Class10-HalloweenMiniPoject

Mahsa Naeimi

1. Importing Candy Data

```
candy_file <- "candy-data.csv"
candy <- read.csv(candy_file, row.names = 1)
head(candy)
```

```
             chocolate fruity caramel peanutyalmondy nougat crispedricewafer
100 Grand            1      0       1              0      0                 1
3 Musketeers         1      0       0              0      1                 0
One dime             0      0       0              0      0                 0
One quarter          0      0       0              0      0                 0
Air Heads            0      1       0              0      0                 0
Almond Joy           1      0       0              1      0                 0
             hard bar pluribus sugarpercent pricepercent winpercent
100 Grand       0   1        0        0.732        0.860   66.97173
3 Musketeers    0   1        0        0.604        0.511   67.60294
One dime        0   0        0        0.011        0.116   32.26109
One quarter     0   0        0        0.011        0.511   46.11650
Air Heads       0   0        0        0.906        0.511   52.34146
Almond Joy      0   1        0        0.465        0.767   50.34755
```

What is in the dataset?

Q1. How many different candy types are in this dataset?

```
num_candy_types <- nrow(candy)
```

85 types of candies

Q2. How many fruity candy types are in the dataset?

```
fruity_candy_types = sum(candy$fruity)
fruity_candy_types
```

```
[1] 38
```

2. What is your favorite candy?

We can find the `winpercent` value for favorite candy by using its name to access the corresponding row of the dataset.

```
candy["Twix",]$winpercent
```

```
[1] 81.64291
```

Q3. What is your favorite candy in the dataset and what is it's `winpercent` value?

```
candy["Sugar Daddy",]$winpercent
```

```
[1] 32.231
```

Q4. What is the `winpercent` value for "Kit Kat"?

```
candy["Kit Kat",]$winpercent
```

```
[1] 76.7686
```

Q5. What is the `winpercent` value for "Tootsie Roll Snack Bars"?

```
candy["Tootsie Roll Snack Bars",]$winpercent
```

```
[1] 49.6535
```

Install `skimr` to get a quick overview of a given dataset"

```
#install.packages("skimr")

#Now checking the candy data

library("skimr")
skim(candy)
```

Table 1: Data summary

Name	candy
Number of rows	85
Number of columns	12
_____	
Column type frequency:	
numeric	12
_____	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
chocolate	0	1	0.44	0.50	0.00	0.00	0.00	1.00	1.00	
fruity	0	1	0.45	0.50	0.00	0.00	0.00	1.00	1.00	
caramel	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
peanutyalmondy	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
nougat	0	1	0.08	0.28	0.00	0.00	0.00	0.00	1.00	
crispedricewafer	0	1	0.08	0.28	0.00	0.00	0.00	0.00	1.00	
hard	0	1	0.18	0.38	0.00	0.00	0.00	0.00	1.00	
bar	0	1	0.25	0.43	0.00	0.00	0.00	0.00	1.00	
pluribus	0	1	0.52	0.50	0.00	0.00	1.00	1.00	1.00	
sugarpercent	0	1	0.48	0.28	0.01	0.22	0.47	0.73	0.99	
pricepercent	0	1	0.47	0.29	0.01	0.26	0.47	0.65	0.98	
winpercent	0	1	50.32	14.71	22.45	39.14	47.83	59.86	84.18	

Q6. Is there any variable/column that looks to be on a different scale to the majority of the other columns in the dataset?

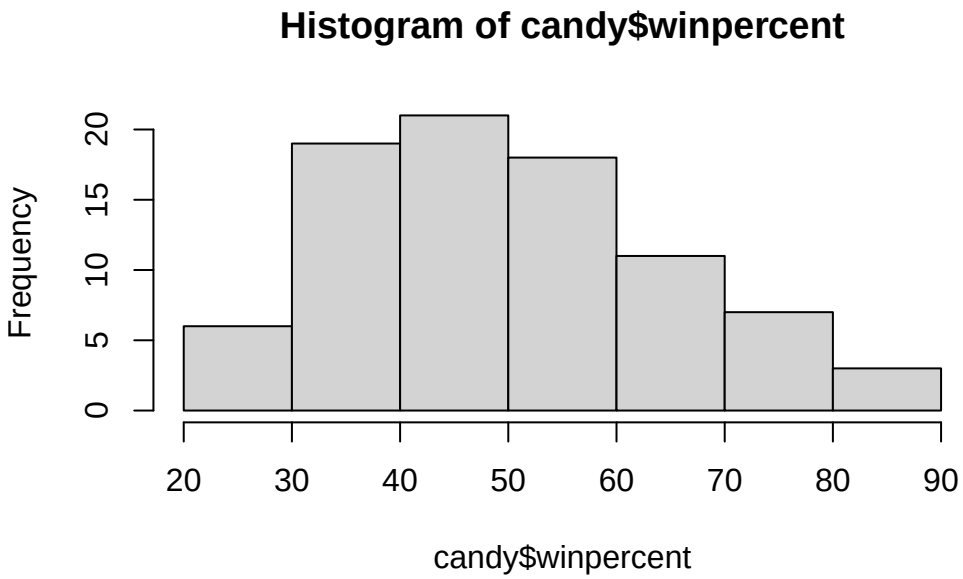
`winpercent` seems to be in a different scale compared to other columns

Q7. What do you think a zero and one represent for the `candy$chocolate` column?

