

# Mobile Analytics

Yuzi Liu

11/19/2018

## Data Processing

```
getwd()
```

```
## [1] "/Users/yuziliu/Downloads"
```

```
mobile <- read.csv("Geo-Fence Analytics.csv", header = TRUE)
```

- 1a. Create dummy variable `imp_large` for the large impression:
- 1b. Create dummy variables `cat_entertainment`, `cat_social` and `cat_tech` for app categories:
- 1c. Create dummy variable `os_ios` for iOS devices:

```
mobile$imp_large <- ifelse(mobile$imp_size == "728x90", 1, 0)
```

```
mobile$cat_entertainment <- ifelse(mobile$app_topcat %in% c("IAB1","IAB1-6"), 1, 0)
```

```
mobile$cat_social <- ifelse(mobile$app_topcat == "IAB14", 1, 0)
```

```
mobile$cat_tech <- ifelse(mobile$app_topcat == "IAB19-6", 1, 0)
```

```
mobile$os_ios <- ifelse(mobile$device_os == "iOS", 1, 0)
```

- 1d. Create variable `distance` using Harvesine formula to calculate the distance for a pair of latitude/longitude coordinates.
- 1e. Create variable `distance_squared` by squaring variable `distance`
- 1f. Create variable `ln_app_review_vol` by taking natural log of `app_review_vol`

```
mobile$distance = 6371*acos(cos(as_radians(mobile$device_lat))*
```

```
cos(as_radians(mobile$geofence_lat))*cos(as_radians(mobile$device_lon)-as_radians(mobile$geofence_lon))
```

```
+sin(as_radians(mobile$device_lat))*sin(as_radians(mobile$geofence_lat)))
```

```
mobile$distance_squared = mobile$distance^2
```

```
mobile$ln_app_review_vol = log(mobile$app_review_vol)
```

# Descriptive Statistics

- 2a. Summarize the data by calculating the summary statistics (i.e., mean, median, std. dev., minimum and maximum) for didclick, distance, imp\_large, cat\_entertainment, cat\_social, cat\_tech, os\_ios, ln\_app\_review\_vol and app\_review\_val.

```
my_data <- as.data.frame(mobile[, c("didclick","distance","imp_large","cat_entertainm
ent","cat_social","cat_tech","os_ios","ln_app_review_vol","app_review_val")])
```

```
stargazer(my_data, type="text", median=TRUE, digits=4, title="Summary Statistics")
```

```
##
## Summary Statistics
## =====
=
## Statistic          N      Mean    St. Dev.   Min    Pctl(25) Median  Pctl(75)    Max
## -----
-
## didclick           121,567 0.0068    0.0822     0      0      0      0      1
## distance           121,567 2.9837    2.6485    0.0208  1.1032  2.0209  4.0292  11.786
7
## imp_large          121,567 0.2309    0.4214     0      0      0      0      1
## cat_entertainment  121,567 0.2839    0.4509     0      0      0      1      1
## cat_social          121,567 0.1251    0.3309     0      0      0      0      1
## cat_tech            121,567 0.5178    0.4997     0      0      1      1      1
## os_ios              121,567 0.2504    0.4332     0      0      0      1      1
## ln_app_review_vol  121,567 10.0568   0.6370    7.0809  9.7915  10.0872 10.0872  12.937
7
## app_review_val     121,567 3.6549    0.3608    1.4000  3.4000  3.4000  3.9000  4.7000
## -----
-
```

- 2b. Report the correlations among the above variables.

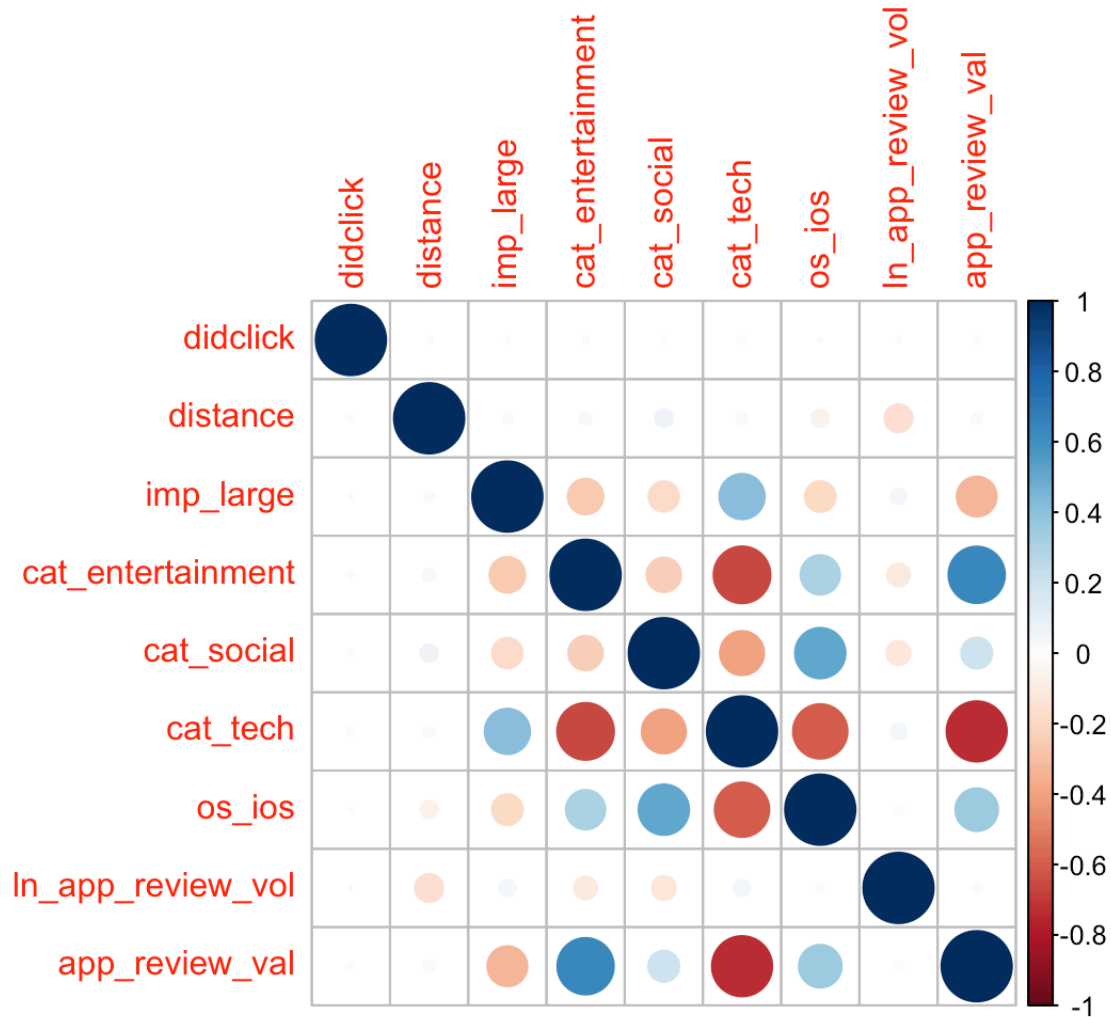
```
res <- cor(my_data)
res
```

```
##          didclick    distance    imp_large cat_entertainment
## didclick      1.000000000 -0.006628356 -0.004786218      -0.007117972
## distance     -0.006628356  1.000000000  0.020024918      -0.028992663
## imp_large    -0.004786218  0.020024918  1.000000000      -0.254731873
## cat_entertainment -0.007117972 -0.028992663 -0.254731873      1.000000000
## cat_social   -0.005623417  0.060484490 -0.185311155      -0.238133905
## cat_tech     0.012454366  0.023499545  0.414049273      -0.652575678
## os_ios       -0.002147325 -0.060281389 -0.190194050      0.312647684
## ln_app_review_vol 0.003982875 -0.157864184  0.049929790      -0.105545185
## app_review_val -0.006523592  0.022481133 -0.321439020      0.642212363
##          cat_social    cat_tech      os_ios ln_app_review_vol
## didclick      -0.005623417  0.01245437 -0.002147325      0.003982875
## distance      0.060484490  0.02349954 -0.060281389      -0.157864184
## imp_large     -0.185311155  0.41404927 -0.190194050      0.049929790
## cat_entertainment -0.238133905 -0.65257568  0.312647684      -0.105545185
## cat_social     1.000000000 -0.39192721  0.513672844      -0.115376574
## cat_tech      -0.391927215  1.00000000 -0.598919227      0.049503835
## os_ios        0.513672844 -0.59891923  1.000000000      -0.013523794
## ln_app_review_vol -0.115376574  0.04950383 -0.013523794      1.000000000
## app_review_val  0.194394425 -0.73206714  0.366139311      0.014457854
##          app_review_val
## didclick      -0.006523592
## distance      0.022481133
## imp_large     -0.321439020
## cat_entertainment 0.642212363
## cat_social     0.194394425
## cat_tech      -0.732067145
## os_ios        0.366139311
## ln_app_review_vol 0.014457854
## app_review_val 1.000000000
```

```
round(res, 4)
```

```
##          didclick distance imp_large cat_entertainment cat_social
## didclick      1.0000  -0.0066  -0.0048          -0.0071  -0.0056
## distance     -0.0066   1.0000   0.0200          -0.0290   0.0605
## imp_large     -0.0048   0.0200   1.0000          -0.2547  -0.1853
## cat_entertainment -0.0071 -0.0290 -0.2547           1.0000  -0.2381
## cat_social    -0.0056   0.0605  -0.1853          -0.2381   1.0000
## cat_tech       0.0125   0.0235   0.4140          -0.6526  -0.3919
## os_ios        -0.0021  -0.0603  -0.1902           0.3126   0.5137
## ln_app_review_vol 0.0040  -0.1579   0.0499          -0.1055  -0.1154
## app_review_val -0.0065   0.0225  -0.3214           0.6422   0.1944
##          cat_tech os_ios ln_app_review_vol app_review_val
## didclick      0.0125 -0.0021           0.0040      -0.0065
## distance      0.0235 -0.0603           -0.1579       0.0225
## imp_large      0.4140 -0.1902           0.0499      -0.3214
## cat_entertainment -0.6526 0.3126          -0.1055       0.6422
## cat_social     -0.3919 0.5137           -0.1154       0.1944
## cat_tech       1.0000 -0.5989           0.0495      -0.7321
## os_ios        -0.5989 1.0000          -0.0135       0.3661
## ln_app_review_vol 0.0495 -0.0135          1.0000       0.0145
## app_review_val -0.7321 0.3661           0.0145       1.0000
```

```
corrplot(res, method = "circle")
```



