class10

Stefanie Hodapp (PID: A53300084)

10/29/2021

1. Importing candy data

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

```
nrow(candy) # Q1. There are 85 different candy types in this dataset
```

```
## [1] 85
```

```
table(candy$fruity) # Q2. There are 38 fruity candy types
```

```
##
##  0  1
## 47 38
```

2. What is your favorite candy?

Q3. What is your favorite candy in the dataset and what is it's winpercent value? Q4. What is the winpercent value for "Kit Kat"? Q5. What is the winpercent value for "Tootsie Roll Snack Bars"?

```
candy["ReeseÕs Peanut Butter cup", ]$winpercent # Q3. 84.18029%
```

```
## [1] 84.18029
```

```
candy["Kit Kat", ]$winpercent # Q4. 76.7686%
```

```
## [1] 76.7686
```

```
candy["Tootsie Roll Snack Bars", ]$winpercent # Q5. 49.6535%
```

```
## [1] 49.6535
```

Use the skim() function in the skimr package to give a quick overview of the candy dataset.

```
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	

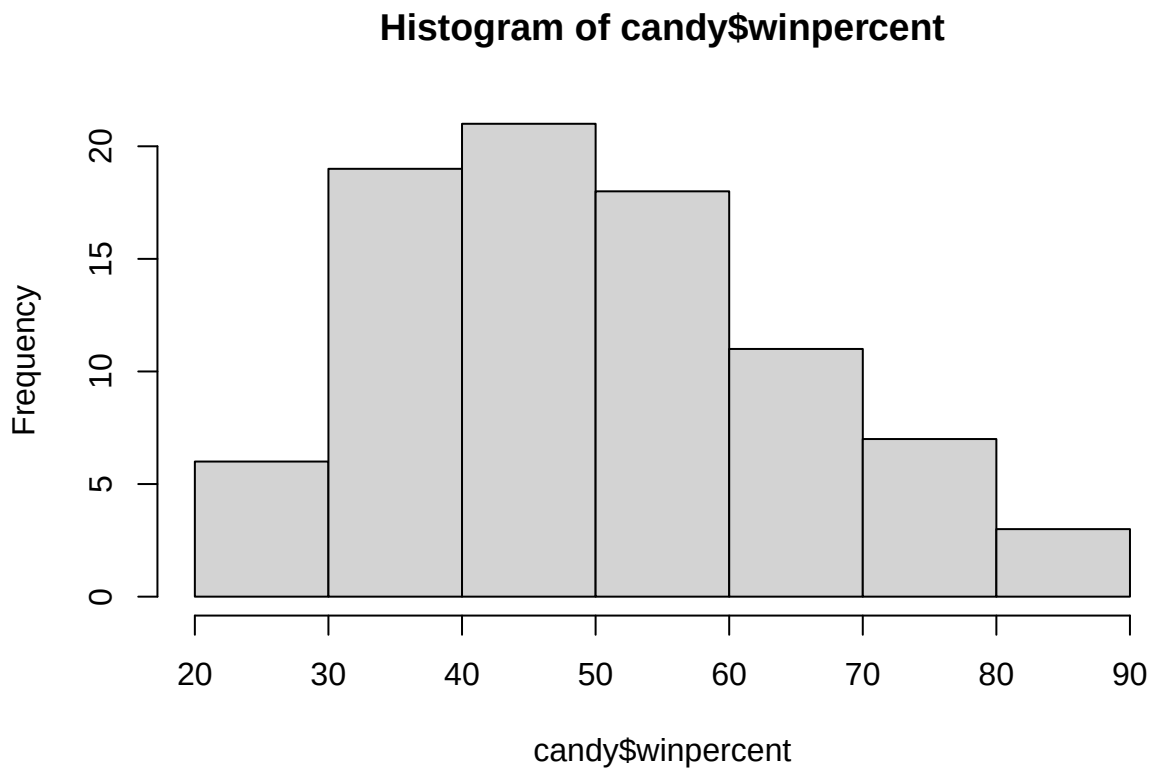
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
winpercent	0	1	50.32	14.71	22.45	39.14	47.83	59.86	84.18	

Q6. Is there any variable/column that looks to be on a different scale to the majority of the other columns in the dataset? There are 3 variables that are on a different scale to the others: sugarpercent, pricepercent, and winpercent. These variables are on a 0 to 1 scale representing percentages, whereas whereas the other variables are either 0 or 1 values.

Q7. What do you think a zero and one represent for the candy$chocolate column? A zero value represents that the candy does not contain chocolate, while a one value represents that the candy bar contains chocolate.

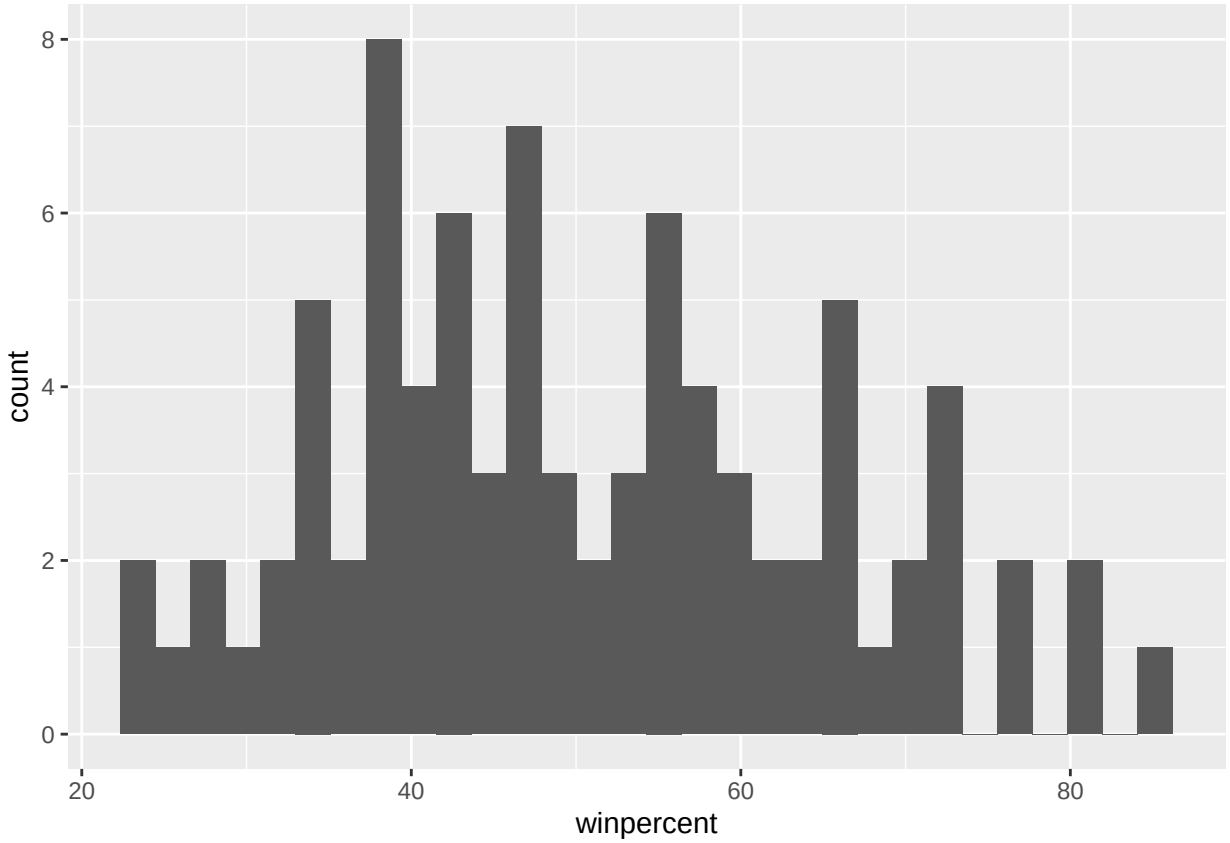
Q8. Plot a histogram of winpercent values Q9. Is the distribution of winpercent values symmetrical?