0 are candies that don't contain chocolate and 1 shows candies that contain choclate

Q8. Plot a histogram of `winpercent` values:

```
hist(candy$winpercent)
```



Q9. Is the distribution of `winpercent` values symmetrical?

roughly symmetric, with longer right tail

Q10. Is the center of the distribution above or below 50%?

Center of the distribution is above 50%

We want to compare chocolate and fruity candy:

Q11. On average is chocolate candy higher or lower ranked than fruit candy?

```
table(as.logical(candy$chocolate))
```

```
FALSE  TRUE
   48    37
```

```r
length(candy$chocolate)
```

```
[1] 85
```

```r
candy$winpercent[ as.logical(candy$chocolate)]
```

```
 [1] 66.97173 67.60294 50.34755 56.91455 38.97504 55.37545 62.28448 56.49050
 [9] 59.23612 57.21925 76.76860 71.46505 66.57458 55.06407 73.09956 60.80070
[17] 64.35334 47.82975 54.52645 70.73564 66.47068 69.48379 81.86626 84.18029
[25] 73.43499 72.88790 65.71629 34.72200 37.88719 76.67378 59.52925 48.98265
[33] 43.06890 45.73675 49.65350 81.64291 49.52411
```

```r
# Chocolate Candies:
winpercent_chocolate <- candy$winpercent[as.logical(candy$chocolate) ]
mean (winpercent_chocolate)
```

```
[1] 60.92153
```

```r
#Fruity Candies:
winpercent_fruit <- candy$winpercent[as.logical(candy$fruity)]
mean(winpercent_fruit)
```

```
[1] 44.11974
```

Statistical Test:

Q12. Is this difference statistically significant?

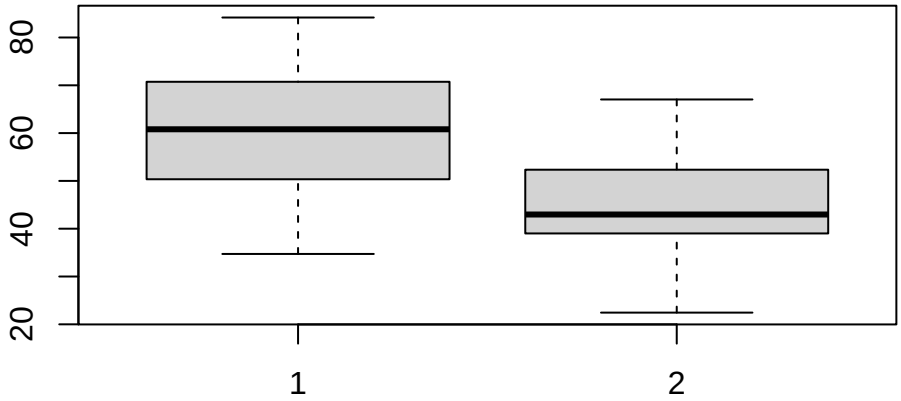
```r
t.test(winpercent_chocolate, winpercent_fruit)
```

```
    Welch Two Sample t-test

data:  winpercent_chocolate and winpercent_fruit
t = 6.2582, df = 68.882, p-value = 2.871e-08
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
 11.44563 22.15795
sample estimates:
mean of x mean of y
 60.92153  44.11974
```

An alternative way to look at the significance:

```
boxplot(winpercent_chocolate, winpercent_fruit)
```



3. Overall Candy Rankings

Q13. What are the five least liked candy types in this set?

```
head(candy[order(candy$winpercent), ], 5)
```

```
                  chocolate fruity caramel peanutyalmondy nougat
Nik L Nip                 0      1       0              0      0
Boston Baked Beans        0      0       0              1      0
Chiclets                  0      1       0              0      0
```

```
Super Bubble               0      1       0              0      0
Jawbusters                 0      1       0              0      0
                   crispedricewafer hard bar pluribus sugarpercent pricepercent
Nik L Nip                         0    0   0        1        0.197        0.976
Boston Baked Beans                0    0   0        1        0.313        0.511
Chiclets                          0    0   0        1        0.046        0.325
Super Bubble                      0    0   0        0        0.162        0.116
Jawbusters                        0    1   0        1        0.093        0.511
                   winpercent
Nik L Nip            22.44534
Boston Baked Beans   23.41782
Chiclets             24.52499
Super Bubble         27.30386
Jawbusters           28.12744
```

