Class 09: Halloween Mini-Project

Andres Vasquez (PID:16278181

In today's class we will examine some data

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
```

Data Exploration

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

There are 85 candy in this dataset

```
nrow(candy)
```

```
[1] 85
```

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

There are 4554.4649233 fruity candy types in the dataset

```
sum(candy[,2])
```

```
[1] 38
```

```
# sum(candy$fruity) works as well
```

How many chocolate candys are in the dataset?

```
sum(candy[, 1])
```

```
[1] 37
```

```
# sum(candy$chocolate) works too
```

2. What is your favorite candy?

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

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

```
[1] 76.67378
```

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
```

```
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	

```
# skimr::skim(candy) works too
```

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

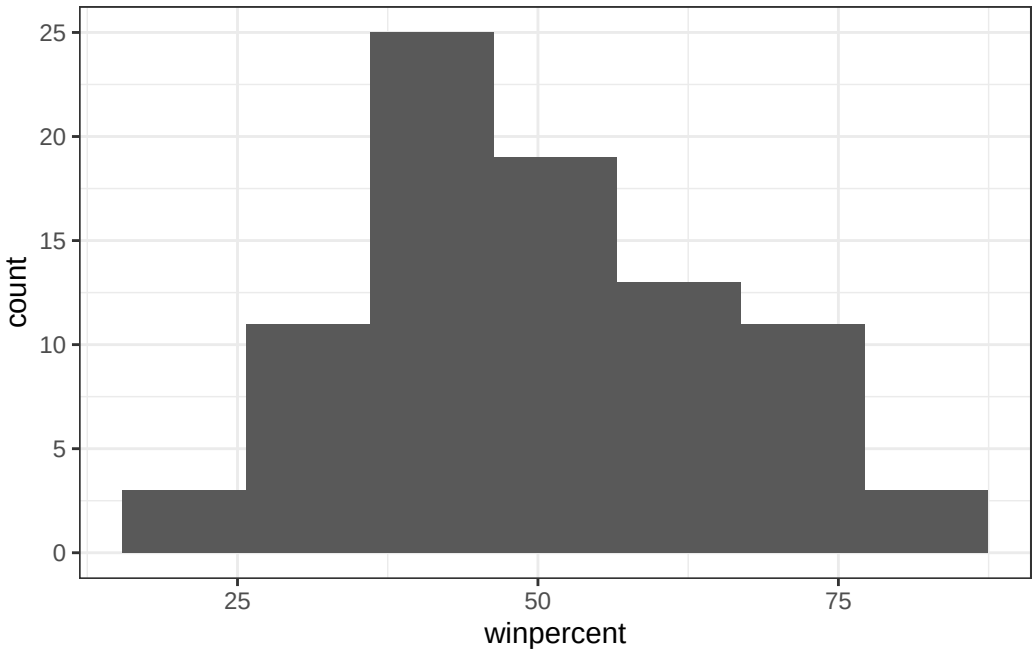
The winpercent column is scaled significantly higher than the other columns in the dataset.

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

Represents if the candy contains that trait.

Q8. Plot a histogram of winpercent values

```
library(ggplot2)
c <- ggplot(candy) + aes(winpercent) +geom_histogram(bins = 7) +theme_bw()
c
```



Q9. Is the distribution of winpercent values symmetrical?

The distribution is not symmetrical.

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

It is below 50%

```
summary(candy$winpercent)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  22.45   39.14   47.83   50.32   59.86   84.18
```

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

-first find all chocolate candy -find their winpercent values - calculate the mean of these values

- then do the same for fruity candy and compare On average, the fruit candy is ranked higher than the chocolate candy

```
chocolate.inds <- candy$chocolate == 1
chocolate.win <- candy[chocolate.inds, ]$winpercent
c <- mean(chocolate.win)
```

```
fruity.inds <- candy$fruity == 1
fruity.win <- candy[fruity.inds, ]$winpercent
f <- mean(fruity.win)
```

```
c > f
```

```
[1] TRUE
```

Chocolate candy is ranker higher than fruit candy, on average. >Q12. Is this difference statistically significant?

```
t.test(chocolate.win, fruity.win)
```

```
    Welch Two Sample t-test

data:  chocolate.win and fruity.win
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
```

Yes, the difference is statistically significant based on the low p-value.

3. Overall Candy Rankings

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

The `order()` function returns the indices that make the input sorted.

```
head(candy[order(candy$winpercent),], n=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?

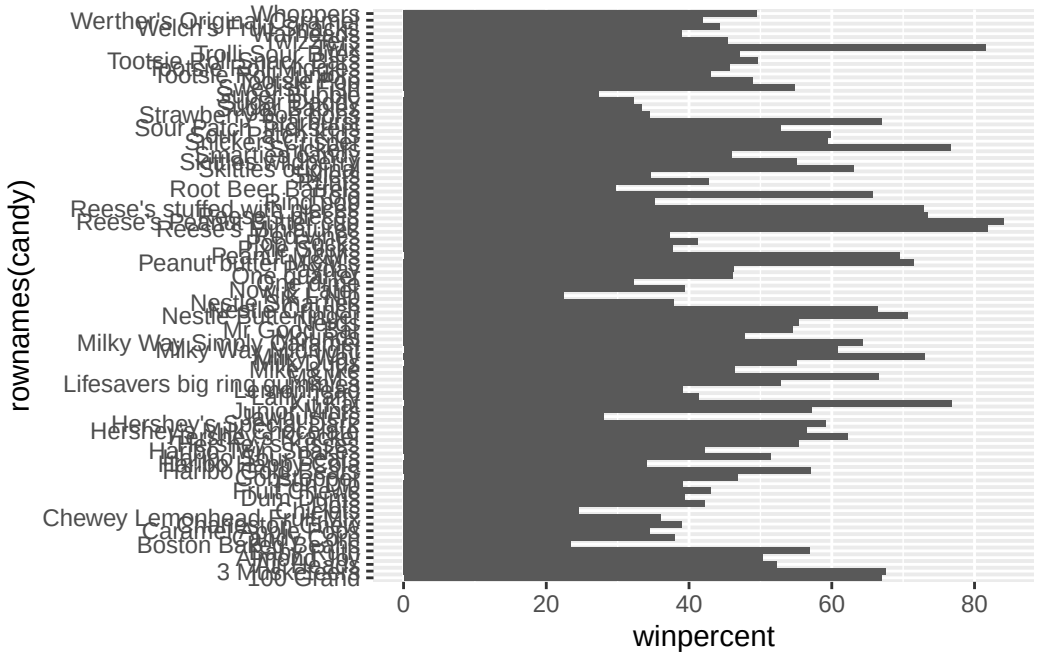
```
head(candy[order(candy$winpercent, decreasing = T),], n=5)
```

```
                        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
```

