Class 10: Halloween Candy Mini Project

Nicholas Yousefi

Importing and Inspecting the Data

Here we explore 538 Halloween candy data.

```
candy_file <- "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?

```
nrow(candy)
```

```
[1] 85
```

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

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

```
[1] 38
```

What is your favorite candy?

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

This is all the candy in the dataset:

```
row.names(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] "HersheyÕs Kisses"          "HersheyÕs Krackel"
[25] "HersheyÕs Milk Chocolate"  "HersheyÕs Special Dark"
[27] "Jawbusters"                "Junior Mints"
[29] "Kit Kat"                   "Laffy Taffy"
[31] "Lemonhead"                 "Lifesavers big ring gummies"
[33] "Peanut butter M&MÕs"       "M&MÕs"
[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"                 "ReeseÕs Miniatures"
[53] "ReeseÕs Peanut Butter cup" "ReeseÕs pieces"
[55] "ReeseÕs 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] "WelchÕs Fruit Snacks"         "WertherÕs Original Caramel"
[85] "Whoppers"
```

My favorite candy in the dataset is Smarties candy. Its winpercent value is:

```
candy["Smarties candy", "winpercent"]
```

```
[1] 45.99583
```

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
```

We can use the `skimr::skim()` function to get a general idea of the 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	
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	

```
# alternatively, you can write the following, without calling library(package) (this is us
#skimr::skim(candy)
```

Q6. Is there any variable/column that looks to be on a different scale to the majority of the other columns in the dataset?

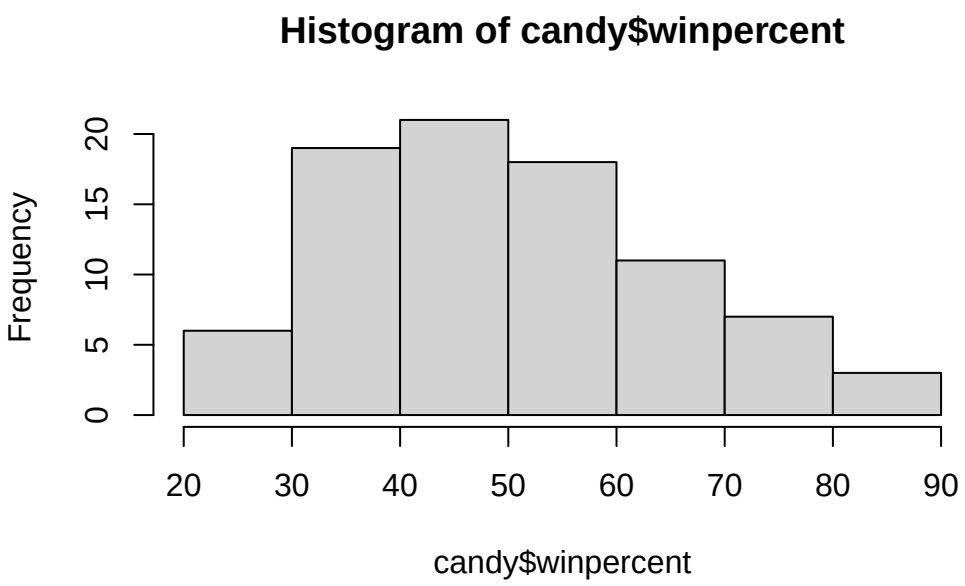
Winnpercent seems to be on a different scale than the majority of other columns in the dataset. It is the only column whose mean and standard deviation are greater than 1.

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

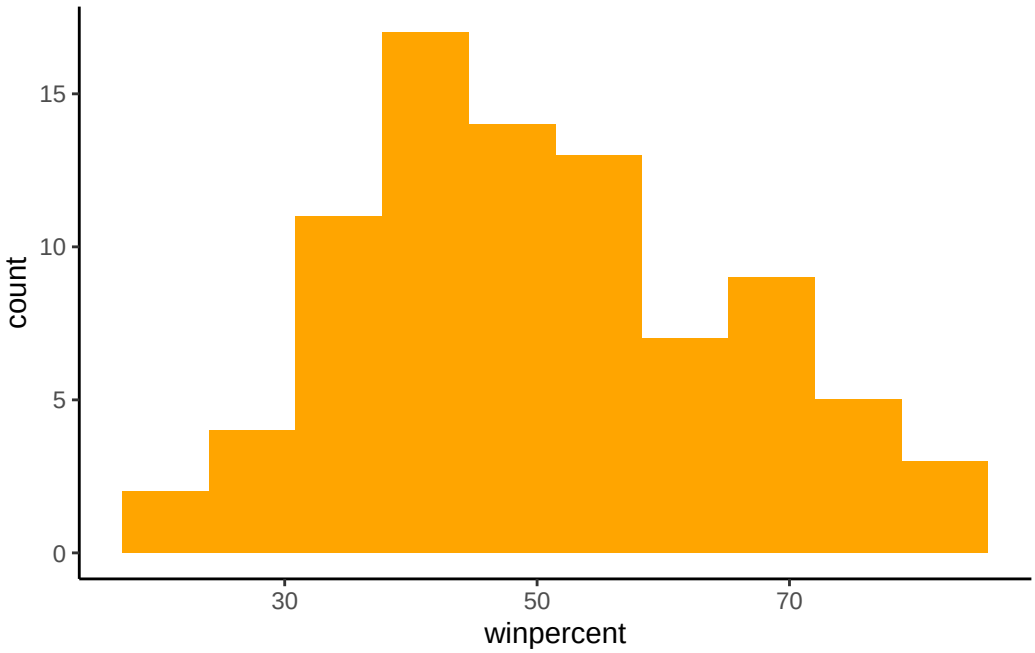
Zero represents that the candy does not contain chocolate. One represents that the candy does contain chocolate.