```
hist(candy$winpercent) # Plotting a histogram using hist()

library(ggplot2)
```




```
ggplot(candy) + aes(x=winpercent) + geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Plotting a histogram using ggplot()

# Q9. The distribution of the winpercent values is asymmetrical (it is slightly skewed to the right)
```

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

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

```
# Using the mean function to compare chocolate and fruity win percentages

mean_chocolate <- mean(candy$winpercent[as.logical(candy$chocolate)])
print(mean_chocolate)
```

```
## [1] 60.92153
```

```
mean_fruity <- mean(candy$winpercent[as.logical(candy$fruity)])
print(mean_fruity)
```

```
## [1] 44.11974
```

```
mean_chocolate > mean_fruity # Chocolate candy is higher ranked than fruit candy (60.92153 vs 44.11974)
```

```
## [1] TRUE
```

Q12. Is this difference statistically significant?

```
# Using the T test function to compare chocolate and fruity win percentages

x <- candy$winpercent[as.logical(candy$chocolate)]
y <- candy$winpercent[as.logical(candy$fruity)]
t.test(x,y) # This difference is statistically significant (p-value = 2.871e-08)
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## 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

Q13. What are the five least liked candy types in this set? Q14. What are the top 5 all time favorite candy types out of this set?

```
head(candy[order(candy$winpercent),], n=5) # 5 least liked candy types: Nik L Nip, Boston Baked Beans,
```

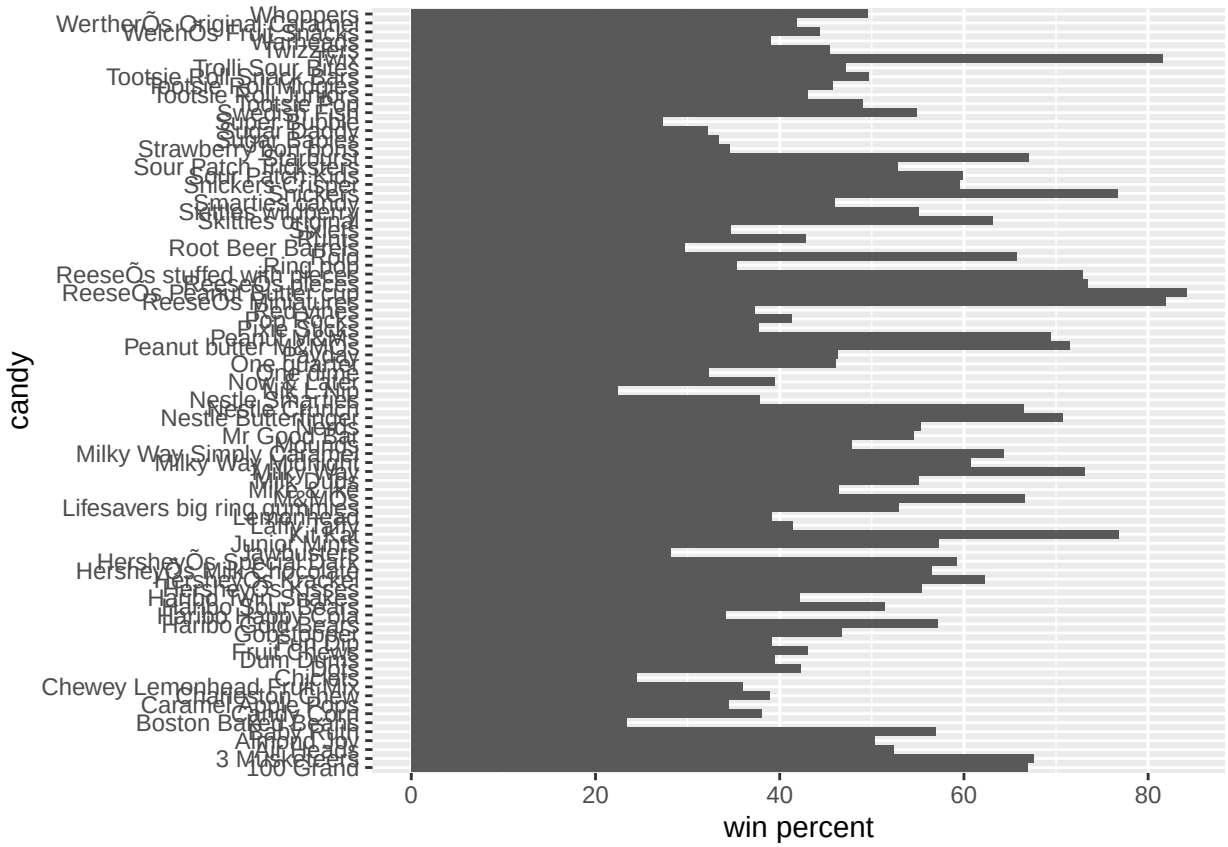
```
##                    chocolate fruity caramel peanutyalmondy nougat
## Nik L Nip                  0      1       0              0      0
## Boston Baked Beans         0      0       0              1      0
## Chiclets                   0      1       0              0      0
## Super Bubble               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
```

```
head(candy[order(candy$winpercent, decreasing = TRUE),], n=5) # 5 all time favorite candy types: Reese'
```

```
##                            chocolate fruity caramel peanutyalmondy nougat
## ReeseÕs Peanut Butter cup          1      0       0              1      0
## ReeseÕs 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
## ReeseÕs Peanut Butter cup                 0    0   0        0        0.720
## ReeseÕs 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
## ReeseÕs Peanut Butter cup         0.651   84.18029
## ReeseÕs Miniatures                0.279   81.86626
## Twix                              0.906   81.64291
## Kit Kat                           0.511   76.76860
## Snickers                          0.651   76.67378
```

