Ch13Teetor

Muhammed Khan

Thursday, April 23, 2015

```r
#-----------Beyond Basic Numerics and Statistics------------------#

#importing actual clickthrough data set to perform r commands and operations on

actualdata<-read.csv("C:/Users/AliDesktop/Desktop/Bit Briefcase/Big Data/Kaggle/CTR/train.csv",
                     nrow=1000)

#Checking variables of the clickthrough data set

names(actualdata)
```

```
##  [1] "id"               "click"            "hour"
##  [4] "C1"               "banner_pos"       "site_id"
##  [7] "site_domain"      "site_category"    "app_id"
## [10] "app_domain"       "app_category"     "device_id"
## [13] "device_ip"        "device_model"     "device_type"
## [16] "device_conn_type" "C14"              "C15"
## [19] "C16"              "C17"              "C18"
## [22] "C19"              "C20"              "C21"
```

```r
#Minimizing or Maximizing a Single-Parameter Function
f <- function(x) 3*x^4 - 2*x^3 + 3*x^2 - 4*x + 5
optimize(f, lower=-20, upper=20)  #minimise
```

```
## $minimum
## [1] 0.5972778
##
## $objective
## [1] 3.636756
```

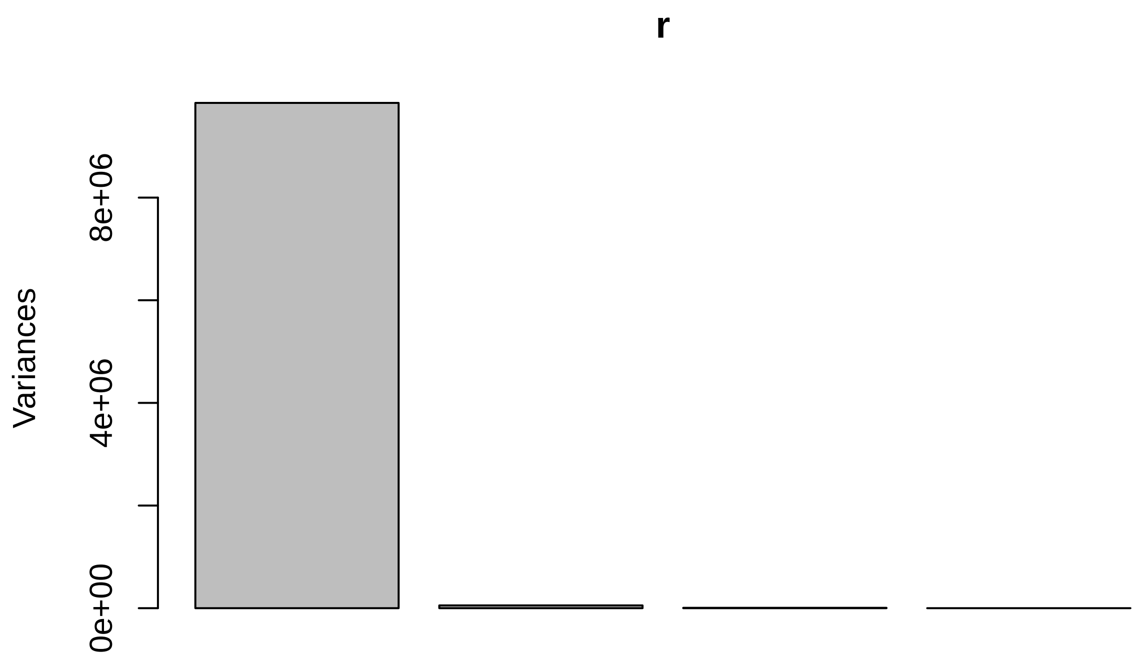
```r
#Performing Principal Component Analysis-using prcomp function
r <- prcomp(~actualdata$C14+actualdata$C17+actualdata$C18+actualdata$C19)
summary(r)  #summary shows the proportion of variation captured by each component
```

```
## Importance of components:
##                              PC1       PC2      PC3   PC4
## Standard deviation     3137.7415 232.61219 82.26517 1.157
## Proportion of Variance    0.9939   0.00546  0.00068 0.000
## Cumulative Proportion     0.9939   0.99932  1.00000 1.000
```

```r
#above summary shows that first component PC1 captures ~99.5% of the variance and other
# components capturing remaining


#The PCA recasts data into a vector space where the 1st dimension captures the most variance
#and the second dimension captures the second most, and so forth.
