Class 10: Halloween Mini-Project

Isabel Hui - A16887852

Today is Halloween, an old Irish holiday. Let's celebrate by eating candy.

We will explore some data all about Halloween candy from the 538 website.

```
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
```

```
rownames(candy)
```

```
 [1] "100 Grand"                   "3 Musketeers"
 [3] "One dime"                    "One quarter"
 [5] "Air Heads"                   "Almond Joy"
 [7] "Baby Ruth"                   "Boston Baked Beans"
 [9] "Candy Corn"                  "Caramel Apple Pops"
[11] "Charleston Chew"             "Chewey Lemonhead Fruit Mix"
[13] "Chiclets"                    "Dots"
[15] "Dum Dums"                    "Fruit Chews"
[17] "Fun Dip"                     "Gobstopper"
[19] "Haribo Gold Bears"           "Haribo Happy Cola"
[21] "Haribo Sour Bears"           "Haribo Twin Snakes"
[23] "Hershey's Kisses"            "Hershey's Krackel"
[25] "Hershey's Milk Chocolate"    "Hershey's Special Dark"
[27] "Jawbusters"                  "Junior Mints"
[29] "Kit Kat"                     "Laffy Taffy"
[31] "Lemonhead"                   "Lifesavers big ring gummies"
[33] "Peanut butter M&M's"         "M&M's"
[35] "Mike & Ike"                  "Milk Duds"
[37] "Milky Way"                   "Milky Way Midnight"
[39] "Milky Way Simply Caramel"    "Mounds"
[41] "Mr Good Bar"                 "Nerds"
[43] "Nestle Butterfinger"         "Nestle Crunch"
[45] "Nik L Nip"                   "Now & Later"
[47] "Payday"                      "Peanut M&Ms"
[49] "Pixie Sticks"                "Pop Rocks"
[51] "Red vines"                   "Reese's Miniatures"
[53] "Reese's Peanut Butter cup"   "Reese's pieces"
[55] "Reese's stuffed with pieces" "Ring pop"
[57] "Rolo"                        "Root Beer Barrels"
[59] "Runts"                       "Sixlets"
[61] "Skittles original"           "Skittles wildberry"
[63] "Nestle Smarties"             "Smarties candy"
[65] "Snickers"                    "Snickers Crisper"
[67] "Sour Patch Kids"             "Sour Patch Tricksters"
[69] "Starburst"                   "Strawberry bon bons"
[71] "Sugar Babies"                "Sugar Daddy"
[73] "Super Bubble"                "Swedish Fish"
[75] "Tootsie Pop"                 "Tootsie Roll Juniors"
[77] "Tootsie Roll Midgies"        "Tootsie Roll Snack Bars"
[79] "Trolli Sour Bites"           "Twix"
```

```
[81] "Twizzlers"                   "Warheads"
[83] "Welch's Fruit Snacks"        "Werther's Original Caramel"
[85] "Whoppers"
```

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

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

```
[1] 38
```

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

```
[1] 37
```

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

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

```
[1] 60.8007
```

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
```

```
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 |>
  filter(rownames(candy)=="Haribo Happy Cola") |>
  select(winpercent)
```

```
                 winpercent
Haribo Happy Cola   34.15896
```

Q. Find fruity candy with a winpercent above 50%.

```
candy |>
  filter(winpercent > 50) |>
  filter(fruity==1)
```

```
                           chocolate fruity caramel peanutyalmondy nougat
Air Heads                          0      1       0              0      0
Haribo Gold Bears                  0      1       0              0      0
Haribo Sour Bears                  0      1       0              0      0
Lifesavers big ring gummies        0      1       0              0      0
Nerds                              0      1       0              0      0
Skittles original                  0      1       0              0      0
Skittles wildberry                 0      1       0              0      0
Sour Patch Kids                    0      1       0              0      0
Sour Patch Tricksters              0      1       0              0      0
Starburst                          0      1       0              0      0
Swedish Fish                       0      1       0              0      0
                           crispedricewafer hard bar pluribus sugarpercent
Air Heads                                 0    0   0        0        0.906
Haribo Gold Bears                         0    0   0        1        0.465
Haribo Sour Bears                         0    0   0        1        0.465
Lifesavers big ring gummies               0    0   0        0        0.267
Nerds                                     0    1   0        1        0.848
Skittles original                         0    0   0        1        0.941
Skittles wildberry                        0    0   0        1        0.941
```

