Class 09: Halloween Mini-Project

Yu (Ericsson) Cao (PID: A16421048)

Here we analyze a candy dataset from the 538 websire.This is a CSV file from their Github repository.

Data Import

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

```
nrow(candy)
```

```
[1] 85
```

A: There are 85 different candy types are in this dataset.

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

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

```
[1] 38
```

A: There are 38 fruity candy types are in the dataset.

```
sum(candy$chocolate) #We can use the same approach for chocolate candy types
```

```
[1] 37
```

Data Exploration

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

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

```
[1] 76.7686
```

A: My favorite candy is Kit Kat and its winpercent value is 76.7686.

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

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

```
[1] 76.7686
```

A: Kit Kat's winpercent value is 76.7686.

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

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

```
[1] 49.6535
```

A: Tootsie Roll Snack Bars's winpercent value is 76.7686.

Q. What is the least liked candy in this dataset?

```
x <- c(5, 3, 4, 1)
sort(x)
```

```
[1] 1 3 4 5
```

```
order(x)
```

```
[1] 4 2 3 1
```

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

```
                  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
Root Beer Barrels         0      0       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
Root Beer Barrels                0    1   0        1        0.732        0.069
                  winpercent
Nik L Nip           22.44534
Boston Baked Beans  23.41782
Chiclets            24.52499
Super Bubble        27.30386
Jawbusters          28.12744
Root Beer Barrels   29.70369
```

```
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?

A: sugarpercent,pricepercent,winpercent look like they are on a different scale relative to all other columns.

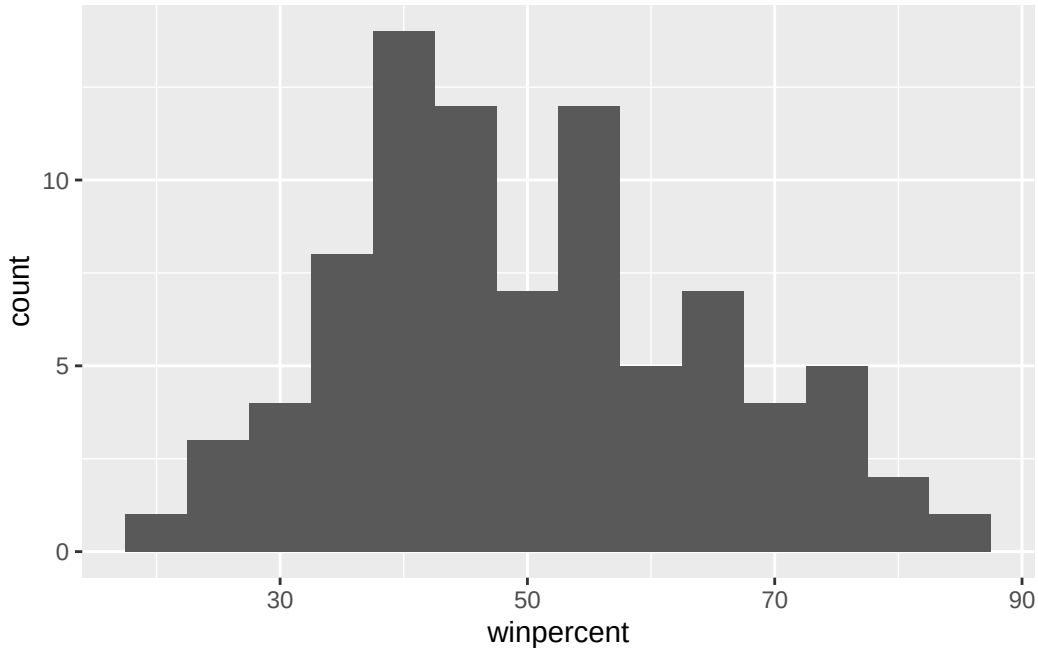
Q7. What do you think a zero and one represent for the candy$chocolate column?

A: Zero would indicate this candy does not belong under the chocolate category, while one does, this column represents a logical statement.

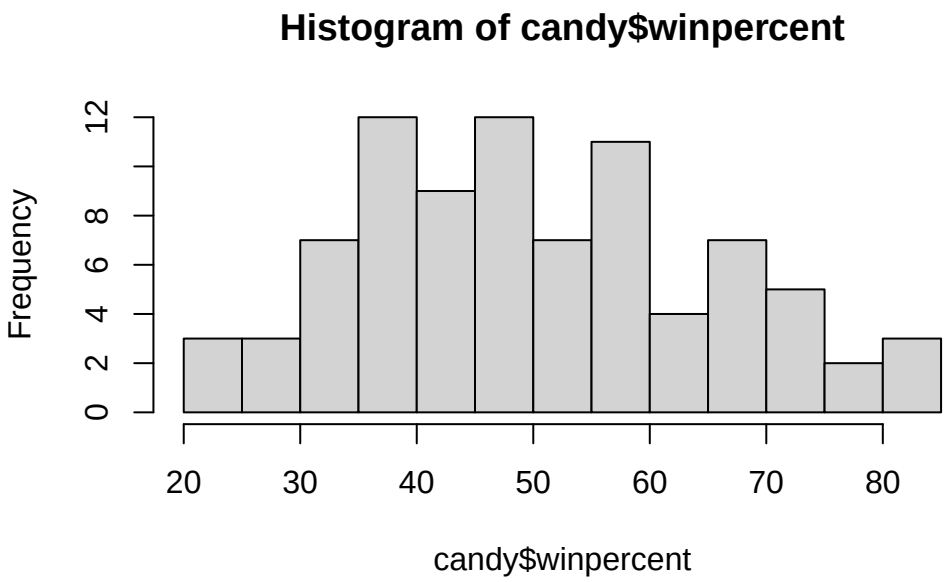
Q8. Plot a histogram of winpercent values