print(r)
```

```
## Standard deviations:
## [1] 3137.741496  232.612190   82.265173    1.157193
##
## Rotation:
##                          PC1          PC2          PC3           PC4
## actualdata$C14 9.929803e-01  0.006271059  0.118113084 -0.0002597791
## actualdata$C17 1.182074e-01 -0.087520751 -0.989121214  0.0025186994
## actualdata$C18 4.627909e-05  0.001345340 -0.002659899 -0.9999955564
## actualdata$C19 4.134454e-03  0.996142049 -0.087643873  0.0015734716
```

```
#in order to view a bar chart of the variances of the principal components,
# use the below command
plot(r)
```




```
#in order to rotate our actualdata to the principal components, use the below command-
predict(r)
```

```
##              PC1        PC2        PC3        PC4
## 1     -2025.59605  -80.39039   13.895863  0.55014573
## 2     -2027.58201  -80.40293   13.659637  0.55066529
## 3     -2027.58201  -80.40293   13.659637  0.55066529
## 4     -2025.59605  -80.39039   13.895863  0.55014573
## 5      1290.22331  -98.19903  -32.090643  0.80196076
## 6      -797.55799  306.20376  -52.496282  1.30367842
## 7      2669.96157 -100.68295  -40.873256  0.88583331
```

```
## 8      2942.91290 -102.57408  -49.544673 -2.08102705
## 9     -2024.60307  -80.38412   14.013976  0.54988595
## 10     4007.44361   20.88308  -56.590347 -1.84692790
## 11       30.88167  -85.65913    5.350226 -1.33904846
## 12    -2030.56096  -80.42174   13.305298  0.55144463
## 13     2881.97075  -88.14652  157.244630  0.38553480
## 14     2073.25935  550.38810  -62.624469  1.79199025
## 15     3294.20405  409.91659  -49.867187  1.62557873
## 16    -2032.54692  -80.43428   13.069071  0.55196419
## 17      204.86570  -90.65079  -43.174253 -1.20864131
## 18    -2023.61009  -80.37785   14.132089  0.54962617
## 19   -11245.42011  -33.03439   71.519994 -1.96611500
## 20     3548.29217   19.47147  -48.655709 -1.89118164
## 21     2660.03177 -100.74566  -42.054387  0.88843110
## 22    -2025.59605  -80.39039   13.895863  0.55014573
## 23     2673.93350 -100.65786  -40.400804  0.88479420
## 24    -2030.56096  -80.42174   13.305298  0.55144463
## 25    -2025.59605  -80.39039   13.895863  0.55014573
## 26     3982.71172 -110.49561  -44.888706 -2.05594755
## 27     3294.20405  409.91659  -49.867187  1.62557873
## 28     3928.63731  151.29274  -61.370932 -1.66241301
## 29     2673.93350 -100.65786  -40.400804  0.88479420
## 30    -2025.59605  -80.39039   13.895863  0.55014573
## 31    -2023.61009  -80.37785   14.132089  0.54962617
## 32     -882.62901  -87.29083  -10.667500 -2.34091901
## 33    -2023.61009  -80.37785   14.132089  0.54962617
## 34    -2032.54692  -80.43428   13.069071  0.55196419
## 35     2669.96157 -100.68295  -40.873256  0.88583331
## 36     1284.96936  157.04231  -52.276769 -1.80121460
## 37     2881.97075  -88.14652  157.244630  0.38553480
## 38     1290.22331  -98.19903  -32.090643  0.80196076
## 39     1290.22331  -98.19903  -32.090643  0.80196076
## 40    -2025.59605  -80.39039   13.895863  0.55014573
## 41    -2027.58201  -80.40293   13.659637  0.55066529
## 42     2176.76552  547.15803  -94.452607  1.87709489
## 43    -2023.61009  -80.37785   14.132089  0.54962617
## 44     3982.71172 -110.49561  -44.888706 -2.05594755
## 45     3973.74633   29.06810  -10.792028 -1.95929517
## 46    -1517.35387   47.42585  -15.539795 -2.17609326
## 47    -2030.56096  -80.42174   13.305298  0.55144463
## 48     -678.86265  -86.61075  -38.945215 -1.25535325
## 49     -797.55799  306.20376  -52.496282  1.30367842
## 50     2660.03177 -100.74566  -42.054387  0.88843110
## 51     2673.93350 -100.65786  -40.400804  0.88479420
## 52    -2026.58903  -80.39666   13.777750  0.55040551
## 53     1290.22331  -98.19903  -32.090643  0.80196076
## 54    -2030.56096  -80.42174   13.305298  0.55144463
## 55    -2030.56096  -80.42174   13.305298  0.55144463
## 56    -1106.29302  -83.04182  -18.563990 -2.32454967
## 57     2944.89886 -102.56153  -49.308447 -2.08154661
## 58    -2027.58201  -80.40293   13.659637  0.55066529
## 59     1290.22331  -98.19903  -32.090643  0.80196076
## 60     1290.22331  -98.19903  -32.090643  0.80196076
## 61     2881.97075  -88.14652  157.244630  0.38553480
```

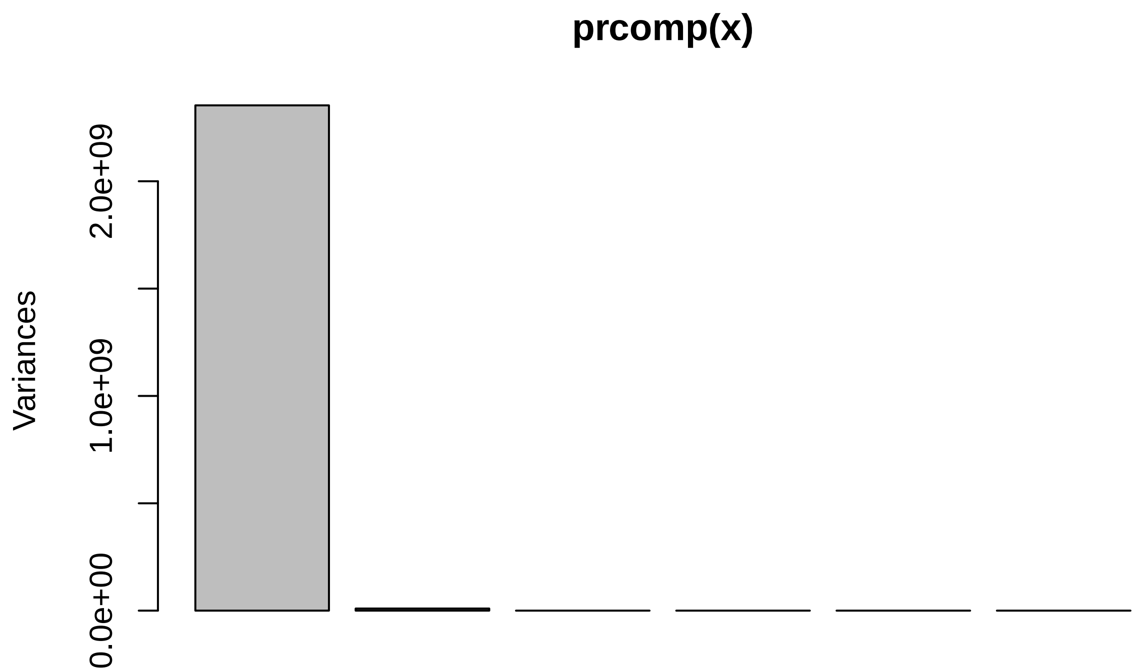
```
## 872    2817.57725   22.18290  -52.308747 -1.91163404
## 873    -460.78493  -79.75959   49.189310 -2.47047803
## 874     204.86570  -90.65079  -43.174253 -1.20864131
## 875    3928.63731  151.29274  -61.370932 -1.66241301
## 876   -2025.59605  -80.39039   13.895863  0.55014573
## 877    -797.55799  306.20376  -52.496282  1.30367842
## 878    2881.97075  -88.14652  157.244630  0.38553480
## 879   -2025.59605  -80.39039   13.895863  0.55014573
## 880   -2028.57499  -80.40920   13.541524  0.55092507
## 881    2073.25935  550.38810  -62.624469  1.79199025
## 882    2944.89886 -102.56153  -49.308447 -2.08154661
## 883    2074.25233  550.39438  -62.506356  1.79173047
## 884   -2023.61009  -80.37785   14.132089  0.54962617
## 885   -2026.58903  -80.39666   13.777750  0.55040551
## 886    1290.22331  -98.19903  -32.090643  0.80196076
## 887    -550.75205   35.86443  -55.705012 -2.04007781
## 888    2944.89886 -102.56153  -49.308447 -2.08154661
## 889     -59.10135  -87.99275  -25.416715 -1.26451501
## 890    3294.20405  409.91659  -49.867187  1.62557873
## 891   -2023.61009  -80.37785   14.132089  0.54962617
## 892   -2025.59605  -80.39039   13.895863  0.55014573
## 893     -59.10135  -87.99275  -25.416715 -1.26451501
## 894    3982.71172 -110.49561  -44.888706 -2.05594755
## 895   -2029.56798  -80.41547   13.423411  0.55118485
## 896    2669.96157 -100.68295  -40.873256  0.88583331
## 897     204.86570  -90.65079  -43.174253 -1.20864131
## 898   -2024.60307  -80.38412   14.013976  0.54988595
## 899    2074.25233  550.39438  -62.506356  1.79173047
## 900    1969.48929  164.93198  -67.211680 -0.71919119
## 901    3928.63731  151.29274  -61.370932 -1.66241301
## 902    2660.03177 -100.74566  -42.054387  0.88843110
## 903   -2029.56798  -80.41547   13.423411  0.55118485
## 904   -2023.61009  -80.37785   