- 2c. Plot the relationship of distance (x-axis) and click-through-rate (y-axis), and any other pairs of variables of interest. Bin the continuous “distance” into discrete “distance\_group” where (0, 0.5] → 1, (0.5, 1] → 2, (1, 2] → 3, (2, 4] → 4, (4, 7] → 5, (7, 10] → 6, >10 → 7.

```
mobile$distance_group = ifelse(mobile$distance>0 & mobile$distance <= 0.5, 1,
                              ifelse(mobile$distance <= 1, 2,
                                      ifelse(mobile$distance <= 2, 3,
                                              ifelse(mobile$distance <= 4, 4,
                                                      ifelse(mobile$distance <= 7, 5,
                                                                ifelse(mobile$distance <=1
0, 6, 7)))))))

impressions = ddply(mobile, c("distance_group"), summarise, nrow = length(distance_g
roup))

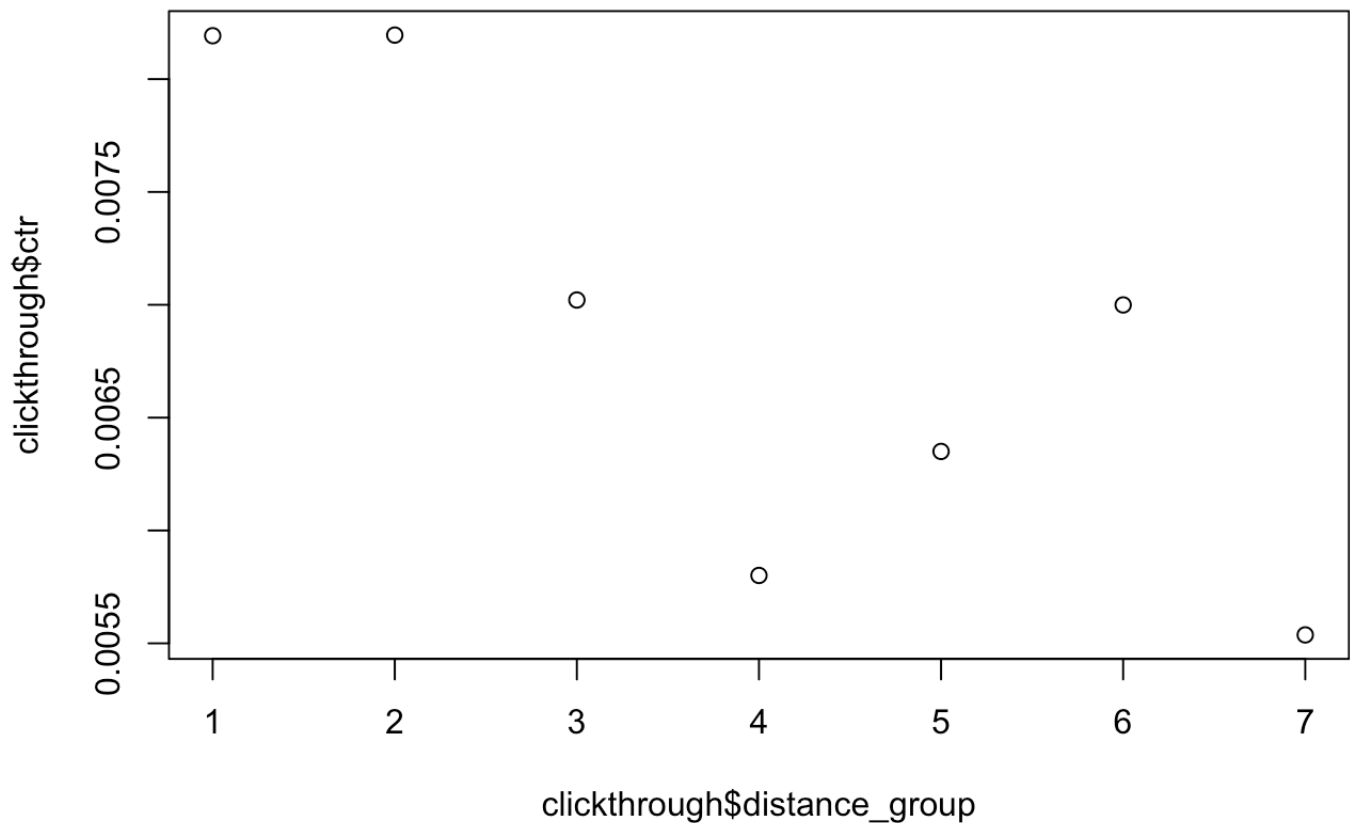
didclick_group = aggregate(mobile$didclick, by=list(distance_group=mobile$distance_gr
oup), FUN=sum)

clickthrough = merge(impressions, didclick_group, by = "distance_group")
clickthrough$ctr = clickthrough$x / clickthrough$nrow
clickthrough
```

distance_group <dbl>	nrow <int>	x <int>	ctr <dbl>
1	7324	60	0.008192245
2	19402	159	0.008195031
3	33470	235	0.007021213
4	30513	177	0.005800806
5	18267	116	0.006350249
6	7715	54	0.006999352
7	4876	27	0.005537326

7 rows

```
plot(clickthrough$distance_group,clickthrough$ctr)
```



## Logistics Regression

- a. Specify the following Logistic regression model: Dependent variable: didclick; Independent variables: distance, distance\_squared, imp\_large, cat\_entertainment, cat\_social, cat\_tech, os\_ios, ln\_app\_review\_vol and app\_review\_val.