```
library(ggplot2)
ggplot(candy, aes(winpercent))+
```

```
  geom_histogram(binwidth=5)
```




```
hist(candy$winpercent, breaks=20)
```



Q9. Is the distribution of winpercent values symmetrical?

A: No, it is not symmetrical

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

A: Below 50%

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

First find all chocolate candy and their $winpercent values

Next summarize these values into one number

Then do the same for fruit candy and compare the numbers.

```
win_choco <- candy$winpercent[candy$chocolate == 1]
mean(win_choco)
```

```
[1] 60.92153
```

```
win_fruity <- candy$winpercent[candy$fruity == 1]
mean(win_fruity)
```

```
[1] 44.11974
```

A: On average chocolate candies are higher than fruity candies.

Q12. Is this difference statistically significant?

```
t.test(win_choco, win_fruity)
```

```

    Welch Two Sample t-test

data:  win_choco and win_fruity
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
```

A: Yes the difference is statistically significant.

Overall Candy Rankings

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

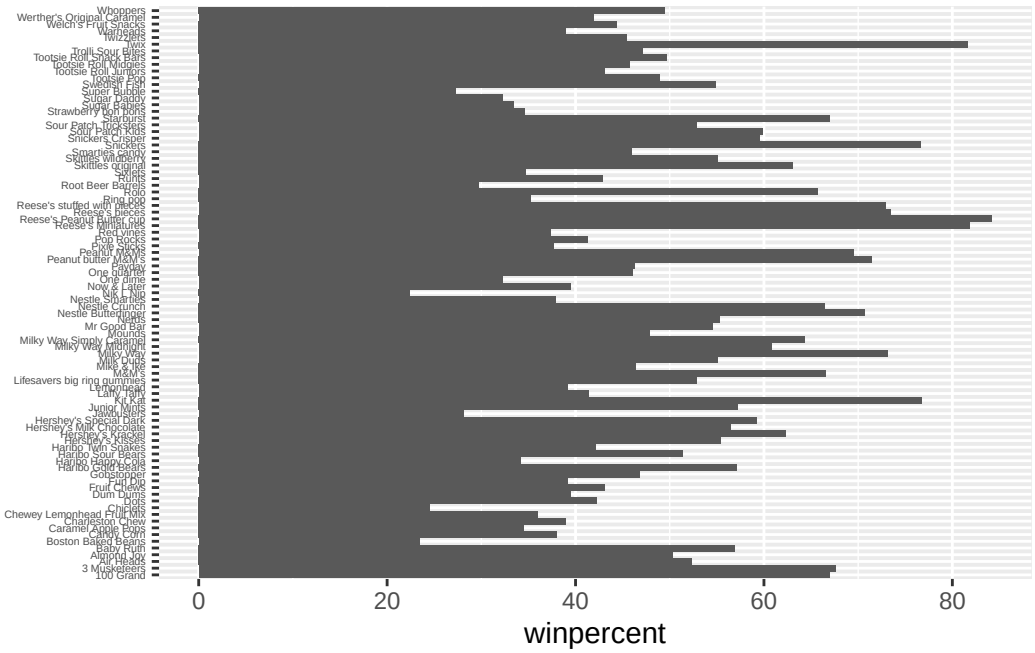
A: They are Nik L Nip, Boston Baked Beans, Chiclets, Super Bubble, and Jawbusters.

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

A: They are Reeses's Peanut Butter cup, Reese's miniatures, Twix, Kit Kat, and Snickers.

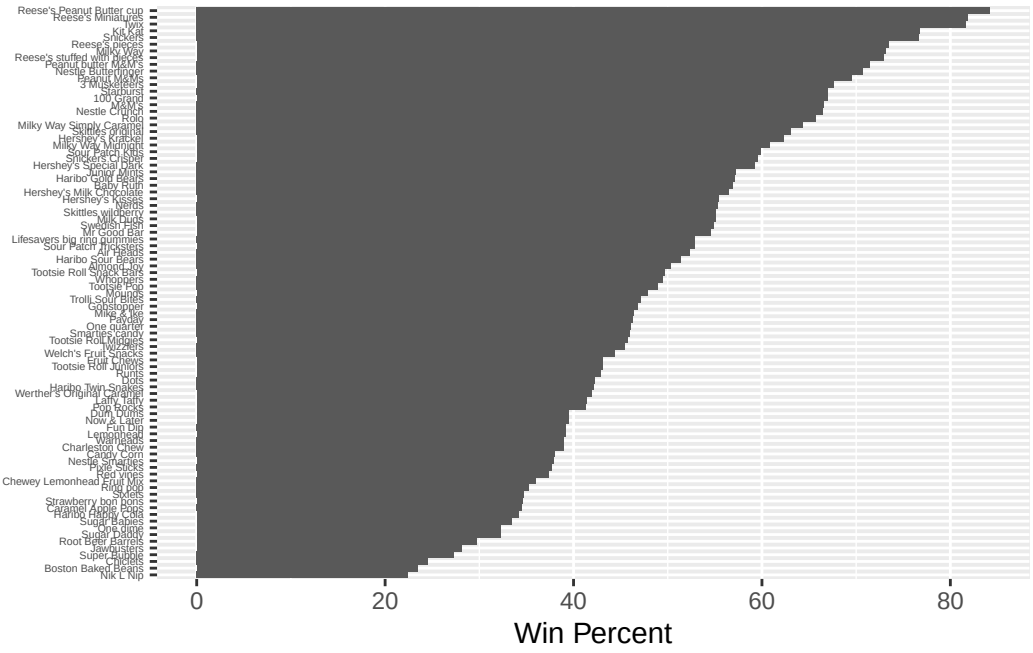
Q15. Make a first barplot of candy ranking based on winpercent values.

