Final Project

## R Markdown

#install.packages("arules")  
#install.packages("ggplot2")  
  
library(arules)

## Loading required package: Matrix

##   
## Attaching package: 'arules'

## The following objects are masked from 'package:base':  
##   
## abbreviate, write

data <- read.csv(file = "AppleStoreClean.csv", head = TRUE)  
tdata <- read.transactions("AppleStoreClean.csv", quote="", sep=",")

## Warning in asMethod(object): removing duplicated items in transactions

cutdata <- read.transactions("AppleStoreCut.csv", quote="", sep=",")

## Warning in asMethod(object): removing duplicated items in transactions

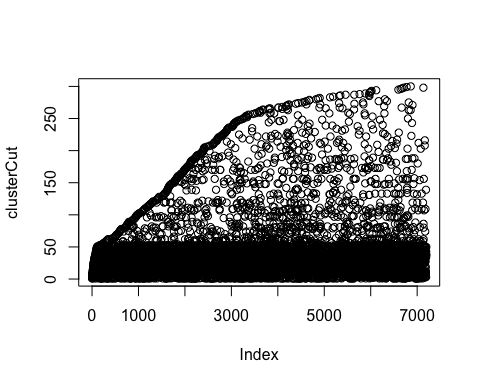
modified\_data <- data

## HCLUST

distance = dist(as.matrix(data), method = "euclidean")

## Warning in dist(as.matrix(data), method = "euclidean"): NAs introduced by  
## coercion

# perform clustering  
hc = hclust(distance)  
  
# cut the tree  
clusterCut <- cutree(hc, 300)  
  
# plot dendrogram  
plot(clusterCut)



#write.csv(clusterCut, file = =hclust.csv")

## Apriori - Regular

apprules <- apriori(tdata, parameter = list(support = 0.01, confidence = 0.35, minlen = 2))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.35 0.1 1 none FALSE TRUE 5 0.01 2  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 71   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[18421 item(s), 7198 transaction(s)] done [0.03s].  
## sorting and recoding items ... [67 item(s)] done [0.00s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 5 6 7 done [0.01s].  
## writing ... [4520 rule(s)] done [0.00s].  
## creating S4 object ... done [0.01s].

summary(apprules)

## set of 4520 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2 3 4 5 6 7   
## 313 1318 1669 937 262 21   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 3.000 4.000 3.907 5.000 7.000   
##   
## summary of quality measures:  
## support confidence lift count   
## Min. :0.01000 Min. :0.3502 Min. :0.5062 Min. : 72.0   
## 1st Qu.:0.01292 1st Qu.:0.4876 1st Qu.:0.9149 1st Qu.: 93.0   
## Median :0.01848 Median :0.5953 Median :1.0263 Median : 133.0   
## Mean :0.03435 Mean :0.5999 Mean :1.0682 Mean : 247.2   
## 3rd Qu.:0.03418 3rd Qu.:0.6949 3rd Qu.:1.1626 3rd Qu.: 246.0   
## Max. :0.43137 Max. :1.0000 Max. :5.9464 Max. :3105.0   
##   
## mining info:  
## data ntransactions support confidence  
## tdata 7198 0.01 0.35

inspect(sort(apprules, by = "lift")[1:20])

