Class 09: Halloween Mini Project

Kristiana Wong A16281367

##Background In this mini-project, you will explore FiveThirtyEight's Halloween Candy dataset. Your task is to explore their candy dataset to find out answers to a various number of questions - but most of all your job is to have fun, learn by doing hands on data analysis, and hopefully make this type of analysis less frightining for the future! Let's get started.

##1. Importing candy dataset 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. Make sure to download the csv file first and redirect it into your Project Class 09 folder.

```
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?

```
nrow(candy)
```

```
[1] 85
```

85 types of candy in this data set.

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

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

```
[1] 38
```

There are 38 fruity candy types.

##2. What is your favorite 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.

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

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

```
[1] 76.7686
```

76.7686%

Q4. What is the winpercent value for "Kit Kat"? 76.7686%

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

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

```
[1] 49.6535
```

49.6535%

There is a useful skim() function in the skimr package that can help give you a quick overview of a given dataset. Let's install this package and try it on our candy data.

```
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

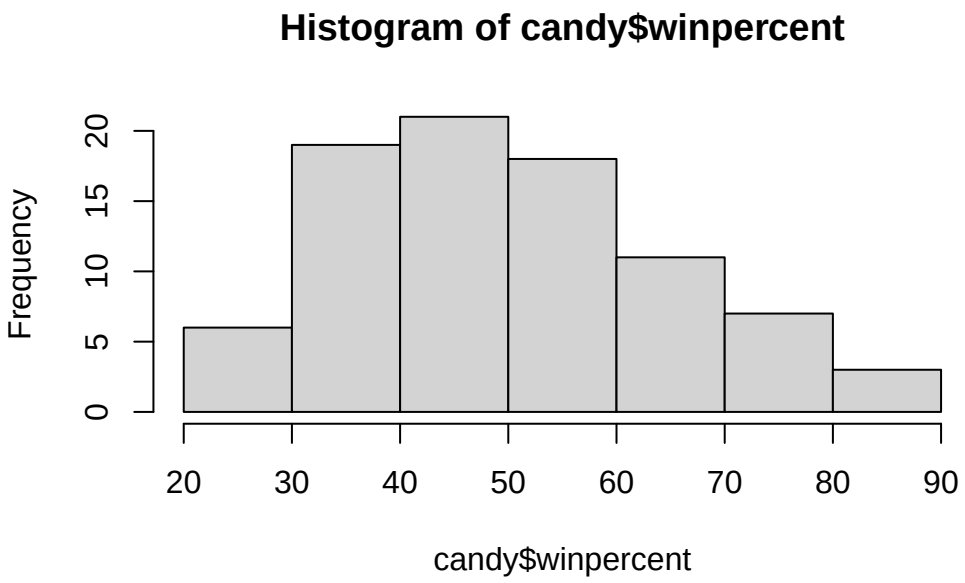
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
chocolate	0	1	0.44	0.50	0.00	0.00	0.00	1.00	1.00	
fruity	0	1	0.45	0.50	0.00	0.00	0.00	1.00	1.00	
caramel	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
peanutyalmondy	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
nougat	0	1	0.08	0.28	0.00	0.00	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? The winpercent variable looks to be on a different scale than the majority of the other columns in the data set.

Q7. What do you think a zero and one represent for the candy$chocolate column I think the 0 represents the number of missing variables within the chocolate data in the candy data set. The 1 represents the percentage of non-missing variables in the chocolate data in the candy data set (i.e complete chocolate data).

Q8. Plot a histogram of winpercent values

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



Q9. Is the distribution of winpercent values symmetrical? No, it is not. It is normally distributed, as can be seen the data is skewed to the left.

Q10. Is the center of the distribution above or below 50%? The center of distribution seems to be just below 50%.

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

```
mean(candy$chocolate)
```

```
[1] 0.4352941
```

```
mean(candy$fruity)
```

```
[1] 0.4470588
```

On average, chocolate candy is ranked lower than fruity candy.

Q12. Is this difference statistically significant?

```
t.test(candy$chocolate, candy$fruity)
```

```
    Welch Two Sample t-test

data:  candy$chocolate and candy$fruity
t = -0.15357, df = 168, p-value = 0.8781
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1630081  0.1394786
sample estimates:
mean of x mean of y
0.4352941 0.4470588
```

This difference is not statistically significant, as can be seen the p-value is 0.8781– very high and not significant at all.

