Halloween Mini Project

Andy Hsu

Analyzing Public Candy Preferences

Initial Data Analysis

The first step, as always, is to download the file.

```
candy_file <- "candy-data.csv"

candy = read.csv("https://raw.githubusercontent.com/fivethirtyeight/data/master/candy-powe
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
```

Taking a quick glance at our dataset, we can see that there are 85 candies in the data set, 38 of which are fruity.

```
nrow(candy)
```

```
[1] 85
```

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

```
[1] 38
```

Looking at individual data points, we find the corresponding win rates for each of the following candies, including Warheads, my personal favorite.

```r
candy["Warheads","winpercent"]
```

```
[1] 39.0119
```

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

```
[1] 76.7686
```

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

```
[1] 49.6535
```

If we use the **skimr** package, we can find even more information on the data set.

```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

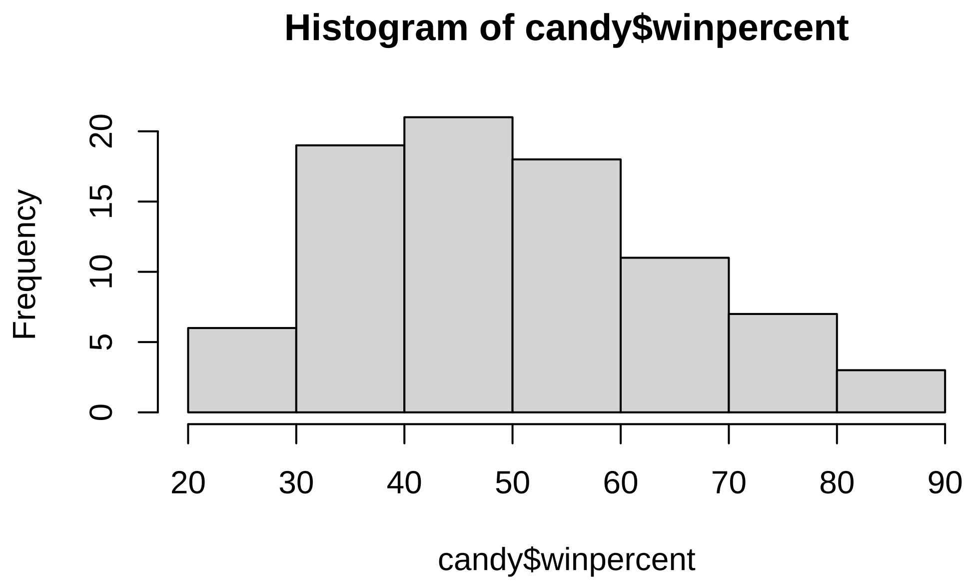
Variable type: numeric

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	

Judging from the results, we can tell that win percent is on a different scale from the rest of the set. We can also assume that 0 and 1 indicate true or false for whether a candy is chocolately, for instance.

Next, we can plot some data to get an idea of distributions. We'll start with a histogram of win percents.

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



We can see from the distribution that it is not symmetrical, and that the center of the distribution is below 50%.

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

```
[1] 60.92153
```

```
mean(candy$winpercent[as.logical(candy$fruity)])
```

```
[1] 44.11974
```

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

```
    Welch Two Sample t-test

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

From the above code, we can tell that chocolate candies are rated higher on average than fruity candies, and that the difference is statistically significant, with a p-value of 2.9e-8.

Candy Rankings

Now, using the **dplyr** package, we can find the top 5 and bottom 5 candies based on win percent in this dataset.

```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
# Bottom 5
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
```