```
Sour Patch Kids                         0    0   0        1         0.069
Sour Patch Tricksters                   0    0   0        1         0.069
Starburst                               0    0   0        1         0.151
Swedish Fish                            0    0   0        1         0.604
                           pricepercent winpercent
Air Heads                         0.511   52.34146
Haribo Gold Bears                 0.465   57.11974
Haribo Sour Bears                 0.465   51.41243
Lifesavers big ring gummies       0.279   52.91139
Nerds                             0.325   55.35405
Skittles original                 0.220   63.08514
Skittles wildberry                0.220   55.10370
Sour Patch Kids                   0.116   59.86400
Sour Patch Tricksters             0.116   52.82595
Starburst                         0.220   67.03763
Swedish Fish                      0.755   54.86111
```

To get a quick insight into a new dataset, some people like using the skimer package and its `skim()` function.

```
library(skimr)
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	

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

Looks like the `winpercent` variable or column is measured on a different scale than everything else! I will need to scale my data before doing any analysis like PCA etc.

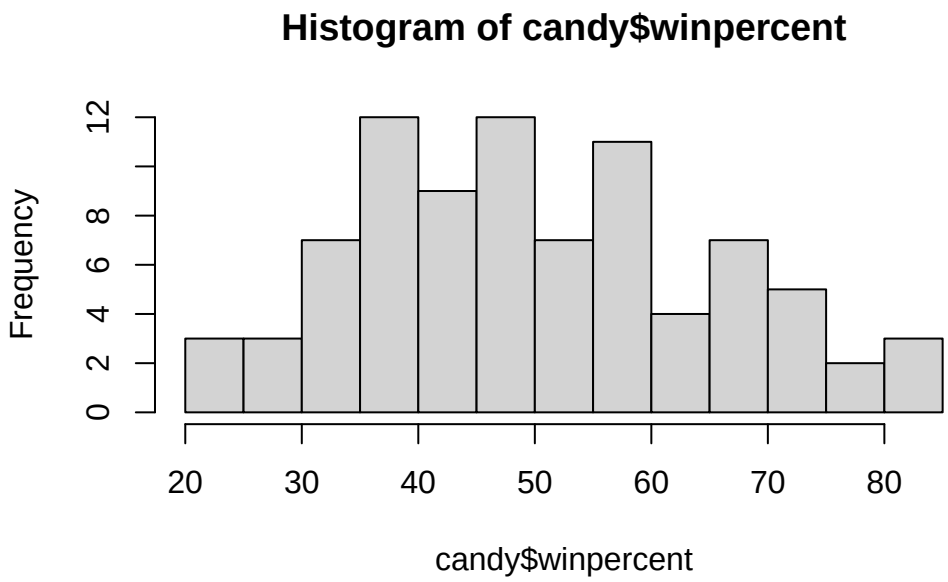
Q7. What do you think a zero and one represent for the candy$chocolate column?

They represent True/False.

