Class 9: Halloween Candy Mini Project

Carly Chang (A16843962)

Table of contents

Today we will examine data from 538 on common Halloween candy. In particular, we will use ggplot, dplyr, and PCA to make sense of this multivariate dataset.

Importing candy data

```
candy_file <- "https://raw.githubusercontent.com/fivethirtyeight/data/master/candy-power-ranl

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
```

How many chocolate candy are there in the datset?

```
sum(candy$chocolate)
```

```
[1] 37
```

What is your favorate candy?

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

Percentage of the time you will choose that particular candy over the other option.

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

```
[1] 81.64291
```

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

```
[1] 81.64291
```

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
```

Quick overview of 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	

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

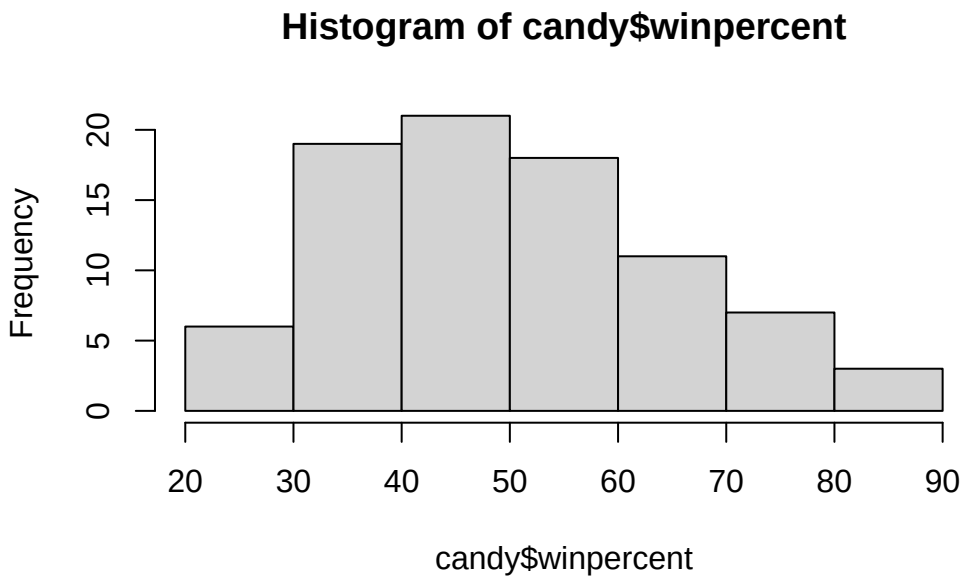
`winpercent` is on a different scale than the others because all of the others range from 0 to 1, but winpercent ranges from 0-100%. I will need to scale this dataset before analysis like PCA.

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

Both n_missing and n_complete shows the number of values that are NA or NULL

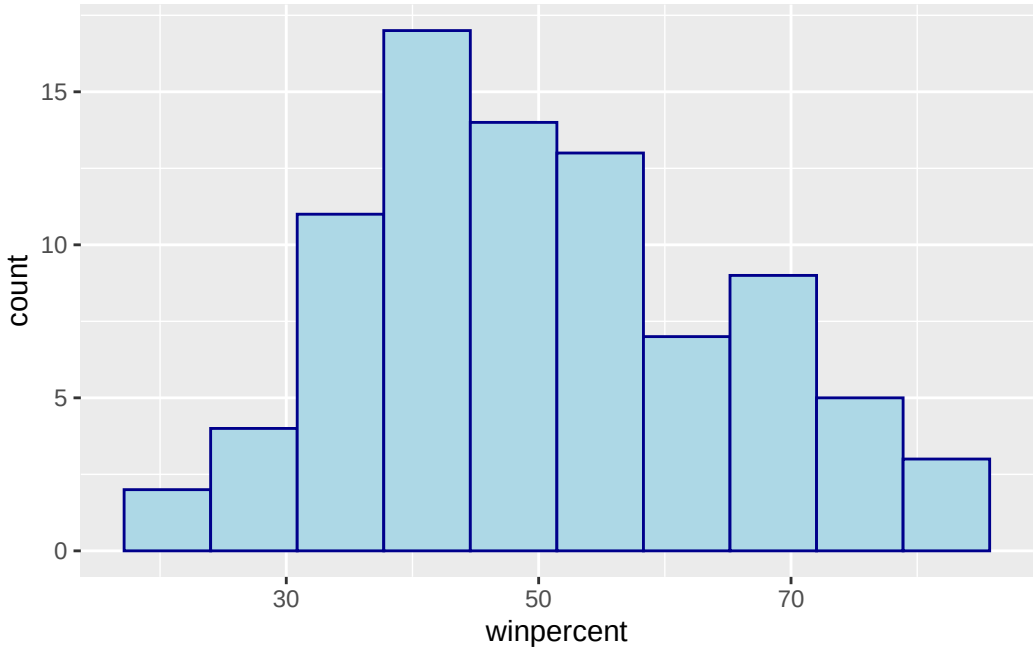
Q8. Plot a histogram of winpercent values

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




