Class10

AUTHOR
Darby Patterson

Importing Candy Data

We first need to import the csv file holding our data

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

```
[1] "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
```

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

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

```
[1] 38
```

Favorite Candy?

```
#example code
candy["Twix", ]$winpercent
```

```
[1] 81.64291
```

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

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

```
[1] 39.0119
```

Q4: What is the `winpercent` value for "Kit Kat"?

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

```
[1] 76.7686
```

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

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

```
[1] 49.6535
```

```
#install.packages("skimr")
library("skimr")
skim(candy)
```

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	▇▁▁▁▇

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

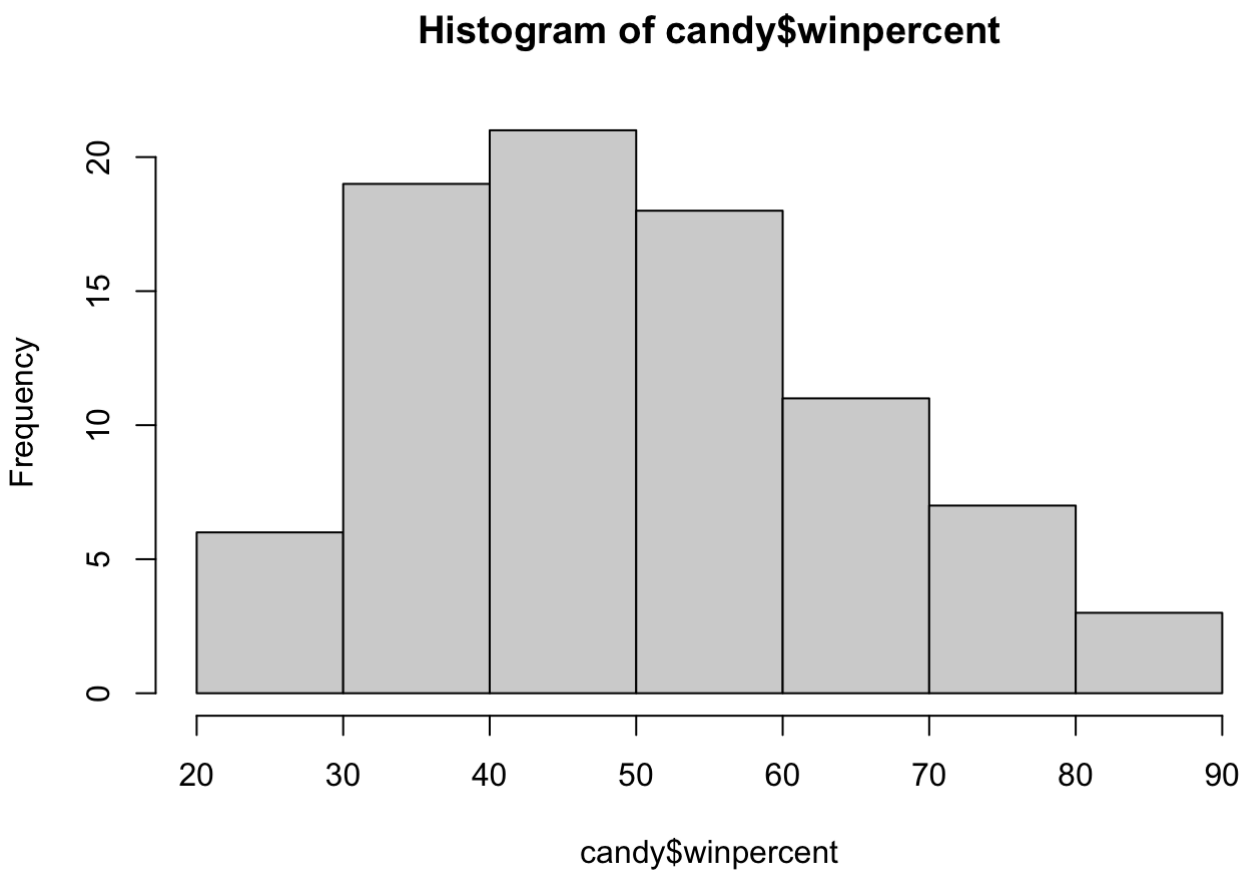
The win percentage seems to be on a larger scale than the rest of the dataset.

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

It represents a yes or no, as the question posed was if it was chocolate or not. There is variability in the other category because it's based on preference and isn't as binary.

Q8: Plot a histogram of `winpercent` values.

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



Q9: Is the distribution of `winpercent` values symmetrical?

It looks to be slightly skewed but mostly symmetrical.

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

The center of distribution is just below 50%

```
# get average win percent for chocolates and fruits
chocolate_mean <- mean(candy$winpercent[as.logical(candy$chocolate)])

fruit_mean <- mean(candy$winpercent[as.logical(candy$fruity)])

# print the results
cat("chocolates:", chocolate_mean, "\n")
```

```
chocolates: 60.92153
```

```
cat("fruits:", fruit_mean, "\n")
```

```
fruits: 44.11974
```

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

Chocolate has a higher ranking than fruit by about 20%.

Q12. Is this difference statistically significant?

```
# perform t-test
t_test <- t.test(candy$winpercent[as.logical(candy$chocolate)],
                 candy$winpercent[as.logical(candy$fruity)],
                 var.equal = FALSE)

# print the results
cat("Welch Two Sample t-test \n")
```

```
Welch Two Sample t-test
```

```
cat("p-value:", t_test$p.value, "\n")
```

```
p-value: 2.871378e-08
```

From the p-value, we can determine that this is statically significant.