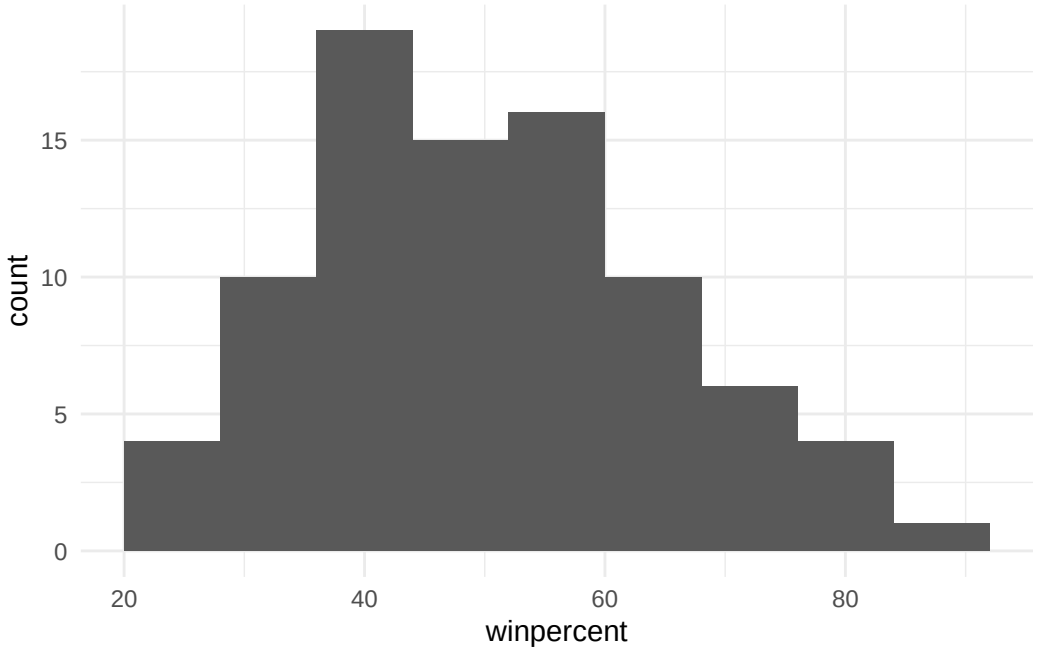
Q8. Plot a histogram of winpercent values.

We can do this a few ways, e.g. the "base" R `hist()` function or with `ggplot()`.

```
hist(candy$winpercent, breaks=10)
```




```
library(ggplot2)

ggplot(candy) +
  aes(winpercent) +
  geom_histogram(binwidth = 8) +
  theme_minimal()
```



Q9. Is the distribution of winpercent values symmetrical?

No.

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 
```

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

```
fruit.candy <- candy |>
  filter(fruity == 1)

summary(fruit.candy$winpercent)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  22.45   39.04   42.97   44.12   52.11   67.04 
```

```
#summary(candy[as.logical(candy$chocolate),]$winpercent)

choc.candy <- candy |>
  filter(chocolate == 1)

summary(choc.candy$winpercent)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  34.72   50.35   60.80   60.92   70.74   84.18
```

Chocolate candy has a higher median and mean compared to fruit candy.

Q12. Is this difference statistically significant?

```
t.test(choc.candy$winpercent, fruit.candy$winpercent)
```

```
    Welch Two Sample t-test

data:  choc.candy$winpercent and fruit.candy$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
```

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

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

```
play <- c("d", "a", "c")
sort(play)
```

```
[1] "a" "c" "d"
```

```
order(play)
```

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

```
sort(c(5,2,10), decreasing = T)
```

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

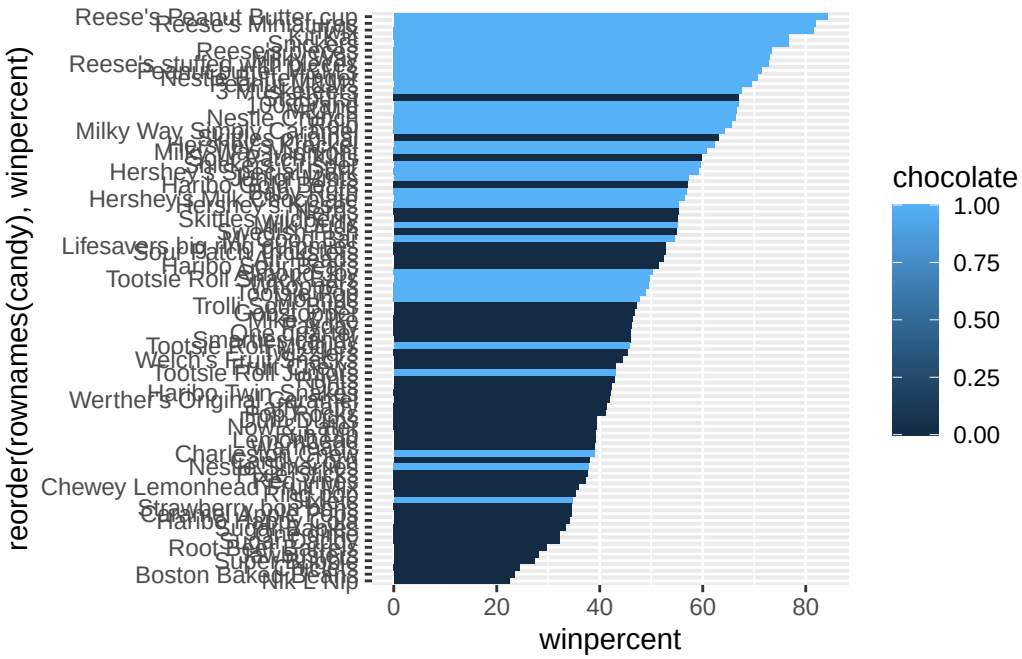
```
head(candy[order(candy$winpercent),], 5)
```