```
ggplot(candy)+
  aes(winpercent, rownames(candy))+
  geom_col()+
  theme(axis.text.y = element_text(size = 4), axis.title.y = element_blank())
```



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(width = 0.9) +
  labs(x="Win Percent") +
  theme(axis.text.y = element_text(size = 4), axis.title.y = element_blank())
```




```
ggsave('barplot1.png', width=7, height=10)
```

You can insert any image using this markdown syntax.

Add some color to our ggplot. We need to make a custom color vector.

```
# start with all black vector of colors
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"
my_cols
```

```
 [1] "brown"     "brown"     "black"     "black"     "pink"      "brown"
 [7] "brown"     "black"     "black"     "pink"      "brown"     "pink"
[13] "pink"      "pink"      "pink"      "pink"      "pink"      "pink"
[19] "pink"      "black"     "pink"      "pink"      "chocolate" "brown"
[25] "brown"     "brown"     "pink"      "chocolate" "brown"     "pink"
[31] "pink"      "pink"      "chocolate" "chocolate" "pink"      "chocolate"
[37] "brown"     "brown"     "brown"     "brown"     "brown"     "pink"
[43] "brown"     "brown"     "pink"      "pink"      "brown"     "chocolate"
[49] "black"     "pink"      "pink"      "chocolate" "chocolate" "chocolate"
```

