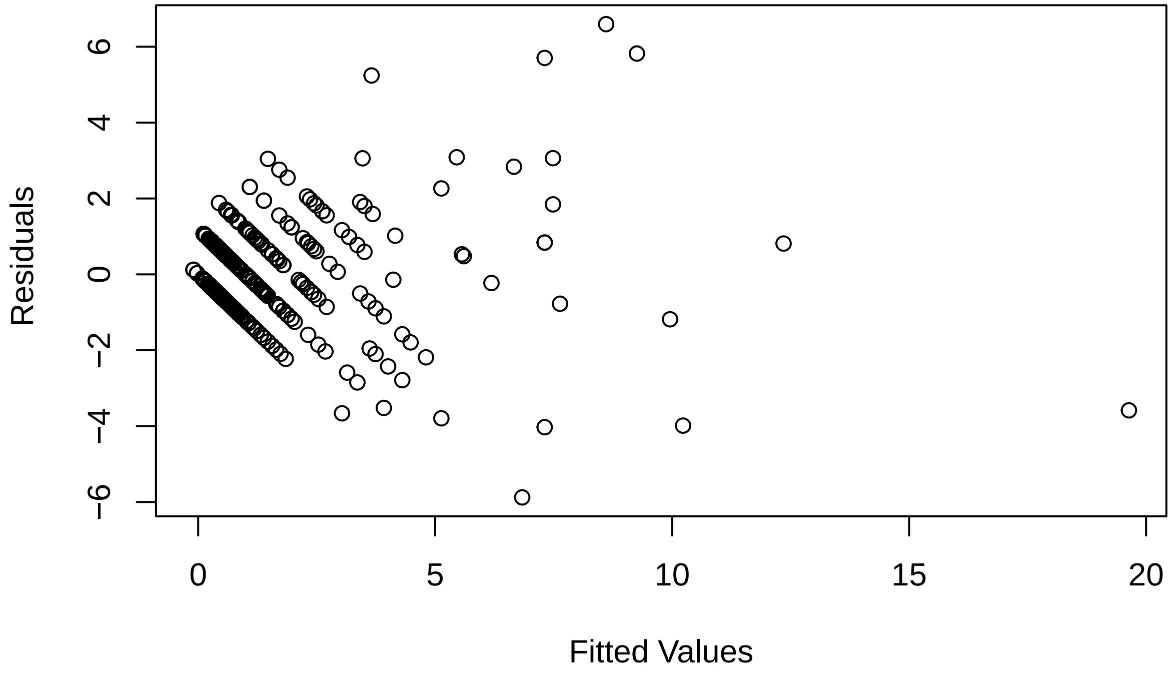
```
vif(modele)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id  1.184243  2        1.043182
## age              1.039801  3        1.006526
## gender           1.025077  1        1.012461
## Total_Conversion 1.153520  1        1.074021
```

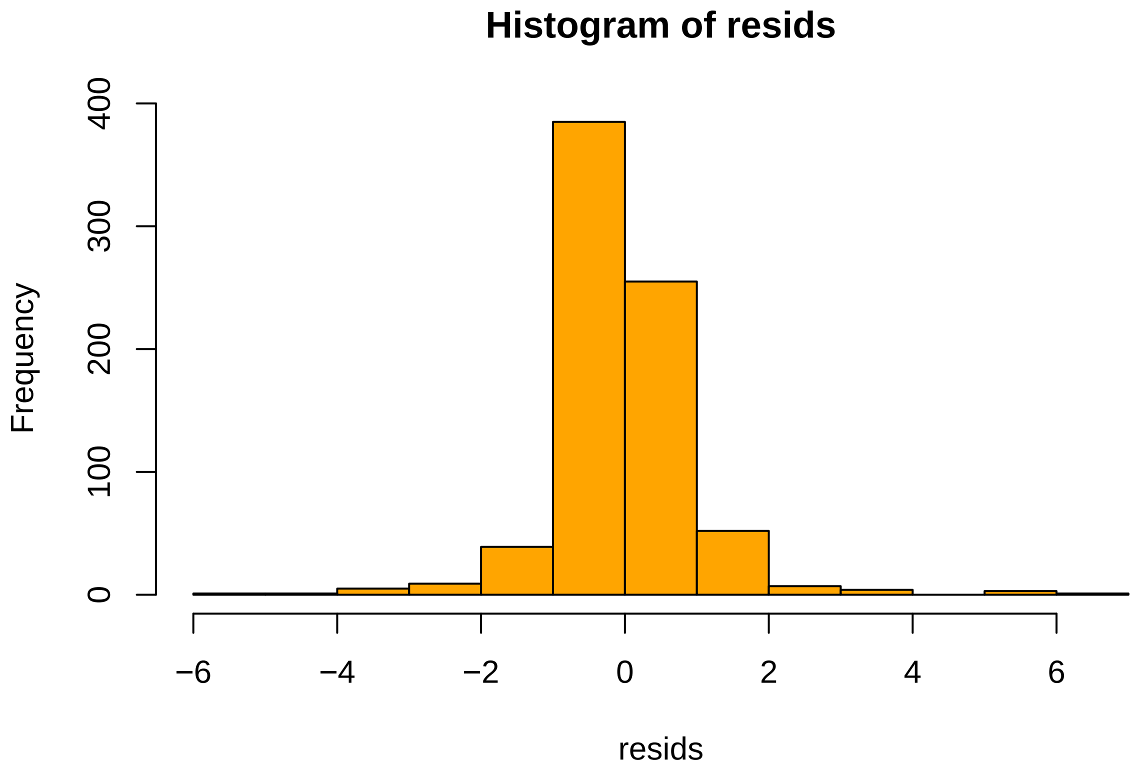
```
resids =rstandard(modele)

plot(dftrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



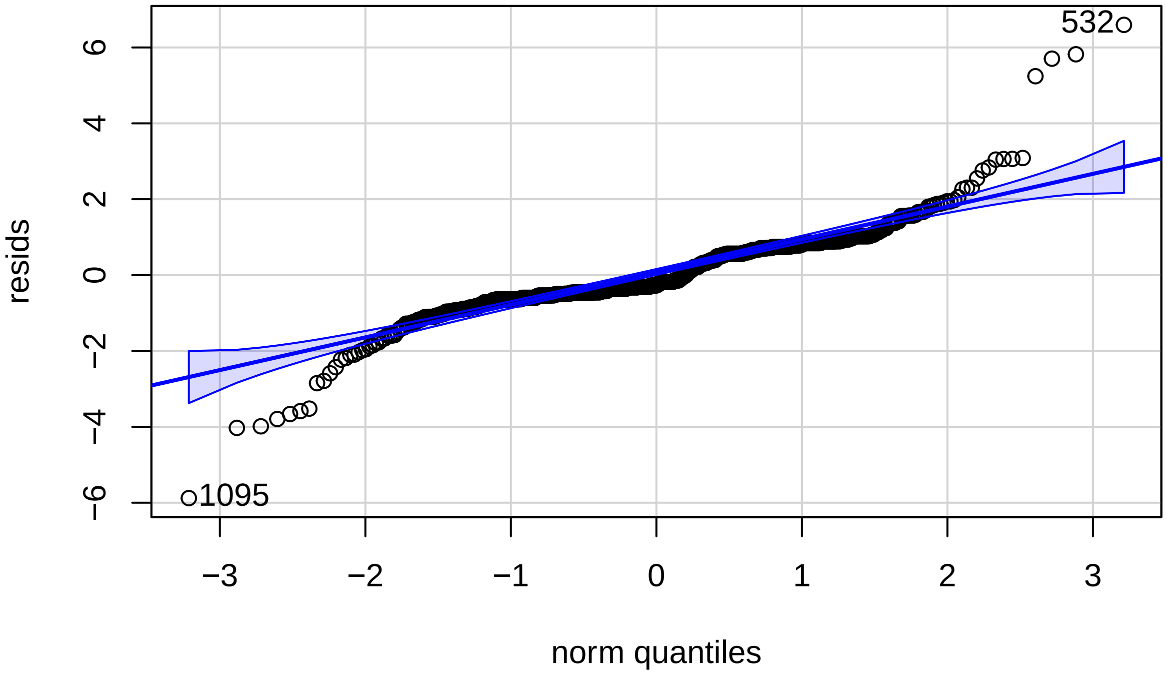
```
plot(modele$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```

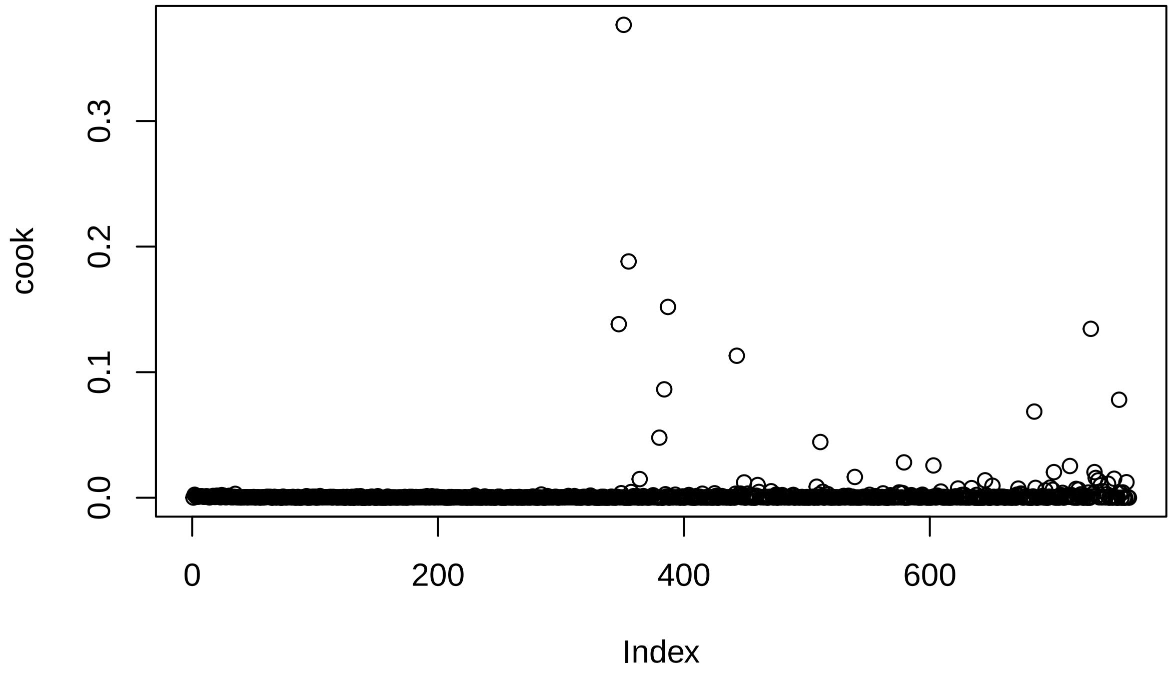


```
##  532 1095
##  355  731
```

There are problems with the linearity and constant variance assumption.

#cooks for modelc - No clicks and impressions

```
cook=cooks.distance(modelc)
plot(cook)
```



```
row_outlier = (which(cook >0.3,arr.ind=TRUE))
row_outlier
```

```
## 526
## 351
```

```
dftrain2 = dftrain[-c(row_outlier),]

modelca = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + Impressions + Total_Conversion, dat
summary(modelca)
```

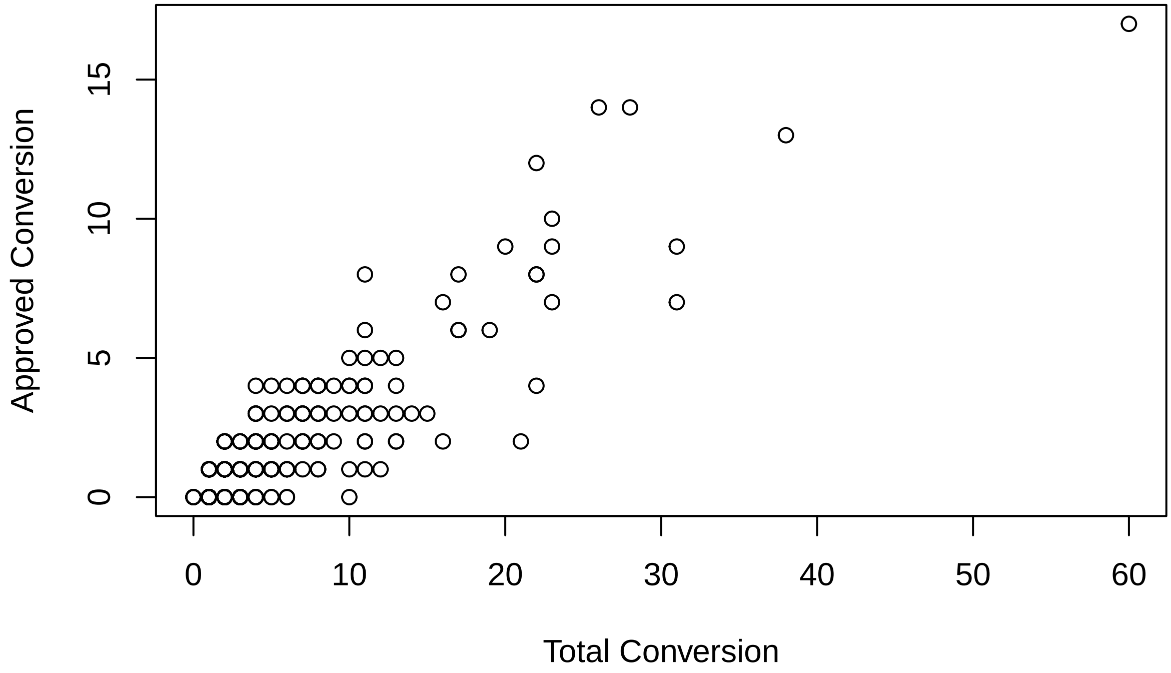
```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     Impressions + Total_Conversion, data = dftrain2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1529 -0.4167 -0.1879  0.5340  5.0408
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.569e-02  1.480e-01   0.106   0.9156
## xyz_campaign_id936    1.310e-02  1.465e-01   0.089   0.9288
## xyz_campaign_id1178 -3.731e-02  1.486e-01  -0.251   0.8018
## age35-39            -7.714e-02  8.121e-02  -0.950   0.3424
## age40-44            -1.118e-01  8.691e-02  -1.287   0.1985
## age45-49            -2.157e-01  8.393e-02  -2.570   0.0104 *
## genderM              1.563e-01  6.110e-02   2.559   0.0107 *
## Impressions         -1.336e-07  1.784e-07  -0.749   0.4539
## Total_Conversion     3.445e-01  1.213e-02  28.409   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8267 on 752 degrees of freedom
## Multiple R-squared:  0.7426, Adjusted R-squared:  0.7399
## F-statistic: 271.2 on 8 and 752 DF,  p-value: < 2.2e-16
```

```
vif(modelca)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## xyz_campaign_id  1.408971  2        1.089496
## age              1.109473  3        1.017465
## gender           1.038112  1        1.018878
## Impressions      3.313481  1        1.820297
## Total_Conversion 2.811977  1        1.676895
```