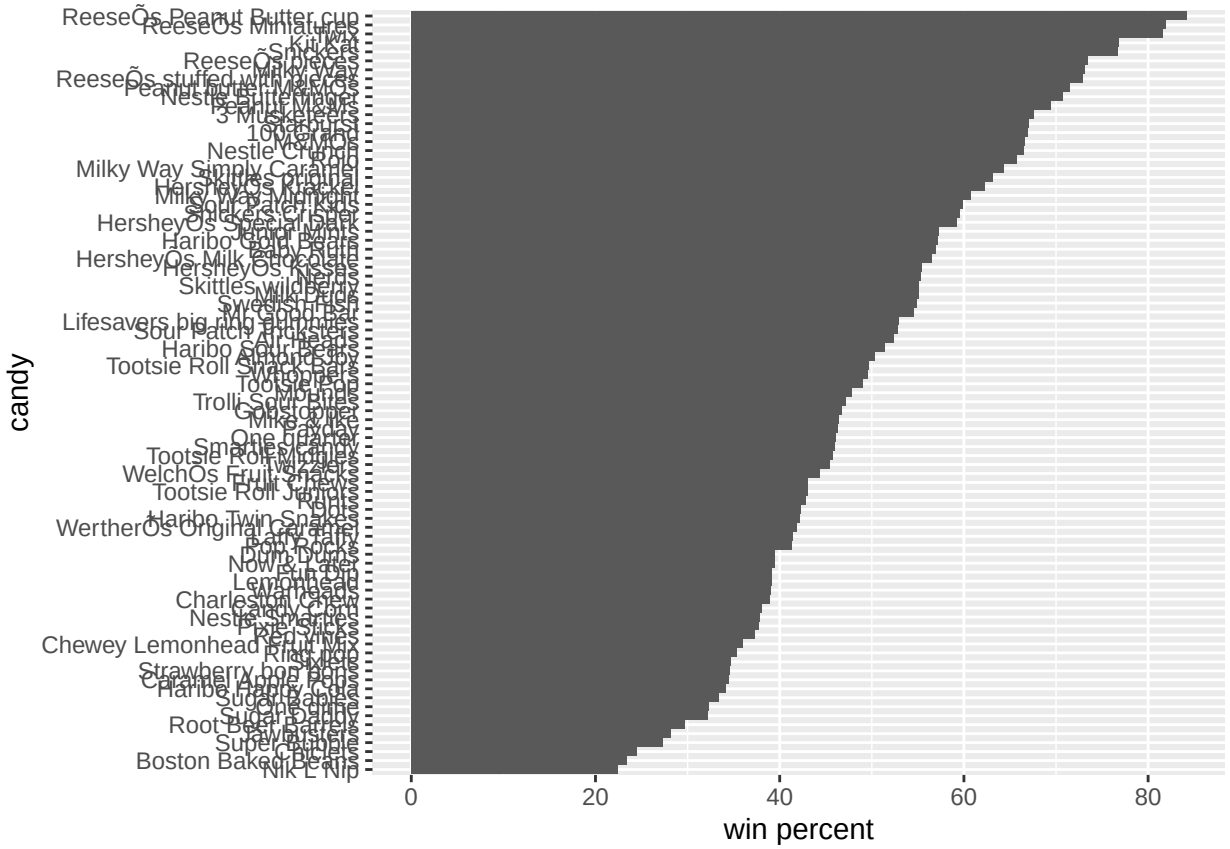
Q15. Make a first barplot of candy ranking based on winpercent values.

```
library(ggplot2)

ggplot(candy) +
  aes(winpercent, rownames(candy)) +
  geom_col() + labs(x = "win percent", y = "candy")
```



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() + labs(x = "win percent", y = "candy")
```

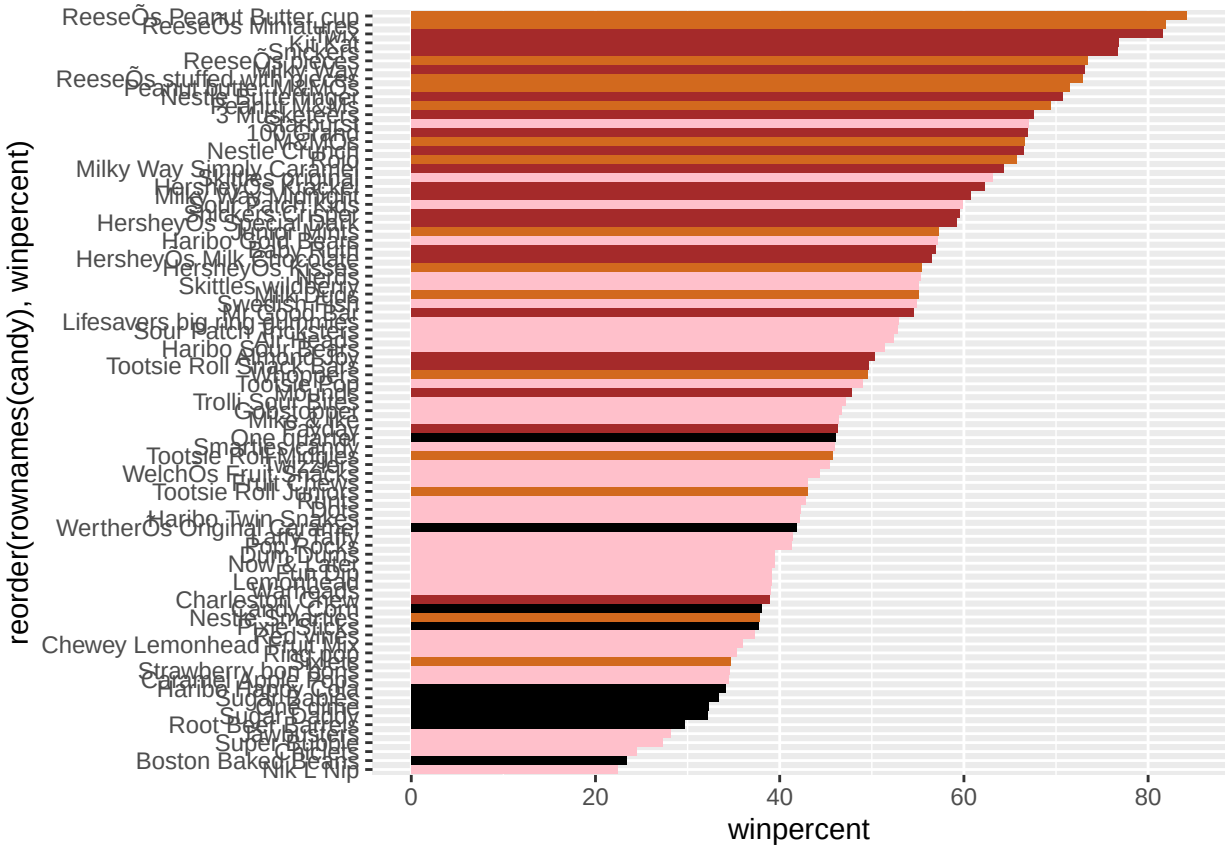


Setup a color vector to be used in future plots. Start by making a vector of all black values (one for each candy). Then overwrite chocolate (for chocolate candy), brown (for candy bars) and red (for fruity candy) values.