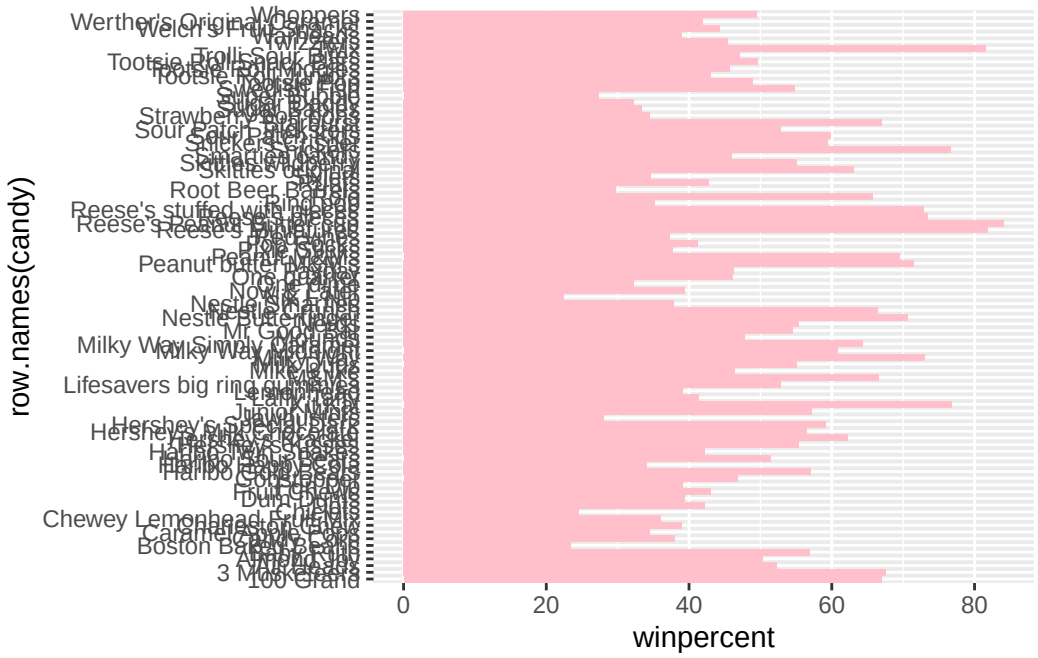
Q14. What are the top 5 all time favorite candy types out of this set?

```
head(candy[order(-candy$winpercent), ], 5)
```

```
                         chocolate fruity caramel peanutyalmondy nougat
Reese's Peanut Butter cup         1      0       0              1      0
Reese's Miniatures                1      0       0              1      0
Twix                              1      0       1              0      0
Kit Kat                           1      0       0              0      0
Snickers                          1      0       1              1      1
                         crispedricewafer hard bar pluribus sugarpercent
Reese's Peanut Butter cup                0    0   0        0        0.720
Reese's Miniatures                       0    0   0        0        0.034
Twix                                     1    0   1        0        0.546
Kit Kat                                  1    0   1        0        0.313
Snickers                                 0    0   1        0        0.546
                         pricepercent winpercent
Reese's Peanut Butter cup        0.651   84.18029
Reese's Miniatures               0.279   81.86626
Twix                             0.906   81.64291
Kit Kat                          0.511   76.76860
Snickers                         0.651   76.67378
```

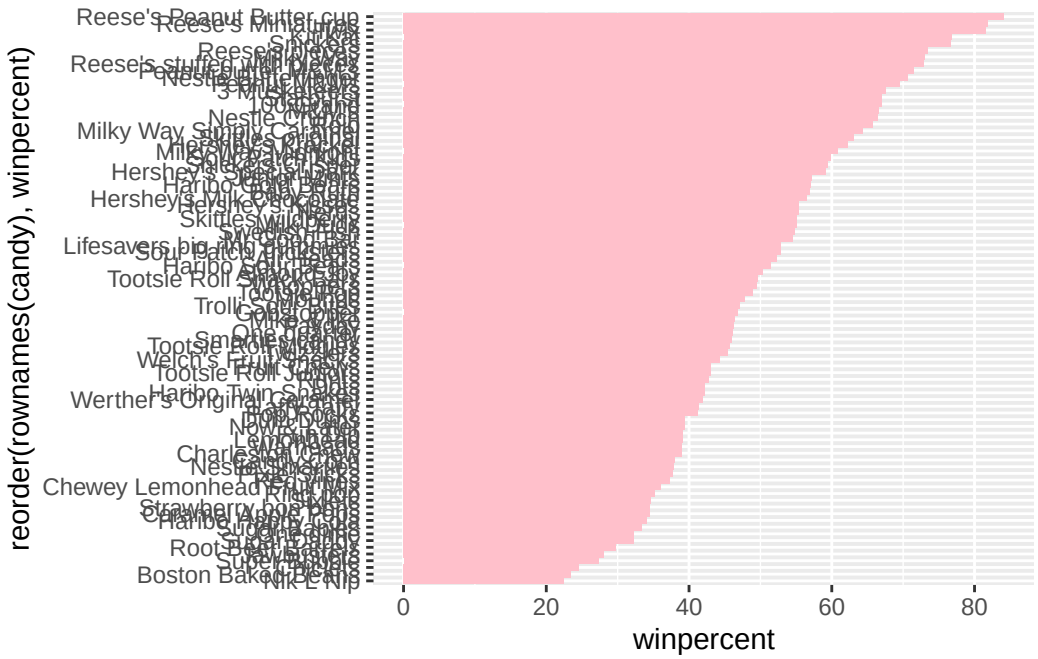
Q15. Make a first barplot of candy ranking based on `winpercent` values.

```
library(ggplot2)

ggplot(candy, aes(y=row.names(candy), x=winpercent) )+
  geom_col(fill = "pink")
```