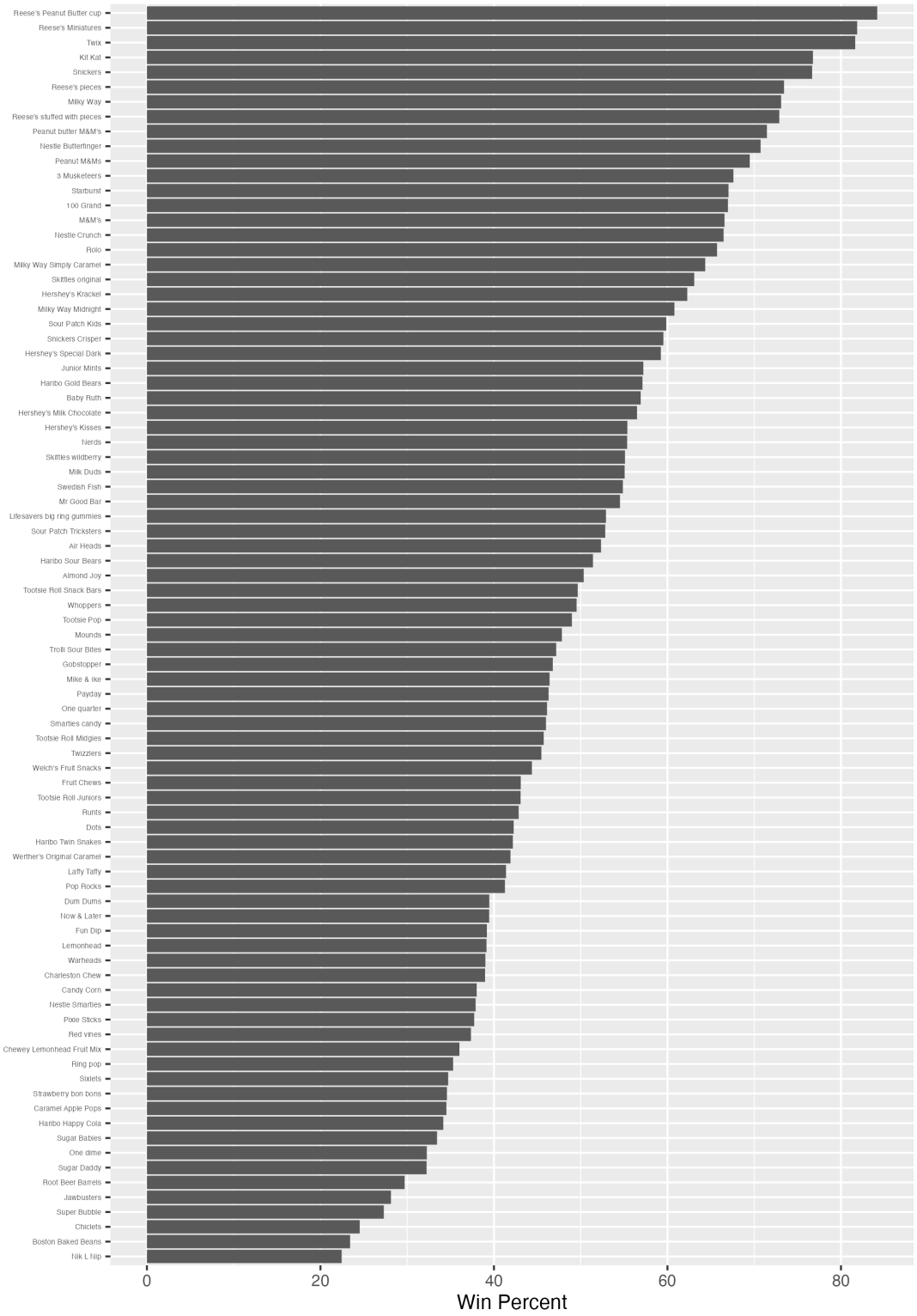
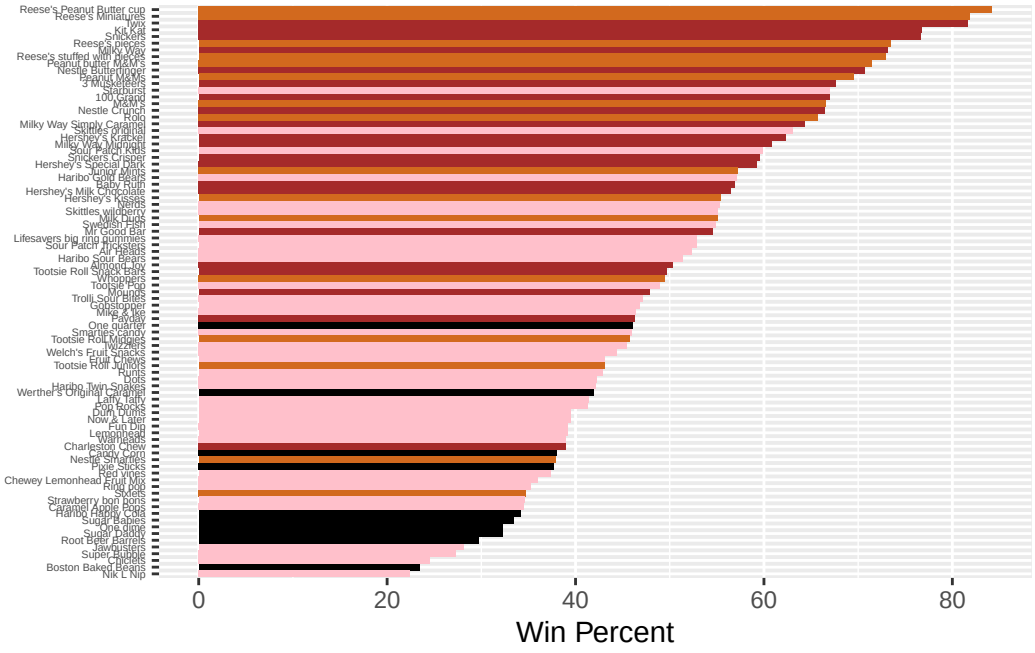


