Challenge-1

Question 1: Create a sparse matrix by transforming the survey data into transactions data (hint: transactions must be categorical, so find and retain only the categorical features). Then, drop SurveyYr and Race3.

The problem asks us to drop `SurveyYr` and `Race3` from the dataset. Apart from these two variables, I also drop the `Gender` variable because I find the `Sex` variable basically is the same with `Gender`. So I only remain `Sex` in the dataset.

```
# load libraries
library(tidyverse)
library(arules)
library("gridExtra")
# clean data and transform it to transactions
load(file = "NHANESraw.rda")
dataset <- NHANESraw %>%
  select_if(~class(.) == 'factor') %>%
  select(-c(SurveyYr, Race3, Gender))
trans = as(dataset, "transactions")
# take a look at the first five items
inspect(trans[1:5])
```

```
##     items                          transactionID
## [1] {Sex=male,
##      SexOrientation=Heterosexual,
##      Race1=White,
##      Education=High School,
##      MaritalStatus=Married,
##      HHIncome=25000-34999,
##      HomeOwn=Own,
##      Work=NotWorking,
##      BMI_WHO=30.0_plus,
##      Diabetes=No,
##      HealthGen=Good,
##      LittleInterest=Most,
##      Depressed=Several,
##      SleepTrouble=Yes,
##      PhysActive=No,
##      Alcohol12PlusYr=Yes,
##      SmokeNow=No,
##      Smoke100=Yes,
##      Marijuana=Yes,
##      RegularMarij=No,
##      HardDrugs=Yes,
##      SexEver=Yes,
##      SameSex=No}                               1
## [2] {Sex=male,
##      Race1=Other,
```

```
##      HHIncome=20000-24999,
##      HomeOwn=Own,
##      BMI_WHO=12.0_18.5,
##      Diabetes=No}                              2
## [3] {Sex=male,
##      Race1=Black,
##      HHIncome=45000-54999,
##      HomeOwn=Own,
##      Work=NotWorking,
##      BMI_WHO=18.5_to_24.9,
##      Diabetes=No,
##      HealthGen=Vgood,
##      SleepTrouble=No,
##      PhysActive=Yes}                           3
## [4] {Sex=male,
##      Race1=Black,
##      HHIncome=20000-24999,
##      HomeOwn=Rent,
##      BMI_WHO=12.0_18.5,
##      Diabetes=No}                              4
## [5] {Sex=female,
##      Race1=Black,
##      Education=High School,
##      MaritalStatus=Widowed,
##      HHIncome=10000-14999,
##      HomeOwn=Rent,
##      Work=NotWorking,
##      BMI_WHO=30.0_plus,
##      Diabetes=Yes,
##      HealthGen=Fair,
##      LittleInterest=Most,
##      Depressed=Most,
##      SleepTrouble=No,
##      PhysActive=No,
##      Alcohol12PlusYr=No,
##      SmokeNow=Yes,
##      Smoke100=Yes,
##      HardDrugs=No,
##      SexEver=Yes,
##      SameSex=No}                               5
```

```
itemFrequency(trans[, 1:5])
```

```
##                  Sex=female                    Sex=male
##                 0.503227714                 0.496772286
##     SexOrientation=Bisexual SexOrientation=Heterosexual
##                 0.009954171                 0.321982950
##   SexOrientation=Homosexual
##                 0.005469866
```

The sex pattern shows here. We can see Sex=female is present in 50.32% of transactions while Sex=female in 49.68% and so on.

