Background:

In this mini-project we will examine Halloween Candy data. What is your favorite candy? What is nougat anyway? How do you say it in America?

FIrst step is to read the data

```
candy <- read.csv("candy-data.txt", 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
```

Q How many different candy types are in this data set ?

```
nrow(candy)
```

```
[1] 85
```

#Q2 How many fruity candy types are in the data set

```
sum(candy$fruity)
```

```
[1] 38
```

win percent of your favorite candy

```
rownames(candy)
```

```
 [1] "100 Grand"                   "3 Musketeers"
 [3] "One dime"                    "One quarter"
 [5] "Air Heads"                   "Almond Joy"
 [7] "Baby Ruth"                   "Boston Baked Beans"
 [9] "Candy Corn"                  "Caramel Apple Pops"
[11] "Charleston Chew"             "Chewey Lemonhead Fruit Mix"
[13] "Chiclets"                    "Dots"
[15] "Dum Dums"                    "Fruit Chews"
[17] "Fun Dip"                     "Gobstopper"
[19] "Haribo Gold Bears"           "Haribo Happy Cola"
[21] "Haribo Sour Bears"           "Haribo Twin Snakes"
[23] "HersheyÕs Kisses"            "HersheyÕs Krackel"
[25] "HersheyÕs Milk Chocolate"    "HersheyÕs Special Dark"
[27] "Jawbusters"                  "Junior Mints"
[29] "Kit Kat"                     "Laffy Taffy"
[31] "Lemonhead"                   "Lifesavers big ring gummies"
[33] "Peanut butter M&MÕs"         "M&MÕs"
[35] "Mike & Ike"                  "Milk Duds"
[37] "Milky Way"                   "Milky Way Midnight"
[39] "Milky Way Simply Caramel"    "Mounds"
[41] "Mr Good Bar"                 "Nerds"
[43] "Nestle Butterfinger"         "Nestle Crunch"
[45] "Nik L Nip"                   "Now & Later"
[47] "Payday"                      "Peanut M&Ms"
[49] "Pixie Sticks"                "Pop Rocks"
[51] "Red vines"                   "ReeseÕs Miniatures"