```
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"
```

Fill the previous bar plot with these colors

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



Q17. What is the worst ranked chocolate candy? Sixlets

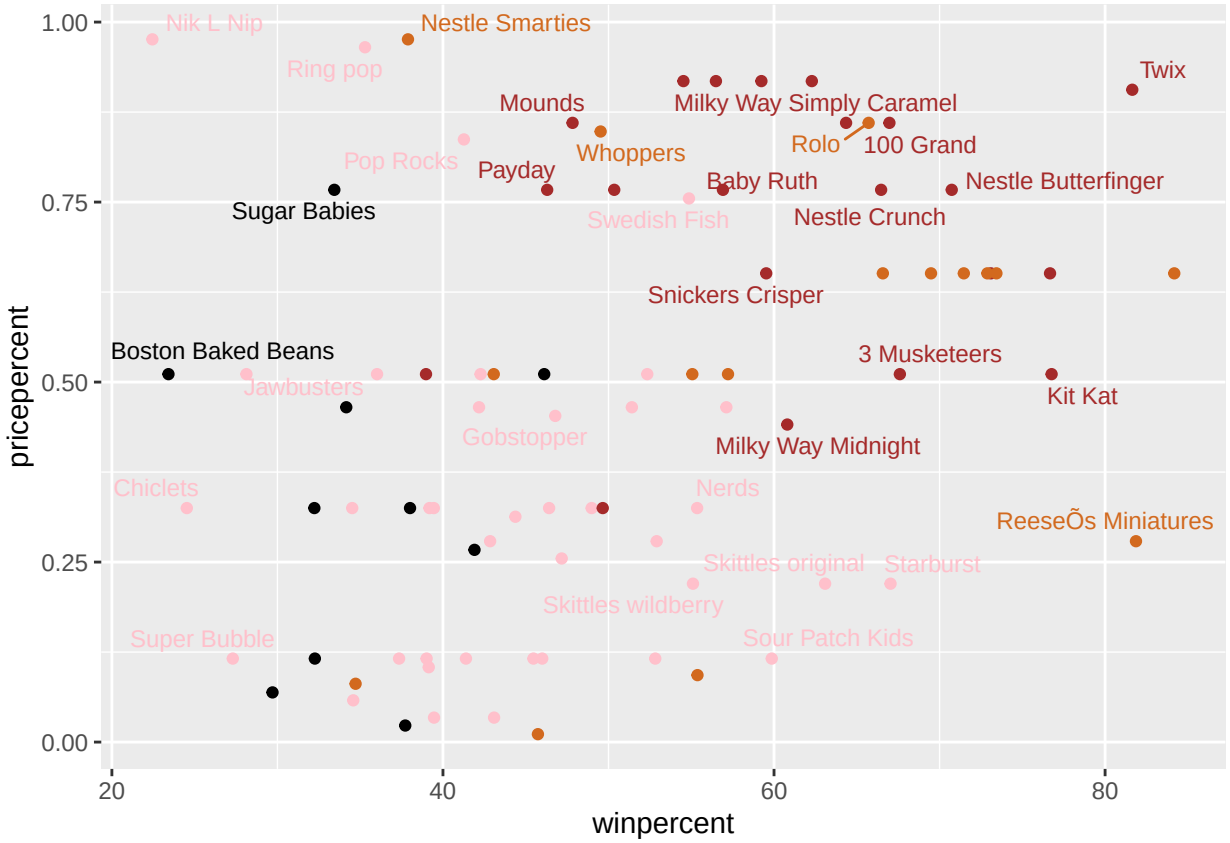
Q18. What is the best ranked fruity candy? Starbursts

4. Taking a look at pricepercent

Plot of winpercent vs the pricepercent

```
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: 54 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?

```
ord_winpercent <- order(candy$winpercent, decreasing = TRUE)
head(candy[ord_winpercent,c(11,12)], n=10)
```

```
##                               pricepercent winpercent
## ReeseÕs Peanut Butter cup            0.651   84.18029
## ReeseÕs Miniatures                   0.279   81.86626
## Twix                                 0.906   81.64291
## Kit Kat                              0.511   76.76860
## Snickers                             0.651   76.67378
## ReeseÕs pieces                       0.651   73.43499
## Milky Way                            0.651   73.09956
## ReeseÕs stuffed with pieces          0.651   72.88790
## Peanut butter M&MÕs                  0.651   71.46505
## Nestle Butterfinger                  0.767   70.73564
```

```
# Reese's Miniatures have the second highest winpercent (81.86626) and a pricepercent of 0.279.
```

Q20. What are the top 5 most expensive candy types in the dataset and of these which is the least popular?