Question 2: Summarize the transactions numerically.

```
summary(trans)
```

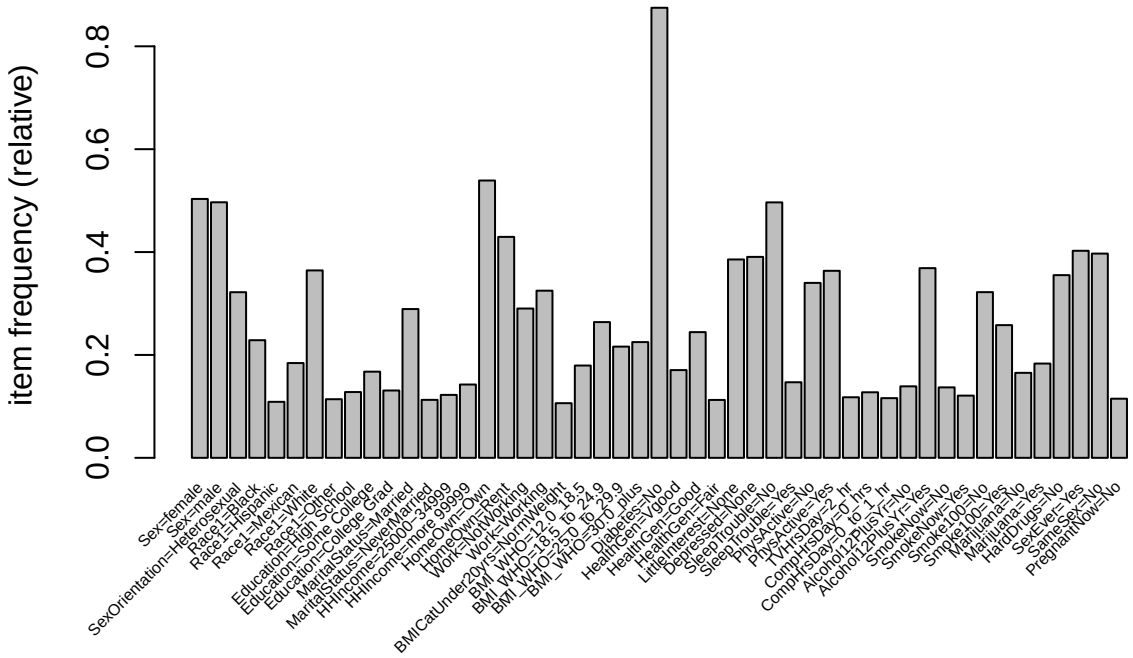
```
## transactions as itemMatrix in sparse format with
##  20293 rows (elements/itemsets/transactions) and
##  97 columns (items) and a density of 0.1568354
##
## most frequent items:
##     Diabetes=No     HomeOwn=Own      Sex=female        Sex=male SleepTrouble=No
##           17754           10939           10212           10081           10077
##         (Other)
##          249655
##
## element (itemset/transaction) length distribution:
## sizes
##    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17
##    2   93  815  801 2002  224  751 2065  378  706  377  749  431  372  584  675
##   18   19   20   21   22   23   24   25   26
##  588  753  620 1335 1714 1764 1395  903  196
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2.00    9.00   16.00   15.21   22.00   26.00
##
## includes extended item information - examples:
##                    labels      variables   levels
## 1              Sex=female            Sex   female
## 2                Sex=male            Sex     male
## 3 SexOrientation=Bisexual SexOrientation Bisexual
##
## includes extended transaction information - examples:
##   transactionID
## 1             1
## 2             2
## 3             3
```

From the summary table, we can see that there are 20293 transactions and 97 items. Density is 15.68% which means there are 15.68% nonzero matrix cells. `Diabetes=No` is the most frequent item, `HomeOwn=Own` the second, and `Sex=female` the third. Further we also get statistics about the size of the transactions. The average items/transaction are 15.21.

Question 3: Summarize the transactions visually.

1. First, we can visualize item frequency plot with all items.

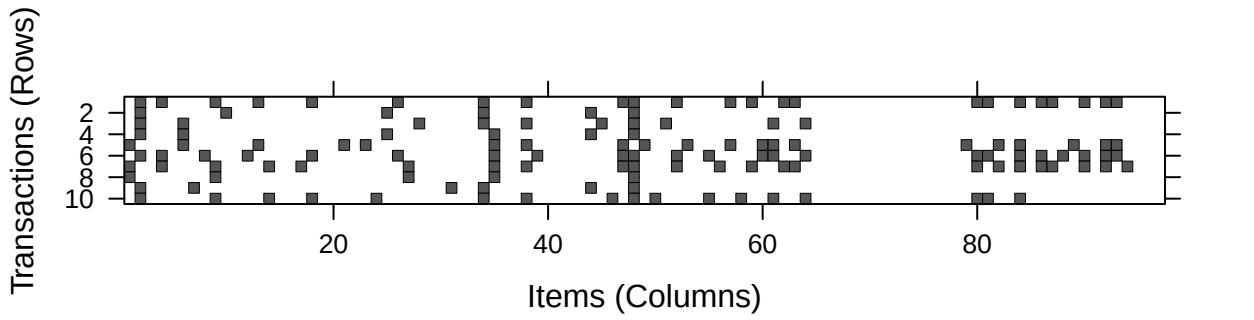
```
itemFrequencyPlot(trans, support=0.1, cex.names=.5)
```



The histogram shows the items in the transaction data with at least 10 percent support. We can see that the most frequent item is `Diabetes=No`, same with our numerical summary result. The second most frequent item is `HomeOwn=Own`.