The point is not influential.

```
plot(dftrain$Total_Conversion,dftrain$Approved_Conversion, xlab= "Total Conversion",ylab="Approved Conve
```

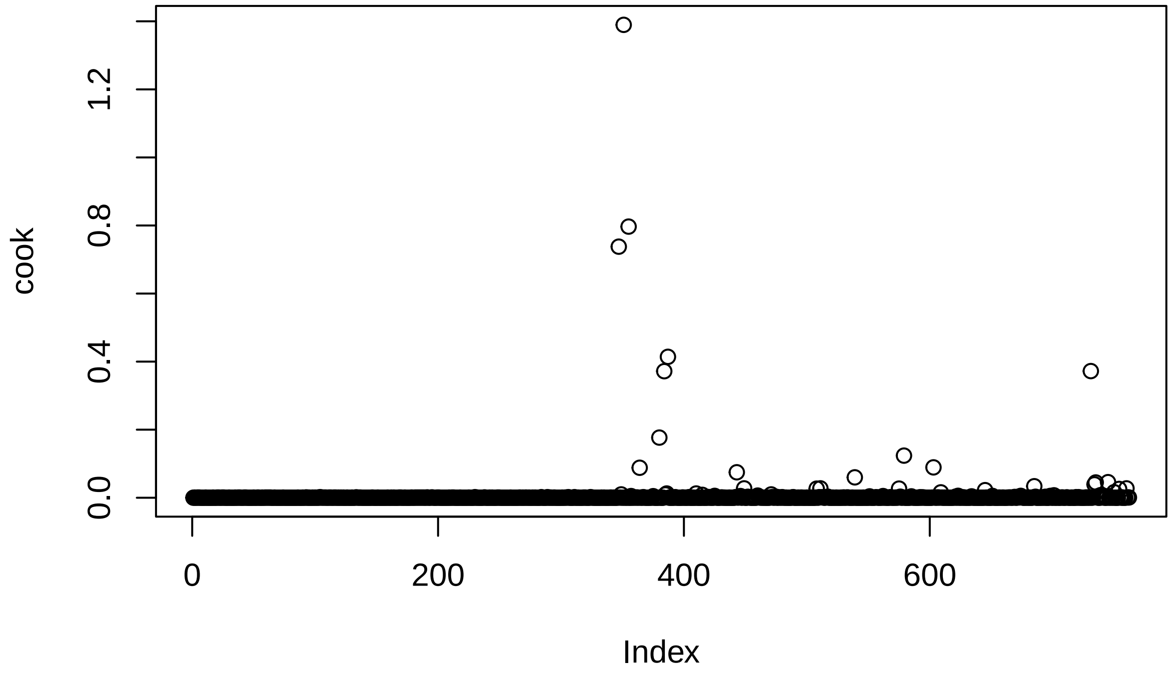


Total vs Approved

```
modelf = lm(Approved_Conversion ~ Total_Conversion, data = dftrain)
summary(modelf)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8366 -0.3473 -0.3473  0.6527  5.5411
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.022865   0.035748    0.64    0.523
## Total_Conversion 0.324464   0.006559   49.47   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8384 on 760 degrees of freedom
## Multiple R-squared:  0.763,  Adjusted R-squared:  0.7627
## F-statistic:  2447 on 1 and 760 DF,  p-value: < 2.2e-16
```

```
cook=cooks.distance(modelf)
plot(cook)
```



```
row_outlier = (which(cook >0.6,arr.ind=TRUE))
row_outlier
```

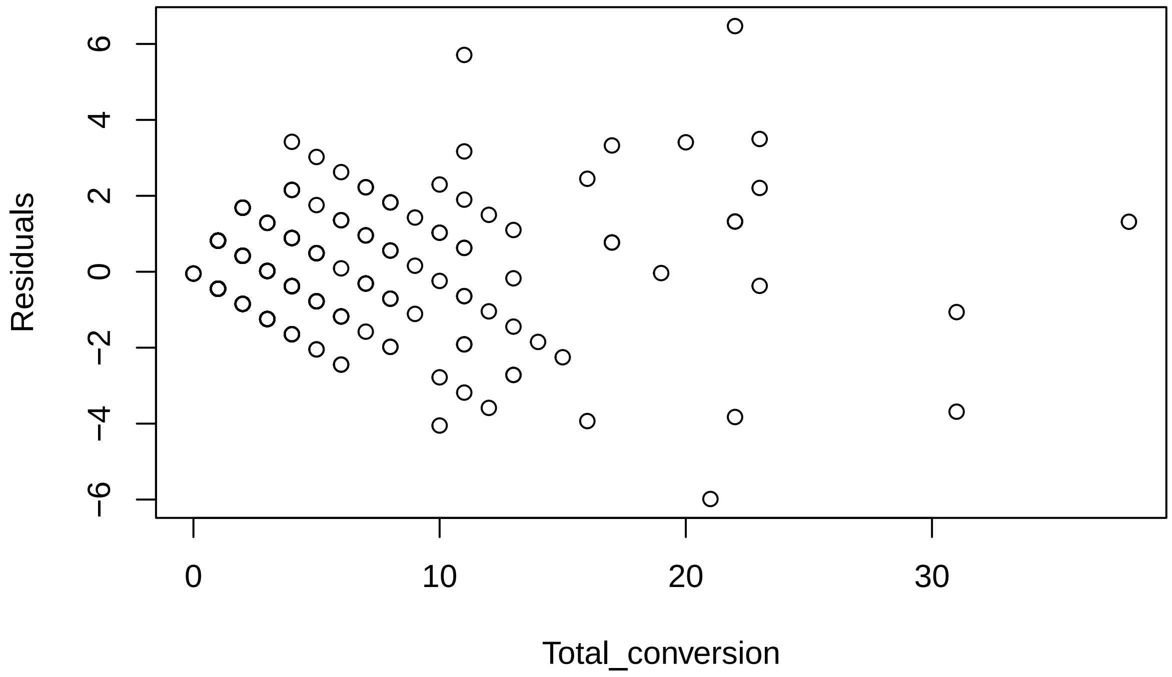
```
## 519 526 532
## 347 351 355
```

```
dftrain2 = dftrain[-c(row_outlier),]

modelf = lm(Approved_Conversion ~ Total_Conversion, data = dftrain2)
summary(modelf)
```

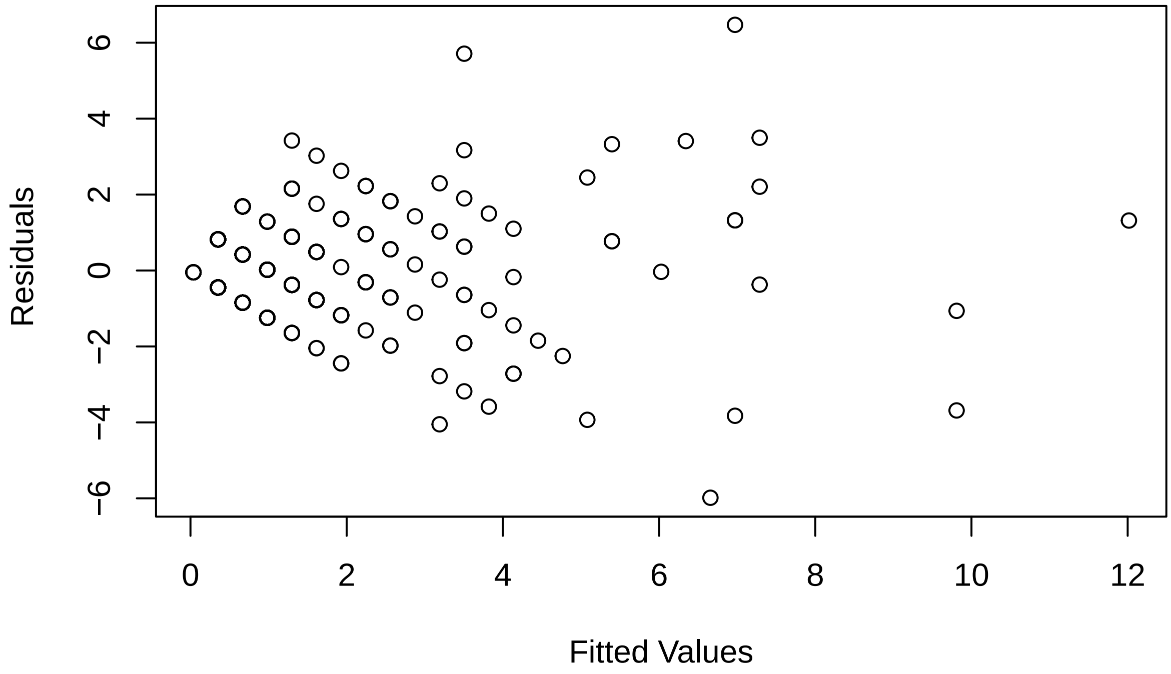
```
##
## Call:
## lm(formula = Approved_Conversion ~ Total_Conversion, data = dftrain2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6574 -0.3529 -0.3529  0.6471  5.0273
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.037665   0.034849   1.081     0.28
## Total_Conversion 0.315227   0.007242  43.526   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.7899 on 757 degrees of freedom
## Multiple R-squared:  0.7145, Adjusted R-squared:  0.7141
## F-statistic:  1894 on 1 and 757 DF,  p-value: < 2.2e-16
```

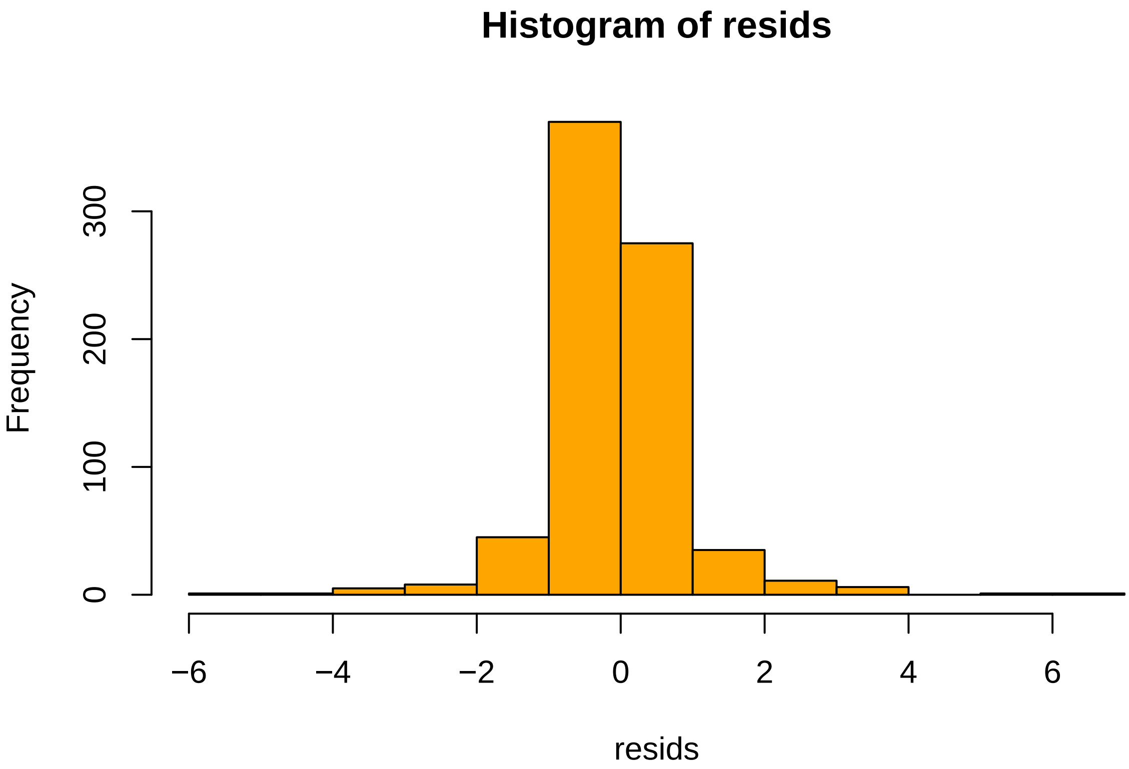
```
resids =rstandard(modelf)
plot(dftrain2$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



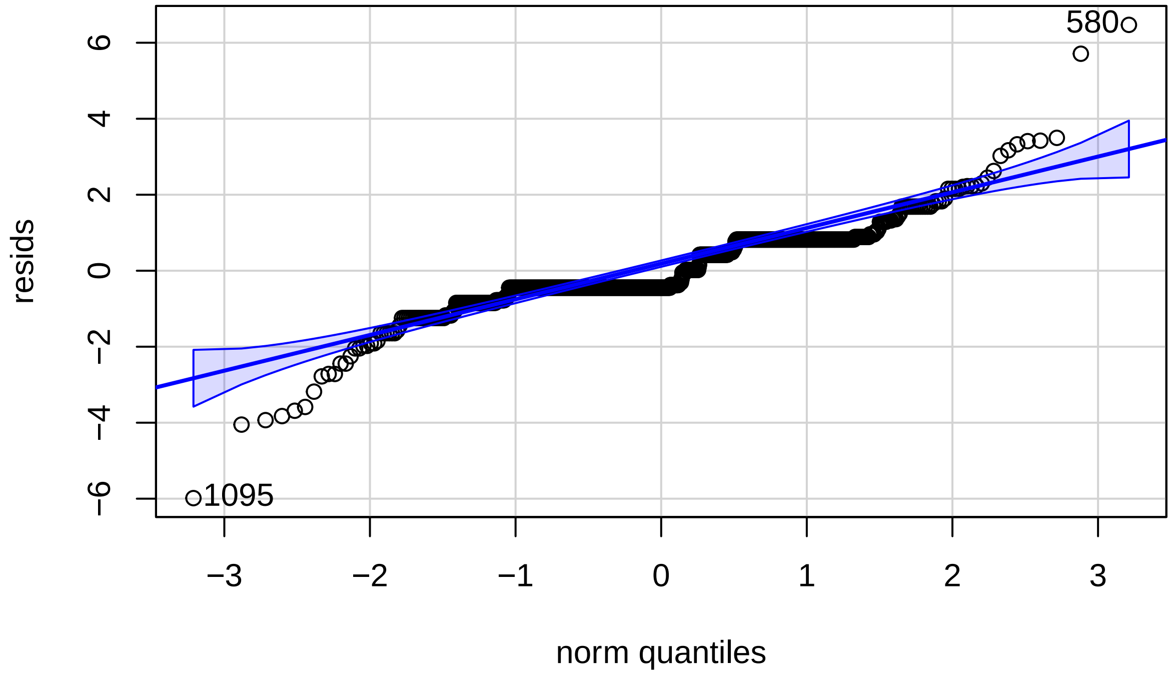
```
plot(modelf$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```



```
##  580 1095
##  384  728
```