```
library(ggplot2)
```

```
ggplot(candy) +
  aes(x=winpercent) +
  geom_histogram(bins=10, fill="lightblue", col="darkblue")
```



Q9. Is the distribution of winpercent values symmetrical?

No, the distribution is not symmetrical, as seen on the histogram.

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

```
summary(candy$winpercent)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  22.45   39.14   47.83   50.32   59.86   84.18 
```

The center is slightly below 50%, with median = 47.83%.

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

- step 1: find all "chocolate" candy
- step 2: find their "winpercent" values

- step 3: summarize these values (mean/median)
- step 4: find all "fruity" candy
- step 5: find their winpercent values
- step 6: summarize these values
- step 7: compare the two summary values

```
mean(candy$winpercent[as.logical(candy$chocolate)]) #gives mean of winpercents for all values
```

```
[1] 60.92153
```

```
mean(candy$winpercent[as.logical(candy$fruity)])
```

```
[1] 44.11974
```

1. Find all chocolate candy

```
choc.inds <- candy$chocolate==1 #returns T/F for every candy
```

2. Find their winpercent values

```
choc.win <- candy[choc.inds,]$winpercent #gives the winpercent value for every column that re
```

3. Summarize these values

```
mean(choc.win)
```

```
[1] 60.92153
```

Repeat for fruit candy

```
fruit.inds <- candy$fruity==1
fruit.win <- candy[fruit.inds,]$winpercent
mean(fruit.win)
```

```
[1] 44.11974
```

On average, chocolate candy is ranked higher than fruit candy.

Q12. Is this difference statistically significant?

```
t.test(choc.win, fruit.win) #t-test of the winpercent columns
```

```
    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
```

The difference is significantly different, with a p-value of 2.87e-8 < 0.05 from the t-test.

Overall Candy Rankings

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

```
x <- c(10,1,100)
order(x) #returns the element position (index) of the vector in order, not the actual element
```

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

```
x[order(x)] #returns the values in order
```

```
[1]   1  10 100
```

The `order` function tells us how to arrange the elements of the input to make them sorted - ie. how to order them.

We can determine the order of winpercent to make them sorted and use that order to arrange the whole dataset.

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

```
                  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
Root Beer Barrels         0      0       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
Root Beer Barrels                0    1   0        1        0.732        0.069
                  winpercent
Nik L Nip           22.44534
Boston Baked Beans  23.41782
Chiclets            24.52499
Super Bubble        27.30386
Jawbusters          28.12744
Root Beer Barrels   29.70369
```

Give only the first 5 of the ordered list:

```
head(candy[order(candy$winpercent),], 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
```

Using dplyr:

