# Project-4

Chitra Karki[*] University of Texas at El Paso (UTEP)

October 25, 2022

## Contents

---

## 1 PageRank

Based on the links in Figure 1,

(a) Obtain the link matrix L and input it into R.

(b) Reproduce the graph similar to Figure 1 to check if you have got the right link matrix L.

(c) Compute the PageRank score for each webpage. Provide a barplot of the PageRank score. Which pages come to the top-3 list? Discuss the results.

```
# setting working directory
setwd("C:/Users/chitr/OneDrive - University of Texas at El Paso/data_science/semesters/sem3-fal
```

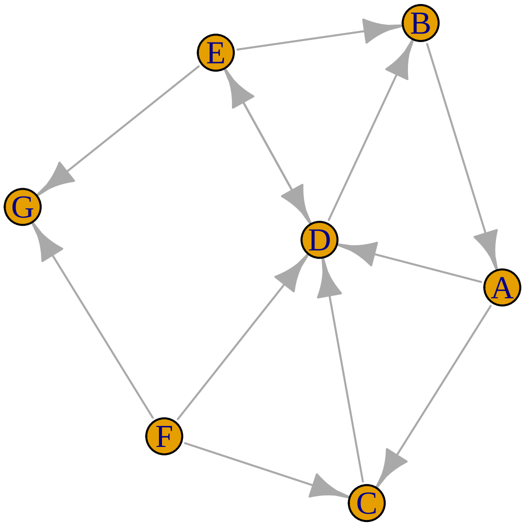
```
# (a)
L <- matrix(c(0, 1, 0, 0, 0, 0, 0,
              0, 0, 0, 1, 1, 0, 0,
              1, 0, 0, 0, 0, 1, 0,
              1, 0, 1, 0, 1, 1, 0,
              0, 0, 0, 1, 0, 0, 0,
              0, 0, 0, 0, 0, 0, 0,
              0, 0, 0, 0, 1, 1, 0),
            ncol=7,byrow=T)
```

[*] cbkarki@miners.utep.edu

```
colnames(L) = LETTERS[1:7]
rownames(L) = colnames(L)

library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

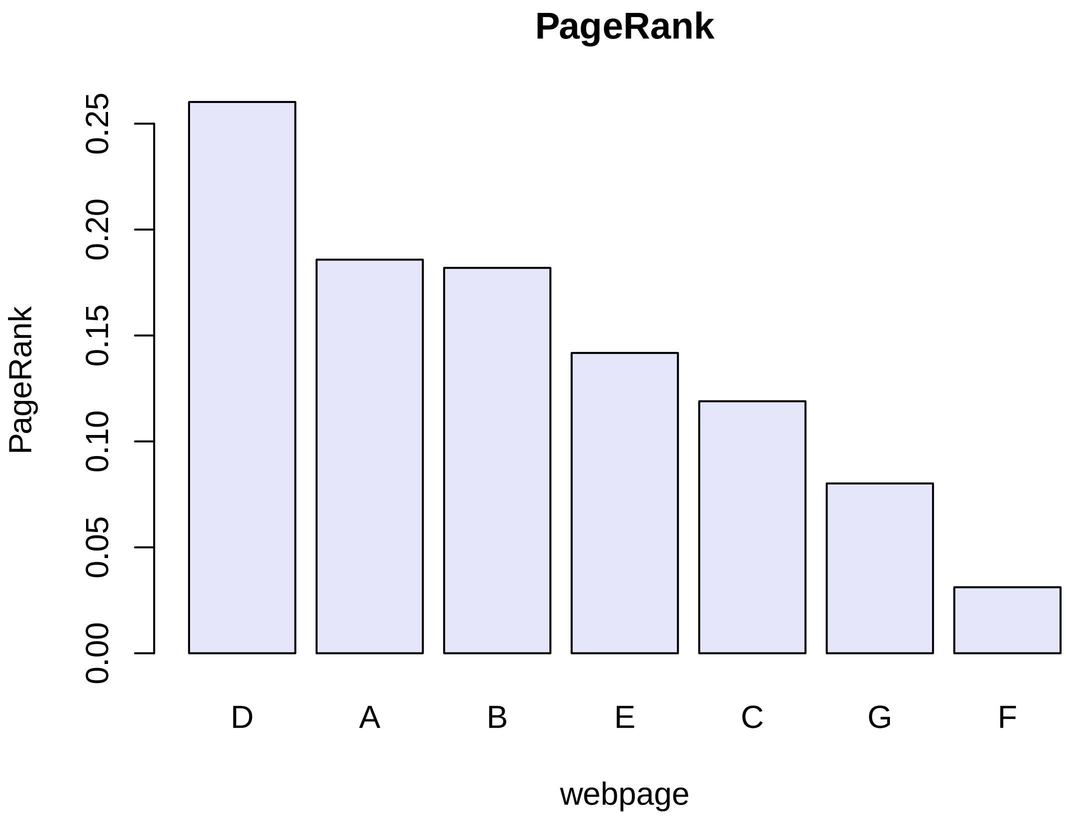
```
graph <- graph_from_adjacency_matrix(t(L))
par(mfrow=c(1,1), mar=rep(4,4))
plot(graph)
```