#Forward Stepwise regression

```
intercept = lm(Approved_Conversion ~1, data = dftrain)
forward <- step(intercept, direction='forward', scope=formula(model), trace=0)
modelg = lm(formula = Approved_Conversion ~ Total_Conversion + interest + Clicks + gender, data = dftrai
summary(modelg)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ Total_Conversion + interest +
##     Clicks + gender, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7379 -0.4209 -0.2706  0.5791  5.1737
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.0453709  0.0562886  -0.806   0.4205
## Total_Conversion  0.3425170  0.0091672  37.363   <2e-16 ***
## interest26-50     0.0769175  0.0696774   1.104   0.2700
## interest51-75     0.1289727  0.0990395   1.302   0.1932
## interest76-100   -0.9230265  0.3720510  -2.481   0.0133 *
## interest101-125  -0.2836026  0.1178161  -2.407   0.0163 *
## Clicks           -0.0016576  0.0007265  -2.282   0.0228 *
## genderM           0.1237457  0.0613712   2.016   0.0441 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8264 on 754 degrees of freedom
## Multiple R-squared:  0.7716, Adjusted R-squared:  0.7695
## F-statistic: 363.9 on 7 and 754 DF,  p-value: < 2.2e-16
```

```
vif(modelg)
```

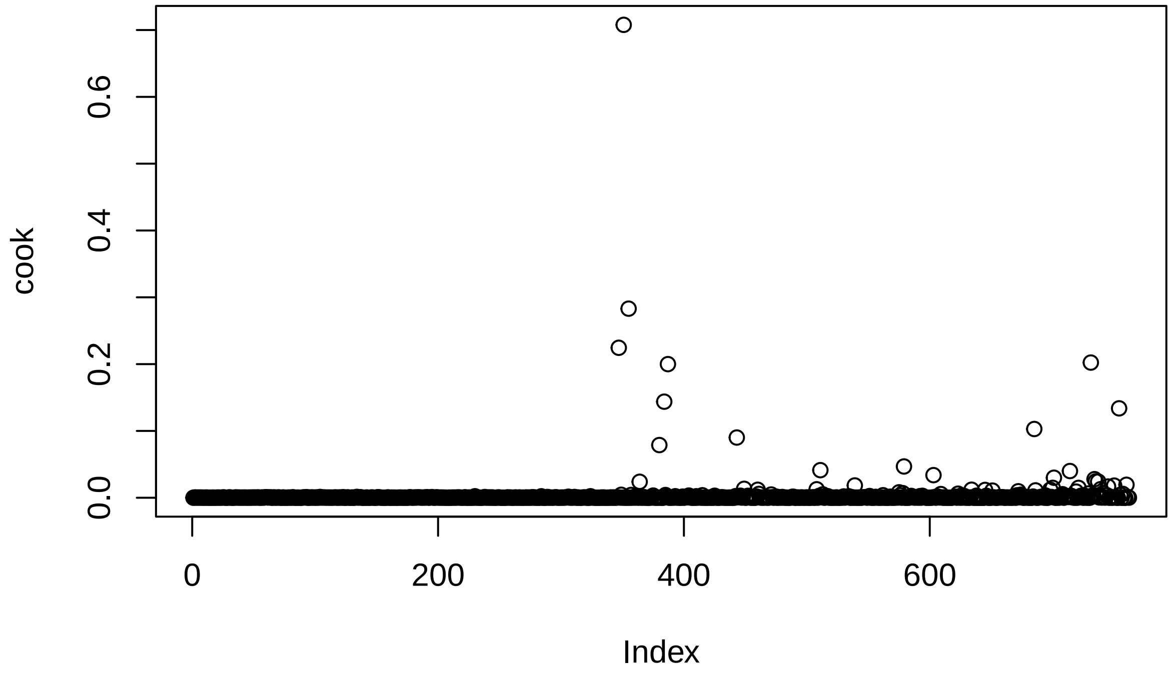
```
##                      GVIF Df GVIF^(1/(2*Df))
## Total_Conversion 2.010647  1        1.417973
## interest         1.038993  4        1.004793
## Clicks           2.016996  1        1.420210
## gender           1.049566  1        1.024483
```

```
forward
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ Total_Conversion + interest +
##     Clicks + gender, data = dftrain)
##
## Coefficients:
##      (Intercept)  Total_Conversion     interest26-50     interest51-75
##        -0.045371          0.342517          0.076918          0.128973
##   interest76-100   interest101-125            Clicks           genderM
##        -0.923027         -0.283603         -0.001658          0.123746
```

```
#forward
```

```
cook=cooks.distance(modelg)
plot(cook)
```



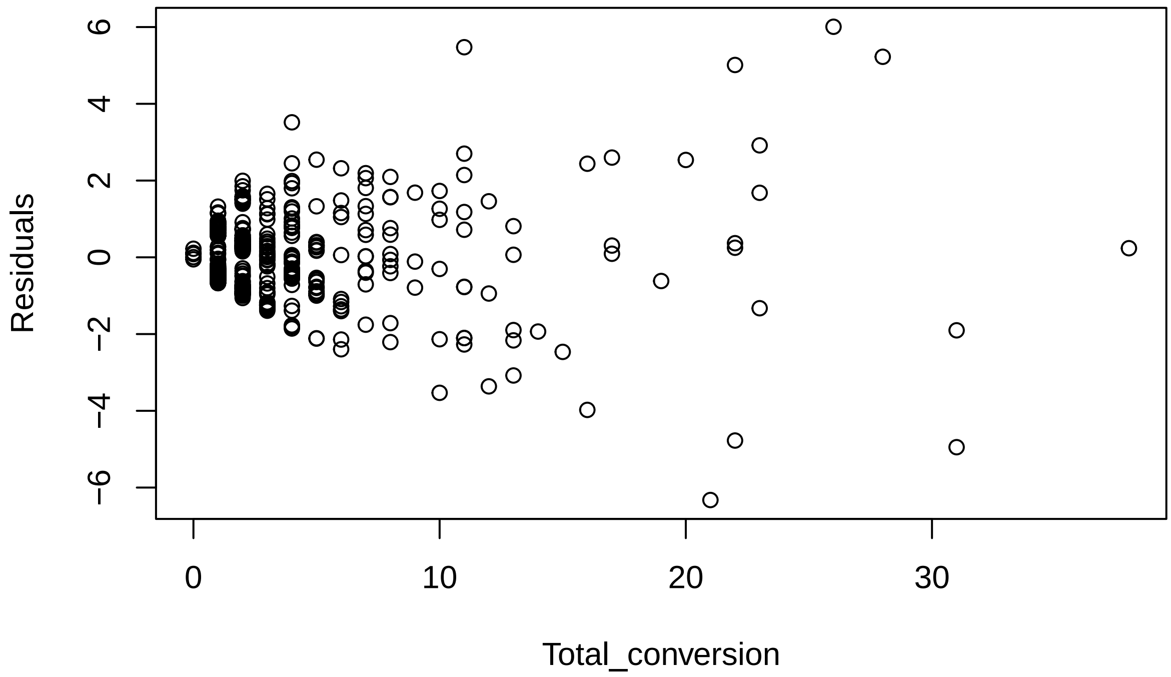
```
row_outlier = (which(cook >0.5,arr.ind=TRUE))
row_outlier
```

```
## 526
## 351
```

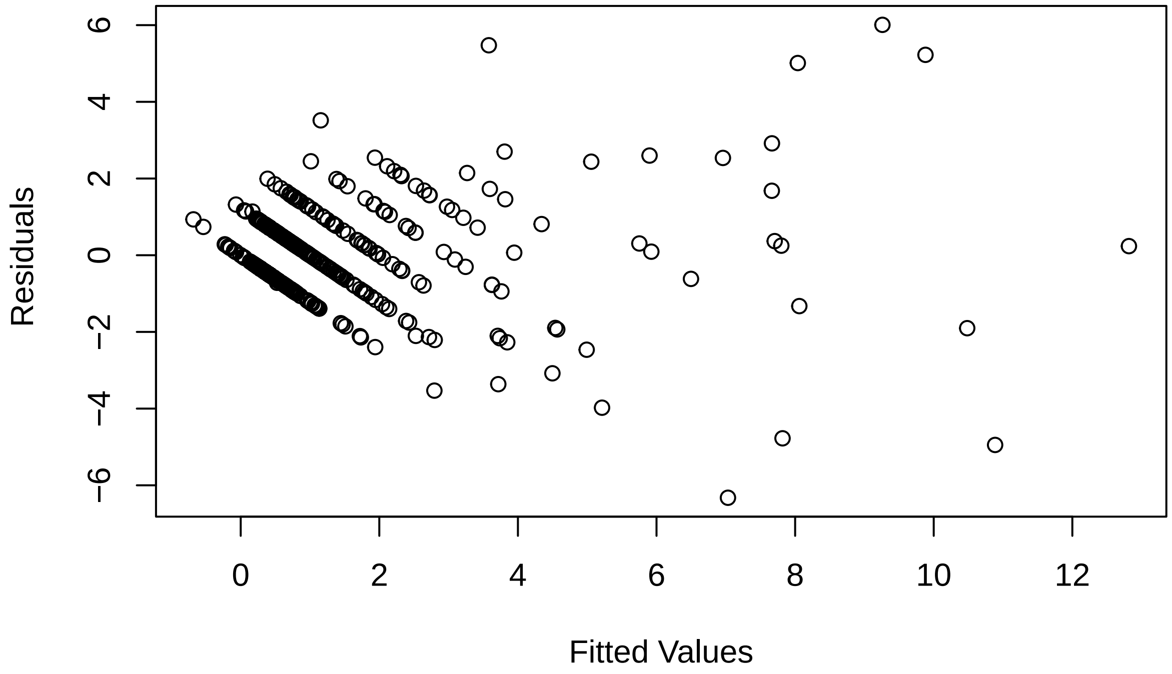
```
dftrain2 = dftrain[-c(row_outlier),]

modelg = lm(formula = Approved_Conversion ~ Total_Conversion + interest + Clicks + gender, data = dftrai
summary(modelg)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ Total_Conversion + interest +
##     Clicks + gender, data = dftrain2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.0309 -0.4164 -0.2580  0.5836  4.7405
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.068354   0.055903  -1.223  0.22181
## Total_Conversion  0.362250   0.010158  35.661  < 2e-16 ***
## interest26-50     0.064117   0.068947   0.930  0.35270
## interest51-75      0.129650   0.097910   1.324  0.18585
## interest76-100    -0.951379   0.367868  -2.586  0.00989 **
## interest101-125   -0.337417   0.117143  -2.880  0.00408 **
## Clicks            -0.002244   0.000731  -3.069  0.00222 **
## genderM            0.122493   0.060672   2.019  0.04385 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.817 on 753 degrees of freedom
## Multiple R-squared:  0.7483, Adjusted R-squared:  0.7459
## F-statistic: 319.8 on 7 and 753 DF,  p-value: < 2.2e-16
```

