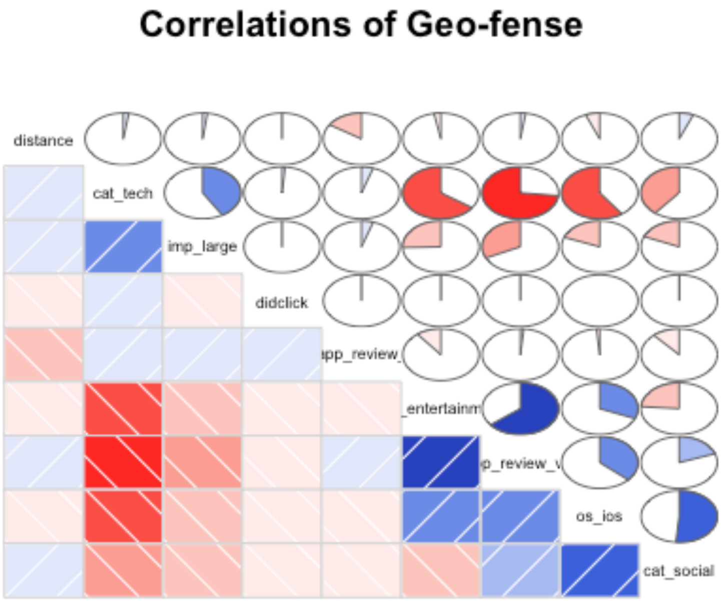
Mobile_Analytics.R

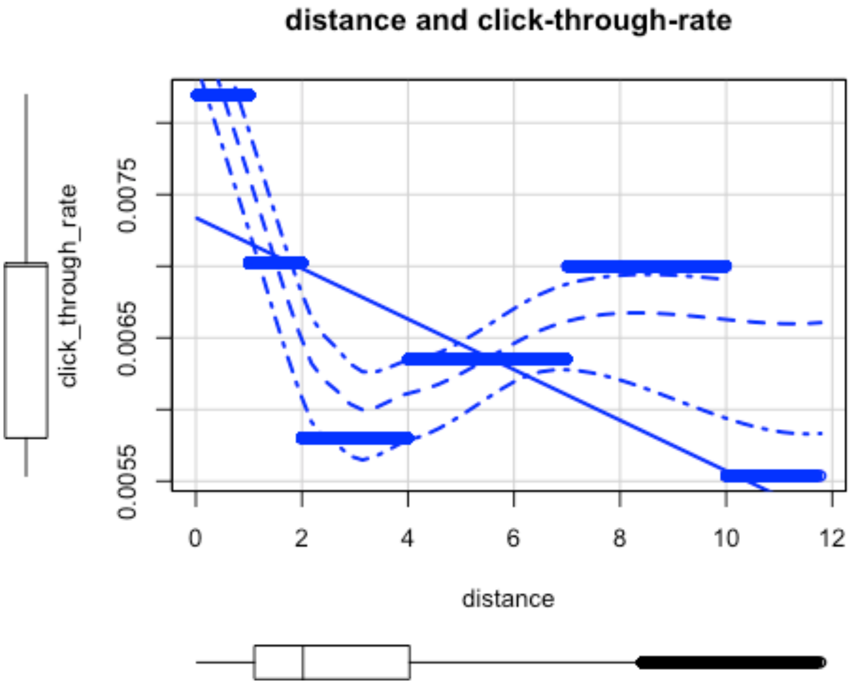
DoryChen
2019-05-10

```
setwd("/Users/DoryChen/Desktop")

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
Geo_fence <- read.csv("Geo-Fence Analytics.csv", header=T)

#Create variable: "imp_large", "cat_entertainment", "cat_social", "cat_tech", "os_ios", "distance"

attach(Geo_fence)

Geo_fence$imp_large <- ifelse(imp_size == "728x90", 1, 0)
Geo_fence$cat_entertainment <-ifelse(app_topcat == "IAB1" | app_topcat == "IAB1-6", 1, 0)
Geo_fence$cat_social <- ifelse(app_topcat == "IAB14", 1, 0)
Geo_fence$cat_tech <- ifelse(app_topcat == "IAB19-6", 1, 0)
Geo_fence$os_ios <- ifelse(device_os == "iOS", 1, 0)

library(aspace)
## Loading required package: splancs
## Loading required package: sp
##
## Spatial Point Pattern Analysis Code in S-Plus
##
##  Version 2 - Spatial and Space-Time analysis
##
## Attaching package: 'splancs'
## The following object is masked from 'package:dplyr':
##
##     tribble
## Loading required package: Hmisc
## Loading required package: lattice
## Loading required package: survival
## Loading required package: Formula
## Loading required package: ggplot2
##
## Attaching package: 'Hmisc'
## The following object is masked from 'package:splancs':
##
##     zoom
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
## The following objects are masked from 'package:base':
##
##     format.pval, units
## Loading required package: shapefiles
## Loading required package: foreign
##
## Attaching package: 'shapefiles'
## The following objects are masked from 'package:foreign':
##
##     read.dbf, write.dbf
Geo_fence$distance <- 6371 * acos(cos(as_radians(device_lat)) * cos(as_radians(geofence_lat))
                    * cos(as_radians(device_lon) - as_radians(geofence_lon))
                    + sin(as_radians(device_lat)) * sin(as_radians(geofence_lat)))

#Create distance group and calculate click-through-rate
attach(Geo_fence)
## The following objects are masked from Geo_fence (pos = 13):
```

```
## The following objects are masked from Geo_fence (pos = 15):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, device_lat, device_lon, device_os, device_zip,
##     didclick, geofence_lat, geofence_lon, gepfence_radius,
##     imp_size
Geo_fence$distance_group <-ifelse(between(distance, 0, 0.5), 1,
                          ifelse(between(distance, 0.5, 1), 2,
                              ifelse(between(distance, 1, 2), 3,
                                  ifelse(between(distance, 2, 4), 4,
                                      ifelse(between(distance, 4, 7), 5,
                                          ifelse(between(distance, 7, 10), 6,
                                              ifelse(distance >10, 7, NA)))))))


Geo_fence <- Geo_fence%>%group_by(distance_group)%>%
 mutate(click_through_rate = mean(didclick))

#Create variables "distance_squared", "ln_app_review_vol"
Geo_fence$ln_app_review_vol <- log(app_review_vol)
Geo_fence$distance_squared <- distance^2

# Descriptive Statisitcs

attach(Geo_fence)
