Class 11 - Mini Halloween Project

AUTHOR
Charlie Rezanka (A15837296)

In today's class, we'll use some sample data about halloween candy preferances to get a greater feeling for how PCA and other methods work.

```
candy <- read.csv("candy-data.txt", 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
```

#the winpercent catagory is how frequently people indicate a preference for a candy #pricepercent analyzes the cost of a candy relative to others

> Q1. How many candy types are in this data?

```
nrow(candy)
```

```
[1] 85
```

> Q2. How many different types fof fruity candy are in this data?

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

```

 0  1 
47 38 
```

What are these fruity candies in question?

we can use ==

```
row.names(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 in these matchups?

```
candy["Twix", "winpercent"]
```

```
[1] 81.64291
```

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

```
candy["Milky Way", "winpercent"]
```

```
[1] 73.09956
```

> 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", repos = "http://cran.us.r-project.org")
```

```
Installing package into 'C:/Users/charl/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)

package 'skimr' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
    C:\Users\charl\AppData\Local\Temp\Rtmpi8fU2Z\downloaded_packages
```

```
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	▇▇▇▇▆

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

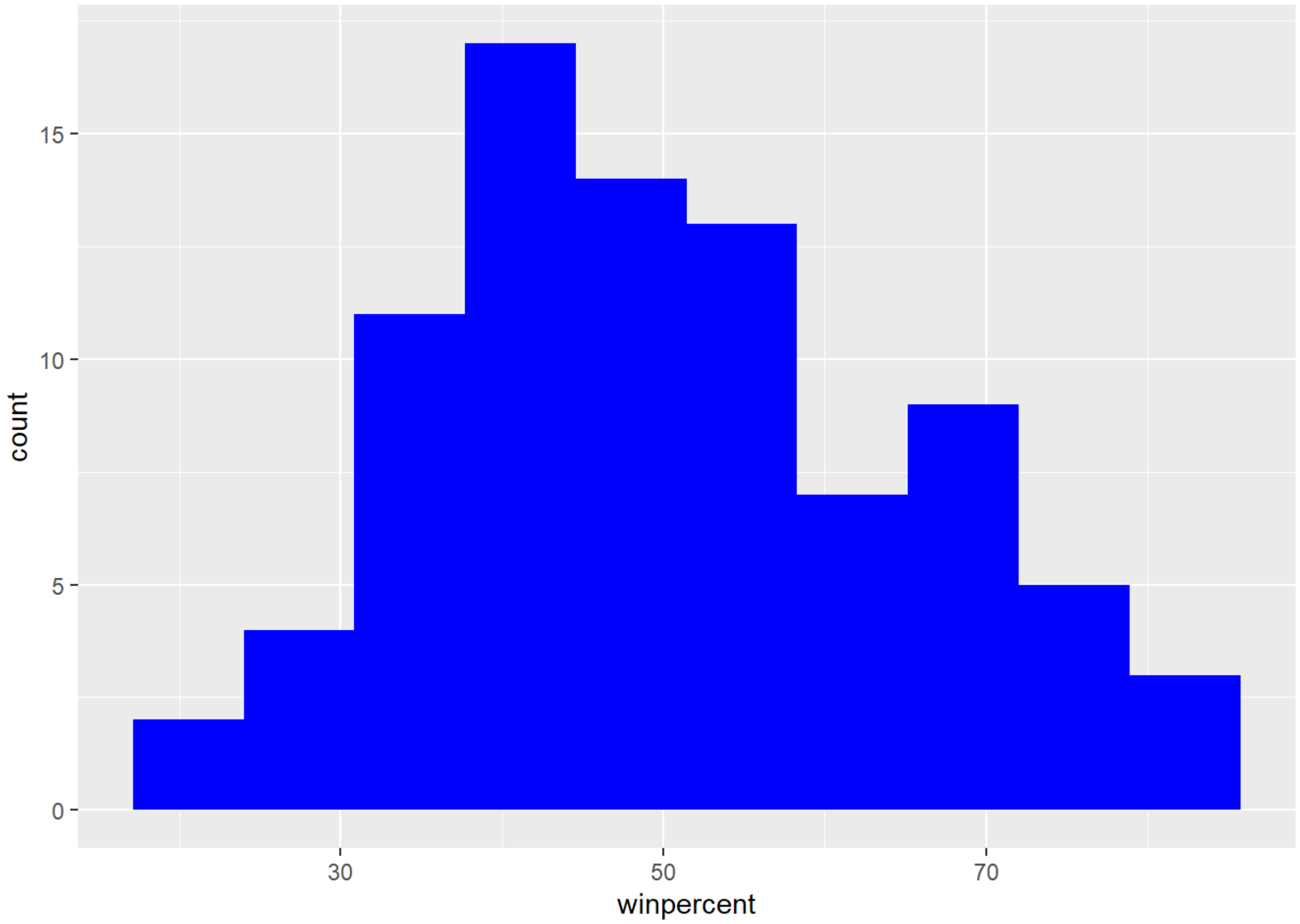
The wineprcent variable is on a completely different scale to every other variable.

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

Whether or not a candy contains chocolate (yes = 1)

> Q8. Plot a histogram of winpercent values