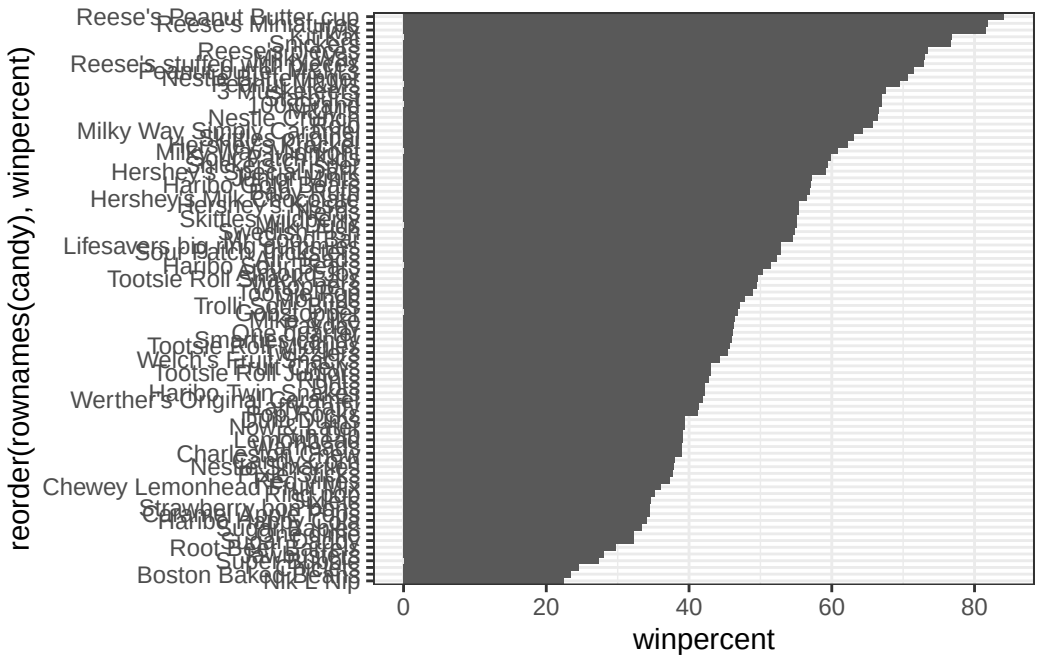
Q15. Make a first barplot of candy ranking based on winpercent values.

```
ggplot(candy) + aes(winpercent, rownames(candy)) +geom_col()
```



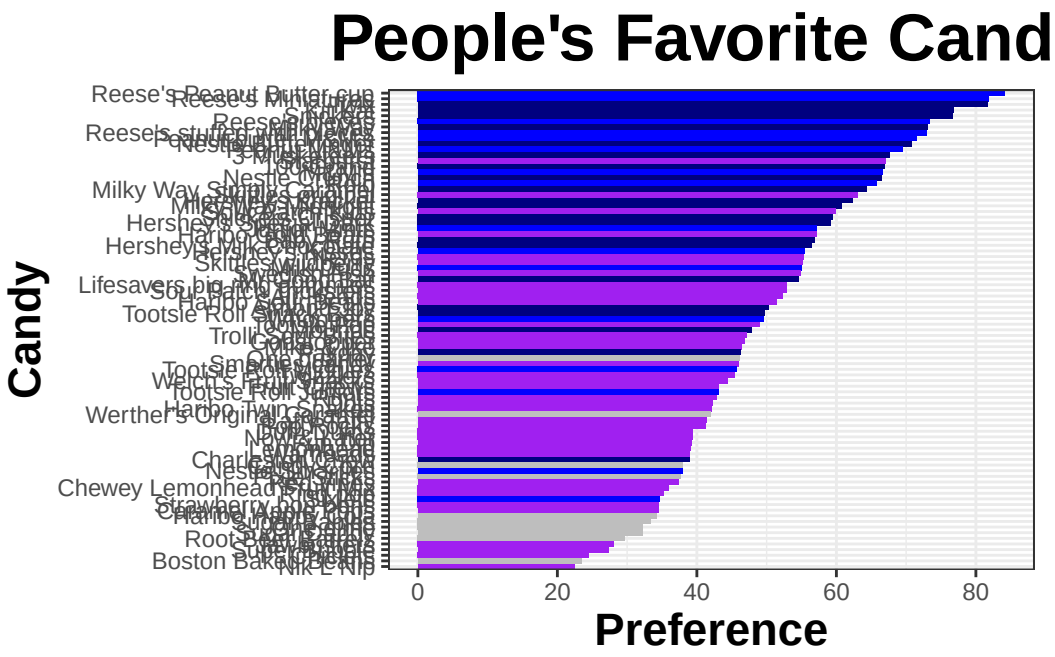
Q16. This is quite ugly, use the reorder() function to get the bars sorted by winpercent?

```
ggplot(candy) + aes(winpercent, reorder(rownames(candy), winpercent)) +geom_col() +theme_b
```