[53] "ReeseÕs Peanut Butter cup"   "ReeseÕs pieces"
[55] "ReeseÕs stuffed with pieces" "Ring pop"
[57] "Rolo"                        "Root Beer Barrels"
[59] "Runts"                       "Sixlets"
[61] "Skittles original"           "Skittles wildberry"
[63] "Nestle Smarties"             "Smarties candy"
[65] "Snickers"                    "Snickers Crisper"
[67] "Sour Patch Kids"             "Sour Patch Tricksters"
[69] "Starburst"                   "Strawberry bon bons"
[71] "Sugar Babies"                "Sugar Daddy"
[73] "Super Bubble"                "Swedish Fish"
```

```
[75] "Tootsie Pop"                 "Tootsie Roll Juniors"
[77] "Tootsie Roll Midgies"        "Tootsie Roll Snack Bars"
[79] "Trolli Sour Bites"           "Twix"
[81] "Twizzlers"                   "Warheads"
[83] "WelchÕs Fruit Snacks"        "WertherÕs Original Caramel"
[85] "Whoppers"
```

```r
candy["Sugar Babies", ]$winpercent
```

```
[1] 33.43755
```

```r
candy["WertherÕs Original Caramel", ]
```

```
                           chocolate fruity caramel peanutyalmondy nougat
WertherÕs Original Caramel         0      0       1              0      0
                           crispedricewafer hard bar pluribus sugarpercent
WertherÕs Original Caramel                0    1   0        0        0.186
                           pricepercent winpercent
WertherÕs Original Caramel        0.267   41.90431
```

```r
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	

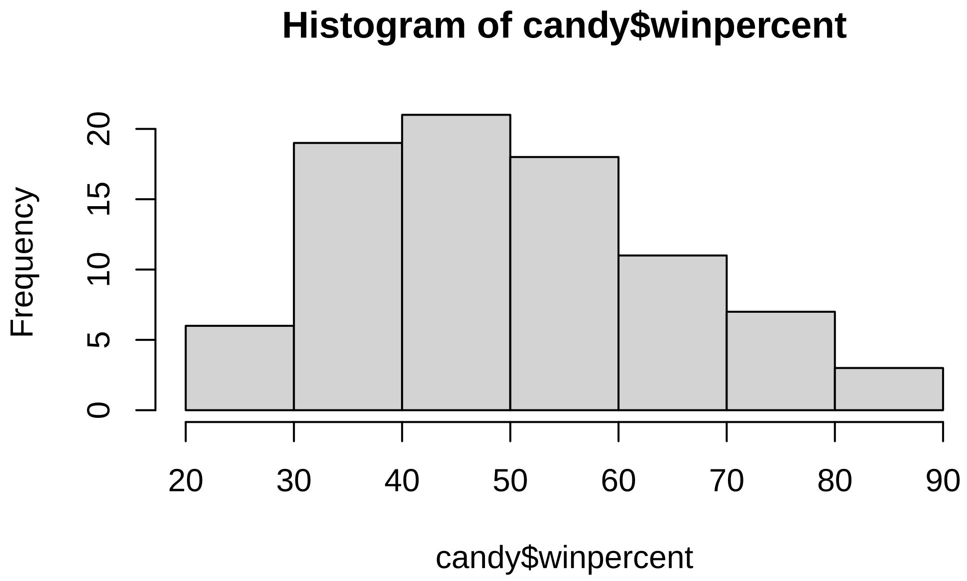
Q6. Is there any variable/column that looks to be on a different scale to the majority of the other columns in the dataset?

win percent

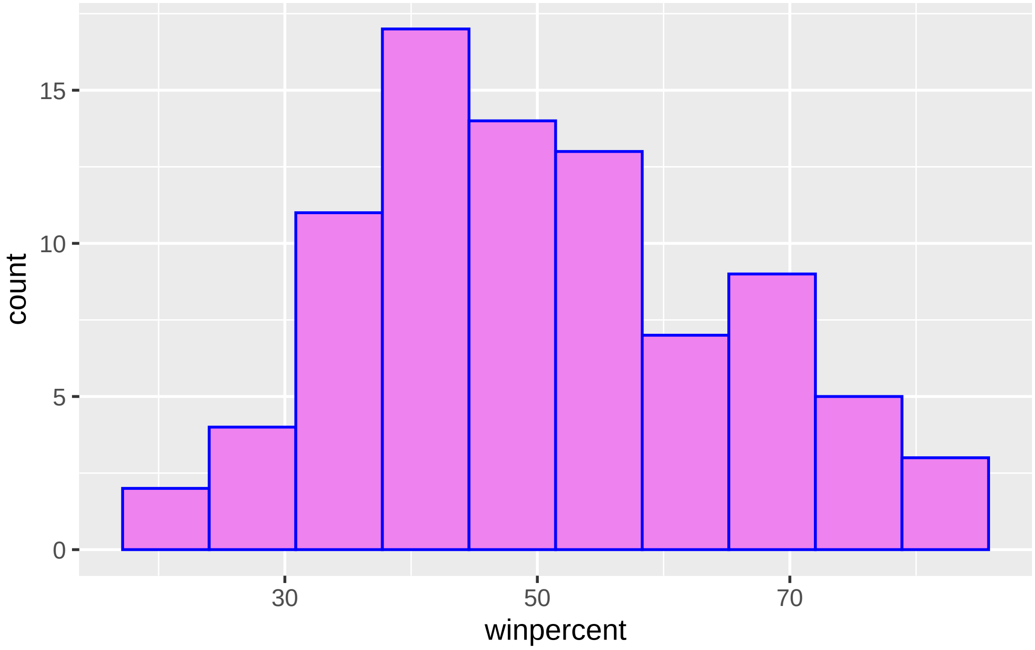
Q7. What do you think a zero and one represent for the candy$chocolate column?

Q8. Plot a histogram of winpercent values:

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