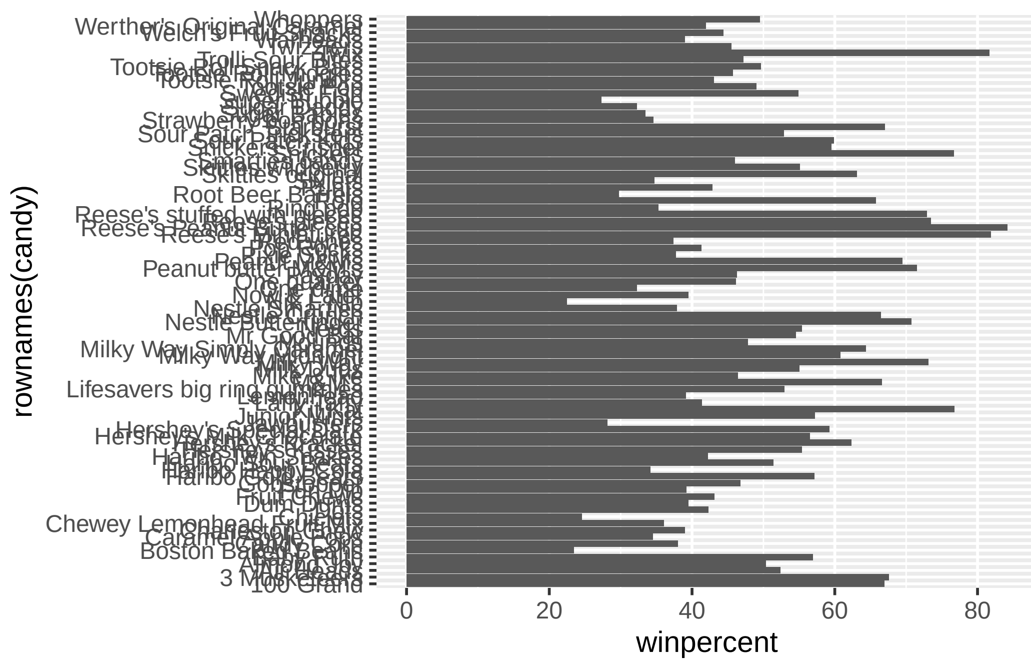
```
Chiclets                         0    0   0        1       0.046        0.325
Super Bubble                     0    0   0        0       0.162        0.116
Jawbusters                       0    1   0        1       0.093        0.511
                   winpercent
Nik L Nip            22.44534
Boston Baked Beans   23.41782
Chiclets             24.52499
Super Bubble         27.30386
Jawbusters           28.12744
```

```
# Top 5
candy %>% arrange(-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
```

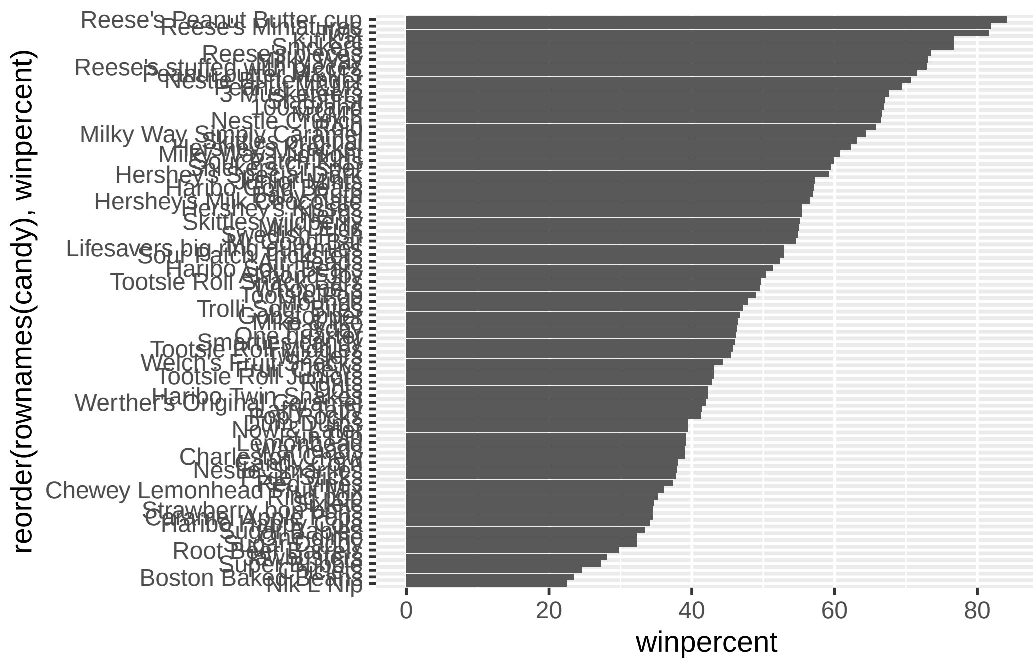
Now, we can use ggplot to plot a bar graph of all the candies according to win rate.

```
library("ggplot2")

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



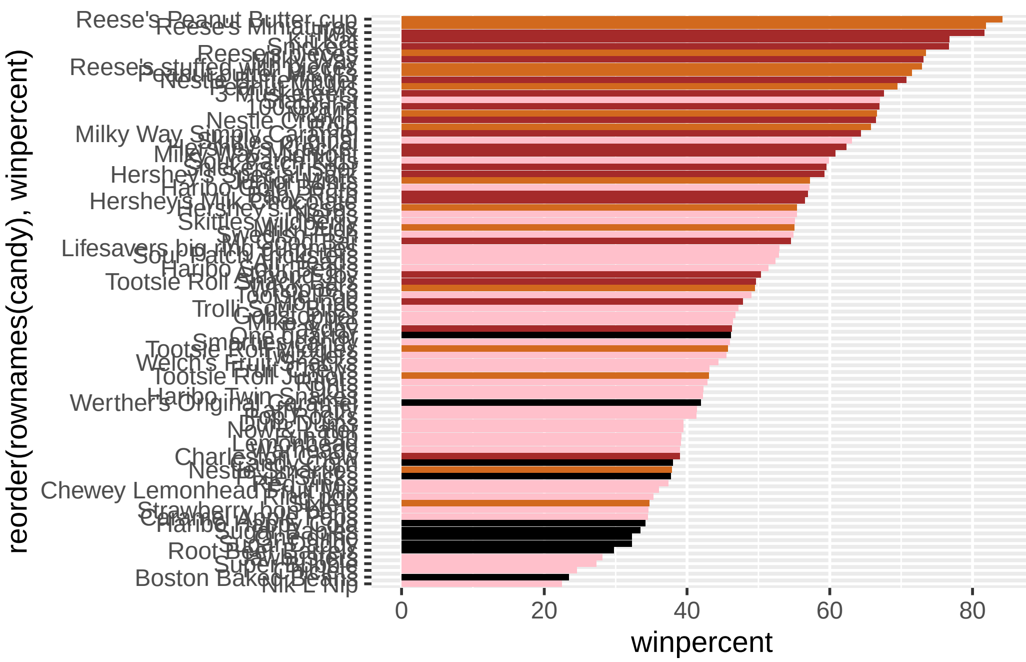
To order by winpercent, we can edit our code.

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



Next, we can label bar colors based on the type of candy. We first create a dataset with the corresponding colors we want, then apply it to the graph.