```r
library(ggplot2)
ggplot(candy) +
  aes(winpercent) +
  geom_histogram(bins=10, fill="BLUE")
```



```r
help(ggplot)
```

```
starting httpd help server ... done
```

> Q9. Is the distribution of winpercent values symmetrical?

The distribution of winpercent values is baised towards the 30-50% region, it is not symetrical.

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

```r
median(candy$winpercent)
```

```
[1] 47.82975
```

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

```r
#Filter your data first to be just chocolate
chocolate.candy <- candy[as.logical(candy$chocolate),]

#Get the winpercent values
chocolate.winpercent <- chocolate.candy$winpercent

#calculate the mean
mean(chocolate.winpercent)
```

```
[1] 60.92153
```

```r
sd(chocolate.winpercent)
```

```
[1] 12.81112
```

```r
fruity.candy <- candy[as.logical(candy$fruity),]
fruity.winpercent <- fruity.candy$winpercent
mean(fruity.winpercent)
```

```
[1] 44.11974
```

```r
sd(fruity.winpercent)
```

```
[1] 10.26379
```

Chocolate candy is more popular than fruity candy

> Q12. Is this difference statistically significant?

```
t.test(chocolate.winpercent, fruity.winpercent)
```

```

	Welch Two Sample t-test

data:  chocolate.winpercent and fruity.winpercent
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 
```

They are not significantly different

OVerall Candy Ratings

There is a base R function called `sort()` for sorting vectors of input.

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

#sort(x, decreasing = FALSE) by default (orders by increasing)
sort (x)
```