```r
library(ggplot2)
ggplot(candy, aes(x=winpercent)) +geom_histogram(bins=10, col="blue", fill="violet")
```



# Q9 is the distribution of winpercent value symmetrical?

no

# Q10 is the center of the distribution above or below 50%?

below 50%

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

```
chocolate.inds <-as.logical(candy$chocolate)
chocolate.win <-candy[chocolate.inds,]$winpercent

mean(chocolate.win)
```

```
[1] 60.92153
```

Q12 Is this statistically significant

Yes

```
fruity.inds <-as.logical(candy$fruity)
fruity.win <-candy[fruity.inds,]$winpercent

mean(fruity.win)
```

```
[1] 44.11974
```

```
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 
```

3. Overall Candy Rankings

The base R `sort()` and `order()` functions are very useful

```
x <-c(5,1,2,6)

sort(x)
```

```
[1] 1 2 5 6
```

order function: tells you the position

```
x[order(x)]
```

```
[1] 1 2 5 6
```

```
y <-c("barry", "alice", "chandra")
y
```

```
[1] "barry"   "alice"   "chandra"
```

```
sort(y)
```

```
[1] "alice"   "barry"   "chandra"
```

```
order(y)
```

```
[1] 2 1 3
```

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

First I want to order/arrange the entire dataset by winpercent values

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

```r
inds <-order(candy$winpercent)
head(candy[inds,], 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
```

```r
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
```

Q15 make a first barplot of candy ranking based on winpercent values

Barplot: the dafult barplot, made with `geom_col()` has the bars in order

```
p <-ggplot(candy) + aes(winpercent, reorder( rownames(candy), winpercent)) + geom_col()
```

```
ggsave("mybarplot.png")
```

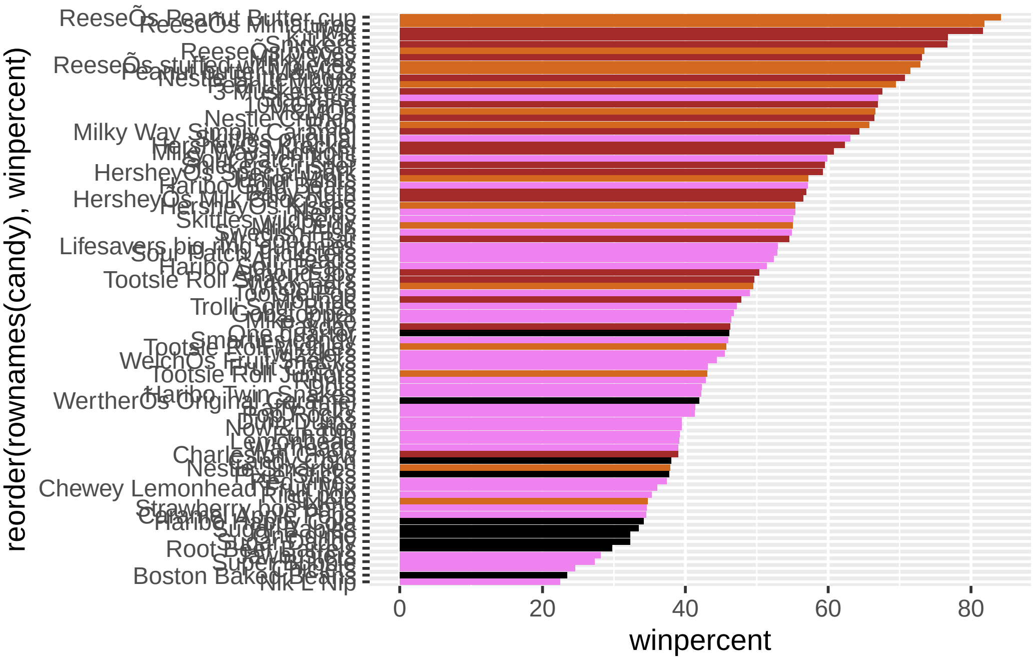
```
Saving 5.5 x 3.5 in image
```

Create color vector: color every single bar in the plot

```
my_cols <- rep("black", nrow(candy))
#my_cols
my_cols[as.logical(candy$chocolate)] <-"chocolate"
my_cols[as.logical(candy$bar)] <-"brown"
my_cols[as.logical(candy$fruity)] <-"violet"
my_cols
```