Q8. Plot a histogram of `winpercent` values

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




```
library(ggplot2)

ggplot(candy) +
  aes(winpercent) +
  geom_histogram(bins = 10, fill="orange") +
  theme_classic()
```



Q9. Is the distribution of winpercent values symmetrical?

The distribution of winpercent values is not symmetrical. It is skewed slightly to the right.

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

The center of the distribution is below 50%.

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

```
chocolate.wins <- candy$winpercent[as.logical(candy$chocolate)]
fruity.wins <- candy$winpercent[as.logical(candy$fruity)]
mean(chocolate.wins)
```

```
[1] 60.92153
```

```
mean(fruity.wins)
```

```
[1] 44.11974
```

Chocolate candy is higher ranked than fruit candy on average.

Q12. Is this difference statistically significant?

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

```
    Welch Two Sample t-test

data:  chocolate.wins and fruity.wins
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
```

Since the p-value is less than 0.05, the difference in means between the two means is statistically significant (i.e. the probability of getting such a difference by random chance is low). Therefore, the difference between chocolate and fruity candy is statistically significant.

Overall Candy Rankings

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

```
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) %>%
  row.names()
```

```
[1] "Nik L Nip"           "Boston Baked Beans" "Chiclets"
[4] "Super Bubble"        "Jawbusters"
```

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

```
candy %>%
  arrange(desc(winpercent)) %>%
  head(5) %>%
  row.names()
```

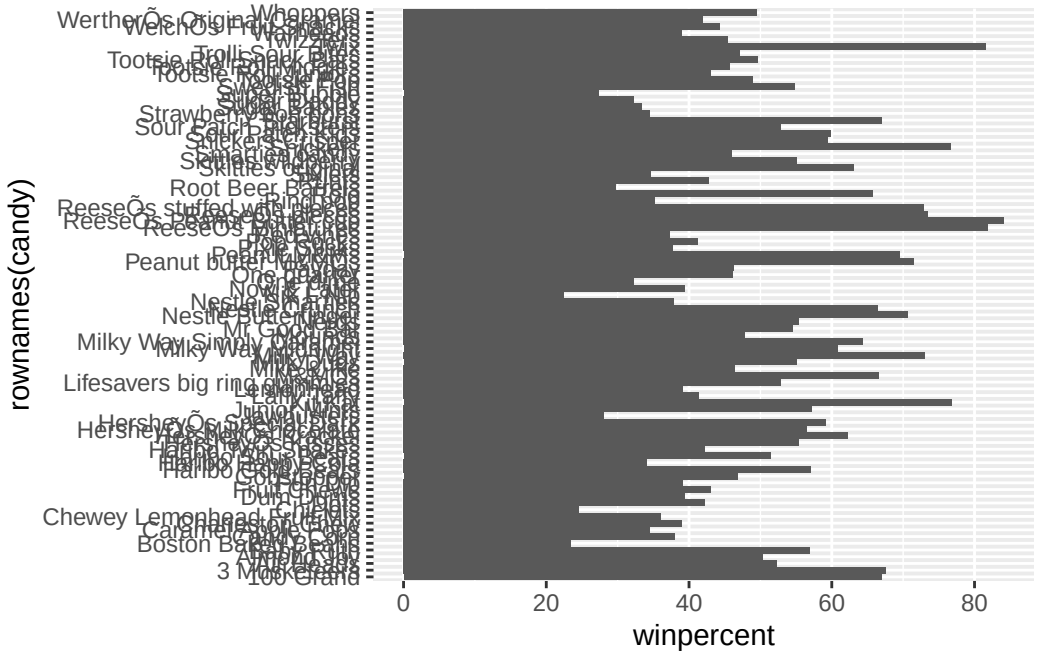
```
[1] "ReeseÕs Peanut Butter cup" "ReeseÕs Miniatures"
[3] "Twix"                      "Kit Kat"
[5] "Snickers"
```

I prefer the dplyr method because it is easier to read the code quickly and see what is being done. With the base R method, you have to know what everything is doing beforehand and really think thorough what is going on to figure out what the code is doing.

Let's make a barplot of the different candy types.

