Class 9

Mudit

1. Importing candy data

First things first, let's get the data from the FiveThirtyEight GitHub repo. You can either read from the URL directely or download this candy-data.csv file and place it in your project directory. Either way we need to load it up with read.csv() and inspect the data to see exactly what we're dealing with.

```r
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
```

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

85

```r
nrow(candy)
```

```
[1] 85
```

> Q2. How many fruity candy types are in the dataset? The functions dim(), nrow(), table() and sum() may be useful for answering the first 2 questions.

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

```
[1] 38
```

2. What is your favorate candy?

One of the most interesting variables in the dataset is winpercent. For a given candy this value is the percentage of people who prefer this candy over another randomly chosen candy from the dataset (what 538 term a matchup). Higher values indicate a more popular candy.

We can find the winpercent value for Twix by using its name to access the corresponding row of the dataset. This is because the dataset has each candy name as rownames (recall that we set this when we imported the original CSV file). For example the code for Twix is:

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

```
[1] 81.64291
```

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

Air Heads

```r
candy["Air Heads", ]$winpercent
```

```
[1] 52.34146
```

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

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

```
[1] 76.7686
```

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

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

```
[1] 49.6535
```

```r
library(dplyr)
```

```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```r
candy |>
filter(rownames(candy) %in% c("Kit Kat", "Tootsie Roll Snack Bars")) |>
  select(winpercent)
```

```
                        winpercent
Kit Kat                    76.7686
Tootsie Roll Snack Bars    49.6535
```

```r
candy |>
  filter(winpercent > 75) |>
  filter(pricepercent < 0.5)
```

```
                   chocolate fruity caramel peanutyalmondy nougat
Reese's Miniatures         1      0       0              1      0
                   crispedricewafer hard bar pluribus sugarpercent pricepercent
Reese's Miniatures                0    0   0        0        0.034        0.279
                   winpercent
Reese's Miniatures   81.86626
```

```r
library(dplyr)

candy |>
filter(rownames(candy) == "Dum Dums") |>
  select(winpercent)
```

```
         winpercent
Dum Dums   39.46056
```

```r
library(dplyr)

candy |>
filter(rownames(candy) %in% c("Dum Dums", "Twix")) |>
  select(winpercent)
```

```
         winpercent
Dum Dums   39.46056
Twix       81.64291
```

```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	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
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?

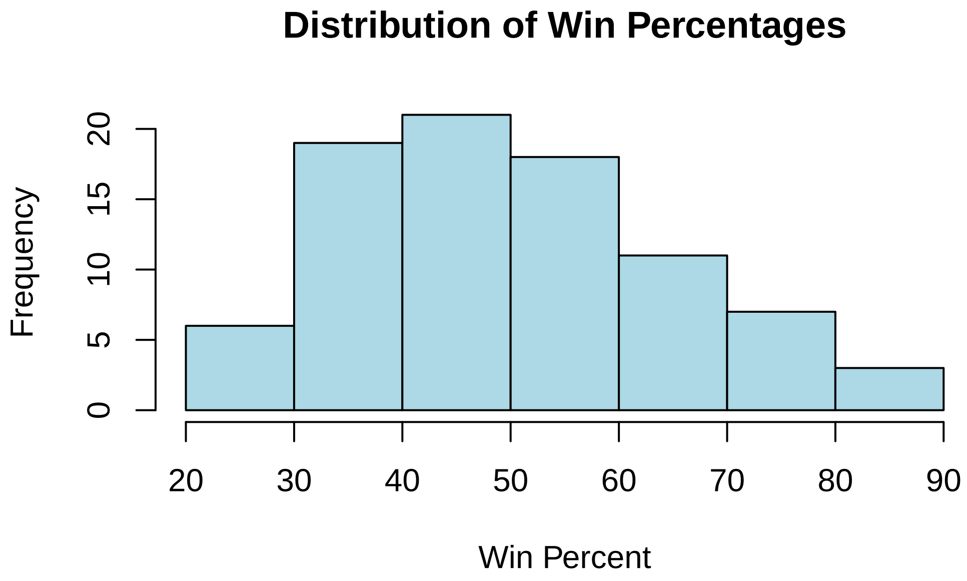
Yes, majority of the columns like chocolate have binary scale 1s or 0s while winpercent, pricepercent and sugarpercent column has a continous numerical scale

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

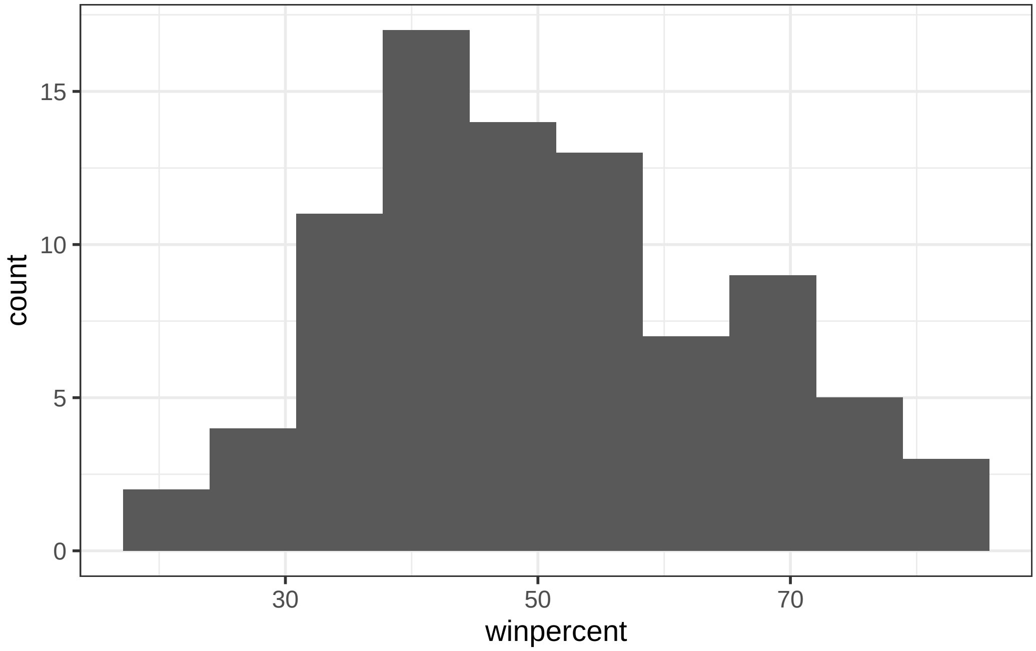
zero means the candy has chocolate while one represent it does not

> Q8. Plot a histogram of winpercent values