```
 [1] "brown"     "brown"     "black"     "black"     "violet"    "brown"
 [7] "brown"     "black"     "black"     "violet"    "brown"     "violet"
[13] "violet"    "violet"    "violet"    "violet"    "violet"    "violet"
[19] "violet"    "black"     "violet"    "violet"    "chocolate" "brown"
[25] "brown"     "brown"     "violet"    "chocolate" "brown"     "violet"
[31] "violet"    "violet"    "chocolate" "chocolate" "violet"    "chocolate"
[37] "brown"     "brown"     "brown"     "brown"     "brown"     "violet"
[43] "brown"     "brown"     "violet"    "violet"    "brown"     "chocolate"
[49] "black"     "violet"    "violet"    "chocolate" "chocolate" "chocolate"
[55] "chocolate" "violet"    "chocolate" "black"     "violet"    "chocolate"
[61] "violet"    "violet"    "chocolate" "violet"    "brown"     "brown"
[67] "violet"    "violet"    "violet"    "violet"    "black"     "black"
[73] "violet"    "violet"    "violet"    "chocolate" "chocolate" "brown"
[79] "violet"    "brown"     "violet"    "violet"    "violet"    "black"
[85] "chocolate"
```

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



Taking a look at pricepercent

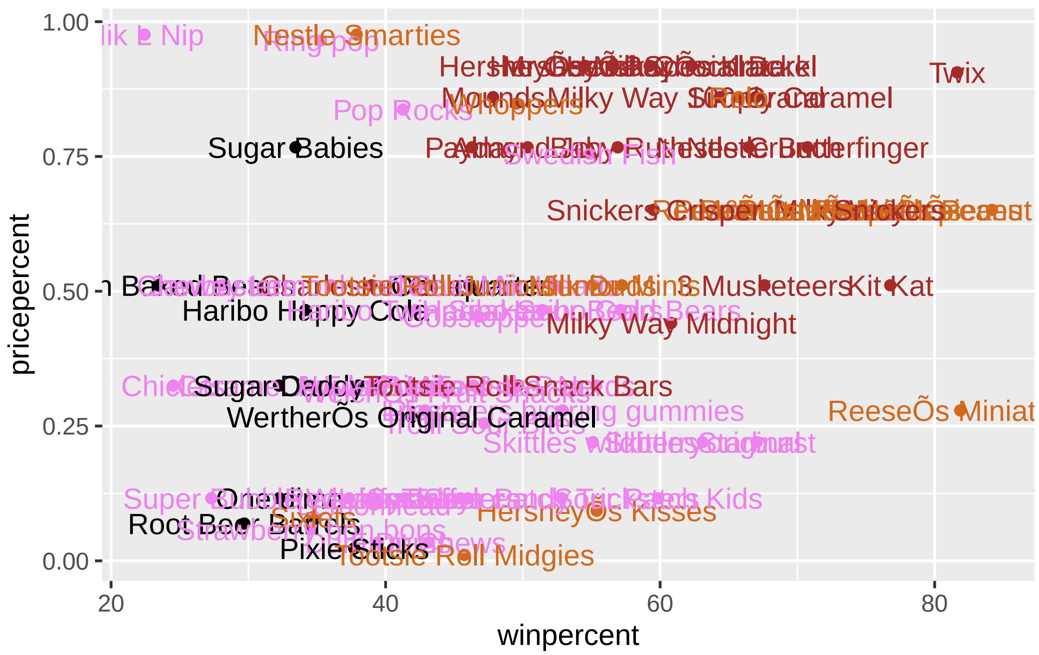
Q19. Which candy type is the highest ranked in terms of winpercent for the least money Reese’s miniature Q20: what are the top 5 msot expensive candy types in the dataset and of these which is least popular?

Nik L Nip Nestle Smarties Ring Pop Sugar Babies POprocks

What about value for money? What is the best candy for the least money?

One way would be to plot `winpercent` vs the `pricepercent`

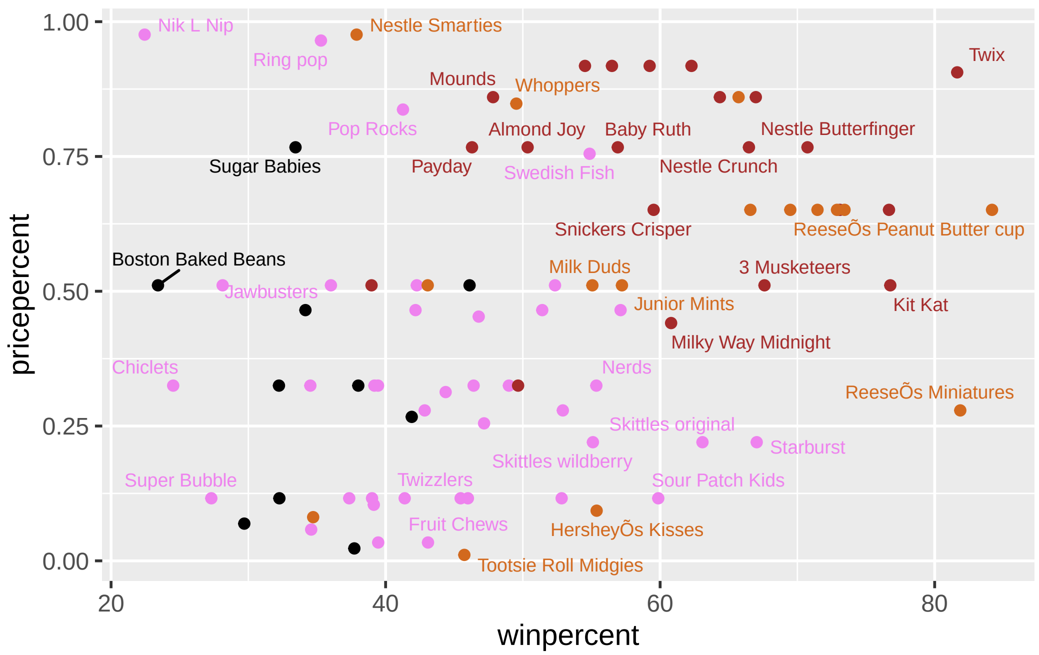
```r
ggplot(candy)+
  aes(winpercent, pricepercent, label=rownames(candy)) +geom_point(col= my_cols) +
geom_text(col=my_cols)
```



This plot sucks. Can't read the labels Use ggrepel package to help

