Class 11 Halloween Mini-Project

AUTHOR
Ryan Chung A15848050

Halloween Mini-Project: Importing candy data

```
candy <- read.csv("candy-data.csv", row.names = 1)
#candy
```

> Q1. How many different candy tpes are in this dataset?

There are 85 different candy types

```
nrow(candy)
```

```
[1] 85
```

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

There are 38 fruity candies in the dataset.

```
candy$fruity
```

```
 [1] 0 0 0 0 1 0 0 0 0 1 0 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 1 0 0 1 1 1 0 0 1 0 0 0
[39] 0 0 0 1 0 0 1 1 0 0 0 1 1 0 0 0 0 1 0 0 1 0 1 1 0 1 0 0 1 1 1 1 0 0 1 1 1 0
[77] 0 0 1 0 1 1 1 0 0
```

```
table(candy$fruity)
```

```

 0  1 
47 38 
```

```
#also: works:
sum(candy$fruity)
```

```
[1] 38
```

> Q What are these fruity candy?

```
rownames(candy[candy$fruity == 1,  ])
```

```
 [1] "Air Heads"                   "Caramel Apple Pops"         
 [3] "Chewey Lemonhead Fruit Mix"  "Chiclets"                   
 [5] "Dots"                        "Dum Dums"                   
 [7] "Fruit Chews"                 "Fun Dip"                    
 [9] "Gobstopper"                  "Haribo Gold Bears"          
[11] "Haribo Sour Bears"           "Haribo Twin Snakes"         
[13] "Jawbusters"                  "Laffy Taffy"                
[15] "Lemonhead"                   "Lifesavers big ring gummies"
[17] "Mike & Ike"                  "Nerds"                      
[19] "Nik L Nip"                   "Now & Later"                
[21] "Pop Rocks"                   "Red vines"                  
[23] "Ring pop"                    "Runts"                      
[25] "Skittles original"           "Skittles wildberry"         
[27] "Smarties candy"              "Sour Patch Kids"            
[29] "Sour Patch Tricksters"       "Starburst"                  
[31] "Strawberry bon bons"         "Super Bubble"               
[33] "Swedish Fish"                "Tootsie Pop"                
[35] "Trolli Sour Bites"           "Twizzlers"                  
[37] "Warheads"                    "Welch's Fruit Snacks"       
```

How often does my favorite candy win?

```r
candy["Reese's Peanut Butter cup", ]$winpercent
```

```
[1] 84.18029
```

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

```
[1] 76.7686
```

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

Reese’s Peanut Butter cup: winpercent = 84%

> Q4. What is the `winpercent` value for “Kit Kat”?

Kit Kat = 76.7686%

> Q5. What is the `winpercent` value for “Tootsie Roll Snack Bars”

