BGGN213-Class9

AUTHOR
Lisanne Stouthart (PID A69036187)

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

Halloween Mini-Project

Importing Candy Data

```
# Save your input data file into your Project directory
candy_file <- "~/Downloads/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?

```
# Number of different candy types
num_candy_types <- nrow(candy)
num_candy_types #answer = 85
```

```
[1] 85
```

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

```
colnames(candy)
```

```
 [1] "chocolate"        "fruity"           "caramel"          "peanutyalmondy"
 [5] "nougat"           "crispedricewafer" "hard"             "bar"
 [9] "pluribus"         "sugarpercent"     "pricepercent"     "winpercent"
```

```
# Number of fruity candy types
num_fruity_types <- sum(candy$fruity == 1)
num_fruity_types #answer = 38
```

```
[1] 38
```

Interim

```
candy["Twix", ]$winpercent #81.6%
```

```
[1] 81.64291
```

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

```
candy["Rolo", ]$winpercent #65.7%
```

```
[1] 65.71629
```

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

```
candy["Kit Kat", ]$winpercent #76.8%
```

```
[1] 76.7686
```

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

```
candy["Tootsie Roll Snack Bars", ]$winpercent #49.7%
```

```
[1] 49.6535
```

```
#candy |>
  #filter(rownames(candy) %in% c("Kit Kat",
```

```
                                  #"Tootsie Roll Snack Bars") ) |>
  #select(winpercent)
```

Interim

```
#candy |>
  #filter(winpercent > 75) |>
  #filter(pricepercent < 0.5)

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	▆▁▁▁▁▆
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?

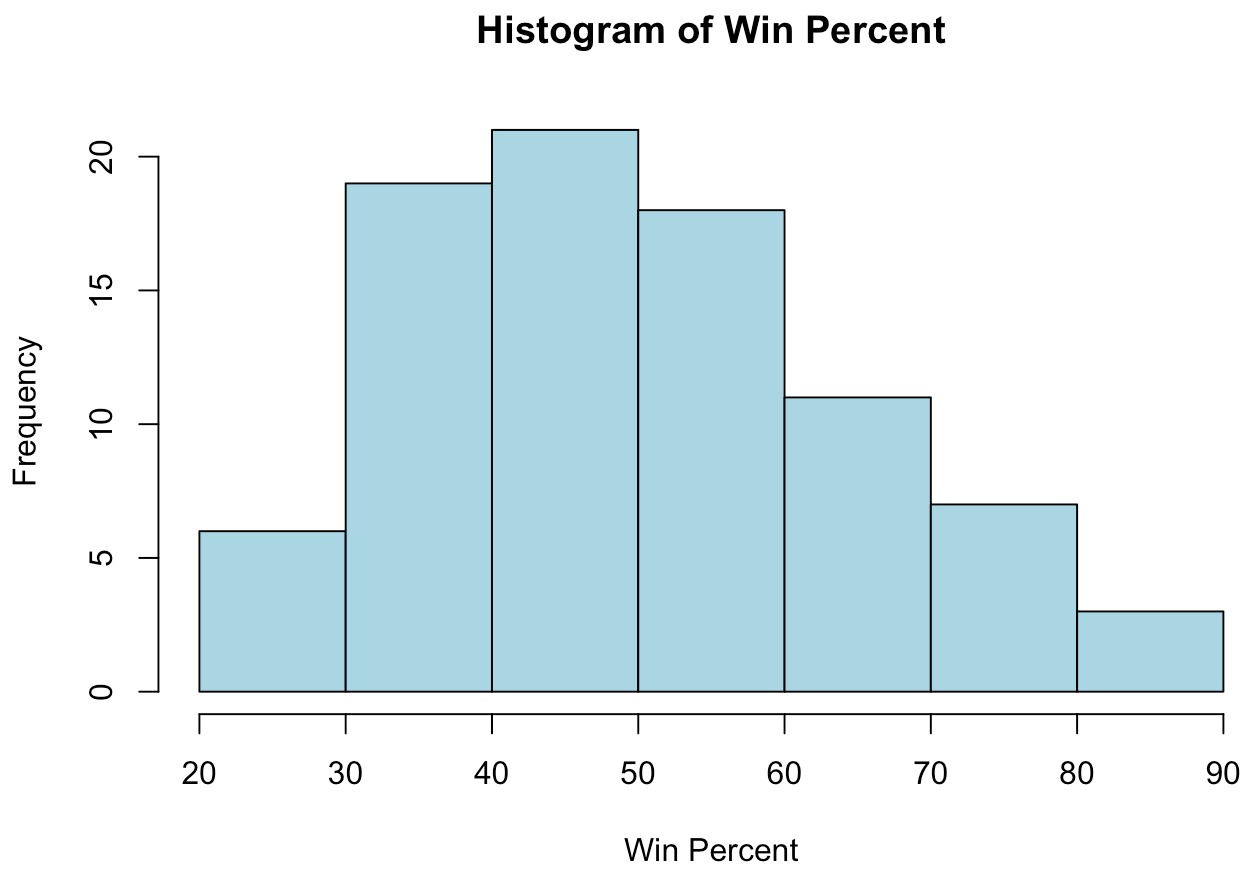
```
#row 12 (winpercent) looks on a different scale then the other rows. Although the
```

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

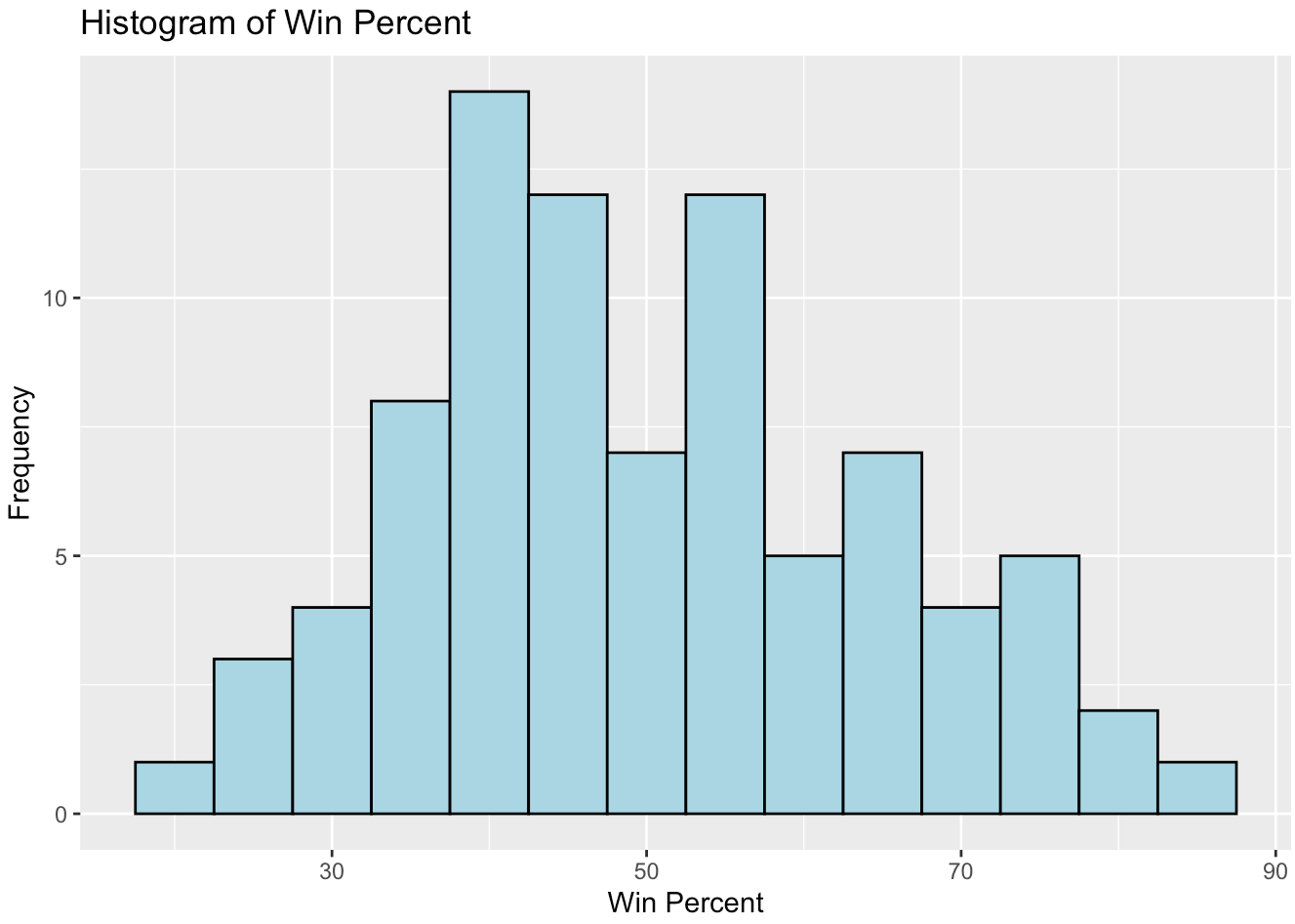
```
#0 indicates the absence of chocolate in that candy type.
#1 indicates the presence of chocolate in that candy type.
```