```
library(ggrepel)
ggplot(candy)+
  aes(winpercent, pricepercent, label=rownames(candy)) +geom_point(col= my_cols) +
geom_text_repel(col=my_cols, size=2.5, max.overlaps=7)
```

```
Warning: ggrepel: 50 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

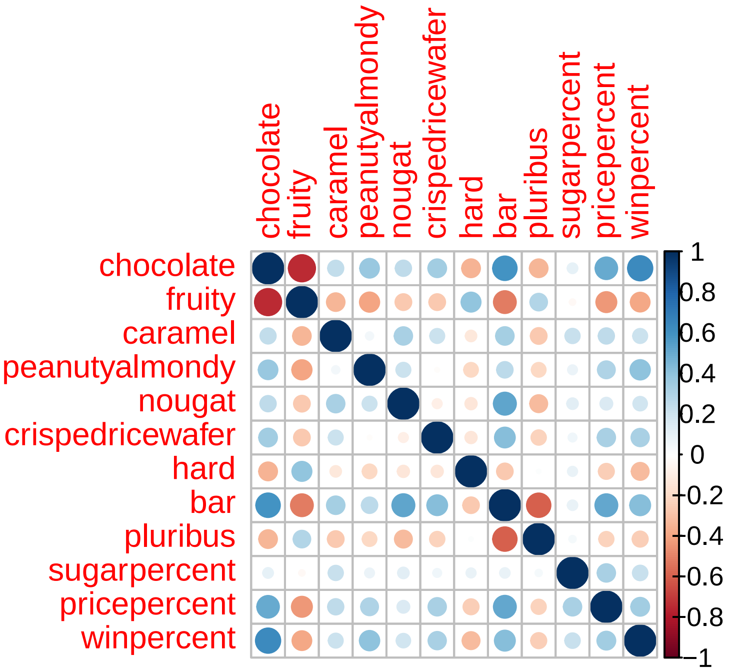


5 Exploring the correlation structure

```
library(corrplot)
```

```
corrplot 0.92 loaded
```

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



Q22: Examining this plot what two variables are anti-correlated (ie have minus values?) chocolate and fruity

Q23: Similarly, what two variables are most positively correlated? chocolate and how popular it is or bar

6 PCA: Principal Component Analysis

The main function that always there for us is `prcomp()`. It has an important arguemtn that is selt to `scale=FALSE`