##3. Overall Candy Rankings Let's use the base R order() function together with head() to sort the whole dataset by winpercent.

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

```
head(candy[order(candy$winpercent, decreasing = F),], 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
```

Nik L Nip, Boston Baked Beans, Chiclets, Super Bubble and Jawbusters are the 5 least liked candy types.

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

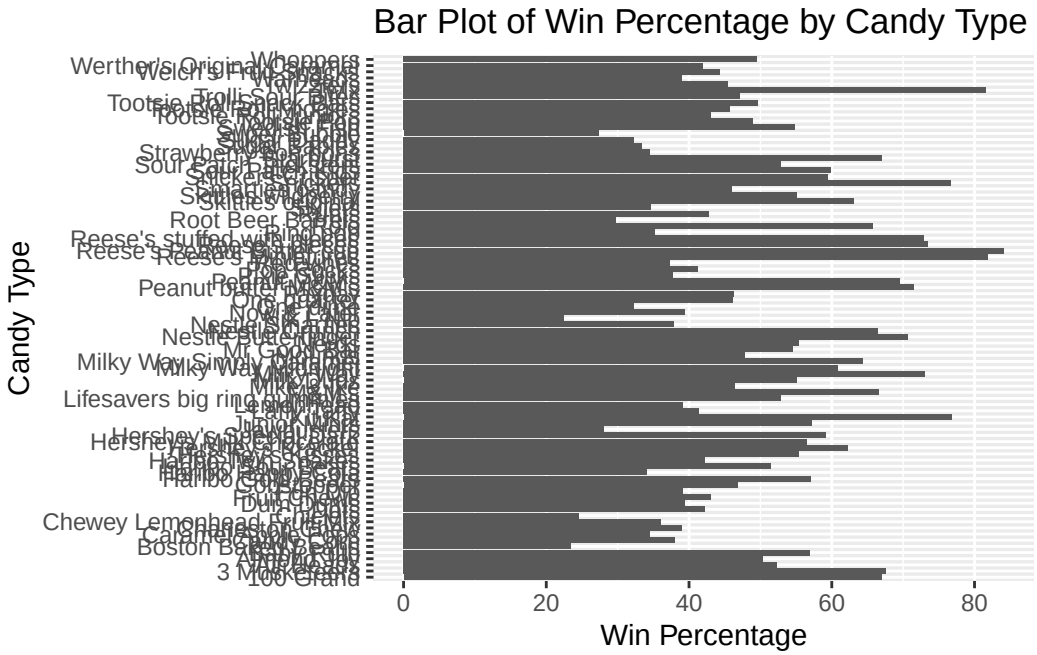
```
head(candy[order(candy$winpercent, decreasing = T),], n=5)
```

```
                         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
```

Snickers, Kit Kat, Twix, Reese's Minatures, and Reese's Peanut Butter Cups are the 5 most liked candy types.

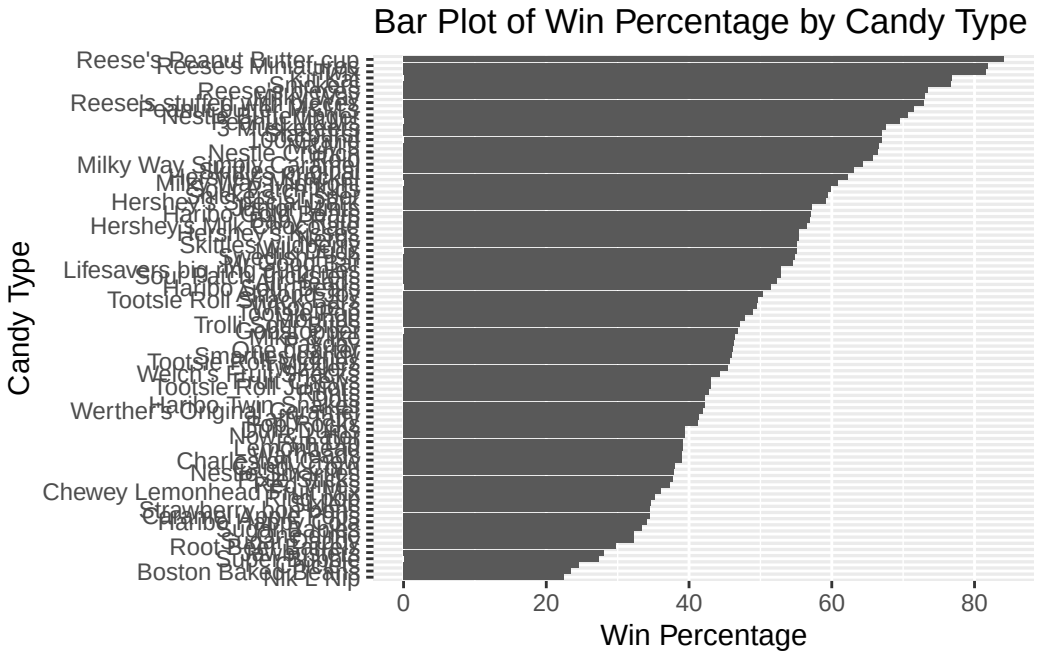
Q15. Make a first barplot of candy ranking based on winpercent values.

```
library(ggplot2)

ggplot(candy, aes(x = winpercent, y = rownames(candy))) +
  geom_bar(stat = "identity") +
  labs(x = "Win Percentage", y = "Candy Type", title = "Bar Plot of Win Percentage by Cand
```