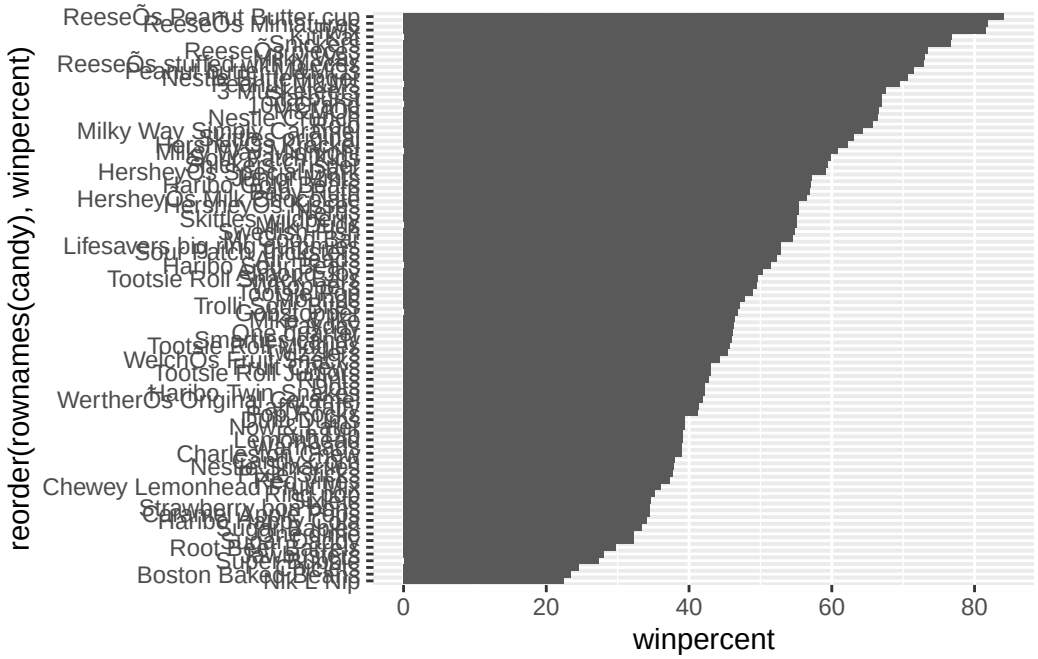
Q15. Make a first barplot of candy ranking based on winpercent values.

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



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

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



Let's add some color to this plot.

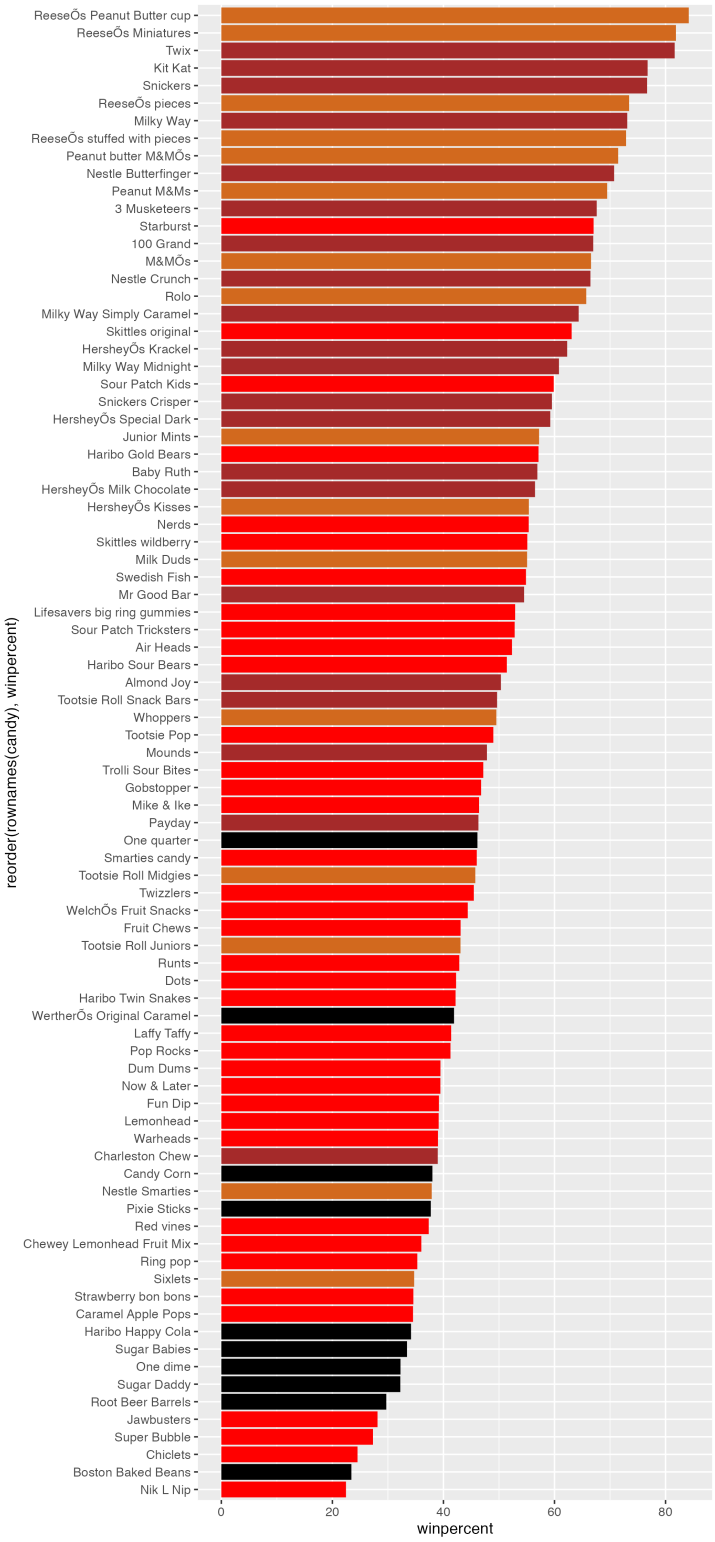
First, set up some colors for different candy types:

```
# first, create a vector of all black colors, then override the cells for different candie
my_cols=rep("black", nrow(candy)) # rep replicates the value in the first argument the sec
my_cols[as.logical(candy$chocolate)] = "chocolate"
my_cols[as.logical(candy$bar)] = "brown" # if something is both chocolate and a bar, overr
my_cols[as.logical(candy$fruity)] = "red"
my_cols
```

```
 [1] "brown"     "brown"     "black"     "black"     "red"       "brown"
 [7] "brown"     "black"     "black"     "red"       "brown"     "red"
[13] "red"       "red"       "red"       "red"       "red"       "red"
[19] "red"       "black"     "red"       "red"       "chocolate" "brown"
[25] "brown"     "brown"     "red"       "chocolate" "brown"     "red"
[31] "red"       "red"       "chocolate" "chocolate" "red"       "chocolate"
[37] "brown"     "brown"     "brown"     "brown"     "brown"     "red"
[43] "brown"     "brown"     "red"       "red"       "brown"     "chocolate"
[49] "black"     "red"       "red"       "chocolate" "chocolate" "chocolate"
[55] "chocolate" "red"       "chocolate" "black"     "red"       "chocolate"
[61] "red"       "red"       "chocolate" "red"       "brown"     "brown"
```

```
[67] "red"       "red"       "red"       "red"       "black"     "black"
[73] "red"       "red"       "red"       "chocolate" "chocolate" "brown"
[79] "red"       "brown"     "red"       "red"       "red"       "black"
[85] "chocolate"
```