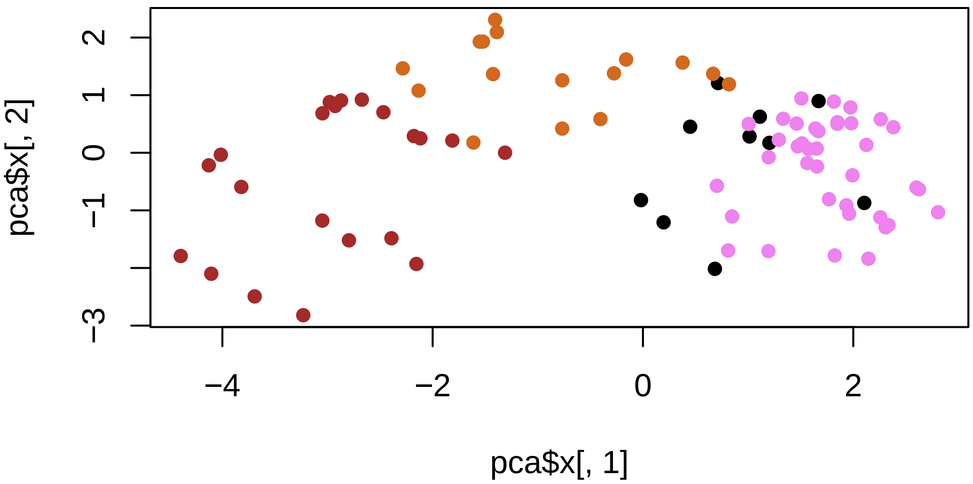
```
pca <-prcomp(candy, scale=TRUE)
summary(pca)
```

```
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
```

my PCA plot (a.k.a) PC1 vs PC2 score plot

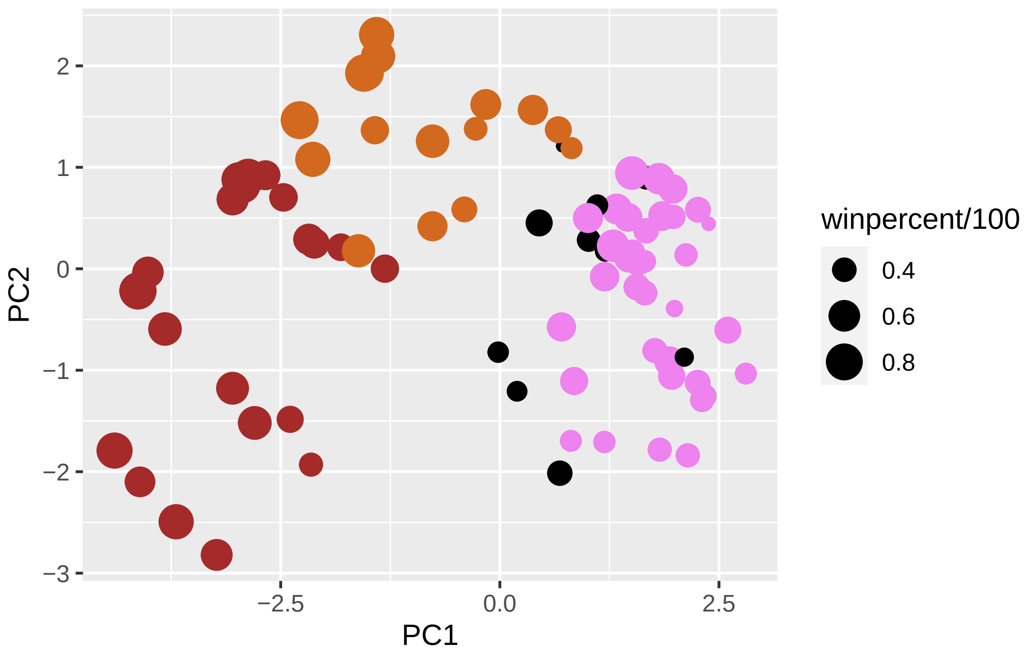
```
plot(pca$x[,1], pca$x[,2], col=my_cols, pch=16)
```



