DAPT 622 Assignment 2

Daniel Erb

February 1, 2019

Contents


```
# this code chunk loads the libraries used as well as reading in the necessary data
library(knitr)
library(data.table)
library(kableExtra)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#library(corrplot)
#library(stats)
#library(factoextra)
setwd(paste("G:/My Drive/vcu/Spring 2019/Statistics/Assignments",
                    "/Assignment 2/DAPT_622_Assignment_2/",
                    sep = ""))
# file path for candy data
candy.file <- "data/CandyBars.txt"

# 75 observations, not 69 like the read.table showed...
cd <- read.csv(candy.file)

# file path for demographics data
```

```
demo.file <- "data/Demographics.txt"

# 440 observations for this data set
dd <- read.csv(demo.file)
```

1

Part I

Consider the candy bar data again. A study was conducted to investigate the nutrition content of a variety of candy brands and products. The data set includes the following variables (most of which should be self-explanatory):

Perform a Factor Analysis on this data by carrying out the following steps:

Factor analysis requires you choose the number of factors that you are looking for. You can't fit the model until you decide how many factors you want to retain.

Question 1

How many factors should be retained (based on eigenvalues and a Scree plot)? You may use any "Factoring method" or "Rotation method" you wish. The defaults, of course, are fine.

```
# perform principal component analysis first to allow for analysis
# of the eigenvalues and the scree plot

# remove non-continuous data
cd_sub <- cd[,c("Oz.pkg","Calories","Total.fat.g"
                  ,"Saturated.fat.g","Cholesterol.g"
                  ,"Sodium.mg","Carbohydrate.g"
                  ,"Dietary.fiber.g","Sugars.g"
                  ,"Protein.g","Vitamin.A..RDI"
                  ,"Vitamin.C..RDI","Calcium..RDI"
                  ,"Iron..RDI")]

cd_sub <- cd_sub %>%
  rename(oz = "Oz.pkg",
         kcal = "Calories",
         tot.fat = "Total.fat.g",
         sat.fat = "Saturated.fat.g",
         cholest = "Cholesterol.g",
         sodium = "Sodium.mg",
         carb = "Carbohydrate.g",
         diet.fib = "Dietary.fiber.g",
         sugar = "Sugars.g",
         protein = "Protein.g",
         vit.a = "Vitamin.A..RDI",
         vit.c = "Vitamin.C..RDI",
         calcium = "Calcium..RDI",
```

Table 1.1: Variance/Covariance Matrix for Candy Data

	oz	kcal	tot.fat	sat.fat	cholest	sodium	carb	diet.fib	sugar	protein	vit.a	vit.c	calcium	iron
oz	1.33	5.80	0.88	0.75	1.28	-7.25	-0.98	0.06	-0.01	0.05	-0.19	-0.80	1.11	-0.37
kcal	5.80	3843.54	286.58	136.39	63.67	333.97	224.96	25.92	189.85	72.15	-27.77	-91.14	49.26	18.09
tot.fat	0.88	286.58	32.80	15.57	13.42	15.70	-10.48	2.60	0.16	7.87	-1.23	-7.49	7.95	2.32
sat.fat	0.75	136.39	15.57	11.61	8.74	-22.86	-3.24	1.08	2.72	1.73	-1.09	-4.54	4.40	0.28
cholest	1.28	63.67	13.42	8.74	29.42	-10.27	-13.80	-0.38	1.79	0.17	1.07	-4.35	7.11	-1.62
sodium	-7.25	333.97	15.70	-22.86	-10.27	2305.41	31.72	-6.14	-26.92	28.24	16.01	-13.77	-34.59	-20.49
carb	-0.98	224.96	-10.48	-3.24	-13.80	31.72	82.63	-0.52	50.10	-4.39	-4.51	-8.17	-9.70	-3.23
diet.fib	0.06	25.92	2.60	1.08	-0.38	-6.14	-0.52	0.79	-0.46	0.91	-0.29	0.49	0.40	0.93
sugar	-0.01	189.85	0.16	2.72	1.79	-26.92	50.10	-0.46	53.95	-3.72	-3.18	-9.83	-3.24	-4.77
protein	0.05	72.15	7.87	1.73	0.17	28.24	-4.39	0.91	-3.72	4.41	0.50	2.00	3.51	2.36
vit.a	-0.19	-27.77	-1.23	-1.09	1.07	16.01	-4.51	-0.29	-3.18	0.50	4.36	5.68	2.91	3.27
vit.c	-0.80	-91.14	-7.49	-4.54	-4.35	-13.77	-8.17	0.49	-9.83	2.00	5.68	34.48	11.58	9.92
calcium	1.11	49.26	7.95	4.40	7.11	-34.59	-9.70	0.40	-3.24	3.51	2.91	11.58	15.78	5.08
iron	-0.37	18.09	2.32	0.28	-1.62	-20.49	-3.23	0.93	-4.77	2.36	3.27	9.92	5.08	8.84