```
# PAGERANK
rank0 <- page.rank(graph)$vector;rank0
```

```
##          A          B          C          D          E          F          G
## 0.18580040 0.18192763 0.11895630 0.26023499 0.14176179 0.03116192 0.08015697
```

```
names(rank0) <- colnames(L)
par(mfrow=c(1,1), mar=rep(4,4))
barplot(sort(rank0,decreasing = T), col="lavender", xlab="webpage",
        ylab="PageRank", main="PageRank")
```



The graph was regenerated. The top three high ranked pages are “D”, “A”, and “B”. According to the algorithm, they will get bigger weights as their inward links are more than the out going links.

## 2 Anomaly Detection

We consider the HTP (high tech part) data available from R Package ICSOutlier. This data set contains the results of p = 88 numerical tests for n = 902 high-tech parts. Based on these results the producer considered all parts functional and all of them were sold. However two parts, 581 and 619, showed defects in use and were returned to the manufacturer. These two observations can thus be considered as outliers and the objective is to detect them by re-examining the test data.

(a) Bring in the data with the following R code
install.packages("ICSOutlier")
library("ICSOutlier")
data(HTP)
dat <- HTP; dim(dat); head(dat)
outliers.true <- c(581, 619)

(b) First obtain robust estimates of the mean vector $\hat{\mu}$ and the VCOV matrix $\hat{\sum}$ of the data with MCD with a breakdown point of your choice. Then compute the robust Mahalanobis distance of each observation with repect to the MCD estimates ($\hat{\mu}$;$\hat{\sum}$) and plot them. You may add a threshold based on the $\chi^2(p)$ distribution and highlight the two defective parts. Are the two defective parts in your top list of potential outliers?

(c) Apply isolation forest (iForest), local outlier factor (LOF), and, optionally, one-class SVM for the same task. Choose the involved parameters appropriately based on your own judgment and you may compare results by varying the parameters. Plot the results.Comment on the similarities and differences of their results. In particular, pay attention to whether the two defective parts are deemed anomalies by each method.

```r
# (a)
#install.packages("ICSOutlier")
library("ICSOutlier")
```

```
## Loading required package: ICS

## Loading required package: mvtnorm

## Loading required package: moments
```

```r
data(HTP)
dat <- HTP; dim(dat); # head(dat)
```

```
## [1] 902  88
```

```r
#outliers.true <- c(581, 619)
```

```r
# (b)
library(robustbase)

# Obtain MCD estimates with a breakdown point of 25%
fit.robust <- covMcd(dat, cor = FALSE, alpha = 0.75)

# Robust (Squared) Mahalanobis distance with MCD/MVE results
RD <- mahalanobis(dat, fit.robust$center, fit.robust$cov)
```

```
# Cut-off based on the chi-square distribution
cutoff.chi.sq <- qchisq(0.975, df = ncol(dat)); cutoff.chi.sq
```