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

The buddy function to `sort()` that is often even more useful is called `order()`. It returns the “indices” of the input that would result in it being sorted (the numbered vector items placed in decreasing order)

```
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?

I can order by winpercent

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

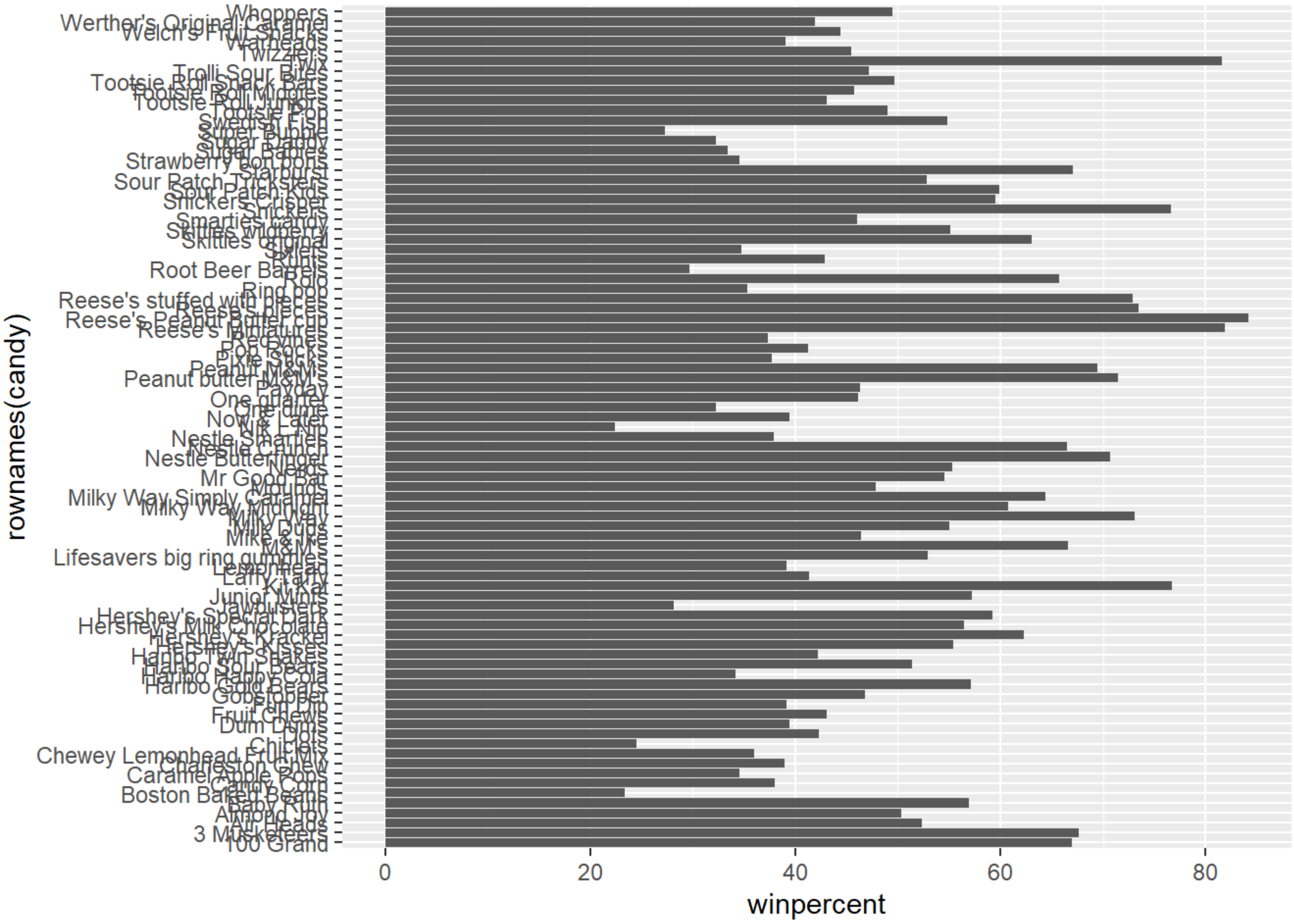
> Q14. What are the top 5 all time favorite candy types out of this set?

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

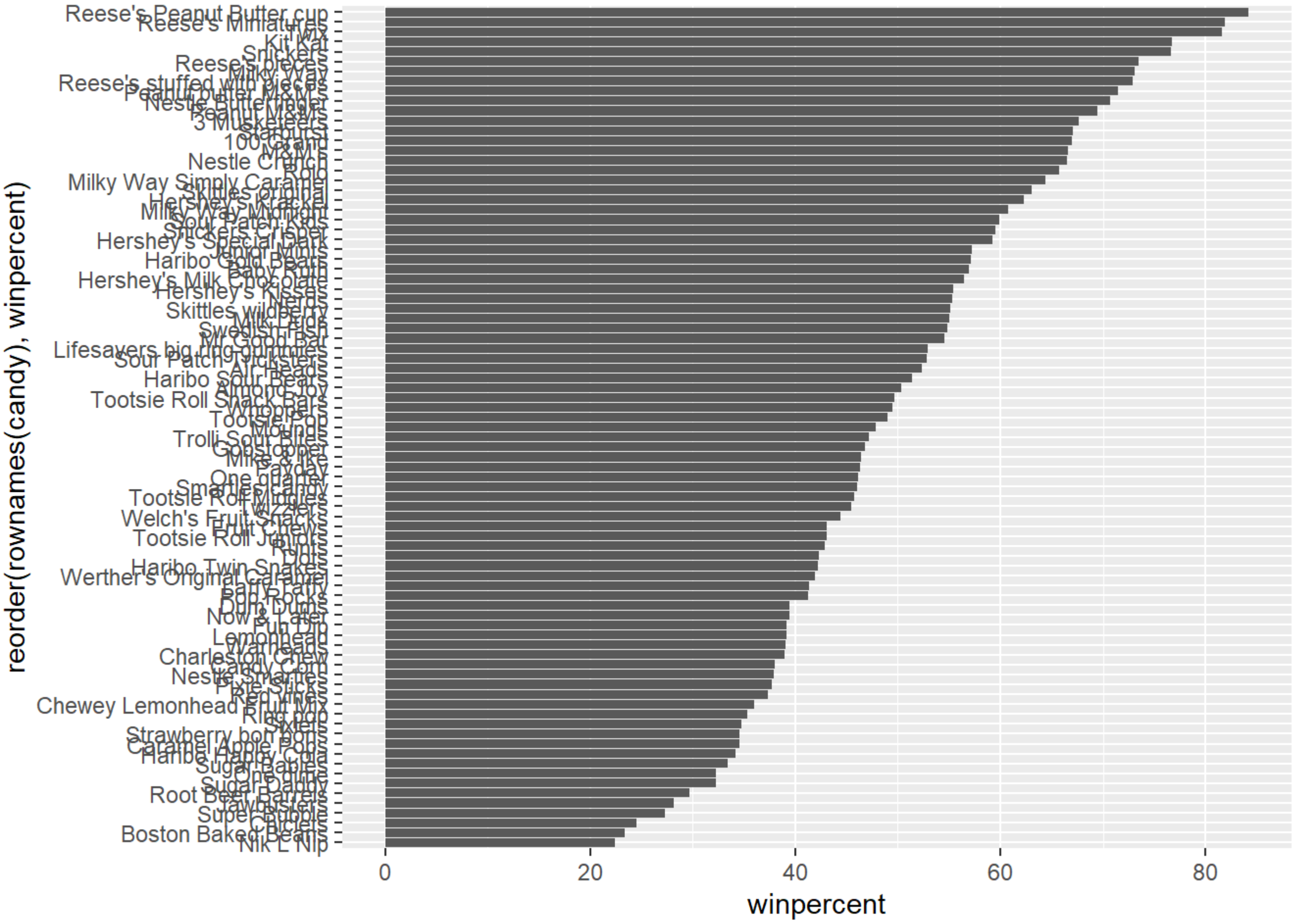
> Q15. Make a first barplot of candy ranking based on winpercent values.