```
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
candy %>%
  arrange(winpercent) %>%
  head(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?

```
head(candy[order(candy$winpercent),], n=5, decreasing=TRUE)
```

```
                  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
```

Using dplyr:

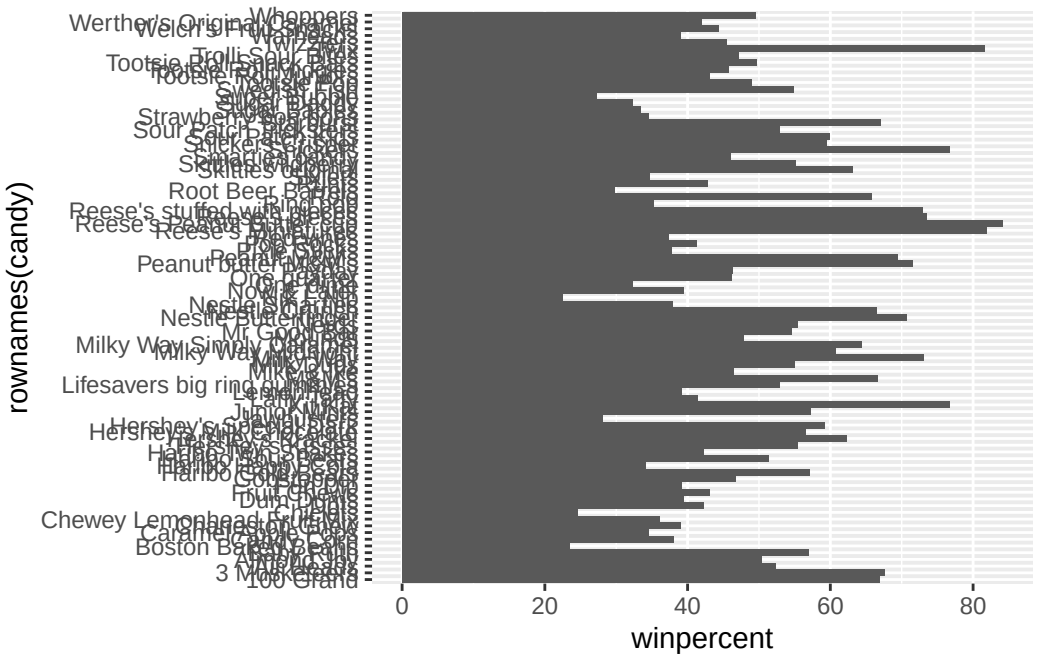
```
candy %>%
  arrange(desc(winpercent)) %>%
  head(5)
```

```
                        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.