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

# cheat and save the plot as a file, so we can set the height and make it so it is not so
ggsave("temp.png", plot = tmp, width=7, height=15)
```


ReeseÕs Peanut Butter cup
ReeseÕs Miniatures
Twix
Kit Kat
Snickers
ReeseÕs pieces
Milky Way
ReeseÕs stuffed with pieces
Peanut butter M&MÕs
Nestle Butterfinger
Peanut M&Ms
3 Musketeers
Starburst
100 Grand
M&MÕs
Nestle Crunch
Rolo
Milky Way Simply Caramel
Skittles original
HersheyÕs Krackel
Milky Way Midnight
Sour Patch Kids
Snickers Crisper
HersheyÕs Special Dark
Junior Mints
Haribo Gold Bears
Baby Ruth
HersheyÕs Milk Chocolate
HersheyÕs Kisses
Nerds
Skittles wildberry
Milk Duds
Swedish Fish
Mr Good Bar
Lifesavers big ring gummies
Sour Patch Tricksters
Air Heads
Haribo Sour Bears
Almond Joy
Tootsie Roll Snack Bars
Whoppers
Tootsie Pop
Mounds
Trolli Sour Bites
Gobstopper
Mike & Ike
Payday
One quarter
Smarties candy
Tootsie Roll Midgies
Twizzlers
WelchÕs Fruit Snacks
Fruit Chews
Tootsie Roll Juniors
Runts
Dots
Haribo Twin Snakes
WertherÕs Original Caramel
Laffy Taffy
Pop Rocks
Dum Dums
Now & Later
Fun Dip
Lemonhead
Warheads
Charleston Chew
Candy Corn
Nestle Smarties
Pixie Sticks
Red vines
Chewey Lemonhead Fruit Mix
Ring pop
Sixlets
Strawberry bon bons
Caramel Apple Pops
Haribo Happy Cola
Sugar Babies
One dime
Sugar Daddy
Root Beer Barrels
Jawbusters
Super Bubble
Chiclets
Boston Baked Beans
Nik L Nip
reorder(rownames(candy), winpercent)
0
20
40
60
80
winpercent

Q17. What is the worst ranked chocolate candy?

The worst ranked chocolate candy is sixlets.

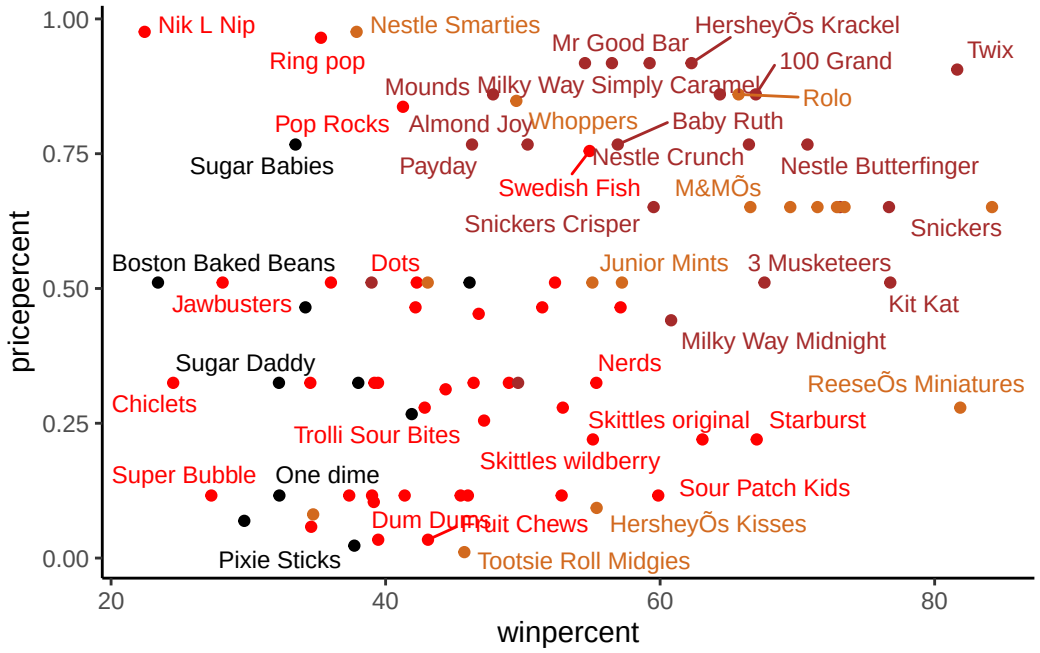
Q18. What is the best ranked fruity candy?

The best ranked fruity candy is starbursts.

Taking a look at pricepercent

```
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) +
  theme_classic()
```

```
Warning: ggrepel: 40 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 is the highest ranked in terms of winpercent for the least money.