```
## [1] 115.8414
```

```
which(RD >= cutoff.chi.sq)
```

```
##   [1]   2  10  15  22  24  32  37  38  39  41  45  51  55  61  64  65  66  67
##  [19]  69  77  82  85  86  91  96  98 103 108 112 113 114 117 120 123 127 135
##  [37] 138 140 141 149 155 156 160 163 164 165 167 169 170 171 181 184 185 191
##  [55] 192 201 205 210 212 214 216 219 221 223 224 226 229 230 231 232 234 238
##  [73] 256 257 263 271 278 280 281 284 288 289 290 294 299 302 303 305 307 308
##  [91] 310 320 321 324 328 329 332 339 345 348 350 351 352 354 358 360 365 369
## [109] 372 379 384 386 387 390 393 398 399 400 405 412 416 417 418 419 424 428
## [127] 430 432 433 436 437 438 441 452 453 454 456 457 460 463 468 472 473 474
## [145] 476 477 486 487 490 492 500 501 502 506 516 517 520 523 524 525 526 527
## [163] 528 532 538 539 540 544 548 565 566 578 579 581 582 585 596 601 607 610
## [181] 611 615 618 619 628 629 632 637 638 642 644 649 655 658 664 665 670 674
## [199] 680 681 688 692 695 696 702 703 707 708 709 712 717 721 725 736 740 743
## [217] 744 753 760 761 764 766 771 772 773 778 779 782 783 801 805 807 808 815
## [235] 826 829 833 834 836 838 839 845 852 860 861 862 864 865 871 872 873 874
## [253] 876 878 881 886 888 889
```

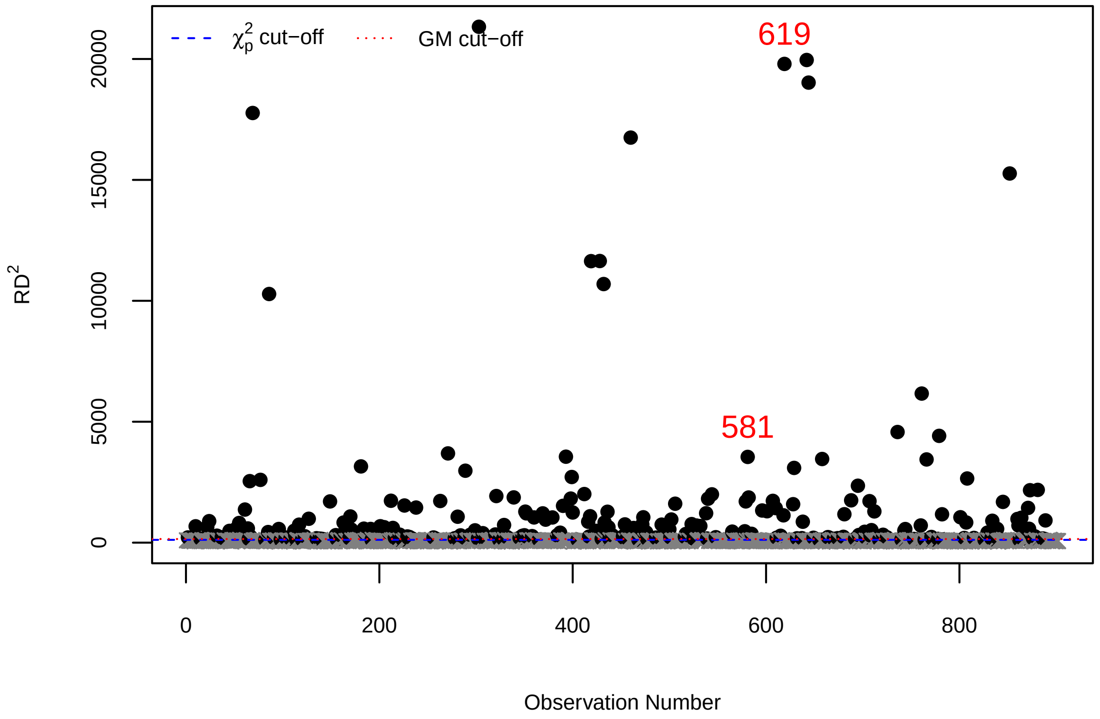
```
# Another Cut-off Suggested by Green and Martin (2017)
# install.packages("CerioliOutlierDetection")
n <- nrow(dat); p <- ncol(dat)
library("CerioliOutlierDetection")
cutoff.GM <- hr05CutoffMvnormal(n.obs = n, p.dim=p, mcd.alpha = 0.75,
    signif.alpha = 0.025, method = "GM14",
    use.consistency.correction = TRUE)$cutoff.asy
cutoff.GM
```