Tootsie Roll = 49.653503%

Skim function

```
library("skimr")
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?

The winpercent variable is not on a 0 to 1 scale, and instead is on a 0 to 100 scale.

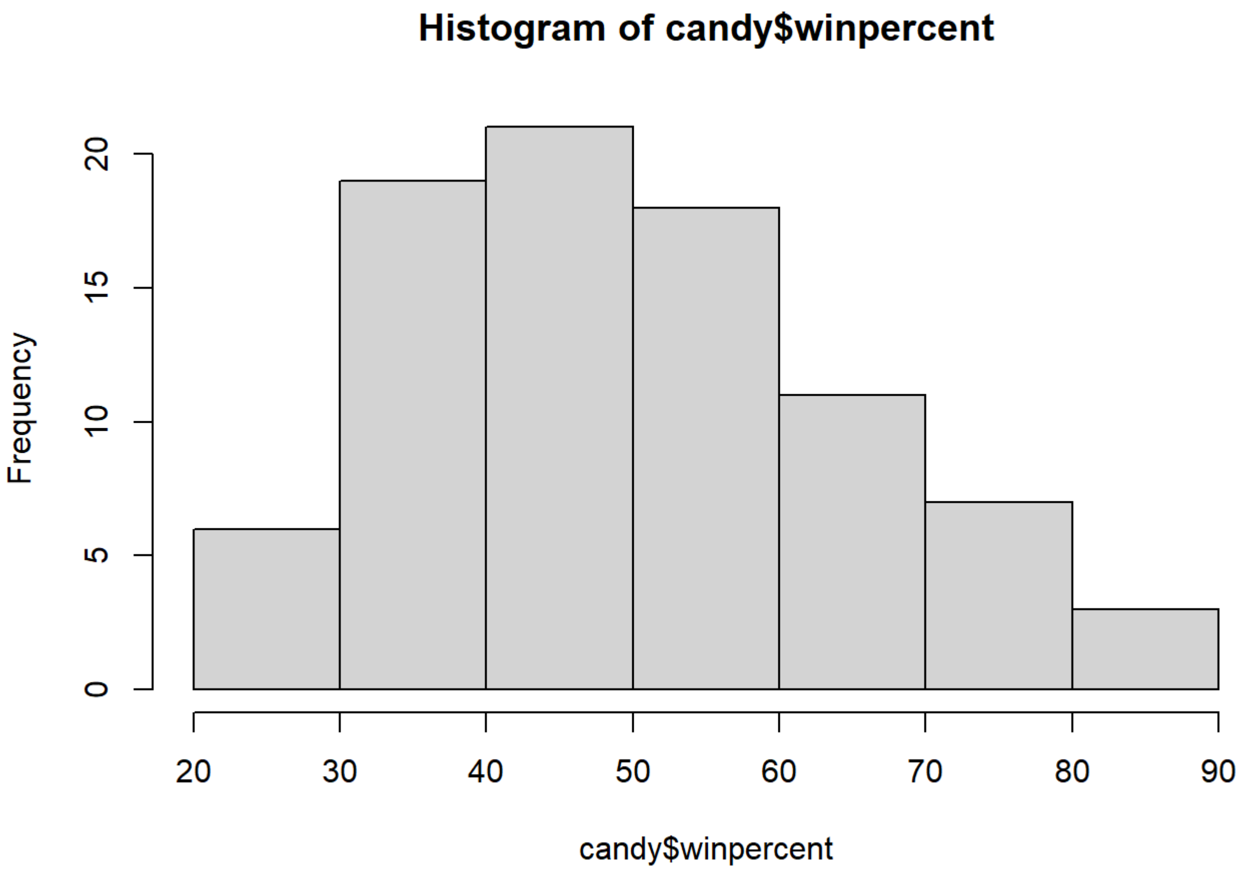
> Q7. What do you think a zero and one represent for the `candy$chocolate` column?

The 0 represents a non-chocolate candy and the 1 represents a chocolate candy classification.

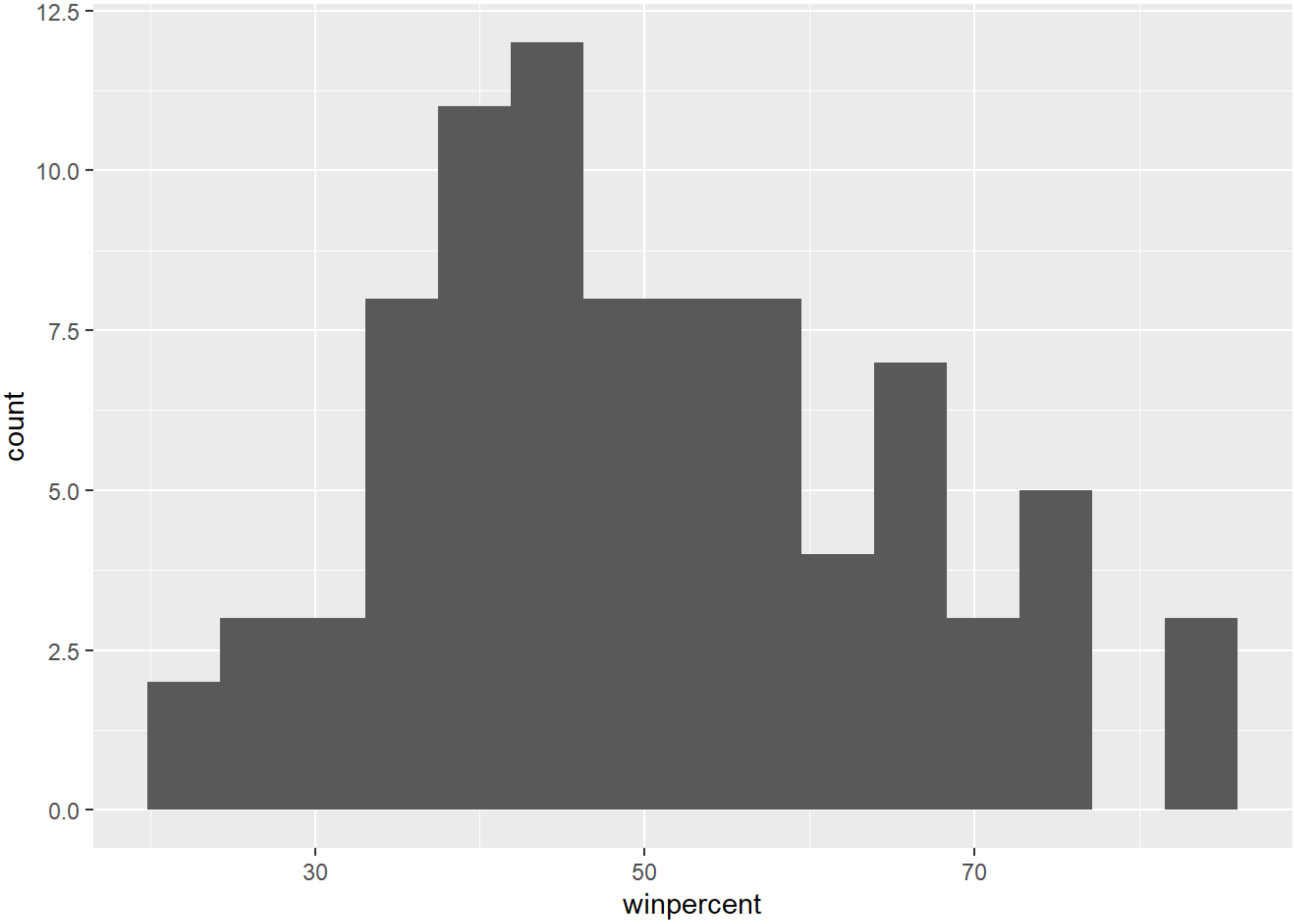
> Q8. Plot a histogram of winpercent values

```
library(ggplot2)

hist(candy$winpercent)
```



```
ggplot(candy, aes(winpercent)) +
  geom_histogram( bins = 15)
```



> Q9. Is the distribution of winpercent values symmetrical?

No

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

The center is below 50% with a mean of 50.3167638

Let’s find the mean

```
mean(candy$winpercent)
```

```
[1] 50.31676
```

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

On average chocolate candy(60.9%) is ranked 16.8% higher than fruity candy(44.1%)

To answer this question I will need to ‘subset’ the candy dataset to just chocolate candy, get their winpercent values, and then calculate the mean of these. Then do the same for fruity candy and compare.

```r
#candy[ , candy$chocolate == 1]$winpercent NO NEED TO DEFINE ROWS
#mean(candy$winpercent[ , candy$chocolate == 1])

#Prof did it this way  it makes the 0/1's into logicals and only works with Trues

#subset for chocolate
chocolate.candy <- candy[as.logical(candy$chocolate), ]
#grab the winpercent
choc.win <- chocolate.candy$winpercent
#calulate the winpercent mean
c <- mean(chocolate.candy$winpercent)

fruity.candy <- candy[as.logical(candy$fruity), ]
fruit.win <- fruity.candy$winpercent
d <- mean(fruity.candy$winpercent)


#My way
a <- mean(candy$winpercent[candy$chocolate == 1])

b <- mean(candy$winpercent[candy$fruity == 1])

print( round(c(a,b,c,d), 2))
```

```
[1] 60.92 44.12 60.92 44.12
```

> Q12. Is this difference statistically significant?

Yes, the difference between chocolate and fruit winpercent is statistically significant.