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

```
most_expensive <- candy %>%
  arrange(desc(pricepercent)) %>%
  head(5)

most_expensive %>% rownames()
```

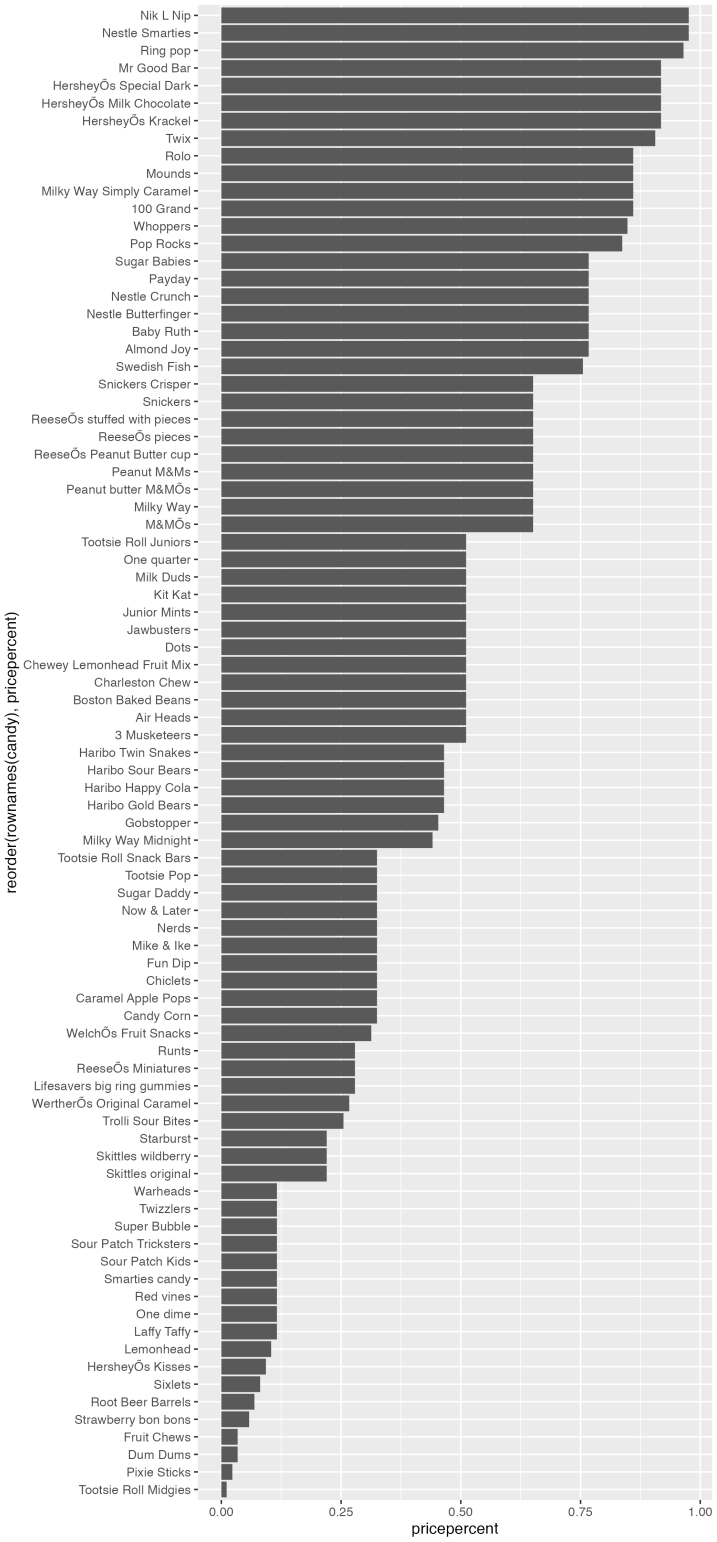
```
[1] "Nik L Nip"                "Nestle Smarties"
[3] "Ring pop"                 "HersheyÕs Krackel"
[5] "HersheyÕs Milk Chocolate"
```

```
most_expensive %>%
  arrange(winpercent) %>%
  head(1) %>%
  rownames()
```