```
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_col
```



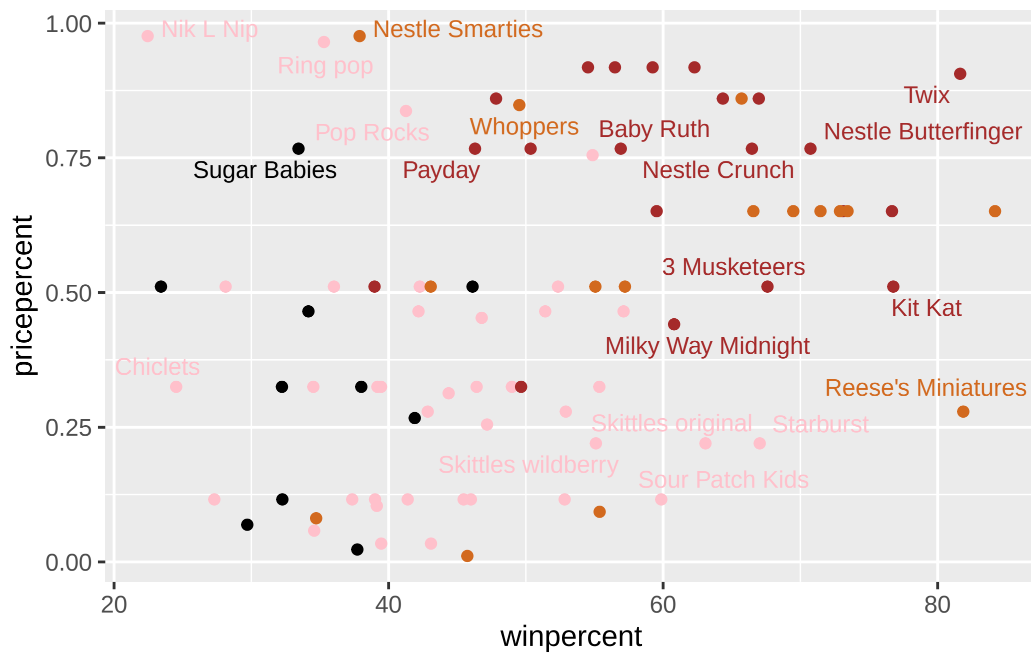
From this informative plot, we can observe that the worst ranked chocolate candy is Sixlets, and the highest ranked fruity candy is Starburst.

Looking at Price Percent

To determine if price plays a part in the winpercent of a candy, we can plot winpercent against pricepercent. In this graph, we will use the **ggrepel** package to ensure no labels overlap.