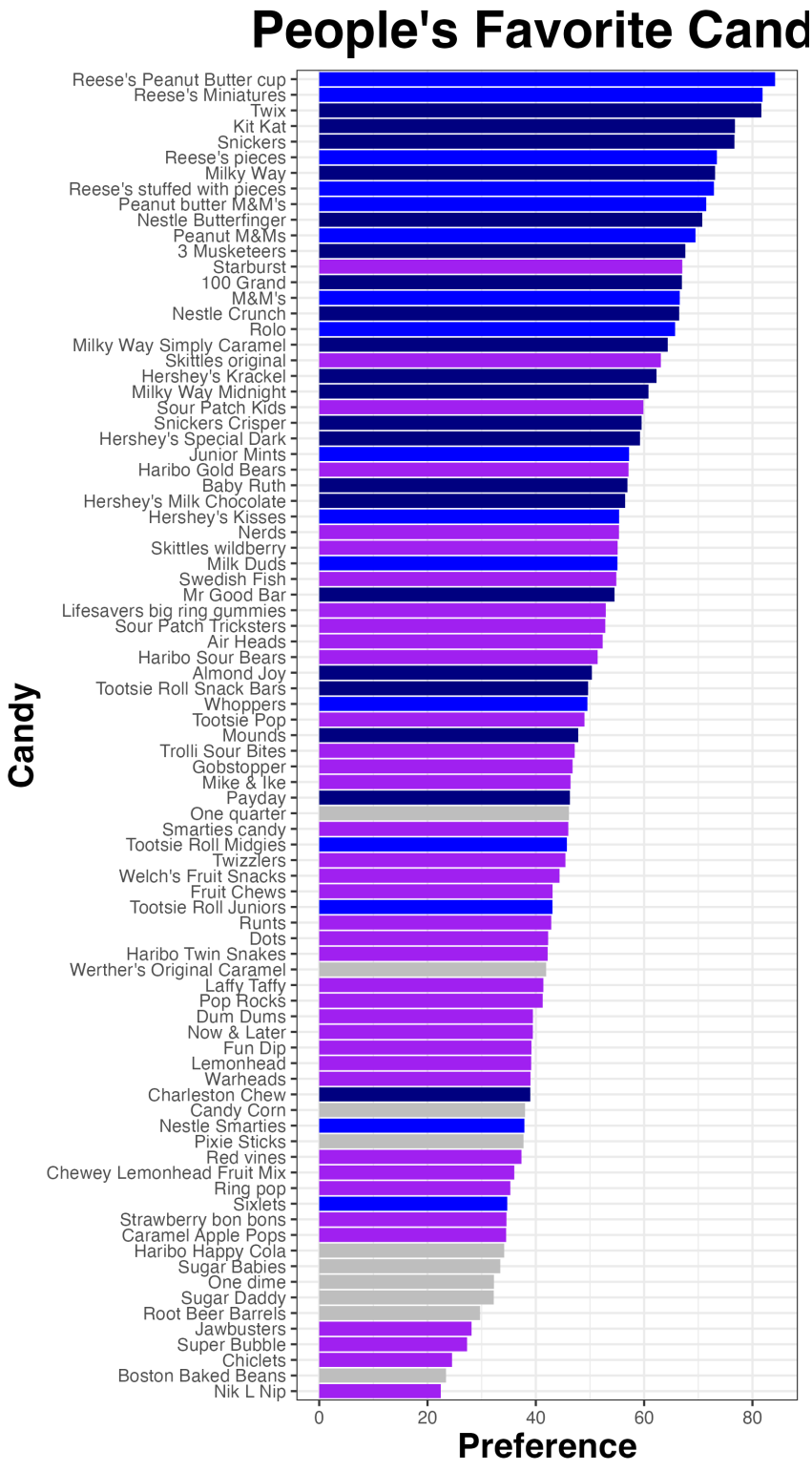

```
my_cols=rep("grey", nrow(candy))
my_cols[as.logical(candy$chocolate)] = "blue"
my_cols[as.logical(candy$bar)] = "navy"
my_cols[as.logical(candy$fruity)] = "purple"
```

```
gg <- ggplot(candy) +
  aes(winpercent, reorder(rownames(candy),winpercent)) +
  geom_col(fill=my_cols) + theme_bw()
gg + labs(title = "People's Favorite Candy", x = "Preference", y = "Candy") + theme(axis.t
plot.title = element_text(hjust = 0.5, size = 25, face = "bold"))
```




```
ggsave("mybarplot.png", height = 10)
```

```
Saving 5.5 x 10 in image
```



> Q17. What is the worst

ranked chocolate candy?