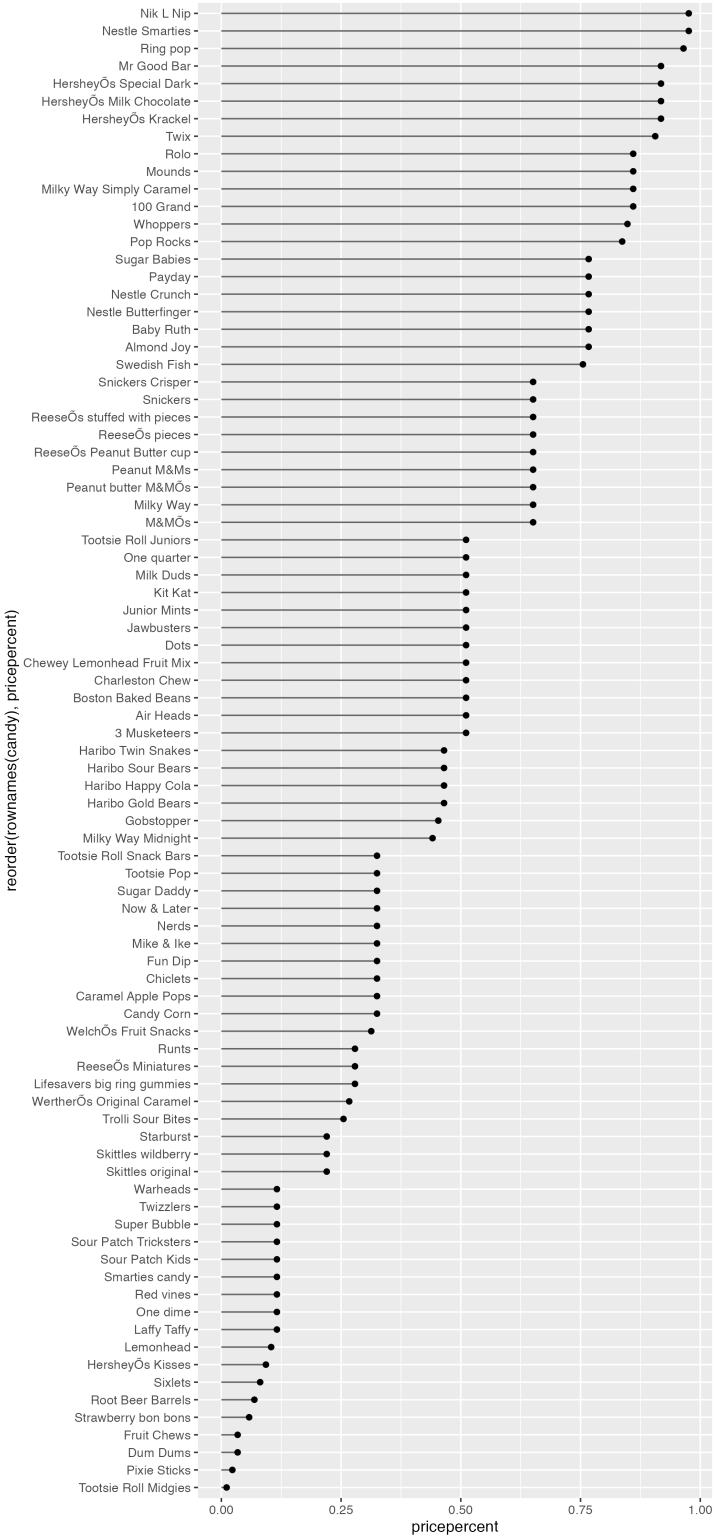
```
[1] "Nik L Nip"
```

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().

```
b <- ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy), pricepercent)) +
  geom_col()

ggsave("tmp_bar.png", plot=b, width=7, height=15)
```


Nik L Nip
Nestle Smarties
Ring pop
Mr Good Bar
HersheyÕs Special Dark
HersheyÕs Milk Chocolate
HersheyÕs Krackel
Twix
Rolo
Mounds
Milky Way Simply Caramel
100 Grand
Whoppers
Pop Rocks
Sugar Babies
Payday
Nestle Crunch
Nestle Butterfinger
Baby Ruth
Almond Joy
Swedish Fish
Snickers Crisper
Snickers
ReeseÕs stuffed with pieces
ReeseÕs pieces
ReeseÕs Peanut Butter cup
Peanut M&Ms
Peanut butter M&MÕs
Milky Way
M&MÕs
Tootsie Roll Juniors
One quarter
Milk Duds
Kit Kat
Junior Mints
Jawbusters
Dots
Chewey Lemonhead Fruit Mix
Charleston Chew
Boston Baked Beans
Air Heads
3 Musketeers
Haribo Twin Snakes
Haribo Sour Bears
Haribo Happy Cola
Haribo Gold Bears
Gobstopper
Milky Way Midnight
Tootsie Roll Snack Bars
Tootsie Pop
Sugar Daddy
Now & Later
Nerds
Mike & Ike
Fun Dip
Chiclets
Caramel Apple Pops
Candy Corn
WelchÕs Fruit Snacks
Runts
ReeseÕs Miniatures
Lifesavers big ring gummies
WertherÕs Original Caramel
Trolli Sour Bites
Starburst
Skittles wildberry
Skittles original
Warheads
Twizzlers
Super Bubble
Sour Patch Tricksters
Sour Patch Kids
Smarties candy
Red vines
One dime
Laffy Taffy
Lemonhead
HersheyÕs Kisses
Sixlets
Root Beer Barrels
Strawberry bon bons
Fruit Chews
Dum Dums
Pixie Sticks
Tootsie Roll Midgies
reorder(rownames(candy), pricepercent)
0.00
0.25
0.50
0.75
1.00
pricepercent

```r
l <- ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy), pricepercent)) +
  geom_segment(aes(yend = reorder(rownames(candy), pricepercent), xend=0), col="gray40") +
  geom_point()

ggsave("tmp_lollipop.png", plot=l, width=7, height=15)
```