Overall Candy Rankings

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

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

head(sorted_candy, 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
```

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

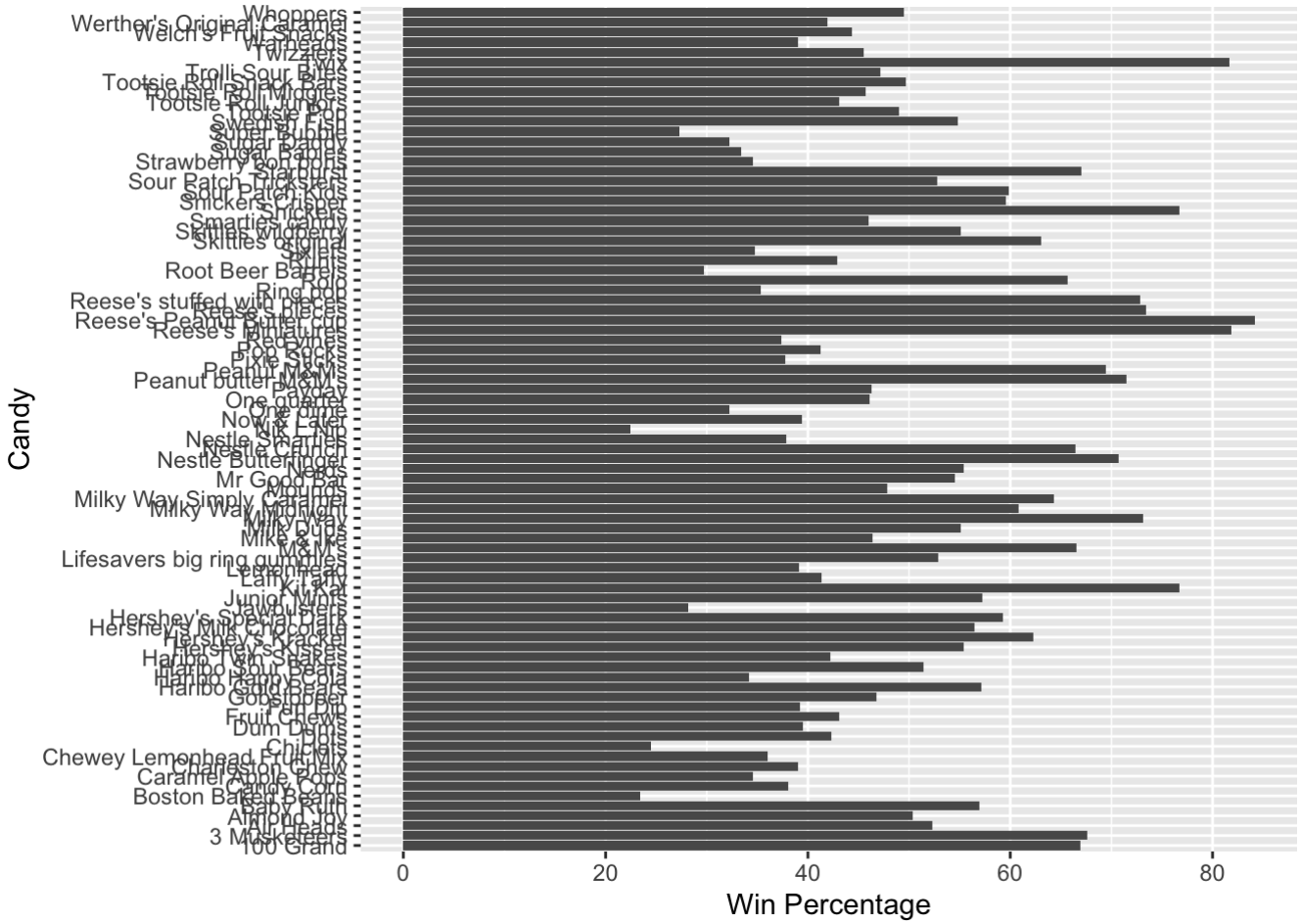
```r
sorted_candy <- candy[order(-candy$winpercent),]

head(sorted_candy, 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
```

Barplot Visualization

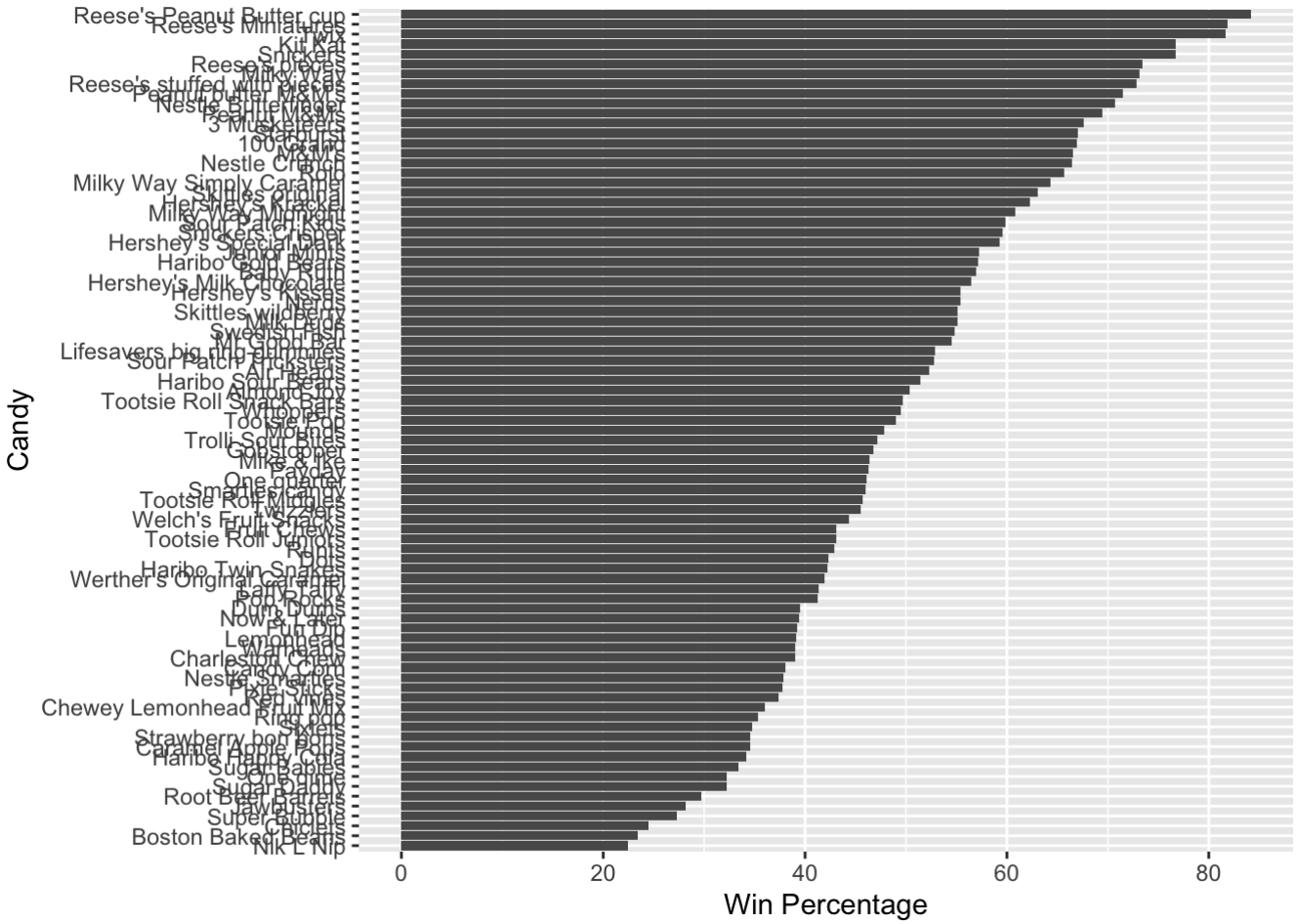
Q15: Make a first barplot of candy ranking based on `winpercent` values.