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

```
library(ggplot2)

ggplot(candy, aes(x = winpercent, y = reorder(rownames(candy), winpercent))) +
  geom_bar(stat = "identity") +
  labs(x = "Win Percentage", y = "Candy Type", title = "Bar Plot of Win Percentage by Cand
```

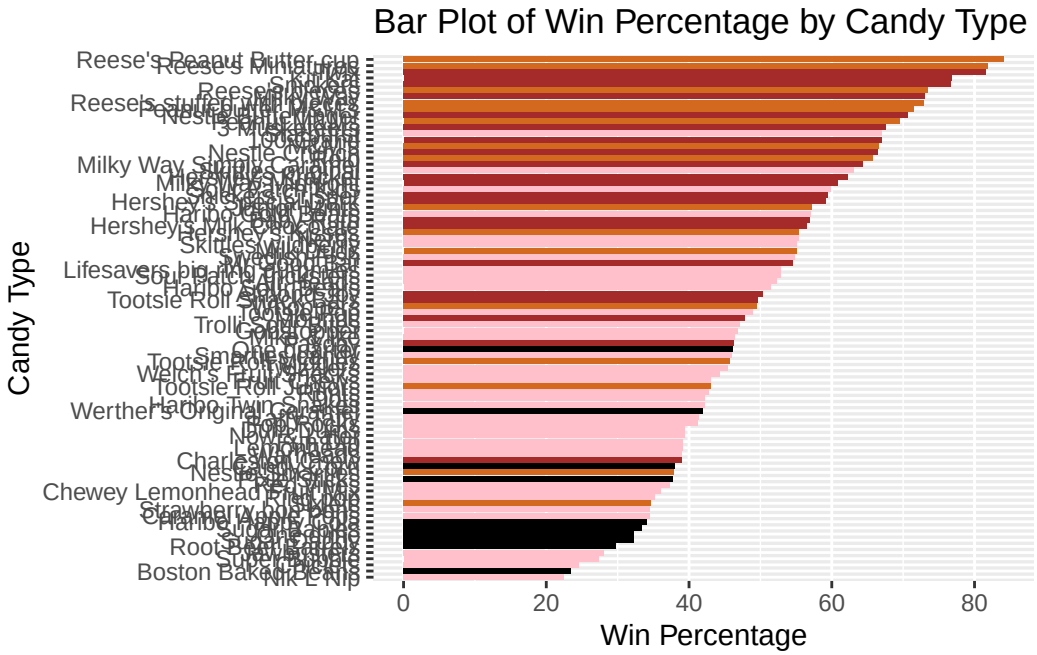


#Time to add some useful color. Let's setup a color vector (that signifies candy type) that we can then use for some future plots. We start by making a vector of all black values (one for each candy). Then we overwrite chocolate (for chocolate candy), brown (for candy bars) and red (for fruity candy) 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(). Experiment to see what happens if you use col=mycols.

```
library(ggplot2)

ggplot(candy, aes(x = winpercent, y = reorder(rownames(candy), winpercent))) +
  geom_bar(stat = "identity") +
  geom_col(fill = my_cols) +
  labs(x = "Win Percentage", y = "Candy Type", title = "Bar Plot of Win Percentage by Cand
```