Create a new data frame including all predictors.

```
mobile2 <- as.data.frame(mobile[,c("didclick", "distance", "distance_squared", "imp_l  
arge", "cat_entertainment",  
                                "cat_social", "cat_tech", "os_ios", "ln_app_review  
_vol", "app_review_val")])
```

Before fitting into the logistic regression model, let's see the correlation between the predictors.

```
res2 <- cor(mobile2)  
res2
```

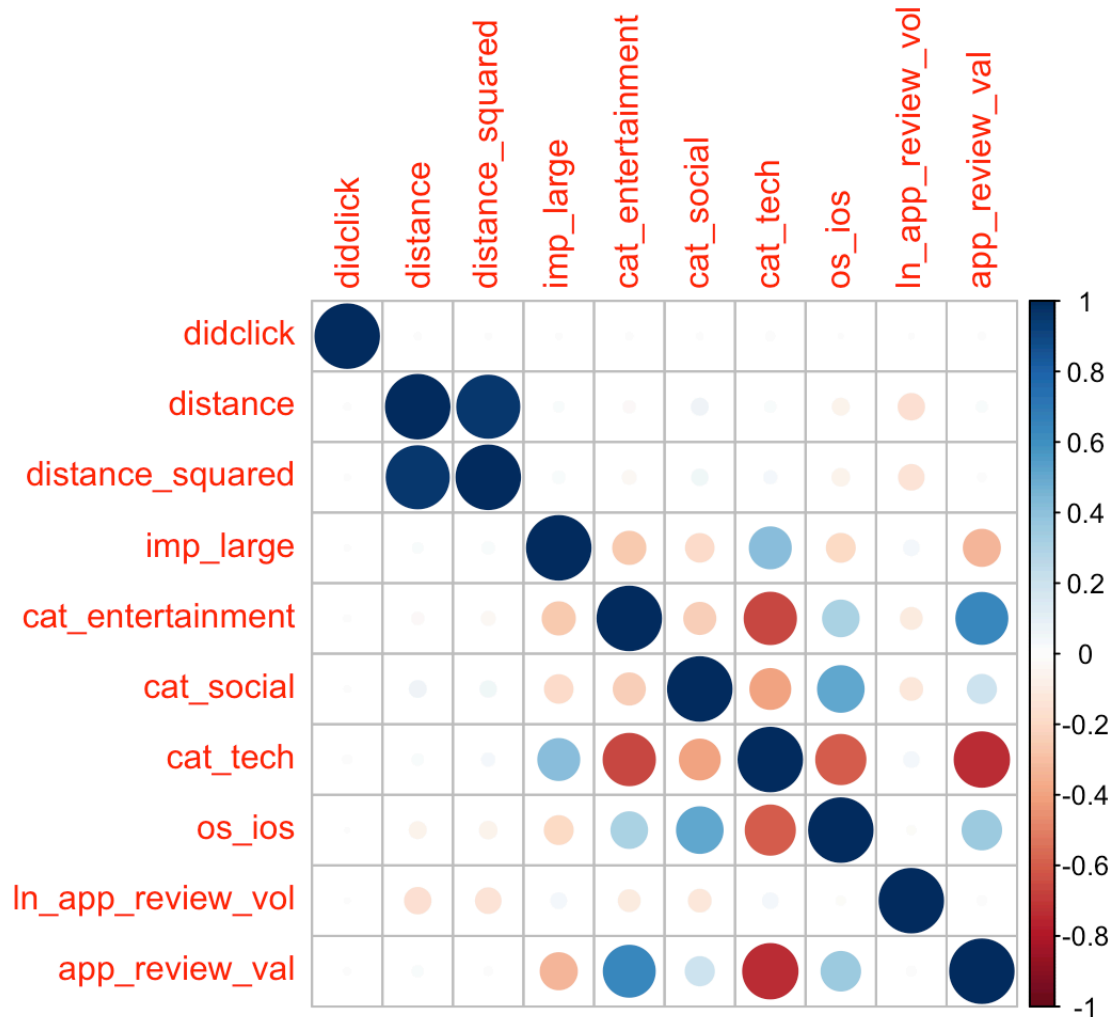
```
##          didclick    distance distance_squared    imp_large
## didclick      1.000000000 -0.006628356      -0.004462031 -0.004786218
## distance     -0.006628356  1.000000000      0.955542600  0.020024918
## distance_squared -0.004462031  0.955542600      1.000000000  0.029075052
## imp_large     -0.004786218  0.020024918      0.029075052  1.000000000
## cat_entertainment -0.007117972 -0.028992663     -0.038385249 -0.254731873
## cat_social     -0.005623417  0.060484490      0.055799705 -0.185311155
## cat_tech       0.012454366  0.023499545      0.034723464  0.414049273
## os_ios        -0.002147325 -0.060281389     -0.065060413 -0.190194050
## ln_app_review_vol 0.003982875 -0.157864184     -0.146892384  0.049929790
## app_review_val -0.006523592  0.022481133      0.010750893 -0.321439020
##          cat_entertainment    cat_social    cat_tech    os_ios
## didclick      -0.007117972 -0.005623417  0.01245437 -0.002147325
## distance     -0.028992663  0.060484490  0.02349954 -0.060281389
## distance_squared -0.038385249  0.055799705  0.03472346 -0.065060413
## imp_large     -0.254731873 -0.185311155  0.41404927 -0.190194050
## cat_entertainment  1.000000000 -0.238133905 -0.65257568  0.312647684
## cat_social     -0.238133905  1.000000000 -0.39192721  0.513672844
## cat_tech       -0.652575678 -0.391927215  1.00000000 -0.598919227
## os_ios         0.312647684  0.513672844 -0.59891923  1.000000000
## ln_app_review_vol -0.105545185 -0.115376574  0.04950383 -0.013523794
## app_review_val  0.642212363  0.194394425 -0.73206714  0.366139311
##          ln_app_review_vol app_review_val
## didclick      0.003982875  -0.006523592
## distance     -0.157864184  0.022481133
## distance_squared -0.146892384  0.010750893
## imp_large     0.049929790  -0.321439020
## cat_entertainment -0.105545185  0.642212363
## cat_social     -0.115376574  0.194394425
## cat_tech       0.049503835  -0.732067145
## os_ios        -0.013523794  0.366139311
## ln_app_review_vol  1.000000000  0.014457854
## app_review_val  0.014457854  1.000000000
```

```
round(res2, 4)
```



```
##          didclick distance distance_squared imp_large
## didclick      1.0000  -0.0066      -0.0045  -0.0048
## distance      -0.0066   1.0000       0.9555   0.0200
## distance_squared -0.0045   0.9555       1.0000   0.0291
## imp_large      -0.0048   0.0200       0.0291   1.0000
## cat_entertainment -0.0071 -0.0290      -0.0384  -0.2547
## cat_social      -0.0056   0.0605       0.0558  -0.1853
## cat_tech        0.0125   0.0235       0.0347   0.4140
## os_ios          -0.0021 -0.0603      -0.0651  -0.1902
## ln_app_review_vol 0.0040 -0.1579      -0.1469   0.0499
## app_review_val  -0.0065   0.0225       0.0108  -0.3214
##          cat_entertainment cat_social cat_tech os_ios
## didclick          -0.0071  -0.0056   0.0125 -0.0021
## distance          -0.0290   0.0605   0.0235 -0.0603
## distance_squared  -0.0384   0.0558   0.0347 -0.0651
## imp_large         -0.2547  -0.1853   0.4140 -0.1902
## cat_entertainment   1.0000  -0.2381  -0.6526  0.3126
## cat_social         -0.2381   1.0000  -0.3919  0.5137
## cat_tech          -0.6526  -0.3919   1.0000 -0.5989
## os_ios            0.3126   0.5137  -0.5989  1.0000
## ln_app_review_vol  -0.1055  -0.1154   0.0495 -0.0135
## app_review_val     0.6422   0.1944  -0.7321  0.3661
##          ln_app_review_vol app_review_val
## didclick          0.0040      -0.0065
## distance         -0.1579       0.0225
## distance_squared  -0.1469       0.0108
## imp_large         0.0499      -0.3214
## cat_entertainment -0.1055       0.6422
## cat_social        -0.1154       0.1944
## cat_tech          0.0495      -0.7321
## os_ios           -0.0135       0.3661
## ln_app_review_vol  1.0000       0.0145
## app_review_val    0.0145       1.0000
```

```
corrplot(res2, method = "circle")
```