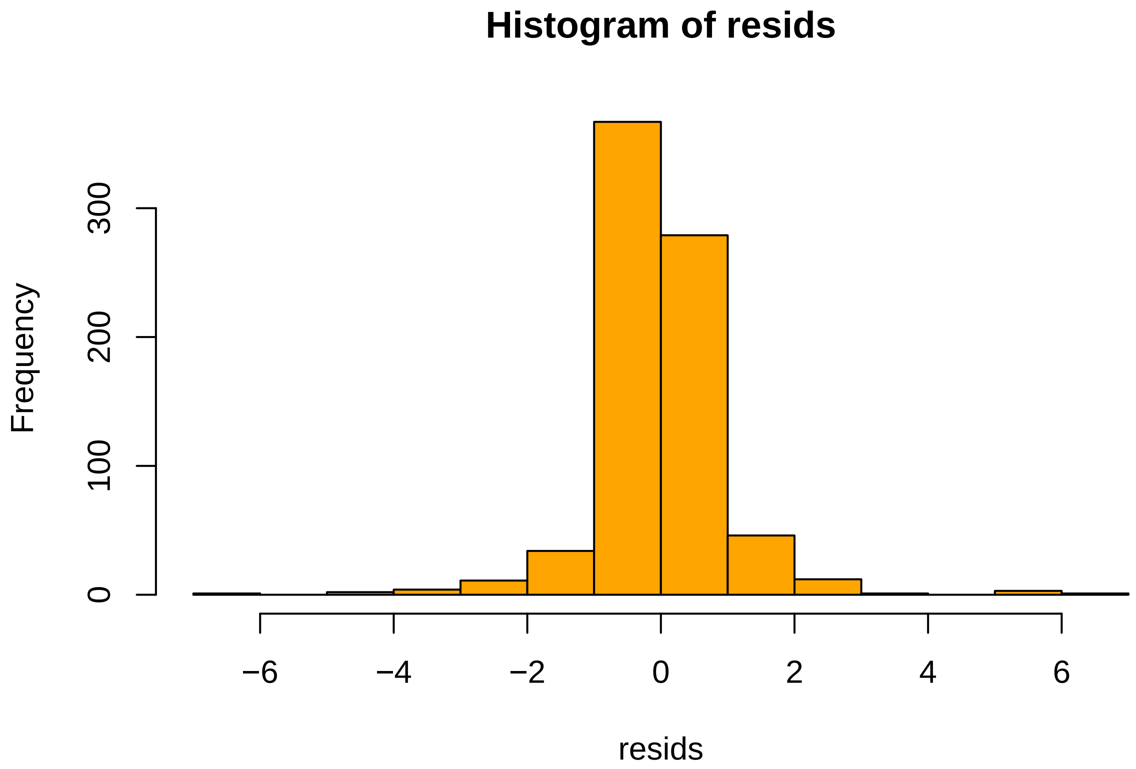
```
resids =rstandard(modelg)
plot(dftrain2$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



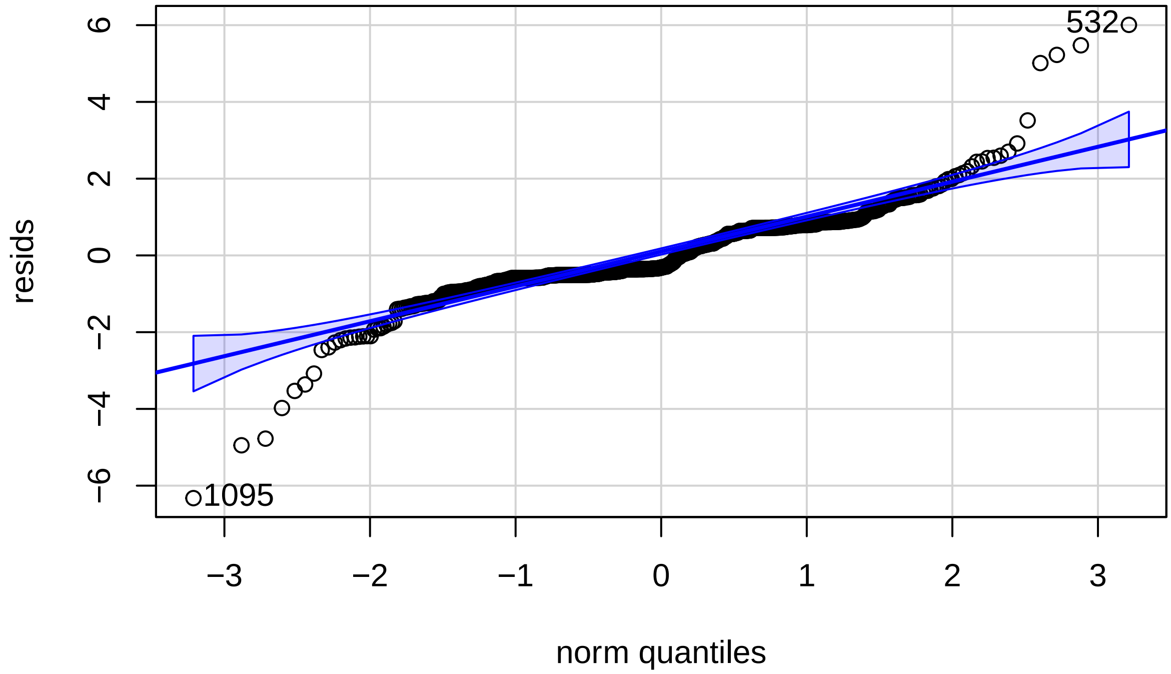
```
plot(modelg$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```

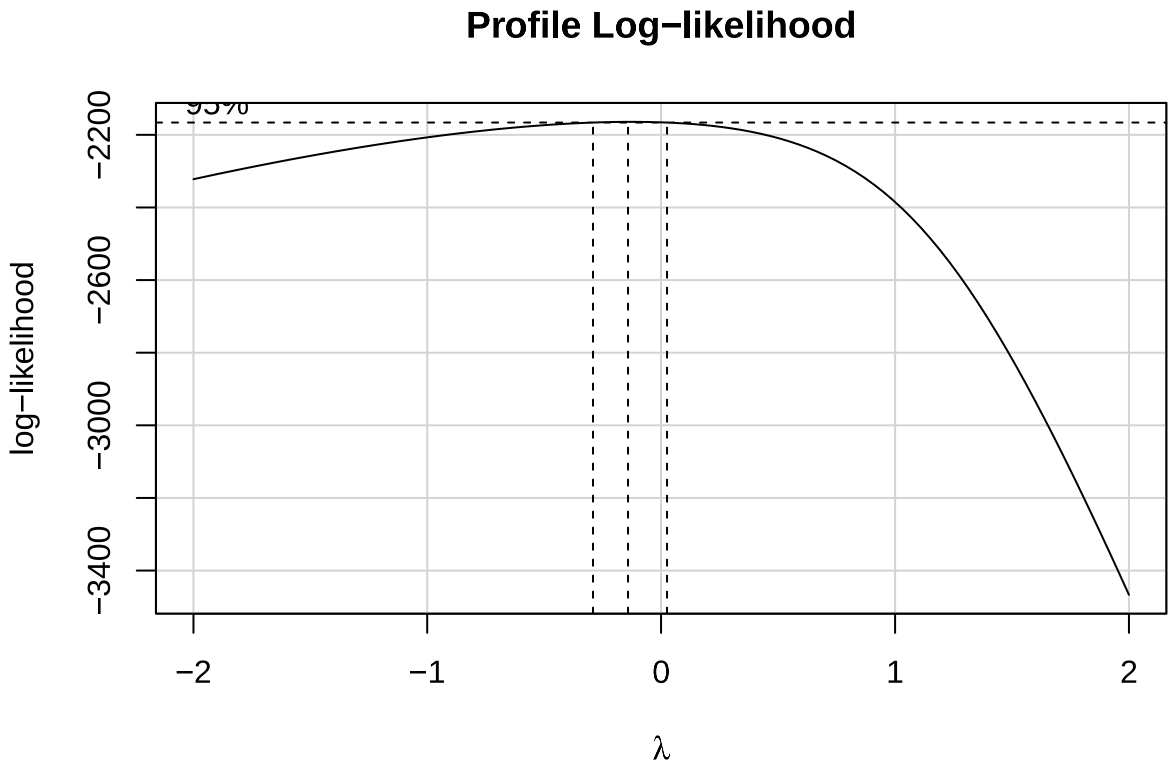


```
qqPlot(resids)
```



```
## 1095  532
##  730  354
```

```
dfnewtrain = dftrain
dfnewtrain$Approved_Conversion = dfnewtrain$Approved_Conversion + 1
box_model = lm(Approved_Conversion ~ xyz_campaign_id + age + gender + Impressions + Total_Conversion, da
bc = boxCox(box_model)
```



```
opt.lambda<-bc$x[which.max(bc$y)]
bc
```

```
## $x
##   [1] -2.00000000 -1.95959596 -1.91919192 -1.87878788 -1.83838384 -1.79797980
##   [7] -1.75757576 -1.71717172 -1.67676768 -1.63636364 -1.59595960 -1.55555556
##  [13] -1.51515152 -1.47474747 -1.43434343 -1.39393939 -1.35353535 -1.31313131
##  [19] -1.27272727 -1.23232323 -1.19191919 -1.15151515 -1.11111111 -1.07070707
##  [25] -1.03030303 -0.98989899 -0.94949495 -0.90909091 -0.86868687 -0.82828283
##  [31] -0.78787879 -0.74747475 -0.70707071 -0.66666667 -0.62626263 -0.58585859
##  [37] -0.54545455 -0.50505051 -0.46464646 -0.42424242 -0.38383838 -0.34343434
##  [43] -0.30303030 -0.26262626 -0.22222222 -0.18181818 -0.14141414 -0.10101010
##  [49] -0.06060606 -0.02020202  0.02020202  0.06060606  0.10101010  0.14141414
##  [55]  0.18181818  0.22222222  0.26262626  0.30303030  0.34343434  0.38383838
##  [61]  0.42424242  0.46464646  0.50505051  0.54545455  0.58585859  0.62626263
##  [67]  0.66666667  0.70707071  0.74747475  0.78787879  0.82828283  0.86868687
##  [73]  0.90909091  0.94949495  0.98989899  1.03030303  1.07070707  1.11111111
##  [79]  1.15151515  1.19191919  1.23232323  1.27272727  1.31313131  1.35353535
##  [85]  1.39393939  1.43434343  1.47474747  1.51515152  1.55555556  1.59595960
##  [91]  1.63636364  1.67676768  1.71717172  1.75757576  1.79797980  1.83838384
##  [97]  1.87878788  1.91919192  1.95959596  2.00000000
##
## $y
##   [1] -2322.203 -2316.577 -2311.022 -2305.540 -2300.133 -2294.800 -2289.544
##   [8] -2284.365 -2279.265 -2274.245 -2269.307 -2264.452 -2259.681 -2254.995
##  [15] -2250.397 -2245.888 -2241.469 -2237.143 -2232.911 -2228.775 -2224.737
##  [22] -2220.800 -2216.964 -2213.233 -2209.609 -2206.094 -2202.691 -2199.403
##  [29] -2196.233 -2193.184 -2190.259 -2187.461 -2184.795 -2182.264 -2179.873
##  [36] -2177.627 -2175.529 -2173.587 -2171.805 -2170.190 -2168.750 -2167.490
##  [43] -2166.421 -2165.550 -2164.888 -2164.447 -2164.237 -2164.274 -2164.571
##  [50] -2165.145 -2166.015 -2167.202 -2168.728 -2170.617 -2172.900 -2175.606
##  [57] -2178.771 -2182.433 -2186.634 -2191.421 -2196.845 -2202.962 -2209.834
##  [64] -2217.525 -2226.108 -2235.658 -2246.254 -2257.982 -2270.927 -2285.178
##  [71] -2300.826 -2317.957 -2336.657 -2357.007 -2379.078 -2402.939 -2428.639
##  [78] -2456.220 -2485.710 -2517.117 -2550.440 -2585.657 -2622.735 -2661.627
##  [85] -2702.270 -2744.598 -2788.529 -2833.981 -2880.865 -2929.091 -2978.568
##  [92] -3029.207 -3080.921 -3133.627 -3187.245 -3241.698 -3296.921 -3352.852
##  [99] -3409.426 -3466.579
```

```
opt.lambda
```