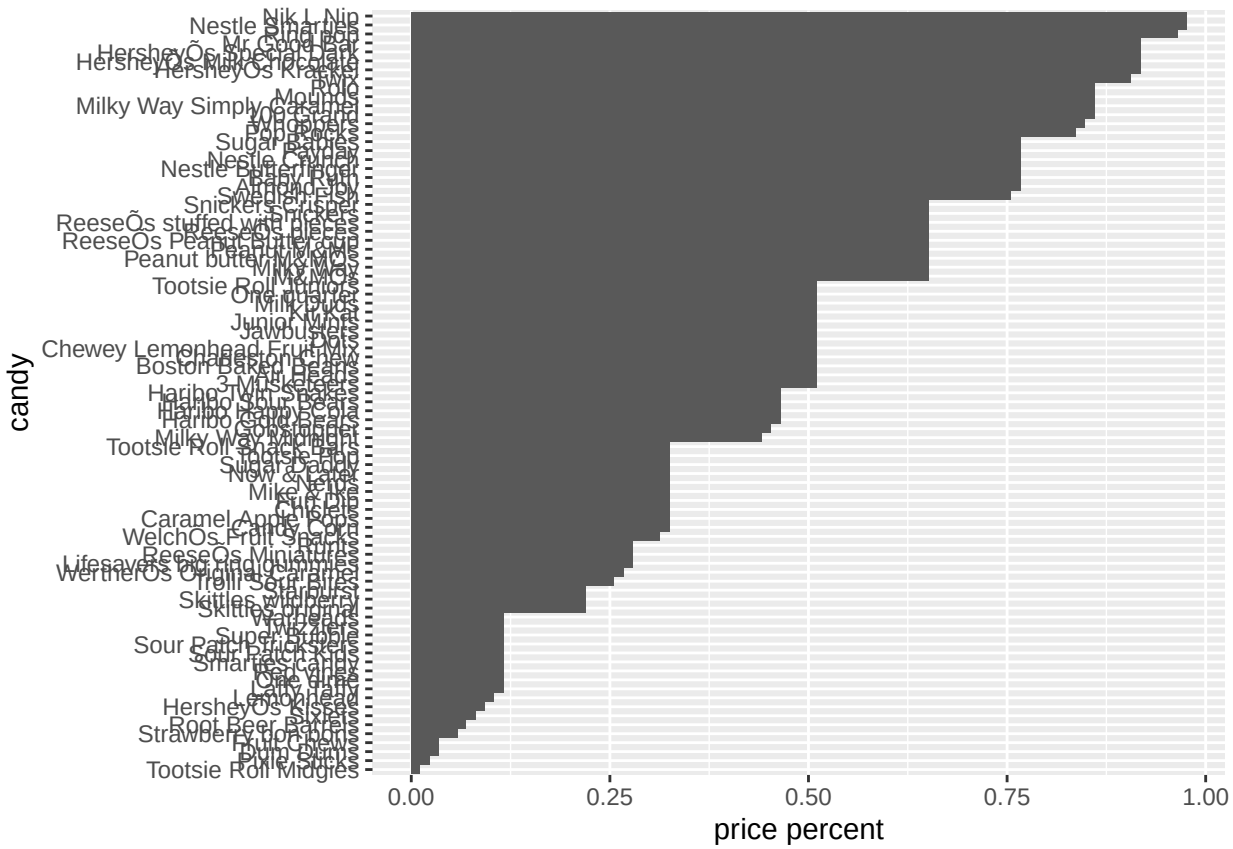
```
ord_pricepercent <- order(candy$pricepercent, decreasing = TRUE)
head(candy[ord_pricepercent,c(11,12)], n=5)
```

```
##                            pricepercent winpercent
## Nik L Nip                         0.976   22.44534
## Nestle Smarties                   0.976   37.88719
## Ring pop                          0.965   35.29076
## HersheyÕs Krackel                 0.918   62.28448
## HersheyÕs Milk Chocolate          0.918   56.49050
```

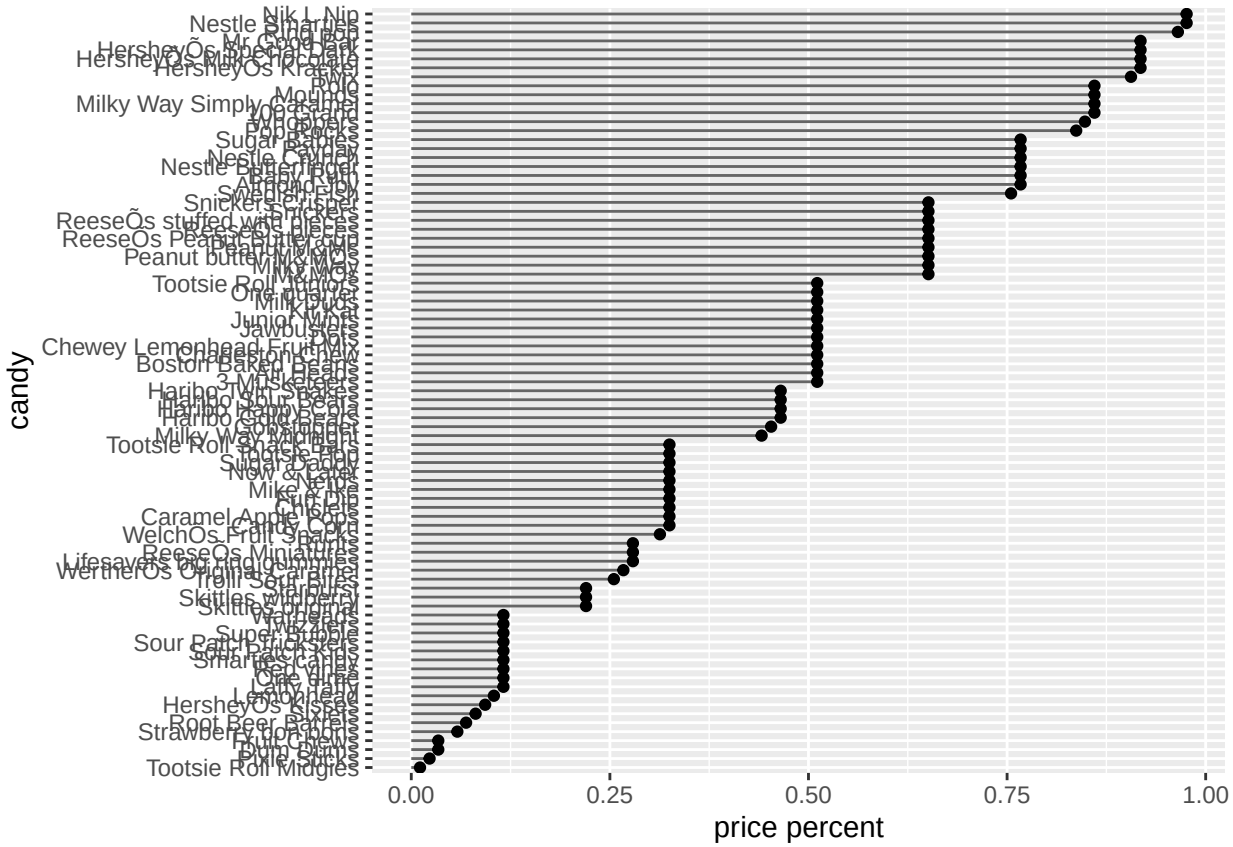
```
# The top 5 most expensive candies are Nik L Nip, Nestle Smarties, Ring pop, Hershey's Krackel, and Her
```

Q21. Make a barplot again with geom_col() this time using pricepercent and then improve this step by step, first ordering the x-axis by value and finally making a so called "dot chat" or "lollipop" chart by swapping geom_col() for geom_point() + geom_segment().

```
ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy),pricepercent)) +
  geom_col() + labs(x = "price percent", y = "candy")
```




```
ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy), pricepercent)) +
  geom_segment(aes(yend = reorder(rownames(candy), pricepercent),
                   xend = 0), col="gray40") +
    geom_point() + labs(x="price percent", y="candy")
```