```
## [1] 149.9075
```

```
which(RD >= cutoff.GM)  # OUTLIER IDs
```

```
##   [1]   2  10  15  22  24  32  37  38  39  45  51  55  61  64  65  66  67  69
##  [19]  77  85  86  91  96 103 108 112 113 117 123 127 135 138 140 141 149 155
##  [37] 160 163 164 165 167 169 170 171 181 184 185 191 201 205 210 212 214 216
##  [55] 221 223 226 229 230 231 232 234 238 256 257 263 271 278 280 281 284 288
##  [73] 289 290 294 299 302 303 307 308 310 320 321 328 329 332 339 345 348 350
##  [91] 351 352 354 358 360 369 372 379 384 386 387 390 393 398 399 400 412 416
## [109] 417 418 419 424 428 430 432 433 436 437 438 441 452 453 454 456 457 460
## [127] 463 472 473 474 476 477 486 487 490 492 500 501 502 506 516 517 520 523
## [145] 524 525 526 527 528 532 538 539 540 544 548 565 566 578 579 581 582 585
## [163] 596 601 607 610 611 615 618 619 628 629 632 637 638 642 644 649 658 664
## [181] 665 670 674 680 681 688 692 695 696 702 703 707 708 709 712 721 725 736
## [199] 743 744 753 760 761 766 771 772 778 779 782 801 805 807 808 815 829 833
## [217] 834 838 839 845 852 860 861 864 865 871 872 873 874 876 878 881 886 888
## [235] 889
```

```
# PLOT THE RESULTS
par(mfrow=c(1,1), mar=rep(4,4))
colPoints <- ifelse(RD >= min(c(cutoff.chi.sq, cutoff.GM)), 1, grey(0.5))
pchPoints <- ifelse(RD >= min(c(cutoff.chi.sq, cutoff.GM)), 16, 4)
plot(seq_along(RD), RD, pch = pchPoints, col = colPoints,
    ylim=c(0, max(RD, cutoff.chi.sq, cutoff.GM) + 2), cex.axis = 0.7, cex.lab = 0.7,
    ylab = expression(RD**2), xlab = "Observation Number")
abline(h = c(cutoff.chi.sq, cutoff.GM), lty = c("dashed", "dotted"), col=c("blue", "red"))
legend("topleft", lty = c("dashed", "dotted"), cex = 0.7, ncol = 2, bty = "n",
    legend = c(expression(paste(chi[p]**2, " cut-off")), "GM cut-off"), col=c("blue", "red"))
text(x=c(581,619),y=RD[c(581,619)],labels=c(581,619),pos=3,col = "red")
```



```
# INSPECT THE MOST OUTLYING OBS
# RD distance of the outlileres mentioned in the question
RD[c(581,619)]
```

```
## [1]  3545.033 19796.057
```

```
id.most <- which(RD >= 3500); id.most # to check 581,619
```

```
##  [1]  69  86 271 303 393 419 428 432 460 581 619 642 644 736 761 779 852
```