Q8 - Plot a histogram of winpercent values

```
# Plot histogram using base R
hist(candy$winpercent, main="Histogram of Win Percent", xlab="Win Percent", col="

library(ggplot2)
```



```
# Plot histogram using ggplot
ggplot(candy, aes(x=winpercent)) +
  geom_histogram(binwidth=5, fill="lightblue", color="black") +
  labs(title="Histogram of Win Percent", x="Win Percent", y="Frequency")
```



Q9 - Is the distribution of winpercent values symmetrical?

```
# Load moments package for skewness calculation
library(moments)

# Calculate skewness
winpercent_skewness <- skewness(candy$winpercent, na.rm = TRUE)
winpercent_skewness #answer = 0.33%
```

```
[1] 0.3264194
```

```
# Since it only has a skewness of 0.33%. It is pretty good symmetrical. 0.33% mea
```

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

```
        # Calculate mean and median
        mean_winpercent <- mean(candy$winpercent, na.rm = TRUE)
        median_winpercent <- median(candy$winpercent, na.rm = TRUE)

        mean_winpercent #answer = 50.3
```

```
[1] 50.31676
```

```
        median_winpercent #answer = 47.8\
```

```
[1] 47.82975
```

```
        # Since the mean (50.3) is above 50%, the center of the distribution is above 50%
```

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

```
        # Calculate average winpercent for chocolate and fruity candies
        mean_chocolate <- mean(candy$winpercent[as.logical(candy$chocolate)], na.rm = TRU
        mean_fruity <- mean(candy$winpercent[as.logical(candy$fruity)], na.rm = TRUE)

        mean_chocolate #answer = 60.9
```

```
[1] 60.92153
```

```
        mean_fruity #answer = 44.1
```

```
[1] 44.11974
```

```
        # So the average of chocolate is higher
```

Q12 - Is this difference statistically significant?

```
        # Perform t-test between chocolate and fruity candies
        t_test_result <- t.test(candy$winpercent[as.logical(candy$chocolate)],
                                candy$winpercent[as.logical(candy$fruity)],
                                na.rm = TRUE)

        t_test_result
```

```
    Welch Two Sample t-test

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

```
candy$winpercent[as.logical(candy$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
```

```
        #  p-value = 2.871e-08. this is significant
```

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

```
        # Get the five least liked candy types
        least_liked_candies <- head(candy[order(candy$winpercent), ], n=5)
        least_liked_candies #Nik L Nip, Boston Baked Beans, Chiclets, Super Bubble, Jawbu
```

	chocolate	fruity	caramel	peanutyalmondy	nougat	crispedricewafer	hard	bar	pluribus	sugarpercent	pricepercent	winpercent
Nik L Nip	0	1	0	0	0	0	0	0	1	0.197	0.976	22.44534
Boston Baked Beans	0	0	0	1	0	0	0	0	1	0.313	0.511	23.41782
Chiclets	0	1	0	0	0	0	0	0	1	0.046	0.325	24.52499
Super Bubble	0	1	0	0	0	0	0	0	0	0.162	0.116	27.30386
Jawbusters	0	1	0	0	0	0	1	0	1	0.093	0.511	28.12744

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

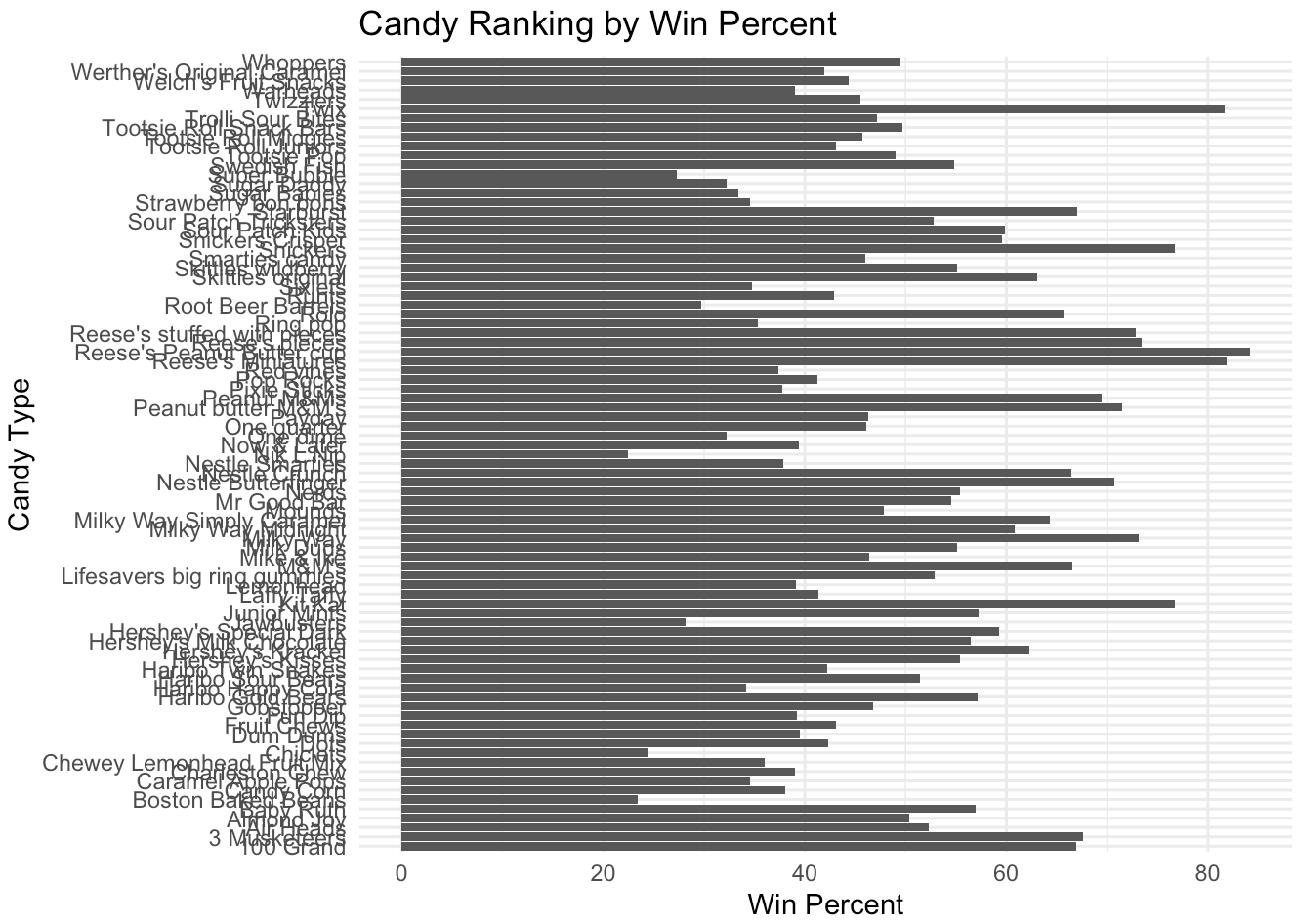
```
        # Get the top 5 favorite candy types
        favorite_candies <- head(candy[order(-candy$winpercent), ], n=5)
        favorite_candies #Reese's Peanut Butter Cup, Reese's Miniatures, Twix, Kit Kat, S
```

	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
```