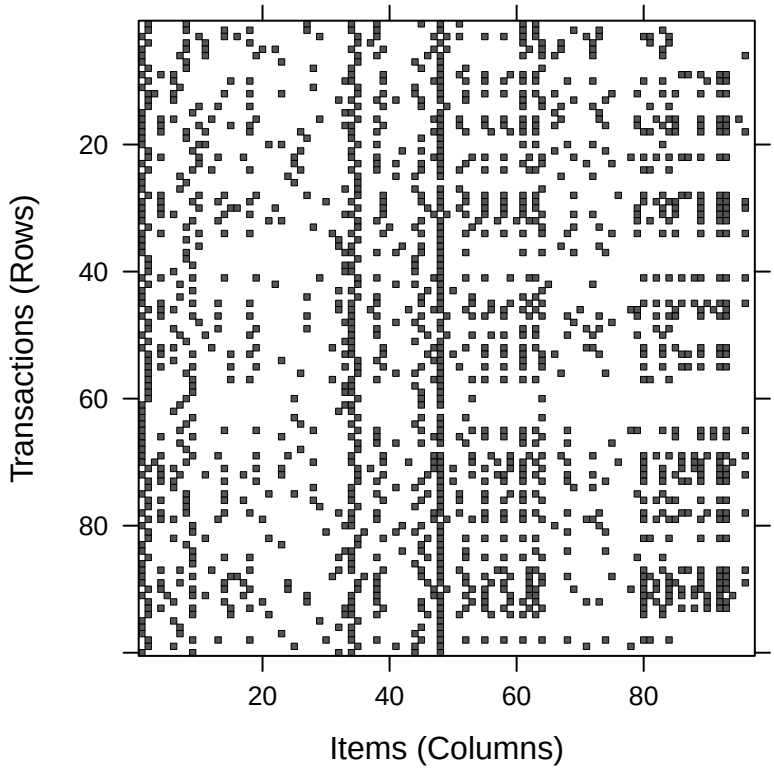
2. Visualizing the sparse matrix.

```
image(trans[1:10])
```



The image function displays first 10 rows and 97 columns. Cells in the matrix are filled with black for transactions (rows) where the item (column) was answered.

```
image(sample(trans, 100))
```



Using sample function with image function creates a big visualization of the sparse matrix. A random 100 sample is plotted and thus we can get insights about the items in a transaction. We can see that some columns are heavily populated with the black dots indicating those items are more popular and are present in many transactions.

Question 4: Fit an apriori algorithm, initialized at support = 0.1 and confidence = 0.5. Tune and update as necessary. Be sure also to set the minimum length to something reasonable.

1. Choice of support and confidence.

The first step in order to create a set of association rules is to determine the optimal thresholds for support and confidence. If we set these values too low, then the algorithm will take longer to execute and we will get a lot of rules (most of them will not be useful). We can try different values of support and confidence and see graphically how many rules are generated for each combination.

```
# Support and confidence values
supportLevels <- c(0.3, 0.2, 0.1, 0.05)
confidenceLevels <- c(0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1)

# Empty integers
rules_sup30 <- integer(length=9)
rules_sup20 <- integer(length=9)
rules_sup10 <- integer(length=9)
rules_sup5 <- integer(length=9)

# Apriori algorithm with a support level of 30%
for (i in 1:length(confidenceLevels)) {

  rules_sup30[i] <- length(apriori(trans, parameter=list(sup=supportLevels[1],
```

```
                                    conf=confidenceLevels[i], minlen = 4, target="rules")))

}

# Apriori algorithm with a support level of 20%
for (i in 1:length(confidenceLevels)){

  rules_sup20[i] <- length(apriori(trans, parameter=list(sup=supportLevels[2],
                                    conf=confidenceLevels[i], minlen = 4, target="rules")))

}

# Apriori algorithm with a support level of 10%
for (i in 1:length(confidenceLevels)){

  rules_sup10[i] <- length(apriori(trans, parameter=list(sup=supportLevels[3],
                                    conf=confidenceLevels[i], minlen = 4, target="rules")))

}

# Apriori algorithm with a support level of 5%
for (i in 1:length(confidenceLevels)){

  rules_sup5[i] <- length(apriori(trans, parameter=list(sup=supportLevels[4],
                                     conf=confidenceLevels[i], minlen = 4, target="rules")))

}
```

In the following graphs we can see the number of rules generated with a support level of 30%, 20%, 10% and 5%.