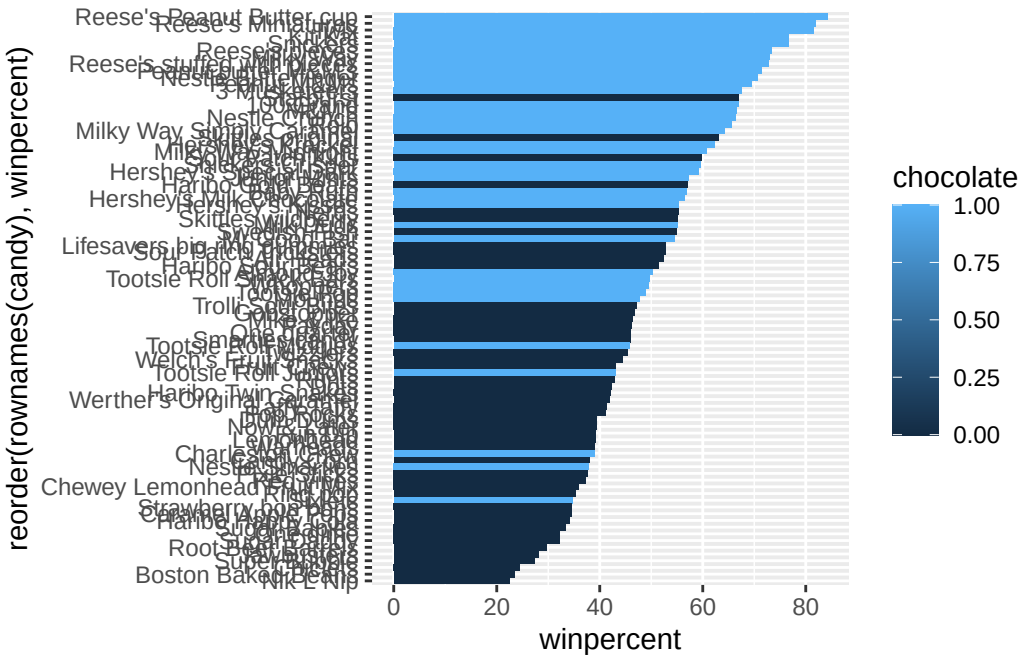
```
ggplot(candy) +
  aes(x=winpercent,y=rownames(candy)) +
  geom_col()
```



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

Sort by winpercent and add color:

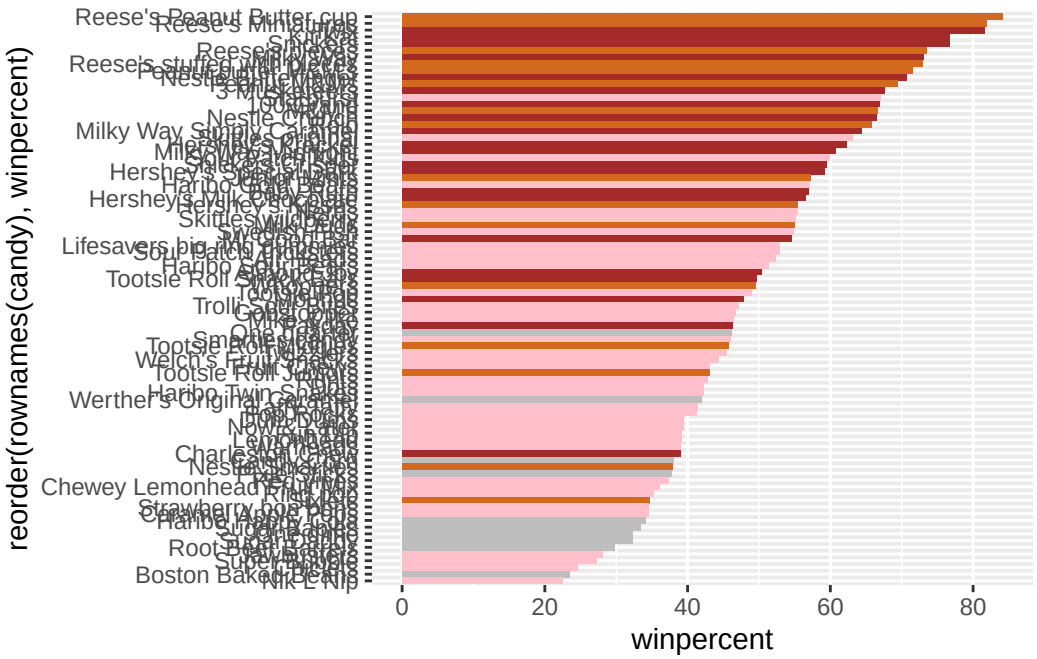
```
ggplot(candy) +
  aes(x=winpercent,reorder(rownames(candy),winpercent), fill=chocolate) +
  geom_col()
```



We need to make our own separate color vector where we can spell out what candy is colored a particular color.