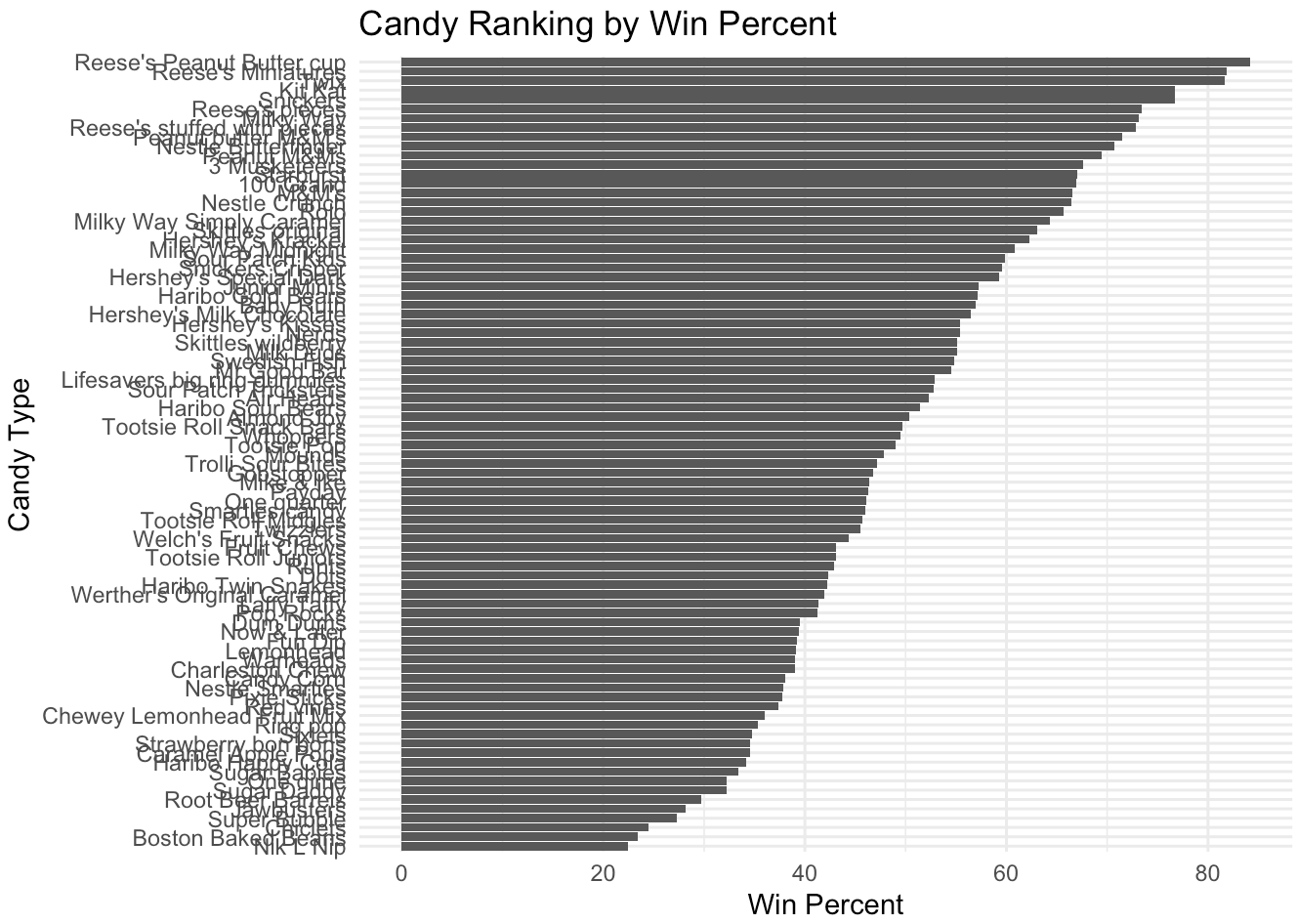
Q15 - Make a first barplot of candy ranking based on winpercent values.

```
# Load the ggplot2 package
library(ggplot2)

# Create the bar plot
ggplot(candy) +
  aes(x = winpercent, y = rownames(candy)) +
  geom_bar(stat = "identity") +
  labs(title = "Candy Ranking by Win Percent", x = "Win Percent", y = "Candy Type
  theme_minimal()
```



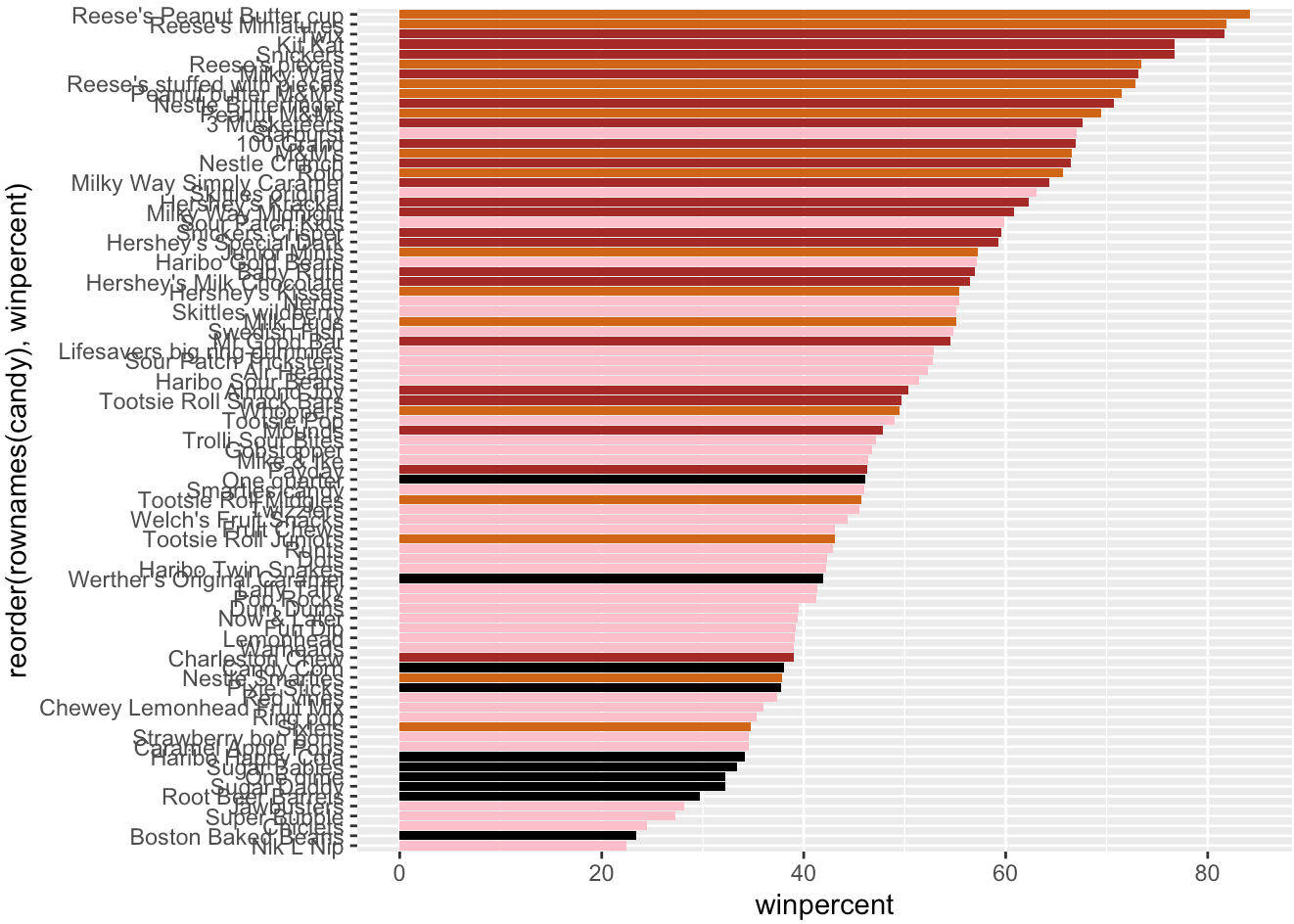
Q16 - This is quite ugly, use the reorder() function to get the bars sorted by winpercent?

```
# Load the ggplot2 package
library(ggplot2)

# Create the bar plot
ggplot(candy) +
  aes(winpercent, reorder(rownames(candy),winpercent)) +
  geom_bar(stat = "identity") +
  labs(title = "Candy Ranking by Win Percent", x = "Win Percent", y = "Candy Type
  theme_minimal()
```