```r
library(ggplot2)

ggplot(candy) +
  aes(winpercent, rownames(candy)) +
  geom_col()
```



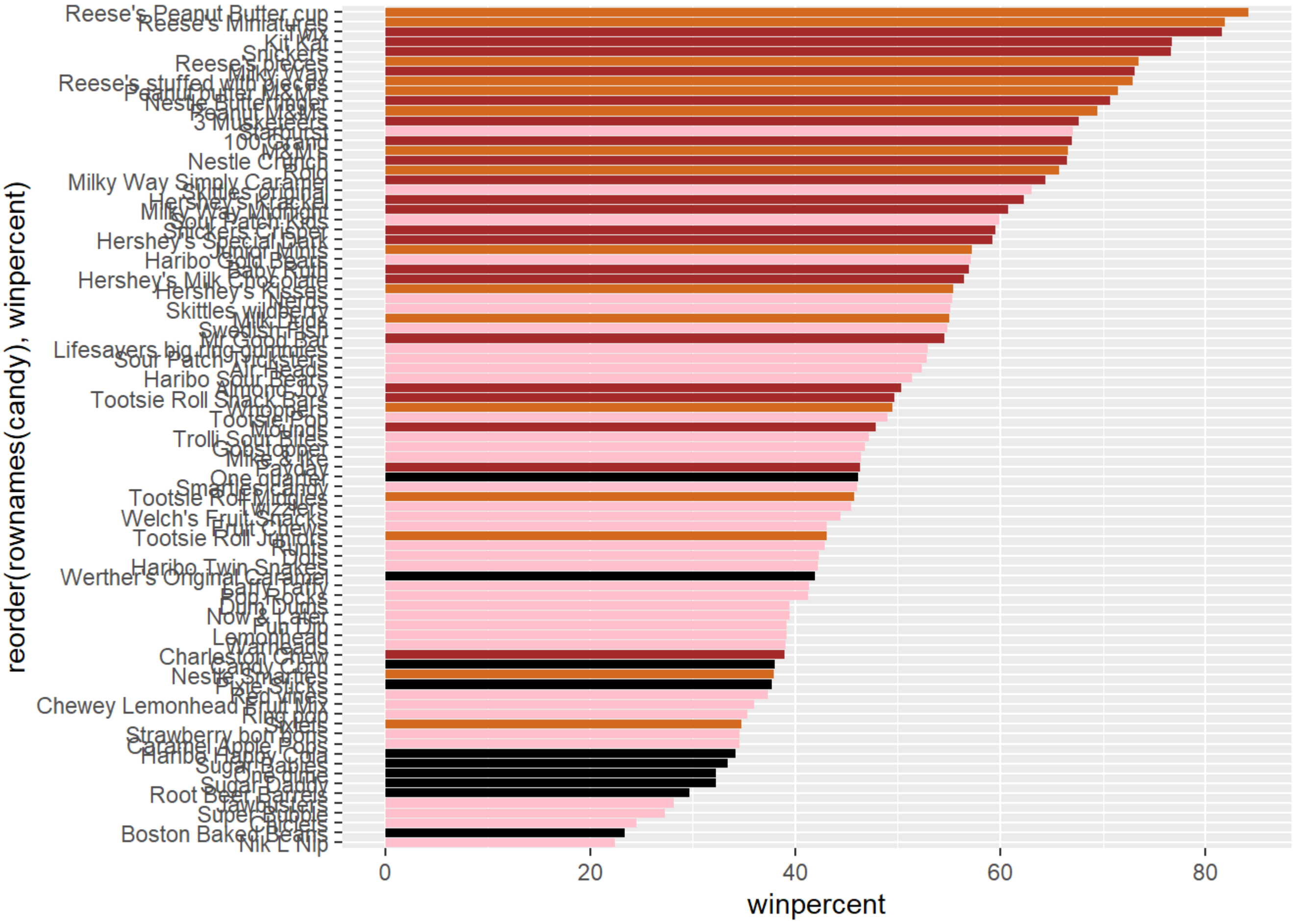
> Q16. This is quite ugly, use the reorder() function to get the bars sorted by winpercent?