```
mycols <- rep("gray",nrow(candy)) #rep repeats a given input for a given number of times
mycols[candy$chocolate==1] <- "chocolate" #overrides with red for every element that is choco
mycols[candy$bar==1] <- "brown"
mycols[candy$fruity==1] <- "pink"
```

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



Q17. What is the worst ranked chocolate candy?

Sixlets

Q18. What is the best ranked fruity candy?

Starbusts

Taking a look at pricepercent

Make a plot of winpercent (x-axis) vs pricepercent (y-axis)

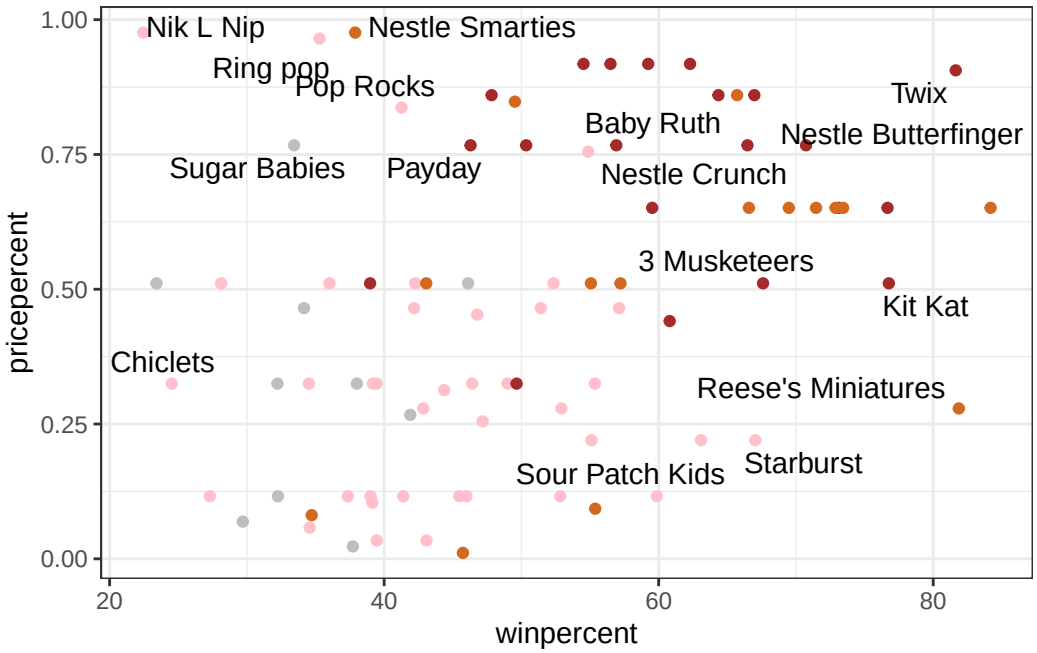
```
ggplot(candy) +
  aes(winpercent,pricepercent, label=rownames(candy)) +
  geom_point(col=mycols) +
  geom_text()
```

To avoid te overplotting of the text labels, we can use the add on package **ggrepel**. geom_text_repel prevents overlap of text labels (which can be changed with the max.overlaps parameter)