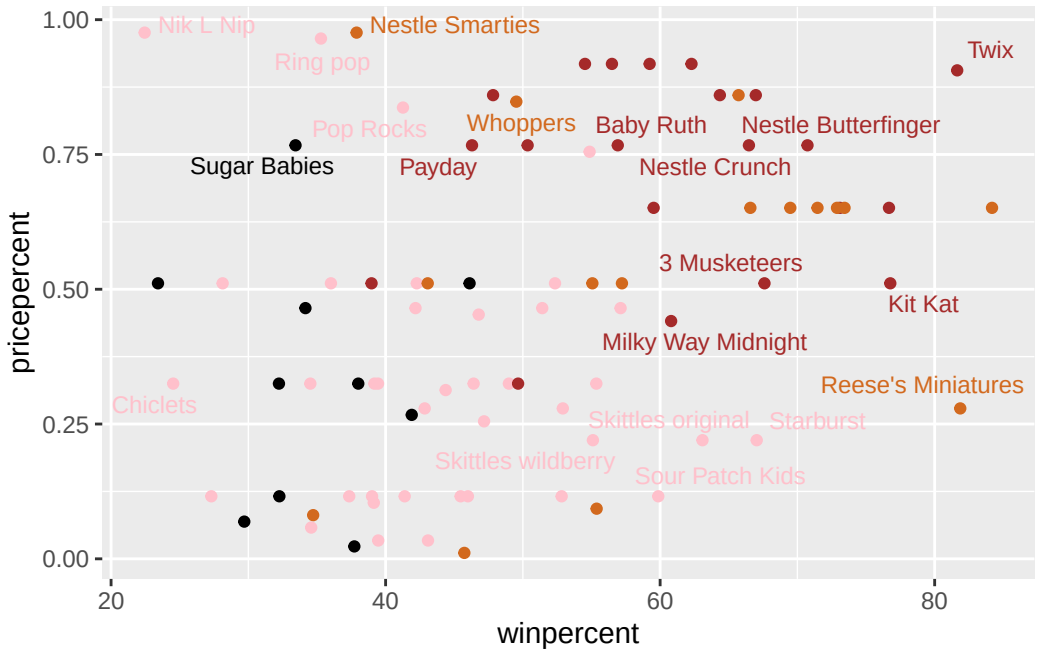
Q17. What is the worst ranked chocolate candy? Sixlets

Q18. What is the best ranked fruity candy? Starburst

##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.

```
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? Reese's Miniatures

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
```

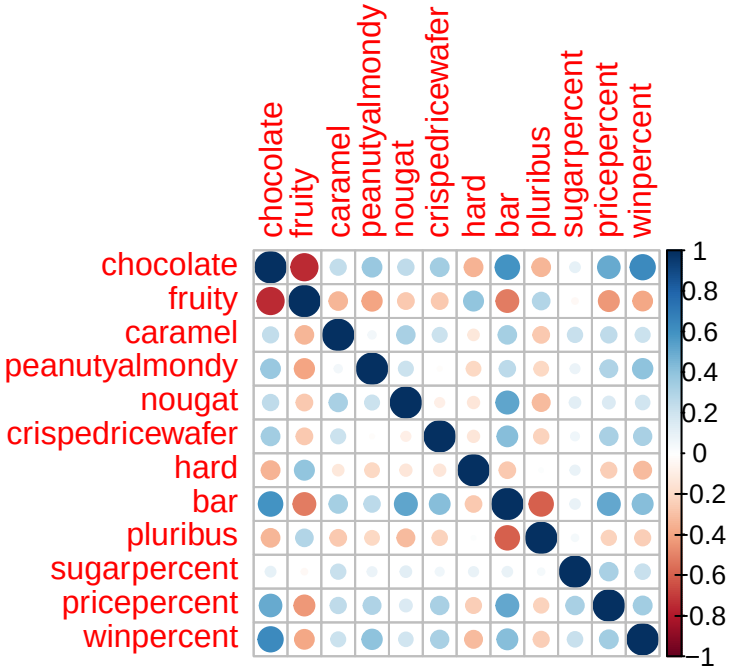
Hershey's Krackel, Nik L Nip, Nestle Smarties, Ring pop, and Hershey's Milk Chocolate. Nik L Nip is the least popular.