```
length(id.most)
```

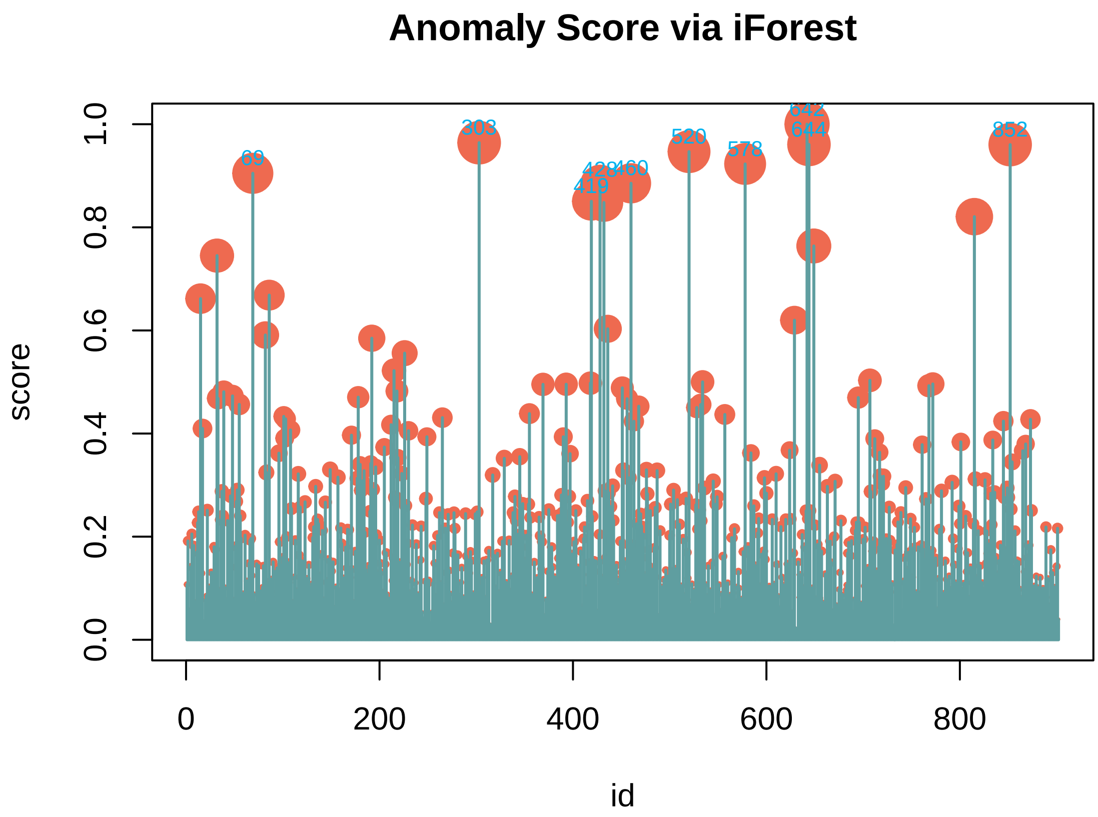
```
## [1] 17
```

The outliers mentioned in the question are in the top list. For which their robust Mahalonobis distances were calculated and found to be 3592.471, 20049.102 respectively for observations 581 and 619. These observations are also pointed in the plot. There are 18 observations for which the Mahlonobis distance is greater then or equal to 3500. It is observed that observation 619 is one of the most outlying but 519 is not.

```
# (c)
# isolation Forest

library(isofor)
# help(package="isofor")

fit.isoforest <- iForest(dat, nt=100, phi=256)
pred <- predict(fit.isoforest, newdata=dat)
#pred # Higher scores correspond to more isolated observations.

# PLOT OF THE SCORES
score <- scale(pred, center = min(pred), scale = max(pred)-min(pred))
par(mfrow=c(1,1), mar=rep(4,4))
plot(x=1:length(score), score, type="p", pch=19,
     main="Anomaly Score via iForest",
     xlab="id", ylab="score", cex=score*3, col="coral2")
add.seg <- function(x) segments(x0=x[1], y0=0, x1=x[1], y1=x[2],
                                 lty=1, lwd=1.5, col="cadetblue")
apply(data.frame(id=1:length(score), score=score), 1, FUN=add.seg)
```

```
## NULL
```

```
eps <- 0.99
id.outliers <- which(score > quantile(score, eps))
text(id.outliers, score[id.outliers]+0.03, label=id.outliers,
     col="deepskyblue2", cex=0.7)
```



The outliers mentioned in the question are not distinctly visible in the plot from using isolation forest method.