```
library(ggrepel)

ggplot(candy) +
  aes(winpercent,pricepercent, label=rownames(candy)) +
  geom_point(col=mycols) +
  geom_text_repel(max.overlaps=6) +
  theme_bw()
```

```
Warning: ggrepel: 69 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?

Reese's minatures - most lower right quartile.

```
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

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

The 5 most expensive are: Nik L Nip, Ring pop, Nestle Smarties, Mr Good bar, Hershey's Krackel. Nik L Nip is the least popular.

Exploring the correlation structure

Now that we have explored the dataset a little, we will see how the variables interact with one another.

First we will use correlation and view the results with the **corrplot** package to plot a correlation matrix.

```
library(corrplot)
```

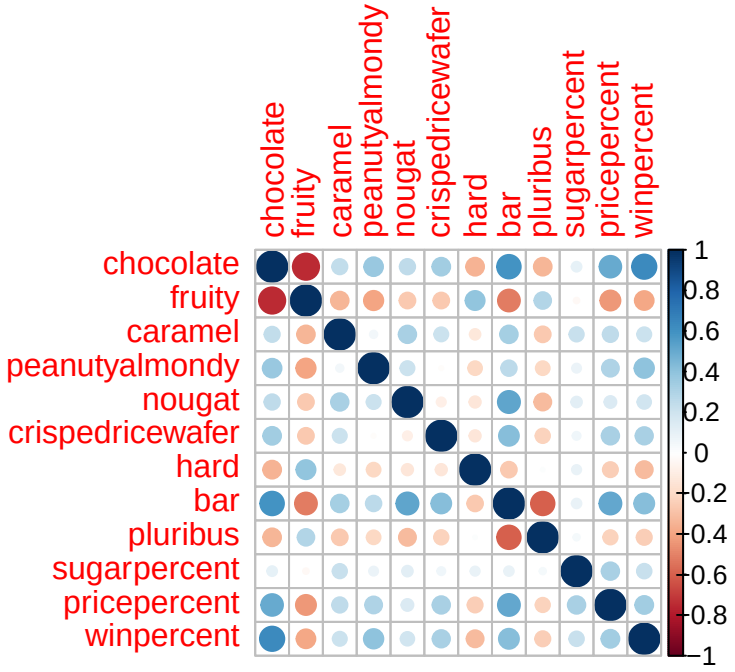
```
corrplot 0.95 loaded
```

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

```
                  chocolate      fruity     caramel peanutyalmondy      nougat
chocolate         1.0000000 -0.74172106  0.24987535     0.37782357  0.25489183
fruity           -0.7417211  1.00000000 -0.33548538    -0.39928014 -0.26936712
caramel           0.2498753 -0.33548538  1.00000000     0.05935614  0.32849280
peanutyalmondy    0.3778236 -0.39928014  0.05935614     1.00000000  0.21311310
nougat            0.2548918 -0.26936712  0.32849280     0.21311310  1.00000000
crispedricewafer  0.3412098 -0.26936712  0.21311310    -0.01764631 -0.08974359
hard             -0.3441769  0.39067750 -0.12235513    -0.20555661 -0.13867505
bar               0.5974211 -0.51506558  0.33396002     0.26041960  0.52297636
pluribus         -0.3396752  0.29972522 -0.26958501    -0.20610932 -0.31033884
sugarpercent      0.1041691 -0.03439296  0.22193335     0.08788927  0.12308135
pricepercent      0.5046754 -0.43096853  0.25432709     0.30915323  0.15319643
winpercent        0.6365167 -0.38093814  0.21341630     0.40619220  0.19937530
                  crispedricewafer        hard         bar    pluribus
chocolate              0.34120978 -0.34417691  0.59742114 -0.33967519
fruity                -0.26936712  0.39067750 -0.51506558  0.29972522
caramel                0.21311310 -0.12235513  0.33396002 -0.26958501
peanutyalmondy        -0.01764631 -0.20555661  0.26041960 -0.20610932
nougat                -0.08974359 -0.13867505  0.52297636 -0.31033884
crispedricewafer       1.00000000 -0.13867505  0.42375093 -0.22469338
hard                  -0.13867505  1.00000000 -0.26516504  0.01453172
bar                    0.42375093 -0.26516504  1.00000000 -0.59340892
pluribus              -0.22469338  0.01453172 -0.59340892  1.00000000
sugarpercent           0.06994969  0.09180975  0.09998516  0.04552282
pricepercent           0.32826539 -0.24436534  0.51840654 -0.22079363
winpercent             0.32467965 -0.31038158  0.42992933 -0.24744787
                  sugarpercent pricepercent winpercent
```