Table 1.2: Correlation Matrix for Candy Data

	oz	kcal	tot.fat	sat.fat	cholest	sodium	carb	diet.fib	sugar	protein	vit.a	vit.c	calcium	iron
oz	1.000	0.081	0.134	0.192	0.204	-0.131	-0.093	0.054	-0.001	0.019	-0.080	-0.118	0.243	-0.108
kcal	0.081	1.000	0.807	0.646	0.189	0.112	0.399	0.470	0.417	0.554	-0.214	-0.250	0.200	0.098
tot.fat	0.134	0.807	1.000	0.798	0.432	0.057	-0.201	0.509	0.004	0.654	-0.103	-0.223	0.349	0.136
sat.fat	0.192	0.646	0.798	1.000	0.473	-0.140	-0.105	0.355	0.109	0.242	-0.153	-0.227	0.325	0.028
cholest	0.204	0.189	0.432	0.473	1.000	-0.039	-0.280	-0.079	0.045	0.015	0.095	-0.137	0.330	-0.100
sodium	-0.131	0.112	0.057	-0.140	-0.039	1.000	0.073	-0.144	-0.076	0.280	0.160	-0.049	-0.181	-0.144
carb	-0.093	0.399	-0.201	-0.105	-0.280	0.073	1.000	-0.064	0.750	-0.230	-0.237	-0.153	-0.269	-0.120
diet.fib	0.054	0.470	0.509	0.355	-0.079	-0.144	-0.064	1.000	-0.070	0.486	-0.156	0.093	0.114	0.351
sugar	-0.001	0.417	0.004	0.109	0.045	-0.076	0.750	-0.070	1.000	-0.241	-0.207	-0.228	-0.111	-0.218
protein	0.019	0.554	0.654	0.242	0.015	0.280	-0.230	0.486	-0.241	1.000	0.115	0.162	0.420	0.378
vit.a	-0.080	-0.214	-0.103	-0.153	0.095	0.160	-0.237	-0.156	-0.207	0.115	1.000	0.463	0.350	0.527
vit.c	-0.118	-0.250	-0.223	-0.227	-0.137	-0.049	-0.153	0.093	-0.228	0.162	0.463	1.000	0.496	0.568
calcium	0.243	0.200	0.349	0.325	0.330	-0.181	-0.269	0.114	-0.111	0.420	0.350	0.496	1.000	0.430
iron	-0.108	0.098	0.136	0.028	-0.100	-0.144	-0.120	0.351	-0.218	0.378	0.527	0.568	0.430	1.000

```
         iron = "Iron..RDI")

#Get variance-covariance matrix
covmat <- cov(cd_sub)
kable(covmat,caption = "Variance/Covariance Matrix for Candy Data", digits = 2) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"),
                full_width = TRUE, font_size = 8)
```

```
#Get correlation matrix
corrmat <- cor(cd_sub)
kable(corrmat,caption = "Correlation Matrix for Candy Data", digits = 3) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"),
                full_width = TRUE, font_size = 8)
```

```
# calculate the eigenvalues and vectors and display the values
ev <- eigen(corrmat)
ev$values
```

```
##  [1] 3.694308981 2.919698481 1.785308252 1.457684213 1.261691263
##  [6] 0.884675063 0.568202493 0.428438986 0.354417068 0.267713492
## [11] 0.162155673 0.152111650 0.060728326 0.002866059
```

```
ev$vectors
```