## The following objects are masked from Geo_fence (pos = 3):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, cat_entertainment, cat_social, cat_tech,
##     device_lat, device_lon, device_os, device_zip, didclick,
##     distance, geofence_lat, geofence_lon, gepfence_radius,
##     imp_large, imp_size, os_ios
## The following objects are masked from Geo_fence (pos = 14):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, device_lat, device_lon, device_os, device_zip,
##     didclick, geofence_lat, geofence_lon, gepfence_radius,
##     imp_size
statfun<- function(x){
 data.frame(Mean=mean(x),Median=median(x),STDEV=sd(x),Min=min(x),Max=max(x))
}


sum_table<-rbind(statfun(didclick),
          statfun(distance),
          statfun(imp_large),
          statfun(cat_entertainment),
          statfun(cat_social),
          statfun(cat_tech),
          statfun(os_ios),
          statfun(ln_app_review_vol),
          statfun(app_review_val)
)
row.names(sum_table)<-c("didclick", "distance", "imp_large", "cat_entertainment",
              "cat_social", "cat_tech","os_ios","ln_app_vol","app_val")
print(sum_table)
##                         Mean    Median      STDEV        Min      Max
## didclick           0.006811059  0.000000 0.08224794 0.00000000  1.00000
## distance           2.983737139  2.020864 2.64852620 0.02075894 11.78666
## imp_large          0.230876800  0.000000 0.42139550 0.00000000  1.00000
## cat_entertainment  0.283925736  0.000000 0.45090308 0.00000000  1.00000
## cat_social         0.125124417  0.000000 0.33086130 0.00000000  1.00000
## cat_tech           0.517846126  1.000000 0.49968347 0.00000000  1.00000
## os_ios             0.250363997  0.000000 0.43322443 0.00000000  1.00000
## ln_app_vol        10.056798904 10.087225 0.63696194 7.08086790 12.93770
## app_val            3.654872622  3.400000 0.36081251 1.40000000  4.70000
#Correlations between the variables
library(corrgram)
##
## Attaching package: 'corrgram'
## The following object is masked from 'package:lattice':
##
##     panel.fill
cor_geofence <- subset(Geo_fence, select = c(didclick, distance, imp_large, cat_entertainment,
cat_social, cat_tech, os_ios, ln_app_review_vol, app_review_val))
cor(cor_geofence)
cor(cor_geofence)
##                      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
corrgram(cor_geofence, order = TRUE, lower.panel = panel.shade,
         upper.panel = panel.pie, text.panel = panel.txt,
         main = "Correlations of Geo-fense")
```



```
#Scatterplot of distance and click-through-rate
library(car)
## Loading required package: carData
##
## Attaching package: 'car'
## The following object is masked from 'package:dplyr':
##
##     recode
attach(Geo_fence)
## The following objects are masked from Geo_fence (pos = 6):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, cat_entertainment, cat_social, cat_tech,
##     click_through_rate, device_lat, device_lon, device_os,
##     device_zip, didclick, distance, distance_group,
##     device_zip, didclick, distance, distance_group,
##     distance_squared, geofence_lat, geofence_lon, gepfence_radius,
##     imp_large, imp_size, ln_app_review_vol, os_ios
## The following objects are masked from Geo_fence (pos = 7):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, cat_entertainment, cat_social, cat_tech,
##     device_lat, device_lon, device_os, device_zip, didclick,
##     distance, geofence_lat, geofence_lon, gepfence_radius,
##     imp_large, imp_size, os_ios
## The following objects are masked from Geo_fence (pos = 18):
##
##     app_id, app_name, app_pub, app_review_val, app_review_vol,
##     app_topcat, device_lat, device_lon, device_os, device_zip,
##     didclick, geofence_lat, geofence_lon, gepfence_radius,
##     imp_size
scatterplot(click_through_rate ~ distance, xlab = "distance",