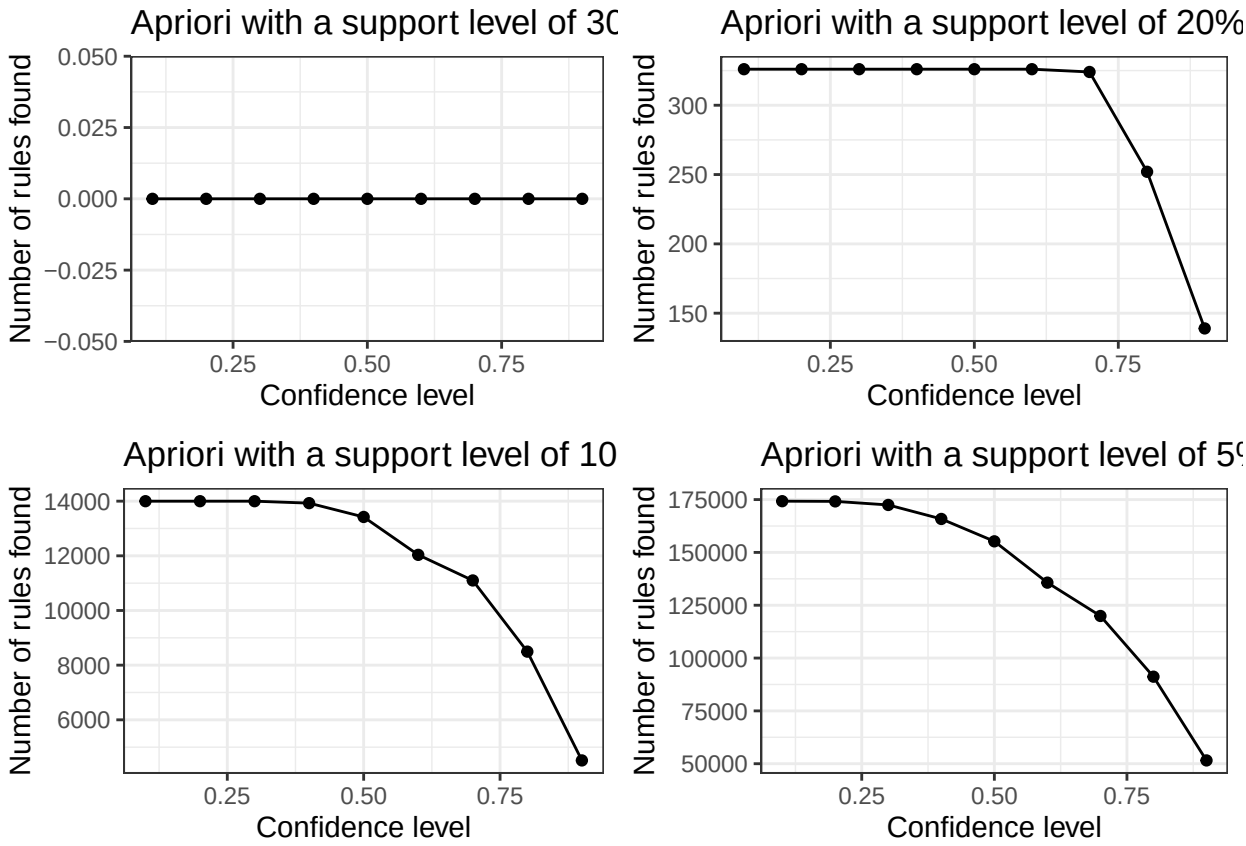
```
# Number of rules found with a support level of 30%
plot1 <- qplot(confidenceLevels, rules_sup30, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 30%") +
  theme_bw()

# Number of rules found with a support level of 20%
plot2 <- qplot(confidenceLevels, rules_sup20, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 20%") +
  theme_bw()

# Number of rules found with a support level of 10%
plot3 <- qplot(confidenceLevels, rules_sup10, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 10%") +
  theme_bw()

# Number of rules found with a support level of 5%
plot4 <- qplot(confidenceLevels, rules_sup5, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 5%") +
  theme_bw()
```

```
# Subplot
grid.arrange(plot1, plot2, plot3, plot4, ncol=2)
```



Let's analyze the results:

- Support level of 30%: No rules.
- Support level of 20%. We started to get dozens of rules, of which around 300 have a confidence of at least 70%. When I look at the exact result when I set the level as 20%, I find that the `Marijuana` variable is omited, which will generate very different results. Therefore, I may not use the level 20%.
- Support level of 10%. There are many rules at the lower confidence intervals. But the number is acceptable when the confidence interval is relevant high. The critical point is that the `Marijuana` variable is of interest to me, so I need to select levels that can include this variable.
- Support level of 5%. Too many rules to analyze!

In this case, I think we want to include the `Marijuana` variable and also have resonable number of rules. So we can choose the 10% support level with a 50% confidence level. To sum up, we are going to use a support level of 10% and a confidence level of 50%.

2. Training a model with Apriori Algorithm.

Note that minlen is set to 4 to remove rules that contain less than four items, which may not be meaningful responses.

```
t_rules <- apriori(trans,
                   parameter = list(support = 0.1,
                                    confidence = 0.5,
                                    minlen = 4))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5     0.1      4
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 2029
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[97 item(s), 20293 transaction(s)] done [0.02s].
## sorting and recoding items ... [49 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 7 8 done [0.15s].
## writing ... [13422 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

Question 5: Summarize the rules numerically.