```
library("ggrepel")

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
```



Looking at the results, we can see that Reese's Miniatures offers the most bang for your buck, with a high winpercent and low pricepercent. We can also look at the 5 most expensive candies, finding that Nik L Nip is the least popular of these.

```
price <- candy %>% arrange(pricepercent) %>% tail(5)
price["winpercent"]
```

```
                      winpercent
Hershey's Special Dark   59.23612
Mr Good Bar              54.52645
Ring pop                 35.29076
Nik L Nip                22.44534
Nestle Smarties          37.88719
```

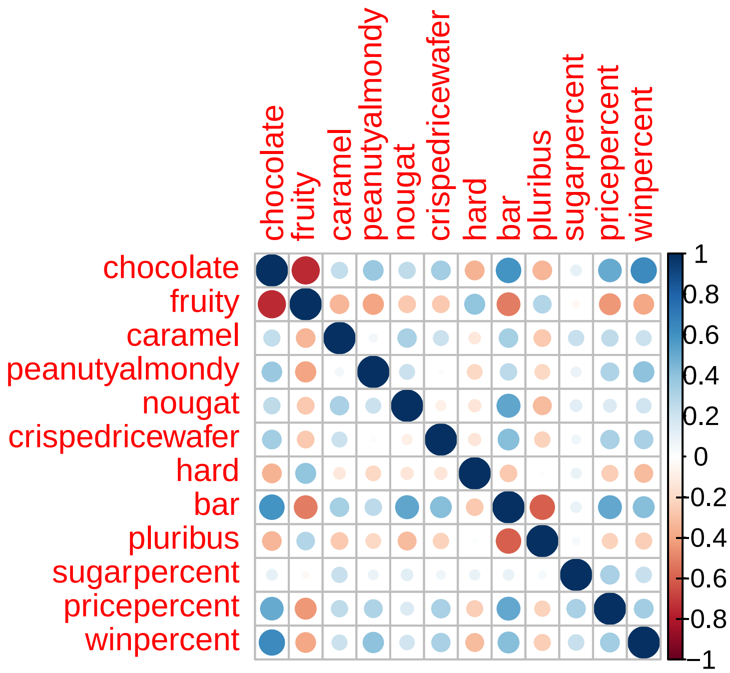
Correlation Structure

Next, we will use the **corrplot** package to plot and analyze a correlation plot to gain more knowledge on the dataset.

```
library("corrplot")
```

```
corrplot 0.92 loaded
```

```r
corrplot(cor(candy))
```



From this graph, we can see that the two most inversely correlated variables are chocolate and fruity. Conversely, the two most positivly correlated variables are chocolate and bar.

Principal Component Analysis

Finally, we can perform PCA on this data set to obtain an idea of relationship between individual candies.

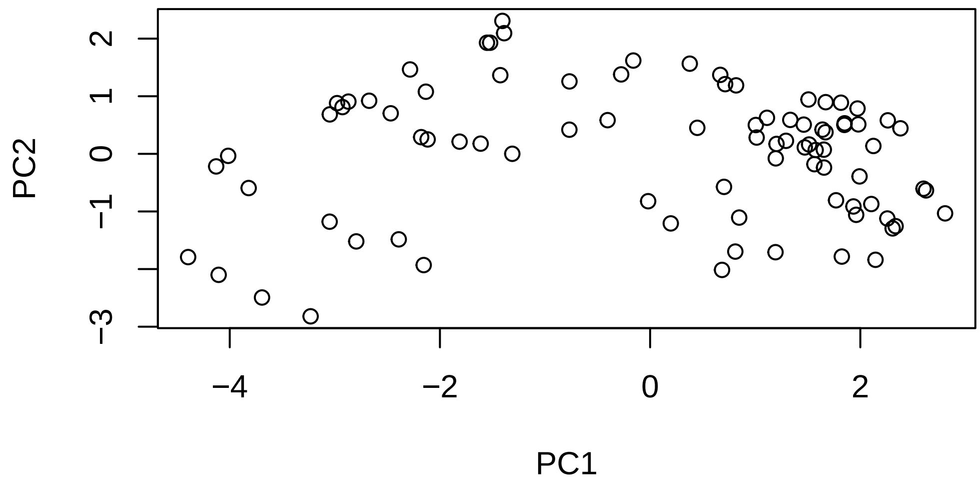
```r
pca <- prcomp(candy,scale=T)
summary(pca)
```