Sixlets

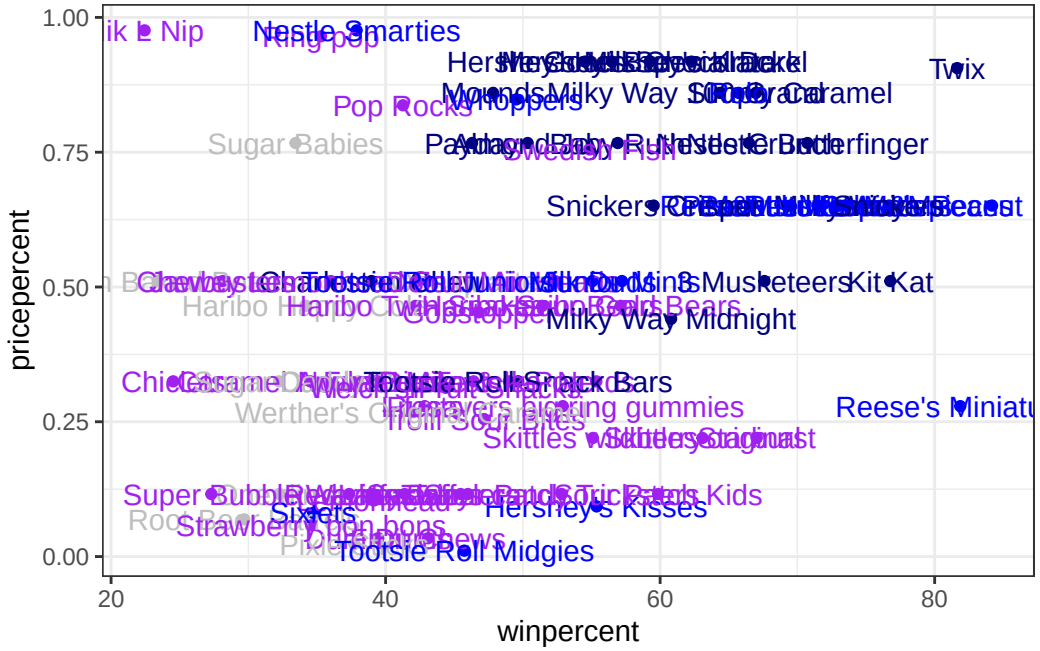
Q18. What is the best ranked fruity candy?

Starburst

4. Taking a look at pricepercent

Plot of winpercent vs pricepercent

```
ggplot(candy) +
  aes(winpercent, pricepercent, label = rownames(candy)) + geom_point(col = my_cols) +
  theme_bw() +geom_text(col = my_cols)
```

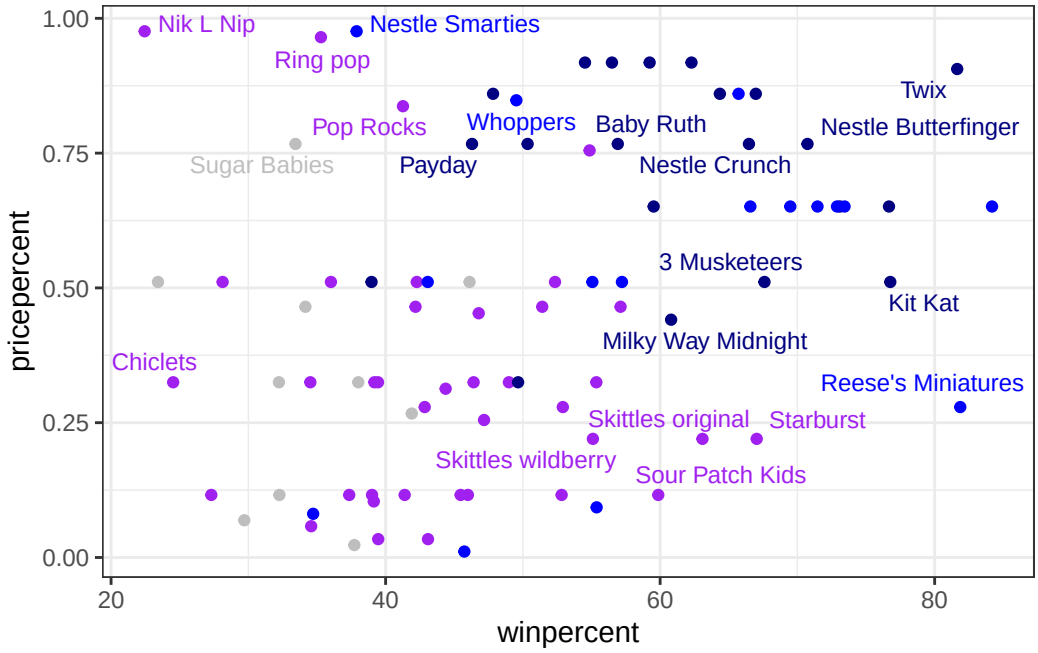


There are too many labels in this above plot to be readable. We can use the `ggrepel` package to do a better job of placing labels so they minimize text overlap.

```
library(ggrepel)
ggplot(candy) +
  aes(winpercent, pricepercent, label = rownames(candy)) + geom_point(col = my_cols) +
```

```
    theme_bw() +geom_text_repel(col = my_cols, max.overlaps = 5, size = 3.3)
```

```
Warning: ggrepel: 65 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



5. Exploring the Correltation Structure

```
  library(corrplot)
```

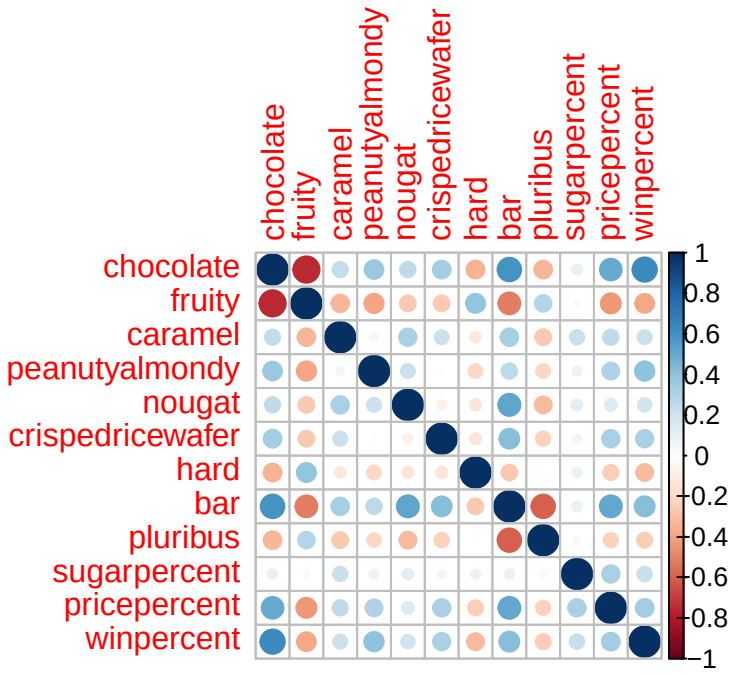
```
corrplot 0.92 loaded
```

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