- b. Estimate the model in R (using the glm function) and report coefficients and p-value of the estimates.

```
reg1 <- glm(didclick ~ distance + distance_squared + imp_large + cat_entertainment
            + cat_social + cat_tech + os_ios + ln_app_review_vol + app_review_val,
            data = mobile2, family = binomial())

summary(reg1)
```

```
##
## Call:
## glm(formula = didclick ~ distance + distance_squared + imp_large +
##      cat_entertainment + cat_social + cat_tech + os_ios + ln_app_review_vol +
##      app_review_val, family = binomial(), data = mobile2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -0.1510  -0.1272  -0.1148  -0.1042   3.4025
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.616818    0.896958  -7.377 1.62e-13 ***
## distance       -0.117927    0.045073  -2.616 0.008888 **
## distance_squared  0.009166    0.004362   2.102 0.035583 *
## imp_large      -0.352164    0.091782  -3.837 0.000125 ***
## cat_entertainment -0.096143    0.178941  -0.537 0.591069
## cat_social     -0.226695    0.211394  -1.072 0.283550
## cat_tech        0.687657    0.176312   3.900 9.61e-05 ***
## os_ios         0.385895    0.126361   3.054 0.002259 **
## ln_app_review_vol  0.030512    0.063037   0.484 0.628368
## app_review_val   0.323831    0.186656   1.735 0.082757 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 9912.5  on 121566  degrees of freedom
## Residual deviance: 9857.1  on 121557  degrees of freedom
## AIC: 9877.1
##
## Number of Fisher Scoring iterations: 8
```

Some of the variables are insignificant, let's try adding some interaction terms.

```
reg2 <- glm(didclick ~ log(distance) + distance_squared + imp_large + cat_entertainme
nt + app_review_val*cat_entertainment
          + cat_social + cat_social*app_review_val + cat_tech*imp_large + os_ios
          + app_review_val*os_ios + ln_app_review_vol + app_review_val
          + distance*ln_app_review_vol + distance_squared*ln_app_review_vol,
          data = mobile2, family = binomial())
summary(reg2)
```

```
##
## Call:
```

```
## glm(formula = didclick ~ log(distance) + distance_squared + imp_large +
##      cat_entertainment + app_review_val * cat_entertainment +
##      cat_social + cat_social * app_review_val + cat_tech * imp_large +
##      os_ios + app_review_val * os_ios + ln_app_review_vol + app_review_val +
##      distance * ln_app_review_vol + distance_squared * ln_app_review_vol,
##      family = binomial()), data = mobile2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -0.3577  -0.1276  -0.1177  -0.1015   4.3105
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -4.134473    1.682279  -2.458 0.013985
## log(distance)       0.062151    0.128553   0.483 0.628764
## distance_squared  -0.180217    0.092936  -1.939 0.052482
## imp_large        -1.146428    0.392473  -2.921 0.003489
## cat_entertainment -5.047402    2.196867  -2.298 0.021588
## app_review_val    -1.037735    0.314183  -3.303 0.000957
## cat_social         4.686830    2.322538   2.018 0.043594
## cat_tech           0.299243    0.170564   1.754 0.079357
## os_ios            -6.922767    1.921620  -3.603 0.000315
## ln_app_review_vol   0.283008    0.131999   2.144 0.032032
## distance           2.249675    0.877819   2.563 0.010383
## cat_entertainment:app_review_val  1.289929    0.552798   2.333 0.019624
## app_review_val:cat_social -1.272401    0.619175  -2.055 0.039880
## imp_large:cat_tech    0.867968    0.404154   2.148 0.031744
## app_review_val:os_ios   1.925950    0.477233   4.036 5.44e-05
## ln_app_review_vol:distance -0.237950    0.088351  -2.693 0.007076
## distance_squared:ln_app_review_vol  0.018894    0.009358   2.019 0.043481
##
## (Intercept)      *
## log(distance)
## distance_squared  .
## imp_large        **
## cat_entertainment *
## app_review_val    ***
## cat_social        *
## cat_tech          .
## os_ios            ***
## ln_app_review_vol *
## distance          *
## cat_entertainment:app_review_val *
## app_review_val:cat_social *
## imp_large:cat_tech *
## app_review_val:os_ios ***
## ln_app_review_vol:distance **
```

```
## distance_squared:ln_app_review_vol *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 9912.5  on 121566  degrees of freedom
## Residual deviance: 9779.1  on 121550  degrees of freedom
## AIC: 9813.1
##
## Number of Fisher Scoring iterations: 8
```

We can see model 2 has a lower AIC score, which means it's an improved model. We can also calculate the McFadden's Pseudo R-squared for the two models.

```
nullmod1 <- glm(didclick~1, data = mobile2, family="binomial")
1-logLik(reg1)/logLik(nullmod1)
```

```
## 'log Lik.' 0.005590261 (df=10)
```

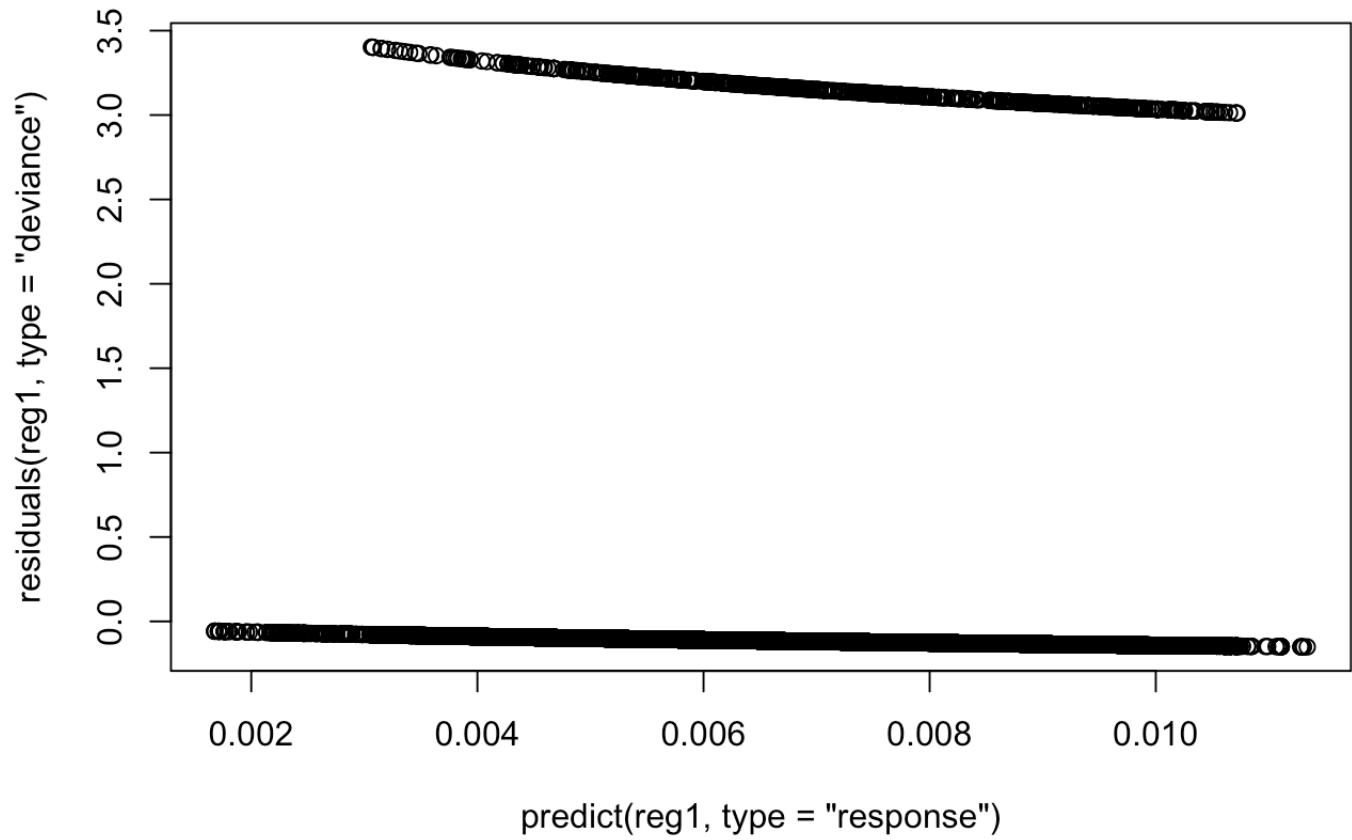
```
nullmod2 <- glm(didclick~1, data = mobile2, family="binomial")
1-logLik(reg2)/logLik(nullmod2)
```

```
## 'log Lik.' 0.01345077 (df=17)
```