```
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
```

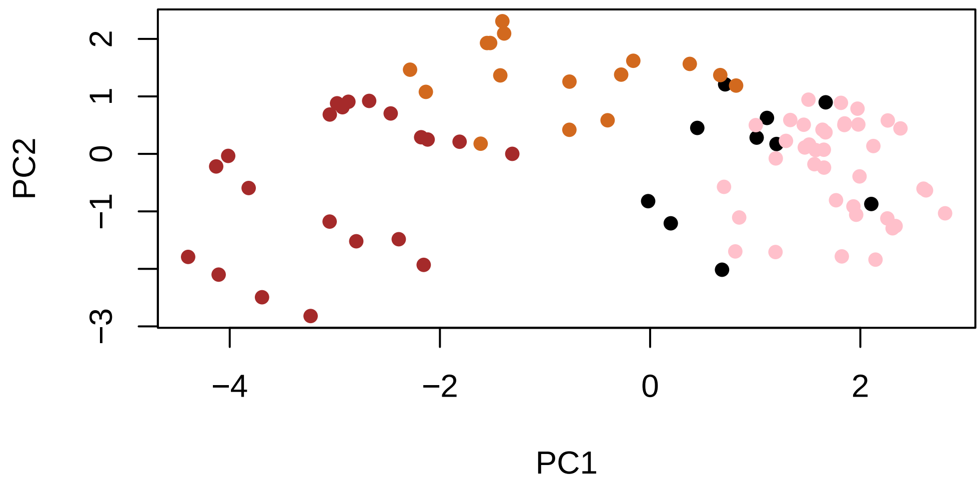
Now, we can plot our PC1 vs PC2 plot.

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



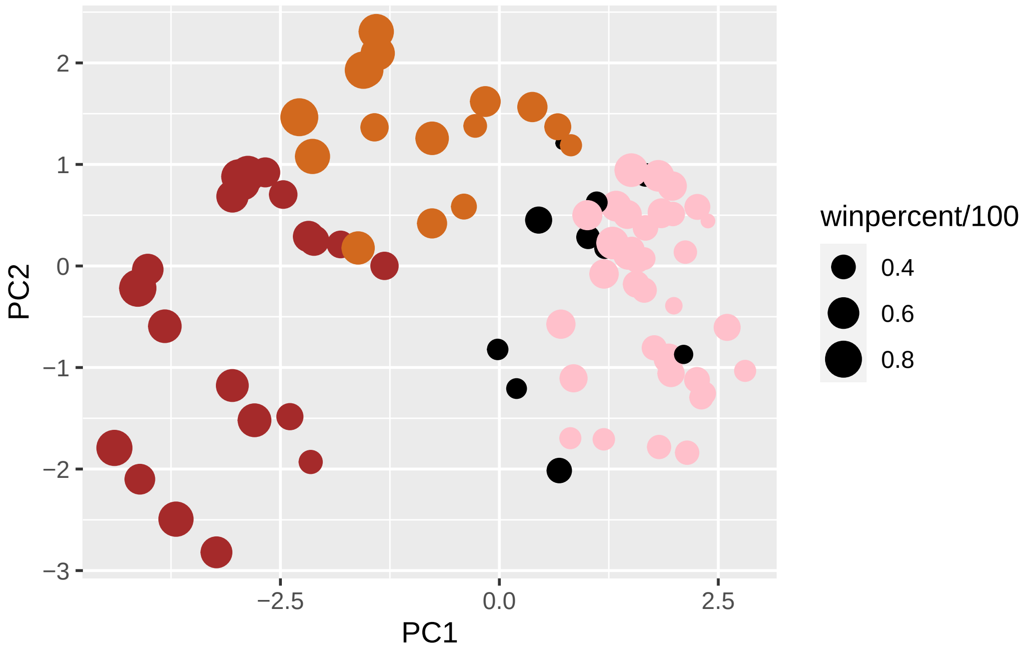
We can add our colors from our earlier bar graph.

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



Let's convert this code to ggplot and a size indicating win rate.

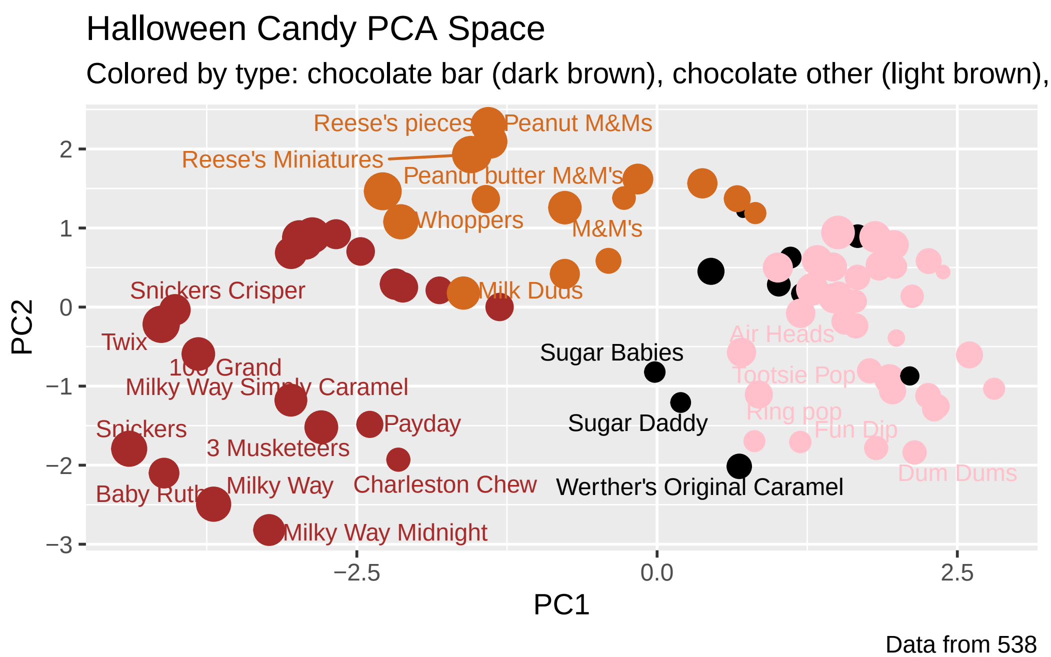
```
cdf <- cbind(candy, pca$x[,1:3])