## lhs rhs support confidence lift   
## [1] {"Games",5,6.99} => {"9+"} 0.01472631 0.8153846 5.946442  
## [2] {"Games",6.99} => {"9+"} 0.01500417 0.8000000 5.834245  
## [3] {"4+",2.99,5} => {"Education"} 0.01861628 0.3592493 5.708337  
## [4] {5,6.99} => {"9+"} 0.01472631 0.6838710 4.987339  
## [5] {6.99} => {"9+"} 0.01500417 0.6506024 4.744717  
## [6] {"4+","Education",5} => {2.99} 0.01861628 0.3711911 3.911909  
## [7] {"Education",5} => {2.99} 0.01889414 0.3588391 3.781733  
## [8] {"Games",24,5} => {"9+"} 0.01028063 0.5174825 3.773900  
## [9] {"Games",24} => {"9+"} 0.01041956 0.5102041 3.720820  
## [10] {"4+",2.5} => {3} 0.01153098 0.3915094 2.368139  
## [11] {"4+","Games",0,11} => {38} 0.01014171 0.6186441 2.290638  
## [12] {2.5} => {3} 0.01667130 0.3726708 2.254189  
## [13] {"4+","Games",11,5} => {38} 0.01139205 0.5899281 2.184312  
## [14] {"4+","Games",0,2} => {40} 0.01166991 0.3544304 2.180504  
## [15] {"Games",0,11,5} => {38} 0.01125313 0.5869565 2.173309  
## [16] {"4+","Games",11} => {38} 0.01222562 0.5751634 2.129643  
## [17] {"Games",0,11} => {38} 0.01180884 0.5666667 2.098182  
## [18] {"4+","Shopping"} => {37} 0.01014171 0.8902439 1.954247  
## [19] {"4+","Shopping",0} => {37} 0.01014171 0.8902439 1.954247  
## [20] {"Games",11,5} => {38} 0.01417060 0.5230769 1.936784  
## count  
## [1] 106   
## [2] 108   
## [3] 134   
## [4] 106   
## [5] 108   
## [6] 134   
## [7] 136   
## [8] 74   
## [9] 75   
## [10] 83   
## [11] 73   
## [12] 120   
## [13] 82   
## [14] 84   
## [15] 81   
## [16] 88   
## [17] 85   
## [18] 73   
## [19] 73   
## [20] 102

#itemFrequencyPlot(tdata, support = 0.1) # items with a support of 0.1  
#itemFrequencyPlot(tdata, topN = 20) # 20 most frequent items

## Apriori - Cut Data

cutrules <- apriori(cutdata, parameter = list(support = 0.01, confidence = 0.35, minlen = 2))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.35 0.1 1 none FALSE TRUE 5 0.01 2  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 71   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[3256 item(s), 7198 transaction(s)] done [0.01s].  
## sorting and recoding items ... [37 item(s)] done [0.00s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 done [0.00s].  
## writing ... [213 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

summary(cutrules)

## set of 213 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2 3 4   
## 78 108 27   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 2.000 3.000 2.761 3.000 4.000   
##   
## summary of quality measures:  
## support confidence lift count   
## Min. :0.01000 Min. :0.3636 Min. :0.6035 Min. : 72.0   
## 1st Qu.:0.01306 1st Qu.:0.4725 1st Qu.:0.9229 1st Qu.: 94.0   
## Median :0.02237 Median :0.5736 Median :1.0646 Median : 161.0   
## Mean :0.04882 Mean :0.5919 Mean :1.1108 Mean : 351.4   
## 3rd Qu.:0.05404 3rd Qu.:0.6678 3rd Qu.:1.1735 3rd Qu.: 389.0   
## Max. :0.36580 Max. :1.0000 Max. :5.8342 Max. :2633.0   
##   
## mining info:  
## data ntransactions support confidence  
## cutdata 7198 0.01 0.35

inspect(sort(cutrules, by = "lift")[1:20])

## lhs rhs support confidence lift   
## [1] {"Games",6.99} => {"9+"} 0.01500417 0.8000000 5.834245  
## [2] {6.99} => {"9+"} 0.01500417 0.6506024 4.744717  
## [3] {"9+",6.99} => {"Games"} 0.01500417 1.0000000 1.863801  
## [4] {"9+",4.5} => {"Games"} 0.05404279 0.9306220 1.734494  
## [5] {"9+",4} => {"Games"} 0.03056405 0.9053498 1.687392  
## [6] {"9+",0,4.5} => {"Games"} 0.02236732 0.8944444 1.667067  
## [7] {"9+",0,4} => {"Games"} 0.01166991 0.8936170 1.665524  
## [8] {"9+",0.99} => {"Games"} 0.01806057 0.8904110 1.659549  
## [9] {"Shopping"} => {0} 0.01681023 0.9918033 1.655612  
## [10] {"4+","Shopping"} => {0} 0.01125313 0.9878049 1.648938  
## [11] {"9+"} => {"Games"} 0.12017227 0.8763931 1.633422  
## [12] {"Education",4} => {"4+"} 0.01486524 1.0000000 1.623731  
## [13] {"Education",2.99} => {"4+"} 0.02042234 0.9865772 1.601936  
## [14] {"9+",0} => {"Games"} 0.05459850 0.8488121 1.582017  
## [15] {"Education"} => {"4+"} 0.06001667 0.9536424 1.548459  
## [16] {"12+",2.99} => {"Games"} 0.01097527 0.8144330 1.517941  
## [17] {6.99} => {"Games"} 0.01875521 0.8132530 1.515742  
## [18] {"Education",4.5} => {"4+"} 0.01722701 0.9253731 1.502557  
## [19] {"Education",0} => {"4+"} 0.02361767 0.9239130 1.500186  
## [20] {"Social Networking"} => {0} 0.02042234 0.8802395 1.469379  
## count  
## [1] 108   
## [2] 108   
## [3] 108   
## [4] 389   
## [5] 220   
## [6] 161   
## [7] 84   
## [8] 130   
## [9] 121   
## [10] 81   
## [11] 865   
## [12] 107   
## [13] 147   
## [14] 393   
## [15] 432   
## [16] 79   
## [17] 135   
## [18] 124   
## [19] 170   
## [20] 147