```r
#hist(candy$winpercent)
hist(candy$winpercent,
     main = "Distribution of Win Percentages",
     xlab = "Win Percent",
     col = "lightblue",
     border = "black")
```



```r
library(ggplot2)

ggplot(candy) +
  aes(winpercent) +
  geom_histogram(bins=10) +
  theme_bw()
```



```
mean_winpercent <- mean(candy$winpercent)
median_winpercent <- median(candy$winpercent)

mean_winpercent
```

```
[1] 50.31676
```

```
median_winpercent
```

```
[1] 47.82975
```

Q9. Is the distribution of winpercent values symmetrical?

No, as mean and median are not equal

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

below 50%, as for skewed distribution median is a better choice for center of distribution

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

higher

```
meanChoco <- mean(candy$winpercent[as.logical(candy$chocolate)])
meanFruit <- mean(candy$winpercent[as.logical(candy$fruity)])

meanChoco > meanFruit
```

```
[1] TRUE
```

> Q12. Is this difference statistically significant?

```
t.test <- t.test(candy$winpercent[as.logical(candy$chocolate)], candy$winpercent[as.logical(

t.test
```

```

	Welch Two Sample t-test

data:  candy$winpercent[as.logical(candy$chocolate)] and candy$winpercent[as.logical(candy$f
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 
```

the p-value is less then 0.05 which indicates that this difference is significant.

3. Overall Candy Rankings

Let's use the base R order() function together with head() to sort the whole dataset by winpercent. Or if you have been getting into the tidyverse and the dplyr package you can use the arrange() function together with head() to do the same thing and answer the following questions:

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

```
library(dplyr)
candy %>%
  arrange(winpercent) %>%
  head(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?

```
library(dplyr)
candy %>%
  arrange(desc(winpercent)) %>%
  head(5)
```

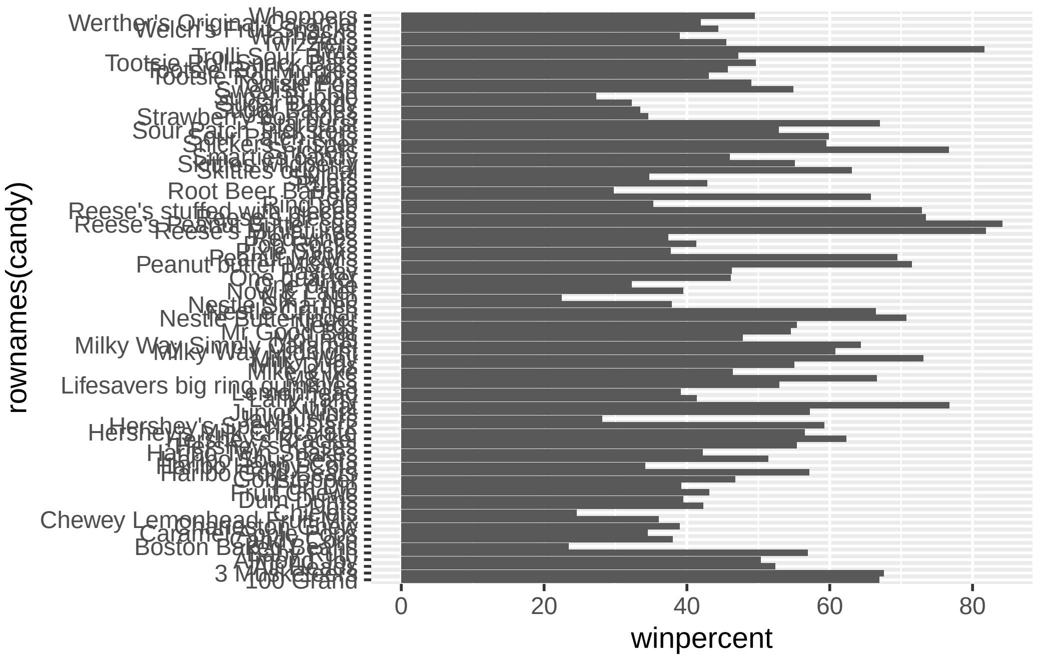
```
                          chocolate fruity caramel peanutyalmondy nougat
Reese's Peanut Butter cup         1      0       0              1      0
Reese's Miniatures                1      0       0              1      0
Twix                              1      0       1              0      0
Kit Kat                           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
```

To examine more of the dataset in this vain we can make a barplot to visualize the overall rankings. We will use an iterative approach to building a useful visulization by getting a rough starting plot and then refining and adding useful details in a stepwise process.