```r
t.test(choc.win, fruit.win)
```

```

	Welch Two Sample t-test

data:  choc.win and fruit.win
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 
```

#Overall Candy Rankings There is a base R function called `sort()` for sorting input vectors

```r
x <- c(5,2,10)
```

```
sort(x)
```

```
[1]  2  5 10
```

The related function to `sort()` that is often more useful is called `order()`. It returns the 'indices' of the input that would result in the 'proper' sort.

```
order(x)
```

```
[1] 2 1 3
```

```
x[order(x)]
```

```
[1]  2  5 10
```

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

Jawbusters,Super Bubble, Chiclets, Boston Baked Beans, and Nik L Nip.

I can order by `winpercent`

```
ord <- order(candy$winpercent, decreasing = FALSE)
#candy[ord, ]
#head(candy[ord,], 5)

# failed attempt
#order(candy$winpercent)
#winp <- candy$winpercent
#c1 <- candy[order(winp)]
```

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

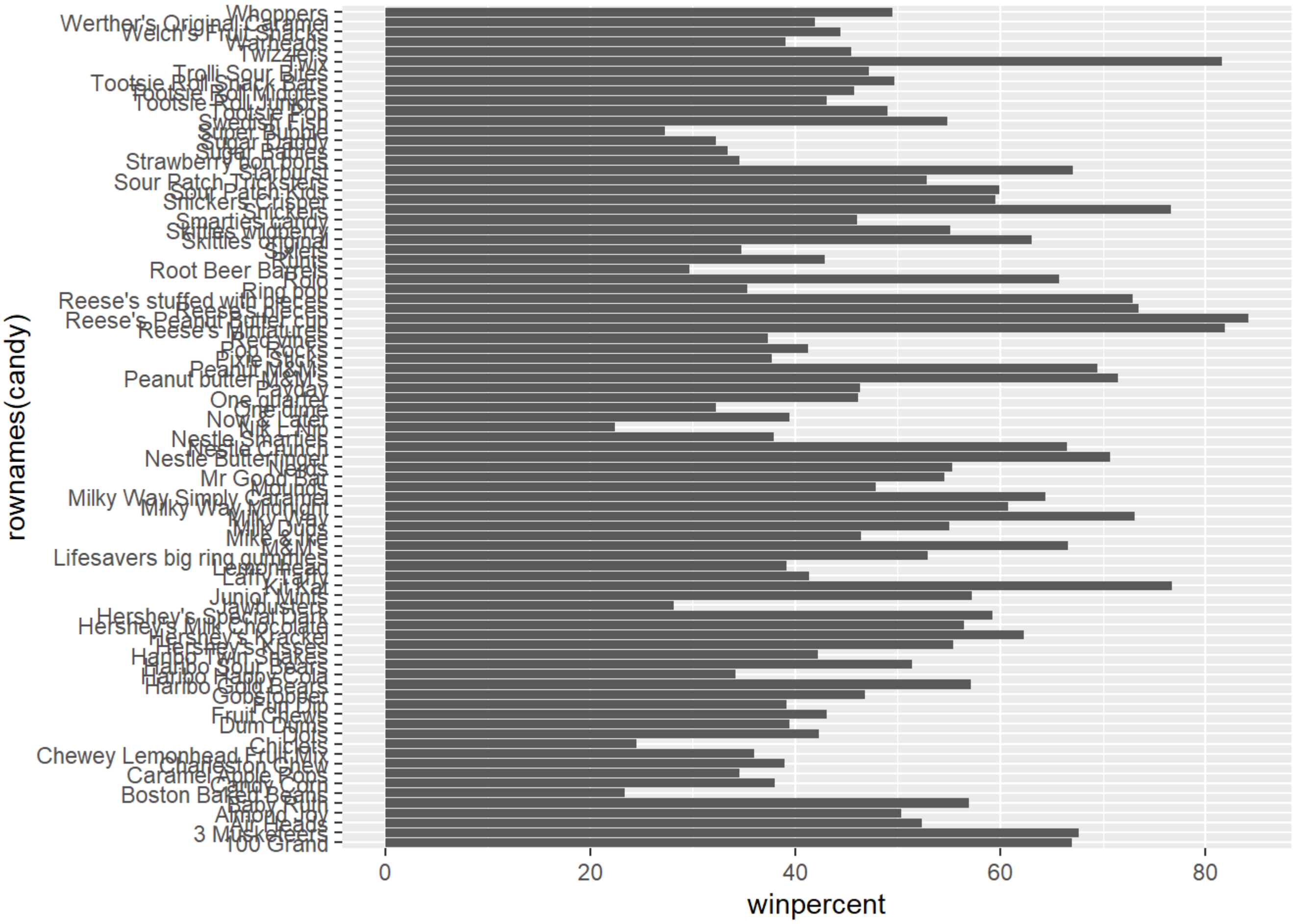
Snickers, Kit kat, Twix, Reese's Miniatures, and Reese's Peanut Butter cup.