#itemFrequencyPlot(tdata, support = 0.1) # items with a support of 0.1  
#itemFrequencyPlot(tdata, topN = 20) # 20 most frequent items

## cut data/label encoding

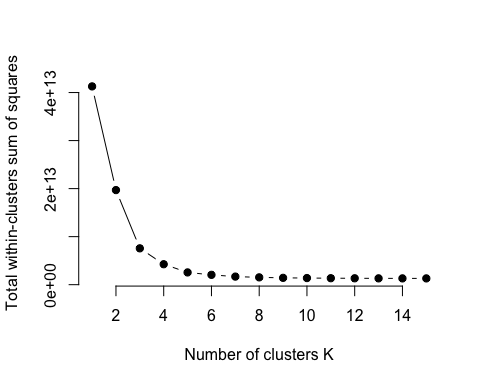
data\_cut <- data[, c(3, 4, 6, 8, 9)]  
  
data2 <- read.csv(file = "AppleStoreEncoded.csv", head = TRUE)  
data\_encoded <- data2[, c(3, 4, 6, 8, 9)]  
  
#for(i in 1:length(data\_cut$prime\_genre)) {  
# if(data\_cut$prime\_genre[i] == "Games") data\_cut$prime\_genre[i] <- 1  
#}

## Optimal K

k.max <- 15  
wss <- sapply(1:k.max,   
 function(k){kmeans(data\_encoded, k, nstart=50,iter.max = 15 )$tot.withinss})  
wss

## [1] 4.127955e+13 1.969658e+13 7.567165e+12 4.260592e+12 2.551816e+12  
## [6] 2.047457e+12 1.683293e+12 1.524413e+12 1.422352e+12 1.380299e+12  
## [11] 1.348339e+12 1.330340e+12 1.316888e+12 1.310290e+12 1.306882e+12

plot(1:k.max, wss,  
 type="b", pch = 19, frame = FALSE,   
 xlab="Number of clusters K",  
 ylab="Total within-clusters sum of squares")



## kmeans

# Cluster into k=5 clusters:  
myClusters = kmeans(data\_encoded, 3)  
  
# Summary of the clusters  
summary(myClusters)

## Length Class Mode   
## cluster 7197 -none- numeric  
## centers 15 -none- numeric  
## totss 1 -none- numeric  
## withinss 3 -none- numeric  
## tot.withinss 1 -none- numeric  
## betweenss 1 -none- numeric  
## size 3 -none- numeric  
## iter 1 -none- numeric  
## ifault 1 -none- numeric

# Centers (mean values) of the clusters  
myClusters$centers

## price rating\_count\_tot user\_rating cont\_rating prime\_genre  
## 1 1.7452779 6845.697 3.517588 7.089349 6.273533  
## 2 0.2313953 408677.872 4.267442 7.337209 4.872093  
## 3 0.0000000 2247896.250 4.250000 8.500000 6.750000