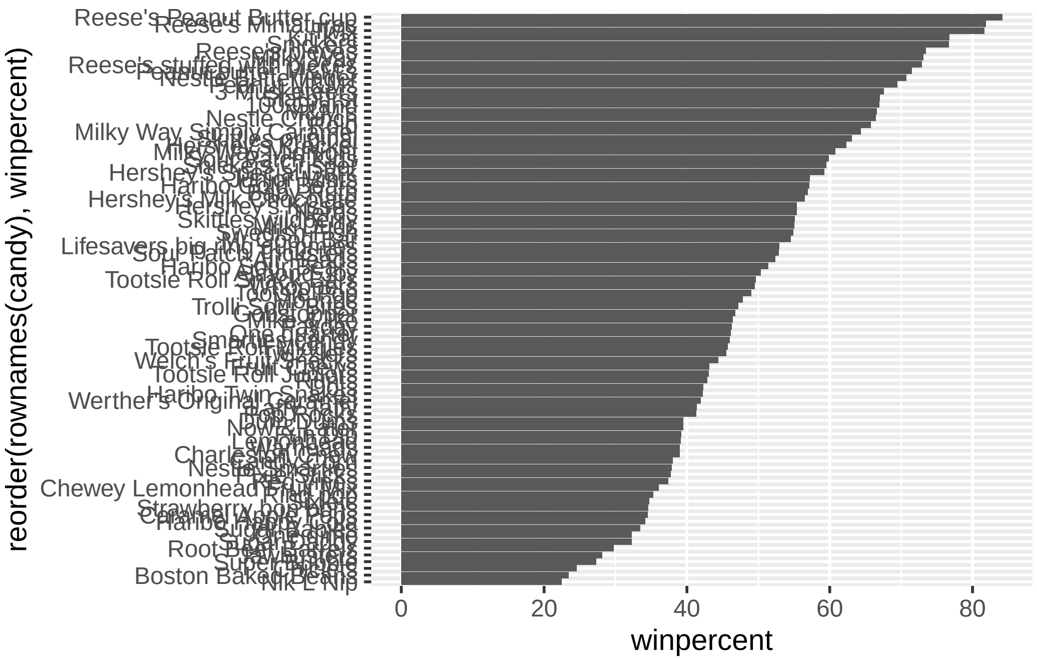
> Q15. Make a first barplot of candy ranking based on winpercent values.

```
library(ggplot2)

ggplot(candy) +
  aes(winpercent, rownames(candy)) +
  geom_bar(stat = "identity")
```



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

```
library(ggplot2)

ggplot(candy) +
  aes(winpercent, reorder(rownames(candy),winpercent)) +
  geom_bar(stat = "identity")
```



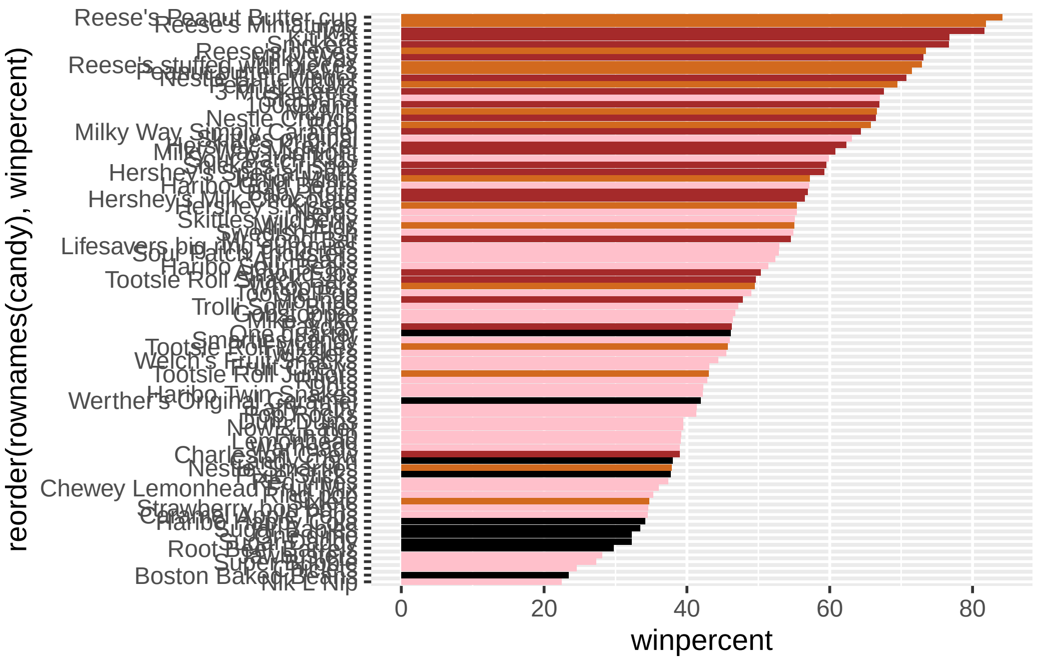
Time to add some useful color

Let's setup a color vector (that signifies candy type) that we can then use for some future plots. We start by making a vector of all black values (one for each candy). Then we overwrite chocolate (for chocolate candy), brown (for candy bars) and red (for fruity candy) values.

```
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"
```