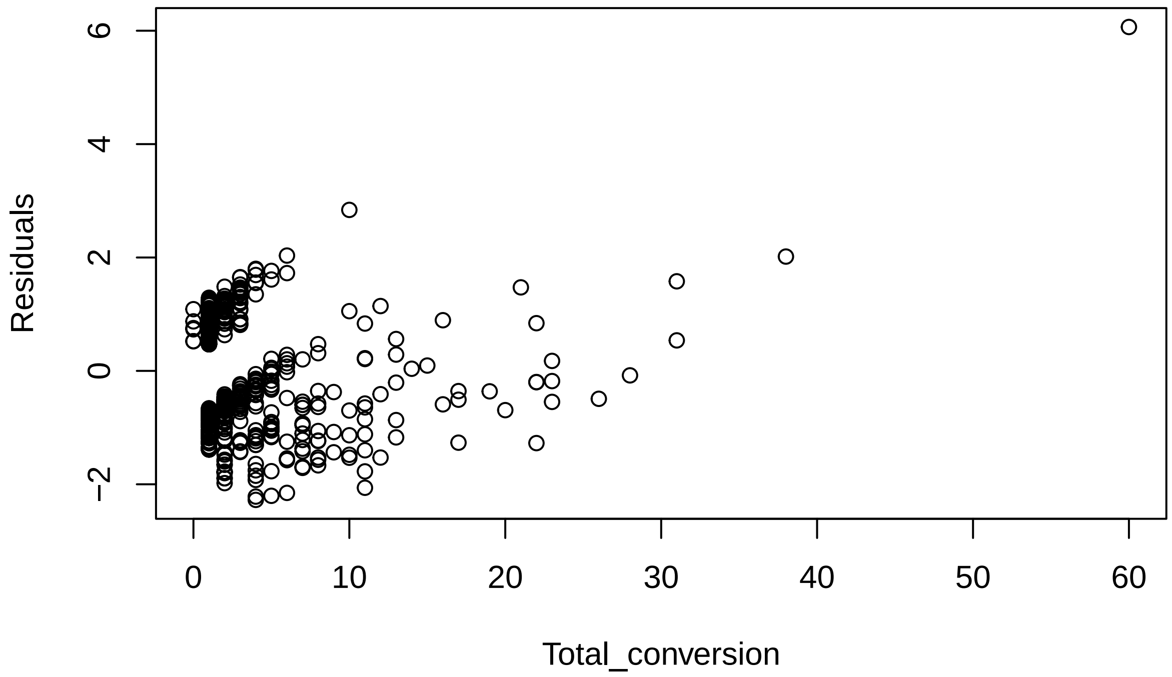
```
## [1] -0.1414141
```

boxcox -0.222

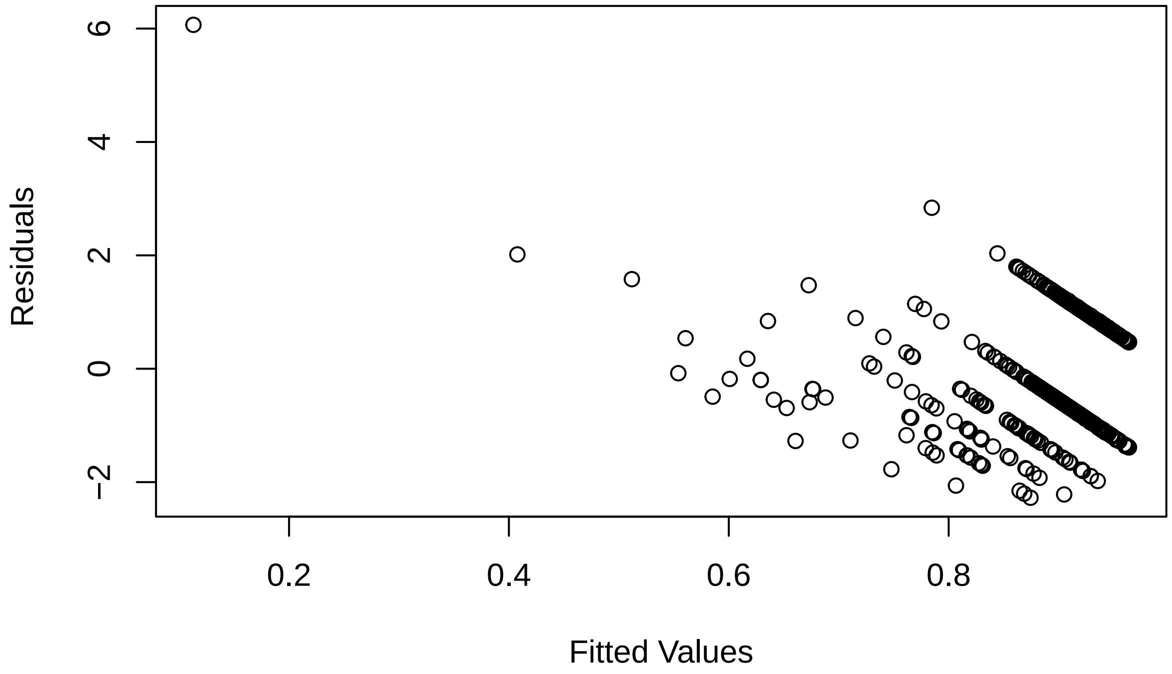
```
newmodel = lm((Approved_Conversion**-0.222) ~ xyz_campaign_id + age + gender + Impressions + Total_Conve
summary(newmodel)
```

```
## 
## Call:
## lm(formula = (Approved_Conversion^-0.222) ~ xyz_campaign_id + 
##     age + gender + Impressions + Total_Conversion, data = dfnewtrain)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.17489 -0.07422  0.01615  0.06699  0.41339 
## 
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)    
## (Intercept)          9.497e-01  1.380e-02  68.834  < 2e-16 ***
## xyz_campaign_id936   2.745e-03  1.367e-02   0.201  0.84092    
## xyz_campaign_id1178 -1.752e-02  1.384e-02  -1.266  0.20603    
## age35-39             7.384e-03  7.578e-03   0.974  0.33021    
## age40-44             1.318e-02  8.109e-03   1.625  0.10457    
## age45-49             2.310e-02  7.832e-03   2.950  0.00328 ** 
## genderM             -8.751e-03  5.695e-03  -1.537  0.12480    
## Impressions         -3.864e-08  1.664e-08  -2.321  0.02053 *  
## Total_Conversion    -1.154e-02  1.074e-03 -10.747  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.07715 on 753 degrees of freedom
## Multiple R-squared:  0.4494, Adjusted R-squared:  0.4436 
## F-statistic: 76.84 on 8 and 753 DF,  p-value: < 2.2e-16
```

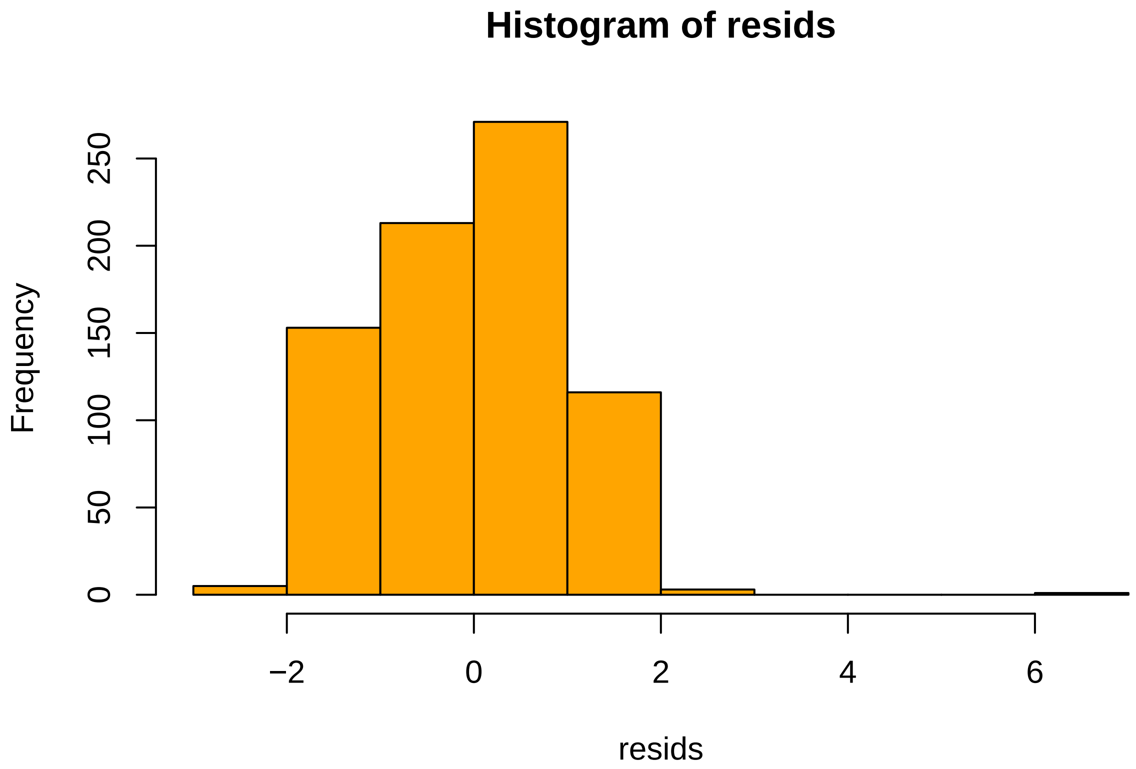
```
resids =rstandard(newmodel)
plot(dfnewtrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



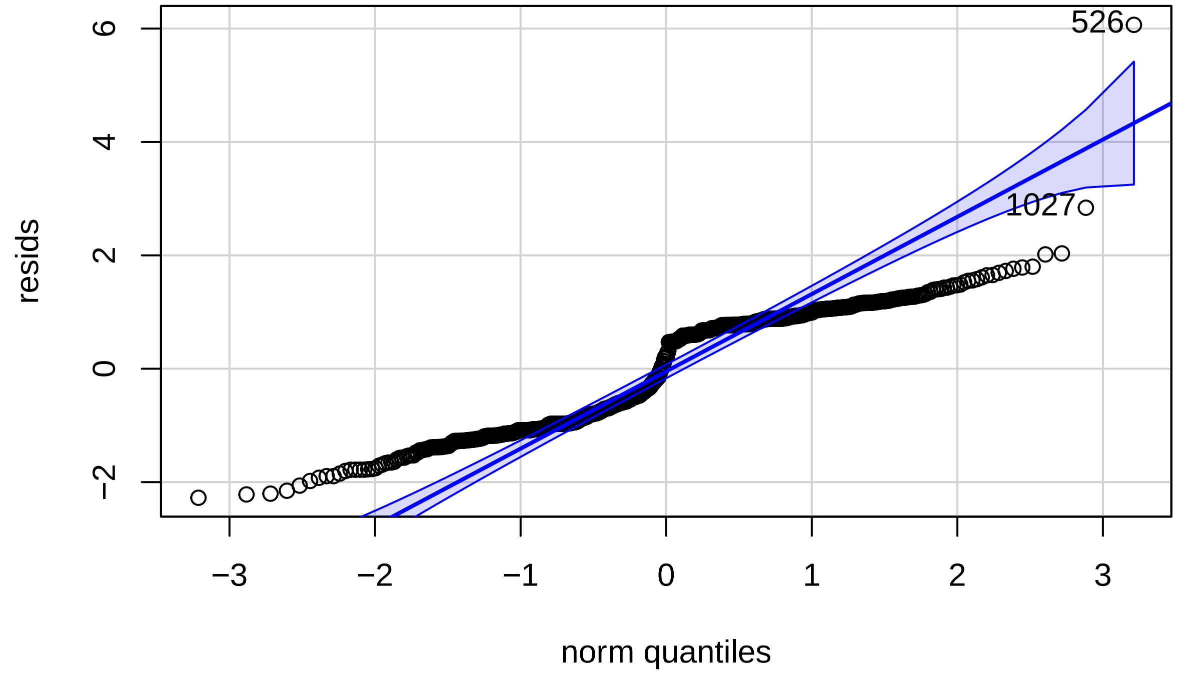
```
plot(newmodel$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```r
hist(resids, col="orange")
```



```
qqPlot(resids)
```

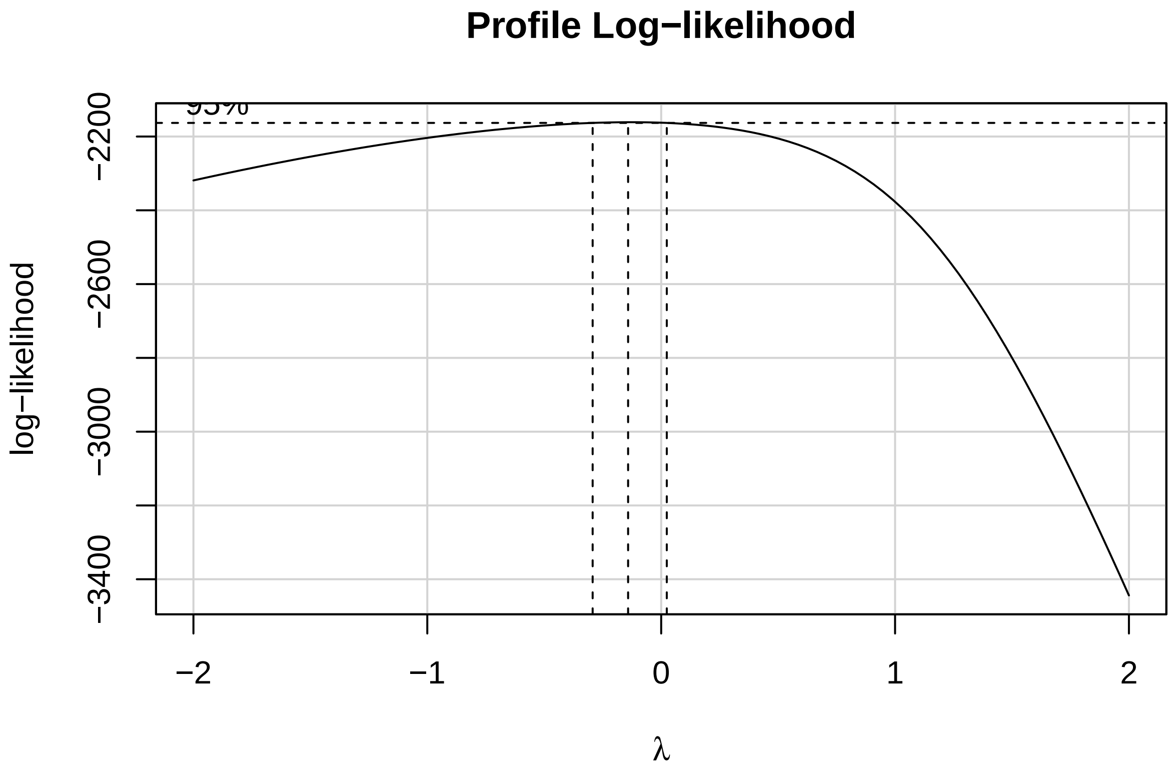


```
##  526 1027
##  351  685
```

```
summary(modelc)
```

```
##
## Call:
## lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender +
##     interest + Spent + Total_Conversion, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7581 -0.4481 -0.2086  0.5415  5.1795
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.0084351  0.1491998   0.057   0.9549
## xyz_campaign_id936  -0.0084156  0.1467304  -0.057   0.9543
## xyz_campaign_id1178  0.0420798  0.1496032   0.281   0.7786
## age35-39            -0.0593319  0.0815315  -0.728   0.4670
## age40-44            -0.0784839  0.0879462  -0.892   0.3725
## age45-49            -0.1648435  0.0868948  -1.897   0.0582 .
## genderM              0.1229708  0.0621760   1.978   0.0483 *
## interest26-50        0.0700540  0.0698633   1.003   0.3163
## interest51-75        0.1142805  0.0999351   1.144   0.2532
## interest76-100      -0.9239827  0.3742771  -2.469   0.0138 *
## interest101-125     -0.2964605  0.1239418  -2.392   0.0170 *
## Spent               -0.0009213  0.0005767  -1.598   0.1106
## Total_Conversion     0.3383856  0.0102186  33.115   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8268 on 749 degrees of freedom
## Multiple R-squared:  0.7729, Adjusted R-squared:  0.7693
## F-statistic: 212.4 on 12 and 749 DF,  p-value: < 2.2e-16
```

```
box_model = lm(formula = Approved_Conversion ~ xyz_campaign_id + age + gender + interest + Spent + Total
bc = boxCox(box_model)
```