# Cluster assignments  
myClusters$cluster

## [1] 1 1 1 2 2 1 1 2 1 1 1 1 2 1 1 1 3 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [35] 1 2 1 1 1 1 1 1 1 1 1 2 1 2 1 1 1 1 1 2 1 1 1 1 1 2 1 1 1 1 1 1 1 2  
## [69] 1 1 1 1 1 2 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1  
## [103] 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 1 1 1 1 1 1 1 1  
## [137] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [171] 1 2 1 1 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [205] 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [239] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1  
## [273] 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1  
## [307] 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1  
## [341] 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1  
## [375] 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [409] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [443] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [477] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1  
## [511] 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [545] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [579] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1  
## [613] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [647] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [681] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 3 1 1 1 2 1 1  
## [715] 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [749] 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [783] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1  
## [817] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [851] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1  
## [885] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1  
## [919] 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [953] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [987] 1 1 1 1 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1021] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1  
## [1055] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1089] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1  
## [1123] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1157] 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1  
## [1191] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1  
## [1225] 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1  
## [1259] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1293] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1327] 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1361] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1395] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1429] 1 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1463] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1  
## [1497] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1531] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1565] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1599] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1633] 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1667] 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1701] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1735] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1769] 1 1 1 1 1 1 1 2 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1803] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1837] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1871] 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1905] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1  
## [1939] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [1973] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1  
## [2007] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1  
## [2041] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2075] 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2109] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2143] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2177] 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1  
## [2211] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2245] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2279] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2313] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2347] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2381] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2415] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2449] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2483] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2517] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2551] 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2585] 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2619] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2653] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2687] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2721] 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2755] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2789] 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2823] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2857] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2891] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2925] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2959] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [2993] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3027] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3061] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1  
## [3095] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3129] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3163] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3197] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3231] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3265] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3299] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3333] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3367] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3401] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3435] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3469] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3503] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3537] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3571] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3605] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3639] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3673] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3707] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3741] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3775] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3809] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3843] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3877] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3911] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1  
## [3945] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [3979] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4013] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4047] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4081] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4115] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4149] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4183] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4217] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4251] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4285] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4319] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4353] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4387] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4421] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4455] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4489] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4523] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4557] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4591] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1  
## [4625] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4659] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4693] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4727] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4761] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4795] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4829] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4863] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4897] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4931] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4965] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [4999] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5033] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5067] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5101] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5135] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5169] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5203] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5237] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5271] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5305] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5339] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5373] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5407] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5441] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5475] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5509] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5543] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5577] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5611] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5645] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5679] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5713] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5747] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5781] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5815] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5849] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5883] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5917] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5951] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [5985] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6019] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6053] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6087] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6121] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6155] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6189] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6223] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6257] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6291] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6325] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6359] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6393] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6427] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6461] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6495] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6529] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6563] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6597] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6631] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6665] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6699] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6733] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6767] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6801] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6835] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6869] 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6903] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6937] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [6971] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7005] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7039] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7073] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7107] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7141] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [7175] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

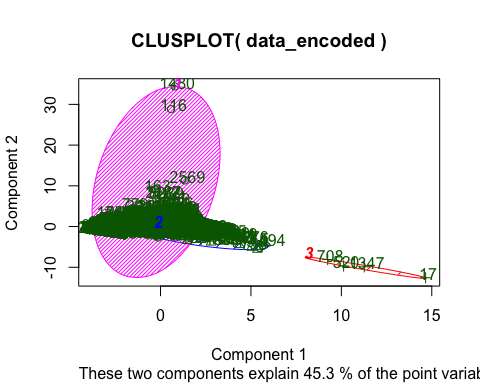
# Within-cluster sum of squares and total sum of squares across clusters  
myClusters$withinss

## [1] 3.381765e+12 3.362132e+12 8.232689e+11

myClusters$tot.withinss

## [1] 7.567165e+12

# Plotting a visual representation of k-means clusters  
library(cluster)  
clusplot(data\_encoded, myClusters$cluster, color=TRUE, shade=TRUE, labels=2, lines=0)



for (i in 1:25) {  
 myClusters = kmeans(data\_encoded, i)  
 print(myClusters$tot.withinss)  
}

## [1] 4.127955e+13  
## [1] 1.969658e+13  
## [1] 7.567165e+12  
## [1] 4.260592e+12  
## [1] 2.551816e+12  
## [1] 2.047457e+12  
## [1] 1.683293e+12  
## [1] 1.524413e+12  
## [1] 1.422352e+12  
## [1] 1.380299e+12  
## [1] 1.348339e+12  
## [1] 1.33034e+12