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



We can also add labels to the points to more clearly indicate individual candies.

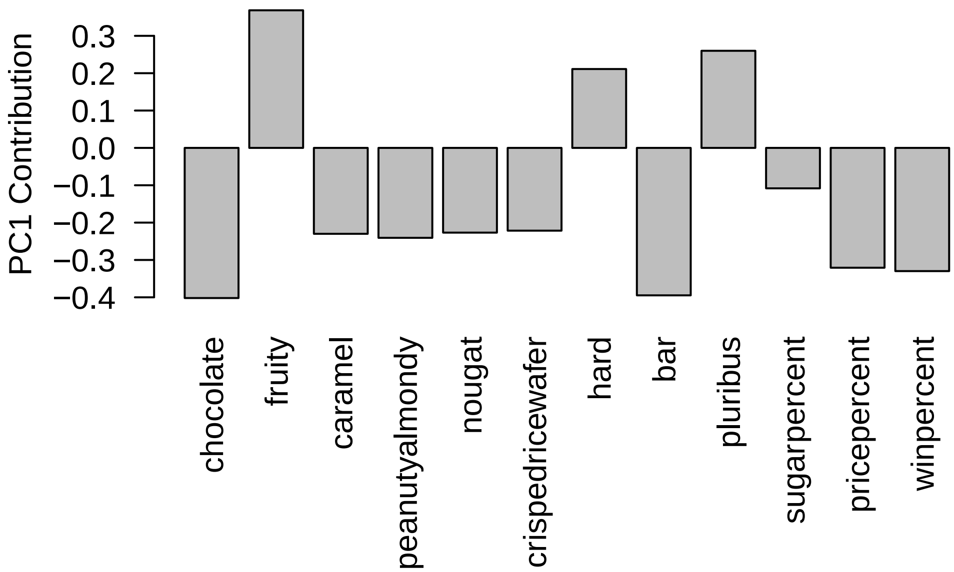
```
library(ggrepel)

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
```



Lastly, let's look at our loadings for the PCA.

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



From the plot, we can tell that the most positive variables were fruity, hard, and plubirus. This makes sense, as most fruity candies are hard and come in packets of many.

And that concludes our analysis of this dataset of popular candies.