```
                  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
```

Q15. Make a first barplot of candy ranking based on winpercent values. Q16. This is quite ugly, use the reorder() function to get the bars sorted by winpercent?

Let's do a barplot of winpercent values.

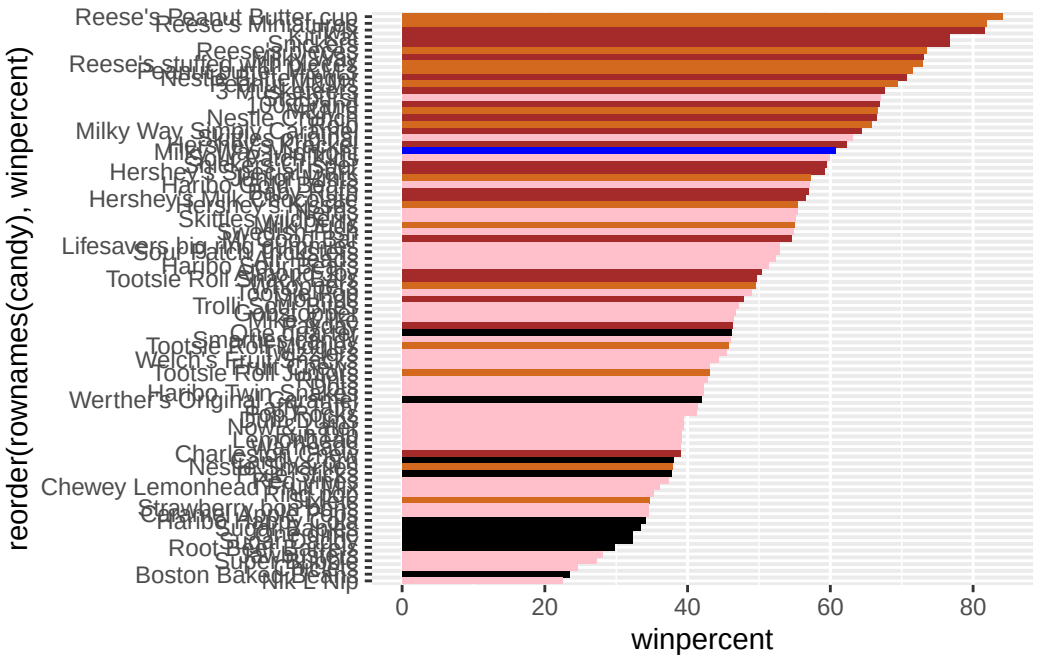
```
ggplot(candy) +
  aes(x = winpercent,
      y = reorder(rownames(candy),winpercent),
      fill = chocolate) +
  geom_col()
```



I want a more custom color scheme where I can see both chocolate and bar and fruity etc. all from the one plot. To do this, we can roll our own color vector…

```
#placeholder color vector
mycols <- rep("black", nrow(candy))
mycols[as.logical(candy$chocolate)] <- "chocolate"
mycols[as.logical(candy$bar)] <- "brown"
mycols[as.logical(candy$fruity)] <- "pink"
mycols[row.names(candy)=="Milky Way Midnight"] <- "blue"