```
chocolate          0.10416906    0.5046754  0.6365167
fruity            -0.03439296   -0.4309685 -0.3809381
caramel            0.22193335    0.2543271  0.2134163
peanutyalmondy     0.08788927    0.3091532  0.4061922
nougat             0.12308135    0.1531964  0.1993753
crispedricewafer   0.06994969    0.3282654  0.3246797
hard               0.09180975   -0.2443653 -0.3103816
bar                0.09998516    0.5184065  0.4299293
pluribus           0.04552282   -0.2207936 -0.2474479
sugarpercent       1.00000000    0.3297064  0.2291507
pricepercent       0.32970639    1.0000000  0.3453254
winpercent         0.22915066    0.3453254  1.0000000
```

```
corrplot(cij)
```



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

fruity vs chocolate are anti-correlated (red dot).

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

chocolate vs bar are most positively correlated (blue dot).

Principal Component Analysis

```
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
```

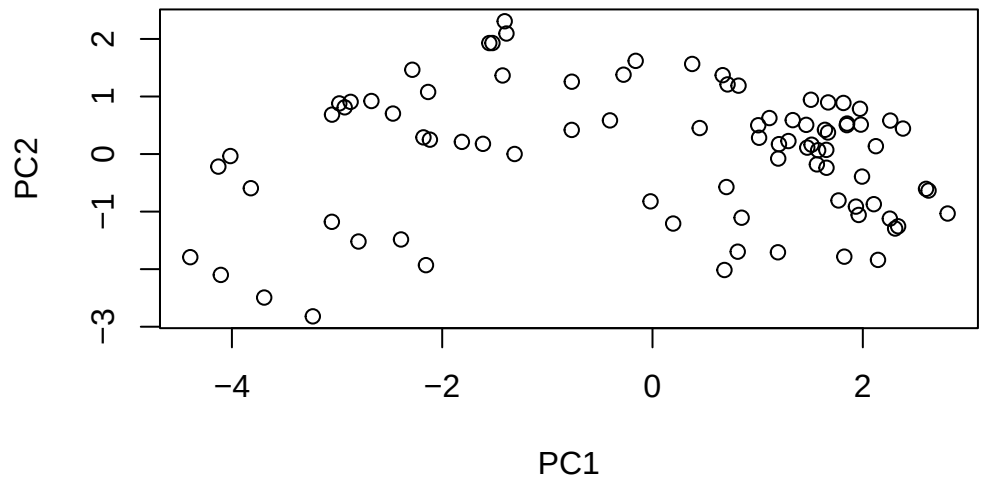
```
attributes(pca)
```

```
$names
[1] "sdev"     "rotation" "center"   "scale"    "x"

$class
[1] "prcomp"
```

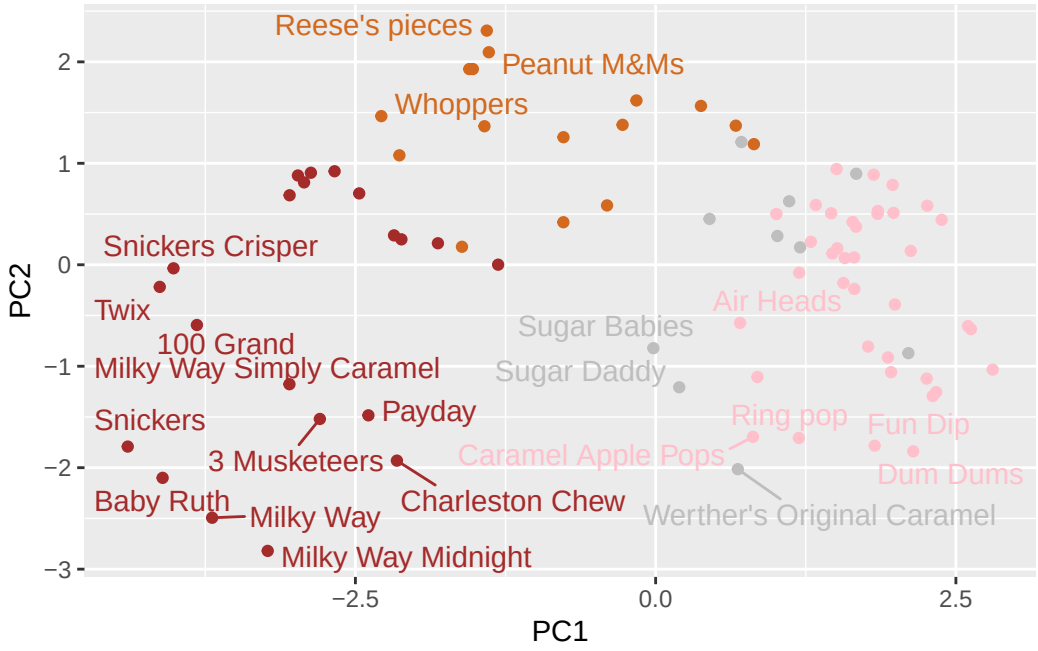
x = scores rotation = loadings (contributions)

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