```
                  chocolate      fruity     caramel peanutyalmondy      nougat
chocolate         1.0000000 -0.74172106  0.24987535     0.37782357  0.25489183
fruity           -0.7417211  1.00000000 -0.33548538    -0.39928014 -0.26936712
```

```
caramel           0.2498753 -0.33548538  1.00000000     0.05935614  0.32849280
peanutyalmondy    0.3778236 -0.39928014  0.05935614     1.00000000  0.21311310
nougat            0.2548918 -0.26936712  0.32849280     0.21311310  1.00000000
crispedricewafer  0.3412098 -0.26936712  0.21311310    -0.01764631 -0.08974359
hard             -0.3441769  0.39067750 -0.12235513    -0.20555661 -0.13867505
bar               0.5974211 -0.51506558  0.33396002     0.26041960  0.52297636
pluribus         -0.3396752  0.29972522 -0.26958501    -0.20610932 -0.31033884
sugarpercent      0.1041691 -0.03439296  0.22193335     0.08788927  0.12308135
pricepercent      0.5046754 -0.43096853  0.25432709     0.30915323  0.15319643
winpercent        0.6365167 -0.38093814  0.21341630     0.40619220  0.19937530
                 crispedricewafer        hard         bar    pluribus
chocolate              0.34120978 -0.34417691  0.59742114 -0.33967519
fruity                -0.26936712  0.39067750 -0.51506558  0.29972522
caramel                0.21311310 -0.12235513  0.33396002 -0.26958501
peanutyalmondy        -0.01764631 -0.20555661  0.26041960 -0.20610932
nougat                -0.08974359 -0.13867505  0.52297636 -0.31033884
crispedricewafer       1.00000000 -0.13867505  0.42375093 -0.22469338
hard                  -0.13867505  1.00000000 -0.26516504  0.01453172
bar                    0.42375093 -0.26516504  1.00000000 -0.59340892
pluribus              -0.22469338  0.01453172 -0.59340892  1.00000000
sugarpercent           0.06994969  0.09180975  0.09998516  0.04552282
pricepercent           0.32826539 -0.24436534  0.51840654 -0.22079363
winpercent             0.32467965 -0.31038158  0.42992933 -0.24744787
                 sugarpercent pricepercent winpercent
chocolate          0.10416906    0.5046754  0.6365167
fruity            -0.03439296   -0.4309685 -0.3809381
caramel            0.22193335    0.2543271  0.2134163
peanutyalmondy     0.08788927    0.3091532  0.4061922
nougat             0.12308135    0.1531964  0.1993753
crispedricewafer   0.06994969    0.3282654  0.3246797
hard               0.09180975   -0.2443653 -0.3103816
bar                0.09998516    0.5184065  0.4299293
pluribus           0.04552282   -0.2207936 -0.2474479
sugarpercent       1.00000000    0.3297064  0.2291507
pricepercent       0.32970639    1.0000000  0.3453254
winpercent         0.22915066    0.3453254  1.0000000
```

```
corrplot(cij)
```



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

Chocolate and fruity are anti-correlated > Q23. Similarly, what two variables are most positively correlated?

Chocolate and winpercent

6. Principal Component Analysis

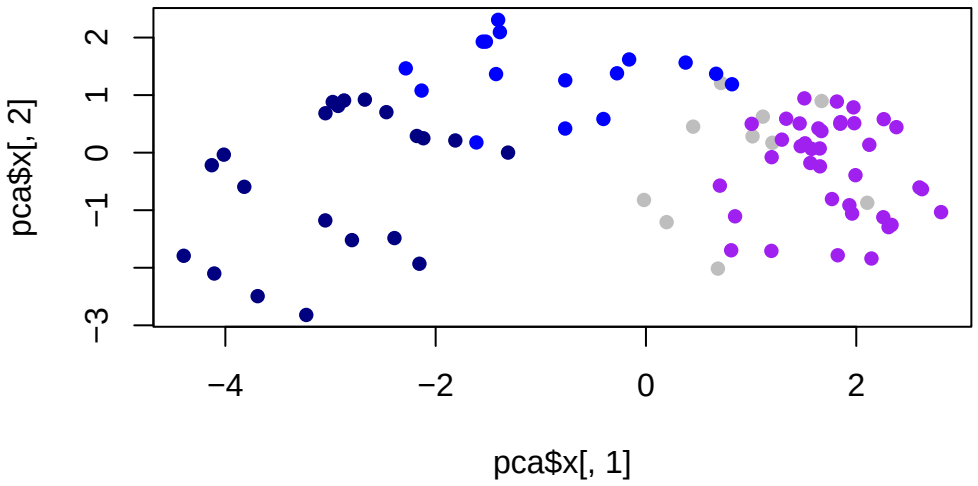
We will perform a PCA of the candy. Key-question: do we need to scale the data before PCA?