```
opt.lambda<-bc$x[which.max(bc$y)]
bc
```

```
## $x
##   [1] -2.00000000 -1.95959596 -1.91919192 -1.87878788 -1.83838384 -1.79797980
##   [7] -1.75757576 -1.71717172 -1.67676768 -1.63636364 -1.59595960 -1.55555556
##  [13] -1.51515152 -1.47474747 -1.43434343 -1.39393939 -1.35353535 -1.31313131
##  [19] -1.27272727 -1.23232323 -1.19191919 -1.15151515 -1.11111111 -1.07070707
##  [25] -1.03030303 -0.98989899 -0.94949495 -0.90909091 -0.86868687 -0.82828283
##  [31] -0.78787879 -0.74747475 -0.70707071 -0.66666667 -0.62626263 -0.58585859
##  [37] -0.54545455 -0.50505051 -0.46464646 -0.42424242 -0.38383838 -0.34343434
##  [43] -0.30303030 -0.26262626 -0.22222222 -0.18181818 -0.14141414 -0.10101010
##  [49] -0.06060606 -0.02020202  0.02020202  0.06060606  0.10101010  0.14141414
##  [55]  0.18181818  0.22222222  0.26262626  0.30303030  0.34343434  0.38383838
##  [61]  0.42424242  0.46464646  0.50505051  0.54545455  0.58585859  0.62626263
##  [67]  0.66666667  0.70707071  0.74747475  0.78787879  0.82828283  0.86868687
##  [73]  0.90909091  0.94949495  0.98989899  1.03030303  1.07070707  1.11111111
##  [79]  1.15151515  1.19191919  1.23232323  1.27272727  1.31313131  1.35353535
##  [85]  1.39393939  1.43434343  1.47474747  1.51515152  1.55555556  1.59595960
##  [91]  1.63636364  1.67676768  1.71717172  1.75757576  1.79797980  1.83838384
##  [97]  1.87878788  1.91919192  1.95959596  2.00000000
## 
## $y
##   [1] -2319.018 -2313.388 -2307.829 -2302.344 -2296.933 -2291.597 -2286.337
##   [8] -2281.155 -2276.052 -2271.029 -2266.088 -2261.230 -2256.456 -2251.768
##  [15] -2247.168 -2242.657 -2238.237 -2233.909 -2229.676 -2225.540 -2221.502
##  [22] -2217.564 -2213.729 -2209.999 -2206.376 -2202.864 -2199.464 -2196.179
##  [29] -2193.013 -2189.968 -2187.048 -2184.256 -2181.596 -2179.073 -2176.690
##  [36] -2174.452 -2172.364 -2170.432 -2168.661 -2167.058 -2165.629 -2164.382
##  [43] -2163.325 -2162.468 -2161.819 -2161.389 -2161.192 -2161.239 -2161.545
##  [50] -2162.126 -2163.001 -2164.188 -2165.711 -2167.592 -2169.859 -2172.542
##  [57] -2175.675 -2179.293 -2183.438 -2188.153 -2193.488 -2199.497 -2206.237
##  [64] -2213.771 -2222.168 -2231.500 -2241.844 -2253.282 -2265.897 -2279.775
##  [71] -2295.005 -2311.673 -2329.863 -2349.658 -2371.130 -2394.350 -2419.371
##  [78] -2446.240 -2474.991 -2505.637 -2538.187 -2572.623 -2608.921 -2647.041
##  [85] -2686.927 -2728.517 -2771.738 -2816.509 -2862.746 -2910.361 -2959.265
##  [92] -3009.371 -3060.590 -3112.840 -3166.040 -3220.111 -3274.988 -3330.606
##  [99] -3386.898 -3443.799
```

```
opt.lambda
```

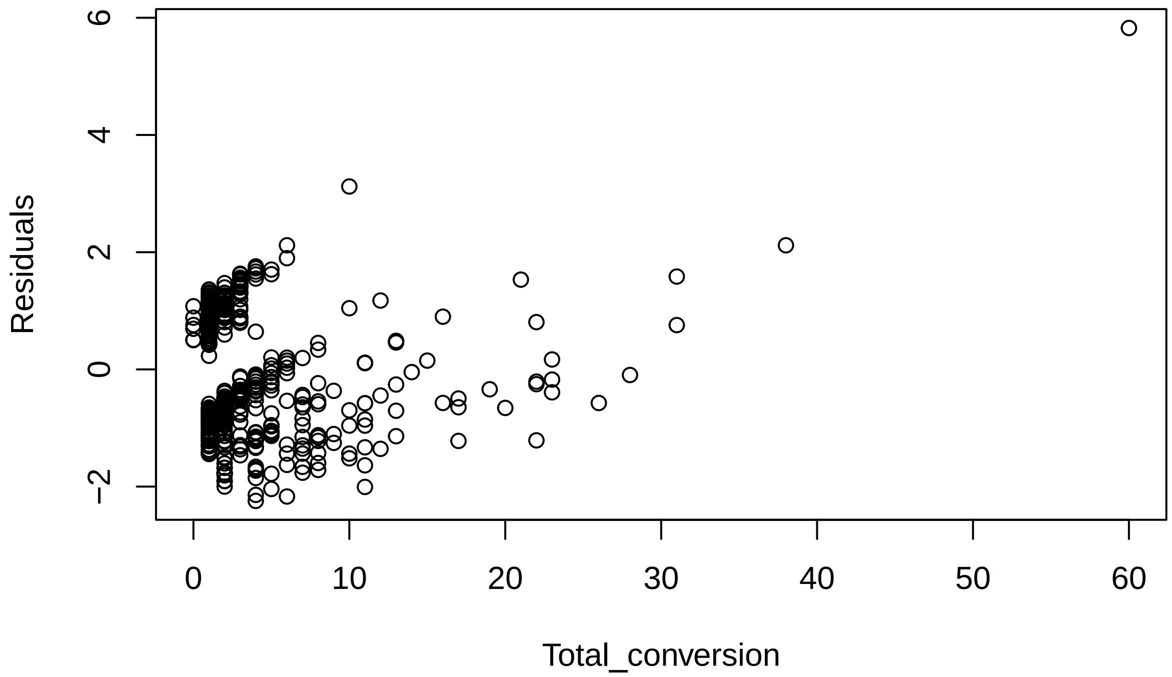
```
## [1] -0.1414141
```

box cox -0.2626

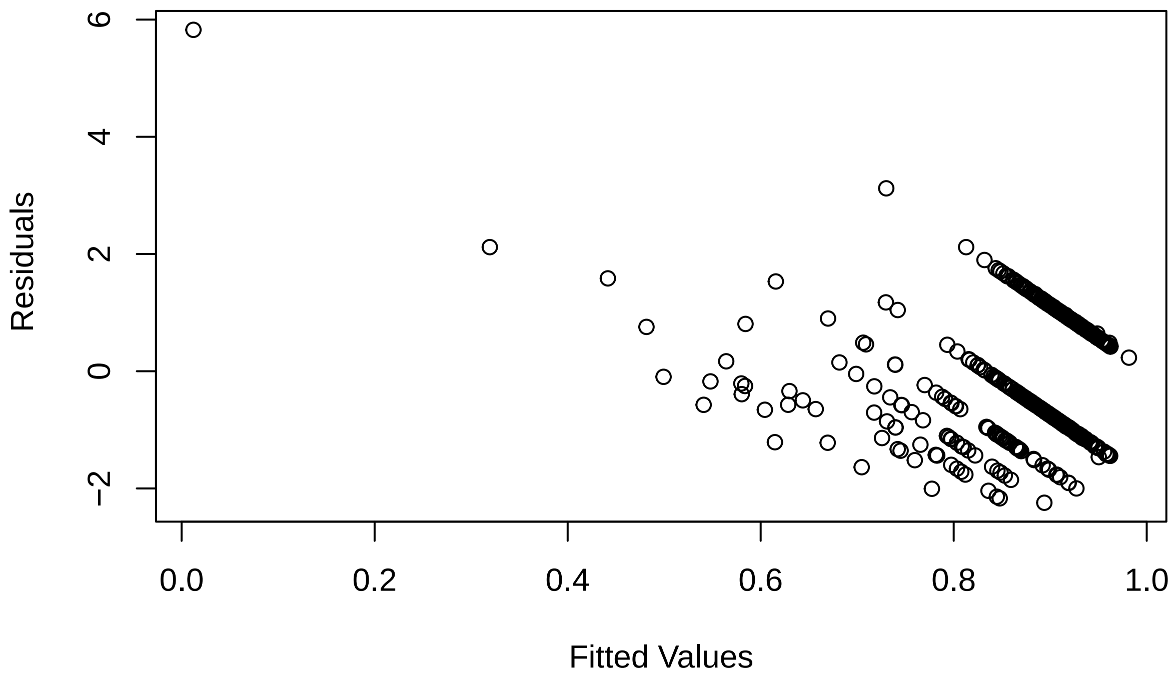
```
newmodel2 = lm((Approved_Conversion**-0.26262) ~ xyz_campaign_id + age + gender + interest + Spent + To
summary(newmodel)
```

```
## 
## Call:
## lm(formula = (Approved_Conversion^-0.222) ~ xyz_campaign_id + 
##     age + gender + Impressions + Total_Conversion, data = dfnewtrain)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.17489 -0.07422  0.01615  0.06699  0.41339 
## 
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)    
## (Intercept)          9.497e-01  1.380e-02  68.834  < 2e-16 ***
## xyz_campaign_id936   2.745e-03  1.367e-02   0.201  0.84092    
## xyz_campaign_id1178 -1.752e-02  1.384e-02  -1.266  0.20603
## age35-39             7.384e-03  7.578e-03   0.974  0.33021
## age40-44             1.318e-02  8.109e-03   1.625  0.10457
## age45-49             2.310e-02  7.832e-03   2.950  0.00328 **
## genderM             -8.751e-03  5.695e-03  -1.537  0.12480
## Impressions         -3.864e-08  1.664e-08  -2.321  0.02053 *
## Total_Conversion    -1.154e-02  1.074e-03 -10.747  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07715 on 753 degrees of freedom
## Multiple R-squared:  0.4494, Adjusted R-squared:  0.4436
## F-statistic: 76.84 on 8 and 753 DF,  p-value: < 2.2e-16
```

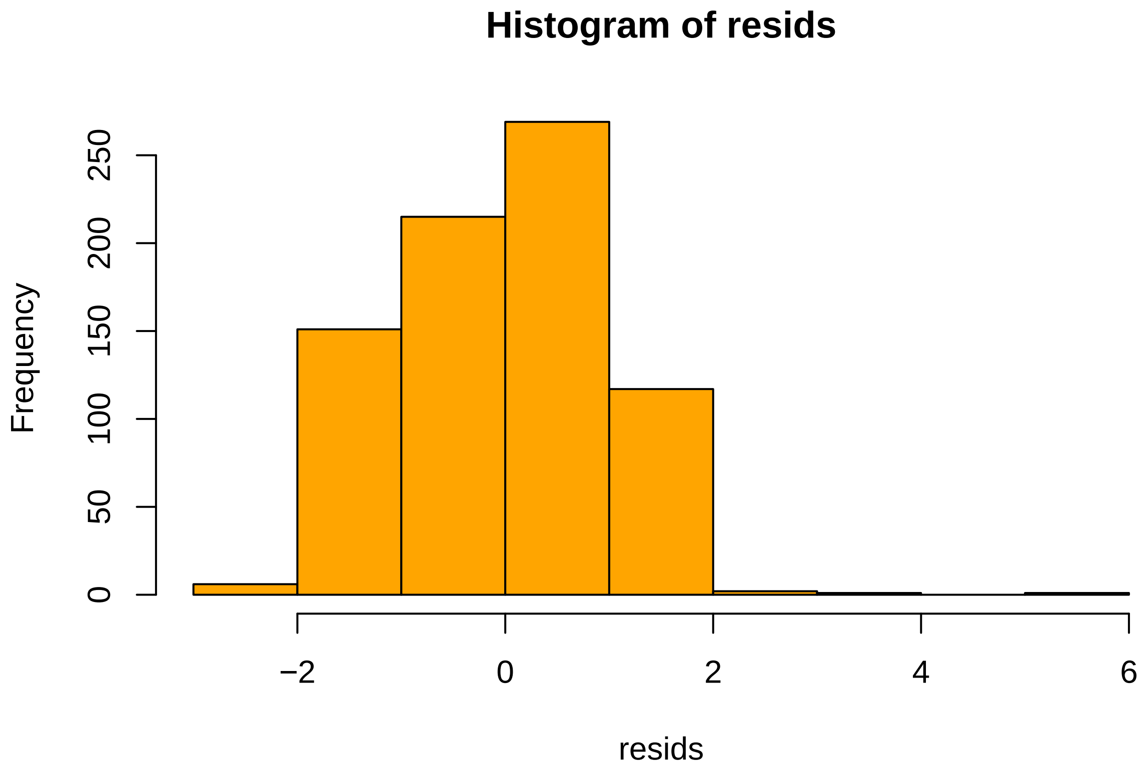
```
resids =rstandard(newmodel2)
plot(dfnewtrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