##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 0.92 loaded
```

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



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

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

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

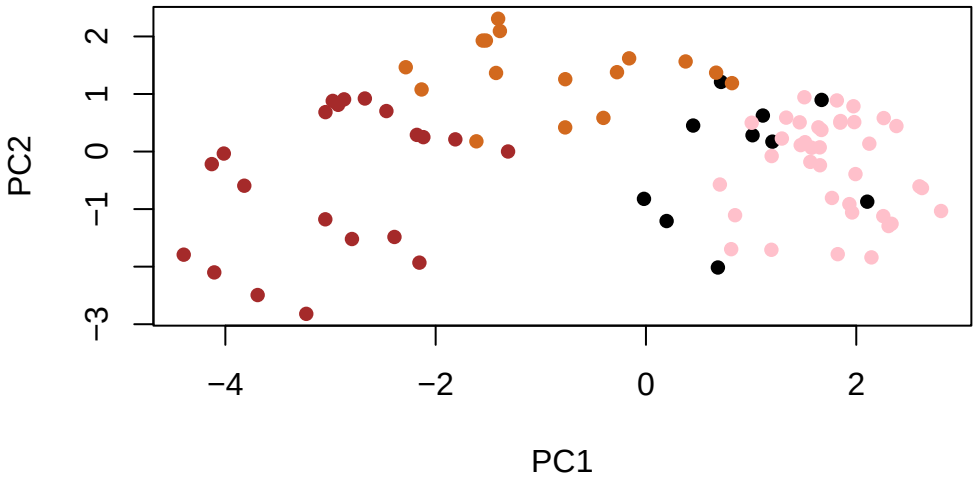
```
pca <- prcomp(candy, scale = TRUE)
summary(pca)
```

```
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
```

Now we can plot our PCA1 vs PCA2

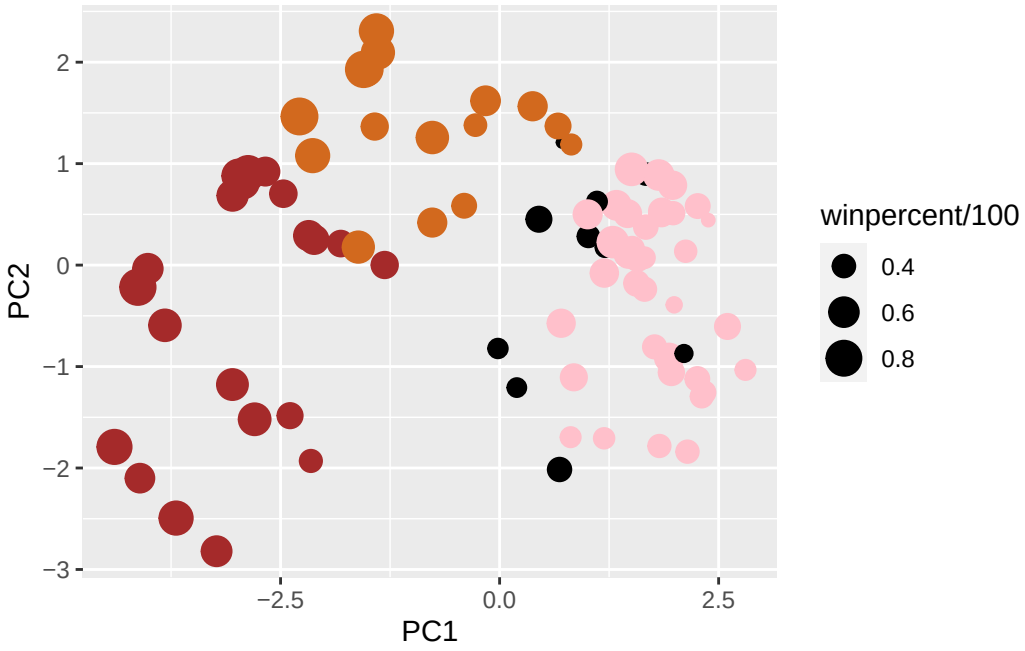
```
plot(pca$x[,1:2], xlab = "PC1", ylab = "PC2", col = my_cols, pch=16)