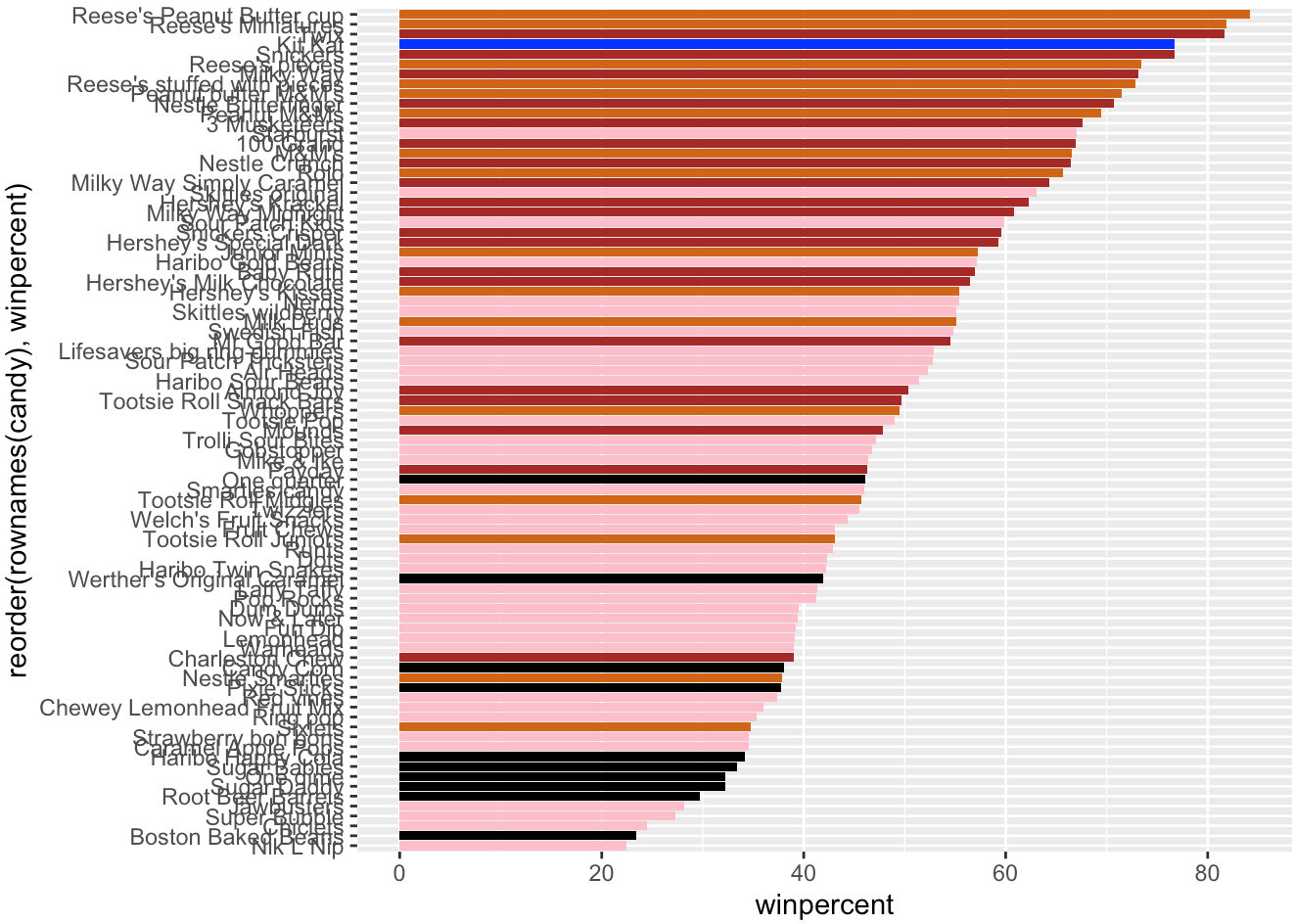


Interim

```
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_cols)
```




```
#If you want to color Kit Kat
my_cols[rownames(candy)=="Kit Kat"] = "blue"

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



Q17 - What is the worst ranked chocolate candy?

```
#Sixlets
```

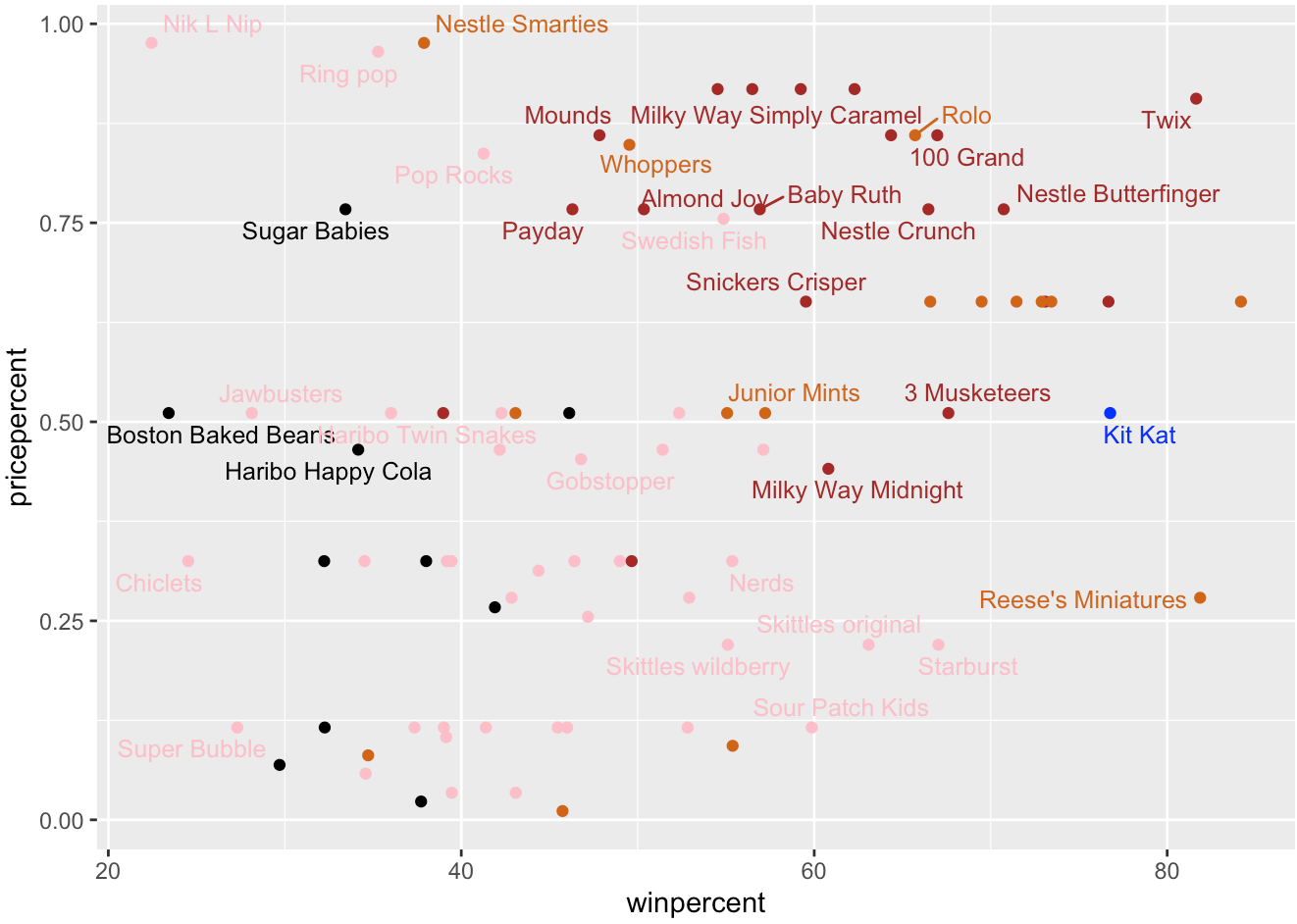
Q18 - What is the best ranked fruity candy?