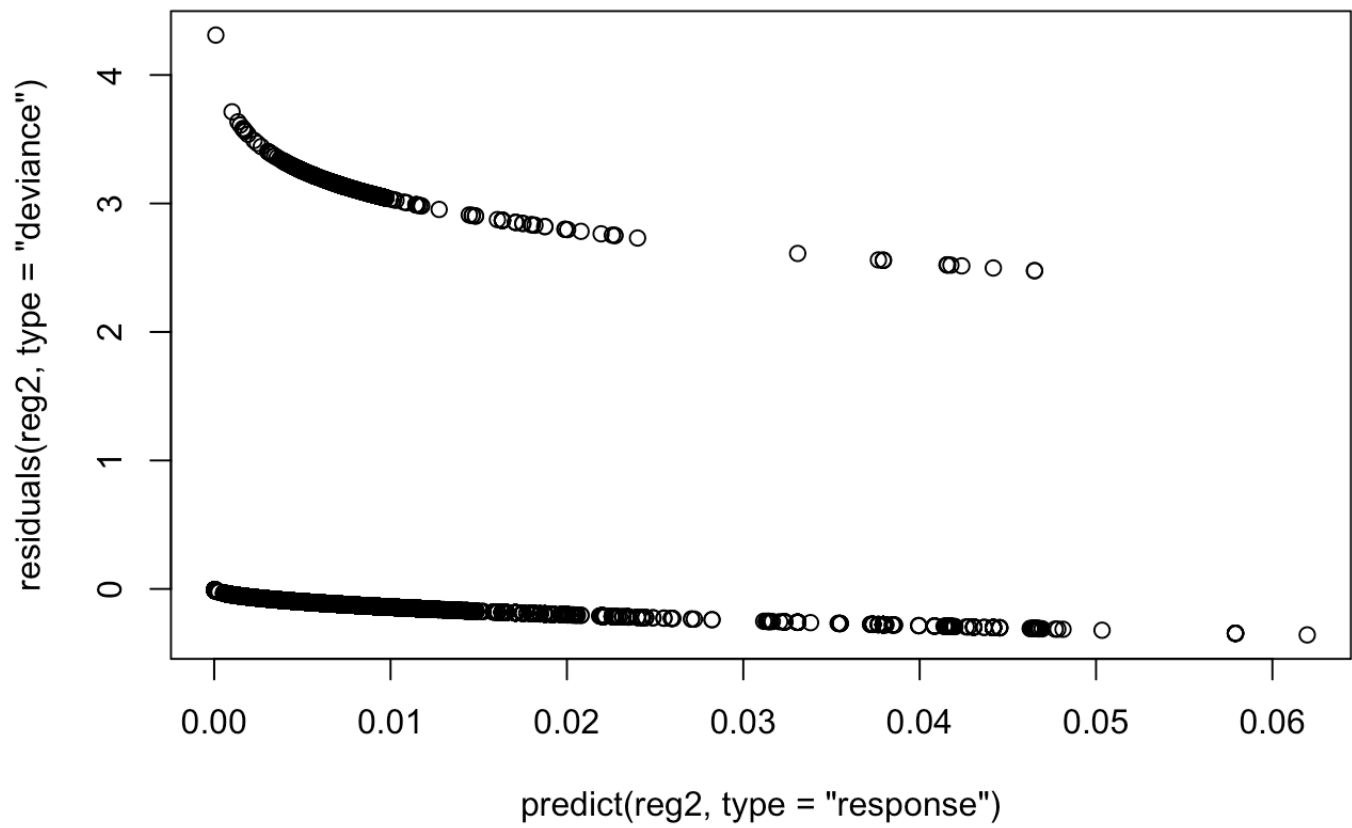
- Diagnostic test for logistic regression

step 1: residual plot

```
plot(predict(reg1, type="response"),
      residuals(reg1, type= "deviance"))
```

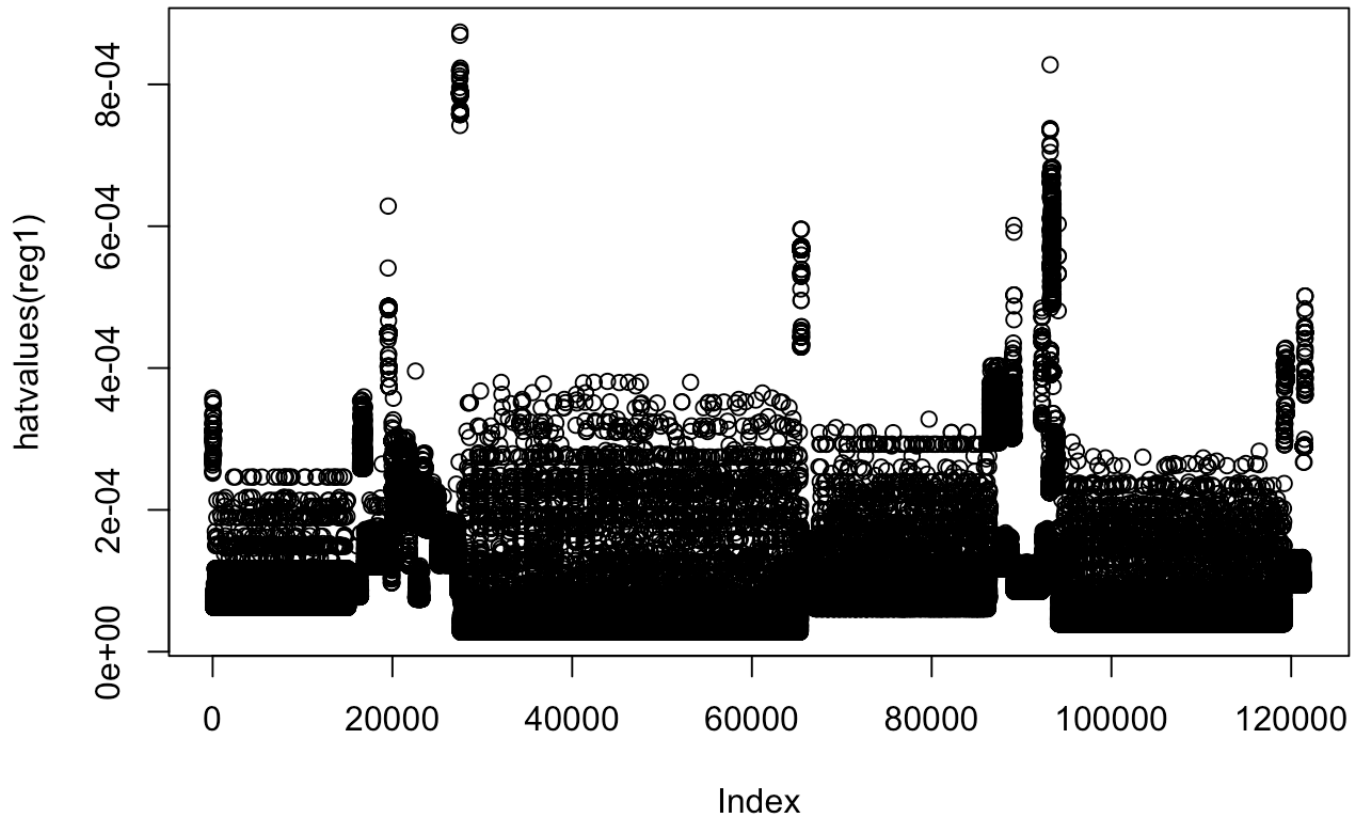


```
plot(predict(reg2, type="response"),  
      residuals(reg2, type= "deviance"))
```