```
summary(t_rules)
```

```
## set of 13422 rules
##
## rule length distribution (lhs + rhs):sizes
##    4    5    6    7    8
## 4573 4977 2854  882  136
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.000   4.000   5.000   5.034   6.000   8.000
##
## summary of quality measures:
##     support         confidence        coverage           lift
##  Min.   :0.1000   Min.   :0.5002   Min.   :0.1009   Min.   :0.9222
##  1st Qu.:0.1070   1st Qu.:0.7502   1st Qu.:0.1269   1st Qu.:1.6844
##  Median :0.1174   Median :0.8375   Median :0.1478   Median :2.1222
##  Mean   :0.1259   Mean   :0.8181   Mean   :0.1578   Mean   :1.9962
##  3rd Qu.:0.1359   3rd Qu.:0.9244   3rd Qu.:0.1795   3rd Qu.:2.3696
##  Max.   :0.2771   Max.   :0.9933   Max.   :0.3305   Max.   :4.0869
##      count
##  Min.   :2030
##  1st Qu.:2172
##  Median :2383
##  Mean   :2556
##  3rd Qu.:2757
##  Max.   :5623
##
## mining info:
##   data ntransactions support confidence
##  trans         20293     0.1        0.5
```

As we can see there are 13422 association rules. Rule length distribution tells us how many items are present in how many rules. 4 items are present in 4573 rules, 5 in 4977 rules, 6 in 2854 rules, and 7 in 882 rules. There is also a summary of quality measures: min, max, median, mean and quantile values for support, confidence and lift.

```
inspect(t_rules[1:5])
```

```
##     lhs                                rhs                                support confidence  coverage
## [1] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      Marijuana=No}                  => {HardDrugs=No}                0.1068349  0.9841126 0.1085596 2.
## [2] {Smoke100=No,
##      Marijuana=No,
##      HardDrugs=No}                  => {SexOrientation=Heterosexual} 0.1068349  0.9264957 0.1153107 2.
## [3] {SexOrientation=Heterosexual,
##      Marijuana=No,
##      HardDrugs=No}                  => {Smoke100=No}                 0.1068349  0.7207447 0.1482285 2.
## [4] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      HardDrugs=No}                  => {Marijuana=No}                0.1068349  0.6707921 0.1592667 4.
## [5] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      Marijuana=No}                  => {SameSex=No}                  0.1072784  0.9881979 0.1085596 2.
```

Question 6: Create three rule subsets conditioned on "interestingness".

Question 7: Inspect and print the top 5 most accurate rule subsets.

1. Rules with rhs containing "not depressed" with a minimum lift ratio of 1.25.

```
depress_rules_1 <- subset(t_rules, subset = rhs %pin% "Depressed=None" & lift >= 1.25)
summary(depress_rules_1)
```

```
## set of 1199 rules
##
## rule length distribution (lhs + rhs):sizes
##   4   5   6   7   8
## 383 437 271  93  15
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.000   4.000   5.000   5.099   6.000   8.000
##
## summary of quality measures:
##     support         confidence        coverage           lift
##  Min.   :0.1001   Min.   :0.5084   Min.   :0.1073   Min.   :1.302
##  1st Qu.:0.1074   1st Qu.:0.8029   1st Qu.:0.1272   1st Qu.:2.056
##  Median :0.1178   Median :0.8348   Median :0.1423   Median :2.137
##  Mean   :0.1271   Mean   :0.8431   Mean   :0.1517   Mean   :2.159
##  3rd Qu.:0.1375   3rd Qu.:0.9042   3rd Qu.:0.1653   3rd Qu.:2.315
##  Max.   :0.2532   Max.   :0.9353   Max.   :0.3305   Max.   :2.395
##      count
##  Min.   :2031
##  1st Qu.:2179
##  Median :2390
##  Mean   :2580
##  3rd Qu.:2790
```

```
##  Max.   :5138
##
## mining info:
##   data ntransactions support confidence
##  trans         20293     0.1        0.5
```

There are total 1199 rules in this subset. In order to get a sense of the top 5 most accurate rule subsets, I think we can sort the rules either by "confidence" or by "lift". But I'm not sure here how you define "accurate". The confidence value indicates how reliable this rule is. The higher the value, the more likely the head items occur in a group if it is known that all body items are contained in that group. The lift value is a measure of importance of a rule.

```
inspect(sort(depress_rules_1, by = "confidence")[1:5])
```

```
##     lhs                          rhs                support confidence  coverage     lift count
## [1] {MaritalStatus=Married,
##      LittleInterest=None,
##      SleepTrouble=No,
##      HardDrugs=No,
##      SameSex=No}             => {Depressed=None} 0.1003794  0.9352617 0.1073276 2.394558  2037
## [2] {MaritalStatus=Married,
##      LittleInterest=None,
##      SleepTrouble=No,
##      HardDrugs=No}           => {Depressed=None} 0.1027448  0.9328859 0.1101365 2.388475  2085
## [3] {MaritalStatus=Married,
##      Work=Working,
##      LittleInterest=None}    => {Depressed=None} 0.1068349  0.9324731 0.1145715 2.387418  2168
## [4] {MaritalStatus=Married,
##      HomeOwn=Own,
##      LittleInterest=None,
##      SleepTrouble=No}        => {Depressed=None} 0.1073276  0.9307692 0.1153107 2.383056  2178
## [5] {MaritalStatus=Married,
##      LittleInterest=None,
##      Smoke100=No}            => {Depressed=None} 0.1032376  0.9302842 0.1109742 2.381814  2095
```