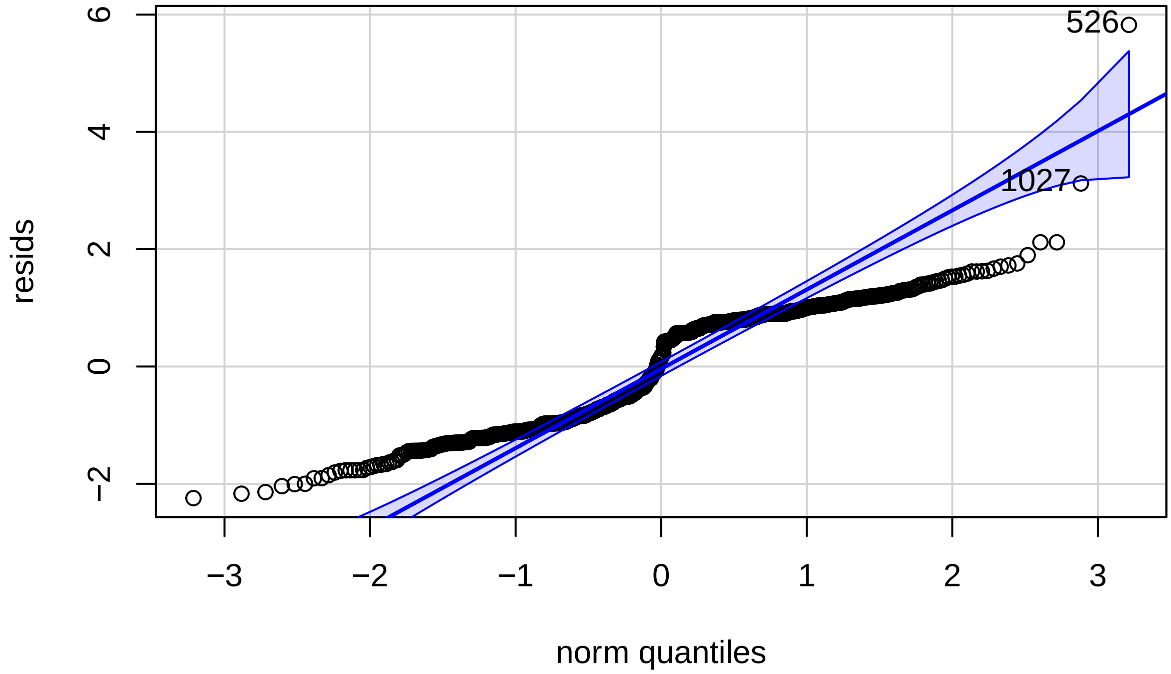
```
plot(newmodel2$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```



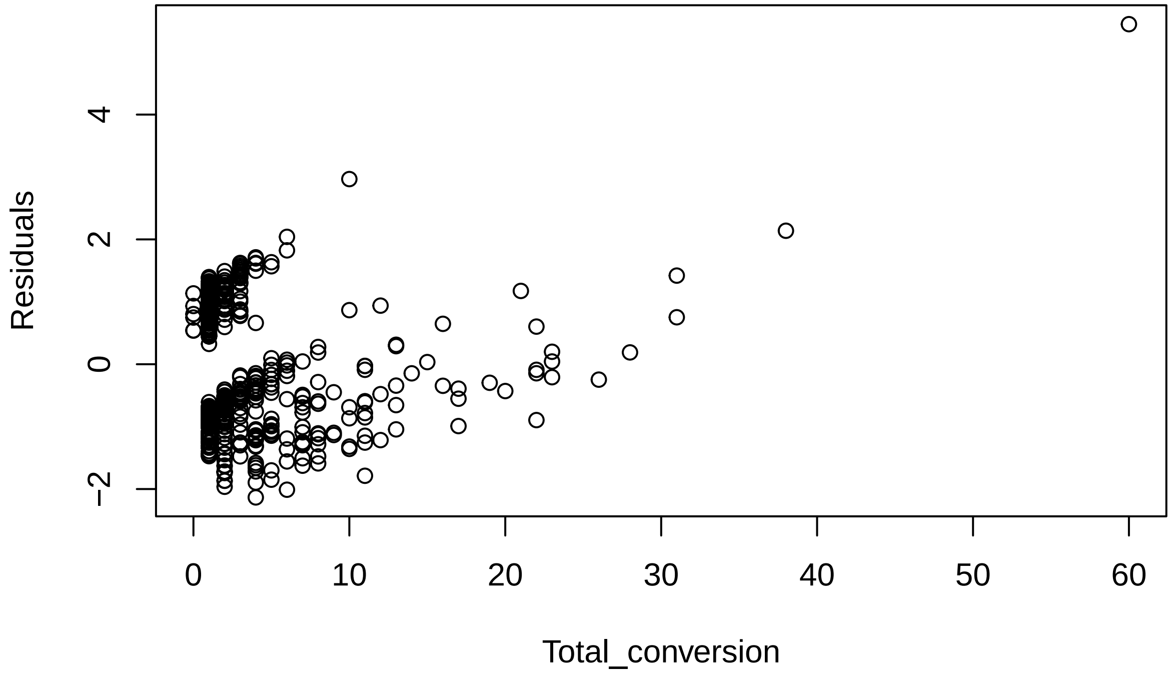
```
##  526 1027
##  351  685
```

Approved conversion ** -0.5

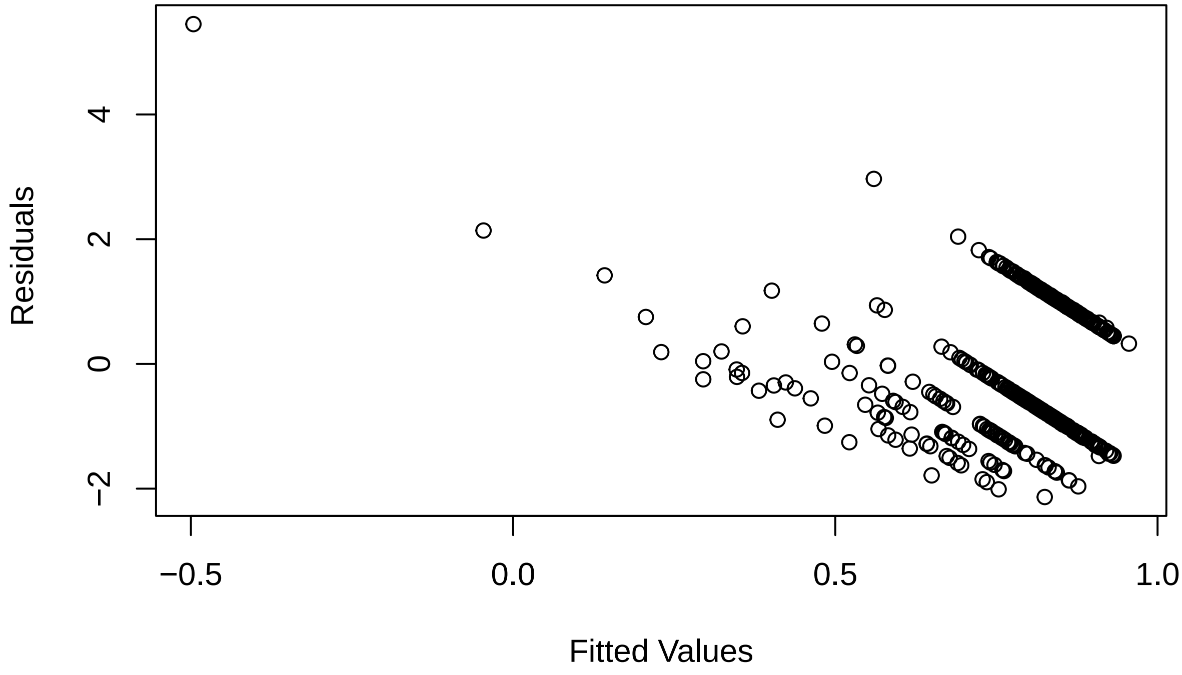
```
newmodel3 = lm((Approved_Conversion**-0.5) ~ xyz_campaign_id + age + gender + interest + Spent + Total_
summary(newmodel3)
```

```
##
## Call:
## lm(formula = (Approved_Conversion^-0.5) ~ xyz_campaign_id + age +
##     gender + interest + Spent + Total_Conversion, data = dfnewtrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32492 -0.14720  0.00885  0.13341  0.73178
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.8942136  0.0277124  32.268  < 2e-16 ***
## xyz_campaign_id936     0.0062263  0.0272537   0.228  0.81935
## xyz_campaign_id1178 -0.0325226  0.0277874  -1.170  0.24221
## age35-39            0.0167413  0.0151437   1.105  0.26930
## age40-44            0.0290092  0.0163352   1.776  0.07616 .
## age45-49            0.0503511  0.0161399   3.120  0.00188 **
## genderM            -0.0215801  0.0115486  -1.869  0.06206 .
## interest26-50       0.0006052  0.0129764   0.047  0.96282
## interest51-75      -0.0214185  0.0185620  -1.154  0.24891
## interest76-100      0.1007274  0.0695184   1.449  0.14778
## interest101-125    -0.0265127  0.0230210  -1.152  0.24982
## Spent              -0.0002993  0.0001071  -2.794  0.00533 **
## Total_Conversion   -0.0190774  0.0018980 -10.051  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1536 on 749 degrees of freedom
## Multiple R-squared:  0.388,  Adjusted R-squared:  0.3782
## F-statistic: 39.58 on 12 and 749 DF,  p-value: < 2.2e-16
```

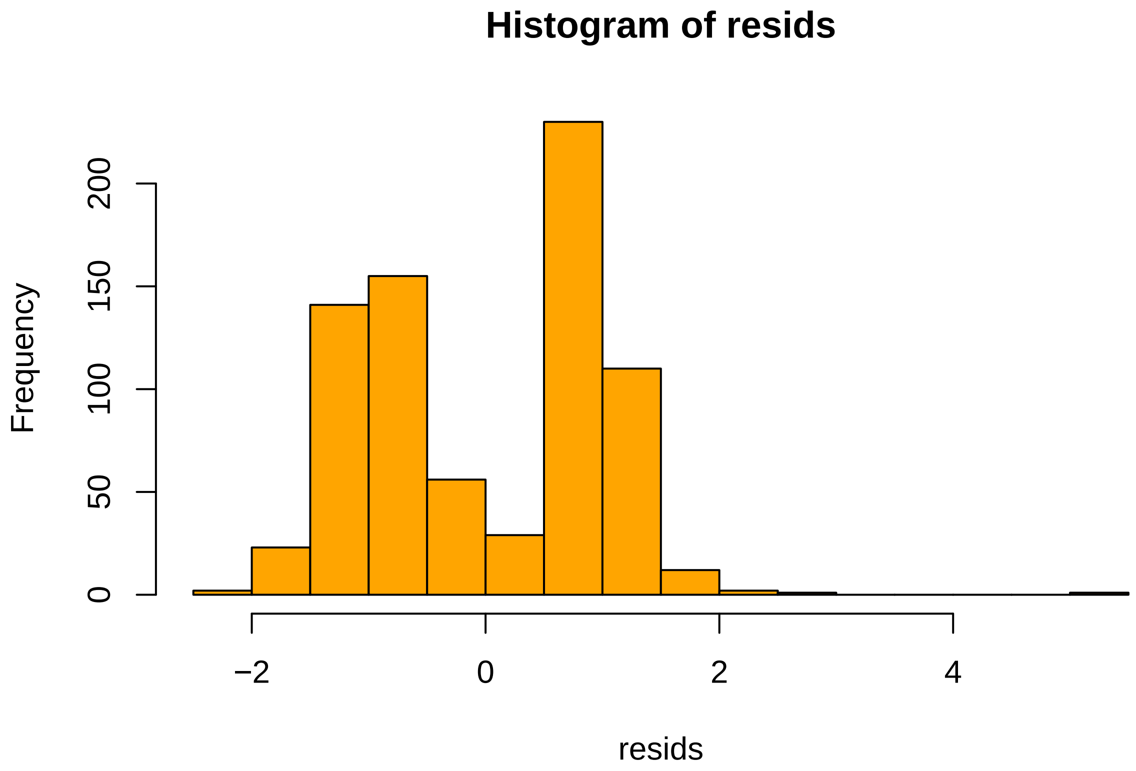
```
resids =rstandard(newmodel3)
plot(dfnewtrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



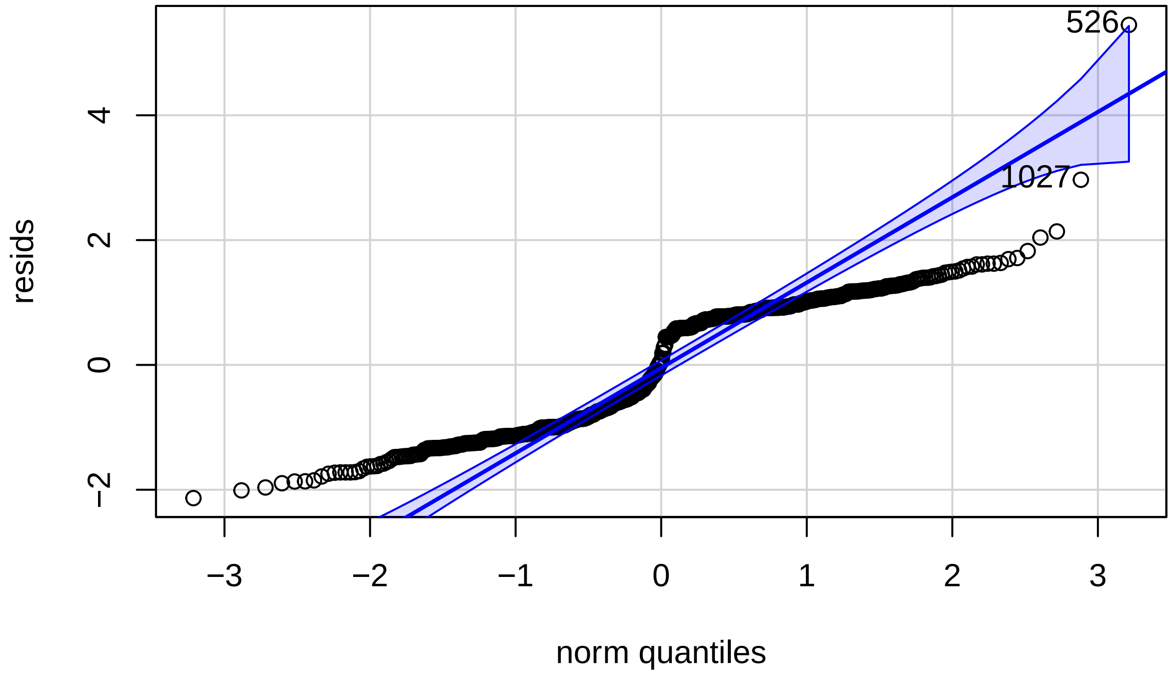
```
plot(newmodel3$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```
hist(resids, col="orange")
```



```
qqPlot(resids)
```