```r
#install.packages("ggplot")
library(ggplot2)
candy$names <- rownames(candy)

# create a bar plot of win percent by candy name
ggplot(candy, aes(x = names, y = winpercent)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  xlab("Candy") +
  ylab("Win Percentage")
```



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

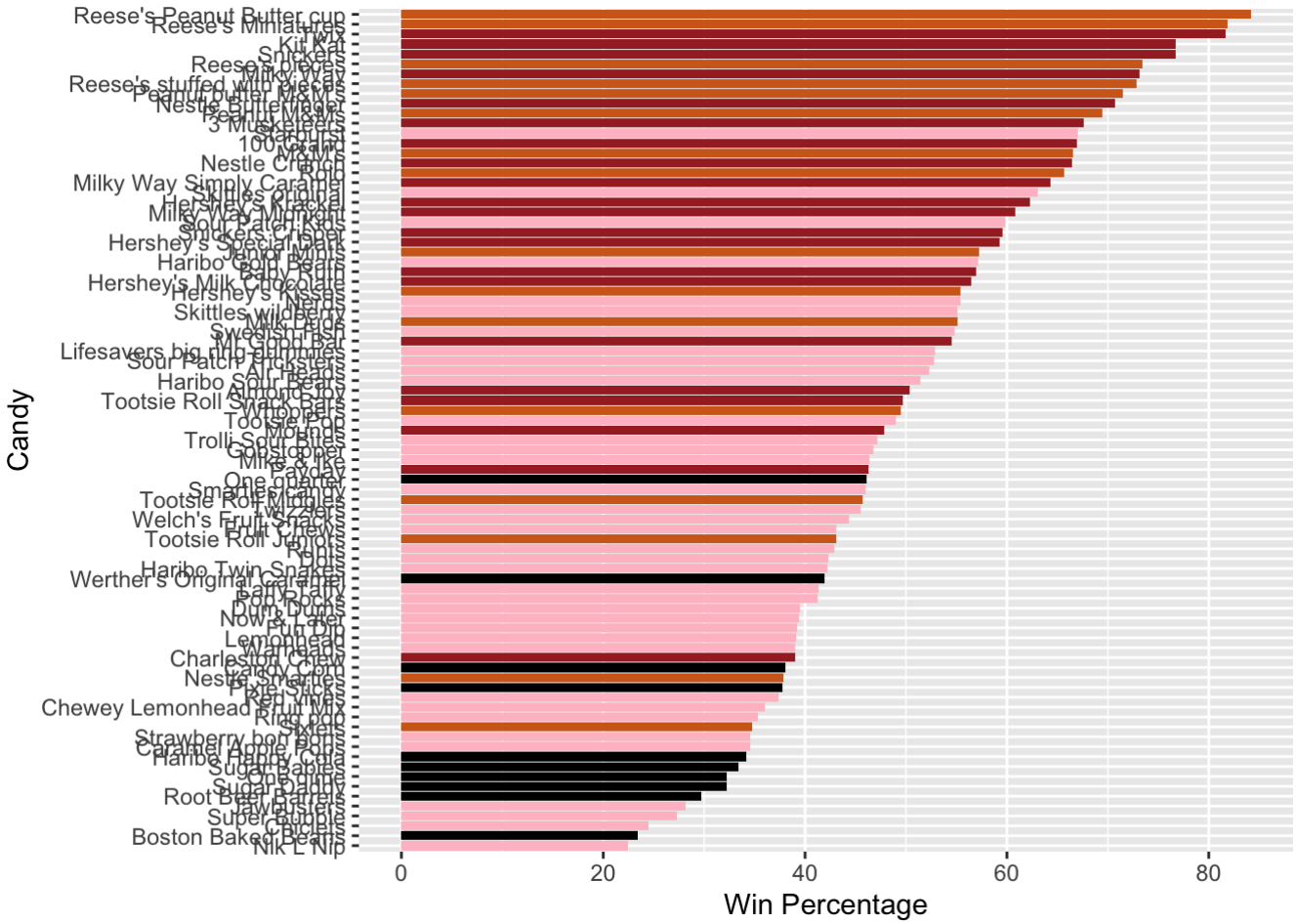
```
ggplot(candy, aes(x = reorder(names, winpercent), y = winpercent)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  xlab("Candy") +
  ylab("Win Percentage")
```



Now we can add color:

```
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(x = reorder(names, winpercent), y = winpercent)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  xlab("Candy") +
  ylab("Win Percentage") +
   geom_col(fill=my_cols)
```



Q**17**. What is the worst ranked chocolate candy?

According to the plot, the worst ranked chocolate candy is Sixlets.

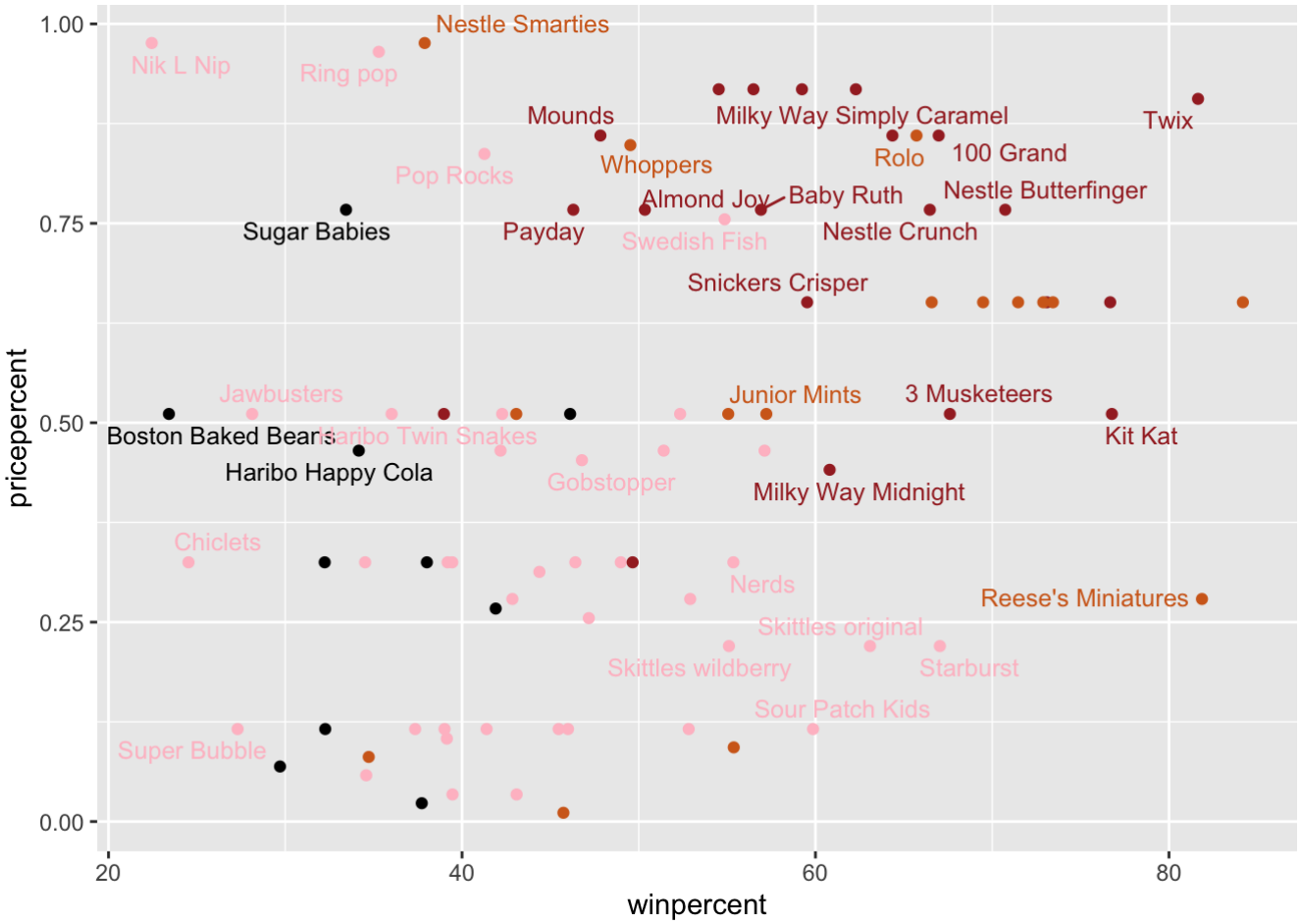
Q**18**. What is the best ranked fruity candy?

According to the plot, the best ranked fruit candy is Starbursts.

Taking a look at price percent

```
#install.packages("ggrepel")
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: 50 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Q**19**. Which candy type is the highest ranked in terms of `winpercent` for the least money - i.e. offers the most bang for your buck?

Tootsie Roll Midgies seem to be the cheapest high-ranked candy.

```
# order the dataset by price percent
candy_ordered <- candy[order(candy$pricepercent),]

# select the candy with the highest win percent and the lowest price percent
top_candy <- candy_ordered[which.min(candy_ordered$pricepercent[candy_ordered$winpercent
top_candy
```

```
                     chocolate fruity caramel peanutyalmondy nougat
Tootsie Roll Midgies         1      0       0              0      0
                     crispedricewafer hard bar pluribus sugarpercent
Tootsie Roll Midgies                0    0   0        1        0.174
                     pricepercent winpercent                names
Tootsie Roll Midgies        0.011   45.73675 Tootsie Roll Midgies
```