```
#Starburst
```

Interim

```
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: 50 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?

```
highest_rank <- rownames(candy)[which.max(candy$winpercent / candy$pricepercent)]
candy["Tootsie Roll Midgies",]$winpercent
```

```
[1] 45.73675
```

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

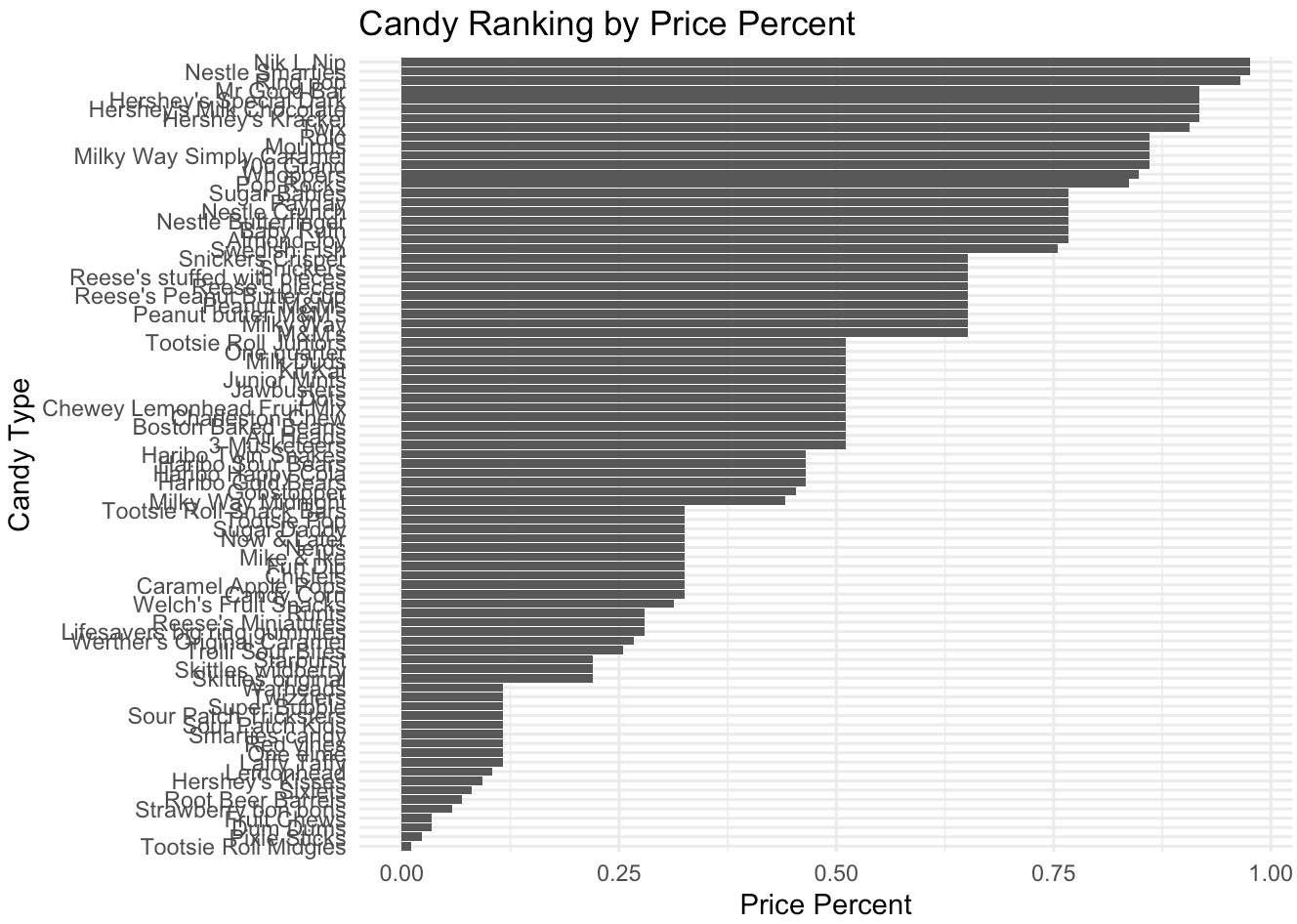
```
ord <- order(candy$pricepercent, decreasing = TRUE)
head( candy[ord,c(11,12)], n=5 )
```

	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

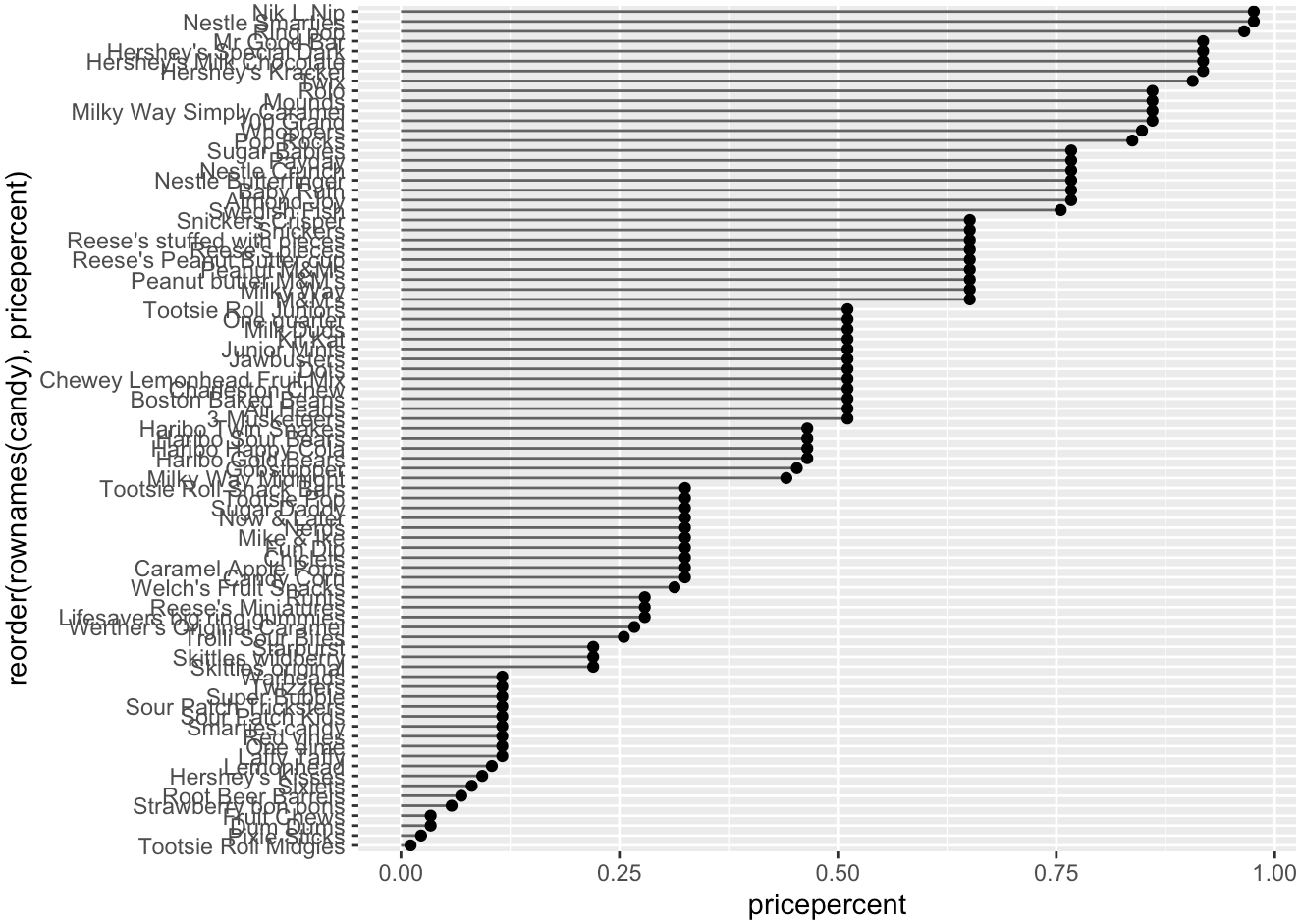
```
#Answer = Nik L Nip, Nestle Smarties, Ring pop, Hershey's Krackel and Hershey's M
#Answer = Nik L Nip is least popular
```

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()

```
# Load the ggplot2 package
library(ggplot2)

# Create the bar plot
ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy),pricepercent)) +
  geom_bar(stat = "identity") +
  labs(title = "Candy Ranking by Price Percent", x = "Price Percent", y = "Candy
  theme_minimal()
```




```
# Make a lollipop chart of pricepercent
ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy), pricepercent)) +
  geom_segment(aes(yend = reorder(rownames(candy), pricepercent),
                   xend = 0), col="gray40") +
    geom_point()
```