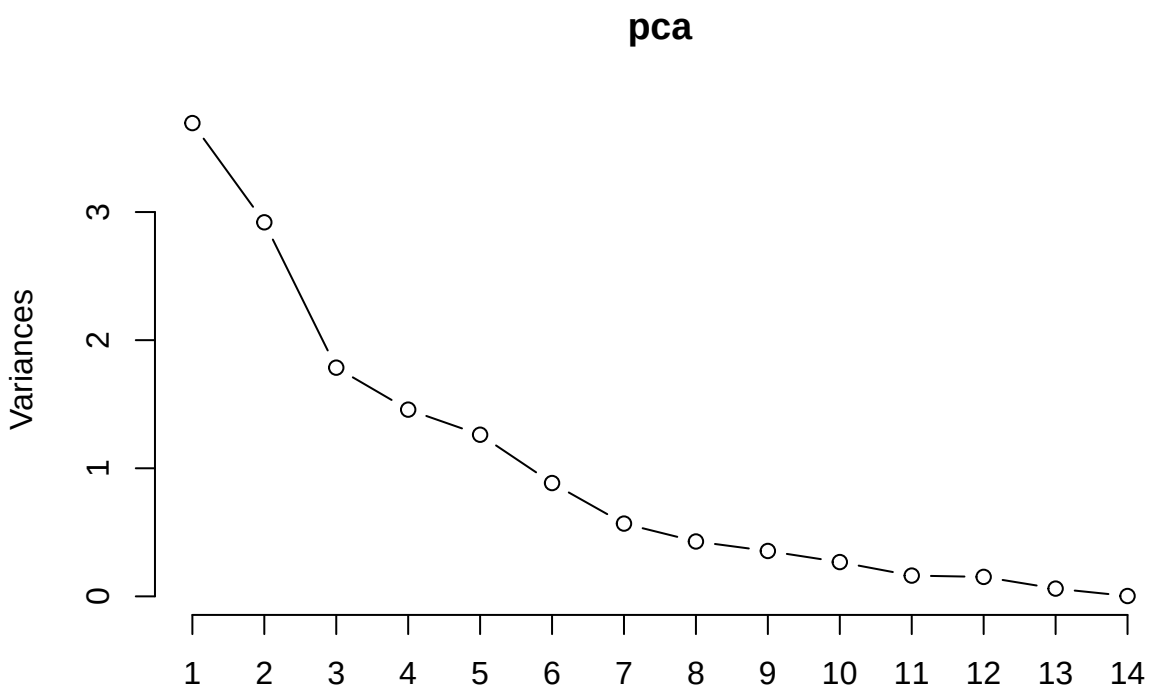
```
##               [,1]         [,2]        [,3]        [,4]         [,5]
##  [1,] -0.113317669  0.053686539 -0.31479891 -0.22725401  0.107033419
##  [2,] -0.402836701  0.284073580  0.25837994 -0.01585576 -0.110727490
##  [3,] -0.486633265  0.118536793 -0.06746888  0.13511138 -0.050603309
##  [4,] -0.402953791  0.176695640 -0.18575154 -0.11183965  0.014600317
##  [5,] -0.218347101  0.044914231 -0.45119326 -0.22086892 -0.342005105
##  [6,]  0.011505264  0.012867318  0.14223866  0.48068412 -0.617539040
##  [7,]  0.103343004  0.342723490  0.47912344 -0.27583793 -0.126148031
##  [8,] -0.318917895 -0.003342341  0.23296724  0.15167987  0.472640353
##  [9,]  0.016771089  0.380444122  0.27928444 -0.43686736 -0.186794863
## [10,] -0.378882228 -0.140404113  0.20068771  0.32823909 -0.072050365
## [11,] -0.014183410 -0.405814738  0.07181759 -0.15558865 -0.415726502
## [12,] -0.009298491 -0.446008539  0.23631870 -0.23069808  0.009375739
## [13,] -0.295812179 -0.281099565 -0.06512421 -0.36862151 -0.113181078
## [14,] -0.184055124 -0.380022355  0.32237363 -0.16279476  0.100375583
##               [,6]         [,7]          [,8]         [,9]         [,10]
##  [1,]  0.838655887 -0.289398358  0.0657840828  0.044124905 -0.072285484
##  [2,]  0.029047686 -0.005754639 -0.1193632619  0.008884305 -0.103451905
##  [3,] -0.108964683 -0.024126316 -0.1087569891  0.013726192  0.053898093
##  [4,] -0.230548505 -0.074454426  0.0002012423  0.677985055  0.045554985
##  [5,] -0.251920954 -0.010686959  0.4361946041 -0.436644024 -0.312748944
##  [6,]  0.247673005  0.001228510  0.3366950146  0.246044055 -0.087631567
##  [7,]  0.117733825 -0.020509253  0.0189175792  0.075529124 -0.281739169
##  [8,]  0.007165189 -0.208733364  0.5635470341 -0.138821018  0.297098616
##  [9,] -0.016123264  0.016596450  0.0931241649 -0.228307178  0.372913995
## [10,]  0.214399737  0.218292716 -0.2548454181 -0.387472448  0.048742593
## [11,] -0.073700038 -0.551044730 -0.1194791883  0.003058837  0.466383436
## [12,]  0.053544582  0.394448631  0.4549524673  0.240133036 -0.002101877
## [13,]  0.157777129  0.459838585 -0.2045179607  0.029507988  0.121599048
## [14,] -0.112465080 -0.383881847 -0.1064021875 -0.033871761 -0.576377972
##              [,11]       [,12]        [,13]        [,14]
##  [1,] -0.13872214 -0.06297655 -0.008413763 -0.014228504
##  [2,]  0.06152000 -0.29772247 -0.275700845  0.695278779
##  [3,] -0.21490915 -0.08895875 -0.571074679 -0.562286844
##  [4,] -0.07835265  0.03937079  0.480438153 -0.015390226
##  [5,]  0.09096198 -0.11354884  0.132242302  0.003444134
##  [6,]  0.03126087  0.34660067 -0.052175461  0.017408731
##  [7,]  0.41183009 -0.27744157  