Now let's try our barplot with these colors. Note that we use fill=my_cols for geom_col(). Experement to see what happens if you use col=mycols.

```
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?

Nik L Nip

> Q18. What is the best ranked fruity candy?

Reese's Peanut Butter cup

4. Taking a look at pricepercent

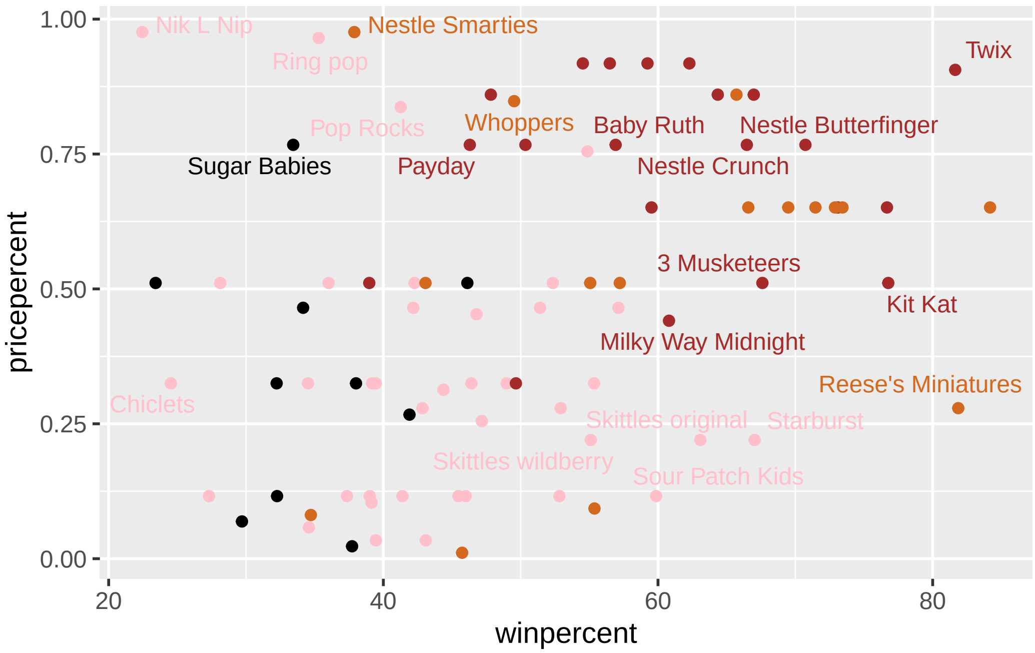
What about value for money? What is the the best candy for the least money? One way to get at this would be to make a plot of winpercent vs the pricepercent variable. The pricepercent variable records the percentile rank of the candy's price against all the other candies in the dataset. Lower vales are less expensive and high values more expensive.

To this plot we will add text labels so we can more easily identify a given candy. There is a regular geom_label() that comes with ggplot2. However, as there are quite a few candys in our dataset lots of these labels will be overlapping and hard to read. To help with this we can use the geom_text_repel() function from the ggrepel package.

```r
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 = 5)
```

```
Warning: ggrepel: 65 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?

Reeses Miniatures (just by looking at the plot)

```
a <- data.frame(candyName = rownames(candy), bang_for_buck = candy$winpercent/candy$priceperc
library(dplyr)
a %>%
  arrange(desc(bang_for_buck)) %>%
  head(5)
```

```
            candyName bang_for_buck
1 Tootsie Roll Midgies     4157.8862
```

```
2         Pixie Sticks     1640.1016
3          Fruit Chews     1267.3212
4             Dum Dums     1160.6045
5  Strawberry bon bons      596.1895
```

Tootsie Roll Midgies (from the analysis)