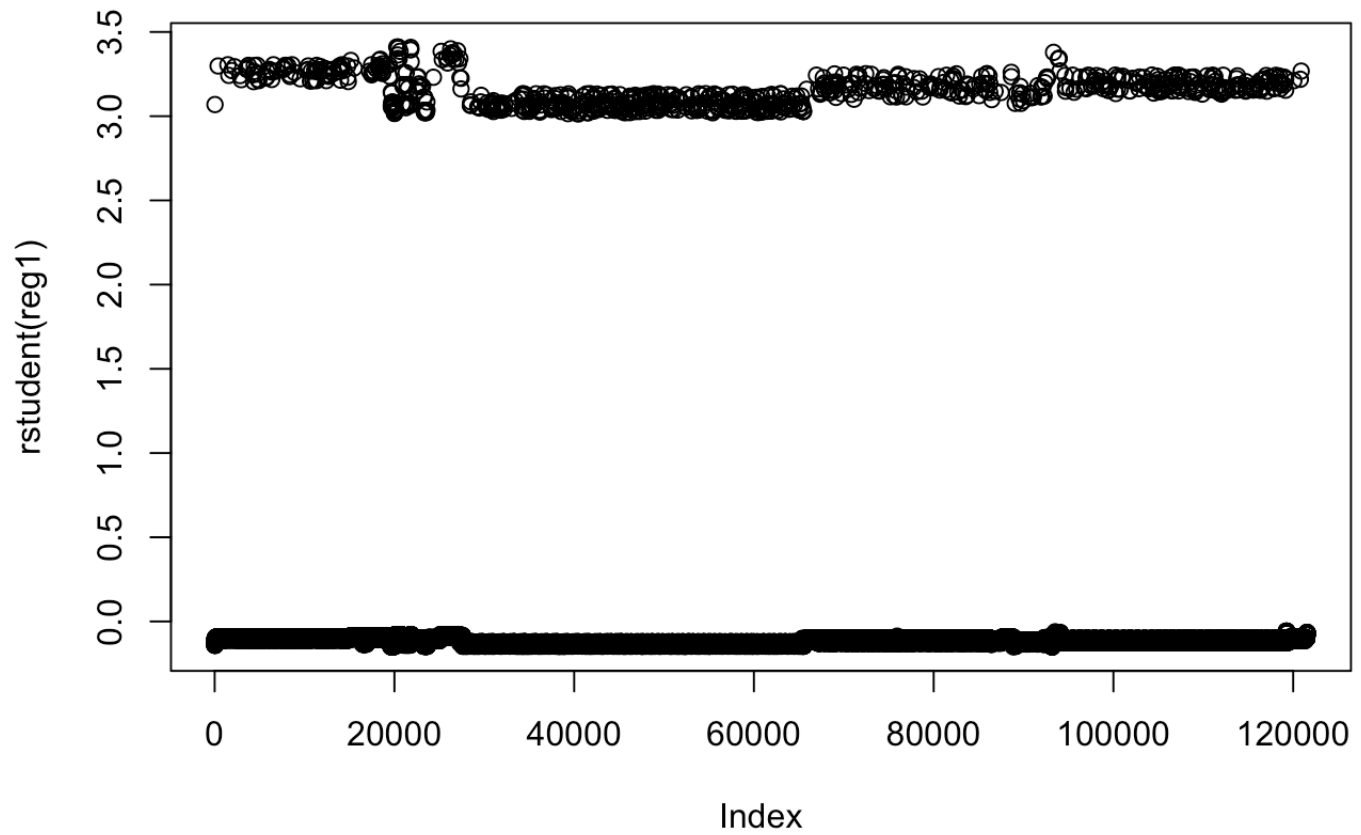
step 2: influential plots

```
plot(hatvalues(reg1))
```