5 Exploring the correlation structure

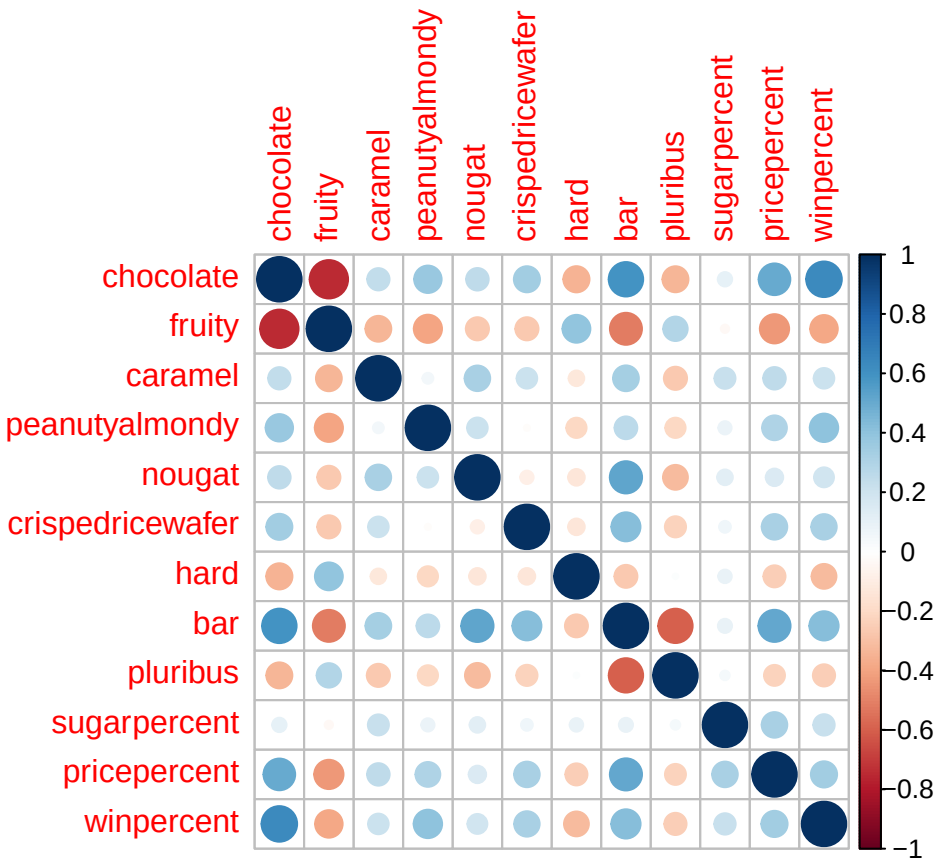
Load corrplot package

```
library(corrplot)
```

```
## corrplot 0.90 loaded
```

Plot a correlation matrix using the candy dataset

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



Q22. Examining this plot what two variables are anti-correlated (i.e. have minus values)? Two variable that are antivorrelated are chocolate and fruity

Q23. Similarly, what two variables are most positively correlated? Two variables that are most positively correlated are chocolate and bar.

6. Principal Component Analysis

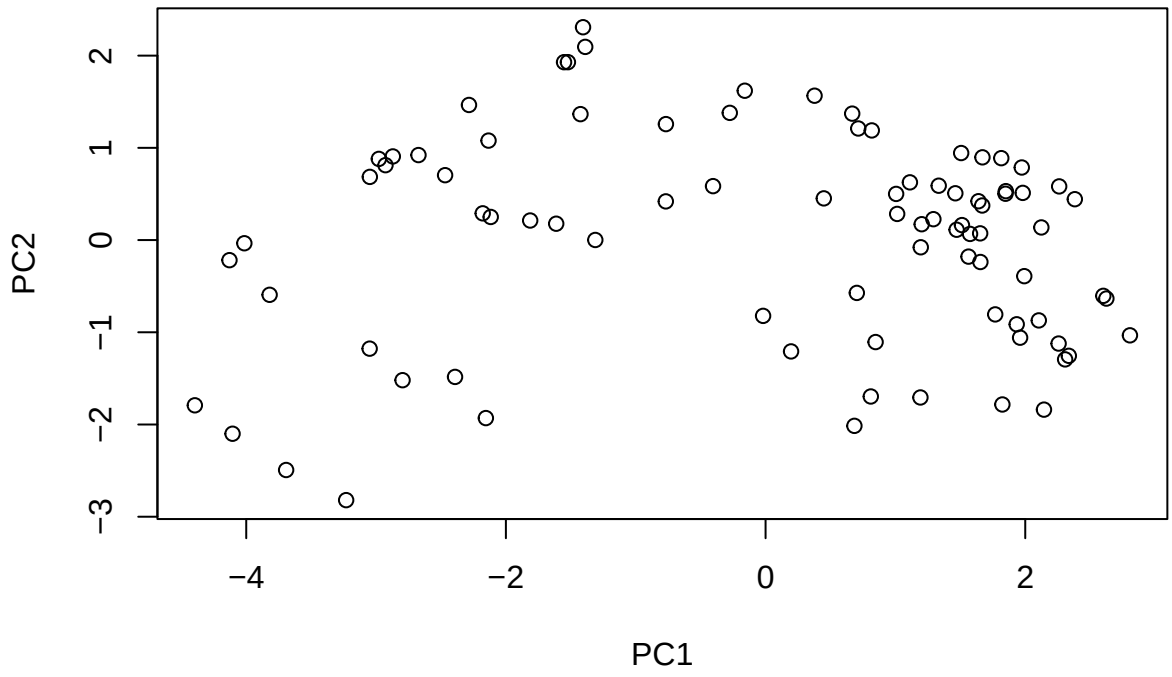
Apply PCA using the prcomp() function to the candy dataset

```
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
```