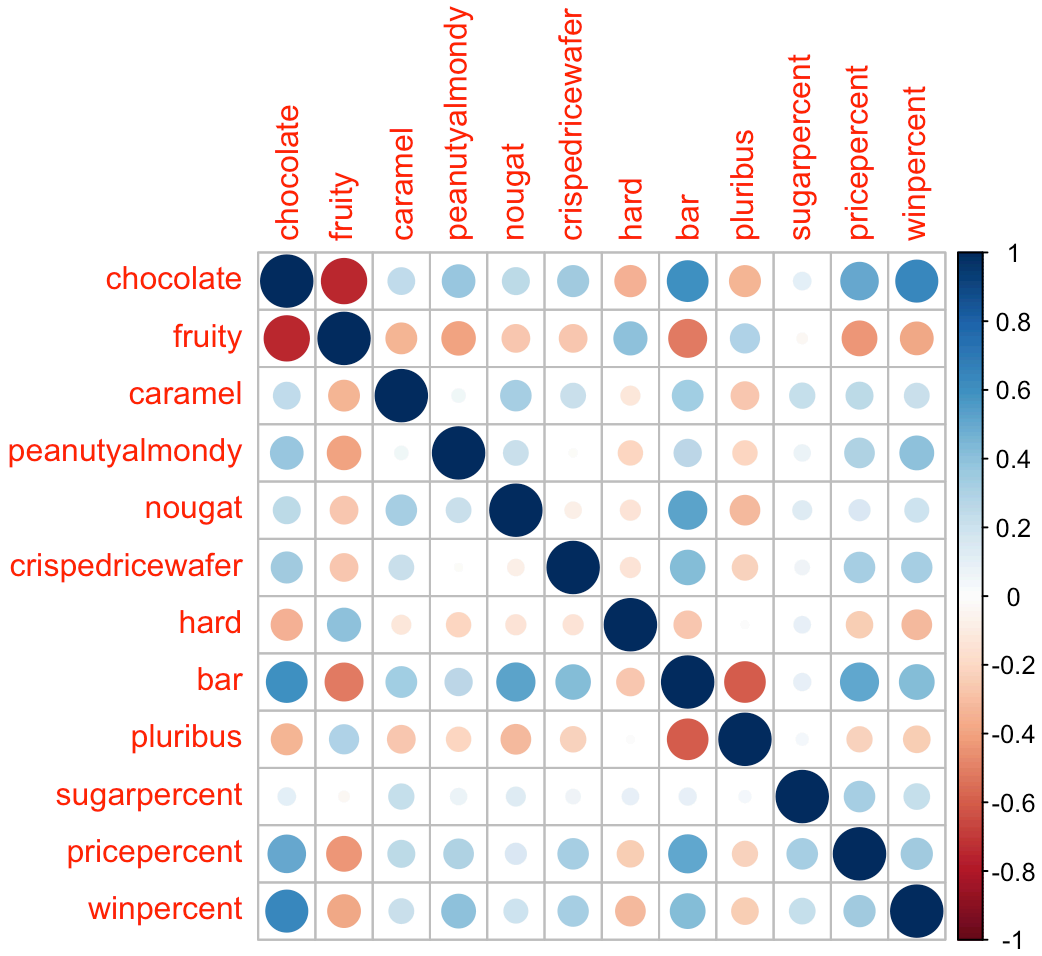
Interim

```
        library(corrplot)
```

```
Warning: package 'corrplot' was built under R version 4.3.3
```

```
corrplot 0.95 loaded
```

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



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

```
# fruity and chocolate & bar and pluribus
```

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

```
# chocolate and bar & chocolate and winpercent
```

Interim

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

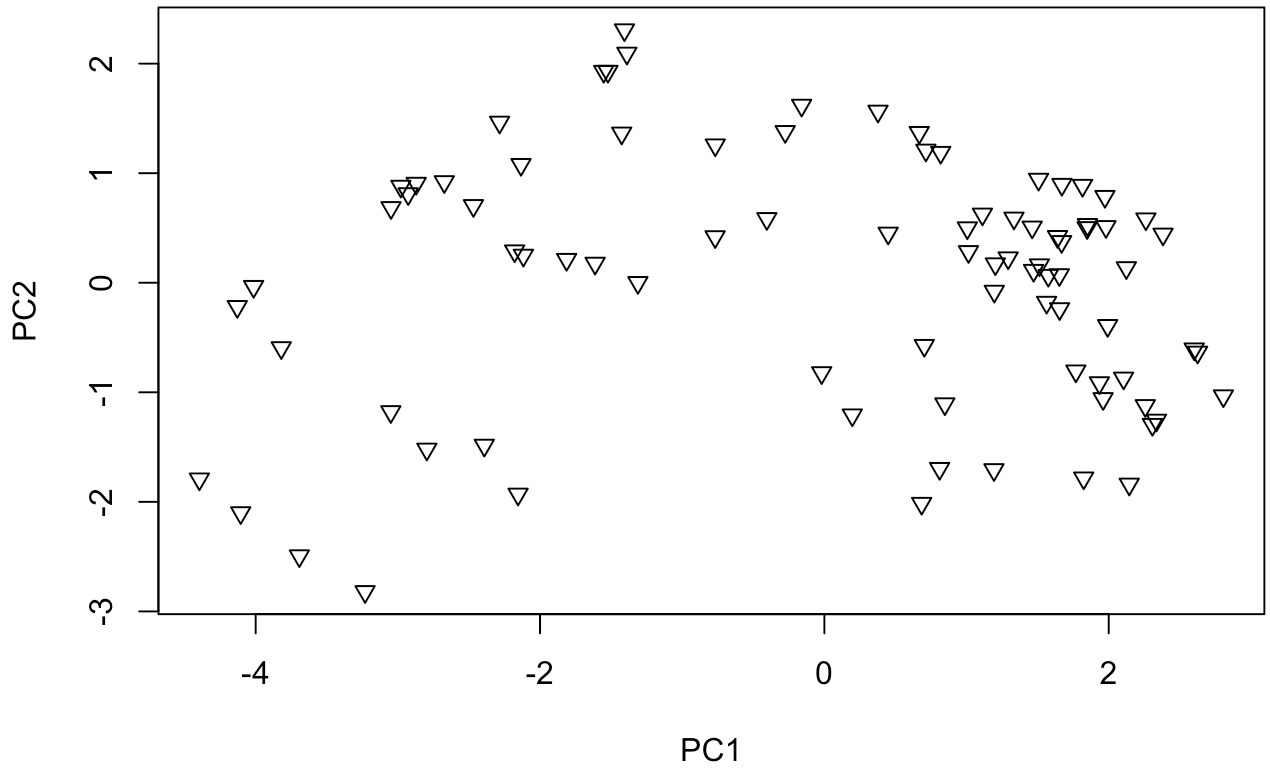
```
Importance of components:
                          PC1     PC2     PC3      PC4     PC5      PC6      PC7
```

```
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
```

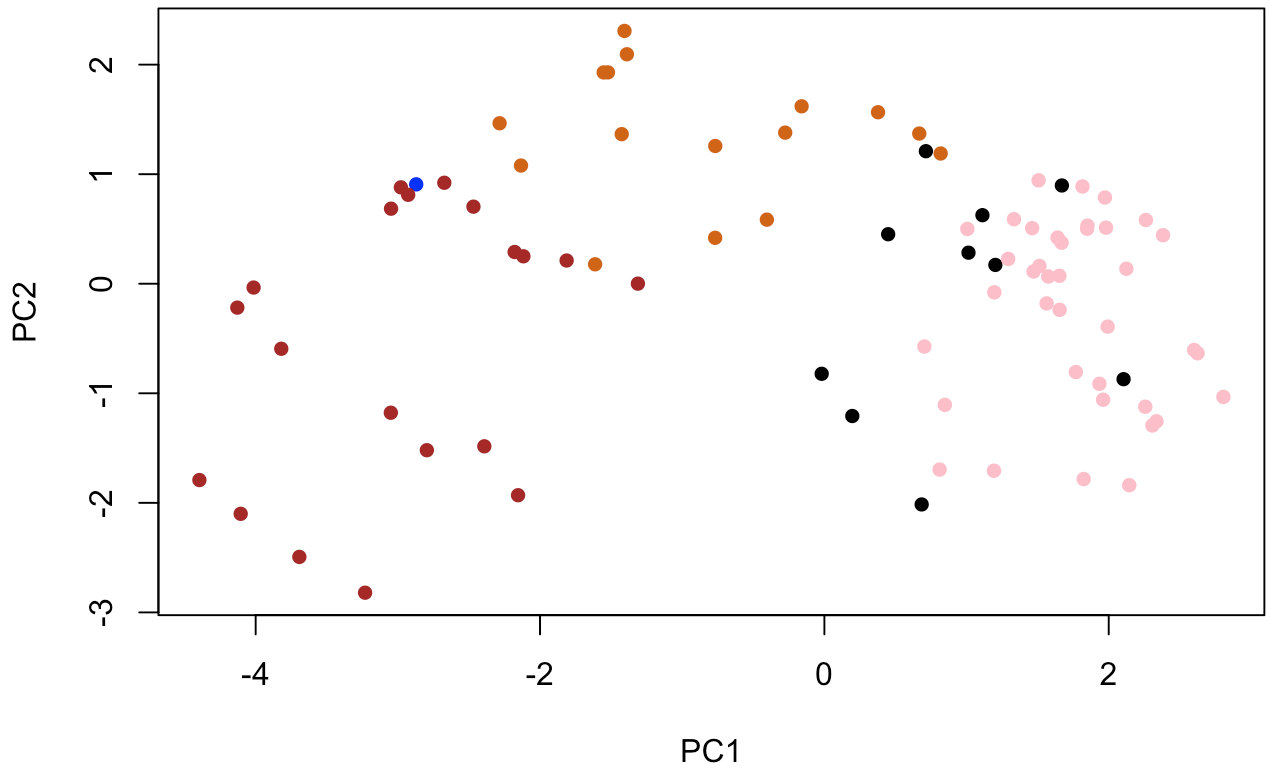
```
pca$rotation[,1]
```

```
  chocolate           fruity          caramel   peanutyalmondy
 -0.4019466        0.3683883       -0.2299709       -0.2407155
     nougat crispedricewafer             hard              bar
 -0.2268102       -0.2215182        0.2111587       -0.3947433
   pluribus     sugarpercent     pricepercent       winpercent
  0.2600041       -0.1083088       -0.3207361       -0.3298035
```