```
inspect(sort(depress_rules_1, by = "lift")[1:5])
```

```
##     lhs                          rhs                support confidence  coverage     lift count
## [1] {MaritalStatus=Married,
##      LittleInterest=None,
##      SleepTrouble=No,
##      HardDrugs=No,
##      SameSex=No}             => {Depressed=None} 0.1003794  0.9352617 0.1073276 2.394558  2037
## [2] {MaritalStatus=Married,
##      LittleInterest=None,
##      SleepTrouble=No,
##      HardDrugs=No}           => {Depressed=None} 0.1027448  0.9328859 0.1101365 2.388475  2085
## [3] {MaritalStatus=Married,
##      Work=Working,
##      LittleInterest=None}    => {Depressed=None} 0.1068349  0.9324731 0.1145715 2.387418  2168
## [4] {MaritalStatus=Married,
##      HomeOwn=Own,
##      LittleInterest=None,
##      SleepTrouble=No}        => {Depressed=None} 0.1073276  0.9307692 0.1153107 2.383056  2178
## [5] {MaritalStatus=Married,
##      LittleInterest=None,
```

```
##      Smoke100=No}           => {Depressed=None} 0.1032376  0.9302842 0.1109742 2.381814  2095
```

2. Rules with rhs containing "never smoked marijuana" with a minimum lift ratio of 1.5.

```
smoke_rules_2 <- subset(t_rules, subset = rhs %pin% "Marijuana=No" & lift >= 1.5)
summary(smoke_rules_2)
```

```
## set of 47 rules
##
## rule length distribution (lhs + rhs):sizes
##  4  5  6  7
## 18 19  9  1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.000   4.000   5.000   4.851   5.000   7.000
##
## summary of quality measures:
##     support         confidence        coverage           lift
##  Min.   :0.1002   Min.   :0.5005   Min.   :0.1565   Min.   :3.029
##  1st Qu.:0.1035   1st Qu.:0.5448   1st Qu.:0.1857   1st Qu.:3.297
##  Median :0.1068   Median :0.5606   Median :0.1953   Median :3.393
##  Mean   :0.1114   Mean   :0.5601   Mean   :0.1998   Mean   :3.390
##  3rd Qu.:0.1150   3rd Qu.:0.5724   3rd Qu.:0.2072   3rd Qu.:3.464
##  Max.   :0.1464   Max.   :0.6753   Max.   :0.2587   Max.   :4.087
##      count
##  Min.   :2033
##  1st Qu.:2100
##  Median :2168
##  Mean   :2261
##  3rd Qu.:2334
##  Max.   :2970
##
## mining info:
##   data ntransactions support confidence
##  trans         20293     0.1        0.5
```

```
inspect(sort(smoke_rules_2, by='confidence')[1:5])
```

```
##     lhs                              rhs              support confidence  coverage     lift count
## [1] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      HardDrugs=No,
##      SameSex=No}                  => {Marijuana=No} 0.1056522  0.6752756 0.1564579 4.086898  2144
## [2] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      HardDrugs=No}                => {Marijuana=No} 0.1068349  0.6707921 0.1592667 4.059763  2168
## [3] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      SameSex=No}                  => {Marijuana=No} 0.1072784  0.6352495 0.1688760 3.844652  2177
## [4] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      SexEver=Yes}                 => {Marijuana=No} 0.1001823  0.6125339 0.1635539 3.707173  2033
## [5] {SexOrientation=Heterosexual,
##      SleepTrouble=No,
##      HardDrugs=No,
##      SameSex=No}                  => {Marijuana=No} 0.1209284  0.5827594 0.2075100 3.526972  2454
```

```
inspect(sort(smoke_rules_2, by='lift')[1:5])
```

```
##     lhs                                rhs              support confidence  coverage     lift count
## [1] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      HardDrugs=No,
##      SameSex=No}                    => {Marijuana=No} 0.1056522  0.6752756 0.1564579 4.086898  2144
## [2] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      HardDrugs=No}                  => {Marijuana=No} 0.1068349  0.6707921 0.1592667 4.059763  2168
## [3] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      SameSex=No}                    => {Marijuana=No} 0.1072784  0.6352495 0.1688760 3.844652  2177
## [4] {SexOrientation=Heterosexual,
##      Smoke100=No,
##      SexEver=Yes}                   => {Marijuana=No} 0.1001823  0.6125339 0.1635539 3.707173  2033
## [5] {SexOrientation=Heterosexual,
##      SleepTrouble=No,
##      HardDrugs=No,
##      SameSex=No}                    => {Marijuana=No} 0.1209284  0.5827594 0.2075100 3.526972  2454
```

3. Rules with rhs containing "no sleep trouble" with a minimum lift ratio of 1.75.