```r
ggplot(candy) +
  aes(winpercent, reorder(rownames(candy), winpercent)) +
  geom_col()
```



```
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)
```



> Q17. What is the worst ranked chocolate candy?

Sixlets

> Q18. What is the best ranked fruity candy?

Starbursts

Taking a look at pricepercent

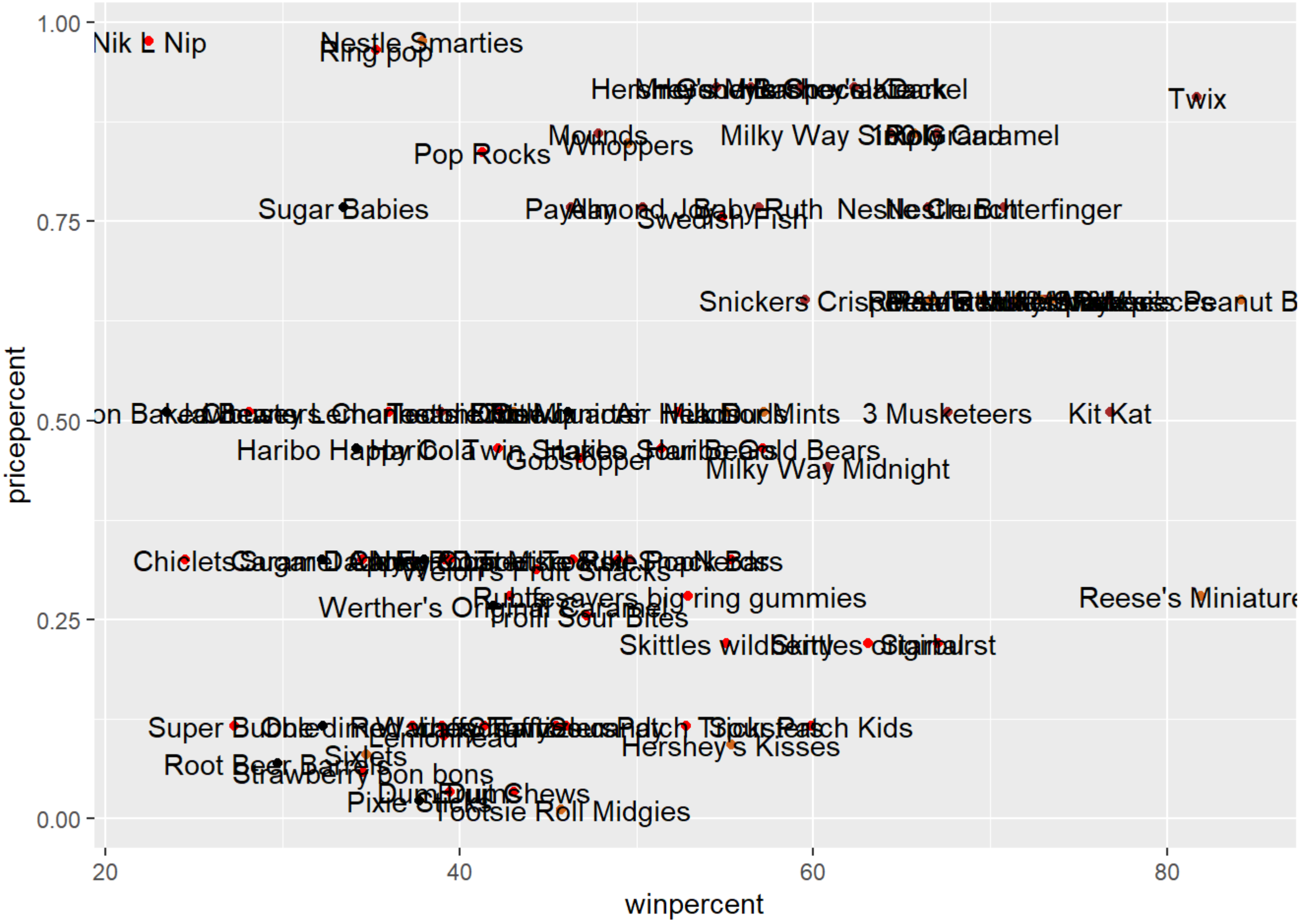
> Q. What's the best candy for the least amout of money?