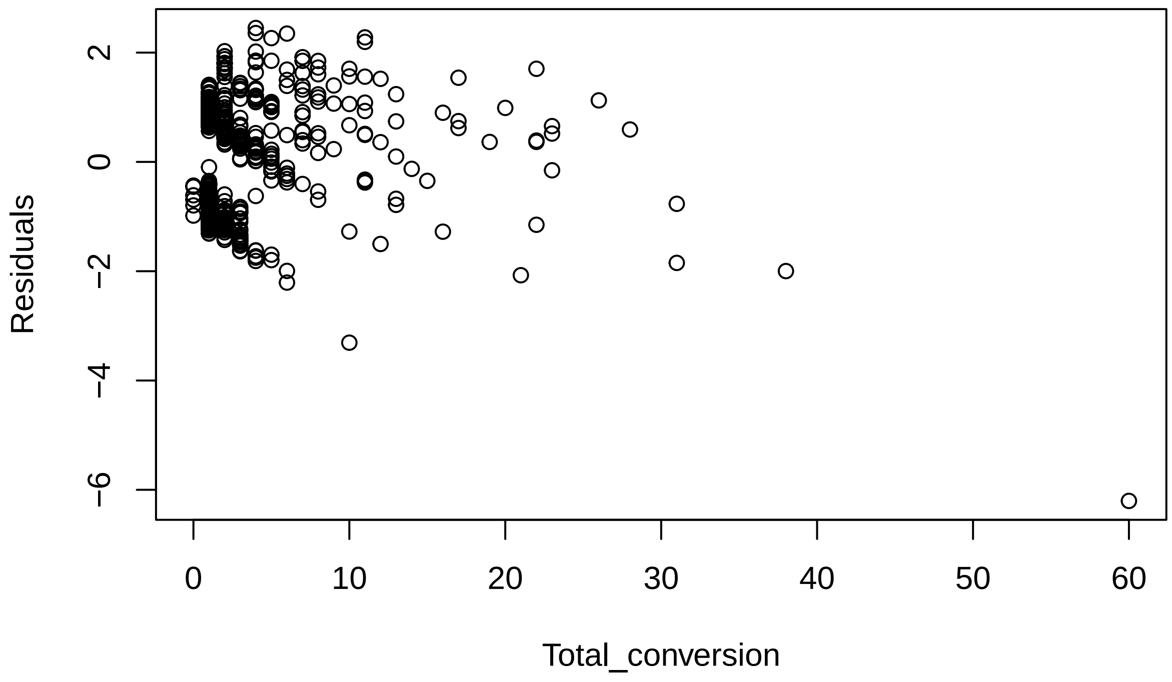
```
##  526 1027
##  351  685
```

#log approved conversion

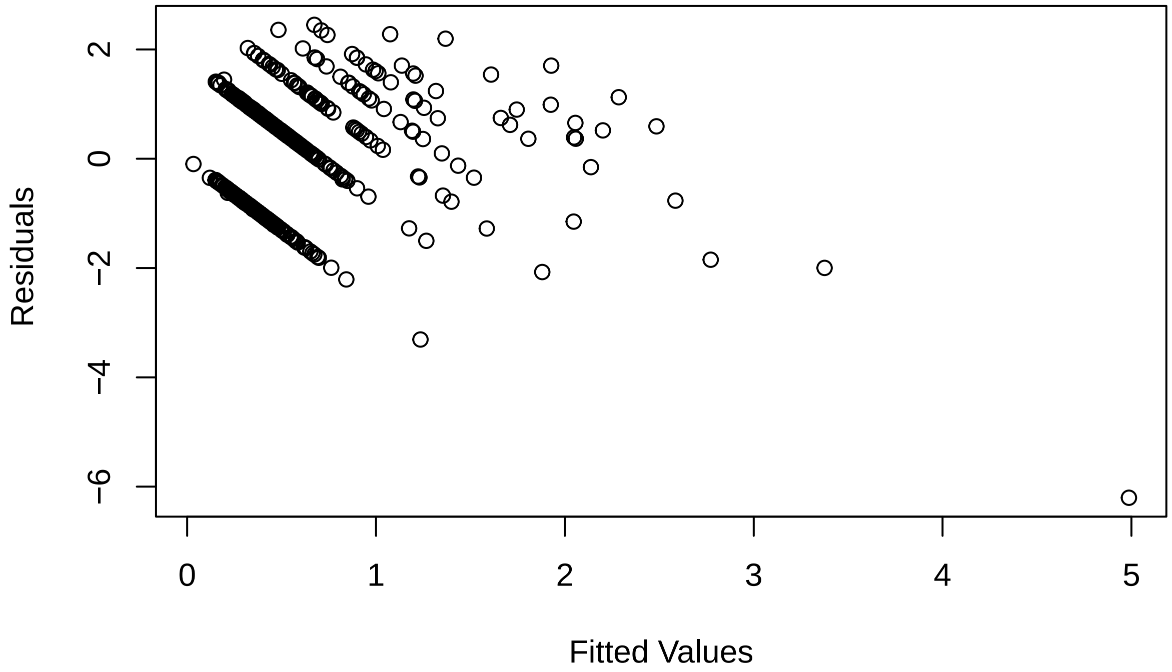
```
newmodel4 = lm(log(Approved_Conversion) ~ xyz_campaign_id + age + gender + interest + Spent + Total_Con
summary(newmodel4)
```

```
##
## Call:
## lm(formula = log(Approved_Conversion) ~ xyz_campaign_id + age +
##     gender + interest + Spent + Total_Conversion, data = dfnewtrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0965 -0.3170 -0.1270  0.3531  0.9360
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.2182986  0.0697334   3.130  0.00181 **
## xyz_campaign_id936   -0.0119476  0.0685792  -0.174  0.86174
## xyz_campaign_id1178  0.0770313  0.0699219   1.102  0.27096
## age35-39             -0.0395639  0.0381064  -1.038  0.29949
## age40-44             -0.0713331  0.0411045  -1.735  0.08308 .
## age45-49             -0.1260899  0.0406131  -3.105  0.00198 **
## genderM                0.0554339  0.0290600   1.908  0.05683 .
## interest26-50          0.0078492  0.0326529   0.240  0.81010
## interest51-75          0.0548959  0.0467079   1.175  0.24025
## interest76-100        -0.3027886  0.1749305  -1.731  0.08388 .
## interest101-125        0.0366593  0.0579282   0.633  0.52703
## Spent                  0.0006565  0.0002695   2.435  0.01510 *
## Total_Conversion       0.0702666  0.0047760  14.712  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3864 on 749 degrees of freedom
## Multiple R-squared:  0.5151, Adjusted R-squared:  0.5073
## F-statistic: 66.29 on 12 and 749 DF,  p-value: < 2.2e-16
```

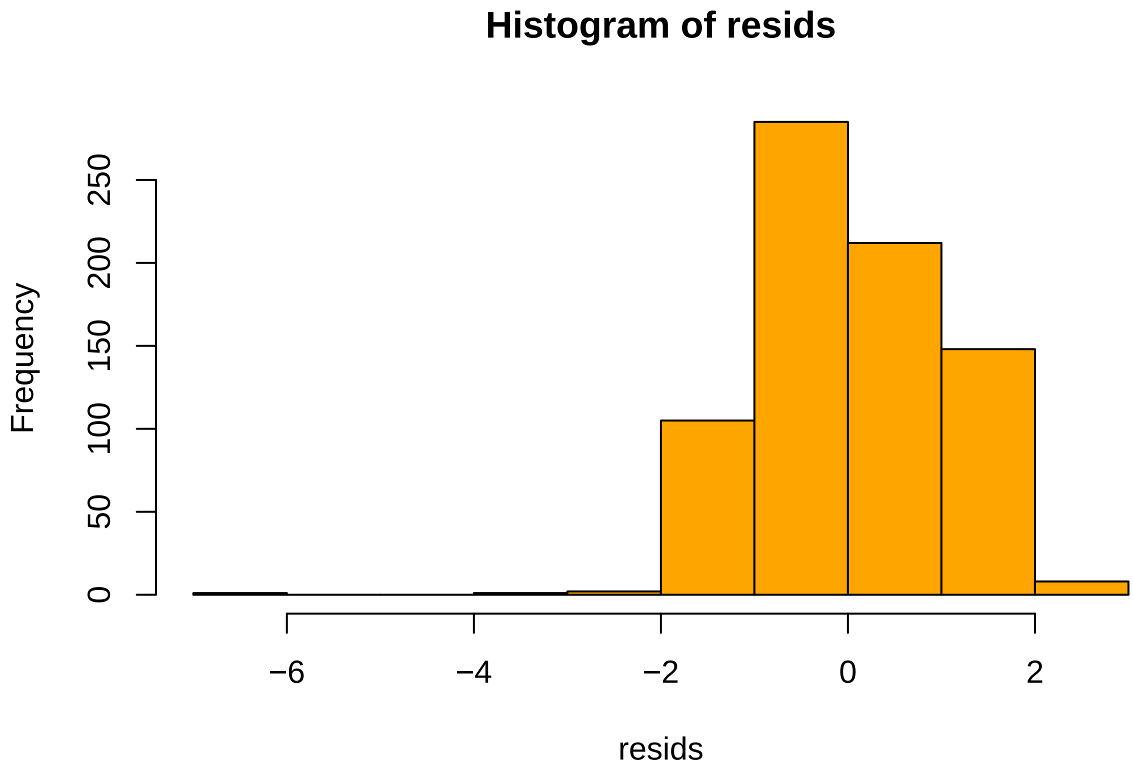
```
resids =rstandard(newmodel4)
plot(dfnewtrain$Total_Conversion, resids, xlab= "Total_conversion", ylab = "Residuals")
```



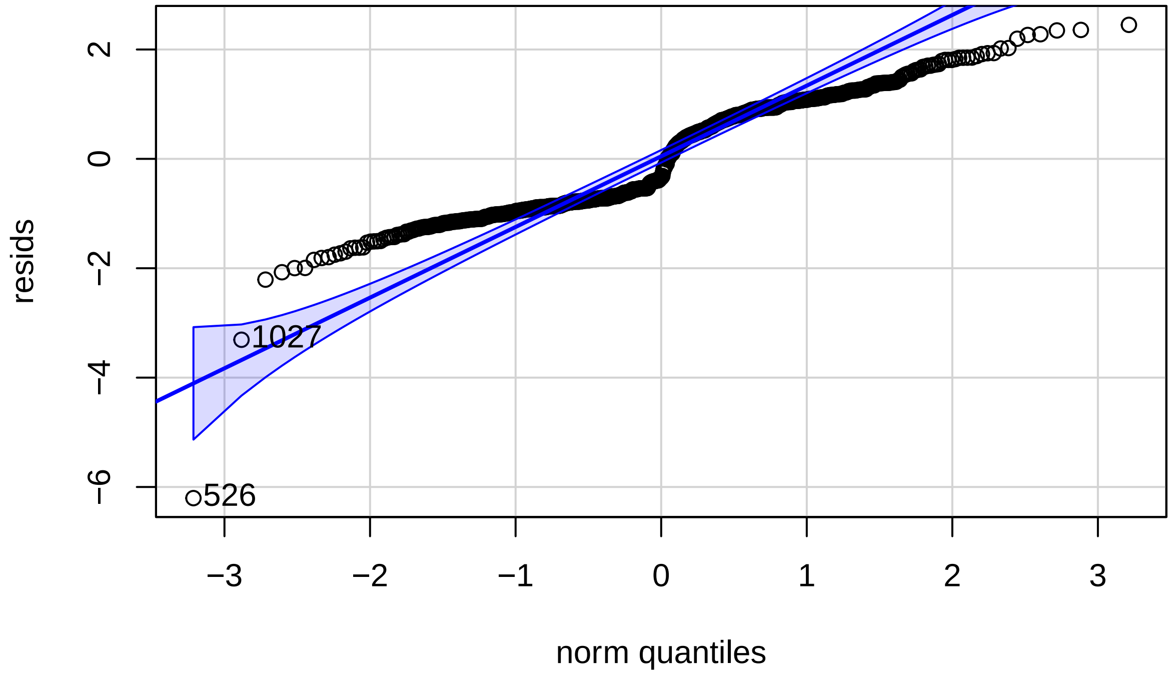
```
plot(newmodel4$fitted.values, resids, xlab="Fitted Values", ylab=" Residuals")
```



```r
hist(resids, col="orange")
```



```
qqPlot(resids)
```



```
## 526 1027
## 351  685
```

```
m = predict(model, dftest, interval="prediction")
m2 = predict(modelb, dftest, interval="prediction")
m3 = predict(modelc, dftest, interval="prediction")
m4 = predict(modelca, dftest, interval="prediction")
m5 = predict(modeld, dftest, interval="prediction")
m6 = predict(modele, dftest, interval="prediction")
m7 = predict(modelf, dftest, interval="prediction")
m8 = predict(modelg, dftest, interval="prediction")
dfnew = dftest
dfnew$Approved_Conversion = log(dfnew$Approved_Conversion)
m9 = predict(newmodel, dfnew, interval="prediction")
```

```
sum((m-dftest$Approved_Conversion)^2)/sum((m-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7370549
```

```
sum((m2-dftest$Approved_Conversion)^2)/sum((m2-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7393274
```

```
sum((m3-dftest$Approved_Conversion)^2)/sum((m3-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7470722
```

```
sum((m4-dftest$Approved_Conversion)^2)/sum((m4-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7149312
```

```
sum((m5-dftest$Approved_Conversion)^2)/sum((m5-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7472712
```

```
sum((m6-dftest$Approved_Conversion)^2)/sum((m6-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7475782
```

```
sum((m7-dftest$Approved_Conversion)^2)/sum((m7-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.7556131
```

```
sum((m8-dftest$Approved_Conversion)^2)/sum((m8-mean(dftest$Approved_Conversion))^2)
```

```
## [1] 0.703249
```

```
#sum((m9-((dftest$Approved_Conversion+1)**-0.2222))^2)/sum((m9-mean(((dftest$Approved_Conversion+1)**-0
#dfnewtest = dftest
#dfnewtest$Approved_Conversion = dfnewtest$Approved_Conversion + 1
```

Fin.