Q**20**. 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
```

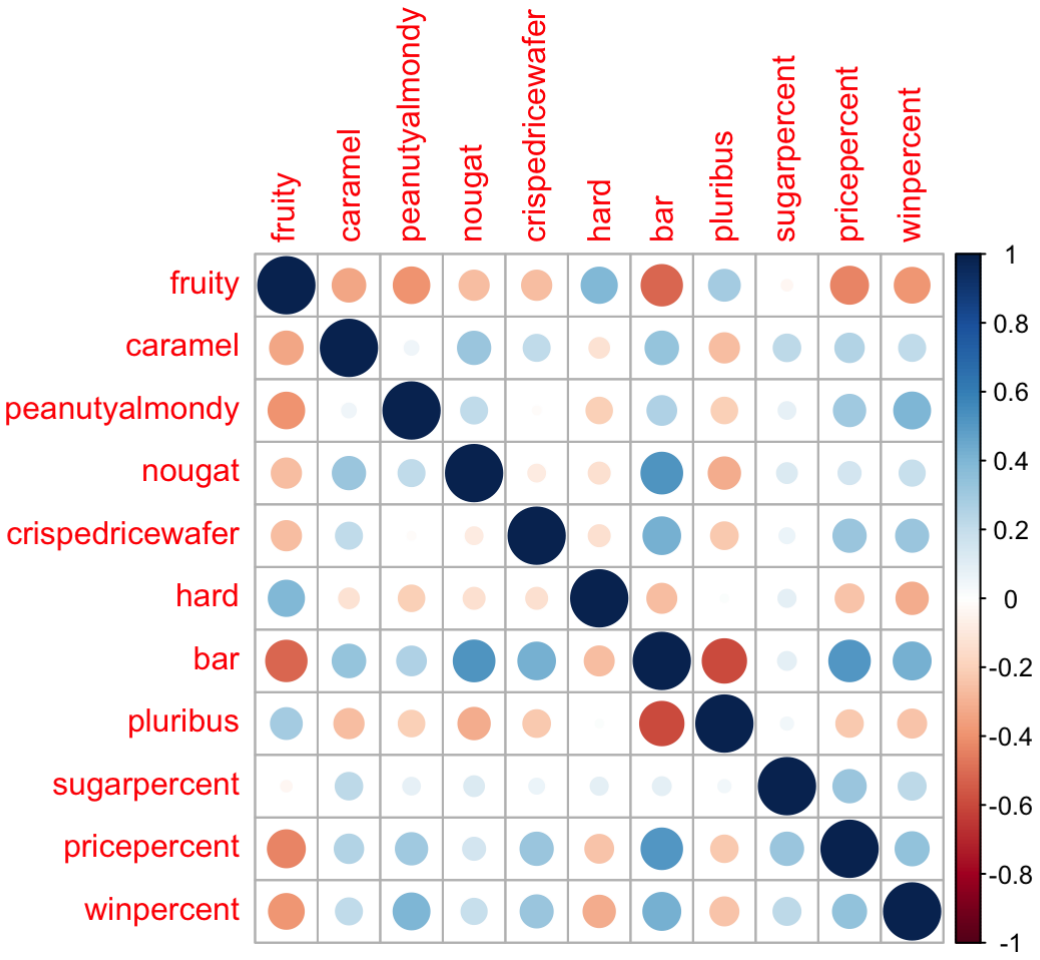
Exploring the correlation structure

Now we want to plot a correlation matrix.

```
#install.packages("corrplot")
library(corrplot)
```

```
corrplot 0.92 loaded
```

```
n_candy <- candy[, c(2:12)]
cij <- cor(n_candy)
corrplot(cij)
```



Q**22**. Examining this plot what two variables are anti-correlated (i.e. have minus values)?

caramel and crisped rice wafer

Q**23**. Similarly, what two variables are most positively correlated?

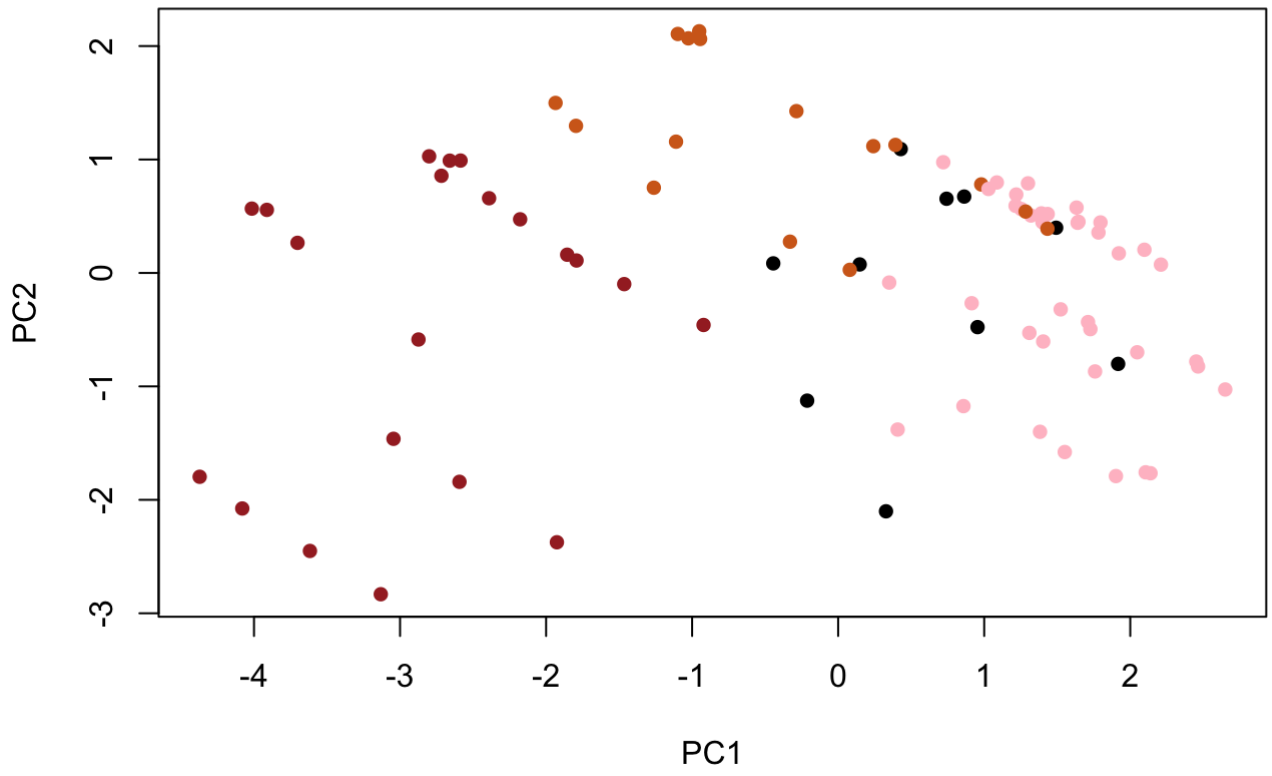
pluribus and peanuty almondy

Principal Component Analysis