14.132089  0.54962617
## 905    2673.93350 -100.65786  -40.400804  0.88479420
## 906    2669.96157 -100.68295  -40.873256  0.88583331
## 907    3928.63731  151.29274  -61.370932 -1.66241301
## 908    1290.22331  -98.19903  -32.090643  0.80196076
## 909   -1517.35387   47.42585  -15.539795 -2.17609326
## 910    2881.97075  -88.14652  157.244630  0.38553480
## 911    2881.97075  -88.14652  157.244630  0.38553480
## 912   -2028.57499  -80.40920   13.541524  0.55092507
## 913   -2025.59605  -80.39039   13.895863  0.55014573
## 914    2198.37847   70.89216  426.439365 -3.14818679
## 915    2881.97075  -88.14652  157.244630  0.38553480
## 916   -2032.54692  -80.43428   13.069071  0.55196419
## 917   -2024.60307  -80.38412   14.013976  0.54988595
## 918   -2024.60307  -80.38412   14.013976  0.54988595
## 919   -2023.61009  -80.37785   14.132089  0.54962617
## 920   -2023.61009  -80.37785   14.132089  0.54962617
## 921    1290.22331  -98.19903  -32.090643  0.80196076
## 922   -2029.56798  -80.41547   13.423411  0.55118485
## 923    2449.30742  -99.68921  -40.041728 -2.12526661
## 924    3294.20405  409.91659  -49.867187  1.62557873
## 925   -2027.58201  -80.40293   13.659637  0.55066529
```

```
## 926    3928.63731  151.29274  -61.370932 -1.66241301
## 927   -1106.29302  -83.04182  -18.563990 -2.32454967
## 928    2669.96157 -100.68295  -40.873256  0.88583331
## 929   -2025.59605  -80.39039   13.895863  0.55014573
## 930   -2029.56798  -80.41547   13.423411  0.55118485
## 931   -2032.54692  -80.43428   13.069071  0.55196419
## 932   -2023.61009  -80.37785   14.132089  0.54962617
## 933    1290.22331  -98.19903  -32.090643  0.80196076
## 934    2881.97075  -88.14652  157.244630  0.38553480
## 935   -2023.61009  -80.37785   14.132089  0.54962617
## 936    1290.22331  -98.19903  -32.090643  0.80196076
## 937   -2030.56096  -80.42174   13.305298  0.55144463
## 938   -2029.56798  -80.41547   13.423411  0.55118485
## 939    2817.57725   22.18290  -52.308747 -1.91163404
## 940   -2024.60307  -80.38412   14.013976  0.54988595
## 941    2669.96157 -100.68295  -40.873256  0.88583331
## 942   -2028.57499  -80.40920   13.541524  0.55092507
## 943   -2032.54692  -80.43428   13.069071  0.55196419
## 944   -1517.35387   47.42585  -15.539795 -2.17609326
## 945   -2026.58903  -80.39666   13.777750  0.55040551
## 946   -2025.59605  -80.39039   13.895863  0.55014573
## 947    2048.75475  287.98135  -79.749693 -1.51663335
## 948    2673.93350 -100.65786  -40.400804  0.88479420
## 949   -2025.59605  -80.39039   13.895863  0.55014573
## 950    2673.93350 -100.65786  -40.400804  0.88479420
## 951     204.86570  -90.65079  -43.174253 -1.20864131
## 952    4007.44361   20.88308  -56.590347 -1.84692790
## 953   -2032.54692  -80.43428   13.069071  0.55196419
## 954    3928.63731  151.29274  -61.370932 -1.66241301
## 955   -2025.59605  -80.39039   13.895863  0.55014573
## 956    1290.22331  -98.19903  -32.090643  0.80196076
## 957   -2030.56096  -80.42174   13.305298  0.55144463
## 958    1290.22331  -98.19903  -32.090643  0.80196076
## 959    2673.93350 -100.65786  -40.400804  0.88479420
## 960    3929.74850  151.21149  -62.241940 -1.66015409
## 961    1312.20371  -94.16132  -30.837172 -1.19493291
## 962    3929.74850  151.21149  -62.241940 -1.66015409
## 963      43.31918  932.76506 -102.481724  0.31848309
## 964   -2028.57499  -80.40920   13.541524  0.55092507
## 965    2561.85665 1612.95198  353.664566  2.13682601
## 966   -2026.58903  -80.39666   13.777750  0.55040551
## 967   -2029.56798  -80.41547   13.423411  0.55118485
## 968    2673.93350 -100.65786  -40.400804  0.88479420
## 969    2881.97075  -88.14652  157.244630  0.38553480
## 970   -2023.61009  -80.37785   14.132089  0.54962617
## 971    2673.93350 -100.65786  -40.400804  0.88479420
## 972    -797.55799  306.20376  -52.496282  1.30367842
## 973      43.31918  932.76506 -102.481724  0.31848309
## 974   -2026.58903  -80.39666   13.777750  0.55040551
## 975    3928.63731  151.29274  -61.370932 -1.66241301
## 976    -473.67049  -88.75596  -53.664759 -2.20959491
## 977   -2032.54692  -80.43428   13.069071  0.55196419
## 978    2881.97075  -88.14652  157.244630  0.38553480
## 979   -2027.58201  -80.40293   13.659637  0.55066529
```

```
## 980   -2025.59605  -80.39039   13.895863  0.55014573
## 981    -473.67049  -88.75596  -53.664759 -2.20959491
## 982     165.34514  -82.31224  -41.560626 -2.20354860
## 983   -2026.58903  -80.39666   13.777750  0.55040551
## 984    2669.96157 -100.68295  -40.873256  0.88583331
## 985   -2027.58201  -80.40293   13.659637  0.55066529
## 986    2670.95455 -100.67668  -40.755143  0.88557353
## 987    3280.51078  670.72422  -75.590699  2.04424368
## 988    2669.96157 -100.68295  -40.873256  0.88583331
## 989   -2027.58201  -80.40293   13.659637  0.55066529
## 990    2669.96157 -100.68295  -40.873256  0.88583331
## 991   -2026.58903  -80.39666   13.777750  0.55040551
## 992    2881.97075  -88.14652  157.244630  0.38553480
## 993   -2025.59605  -80.39039   13.895863  0.55014573
## 994  -11412.06571  -23.99814   75.066434 -2.97110801
## 995    3928.63731  151.29274  -61.370932 -1.66241301
## 996   -2023.61009  -80.37785   14.132089  0.54962617
## 997    2881.97075  -88.14652  157.244630  0.38553480
## 998    1290.22331  -98.19903  -32.090643  0.80196076
## 999   -2028.57499  -80.40920   13.541524  0.55092507
## 1000  -2030.56096  -80.42174   13.305298  0.55144463
```

```r
#-------------Performing Simple Orthogonal Regression-Also called as total least squares---------
#To create a linear model using orthogonal regression in which variances of C18 and C19
# are treated symmetricallyin order to implement a basic orthogonal regression in R,
# we perform PCA

r <- prcomp( ~ actualdata$C18 + actualdata$C19 )
#Now, using the rotations to compute the slope:
slope <- r$rotation[2,1] / r$rotation[1,1]
#Now, calculatng the intercept from the slope:
intercept <- r$center[2] - slope*r$center[1]


#------Finding Clusters in the Data---------------

#creating a subset of the actual clickthrough data set to include only numerical variables