Figure 1: A Plot with better aspect ratio

Figure 2: An example image insertion

```
[55] "chocolate" "pink"      "chocolate" "black"     "pink"      "chocolate"
[61] "pink"      "pink"      "chocolate" "pink"      "brown"     "brown"
[67] "pink"      "pink"      "pink"      "pink"      "black"     "black"
[73] "pink"      "pink"      "pink"      "chocolate" "chocolate" "brown"
[79] "pink"      "brown"     "pink"      "pink"      "pink"      "black"
[85] "chocolate"
```

```
ggplot(candy)+
  aes(winpercent, reorder(rownames(candy), winpercent))+
  geom_col(width = 0.9, fill=my_cols) +
  labs(x="Win Percent") +
  theme(axis.text.y = element_text(size = 4), axis.title.y = element_blank())
```



Q17. What is the worst ranked chocolate candy?

A: Sixlets

Q18. What is the best ranked fruity candy?

A: Starbursts

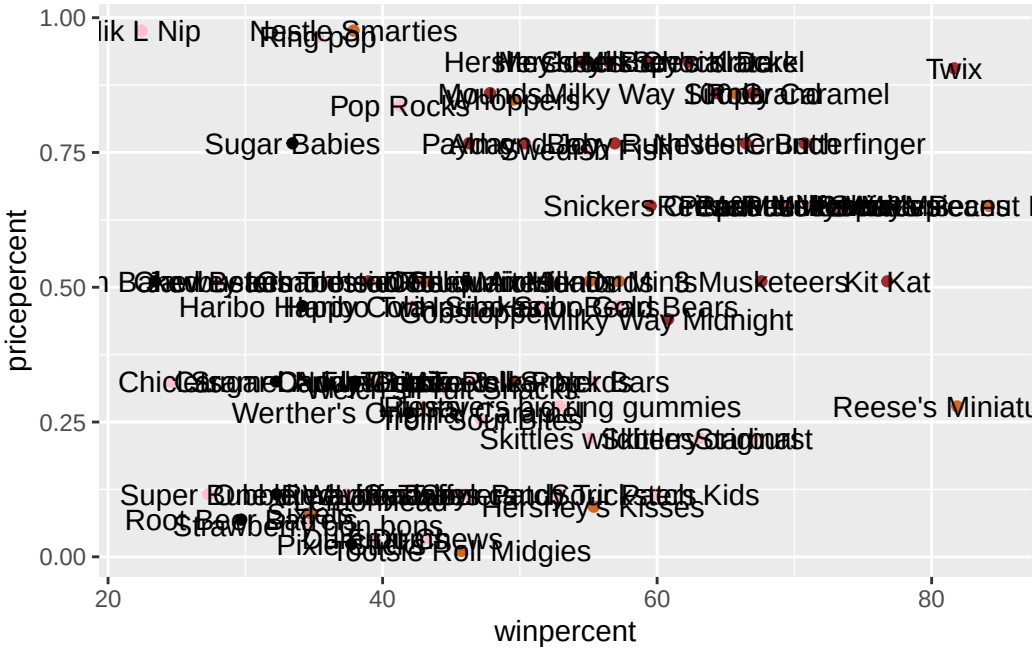
Take a look at pricepercent

```
candy$pricepercent
```

```
 [1] 0.860 0.511 0.116 0.511 0.511 0.767 0.767 0.511 0.325 0.325 0.511 0.511
[13] 0.325 0.511 0.034 0.034 0.325 0.453 0.465 0.465 0.465 0.465 0.093 0.918
[25] 0.918 0.918 0.511 0.511 0.511 0.116 0.104 0.279 0.651 0.651 0.325 0.511
[37] 0.651 0.441 0.860 0.860 0.918 0.325 0.767 0.767 0.976 0.325 0.767 0.651
[49] 0.023 0.837 0.116 0.279 0.651 0.651 0.651 0.965 0.860 0.069 0.279 0.081
[61] 0.220 0.220 0.976 0.116 0.651 0.651 0.116 0.116 0.220 0.058 0.767 0.325
[73] 0.116 0.755 0.325 0.511 0.011 0.325 0.255 0.906 0.116 0.116 0.313 0.267
[85] 0.848
```

If we want to see what is good candy to buy in terms of winpercent and pricepercent we can plot these two variables and then see the best candy for the least amount of money.