```
# create a new data frame with only numeric columns
n_candy <- candy[, c(2:12)]

# apply PCA to the numeric data
pca <- prcomp(n_candy, scale = TRUE)

# print PCA results
summary(pca)
```

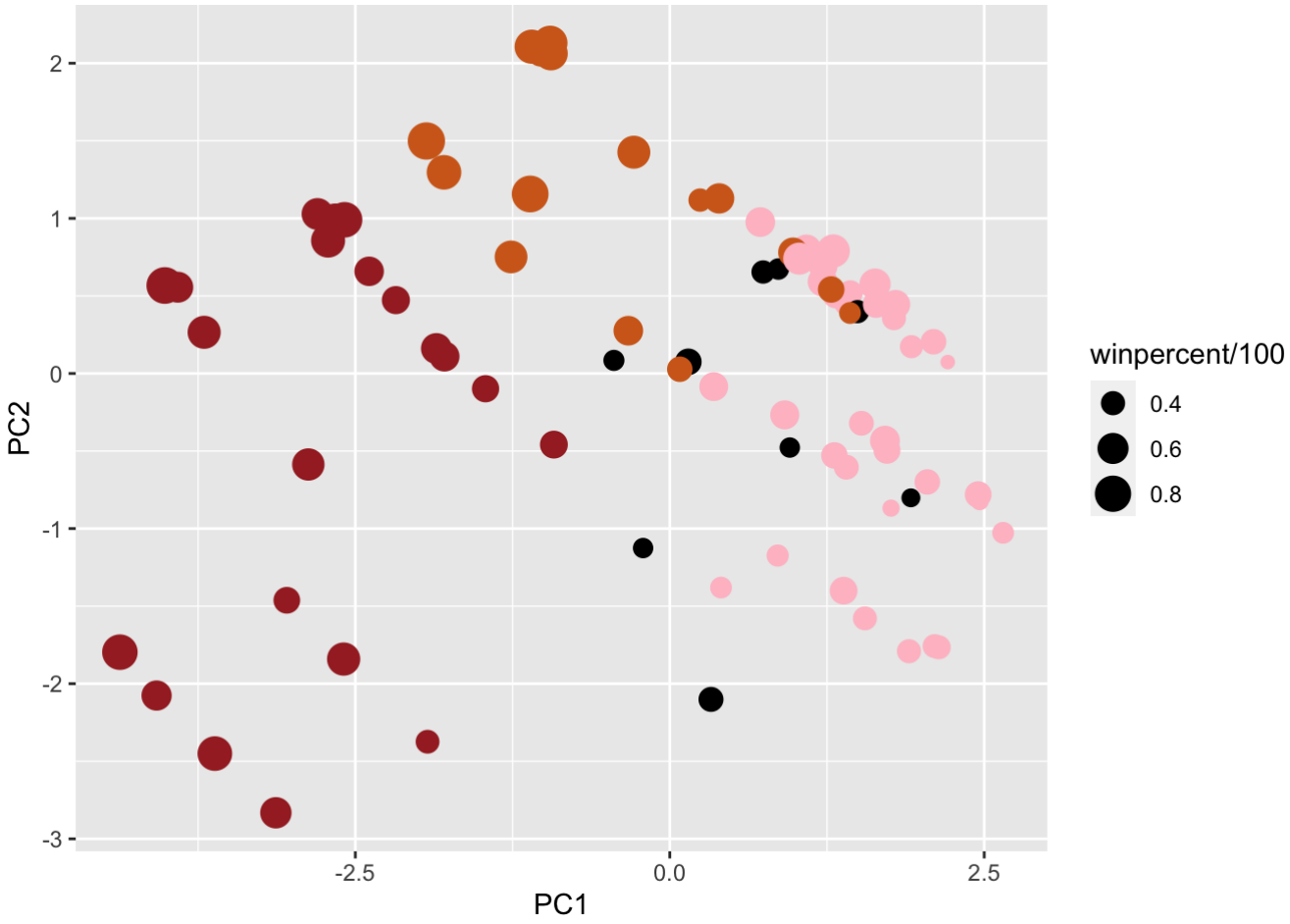
```
Importance of components:
                          PC1    PC2    PC3    PC4     PC5     PC6     PC7
Standard deviation     1.9200 1.1143 1.1085 1.0751 0.95010 0.81815 0.81352
Proportion of Variance 0.3351 0.1129 0.1117 0.1051 0.08206 0.06085 0.06016
