Week 1 Code Examples, CSC 400, Spring 2024

1. Review of Correlation
2. Visualization
   a. Correlation plots
   b. Heat maps
3. Frequent item set mining, association rule mining
   a. Apriori
   b. FP Growth
   c. Eclat
4. Rule induction
5. Visualization
   a. Association rule diagrams

Correlation analysis

```
data("mtcars")
my_data <- mtcars[, c(1,3,4,5,6,7)]
head(my_data, 6)

result <- cor(my_data)
round(result, 2)

library("Hmisc")
result2 <- rcorr(as.matrix(my_data))
result2

library(corrplot)
corrplot(result, type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45)

library("PerformanceAnalytics")
my_data <- mtcars[, c(1,3,4,5,6,7)]
chart.Correlation(my_data, histogram=TRUE, pch=19)

col<- colorRampPalette(c("green", "white", "purple"))(20)
heatmap(x = result, col = col, symm = TRUE)
```

Visualization

```
library(corrplot)
corrplot(result, method="circle")
corrplot(result, method="pie")
corrplot(result, method="color")
corrplot(result, method="number")
corrplot(result, method="circle", type="upper")
corrplot(result, method="circle", type="lower")
corrplot(result, type="upper", order="hclust", col=c("black", "white"),
         bg="lightblue")
library(RColorBrewer)
corrplot(result, type="upper", order="hclust",
         col=brewer.pal(n=8, name="PuOr"))
```

The default correlation method is Pearson correlation, but you can get Spearman or Kendall Tau by modifying the creation of your initial correlation matrix.

```
result3 <- cor(my_data, method="spearman")
round(result3, 2)

result4 <- cor(my_data, method="kendall")
round(result4, 2)
```

Frequent Item Set Mining and Association Rule Mining

We'll create a dataset to play with. You need a dataset of a particular form for this analysis.

```
market_basket <-
  list(
    c("apple", "beer", "rice", "meat"),
    c("apple", "beer", "rice"),
    c("apple", "beer"),
    c("apple", "pear"),
    c("milk", "beer", "rice", "meat"),
    c("milk", "beer", "rice"),
    c("milk", "beer"),
    c("milk", "pear")
  )
names(market_basket) <- paste("T", c(1:8), sep = "")
```

Initial analysis

```
library(arules)
trans <- as(market_basket, "transactions")
dim(trans)
itemLabels(trans)
summary(trans)
image(trans)
itemFrequencyPlot(trans, topN=10,  cex.names=1)
```

Apriori Algorithm

We can look at all rules, and also prune the list.

```
rules <- apriori(trans,
                 parameter = list(supp=0.3, conf=0.5,
                                  maxlen=10,
                                  target= "rules"))
summary(rules)
inspect(rules)

rules <- apriori(trans,
                 parameter = list(supp=0.3, conf=0.5,
                                  maxlen=10,
                                  minlen=2,
                                  target= "rules"))
inspect(rules)
```

We can also look at rules for specific items, such as before purchasing a specific item or after.

```
beer_rules_rhs <- apriori(trans,
                          parameter = list(supp=0.3, conf=0.5,
                                           maxlen=10,
                                           minlen=2),
                          appearance = list(default="lhs", rhs="beer"))
inspect(beer_rules_rhs)

beer_rules_lhs <- apriori(trans,
                          parameter = list(supp=0.3, conf=0.5,
                                           maxlen=10,
                                           minlen=2),
                          appearance = list(lhs="beer", default="rhs"))
inspect(beer_rules_lhs)
```

We have several methods to visualize the result.

```
library(arulesViz)
plot(rules)
plot(rules, measure = "confidence")
plot(rules, method = "two-key plot")
plot(rules, engine = "plotly")

subrules <- head(rules, n = 10, by = "confidence")
plot(subrules, method = "graph",  engine = "htmlwidget")

plot(subrules, method="paracoord")
```

FP Growth and Eclat

```
library(arules)
library(arulesViz)
library(tidyverse)
library(fim4r)
fim4r() #then answer yes

data(Groceries)
transactions <- Groceries
```

We'll use a larger dataset here. The dataset can be examines similarly to the above examples. If the fim4r installation fails, there are resources below that can help. You'll need to install the latest version of RTools first (to compile from a C source).

```
fpgrowth <- fim4r(Groceries, method = "fpgrowth", target = "rules", supp = 10, conf = 10)
summary(fpgrowth)

eclat <- fim4r(transactions, method = "eclat", target = "rules", supp = 10, conf = 10)
summary(eclat)
```

These algorithms can also be run in other packages such as sparklyr which is used in parallel processing contexts.

Resources:

1. http://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide-to-analyze-format-and-visualize-a-correlation-matrix-using-r-software
2. http://www.sthda.com/english/wiki/visualize-correlation-matrix-using-correlogram
3. http://www.sthda.com/english/wiki/elegant-correlation-table-using-xtable-r-package
4. https://bookdown.org/siju_swamy/Stat_Lab/correlation-and-regression-analysis-in-r.html
5. https://r-graph-gallery.com/79-levelplot-with-ggplot2.html
6. https://www.kirenz.com/post/2020-05-14-r-association-rule-mining/
7. https://rpubs.com/zehrausta/1009744
8. https://www.rdocumentation.org/packages/arules/versions/1.7-3/topics/fim4r
9. https://stackoverflow.com/questions/76124418/cant-install-package-error-compilation-failed-for-package-fim4r-arules
10. https://spark.rstudio.com/packages/sparklyr/latest/reference/ml_fpgrowth.html