```
ord2 <- order(candy$winpercent, decreasing = TRUE)
#candy[ord2,]
#head(candy[ord2,], 5)
```

> Q15. Make a barplot of candy ranking based on winpercent

```
library(ggplot2)

candp <- ggplot(candy) +
  aes(winpercent, rownames(candy)) +
  geom_col()
candp
```



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

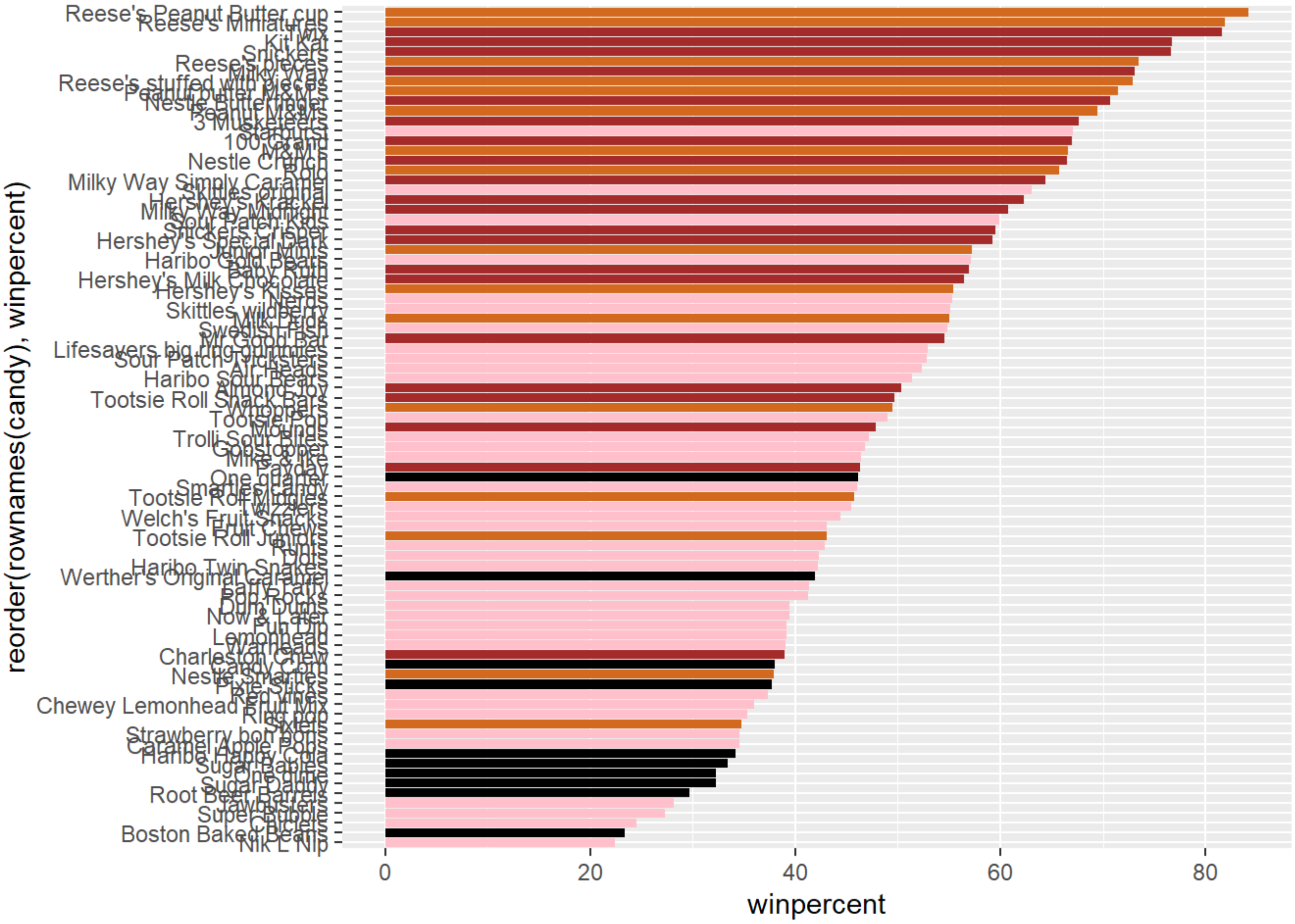
```
candp <-candp + aes(winpercent, reorder(rownames(candy), winpercent))
```

Adding color

```
#make a color vector of all black replicate black as many times as there are rows in the candy da

my_cols = rep("black", nrow(candy))
#overwrite the (TRUE) chocolate entries as the color chocolate
my_cols[as.logical(candy$chocolate)] = "chocolate"
my_cols[as.logical(candy$bar)] = "brown"
my_cols[as.logical(candy$fruity)] = 'pink'