```
pca <- prcomp(candy, scale = T)
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
```

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

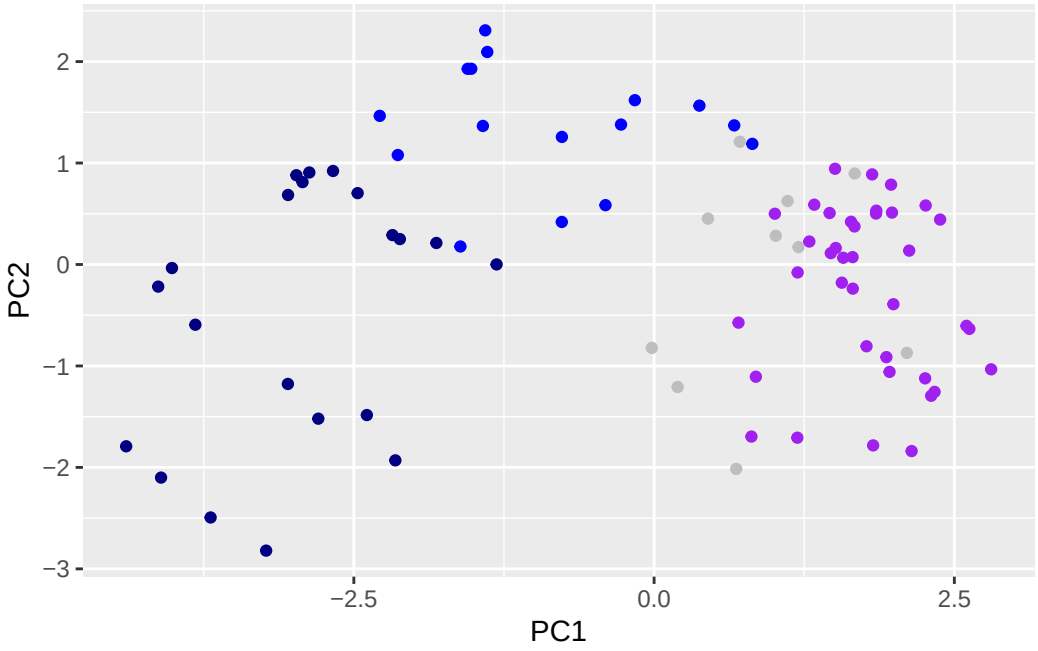



```r
my_data <- cbind(candy, pca$x[,1:3])
head(my_data)
```

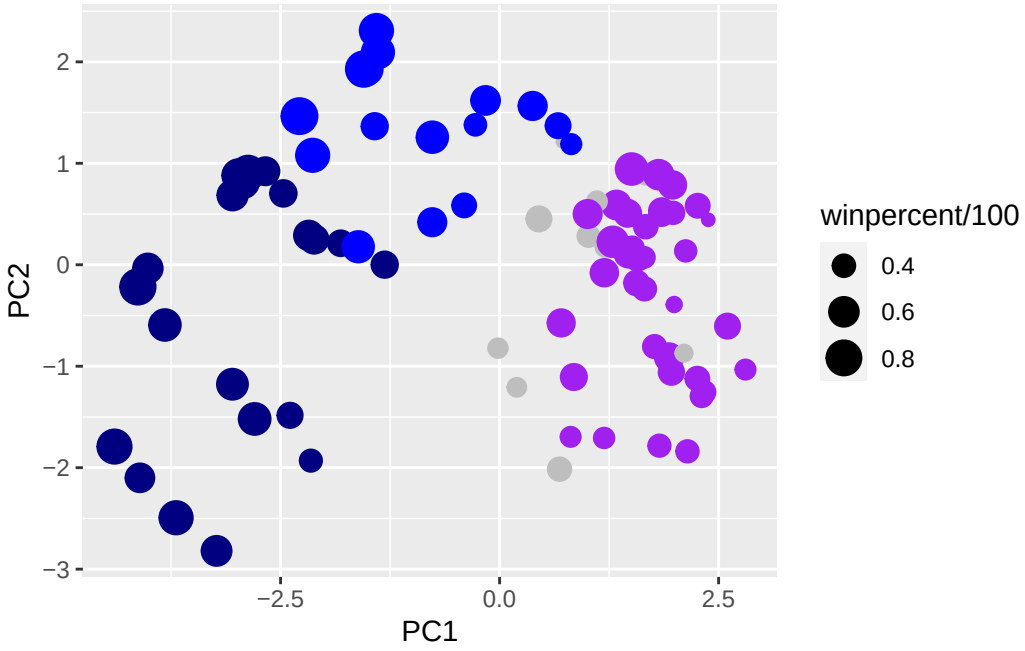
```
             chocolate fruity caramel peanutyalmondy nougat crispedricewafer
100 Grand            1      0       1              0      0                 1
3 Musketeers         1      0       0              0      1                 0
One dime             0      0       0              0      0                 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        PC1
100 Grand       0   1        0        0.732        0.860   66.97173 -3.8198617
3 Musketeers    0   1        0        0.604        0.511   67.60294 -2.7960236
One dime        0   0        0        0.011        0.116   32.26109  1.2025836
```

```
One quarter     0   0        0        0.011        0.511   46.11650  0.4486538
Air Heads       0   0        0        0.906        0.511   52.34146  0.7028992
Almond Joy      0   1        0        0.465        0.767   50.34755 -2.4683383
                     PC2        PC3
100 Grand    -0.5935788 -2.1863087
3 Musketeers -1.5196062  1.4121986
One dime      0.1718121  2.0607712
One quarter   0.4519736  1.4764928
Air Heads    -0.5731343 -0.9293893
Almond Joy    0.7035501  0.8581089
```