```
ggplot(pca$x) +
  aes(PC1,PC2,label=rownames(pca$x)) +
  geom_point(col=mycols) +
  geom_text_repel(col=mycols,max.overlaps=7)
```

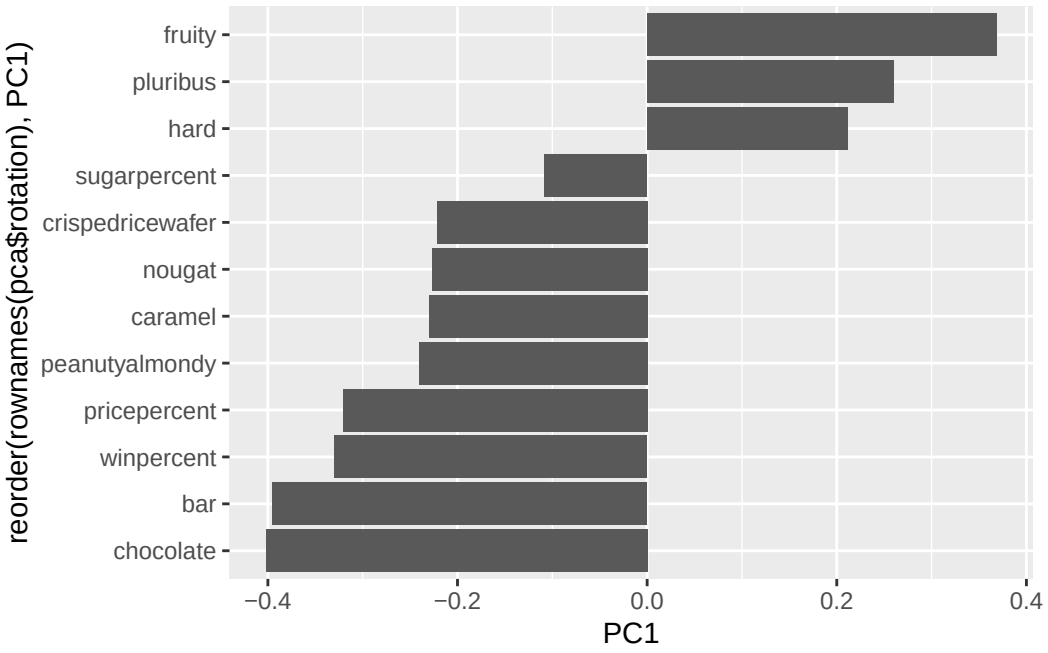
```
Warning: ggrepel: 63 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Finally let's look at how the original variable contribute to the PCs, start with PC1.

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

```
ggplot(pca$rotation) +
  aes(PC1, reorder(rownames(pca$rotation),PC1)) +
  geom_col()
```



fruity has the most positive contribution to PC1. This makes sense because in the score plot, all the fruity candies (in pink) are on the right side. The correlation plot also shows positive correlation between fruity, hard, and pluribus, which are the top 3 contributions to PC1.