```
sleep_rules_3 <- subset(t_rules, subset = rhs %pin% "SleepTrouble=No" & lift >= 1.75)
summary(sleep_rules_3)
```

```
## set of 18 rules
##
## rule length distribution (lhs + rhs):sizes
## 4 5 6 7 8
## 1 6 7 3 1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.000   5.000   6.000   5.833   6.000   8.000
##
## summary of quality measures:
##     support         confidence        coverage           lift
##  Min.   :0.1004   Min.   :0.8692   Min.   :0.1144   Min.   :1.750
##  1st Qu.:0.1010   1st Qu.:0.8700   1st Qu.:0.1160   1st Qu.:1.752
##  Median :0.1029   Median :0.8732   Median :0.1176   Median :1.758
##  Mean   :0.1043   Mean   :0.8730   Mean   :0.1194   Mean   :1.758
##  3rd Qu.:0.1047   3rd Qu.:0.8753   3rd Qu.:0.1203   3rd Qu.:1.763
##  Max.   :0.1196   Max.   :0.8776   Max.   :0.1375   Max.   :1.767
##      count
##  Min.   :2037
##  1st Qu.:2050
##  Median :2088
##  Mean   :2116
##  3rd Qu.:2125
##  Max.   :2427
##
## mining info:
##   data ntransactions support confidence
##  trans         20293     0.1        0.5
```

```r
inspect(sort(sleep_rules_3, by='confidence')[1:5])
```

```
##     lhs                                rhs                 support confidence  coverage     lift count
## [1] {Diabetes=No,
##      Depressed=None,
##      Marijuana=No,
##      SameSex=No}                    => {SleepTrouble=No} 0.1003794  0.8776389 0.1143744 1.767384  2037
## [2] {SexOrientation=Heterosexual,
##      Work=Working,
##      Diabetes=No,
##      LittleInterest=None,
##      Depressed=None,
##      HardDrugs=No,
##      SameSex=No}                    => {SleepTrouble=No} 0.1027448  0.8771561 0.1171340 1.766411  2085
## [3] {SexOrientation=Heterosexual,
##      Work=Working,
##      Diabetes=No,
##      LittleInterest=None,
##      Depressed=None,
##      HardDrugs=No}                  => {SleepTrouble=No} 0.1045188  0.8768086 0.1192037 1.765712  2121
## [4] {Diabetes=No,
##      Depressed=None,
##      Marijuana=No,
##      HardDrugs=No}                  => {SleepTrouble=No} 0.1010201  0.8753202 0.1154093 1.762714  2050
## [5] {Sex=male,
##      Diabetes=No,
##      Depressed=None,
##      HardDrugs=No,
##      SameSex=No}                    => {SleepTrouble=No} 0.1034347  0.8753128 0.1181688 1.762699  2099
```

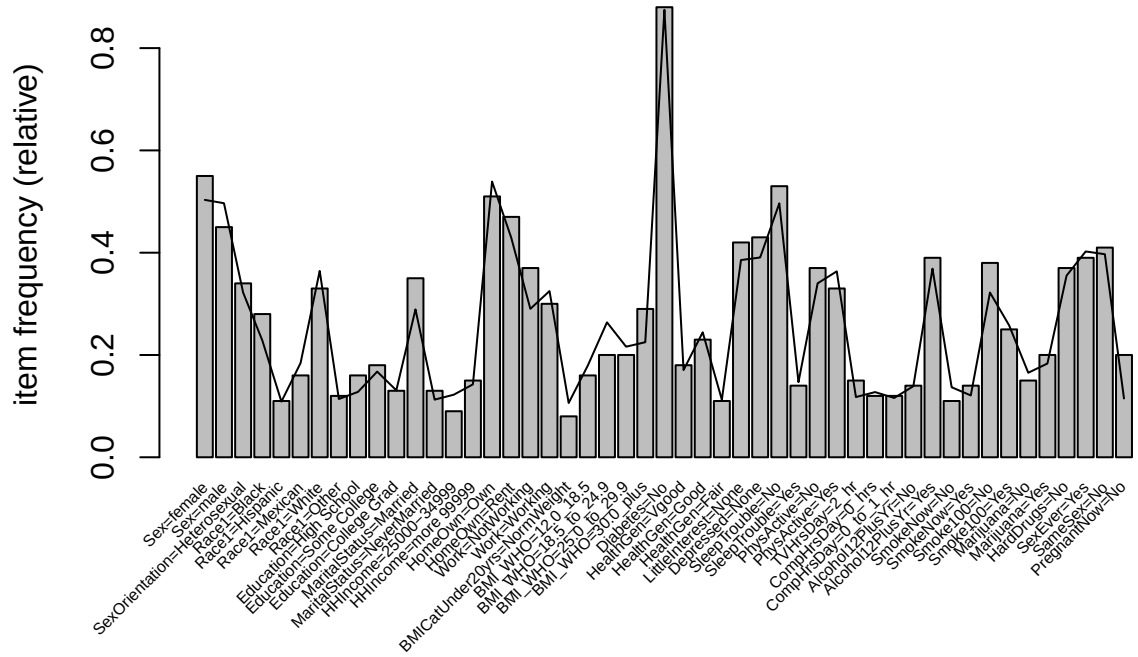
```r
inspect(sort(sleep_rules_3, by='lift')[1:5])
```