# mycols
```

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

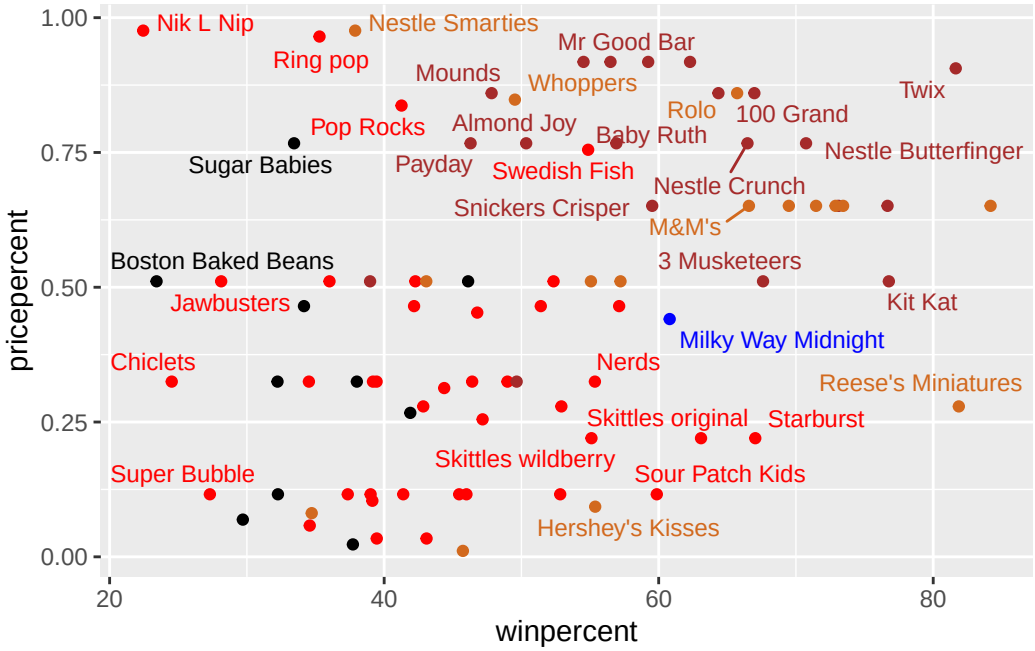
Starburst

Plot of winpercent vs pricepercent to see what would be the best candy to buy...

```
mycols[as.logical(candy$fruity)] <- "red"
```

```
library(ggrepel)

ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=mycols) +
  geom_text_repel(col=mycols, size=3.3, max.overlaps = 8)
```

```
Warning: ggrepel: 52 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 Miniatures

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
```

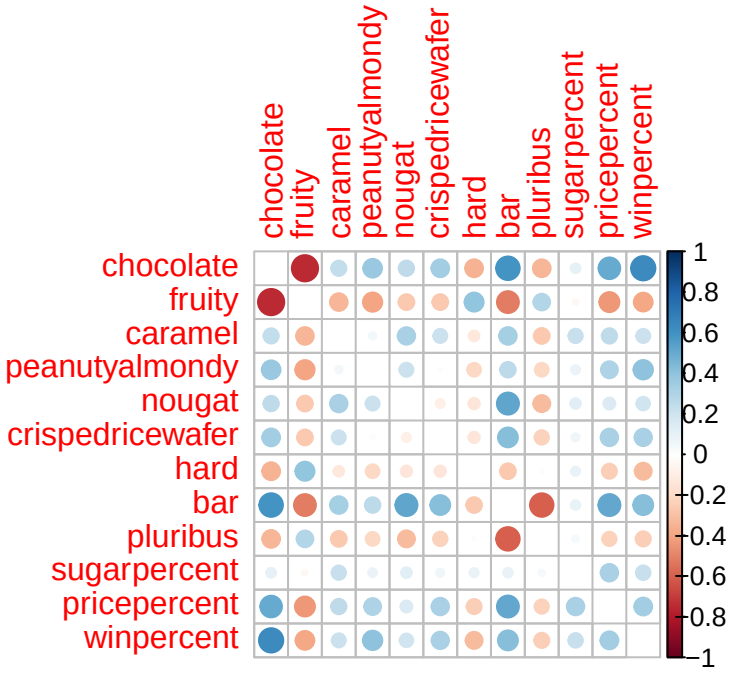
```
library(corrplot)
```