```
# LOF
# install.packages("Rlof")
library(Rlof)
```

```
## Loading required package: doParallel

## Loading required package: foreach

## Loading required package: iterators

## Loading required package: parallel
```

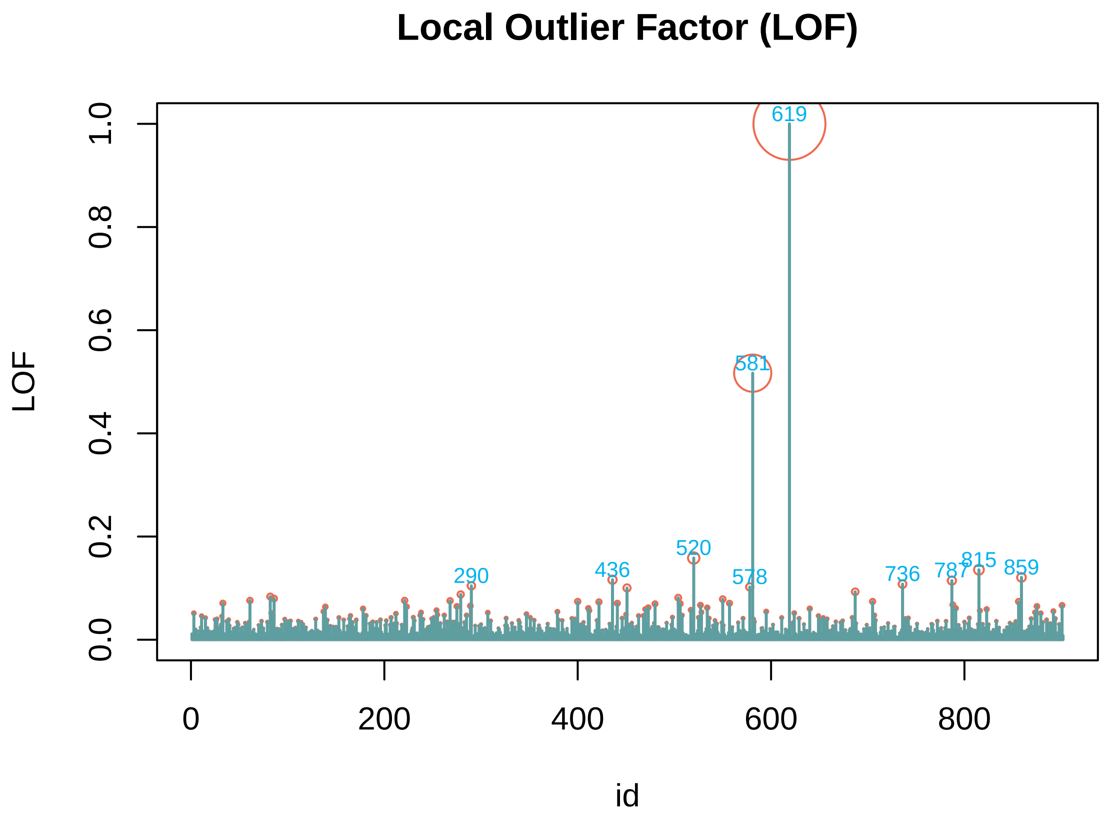
```
# ?Rlof

outlier.scores <- lof(dat, k=5);  #outlier.scores
which(outlier.scores > quantile(outlier.scores, 0.95))
```

```
##  [1]  33  61  82  86 139 178 221 223 268 275 279 289 290 400 411 422 436 441 451
## [20] 470 473 480 504 506 517 520 527 534 550 557 578 581 619 640 687 705 736 787
## [39] 788 791 815 823 856 859 875 901
```

```
# PLOT OF THE LOF SCORES
score <- scale(outlier.scores, center = min(outlier.scores),
    scale = max(outlier.scores)-min(outlier.scores)) # NORMALIZED TO RANGE[0,1]
par(mfrow=c(1,1), mar=rep(4,4))
plot(x=1:length(score), score, type="p", pch=1,
    main="Local Outlier Factor (LOF)",
        xlab="id", ylab="LOF", cex=score*5, col="coral2")
add.seg <- function(x) segments(x0=x[1], y0=0, x1=x[1], y1=x[2],
    lty=1, lwd=1.5, col="cadetblue")
apply(data.frame(id=1:length(score), score=score), 1, FUN=add.seg)
```

```
## NULL
```

```
eps <- 0.99
id.outliers <- which(outlier.scores > quantile(outlier.scores, eps))
text(id.outliers, score[id.outliers]+0.02, label=id.outliers,
    col="deepskyblue2", cex=0.7)
```



The observation 581 and 619 are most outlying as seen form the plot using local outlier factor method.