```
ggplot(my_data) +aes(PC1, PC2, label = rownames(my_data)) + geom_point(col = my_cols)
```

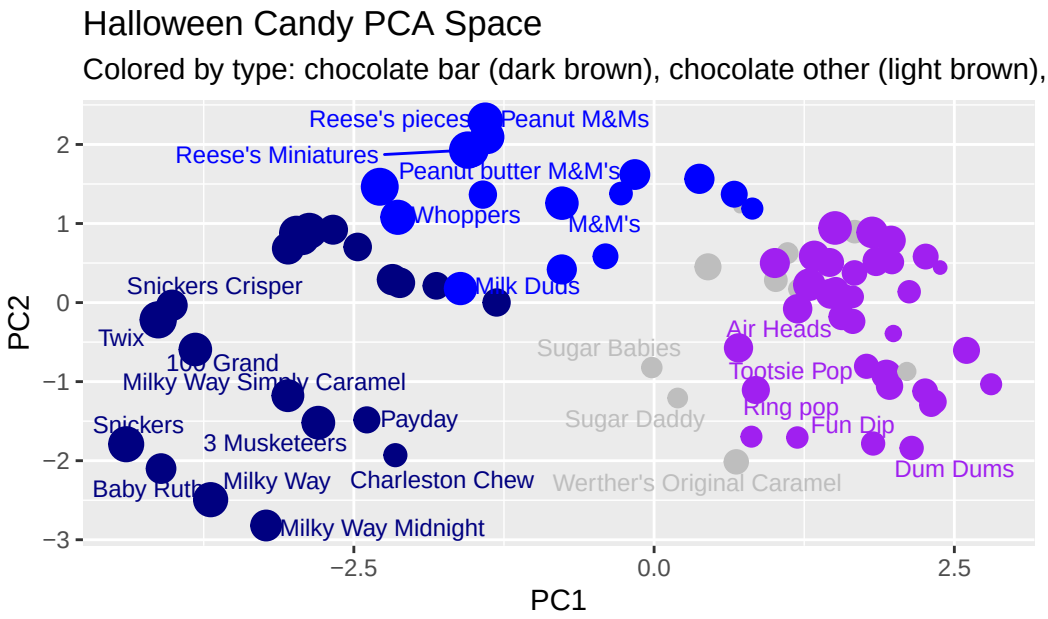



```
p <- ggplot(my_data) +
        aes(x=PC1, y=PC2,
            size=winpercent/100,
            text=rownames(my_data),
            label=rownames(my_data)) +
        geom_point(col=my_cols)
p
```

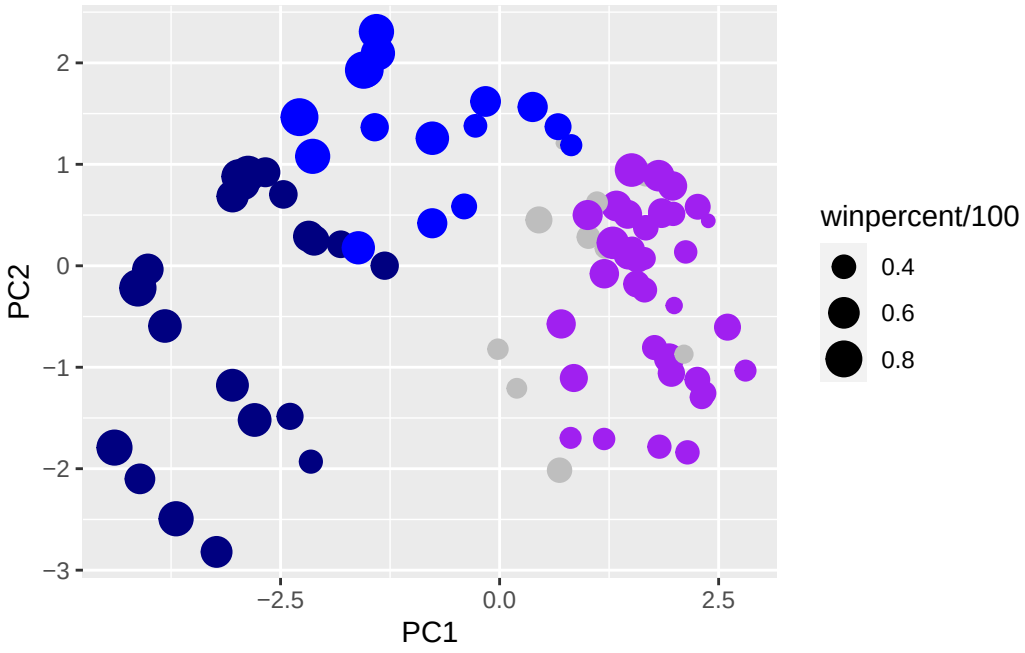



```
p + geom_text_repel(size=3.3, col=my_cols, max.overlaps = 7)  +
  theme(legend.position = "none") +
  labs(title="Halloween Candy PCA Space",
       subtitle="Colored by type: chocolate bar (dark brown), chocolate other (light brown
       caption="Data from 538")
```

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


Halloween Candy PCA Space
Colored by type: chocolate bar (dark brown), chocolate other (light brown),
Reese's pieces
Peanut M&Ms
Reese's Miniatures
Peanut butter M&M's
Whoppers
M&M's
Snickers Crisper
Milk Duds
Twix
Air Heads
Sugar Babies
100 Grand
Tootsie Pop
Milky Way Simply Caramel
Ring pop
Payday
Sugar Daddy
Fun Dip
Snickers
3 Musketeers
Dum Dums
Milky Way
Charleston Chew
Werther's Original Caramel
Baby Ruth
Milky Way Midnight
PC2
PC1
2
1
0
−1
−2
−3
−2.5
0.0
2.5
Data from 538