Q16. This is quite ugly, use the reorder() function to get the bars sorted by winpercent ? To order the data we can use **reorder**command:

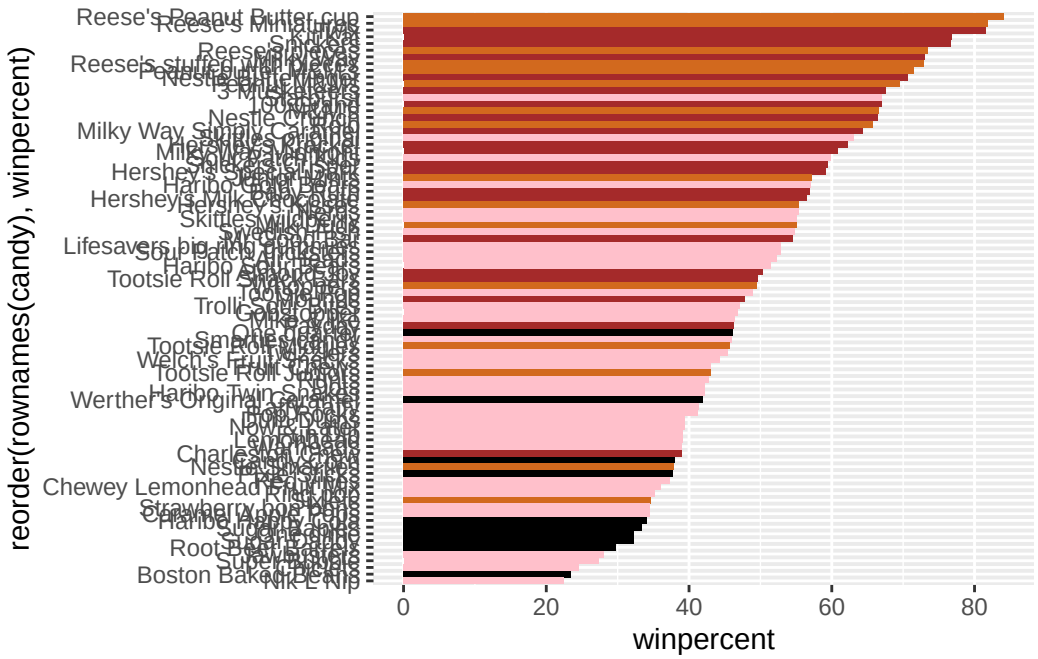
```
ggplot(candy) + aes(winpercent, reorder(rownames(candy), winpercent)) +
  geom_col(fill = "pink")
```



To add color for better analyzing:

```
my_cols=rep("black", nrow(candy))
my_cols[as.logical(candy$chocolate)] = "chocolate"
my_cols[as.logical(candy$bar)] = "brown"
my_cols[as.logical(candy$fruity)] = "pink"
```

```
ggplot(candy) +
  aes(winpercent, reorder(rownames(candy),winpercent)) +
  geom_col(fill=my_cols)
```



Now, for the first time, using this plot we can answer questions like:

Q17. What is the worst ranked chocolate candy?