## Warning: did not converge in 10 iterations

## [1] 1.32838e+12

## Warning: did not converge in 10 iterations

## [1] 1.31029e+12

## Warning: did not converge in 10 iterations

## [1] 1.306882e+12

## Warning: did not converge in 10 iterations

## [1] 1.305852e+12

## Warning: did not converge in 10 iterations

## [1] 1.322032e+12

## Warning: did not converge in 10 iterations

## [1] 1.304073e+12

## Warning: did not converge in 10 iterations

## [1] 1.304145e+12

## Warning: did not converge in 10 iterations

## [1] 1.305477e+12

## Warning: did not converge in 10 iterations

## [1] 1.307254e+12

## Warning: did not converge in 10 iterations

## [1] 1.306069e+12

## Warning: did not converge in 10 iterations

## [1] 1.30192e+12

## Warning: did not converge in 10 iterations

## [1] 1.302887e+12

## Warning: did not converge in 10 iterations

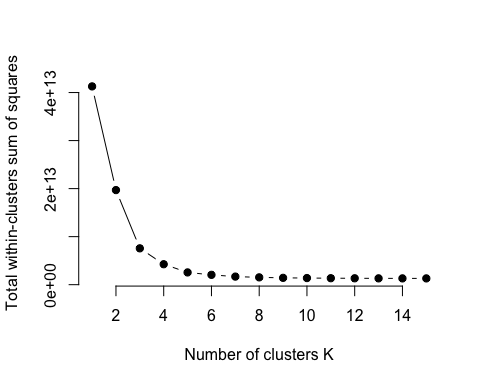
## [1] 1.301325e+12

## optimal k

k.max <- 15  
wss <- sapply(1:k.max,   
 function(k){kmeans(data\_encoded, k, nstart=50,iter.max = 15 )$tot.withinss})  
wss

## [1] 4.127955e+13 1.969658e+13 7.567165e+12 4.260592e+12 2.551816e+12  
## [6] 2.047457e+12 1.683293e+12 1.524413e+12 1.422352e+12 1.380299e+12  
## [11] 1.348339e+12 1.330340e+12 1.316888e+12 1.310290e+12 1.306882e+12

plot(1:k.max, wss,  
 type="b", pch = 19, frame = FALSE,   
 xlab="Number of clusters K",  
 ylab="Total within-clusters sum of squares")



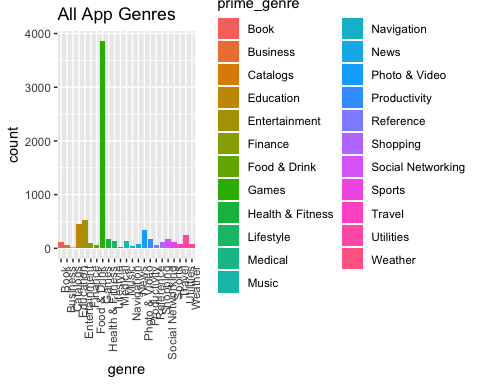
## New column based on cost

modified\_data$isFree <- (data$price == 0)  
split\_data <- split(modified\_data, modified\_data$isFree)  
free\_data <- data.frame(split\_data[2])  
cost\_data <- data.frame(split\_data[1])

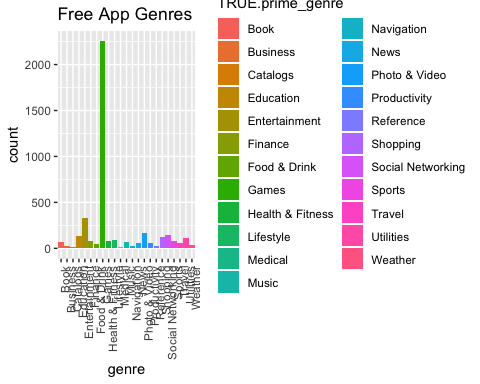
## GG PLOT GRAPHS

## Genre graphs

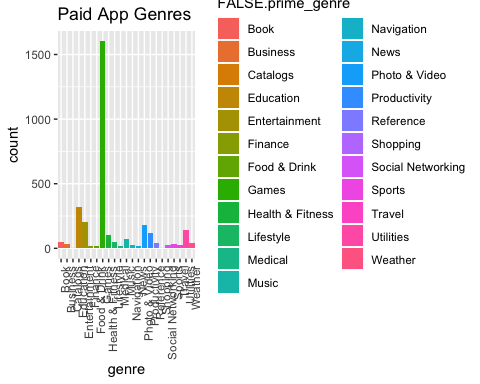
library(ggplot2)  
  