Cumulative Proportion  0.3351 0.4480 0.5597 0.6648 0.74685 0.80770 0.86787
                           PC8     PC9    PC10    PC11
Standard deviation     0.68950 0.64410 0.60875 0.43887
Proportion of Variance 0.04322 0.03772 0.03369 0.01751
Cumulative Proportion  0.91109 0.94880 0.98249 1.00000
```

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



Now with a ggplot2,

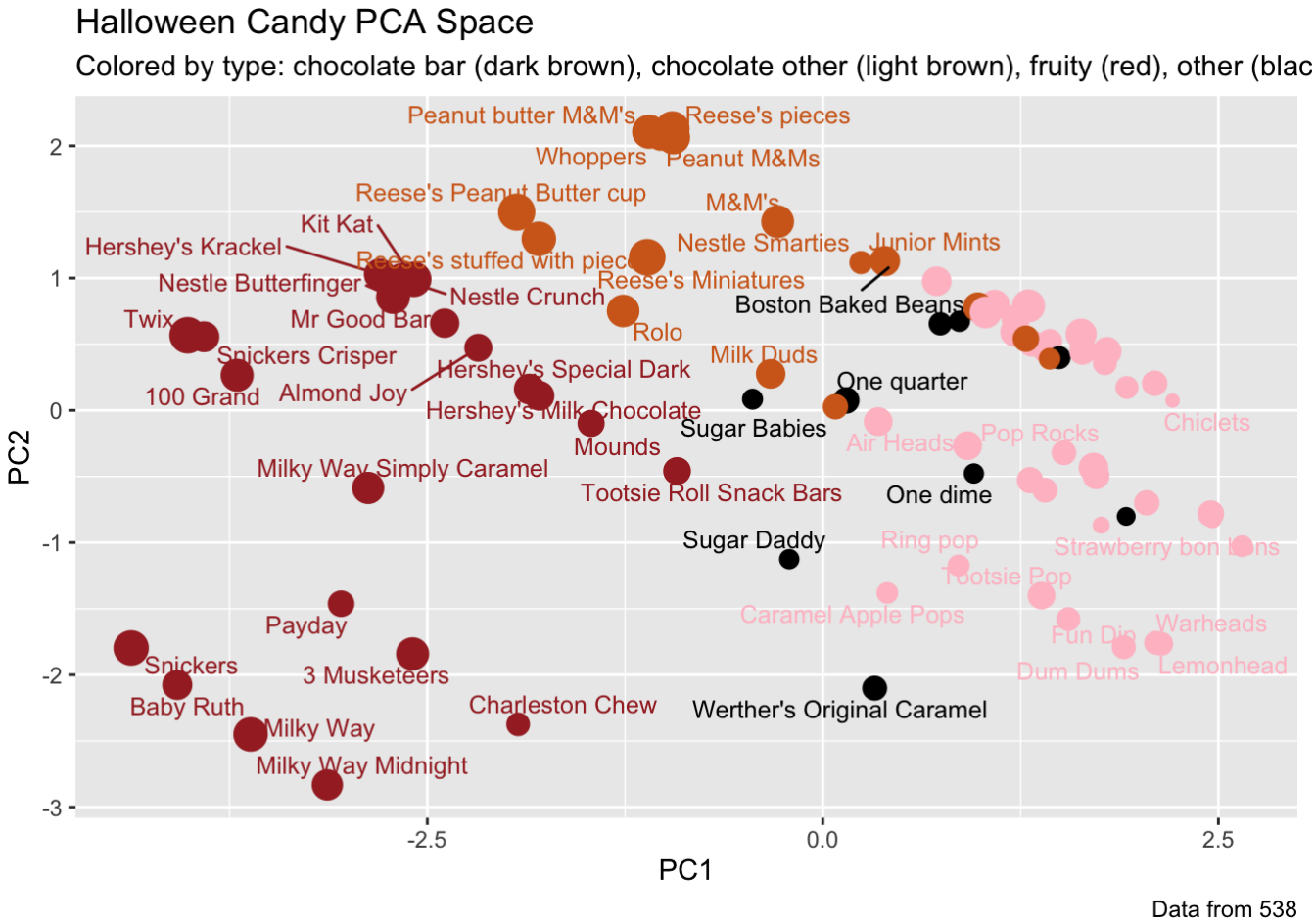
```
#install.packages("ggplot2")
library(ggplot2)
my_data <- cbind(candy, pca$x[,1:3])