```r
install.packages("ggrepel", repos = "http://cran.us.r-project.org")
```

```
Installing package into 'C:/Users/charl/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)

package 'ggrepel' successfully unpacked and MD5 sums checked

Warning: cannot remove prior installation of package 'ggrepel'
```

```
Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
C:\Users\charl\AppData\Local\R\win-library\4.2\00LOCK\ggrepel\libs\x64\ggrepel.dll
to C:\Users\charl\AppData\Local\R\win-library\4.2\ggrepel\libs\x64\ggrepel.dll:
Permission denied

Warning: restored 'ggrepel'


The downloaded binary packages are in
    C:\Users\charl\AppData\Local\Temp\Rtmpi8fU2Z\downloaded_packages
```

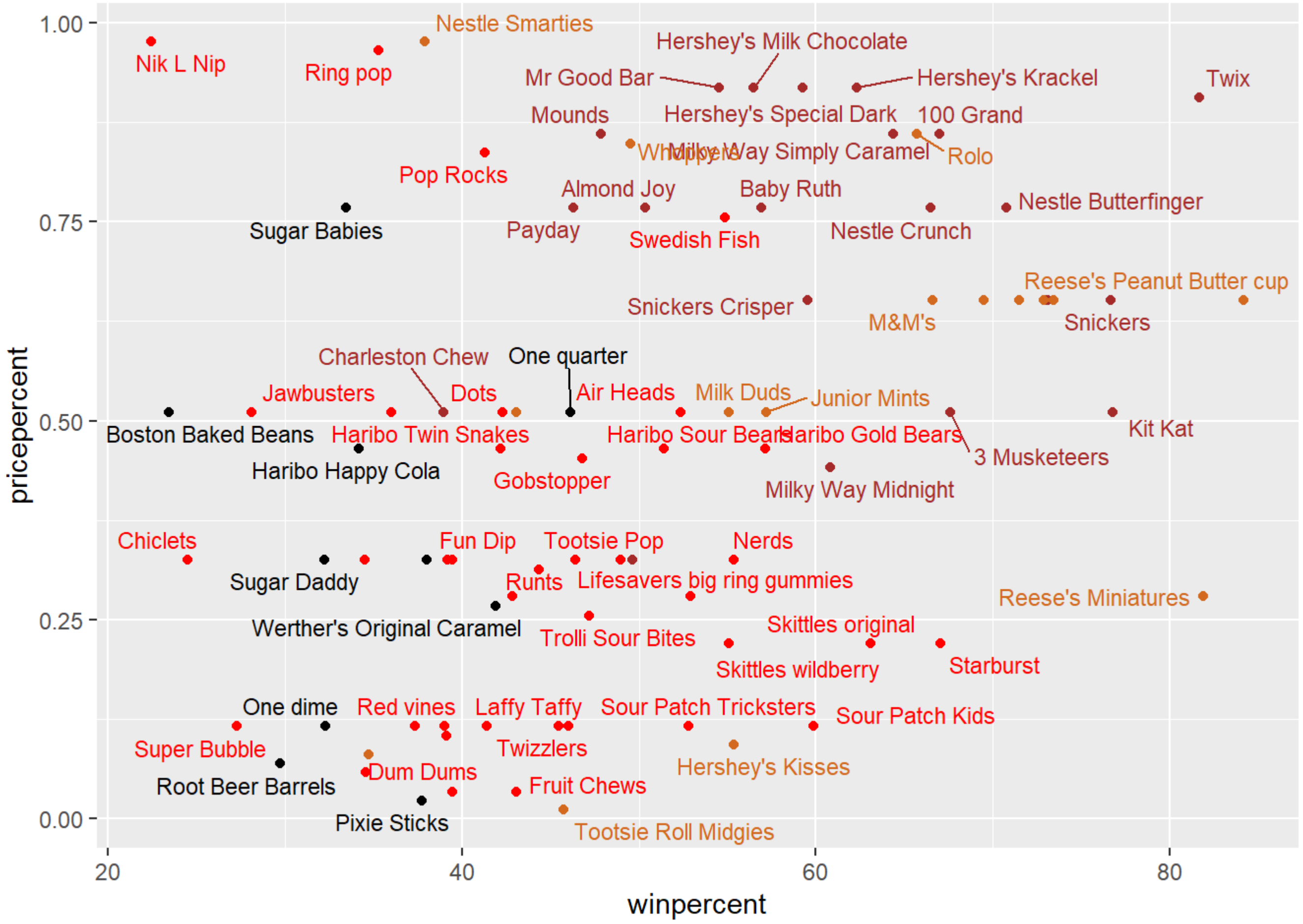
```
library(ggrepel)
my_cols[as.logical(candy$fruity)] = "red"

ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
   geom_text()
```



To negate overlapping labels, I can use the ggrepel package to redesign my labels

```
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
   geom_text_repel(max.overlaps=8, col=my_cols, size=3)
```

```
Warning: ggrepel: 18 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?