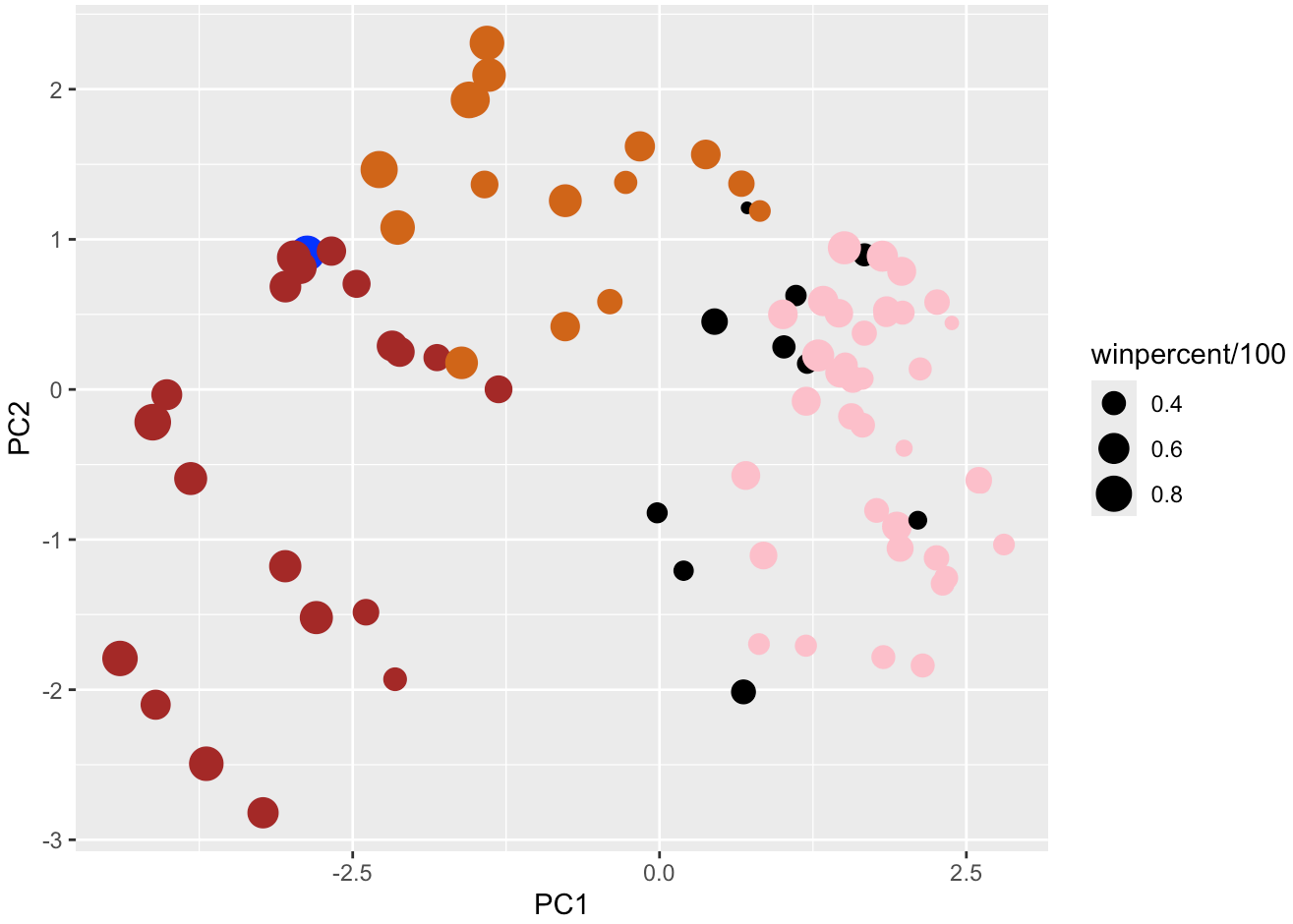
```
plot(pca$x[,1:2], pch=6)
```




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

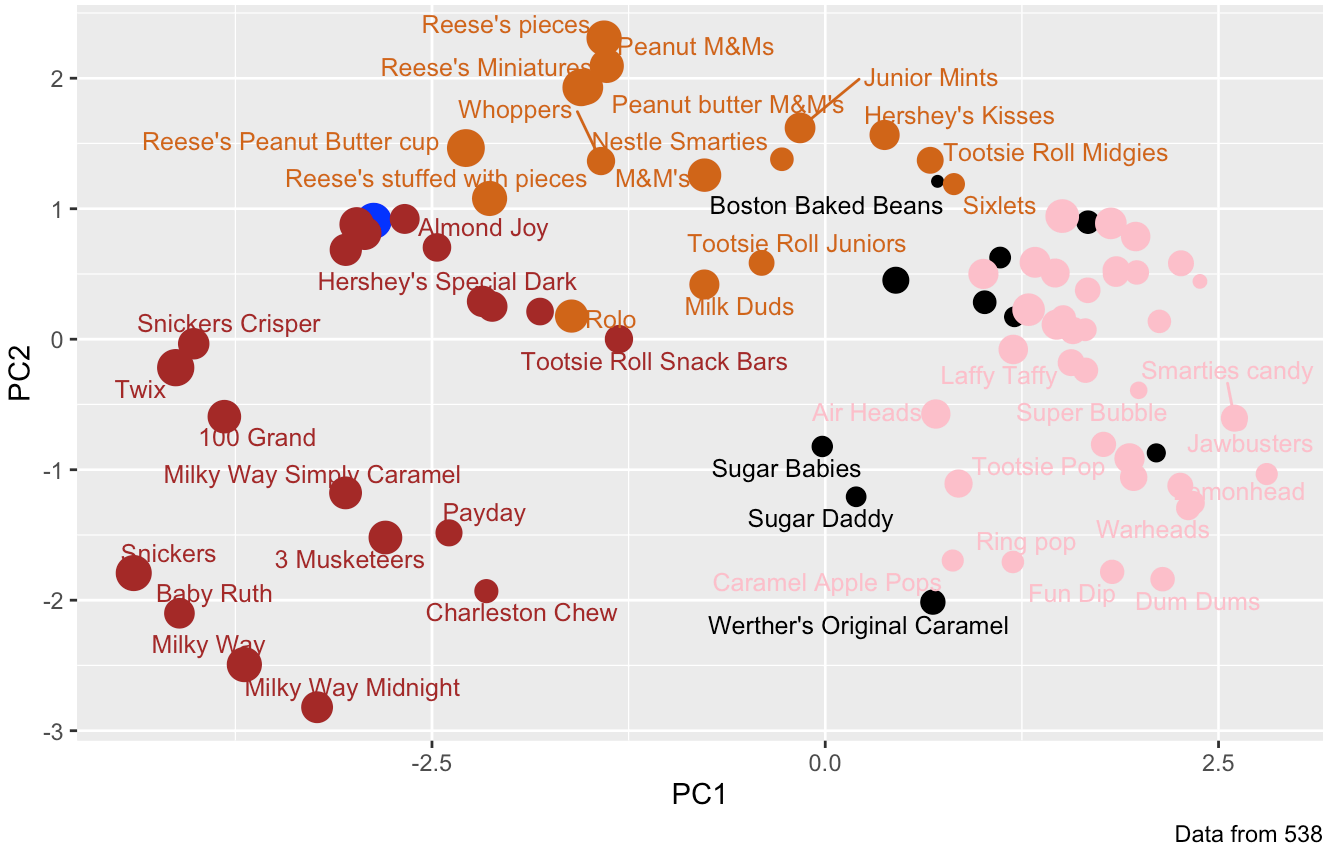



```
# Make a new data-frame with our PCA results and candy data
my_data <- cbind(candy, pca$x[,1:3])

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
```




```
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 (li
       caption="Data from 538")
```

```