```



We can make a much nicer plot with the ggplot2 package but it is important to note that ggplot works best when you supply an input data.frame that includes a separate column for each of the aesthetics you would like displayed in your final plot. To accomplish this we make a new data.frame here that contains our PCA results with all the rest of our candy data. We will then use this for making plots below:

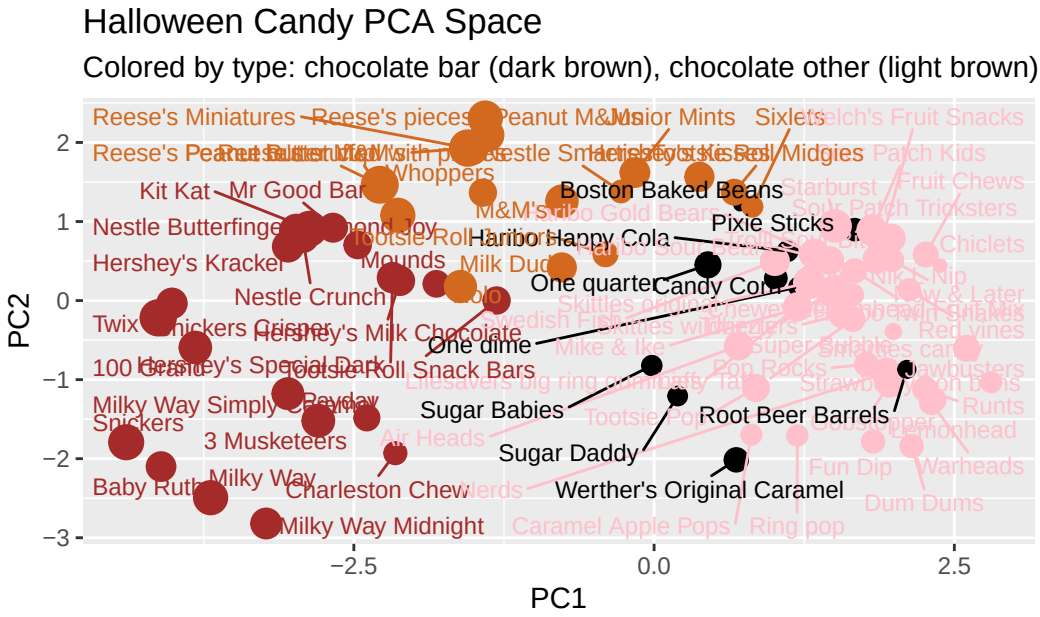
```
# Make a new data-frame with our PCA results and candy data
my_data <- cbind(candy, pca$x[,1:3])
```