      ylab = "click_through_rate", main = "distance and click-through-rate")
```



```
#We found that the closer mobile device to geofence, the higher the click-through-rate.

#logistic regression of "didclick"
fit.didclick <- glm(didclick ~ distance + imp_large + cat_entertainment
              + cat_social + cat_tech + os_ios + ln_app_review_vol + app_review_val,
              data = Geo_fence, family = binomial())
summary(fit.didclick)
##
## Call:
## glm(formula = didclick ~ distance + imp_large + cat_entertainment +
##     cat_social + cat_tech + os_ios + ln_app_review_vol + app_review_val,
##     family = binomial(), data = Geo_fence)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.1439  -0.1307  -0.1144  -0.1041   3.4177
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -6.76438    0.89749  -7.537 4.81e-14 ***
## distance          -0.02880    0.01425  -2.020 0.043333 *
## imp_large         -0.34794    0.09173  -3.793 0.000149 ***
## cat_entertainment -0.09613    0.17893  -0.537 0.591087
## cat_social        -0.22237    0.21137  -1.052 0.292777
## cat_tech           0.69306    0.17660   3.924 8.70e-05 ***
## os_ios             0.38421    0.12634   3.041 0.002357 **
## ln_app_review_vol  0.03263    0.06301   0.518 0.604590
## app_review_val     0.32515    0.18716   1.737 0.082337 .
## ---
## 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: 9861.4  on 121558  degrees of freedom
## AIC: 9879.4
##
## Number of Fisher Scoring iterations: 8
fit.didclick_distance_sqrt <- glm(didclick ~ distance + distance_squared + imp_large +
cat_entertainment
                                  + cat_social + cat_tech + os_ios + ln_app_review_vol + app_review_val,
                                  data = Geo_fence, family = binomial())
summary(fit.didclick_distance_sqrt)
##
## 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 = Geo_fence)
##
## 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
fit.didclick_reduced <- glm(didclick ~ distance + distance_squared + imp_large
                            + cat_tech + os_ios,
                            data = Geo_fence, family = binomial())
summary(fit.didclick_reduced)
##
## Call:
## glm(formula = didclick ~ distance + distance_squared + imp_large +
##     cat_tech + os_ios, family = binomial(), data = Geo_fence)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.1471  -0.1254  -0.1152  -0.1045   3.3277
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -5.091004   0.110148 -46.220  < 2e-16 ***
## distance         -0.119992   0.045032  -2.665 0.007708 **
## distance_squared  0.009158   0.004364   2.099 0.035858 *
## imp_large        -0.346716   0.091717  -3.780 0.000157 ***
## cat_tech          0.574871   0.100695   5.709 1.14e-08 ***
## os_ios            0.292210   0.111257   2.626 0.008628 **
## ---
## 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: 9862.8  on 121561  degrees of freedom
## AIC: 9874.8
##
## Number of Fisher Scoring iterations: 8
```