Warning: ggrepel: 39 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

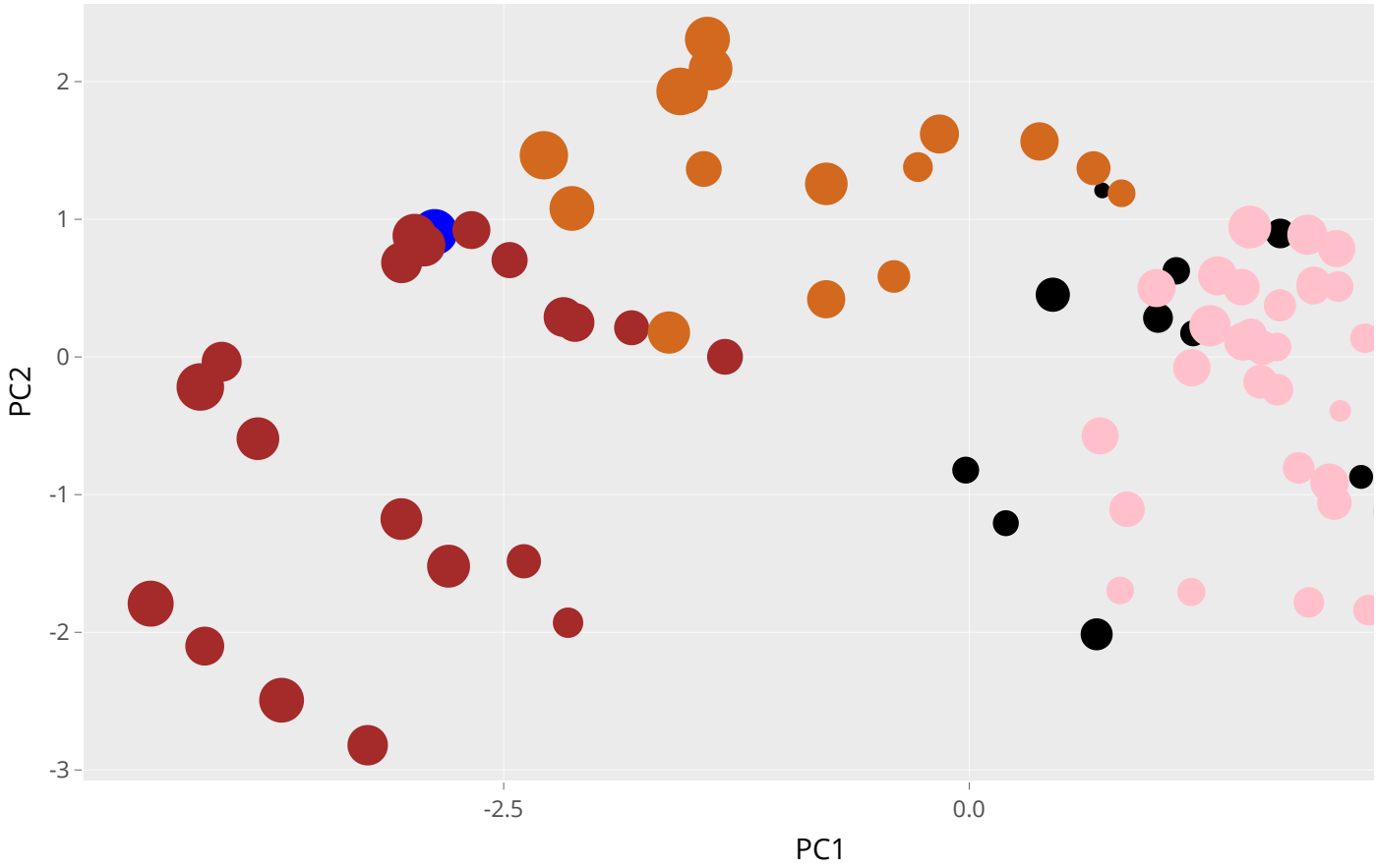


Interim 2

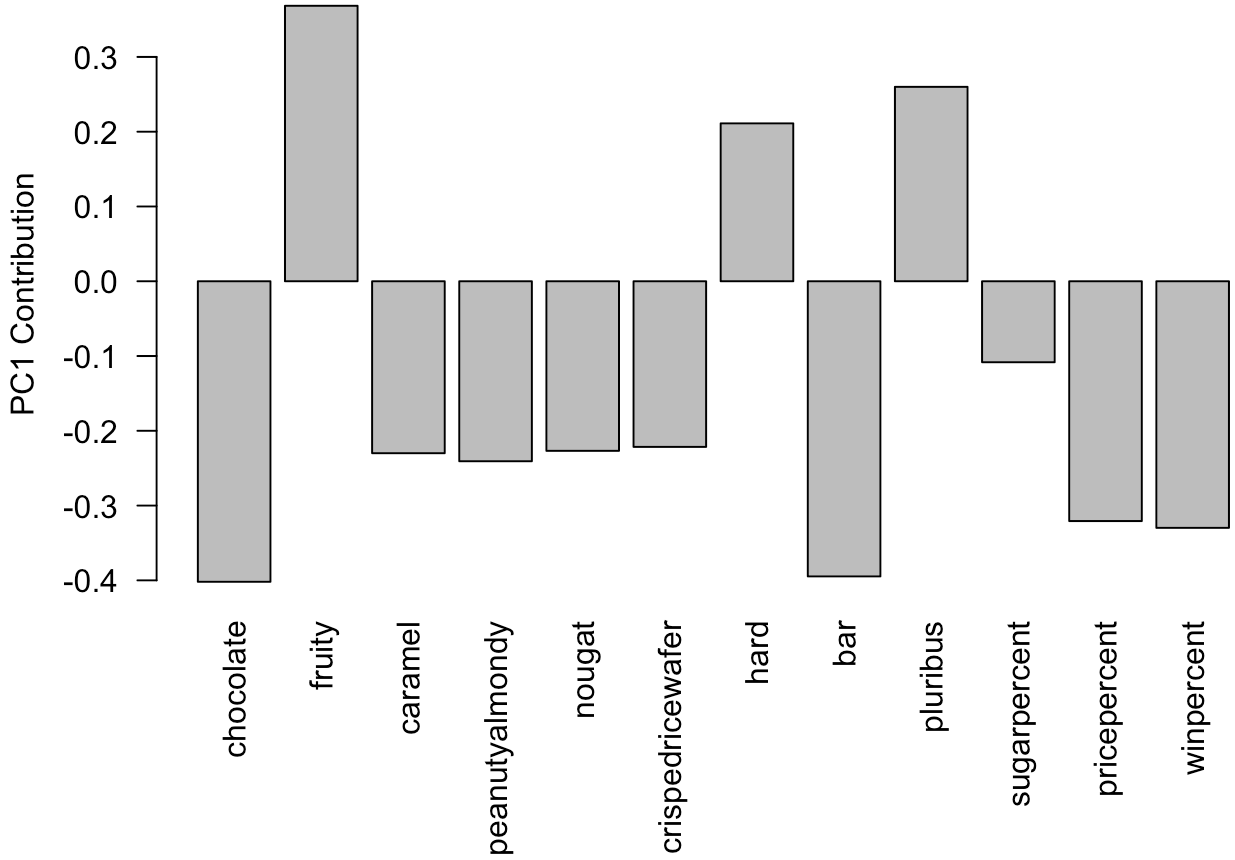
```
library(plotly)
```

```
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")
```



Q24 - What original variables are picked up strongly by PC1 in the positive direction? Do these make sense to you?

```
#Strongly picked up are fruity, hard, and pluribus
# Yes it makes sense Fruity hard candies in a bag.
```