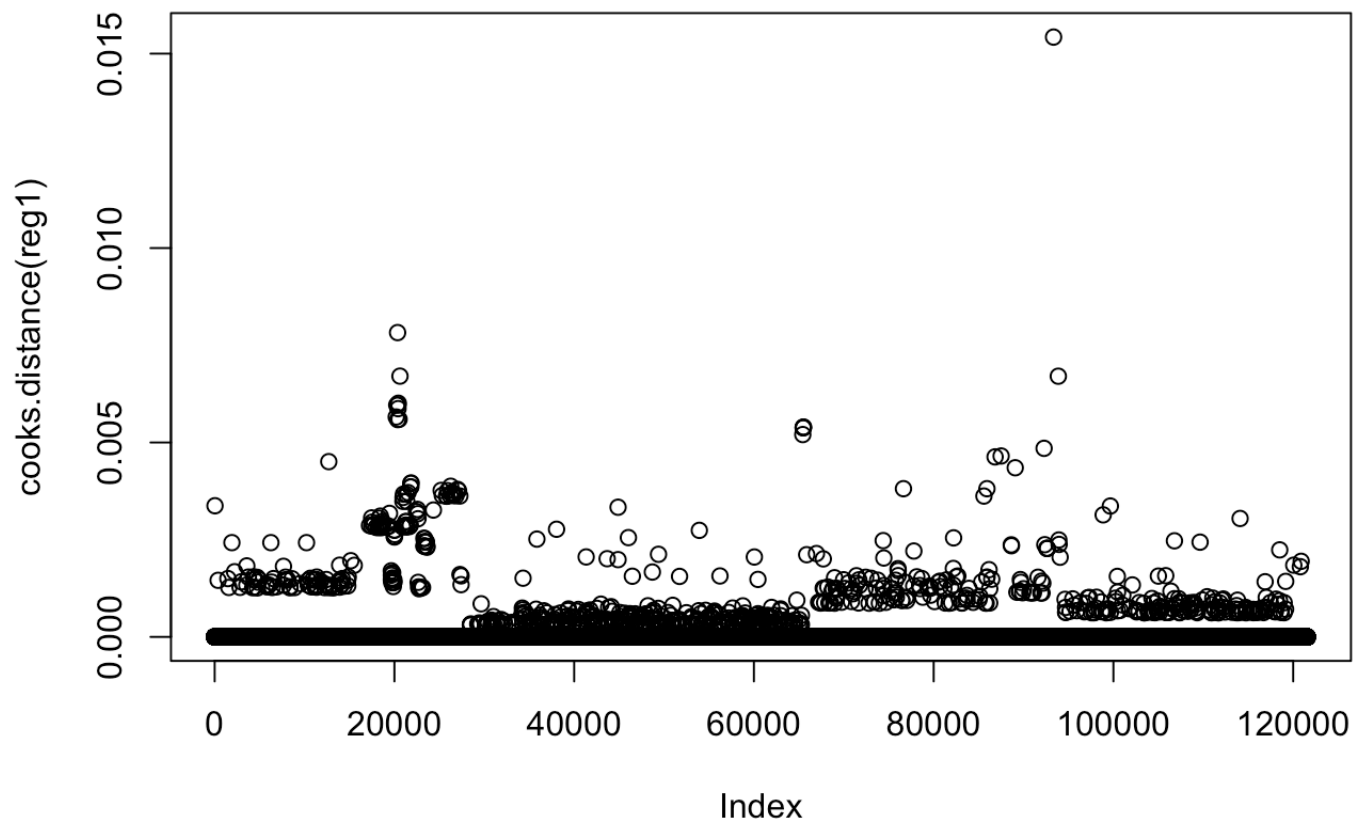


```
plot(rstudent(reg1))
```

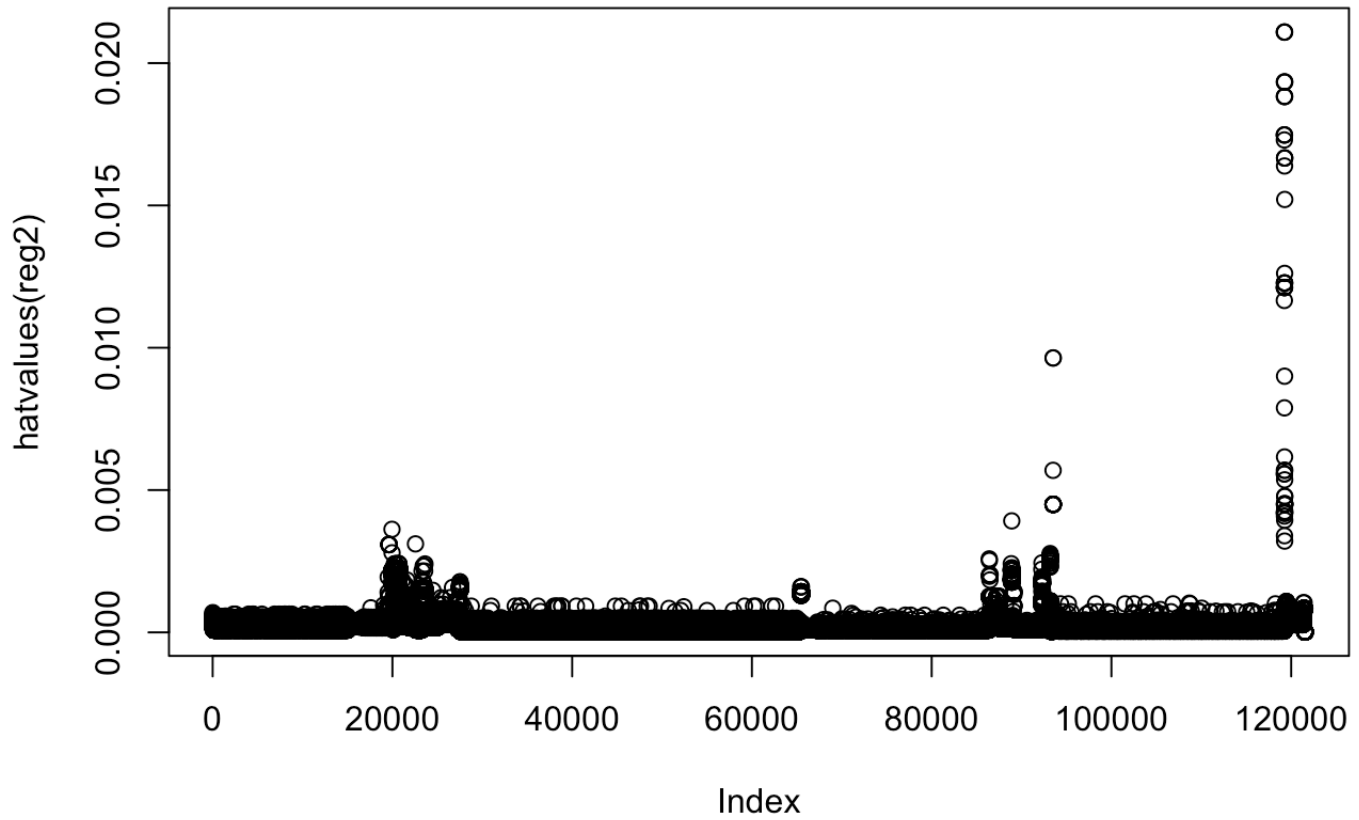




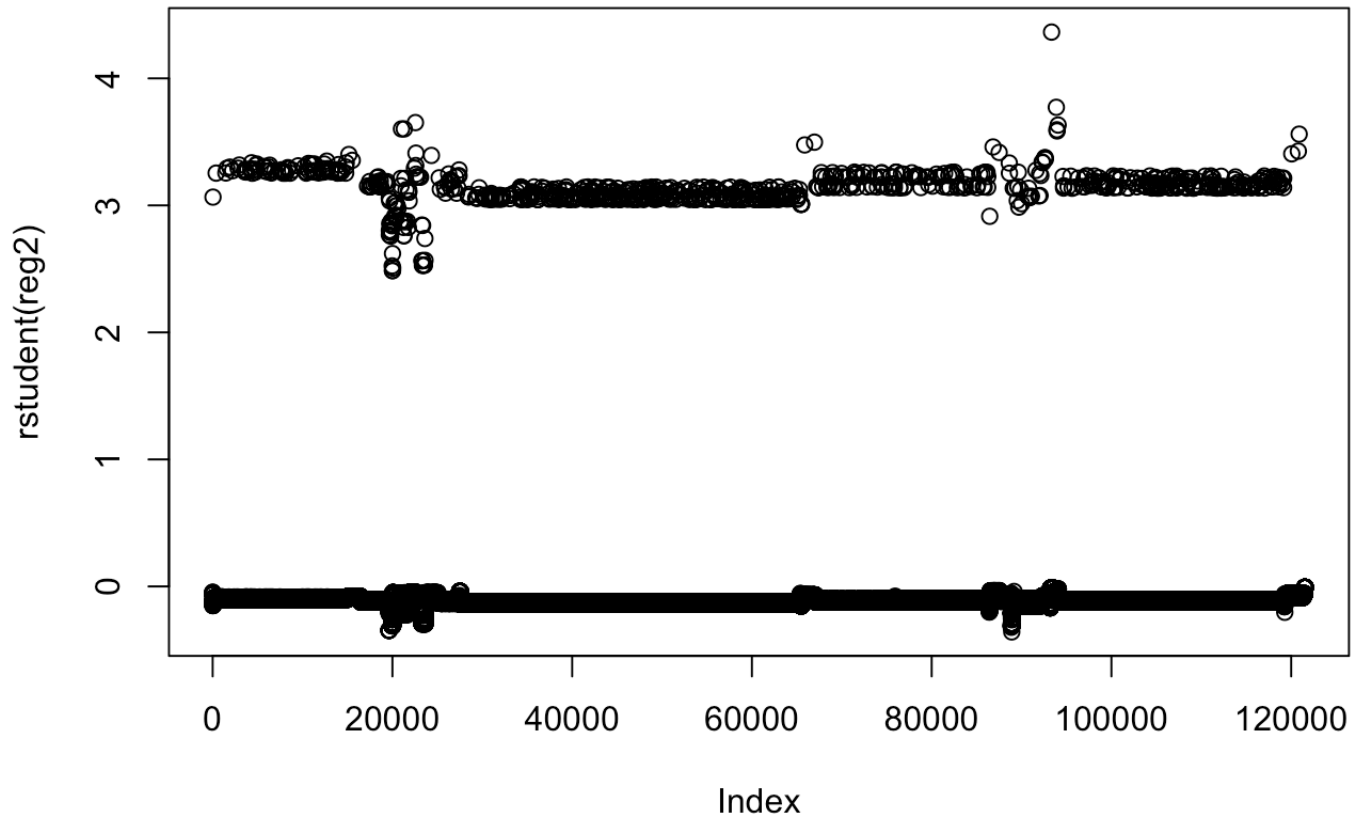
```
plot(cooks.distance(reg1))
```



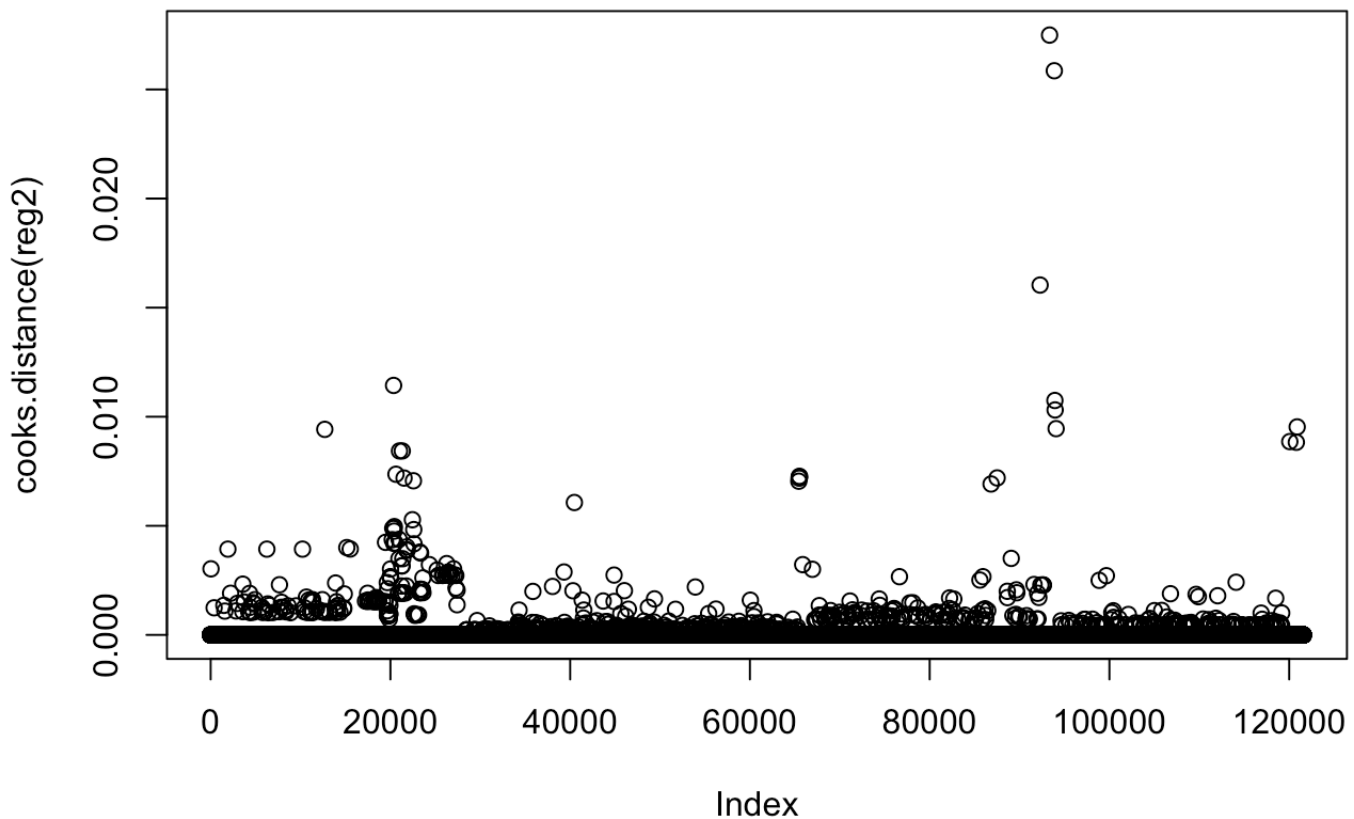
```
plot(hatvalues(reg2))
```



```
plot(rstudent(reg2))
```



```
plot(cooks.distance(reg2))
```