```
corrplot 0.95 loaded
```

```r
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, diag = F)
```



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

Fruity and chocolate.

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

Chocolate and bar, nougat and bar, chocolate and winpercent.

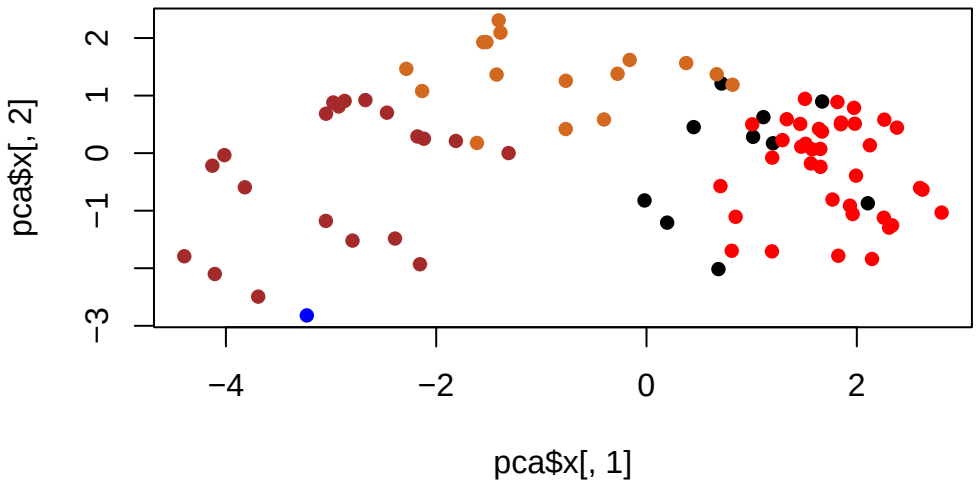
Principal Component Analysis