```
##     lhs                                rhs                 support confidence  coverage     lift count
## [1] {Diabetes=No,
##      Depressed=None,
##      Marijuana=No,
##      SameSex=No}                    => {SleepTrouble=No} 0.1003794  0.8776389 0.1143744 1.767384  2037
## [2] {SexOrientation=Heterosexual,
##      Work=Working,
##      Diabetes=No,
##      LittleInterest=None,
##      Depressed=None,
##      HardDrugs=No,
##      SameSex=No}                    => {SleepTrouble=No} 0.1027448  0.8771561 0.1171340 1.766411  2085
## [3] {SexOrientation=Heterosexual,
##      Work=Working,
##      Diabetes=No,
##      LittleInterest=None,
##      Depressed=None,
##      HardDrugs=No}                  => {SleepTrouble=No} 0.1045188  0.8768086 0.1192037 1.765712  2121
## [4] {Diabetes=No,
##      Depressed=None,
##      Marijuana=No,
##      HardDrugs=No}                  => {SleepTrouble=No} 0.1010201  0.8753202 0.1154093 1.762714  2050
```

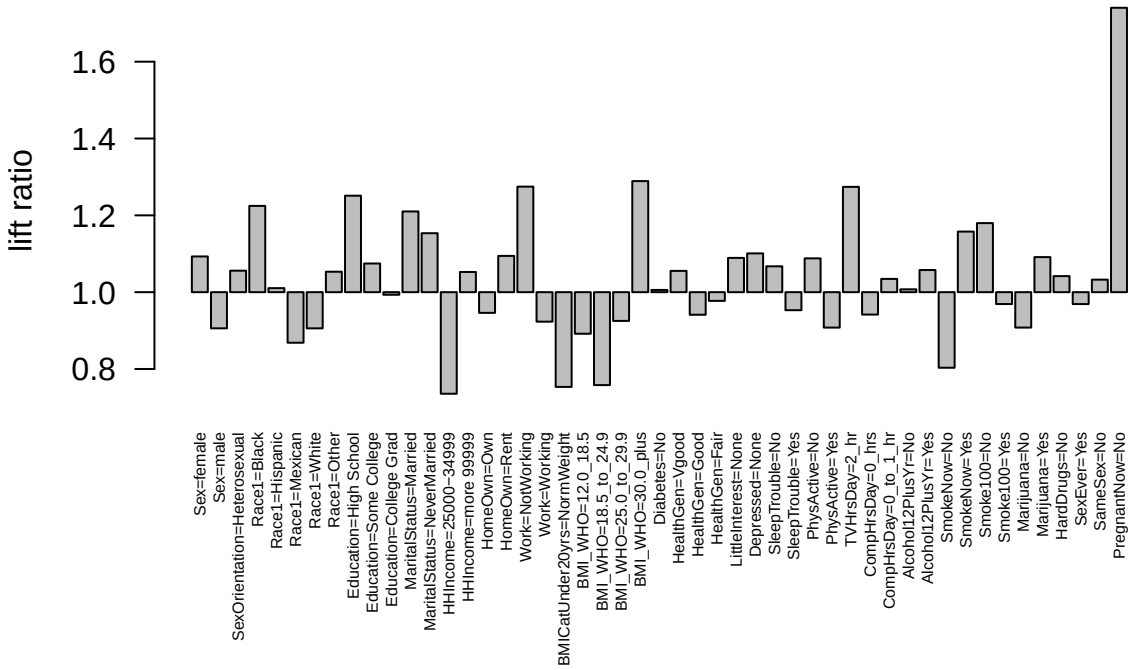
```
## [5] {Sex=male,
##      Diabetes=No,
##      Depressed=None,
##      HardDrugs=No,
##      SameSex=No}                  => {SleepTrouble=No} 0.1034347  0.8753128 0.1181688 1.762699  2099
```

Bonus Questions

```
# draw a sample
set.seed(123)
new_trans <- sample(trans, size = 100)
# plot the sample against the full population
itemFrequencyPlot(new_trans, population = trans, support=0.1, cex.names=.5)
```




```
# compare the sample to the population via lift ratios
itemFrequencyPlot(new_trans, population = trans, support=0.1, lift=TRUE, cex.names=.5)
```



From the graph, we can see that there are some indicators with lift ratios above 1 and some below 1. It tells us that some factors (lift ratios > 1) such as `PregnantNow=No`, `Race1=Black`, etc. are overrepresented in this sample, while some factors (lift ratios < 1) such as `HHIncome=25000-34999`, `SmokeNow=No`, etc. are underrepresented in this sample. Few indicators occur in the sample in the same proportion as in the population (lift ratios $= 1$).