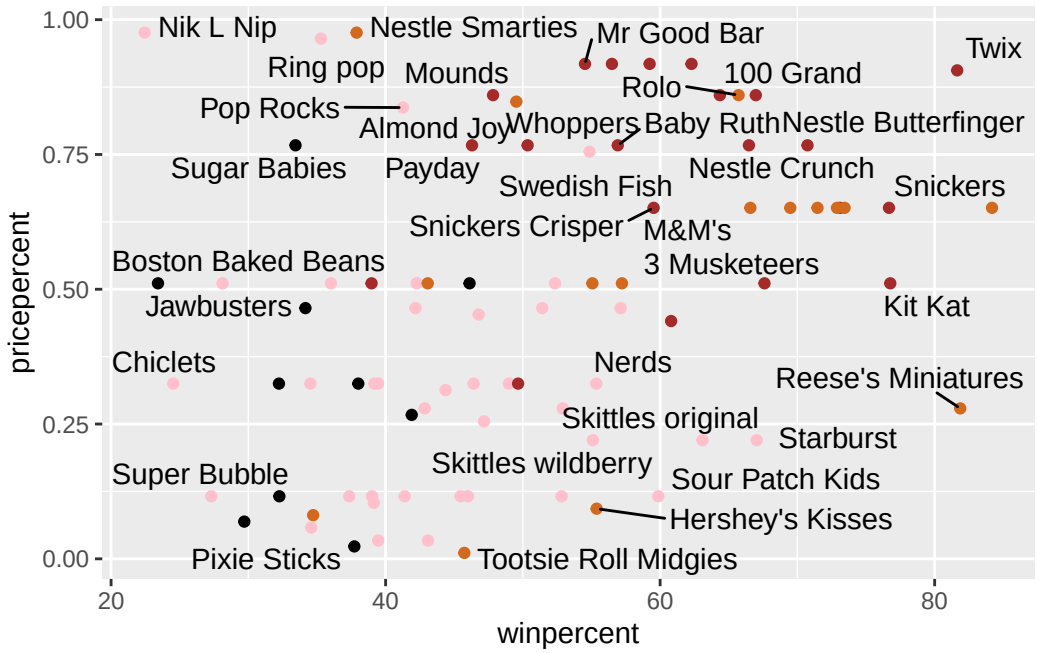
```
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy))+
  geom_point(col=my_cols)+
  geom_text()
```



To avoid the overplotting of all these labels we can use an add on package called ggrepl

```
library(ggrepel)

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

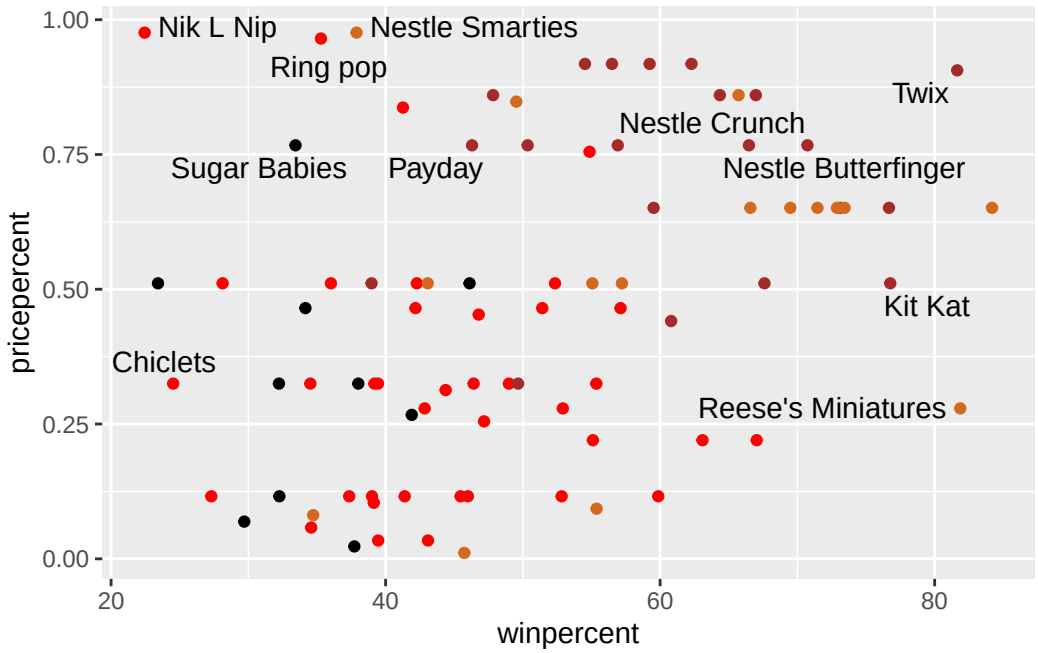
```
Warning: ggrepel: 50 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Play with the `max.overlaps` parameter to `geom_text_repel()`

```
# since fruity candies are hard to see under pink
my_cols[as.logical(candy$fruity)] = "red"

ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy))+
  geom_point(col=my_cols)+
  geom_text_repel(max.overlaps = 5)
```

```
Warning: ggrepel: 74 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

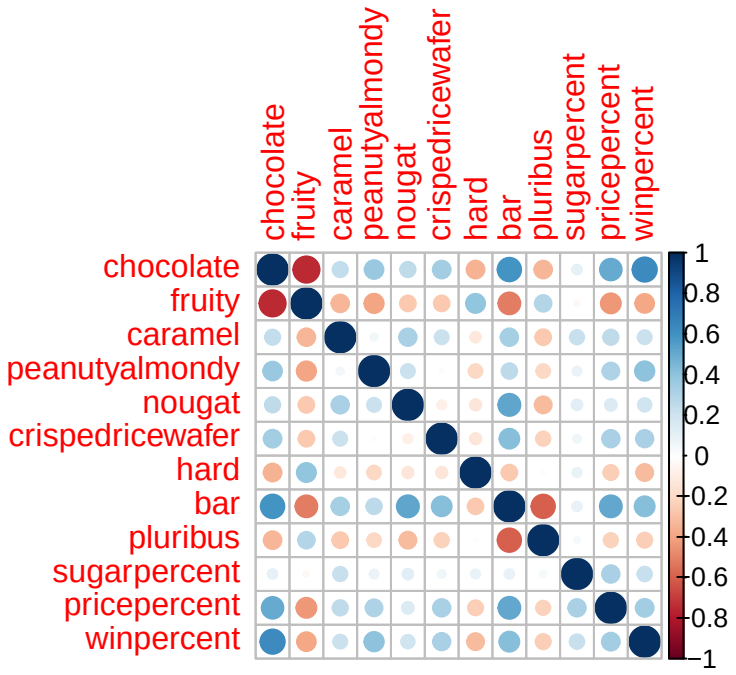


5 Exploring the correlation structure