Nik L Nip
Nestle Smarties
Ring pop
Mr Good Bar
HersheyÕs Special Dark
HersheyÕs Milk Chocolate
HersheyÕs Krackel
Twix
Rolo
Mounds
Milky Way Simply Caramel
100 Grand
Whoppers
Pop Rocks
Sugar Babies
Payday
Nestle Crunch
Nestle Butterfinger
Baby Ruth
Almond Joy
Swedish Fish
Snickers Crisper
Snickers
ReeseÕs stuffed with pieces
ReeseÕs pieces
ReeseÕs Peanut Butter cup
Peanut M&Ms
Peanut butter M&MÕs
Milky Way
M&MÕs
Tootsie Roll Juniors
One quarter
Milk Duds
Kit Kat
Junior Mints
Jawbusters
Dots
Chewey Lemonhead Fruit Mix
Charleston Chew
Boston Baked Beans
Air Heads
3 Musketeers
Haribo Twin Snakes
Haribo Sour Bears
Haribo Happy Cola
Haribo Gold Bears
Gobstopper
Milky Way Midnight
Tootsie Roll Snack Bars
Tootsie Pop
Sugar Daddy
Now & Later
Nerds
Mike & Ike
Fun Dip
Chiclets
Caramel Apple Pops
Candy Corn
WelchÕs Fruit Snacks
Runts
ReeseÕs Miniatures
Lifesavers big ring gummies
WertherÕs Original Caramel
Trolli Sour Bites
Starburst
Skittles wildberry
Skittles original
Warheads
Twizzlers
Super Bubble
Sour Patch Tricksters
Sour Patch Kids
Smarties candy
Red vines
One dime
Laffy Taffy
Lemonhead
HersheyÕs Kisses
Sixlets
Root Beer Barrels
Strawberry bon bons
Fruit Chews
Dum Dums
Pixie Sticks
Tootsie Roll Midgies
reorder(rownames(candy), pricepercent)
0.00
0.25
0.50
0.75
1.00
pricepercent

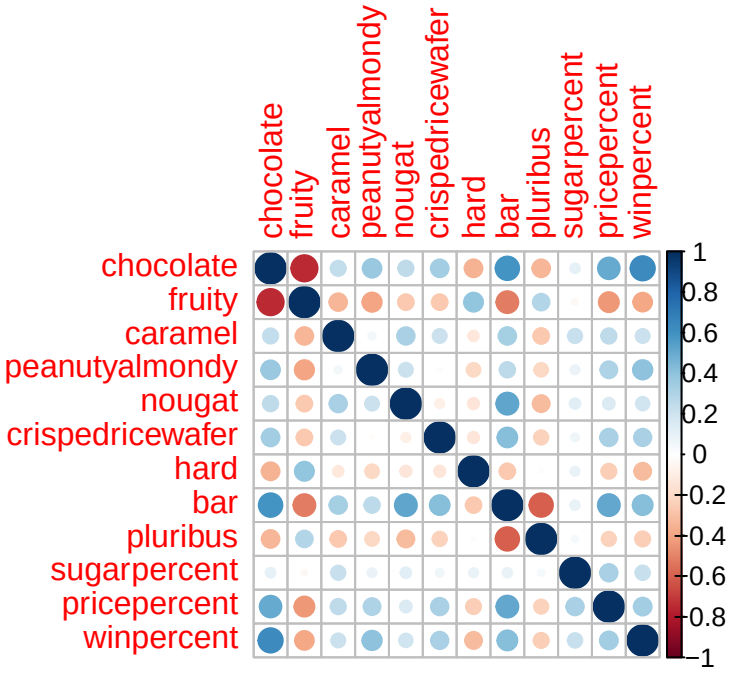
It looks like a lot of candies are about the same price.

Exploring the correlation structure

```
library(corrplot)
```

```
corrplot 0.92 loaded
```

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



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

Chocolate and fruity are anti-correlated.

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

It looks like winpercent and chocolate are the most positively correlated (although it is slightly hard to tell since a couple of variables have similar colors).

Principal Component Analysis

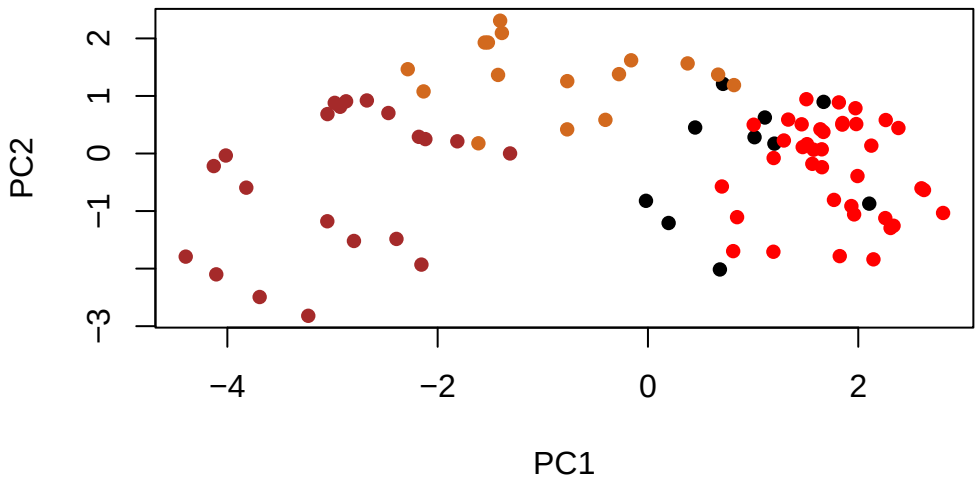
Let's run PCA on our candy data!

```
pca <- prcomp(candy, scale=T)
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
```