Recee’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 (least popular), Ring pop, Nestle smarties, Mr good bar, whoppers

Exploring the correlation structure

pearson correlation goes between -1 and +1 with 0 indicating no correlation, and values close to one being very highly (ani) correlated.

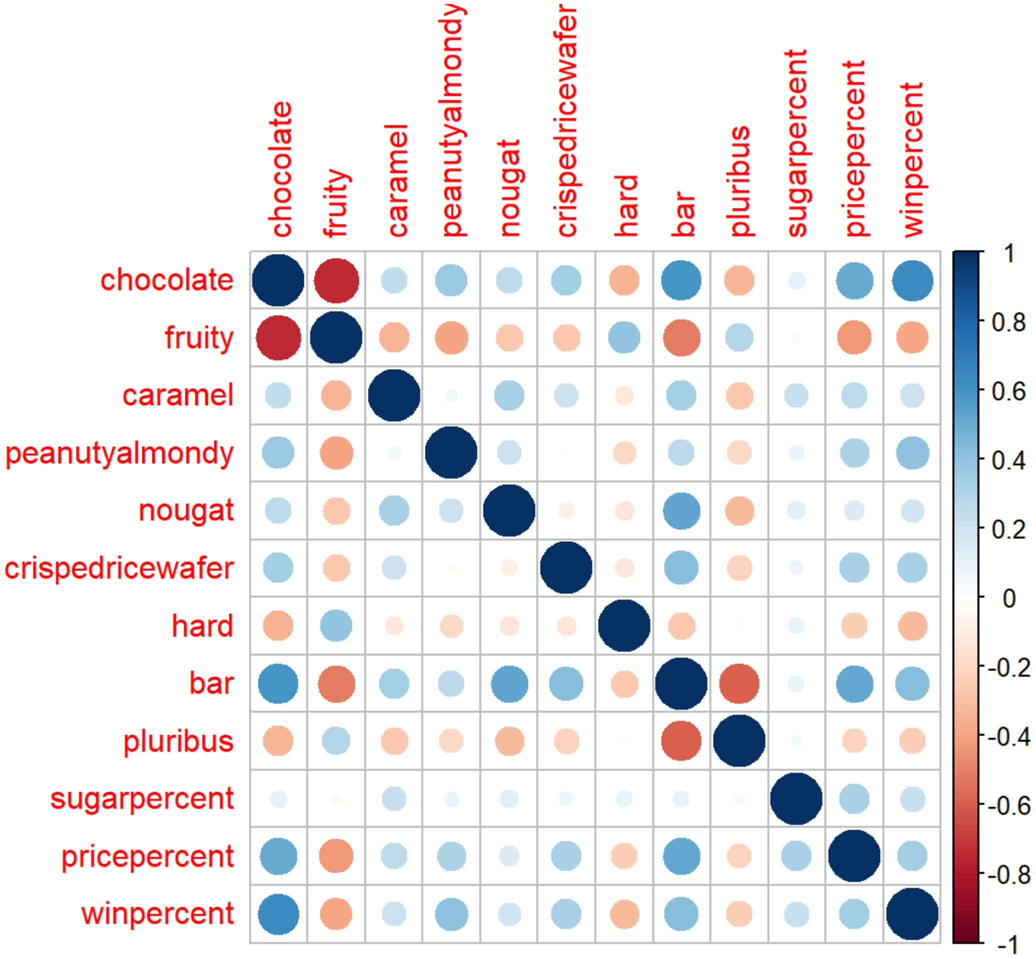
```r
install.packages("corrplot", repos = "http://cran.us.r-project.org")
```

```
Installing package into 'C:/Users/charl/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)

package 'corrplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
    C:\Users\charl\AppData\Local\Temp\Rtmpi8fU2Z\downloaded_packages
```

```r
library(corrplot)
```

```
corrplot 0.92 loaded
```

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



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

Chocolate and fruity candies

> Q23. Similarly, what two variables are most positively correlated?

Chocolate & bar with pricepercent and winpercent

Principal Component Analysis

The base R function for PCA is `prcomp()` and we can set "scale=TRUE/FALSE".