candp <-  candp + geom_col(fill = my_cols)
candp
```



> Q17. What is the worst ranked chocolate candy?

Sixlets

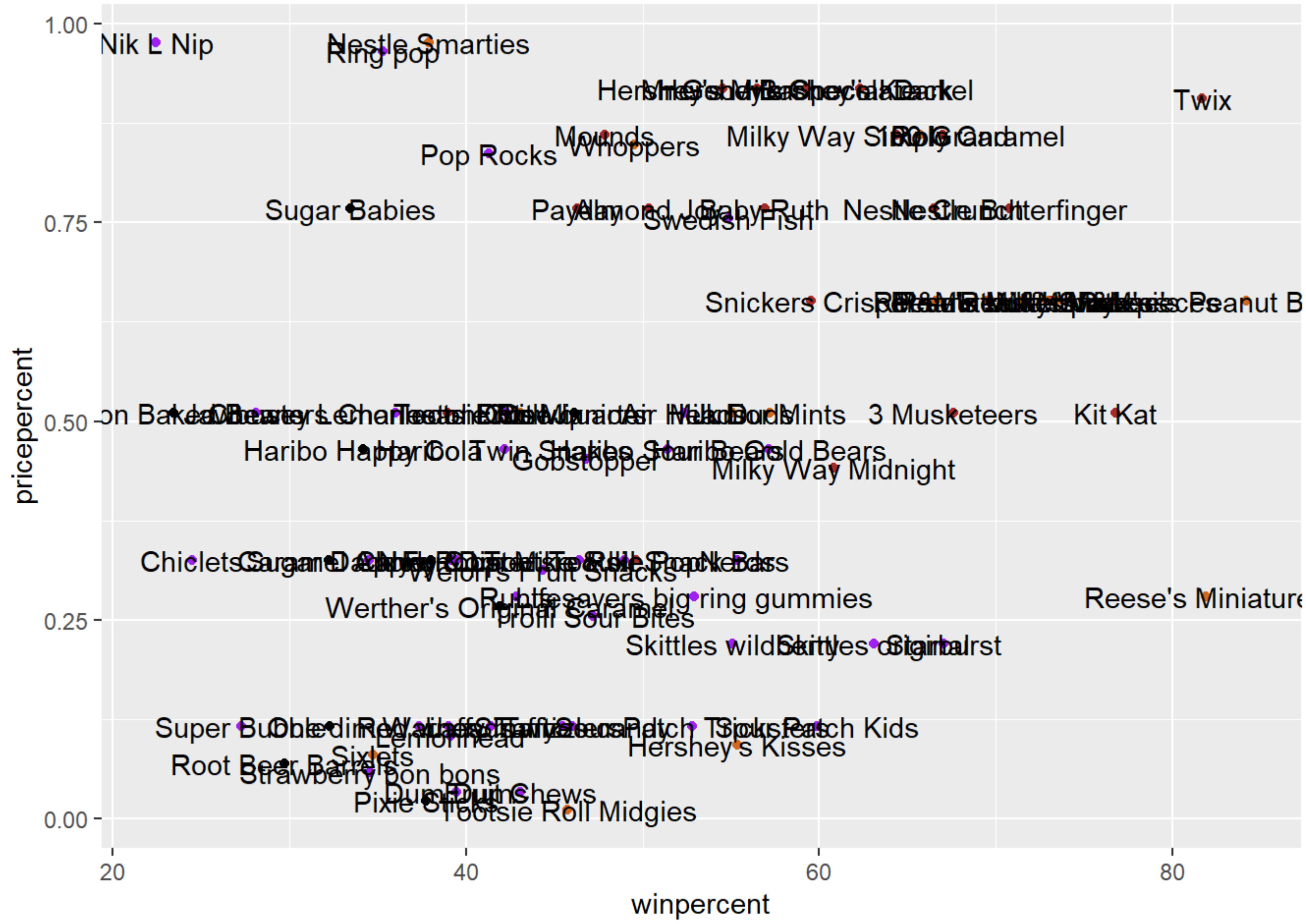
> Q18. What is the best ranked fruity candy?

Nik L Nip

#Taking a look at pricepoint

```
#change col for clarity
my_cols[as.logical(candy$fruity)] = 'purple'

# How about a plot of price vs win
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
  geom_text()
```

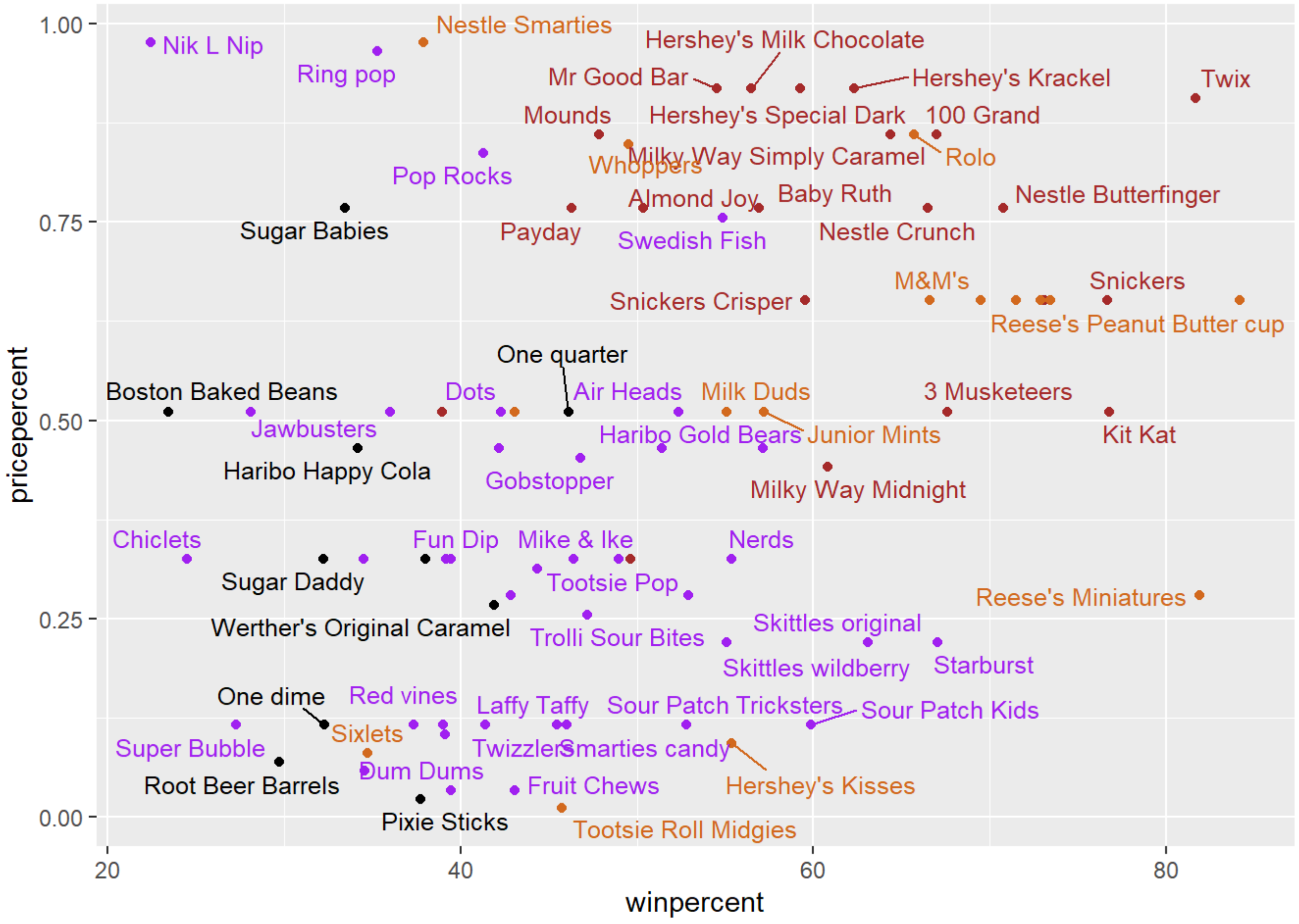


```
#becomes unreadable cause overlaps
```

To deal with overlapping labels I can sue the **ggrepel** package

```
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 = 10)
```

```
Warning: ggrepel: 20 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



> Q19. Which candy is the highest rank in terms of winpercent for the least money?