### step 3: overdispersion

The expected variance for data drawn from a binomial distribution is  $\sigma^2 = n\pi(1 - \pi)$ , where  $n$  is the number of observations and  $\pi$  is the probability of belonging to the  $Y = 1$  group. Overdispersion occurs when the observed variance of the response variable is larger than what would be expected from a binomial distribution. Overdispersion can lead to distorted test standard errors and inaccurate tests of significance.

One way to detect overdispersion is to compare the residual deviance with the residual degrees of freedom in your binomial model. If the ratio is considerably larger than 1, you have evidence of overdispersion. Applying this to the mobile example, I have

```
deviance(reg2)/df.residual(reg2)
```

```
## [1] 0.08045368
```

With logistic regression, overdispersion is suggested if the ratio of the residual deviance to the residual degrees of freedom is much larger than 1, which is not our case here.

We will choose the second model as the final model.

## Recall:

```
##
## Call:
## glm(formula = didclick ~ log(distance) + distance_squared + imp_large +
##      cat_entertainment + app_review_val * cat_entertainment +
##      cat_social + cat_social * app_review_val + cat_tech * imp_large +
##      os_ios + app_review_val * os_ios + ln_app_review_vol + app_review_val +
##      distance * ln_app_review_vol + distance_squared * ln_app_review_vol,
##      family = binomial(), data = mobile2)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.3577  -0.1276  -0.1177  -0.1015   4.3105
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.134473    1.682279  -2.458 0.013985
## log(distance)     0.062151    0.128553   0.483 0.628764
## distance_squared -0.180217    0.092936  -1.939 0.052482
## imp_large       -1.146428    0.392473  -2.921 0.003489
## cat_entertainment -5.047402    2.196867  -2.298 0.021588
## app_review_val   -1.037735    0.314183  -3.303 0.000957
## cat_social        4.686830    2.322538   2.018 0.043594
## cat_tech          0.299243    0.170564   1.754 0.079357
## os_ios          -6.922767    1.921620  -3.603 0.000315
## ln_app_review_vol  0.283008    0.131999   2.144 0.032032
## distance         2.249675    0.877819   2.563 0.010383
## cat_entertainment:app_review_val  1.289929    0.552798   2.333 0.019624
## app_review_val:cat_social -1.272401    0.619175  -2.055 0.039880
## imp_large:cat_tech   0.867968    0.404154   2.148 0.031744
## app_review_val:os_ios  1.925950    0.477233   4.036 5.44e-05
## ln_app_review_vol:distance -0.237950    0.088351  -2.693 0.007076
## distance_squared:ln_app_review_vol  0.018894    0.009358   2.019 0.043481
##
## (Intercept)          *
## log(distance)         .
## distance_squared      .
## imp_large             **
## cat_entertainment     *
## app_review_val        ***
## cat_social            *
## cat_tech              .
## os_ios                ***
## ln_app_review_vol     *
## distance              *
## cat_entertainment:app_review_val *
```

```
## app_review_val:cat_social      *
## imp_large:cat_tech            *
## app_review_val:os_ios         ***
## ln_app_review_vol:distance    **
## distance_squared:ln_app_review_vol *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 9912.5  on 121566  degrees of freedom
## Residual deviance: 9779.1  on 121550  degrees of freedom
## AIC: 9813.1
##
## Number of Fisher Scoring iterations: 8
```

By looking at p-value, except for log(distance), all other variables, including the intercept, are significant. app\_review\_val, os\_ios, app\_review\_val:os\_ios, imp\_large, distance:ln\_app\_review\_vol are the most important ones. Let's look at the regression coefficients:

```
coef(reg2)
```

```
##              (Intercept)              log(distance)
##              -4.13447269              0.06215146
##              distance_squared              imp_large
##              -0.18021662              -1.14642790
##              cat_entertainment              app_review_val
##              -5.04740172              -1.03773495
##              cat_social              cat_tech
##              4.68683009              0.29924264
##              os_ios              ln_app_review_vol
##              -6.92276709              0.28300834
##              distance cat_entertainment:app_review_val
##              2.24967541              1.28992918
##              app_review_val:cat_social              imp_large:cat_tech
##              -1.27240063              0.86796831
##              app_review_val:os_ios              ln_app_review_vol:distance
##              1.92595029              -0.23794965
## distance_squared:ln_app_review_vol
##              0.01889372
```

In a logistic regression, the response being modeled is the log(odds) that  $Y = 1$ . The regression coefficients give the change in log(odds) in the response for a unit change in the predictor variable, holding all other predictor variables constant. Because log(odds) are difficult to interpret, we can exponentiate them to put the results on an odds scale:

```
exp(coef(reg2))
```

```
##              (Intercept)              log(distance)
##          1.601111e-02          1.064124e+00
##          distance_squared          imp_large
##          8.350893e-01          3.177699e-01
##          cat_entertainment          app_review_val
##          6.426008e-03          3.542562e-01
##          cat_social              cat_tech
##          1.085087e+02          1.348837e+00
##          os_ios              ln_app_review_vol
##          9.851003e-04          1.327116e+00
##          distance  cat_entertainment:app_review_val
##          9.484657e+00          3.632529e+00
##          app_review_val:cat_social          imp_large:cat_tech
##          2.801583e-01          2.382066e+00
##          app_review_val:os_ios          ln_app_review_vol:distance
##          6.861666e+00          7.882424e-01
## distance_squared:ln_app_review_vol
##          1.019073e+00
```

Now we can see that the odds of an impression click are decreased by a factor of -1.146428 if the `imp_size`="728x90", (holding other variables constant). Conversely, the odds of an impression click are multiplied by a factor of 3.177699e-01 for `imp_size`="728x90".

The odds of an impression click increase with `log(distance)`, `cat_social`, `cat_tech`, `ln_app_review_vol`, `imp_large:cat_tech`, `app_review_val:os_ios`, `distance_squared:ln_app_review_vol`, and decrease with `distance_squared`, `imp_large`, `cat_entertainment`, `app_review_val`, `os_ios`, `app_review_val:cat_social` and `distance:ln_app_review_vol`.

The interaction terms such as `app_review_val:os_ios`, means the relationship between `app_review_val` and whether a user clicked the impression, depending on whether the user is using an ios device or not.

When the predictor variables all equal 0, the intercept in this case means when `imp_size` not equal "728x90", `app_topcat` not equal "IAB1", "IAB1-6", "IAB14", or "IAB19-6", and the user is not using an ios device, the odds of an impression click are decreased by a factor of -4.13447269, or Conversely, the odds of an impression click are multiplied by a factor of 1.601111e-02.

- c. Findings and their implications

Users tend to click on impressions higher in `app_review_val` and choose establishments that are closer to them. Categories of the app is important. Impression size matters. ios users on average are less likely to click. However, they are more likely to click if the app has higher star rating. App review volume matters. For a click decision, distance is not significant.