> 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
```

Nik L Nip

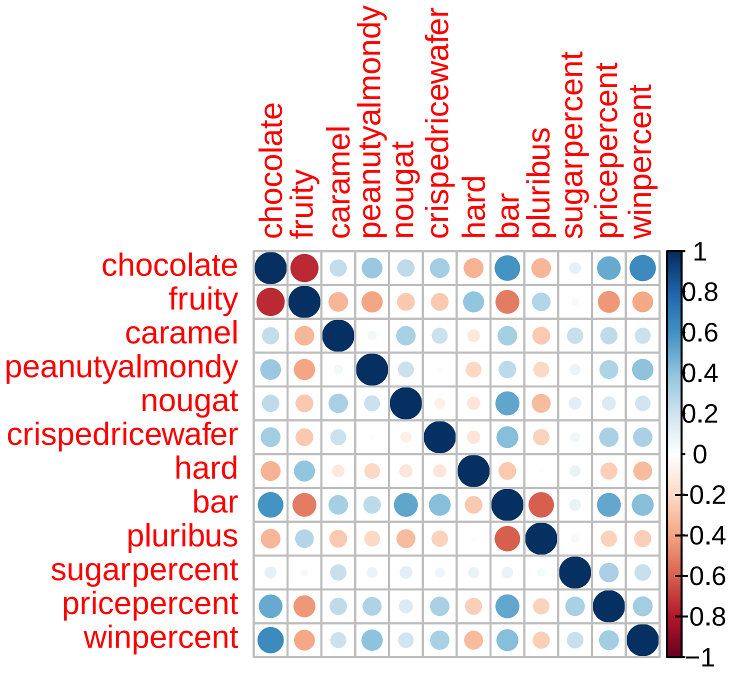
5 Exploring the correlation structure

Now that we've explored the dataset a little, we'll see how the variables interact with one another. We'll use correlation and view the results with the corrplot package to plot a correlation matrix.

```
library(corrplot)
```

```
corrplot 0.95 loaded
```

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



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

chocolate and fruity

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

chocolate and winpercent

6. Principal Component Analysis

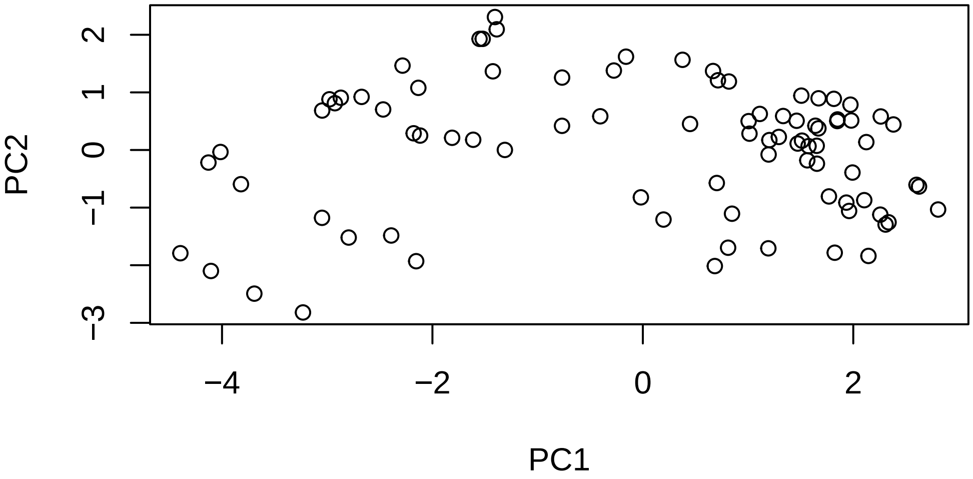
Let's apply PCA using the prcom() function to our candy dataset remembering to set the scale=TRUE argument.

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

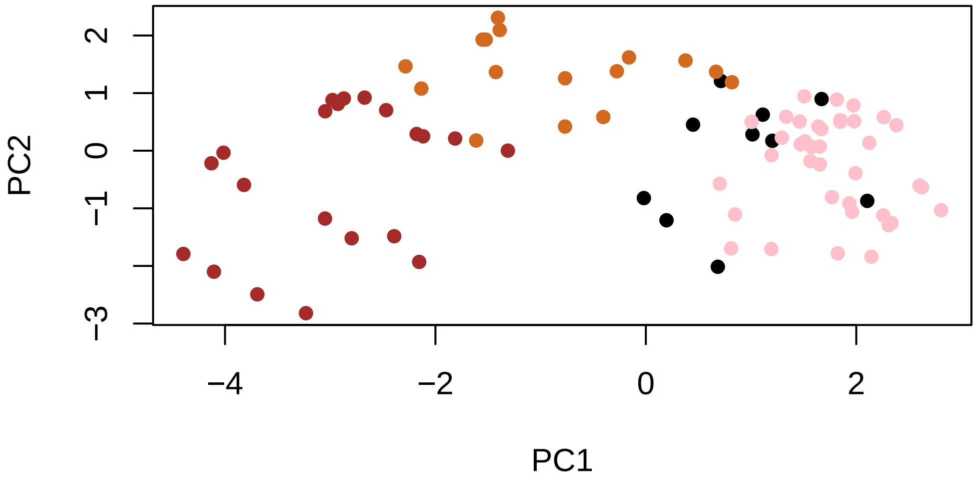
```
Importance of components:
                          PC1    PC2    PC3     PC4    PC5     PC6     PC7
Standard deviation     2.0788 1.1378 1.1092 1.07533 0.9518 0.81923 0.81530
Proportion of Variance 0.3601 0.1079 0.1025 0.09636 0.0755 0.05593 0.05539
Cumulative Proportion  0.3601 0.4680 0.5705 0.66688 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
```