```
library(corrplot)
```

```
corrplot 0.92 loaded
```

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



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

A: Being fruity and chocolate at the same time.

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

A: chocolate and win percent, if the candy is chocoalte it is more likely to win.

Onto PCA

The main function for this is called `prcomp()` and here we know we need to scale our data with the `scale=T` argument.

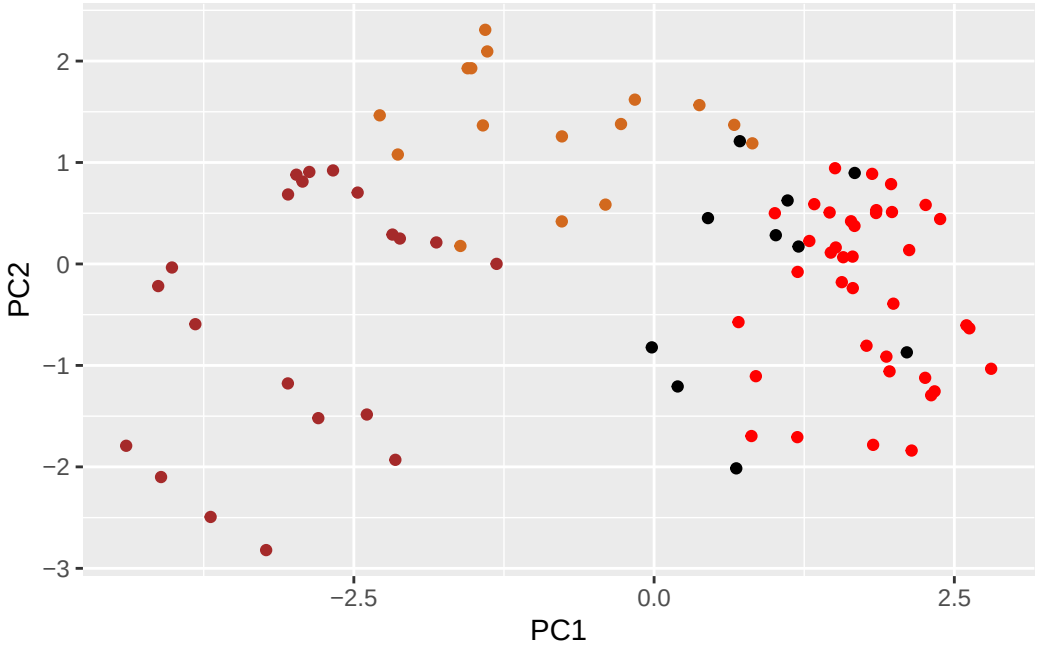
```
pca <- prcomp(candy, scale=T)
summary(pca)
```