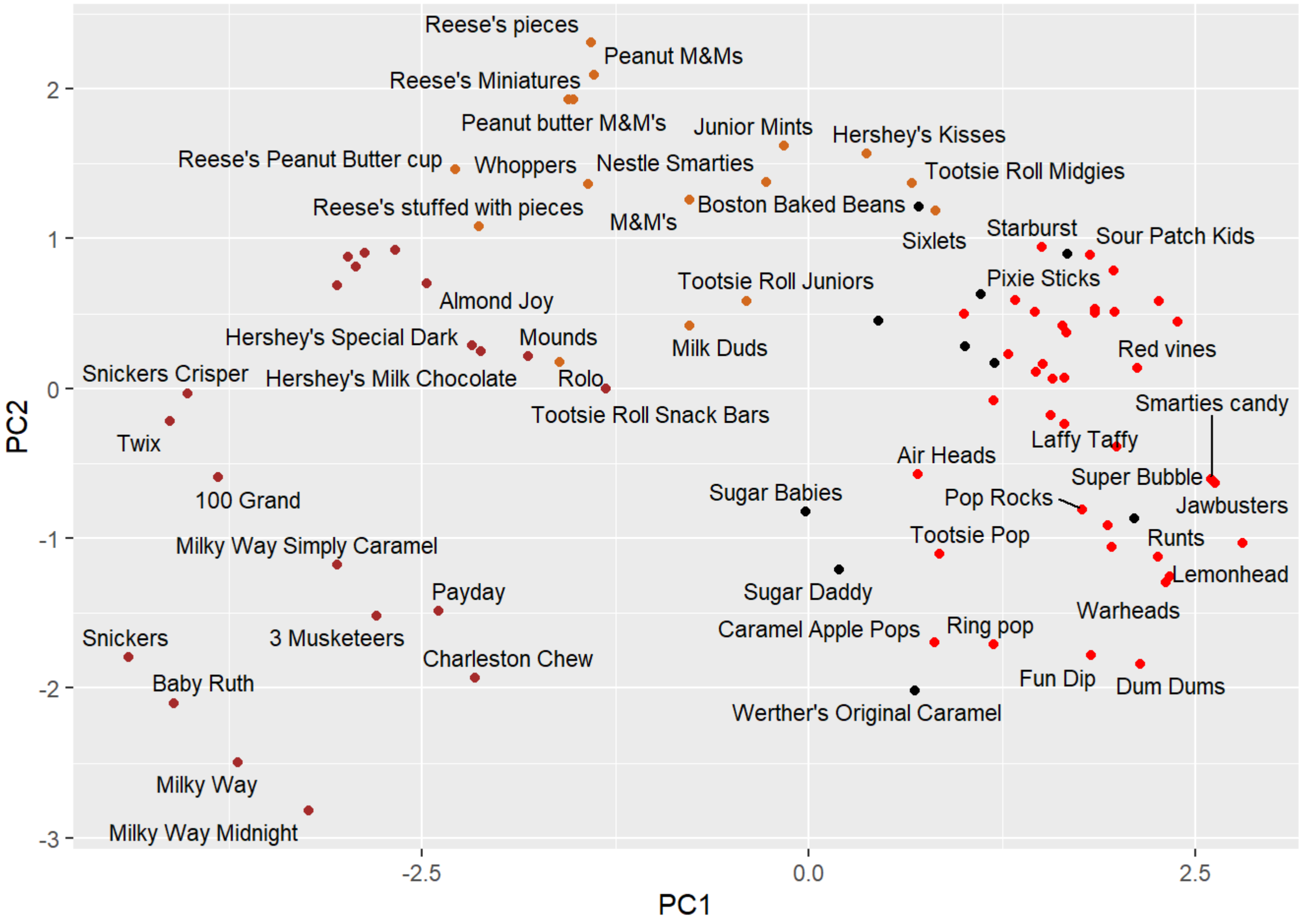
```
pca <- prcomp(candy, scale=TRUE)
summary(pca)
```

```
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) is contained in `pca$x()`

```
pc <- as.data.frame(pca$x)

ggplot(pc) +
  aes(PC1, PC2, label=rownames(pc)) +
  geom_point(col=my_cols) +
  geom_text_repel(max.overlaps = 7, size=3)
```

```
Warning: ggrepel: 31 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



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

Fruity, hard, and pluribus were all picked up with a positive correlation in the PC analysis. This makes sense as all fruity candies were very closely grouped on our PCA plot, and many of the popular fruity candies I can think of come in packages of many small candies (ex. sour patch kids & starbursts)