```
p
```


PC2
PC1
2
1
0
−1
−2
−3
−2.5
0.0
2.5
winpercent/100
0.4
0.6
0.8

```
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
```

```
#ggplotly(p)
```

How do the original variables contribute to our PCs? For this we look at the loadings component of our results object i.e. the `pca$rotation` object

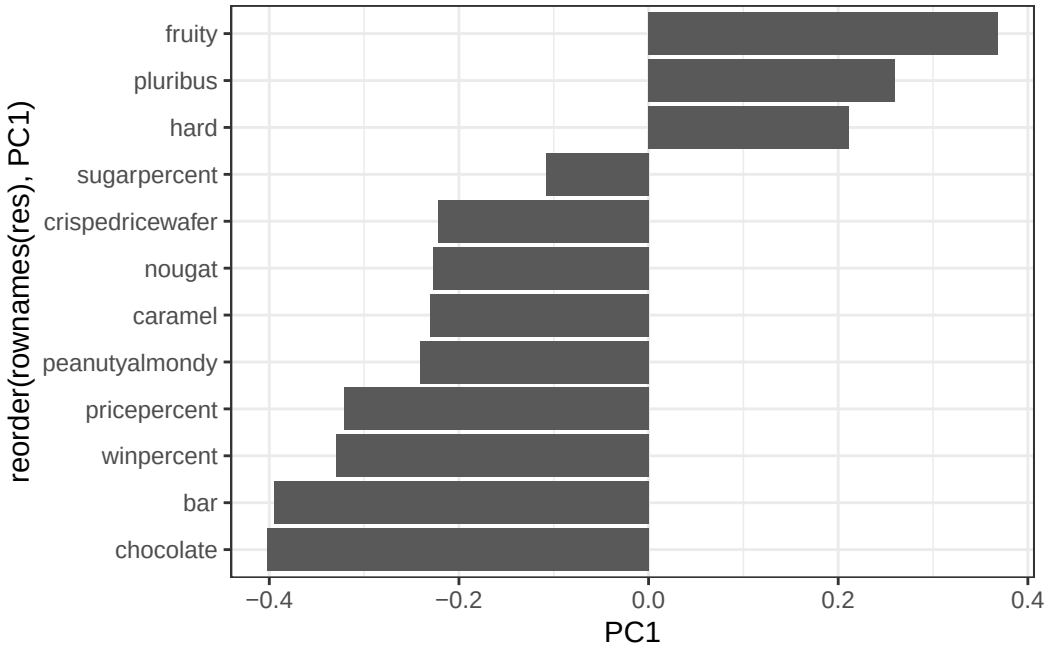
```
head(pca$rotation)
```

```
                       PC1         PC2         PC3          PC4         PC5
chocolate        -0.4019466  0.21404160  0.01601358 -0.016673032  0.06603585
fruity            0.3683883 -0.18304666 -0.13765612 -0.004479829  0.14353533
caramel          -0.2299709 -0.40349894 -0.13294166 -0.024889542 -0.50730150
peanutyalmondy   -0.2407155  0.22446919  0.18272802  0.466784287  0.39993025
nougat           -0.2268102 -0.47016599  0.33970244  0.299581403 -0.18885242
crispedricewafer -0.2215182  0.09719527 -0.36485542 -0.605594730  0.03465232
                        PC6         PC7        PC8          PC9        PC10
chocolate        -0.09018950 -0.08360642 -0.4908486 -0.151651568  0.10766136
fruity           -0.04266105  0.46147889  0.3980580 -0.001248306  0.36206250
caramel          -0.40346502 -0.44274741  0.2696345  0.019186442  0.22979901
peanutyalmondy   -0.09416259 -0.25710489  0.4577145  0.381068550 -0.14591236
nougat            0.09012643  0.36663902 -0.1879396  0.385278987  0.01132345
crispedricewafer -0.09007640  0.13077042  0.1356774  0.511634999 -0.26481014
                       PC11       PC12
```

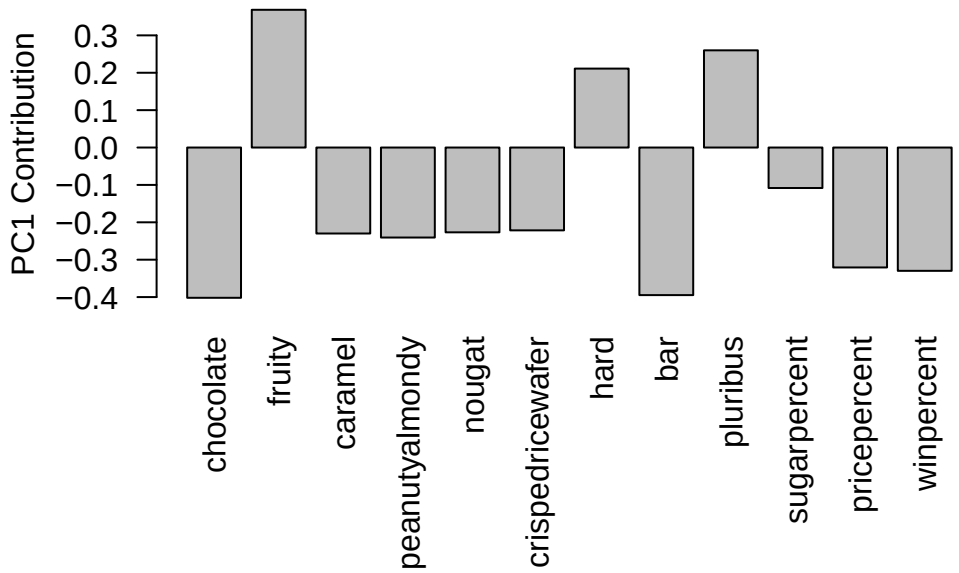
```
chocolate          0.1004528 0.69784924
fruity             0.1749490 0.50624242
caramel            0.1351582 0.07548984
peanutyalmondy     0.1124428 0.12972756
nougat            -0.3895447 0.09223698
crispedricewafer  -0.2261562 0.11727369
```

Make a barplot with ggplot and order the bars by their value. Recall that you need a data.frame as input for ggplot

```
res <- as.data.frame(pca$rotation)
ggplot(res) + aes(PC1, reorder(rownames(res), PC1)) +geom_col() + theme_bw()
```




```
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 are in the positive direction. These do make sense based on the correlation structure in the dataset. If you are a fruity candy you will tend to be hard and come in a packet with multiple candies in it.