```
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
```

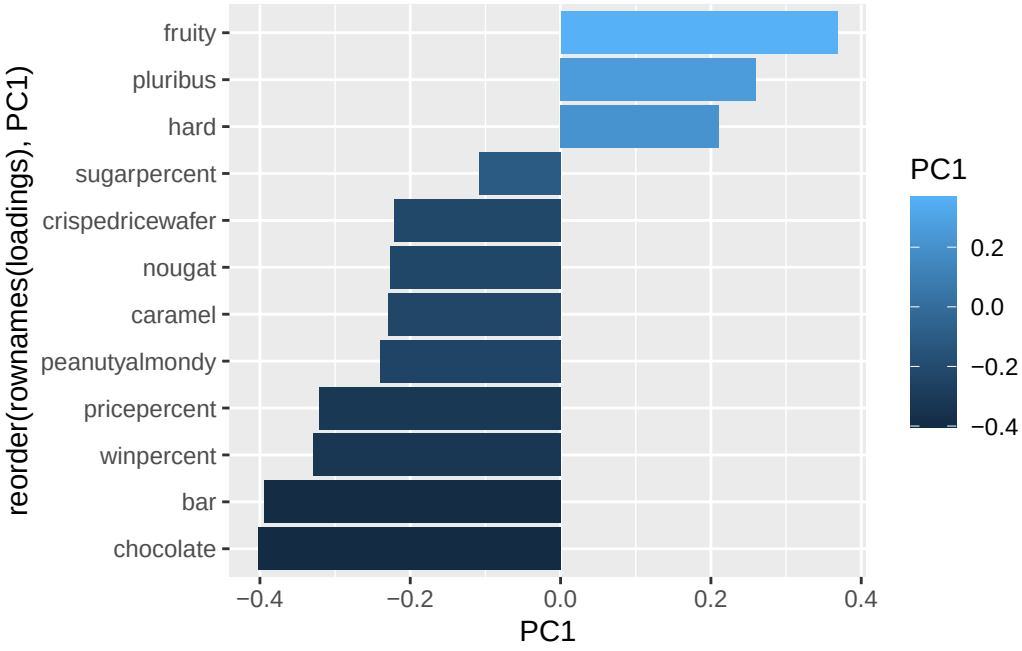
```
                            PC8     PC9     PC10    PC11    PC12
Standard deviation     0.74530 0.67824 0.62349 0.43974 0.39760
Proportion of Variance 0.04629 0.03833 0.03239 0.01611 0.01317
Cumulative Proportion  0.89998 0.93832 0.97071 0.98683 1.00000
```

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



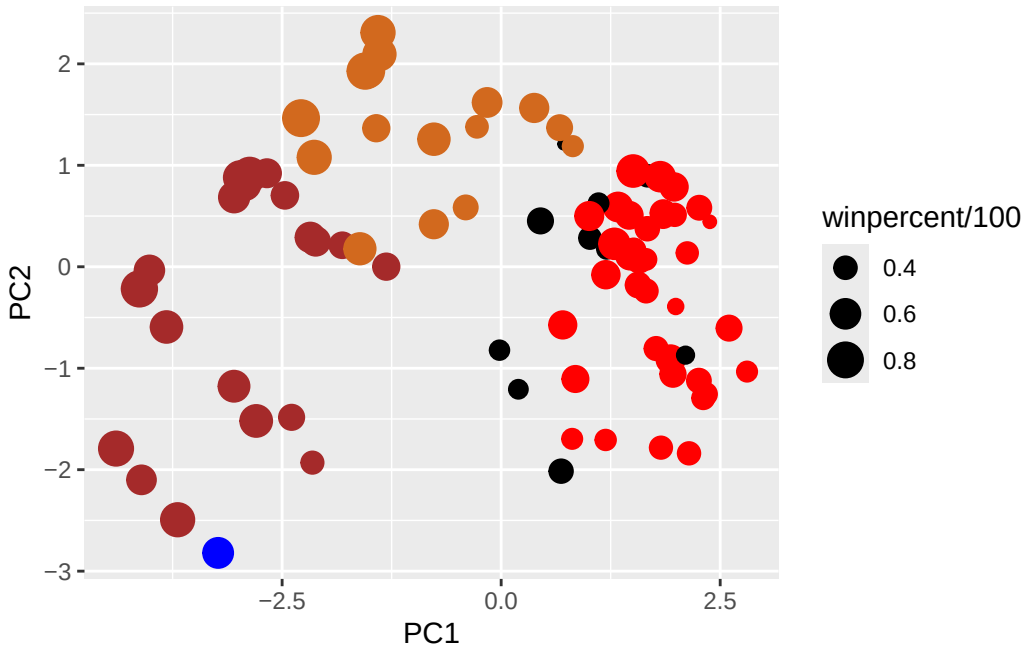
How do the original variables (columns) contribute to the new PCs. I will lok at PC1 here.

```
loadings <- as.data.frame(pca$rotation)