# All Apps  
ggplot(data, aes(prime\_genre, fill = prime\_genre)) + theme(axis.text.x = element\_text(angle = 90, hjust = 1)) + geom\_bar() + ggtitle("All App Genres") + xlab("genre") + ylab("count")



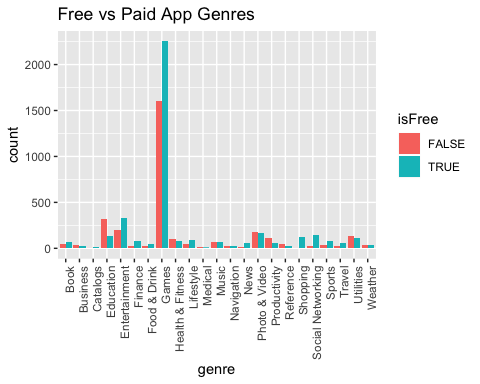
# Free Apps  
ggplot(free\_data, aes(TRUE.prime\_genre, fill = TRUE.prime\_genre)) + theme(axis.text.x = element\_text(angle = 90, hjust = 1)) + geom\_bar() + ggtitle("Free App Genres") + xlab("genre") + ylab("count")



# Paid Apps  
ggplot(cost\_data, aes(FALSE.prime\_genre, fill = FALSE.prime\_genre)) + theme(axis.text.x = element\_text(angle = 90, hjust = 1)) + geom\_bar() + ggtitle("Paid App Genres") + xlab("genre") + ylab("count")



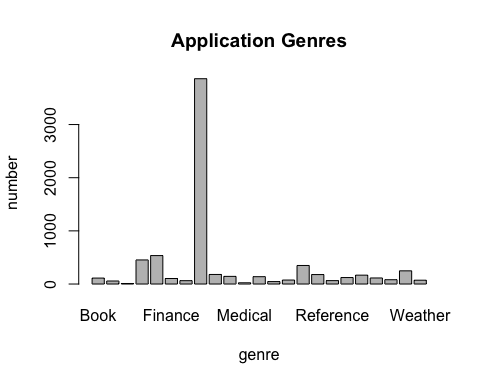
# Both  
ggplot(modified\_data, aes(x = prime\_genre, fill = isFree)) + theme(axis.text.x = element\_text(angle = 90, hjust = 1)) + geom\_bar(position="dodge") + ggtitle("Free vs Paid App Genres") + xlab("genre") + ylab("count")



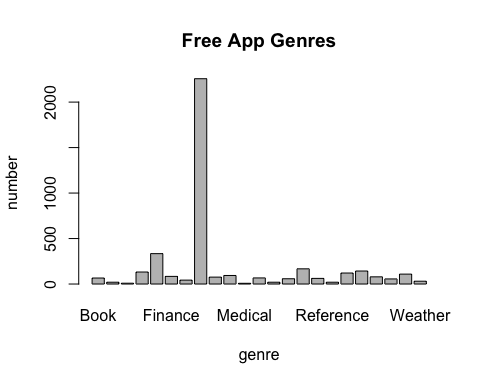
## USELESS ONCE THE GG PLOT GRAPHS ARE FUNCTIONING

## Bar Graph - Genres

tbl <- with(data, table(prime\_genre))  
barplot(tbl, beside = TRUE, legend = FALSE, main = "Application Genres", xlab = "genre", ylab = "number")



tbl <- with(free\_data, table(TRUE.prime\_genre))  
barplot(tbl, beside = TRUE, legend = FALSE, main = "Free App Genres", xlab = "genre", ylab = "number")



tbl <- with(cost\_data, table(FALSE.prime\_genre))  
barplot(tbl, beside = TRUE, legend = FALSE, main = "Paid App Genres", xlab = "genre", ylab = "number")