```
plot(pca$x[,1:2])
```

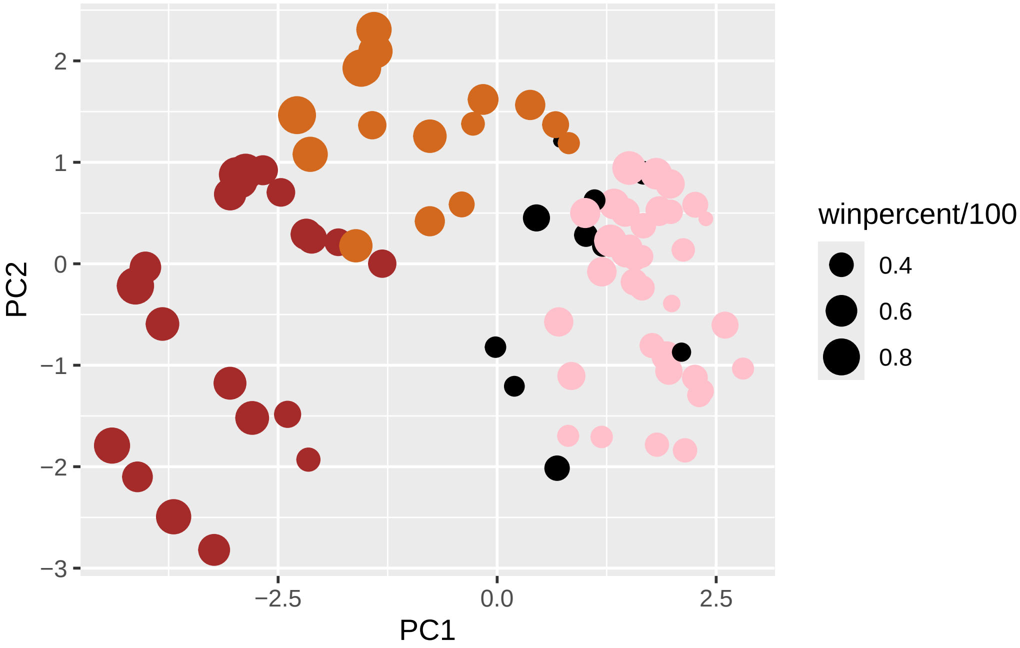


```
plot(pca$x[,1:2], col=my_cols, pch=16)
```



```
# Make a new data-frame with our PCA results and candy data
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
```

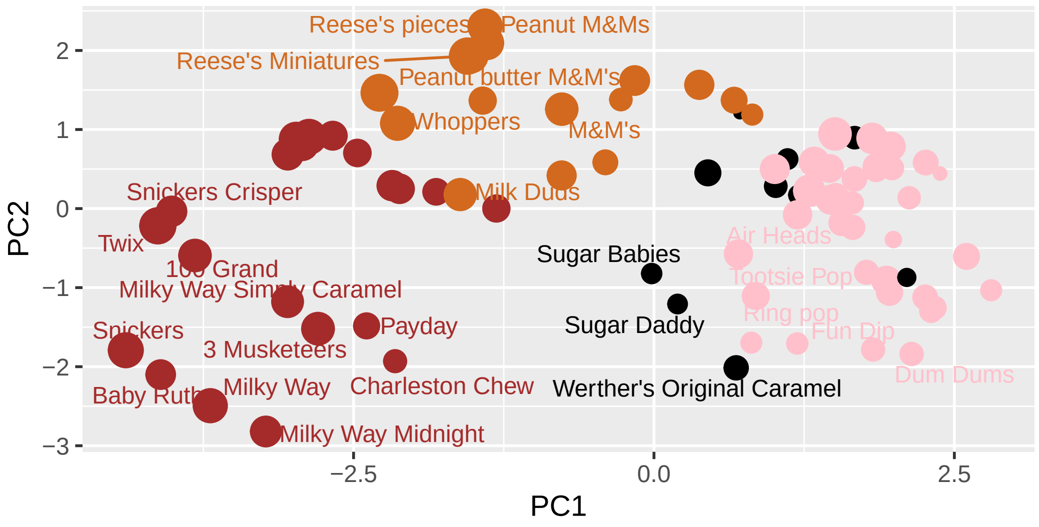


```r
library(ggrepel)

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
```

Halloween Candy PCA Space