```
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
```

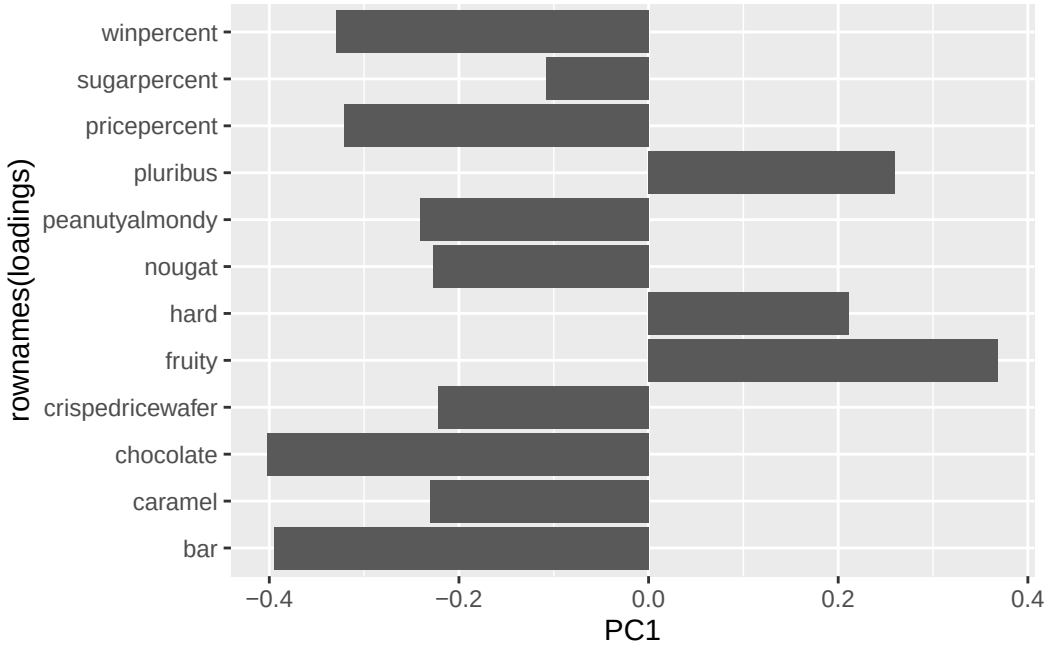
Plot my main PCA score plot with ggplot

```
# Make a new data-frame with our PCA results abd candy data
my_data <- cbind(candy, pca$x[,1:3])

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

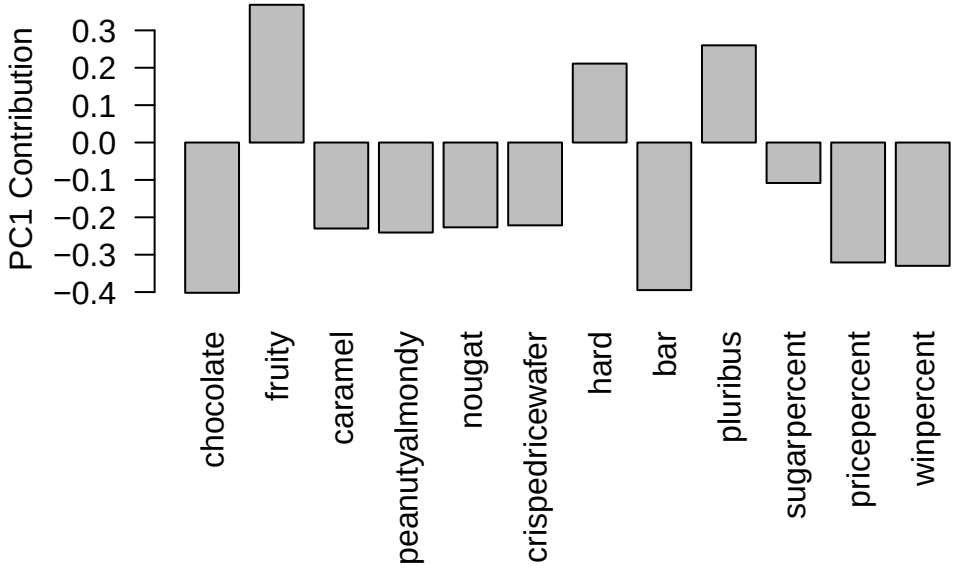


loadings plot

```
loadings <- as.data.frame(pca$rotation)

ggplot(loadings)+
  aes(PC1, rownames(loadings))+
  geom_col()
```




```
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?

A: Fruity, hard, and pluribus; and yes they make sense to me, fruity candies are more likely to be hard and pluribus relative to other candy types.