Reese’s Miniatures

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

Nik L Nip, Nestle Smarties, Ring pops, Hershey’s krack and Milk chocolate, with the Nik L Nip being the least popular

```r
ord3 <- order(candy$pricepercent, decreasing = TRUE)
ord3
```

```
 [1] 45 63 56 24 25 26 41 80  1 39 40 57 85 50  6  7 43 44 47 71 74 33 34 37 48
[26] 53 54 55 65 66  2  4  5  8 11 12 14 27 28 29 36 76 19 20 21 22 18 38  9 10
[51] 13 17 35 42 46 72 75 78 83 32 52 59 84 79 61 62 69  3 30 51 64 67 68 73 81
[76] 82 31 23 60 58 70 15 16 49 77
```

```r
head(candy[ord3, ], 5)
```

```
                         chocolate fruity caramel peanutyalmondy nougat
Nik L Nip                        0      1       0              0      0
```

```
Nestle Smarties                  1      0       0              0      0
Ring pop                         0      1       0              0      0
Hershey's Krackel                1      0       0              0      0
Hershey's Milk Chocolate         1      0       0              0      0
                         crispedricewafer hard bar pluribus sugarpercent
Nik L Nip                               0    0   0        1        0.197
Nestle Smarties                         0    0   0        1        0.267
Ring pop                                0    1   0        0        0.732
Hershey's Krackel                       1    0   1        0        0.430
Hershey's Milk Chocolate                0    0   1        0        0.430
                         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
```

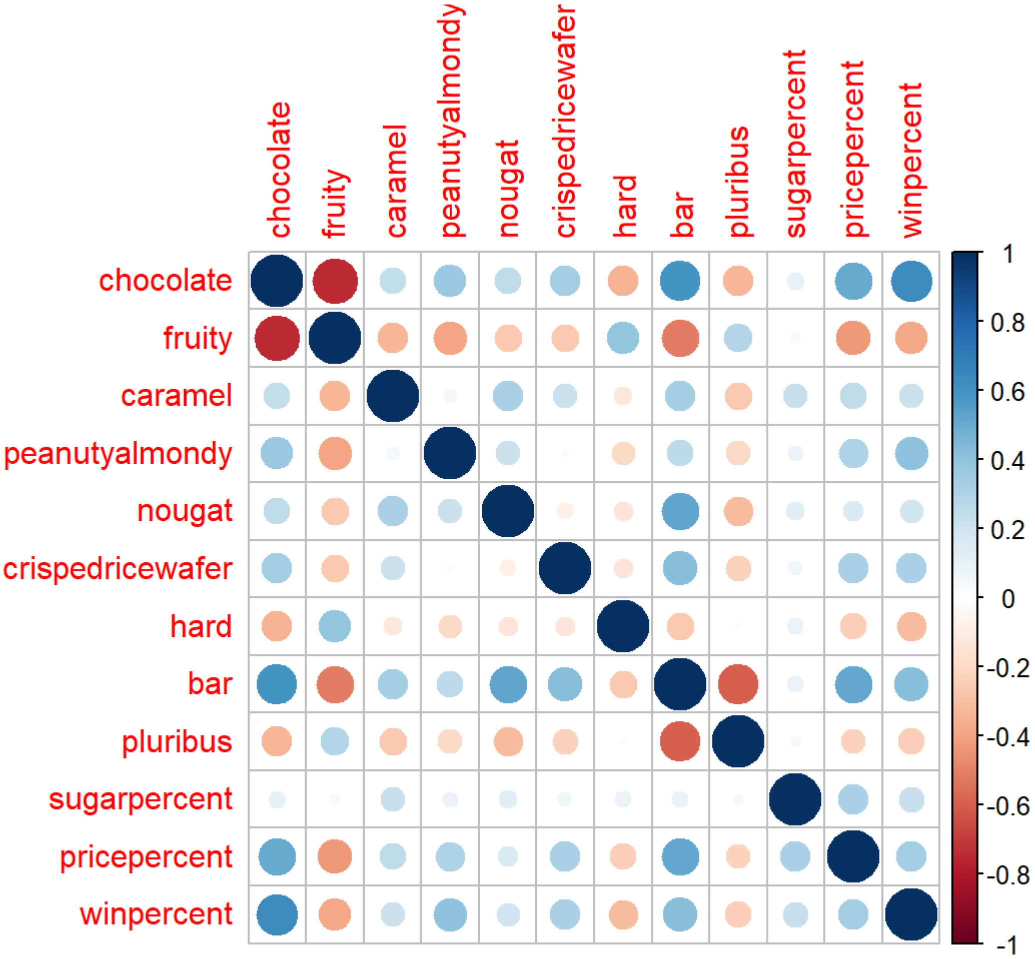
#Exploring the correlation structure

Pearson correlation goes between -1 and +1 with zero indicating no correlation. Values close to 1 are highly correlated.

```
library(corrplot)
```

```
corrplot 0.92 loaded
```

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



> Q22. What two variables are anti-correlated

Fruit and bar

> Q23. Similarly, what two variables are mostly positively correlated

Chocolate and winpercent

#Principal Coordinate Analysis

The base R function for PCE is called `prcomp()` and we can set “scale =TRUE/FALSE”