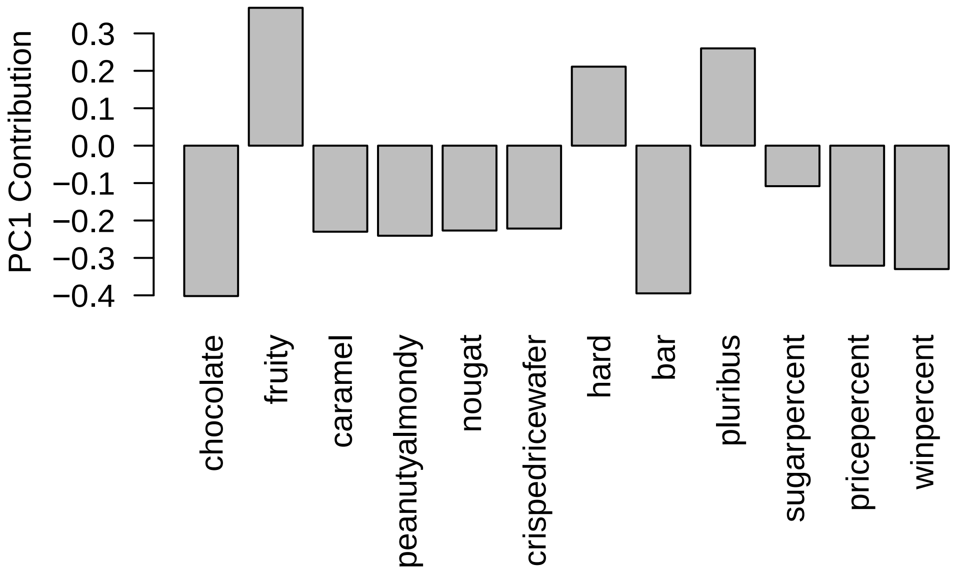
Colored by type: chocolate bar (dark brown), chocolate other (light brown),



Data from 538

```
#library(plotly)
#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?

pluribus, fruitly and hard. The ones positively correlated are on the same direction, while the ones that are negatively correlated are on opposite direction