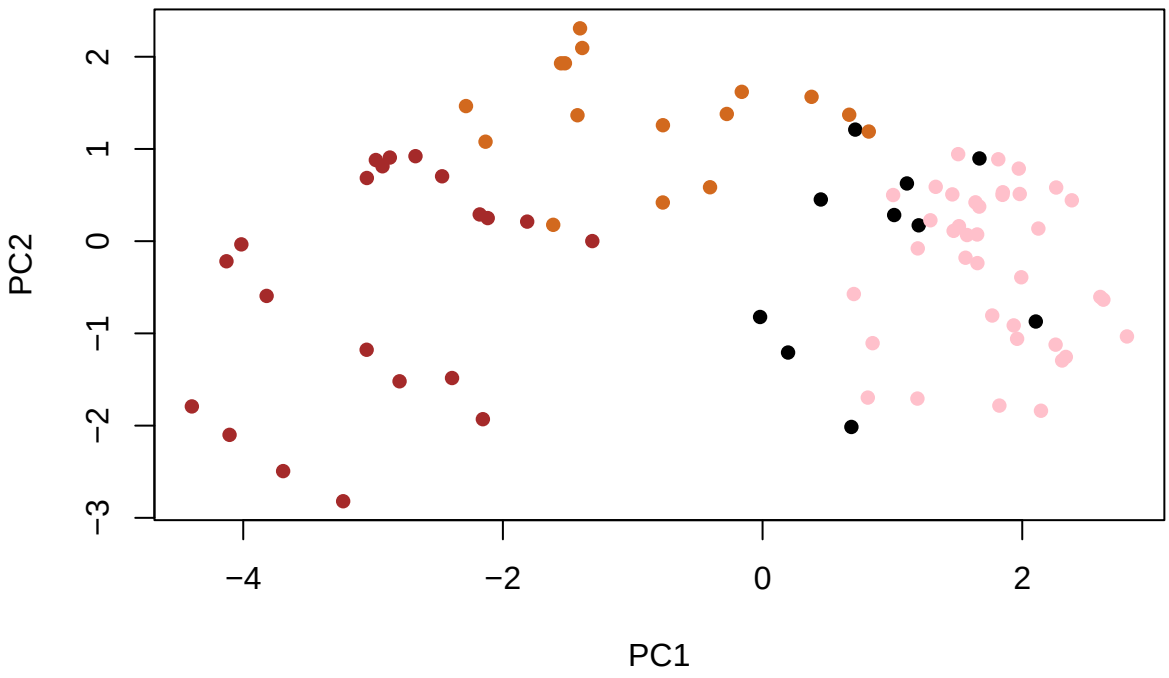
Plot PC1 vs PC2

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



Change the plotting character and add some color using the color vector define previously.

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



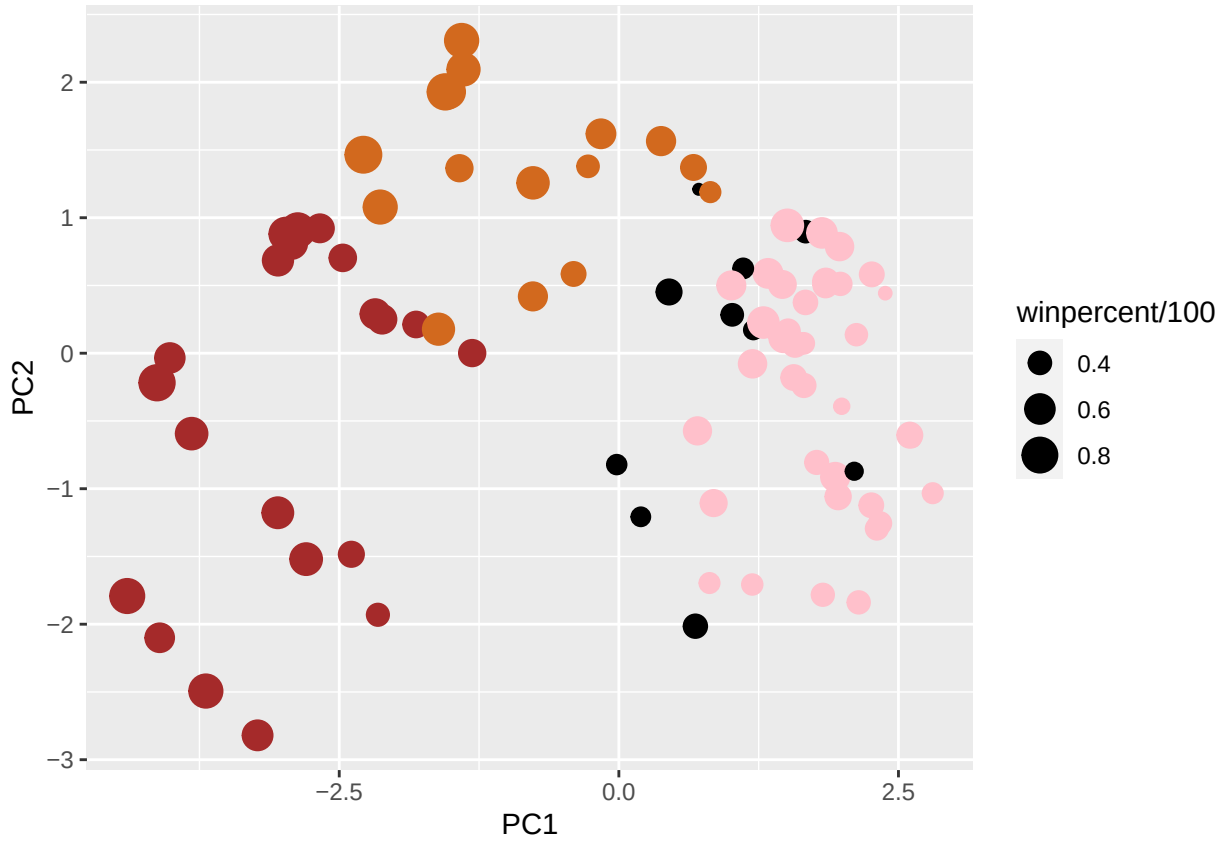
Make a new data-frame with the PCA results and candy data to be used with the ggplot() function

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

```
##              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        PC1
## 100 Grand       0   1        0        0.732        0.860   66.97173 -3.8198617
## 3 Musketeers    0   1        0        0.604        0.511   67.60294 -2.7960236
## One dime        0   0        0        0.011        0.116   32.26109  1.2025836
## One quarter     0   0        0        0.011        0.511   46.11650  0.4486538
## Air Heads       0   0        0        0.906        0.511   52.34146  0.7028992
## Almond Joy      0   1        0        0.465        0.767   50.34755 -2.4683383
##                     PC2        PC3
## 100 Grand    -0.5935788  2.1863087
## 3 Musketeers -1.5196062 -1.4121986
## One dime      0.1718121 -2.0607712
## One quarter   0.4519736 -1.4764928
## Air Heads    -0.5731343  0.9293893
## Almond Joy    0.7035501 -0.8581089
```