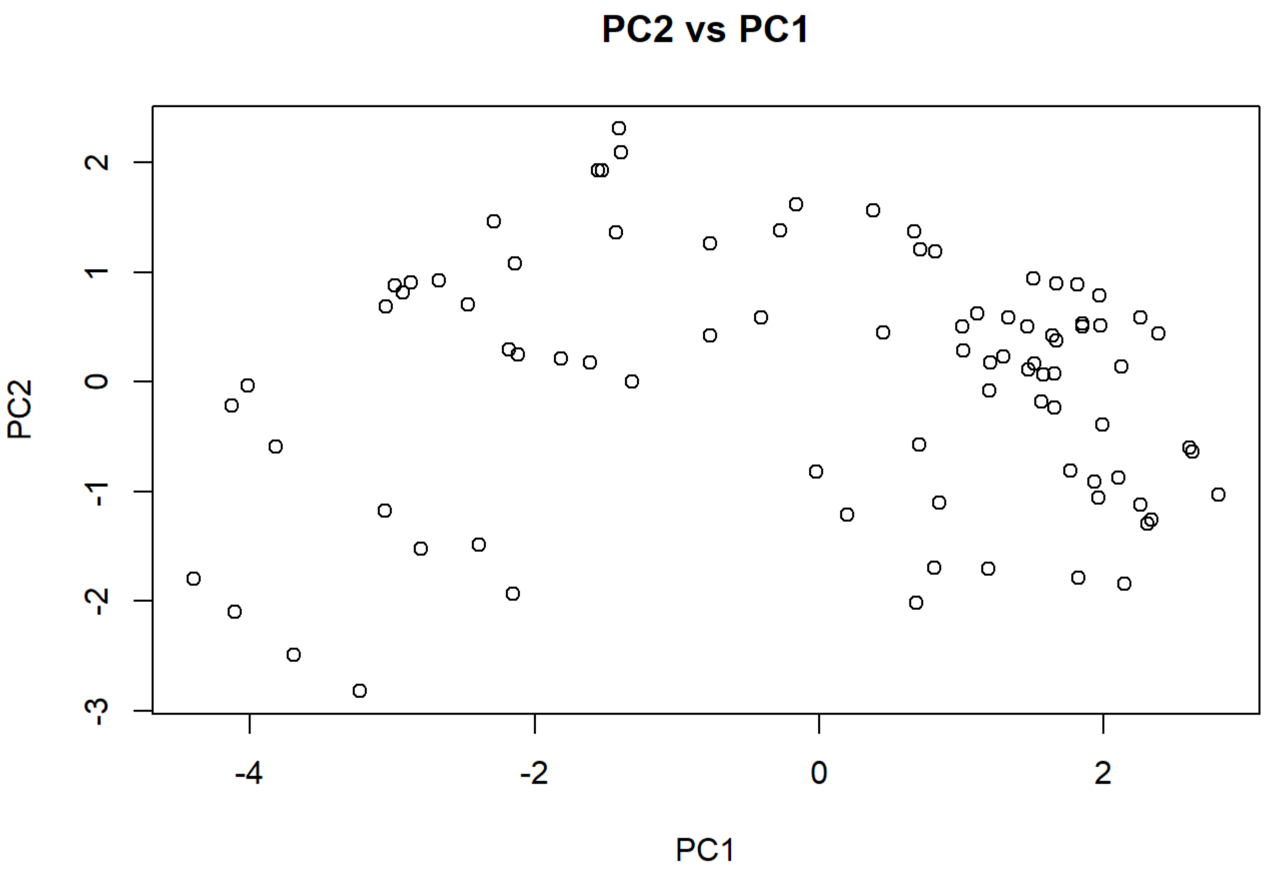
```
pca <-  prcomp(candy, scale = TRUE)
summary(pca)
```

```
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
```

The main result of PCA - i.e. the new PC plot (projection fo candy on our new PC axis) is contained in `pc$x`

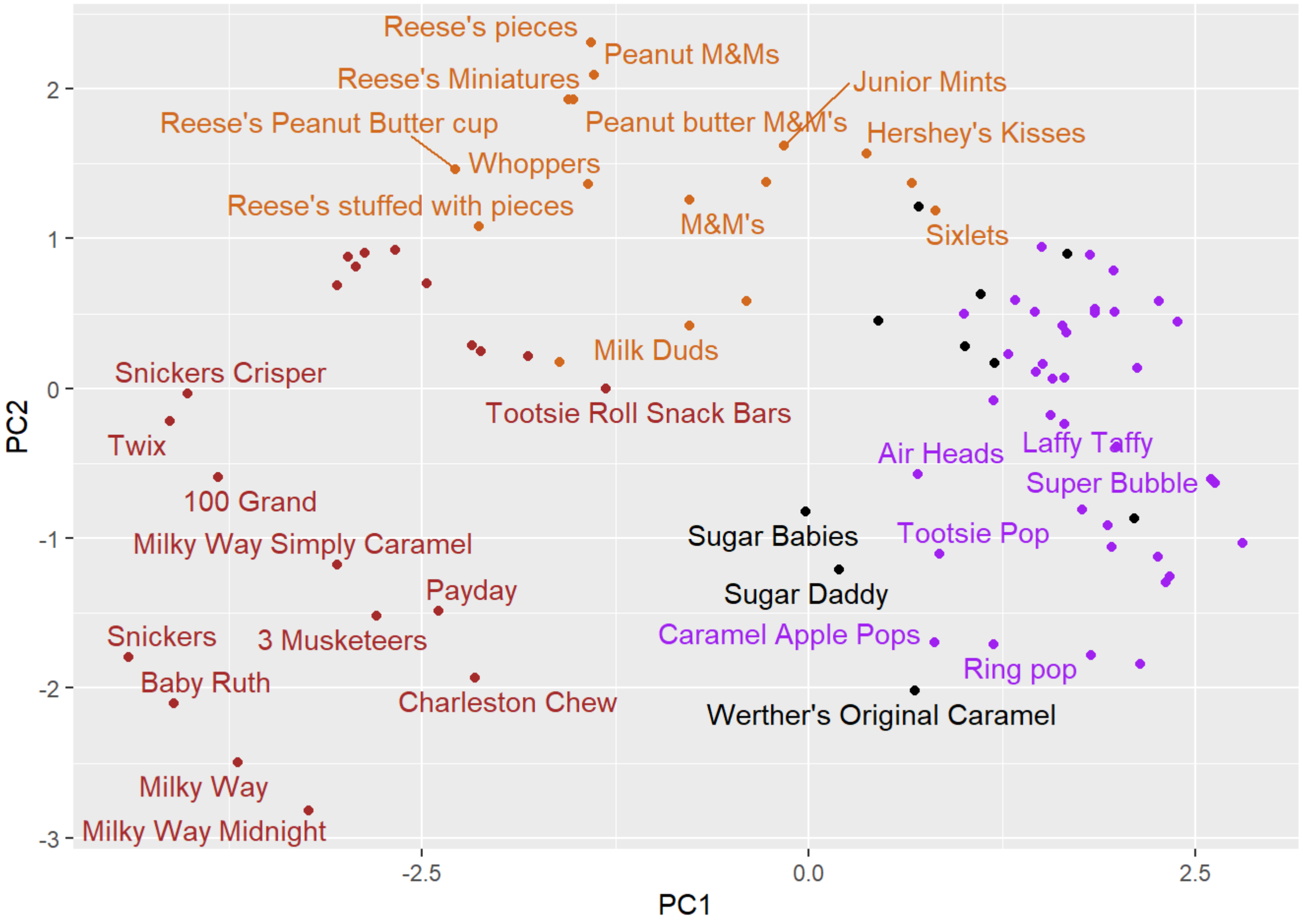
```
plot(pca$x[ , 1:2], main = 'PC2 vs PC1')
```