make a nicer plot with gg plot

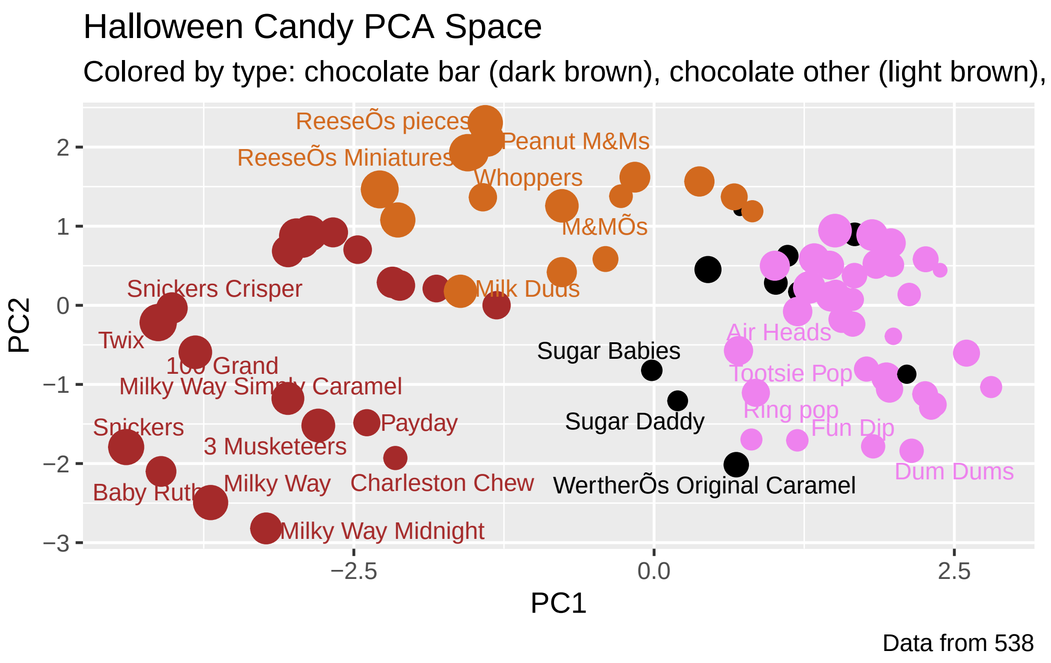
```
my_data <- cbind(candy, pca$x[,1:3])
```

```
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
```



```
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: 60 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),

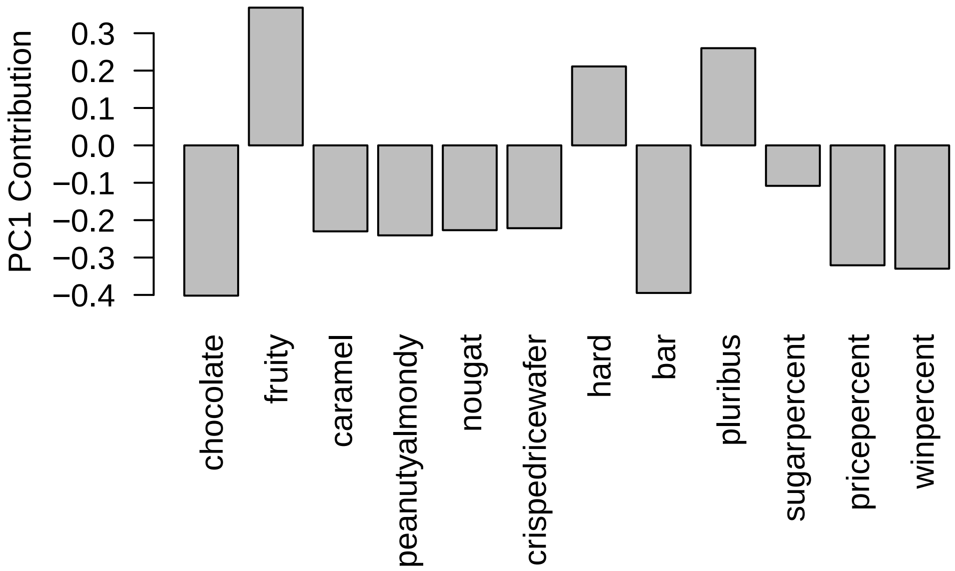
Data from 538

```
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)
```

```
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