# to understand clustering

#d<-dist(x)      #Compute distances between observations
#hc <- hclust(d)  #Form hierarchical clusters

#the result clust below is the vector of numbers between 1 and 3, one for each observation in x
#Each number classifies its corresponding observation into one of the n clusters.
#clust <- cutree(hc, k=3)    #Organize them into the 3 largest clusters

#-----------Predicting a Binary-Valued Variable (Logistic Regression)---------
#A regression model to predict the probability of a binary event occuring

# install.packages("faraway")
# library(faraway)
#
# #Faraway gives an example of predicting a binary-valued variable:
# #test from the dataset pima is true if the patient tested positive for diabetes.
```

```r
#
# data(pima, package="faraway")
# b <- factor(pima$test)
#
# #The predictors are diastolic blood pressure and body mass index (BMI).
# m <- glm(b ~ diastolic + bmi, family=binomial, data=pima)
#
# summary(m)   #results show that only the bmi variable is significant, p-value for it is
# # 1.95e-14
#
# #Since only bmi variable is significant, a reduced model can be created like below:
# m.red <- glm(b ~ bmi, family=binomial, data=pima)
#
#
# #Now using the model to calculate the probability that someone with an avg BMI(32.0)
# # will test positive for diabetes
# newdata <- data.frame(bmi=32.0)
#
# predict(m.red, type="response", newdata=newdata)
# #According to this model, the probability is about 33.3%

#-------Factor Analysis------

#in order to discover what the variables in a dataset have in common, we use the factanal
# function

#creating a subset of the actual clickthrough data set to include only numerical variables
# since factor analysis is for numerical ones

x<-data.frame(actualdata$C14, actualdata$C18, actualdata$C19, actualdata$C17, actualdata$C20,
              actualdata$C21)

#Plotting the PCA to see the variance captured by the components
plot(prcomp(x))
```




```r
factanal(x,factors=2) #The p-value is 9.4e-12. Small p-value (<0.05) indicates that the two
```

```
##
## Call:
## factanal(x = x, factors = 2)
##
## Uniquenesses:
## actualdata.C14 actualdata.C18 actualdata.C19 actualdata.C17 actualdata.C20
##          0.005          0.464          0.858          0.045          0.914
## actualdata.C21
##          0.220
##
## Loadings:
##                Factor1 Factor2
## actualdata.C14  0.997
## actualdata.C18  0.135   0.719
## actualdata.C19          0.372
## actualdata.C17  0.977
## actualdata.C20          0.290
## actualdata.C21  0.377  -0.799
##
##                Factor1 Factor2
## SS loadings      2.115   1.378
## Proportion Var   0.353   0.230
## Cumulative Var   0.353   0.582
```

```
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 95.99 on 4 degrees of freedom.
## The p-value is 7.03e-20
```

```
# factors are insufficient

#In cases where p-value>0.05, it will help us to conclude that factors are sufficient
#and % of individual variance and cumulative variance they explain
```