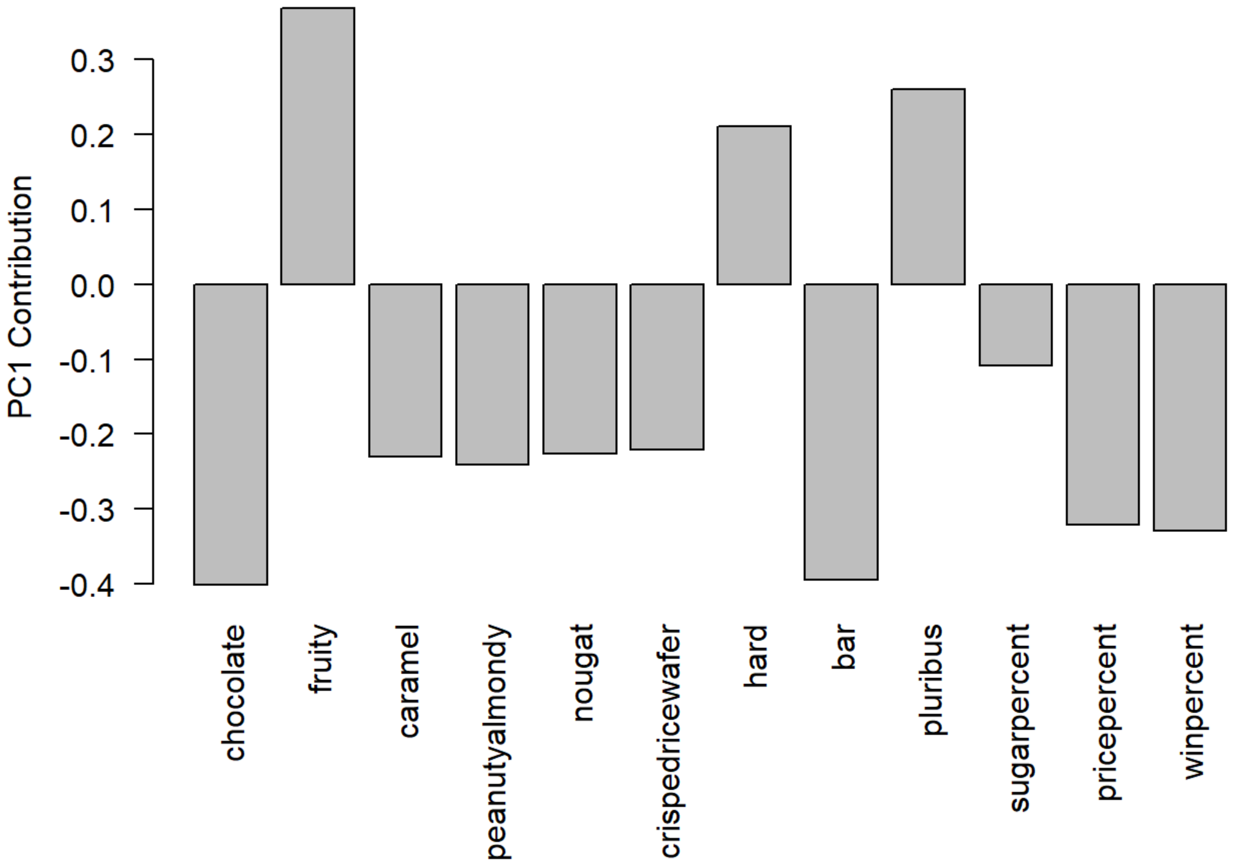
```
#ggplot always wants dataframes
pc <- as.data.frame(pca$x)

ggplot(pc) +
  aes(PC1,PC2, label = rownames(pc)) +
  geom_point(col = my_cols)+
  geom_text_repel(col = my_cols, max.overlaps = 5)
```

```
Warning: ggrepel: 52 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



```
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 they make sense to you?

The PC1 picks up the hard, pluribus, and fruity variables strongest and it does make sense to me.