0.135026918 -0.431010662
##  [8,]  0.32810680  0.14326108  0.024849851 -0.008389245
##  [9,] -0.45702928  0.37131318  0.063095806  0.010437474
## [10,] -0.16978565 -0.16312194  0.551014291 -0.115752612
## [11,]  0.15022729 -0.26276157  0.017991310 -0.010168995
## [12,] -0.34532255 -0.37739283 -0.083922915 -0.003321935
## [13,]  0.46995769  0.39104796 -0.113725160  0.001819807
## [14,] -0.19431391  0.37220813 -0.003615515  0.014168783
```

```
# perform pca
pca <- prcomp(cd_sub, scale = TRUE)
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     1.9221 1.7087 1.3362 1.2073 1.12325 0.94057 0.75379
```

```
## Proportion of Variance 0.2639 0.2086 0.1275 0.1041 0.09012 0.06319 0.04059
## Cumulative Proportion  0.2639 0.4724 0.5999 0.7041 0.79419 0.85738 0.89797
##                            PC8     PC9    PC10    PC11    PC12    PC13
## Standard deviation     0.6546 0.59533 0.51741 0.40269 0.39001 0.24643
## Proportion of Variance 0.0306 0.02532 0.01912 0.01158 0.01087 0.00434
## Cumulative Proportion  0.9286 0.95389 0.97301 0.98459 0.99546 0.99980
##                            PC14
## Standard deviation     0.05354
## Proportion of Variance 0.00020
## Cumulative Proportion  1.00000
```

```
# create a screeplot
screeplot(pca, npcs = 14, type = "lines")
```



for factor choice-1.bb

Question 2

Provide the rotated factor loadings matrix. You should notice that (as compared with Principal Component Analysis) fewer variables load onto each factor. Interpret (as best as possible) your factors. Can you give an overall name to each factor?

Question 3

Plot factor scores for your first two factors. Color the points by Brand. Are there any natural groupings of observations?

2

Part II

Consider the demographics data (Demographics.jmp or Demographics.txt). Each of the 440 rows contains demographic information on particular localities in the United States. The following is a brief description of the variables.

Question 1

Perform a linear discriminant analysis using the highlighted variables for discriminating the variable Pop_Size_Group.

- Pct_Age18_to_34
- Pct_65_or_over
- Num_physicians
- Num_hospital_beds
- Num_serious_crimes
- Pct_High_Sch_grads
- Pct_Bachelors
- Pct_below_poverty
- Pct_unemployed
- Per_cap_1990income
- Total_personal_income

Question 2

Show a plot of the two discriminant functions with points colored by Pop_Size_Group. Do both discriminant functions appear necessary to describe group separation?

Question 3

Give the scoring coefficients for the two discriminant functions. Which variables appear most important for describing group separation?

Question 4

Provide a classification matrix and the percent of observations misclassified.