Use ggplot() to make a plot with our PCA results

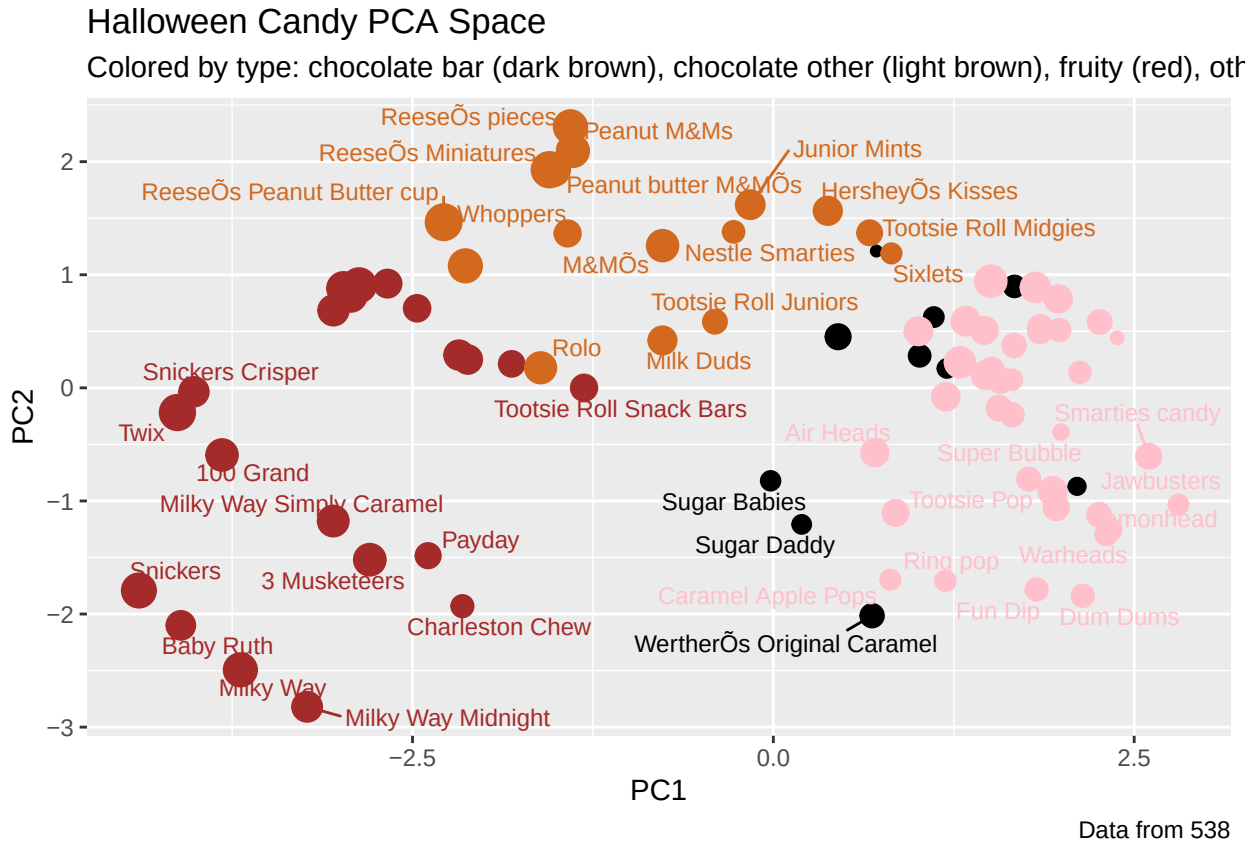
```
p <- ggplot(my_data) +
        aes(x=PC1, y=PC2,
            size=winpercent/100,
            text=rownames(my_data),
            label=rownames(my_data)) +
        geom_point(col=my_cols)
p
```



Use the ggrepel package and the function ggrepel::geom_text_repel() to label the plot with non overlapping candy names, a title, and subtitle.

```
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), fruity (re
       caption="Data from 538")
```

```
## Warning: ggrepel: 44 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```



Pass the ggplot object p to plotly to generate an interactive plot that you can mouse over to see labels

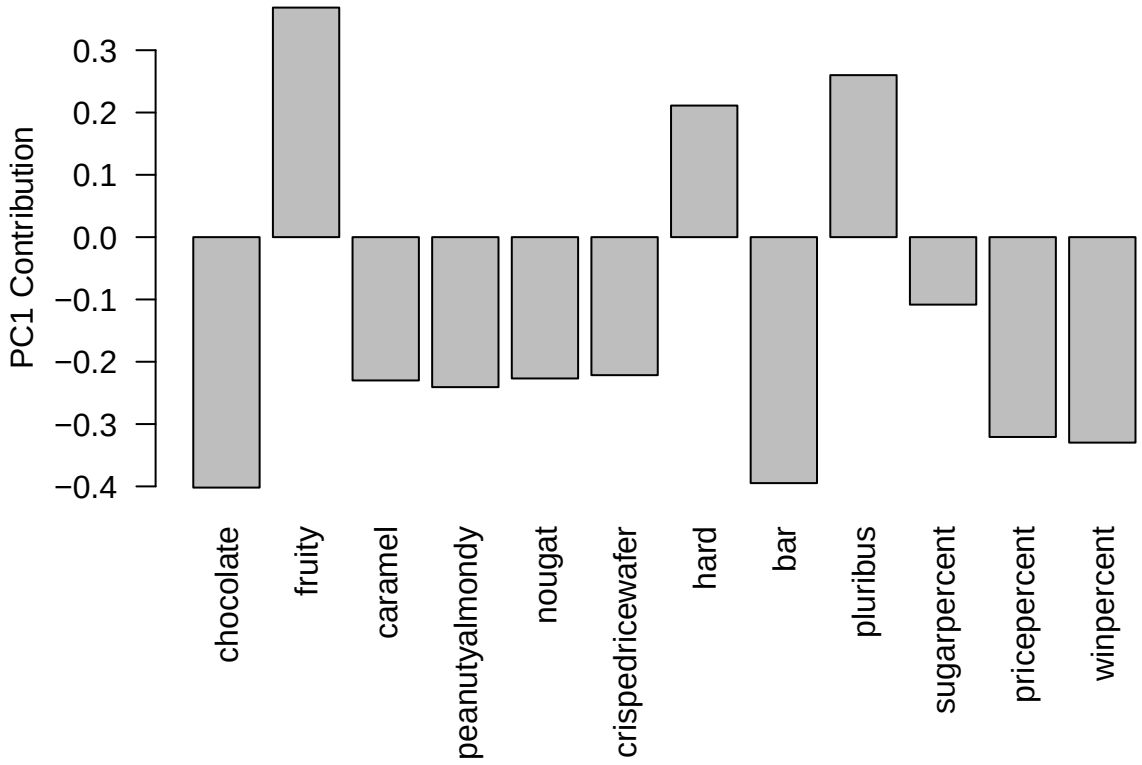
```
library(plotly)
```

```
##
## 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)
```

Look at PCA our loadings.

```
par(mar=c(8,4,2,2)) # Set the margins of the graph by calling the par() function with the mar argument.
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? The variables that are picked up stringly by PC1 in the positive direction are fruity, hard, and pluribus. These make sense since many hard candies are fruit flavored and are sold as multiples in one package (e.g. jolly ranchers).