based on the graph, Sixlet is the worst ranked chocolate

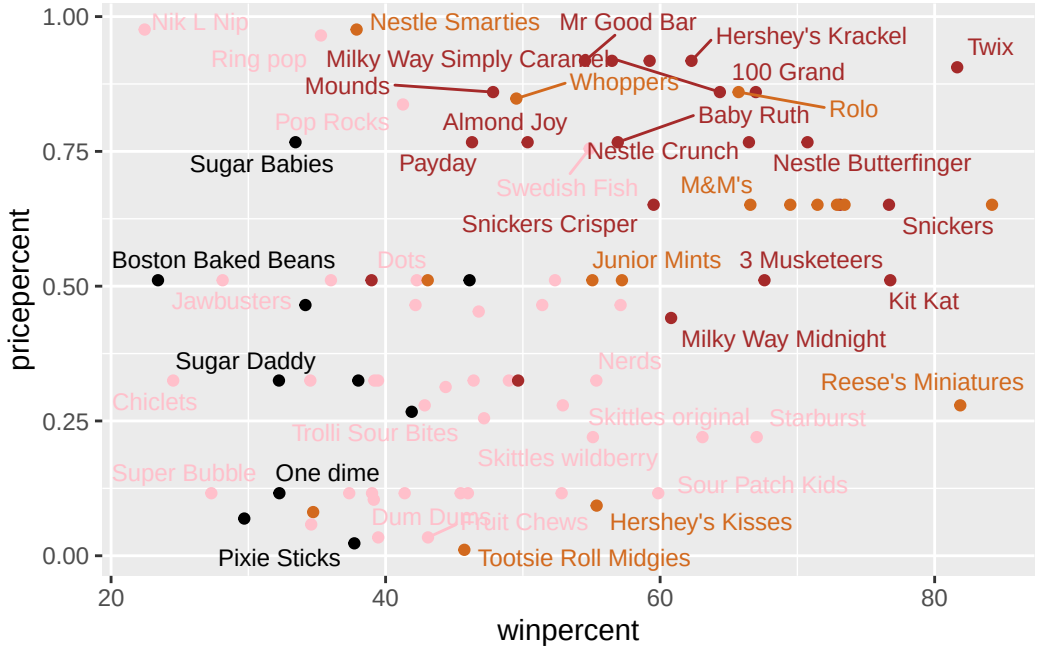
Q18. What is the best ranked fruity candy?

based on the graph, Starburst is the best fruity

4. Taking a look at pricepercent

```
#install.packages("ggrepel")
library(ggrepel)

# How about a plot of price vs win
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
  geom_text_repel(col=my_cols, size=3.3, max.overlaps = 10)
```

```
Warning: ggrepel: 40 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Q19. Which candy type is the highest ranked in terms of `winpercent` for the least money - i.e. offers the most bang for your buck?

```
candy_win_more_than_80 <- candy[candy$winpercent>80,]
rownames(candy_win_more_than_80)[order(candy_win_more_than_80$pricepercent)]
```

```
[1] "Reese's Miniatures"        "Reese's Peanut Butter cup"
[3] "Twix"
```

Q20. What are the top 5 most expensive candy types in the dataset and of these which is the least popular?

```
ord <- order(candy$pricepercent, decreasing = TRUE)
head( candy[ord,c(11,12)], n=5 )
```

```
                              pricepercent winpercent
Nik L Nip                            0.976   22.44534
```

```
Nestle Smarties                 0.976   37.88719
Ring pop                        0.965   35.29076
Hershey's Krackel               0.918   62.28448
Hershey's Milk Chocolate        0.918   56.49050
```

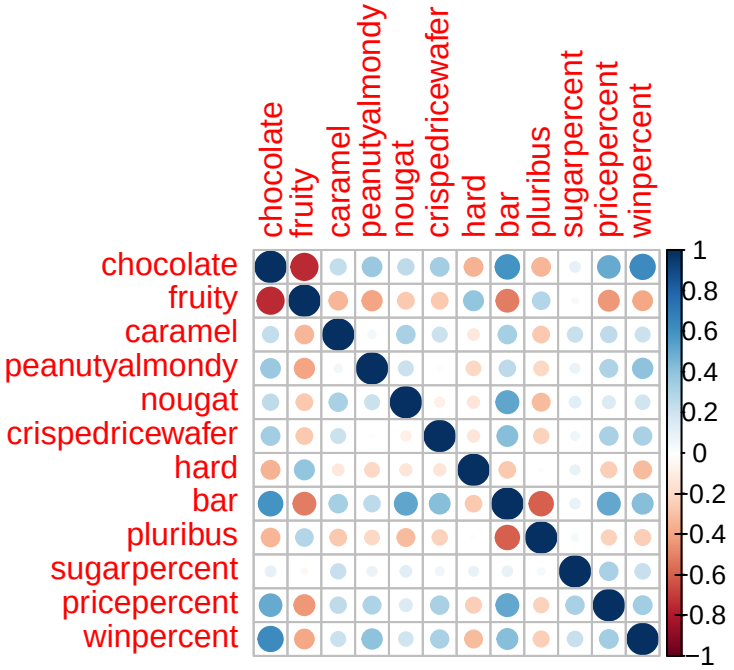
5. Exploring the Correlation Structure

To see see how the variables interact with one another we use `corrplot`:

```
#install.packages("corrplot")
library(corrplot)
```

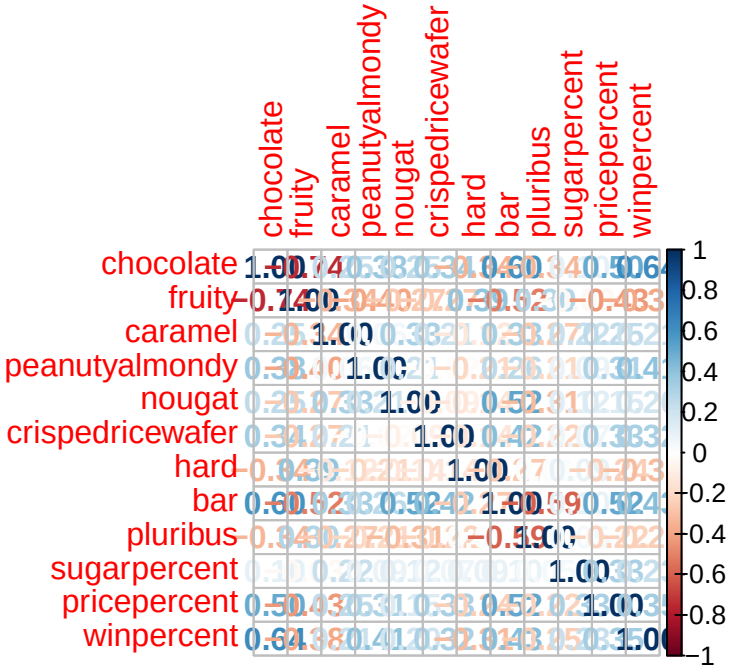
```
corrplot 0.92 loaded
```

```
cij <- cor(candy)
corrplot(cij)
```



Q22. Examining this plot what two variables are anti-correlated (i.e. have minus values)?

```
library(corrplot)
cij <- cor(candy)
corrplot(cij, method = 'number')
```



Chocolate and fruity

Q23. Similarly, what two variables are most positively correlated?

Chocolate and `winpercent`

6. Principal Component Analysis

Let's apply PCA using the `prcom()` function to our candy dataset remembering to set the `scale=TRUE`argument

```
pca <- prcomp(candy, scale = TRUE)
summary(pca)
```

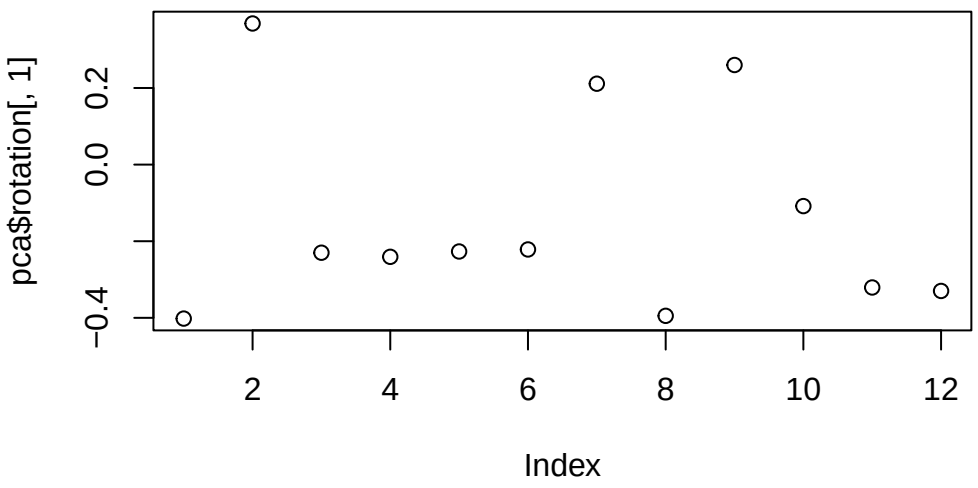
```
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7
```

```
Standard deviation     2.0788 1.1378 1.1092 1.07533 0.9518 0.81923 0.81530
Proportion of Variance 0.3601 0.1079 0.1025 0.09636 0.0755 0.05593 0.05539
Cumulative Proportion  0.3601 0.4680 0.5705 0.66688 0.7424 0.79830 0.85369
                           PC8     PC9    PC10    PC11    PC12
Standard deviation     0.74530 0.67824 0.62349 0.43974 0.39760
Proportion of Variance 0.04629 0.03833 0.03239 0.01611 0.01317
Cumulative Proportion  0.89998 0.93832 0.97071 0.98683 1.00000
```