```
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
```



Again we can use the ggrepel package and the function ggrepel::geom_text_repel() to label up the plot with non overlapping candy names like. We will also add a title and subtitle like so:

```
library(ggrepel)

p + geom_text_repel(size=3.3, col=my_cols, max.overlaps = 40)  +
  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")
```



Data from 538

```
library(plotly)
```

```
Attaching package: 'plotly'
```

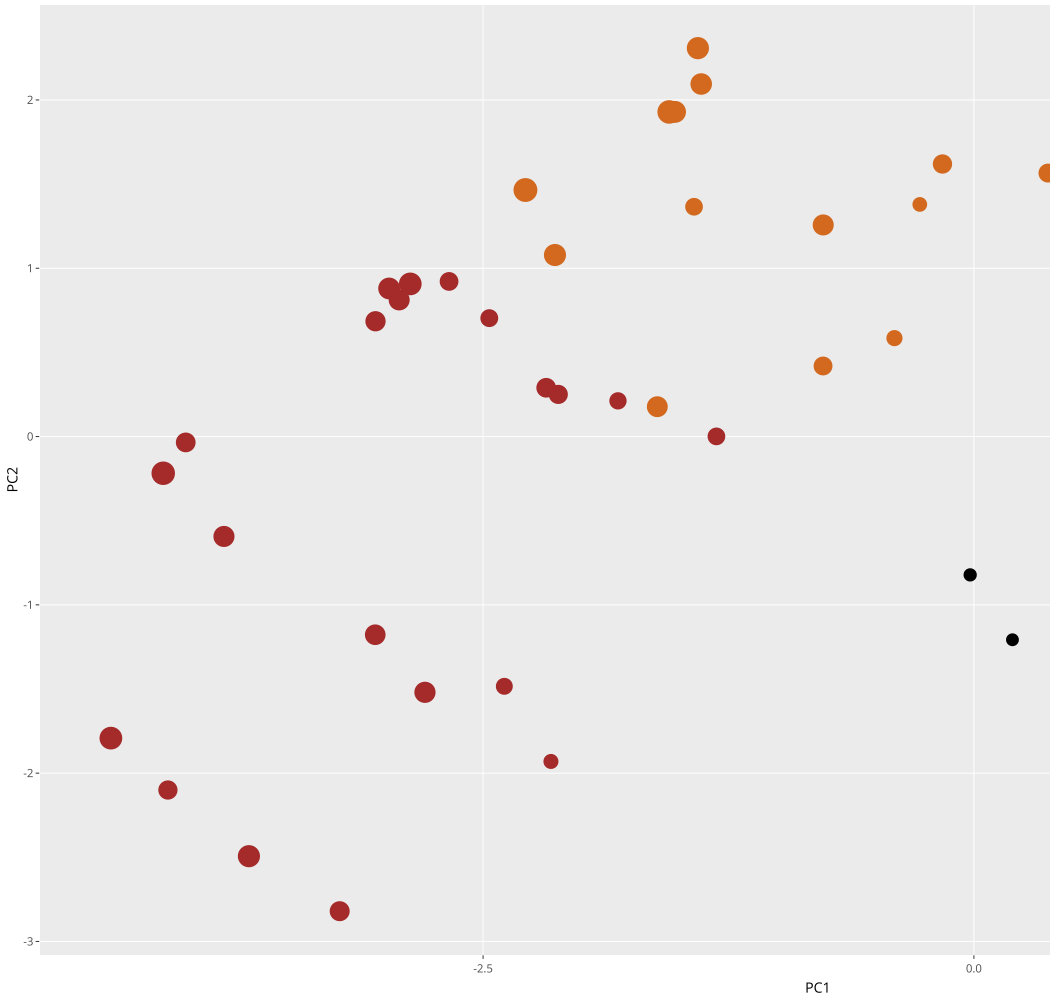
```
The following object is masked from 'package:ggplot2':

    last_plot
```

```
The following object is masked from 'package:stats':

    filter
```

```
The following object is masked from 'package:graphics':

    layout
```

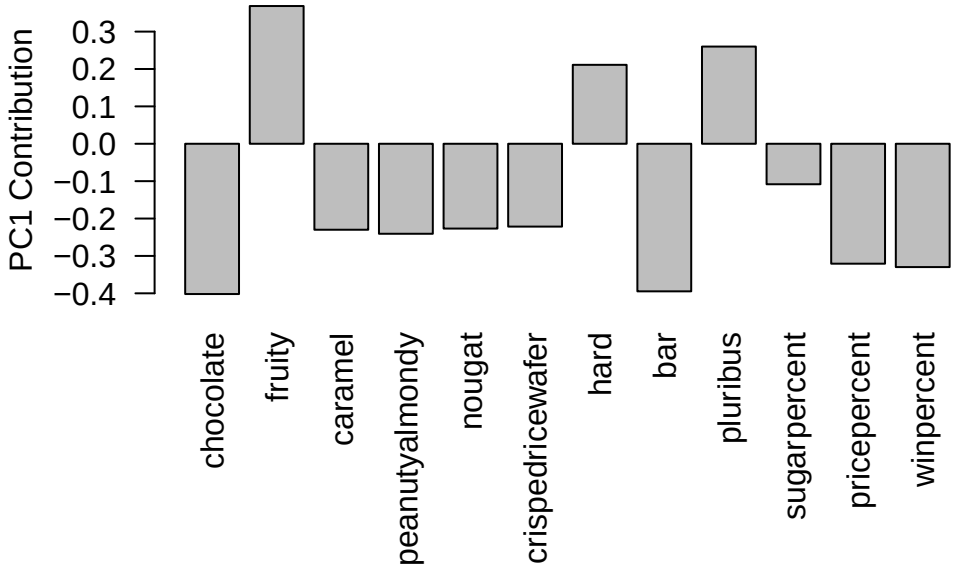
```
ggplotly(p)
```

```
PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installe
```


2
1
0
-1
-2
-3
PC2
-2.5
0.0
PC1

Let's finish by taking a quick look at PCA our loadings. DNotice the opposite effects of chocolate and fruity and the similar effects of chocolate and bar (i.e. we already know they are correlated).

```
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 and hard and pluribus are strongly picked up in the positive direction. These do make sense to me, as we know that the fruity candies such as Starbursts or Air Heads are usually harder candies that come in packages of multiple. Hence, these variables are strongly correlated to one another.