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
```



We can use ggrepel to add names to each point.

```
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 brow
       caption="Data from 538")
```

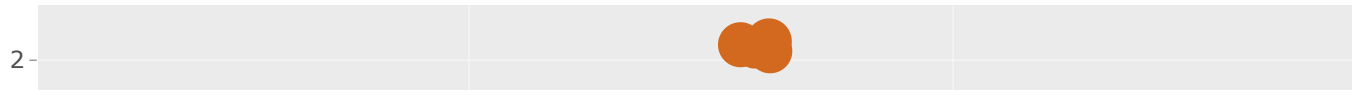
```
Warning: ggrepel: 35 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

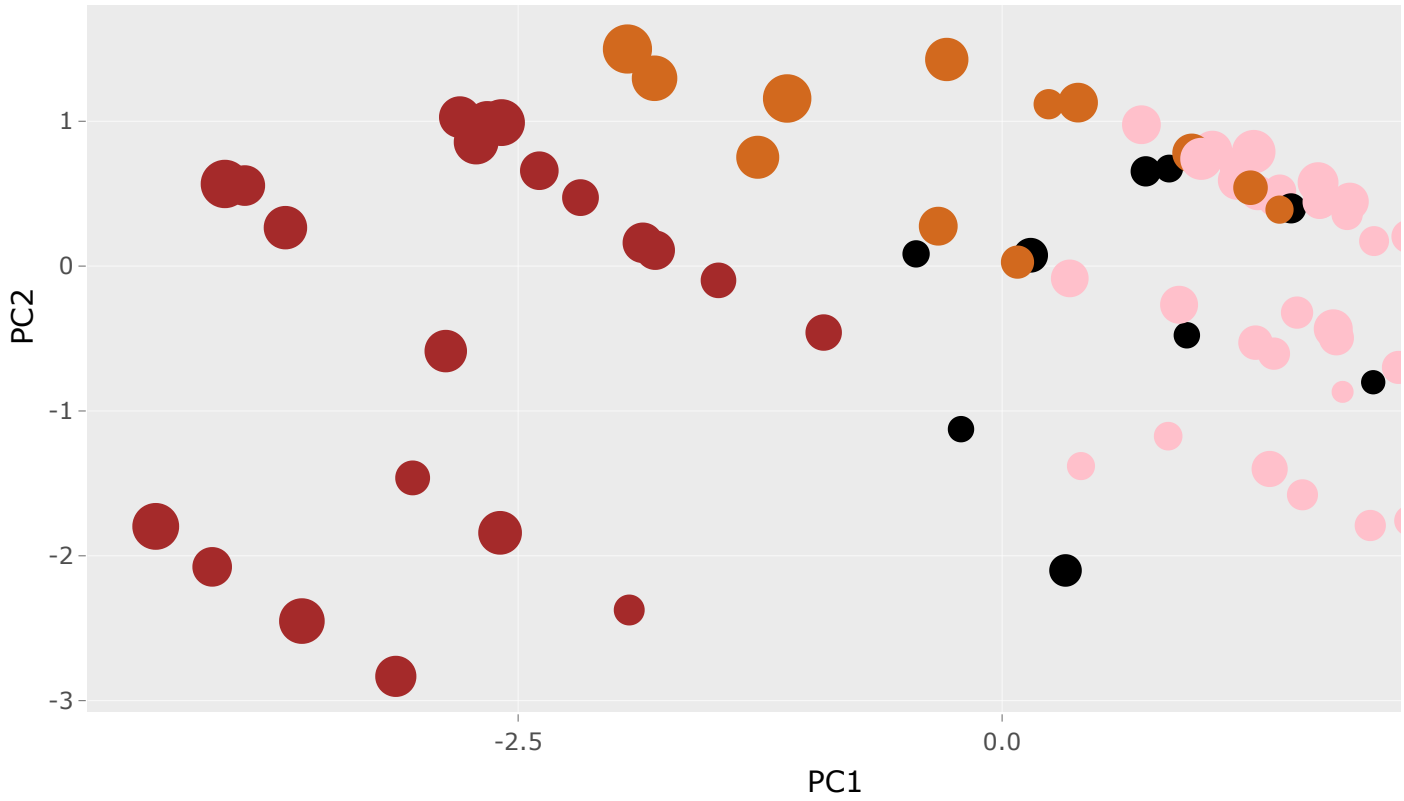



```
#install.packages("plotly")
library(plotly)
```

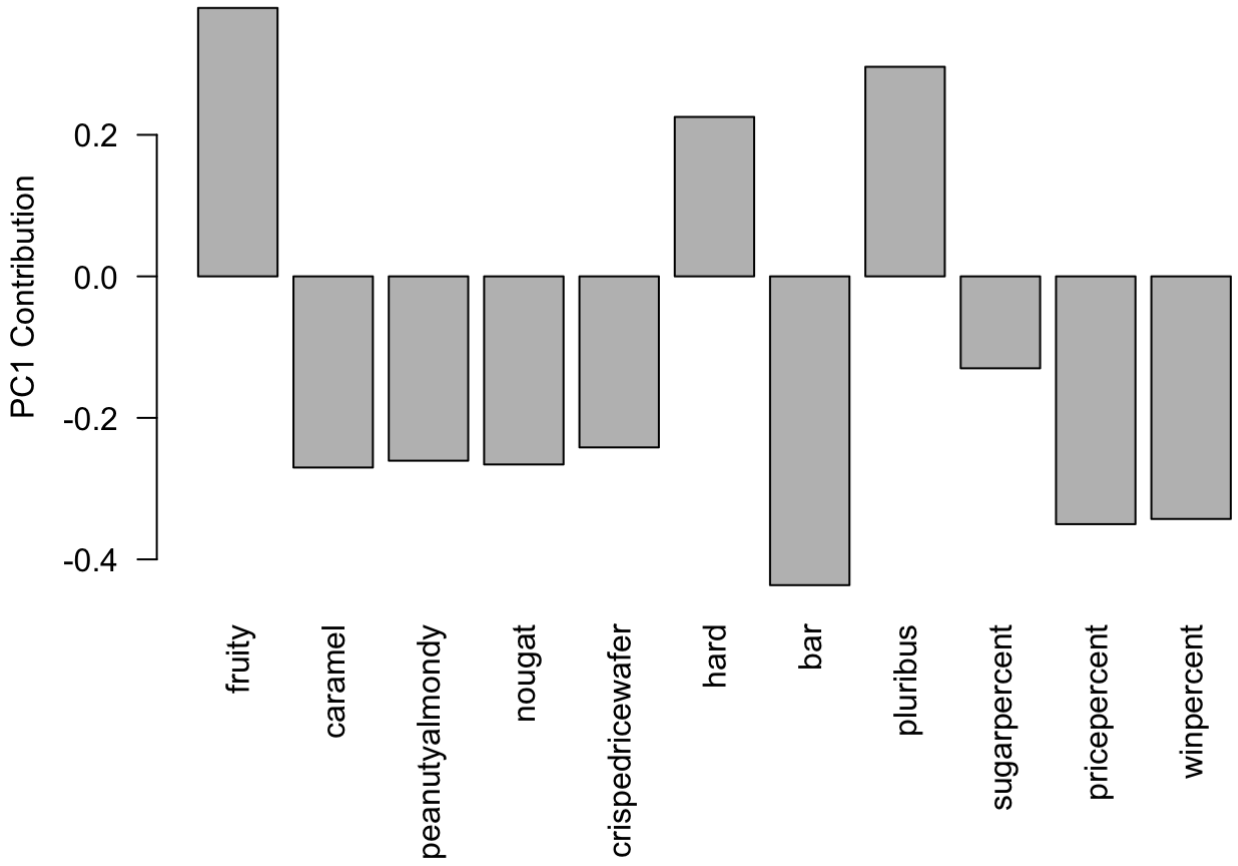
```
Attaching package: 'plotly'

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)
```






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



```r
loadings <- pca$rotation[,1]
loadings_sorted <- sort(abs(loadings), decreasing = TRUE)
names(loadings_sorted)[1:5]
```

```
[1] "bar"          "fruity"       "pricepercent" "winpercent"   "pluribus"
```

Q**24**. What original variables are picked up strongly by PC1 in the positive direction? Do these make sense to you?

Fruity and pluribus both were strongly picked up by PC1 which makes sense as they were both ranked highly.