```
pca$rotation[,1]
```

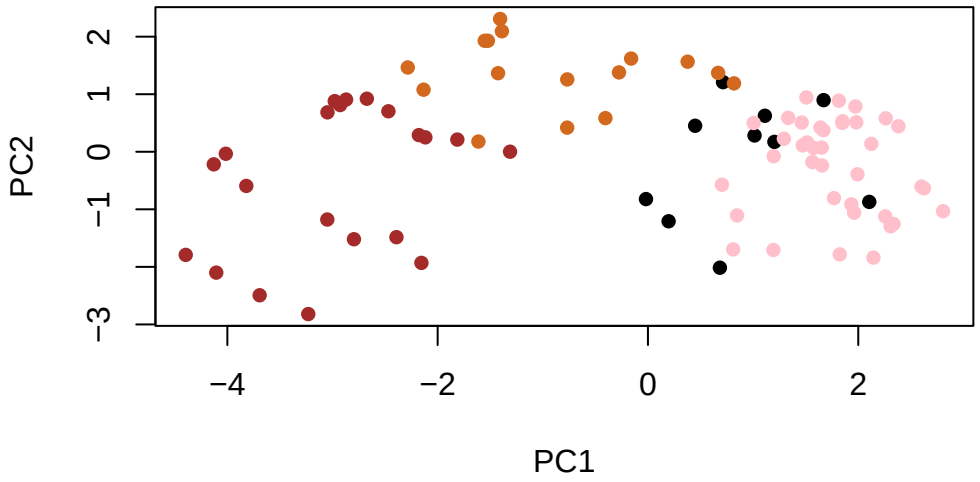
```
     chocolate          fruity         caramel   peanutyalmondy
    -0.4019466       0.3683883      -0.2299709      -0.2407155
        nougat crispedricewafer            hard             bar
    -0.2268102      -0.2215182       0.2111587      -0.3947433
      pluribus     sugarpercent   pricepercent      winpercent
     0.2600041      -0.1083088      -0.3207361      -0.3298035
```

```
plot(pca$rotation[,1])
```



to change the plotting character and add some color:

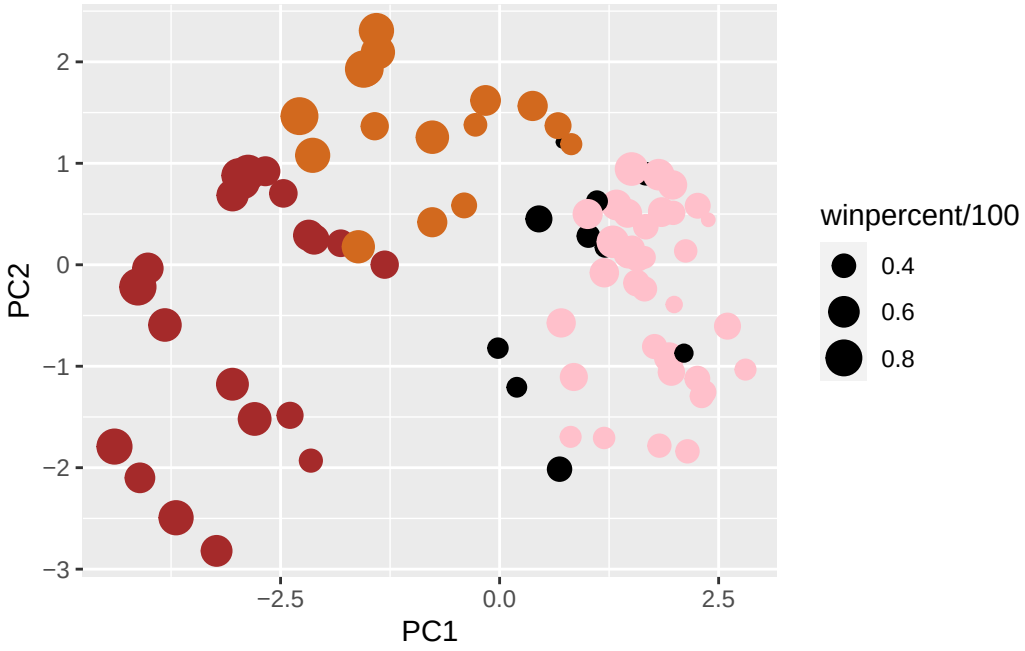
```
plot(pca$x[,1:2], col=my_cols, pch=16)
```



We can use ggplot as well:

```
my_data <- cbind(candy, pca$x[,1:3])
```

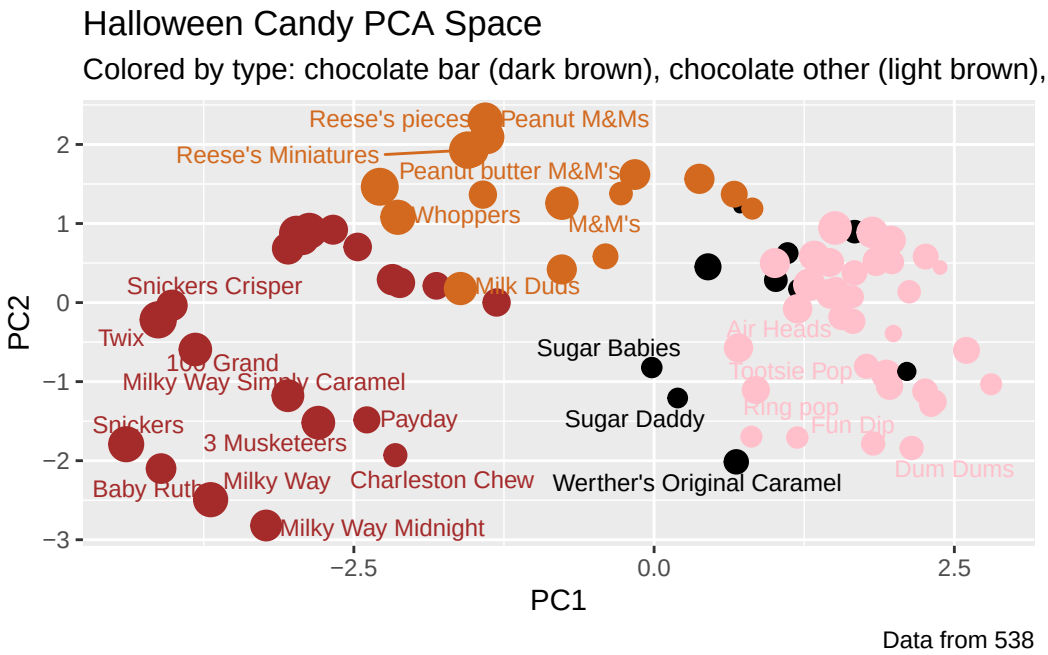
```
p <- ggplot(my_data) +
        aes(x=PC1, y=PC2,
            size=winpercent/100,
            text=rownames(my_data),
            label=rownames(my_data)) +
        geom_point(col=my_cols)
p
```



We can add. labels to the plot :

```
library(ggrepel)

p + geom_text_repel(size=3.3, col=my_cols, max.overlaps = 7)  +
  theme(legend.position = "none") +
  labs(title="Halloween Candy PCA Space",
       subtitle="Colored by type: chocolate bar (dark brown), chocolate other (light brown
       caption="Data from 538")
```

```
Warning: ggrepel: 59 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



we can use `plotly` to have more interactive plot where we can see each data points information by leaving mousing over the point:

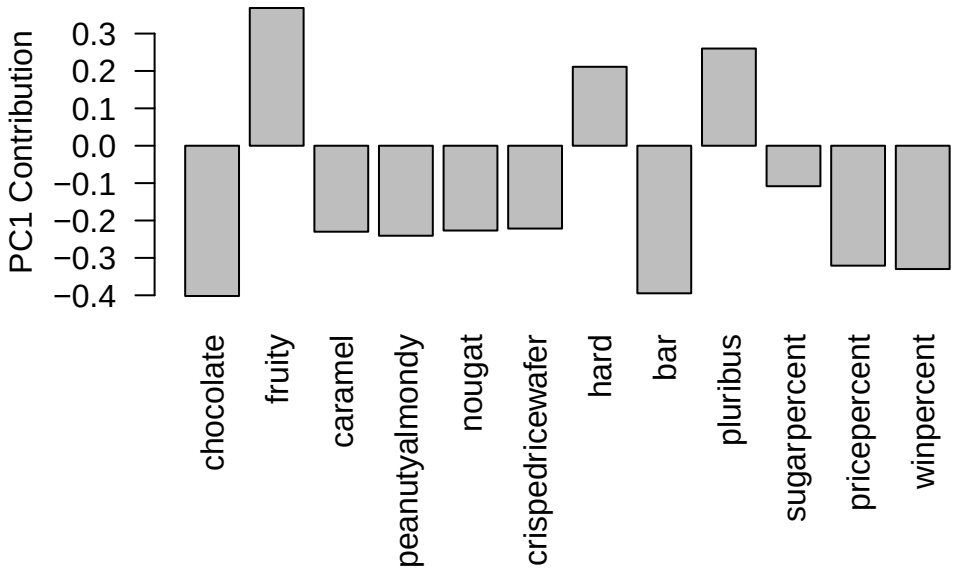
```
#install.packages("plotly")
library(plotly)
```

```
Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

    last_plot

The following object is masked from 'package:stats':

    filter

The following object is masked from 'package:graphics':

    layout
```

```
# to get pdf, we have to comment ggplotly for p
#ggplotly(p)
```

To see correlation better :

```
par(mar=c(8,4,2,2))
barplot(pca$rotation[,1], las=2, ylab="PC1 Contribution")
```



Q24. What original variables are picked up strongly by PC1 in the positive direction? Do these make sense to you? fruity, hard, and pluribus