ggplot(loadings) +
  aes(PC1, reorder(rownames(loadings), PC1), fill=PC1) +
  geom_col()
```



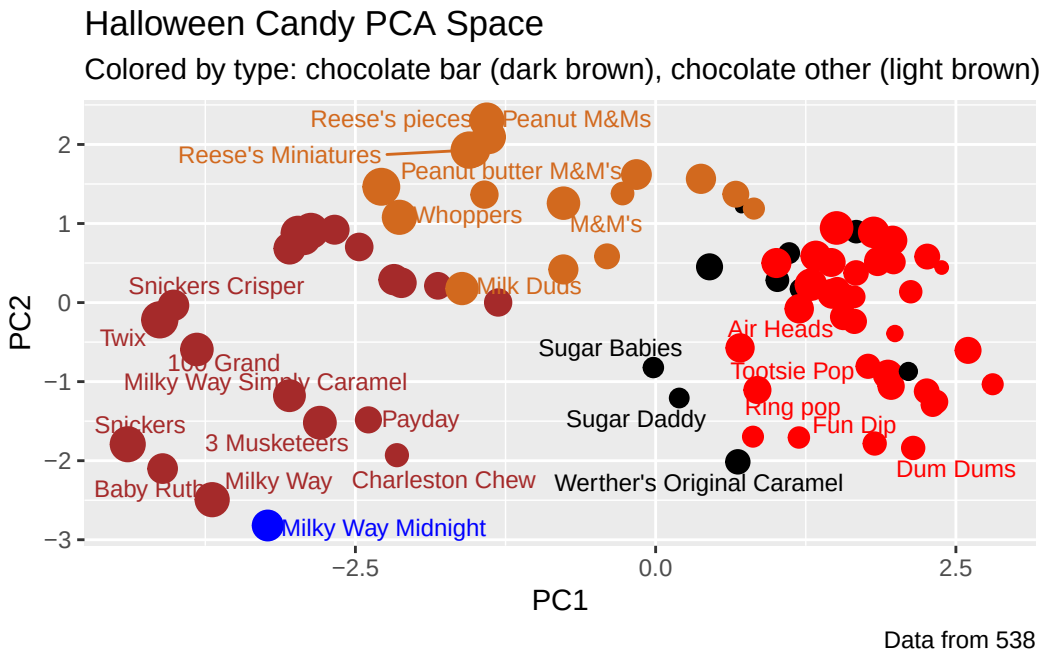

```
my_data <- cbind(candy, pca$x[,1:3])
```

```
p <- ggplot(my_data) +
        aes(x=PC1, y=PC2,
            size=winpercent/100,
            text=rownames(my_data),
            label=rownames(my_data)) +
        geom_point(col=mycols)

p
```




```
library(ggrepel)

p + geom_text_repel(size=3.3, col=mycols, max.overlaps = 7)  +
  theme(legend.position = "none") +
  labs(title="Halloween Candy PCA Space",
       subtitle="Colored by type: chocolate bar (dark brown), chocolate other (light brown),
       caption="Data from 538")
```

```
Warning: ggrepel: 59 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

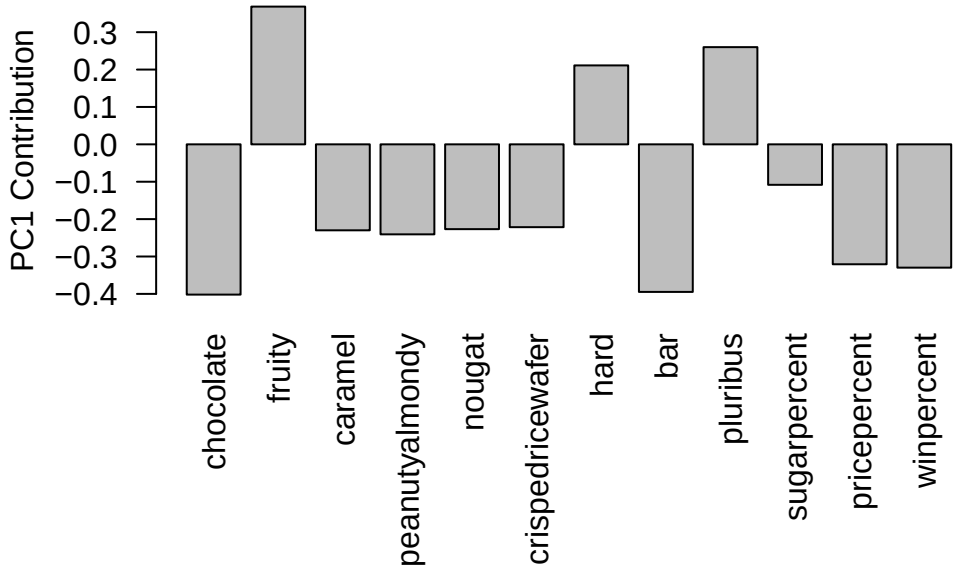



```
library(plotly)
```

```
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)
```

```r
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?

These are fruity candies. The three categories show that PC1 positive direction corresponds to hard, fruity candies that come in multiples. This makes sense since those are most fruity candies.