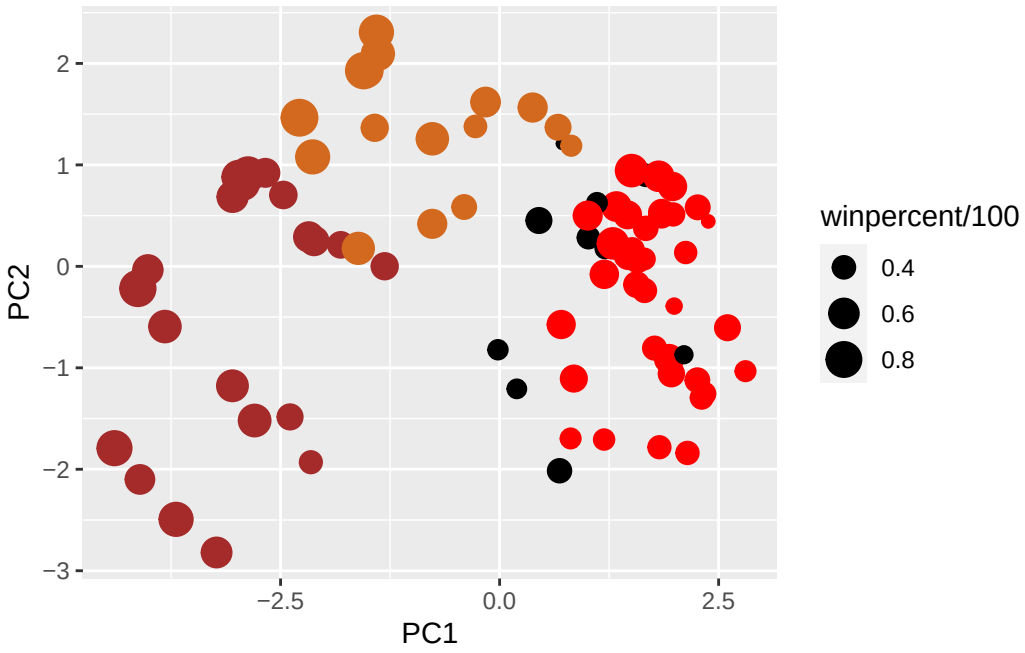
Let's plot PC1 vs. PC2.

```
plot(pca$x[,"PC1"], pca$x[,"PC2"], xlab="PC1", ylab="PC2", col=my_cols, pch=16)
```



Let's use ggplot to make this look nicer.

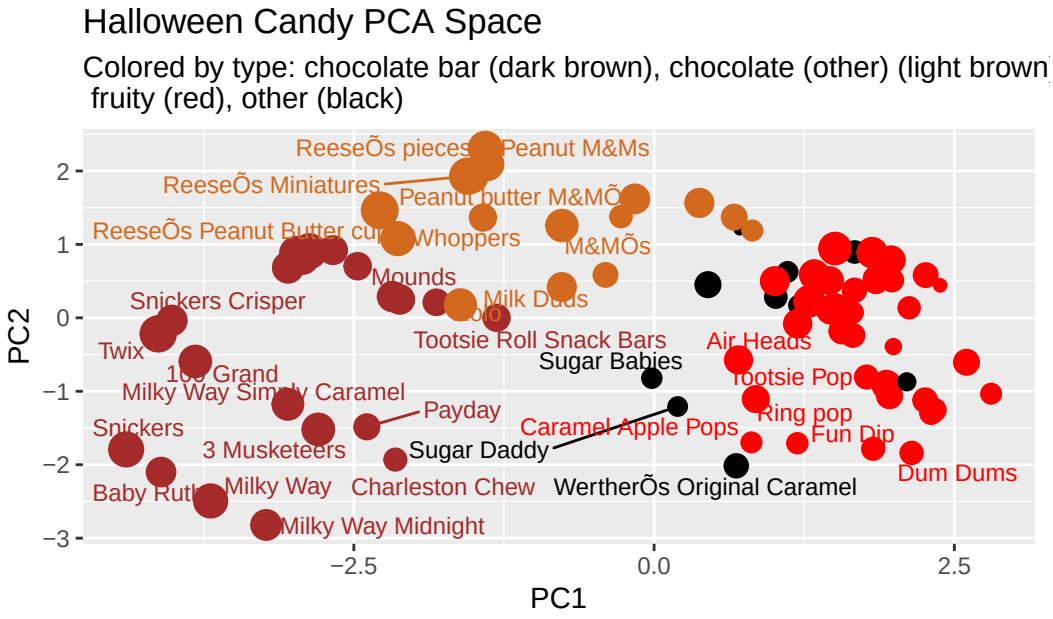
```
my_data <- cbind(candy, pca$x[,1:3])
p <- ggplot(my_data) +
  aes(PC1, PC2,
      size=winpercent / 100,
      text=rownames(my_data),
      label=rownames(my_data)) +
  geom_point(col=my_cols)
p
```



Let's label this plot with names. We don't want them to overlap, so let's use `ggrepel`.

```
p + geom_text_repel(size=3.3, col=my_cols, max.overlaps=10) +
  theme(legend.position = "none") +
  labs(title="Halloween Candy PCA Space",
       subtitle="Colored by type: chocolate bar (dark brown), chocolate (other) (light bro
       caption="Data from 538")
```

```
Warning: ggrepel: 54 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



Let's generate an interactive plot using `plotly`.

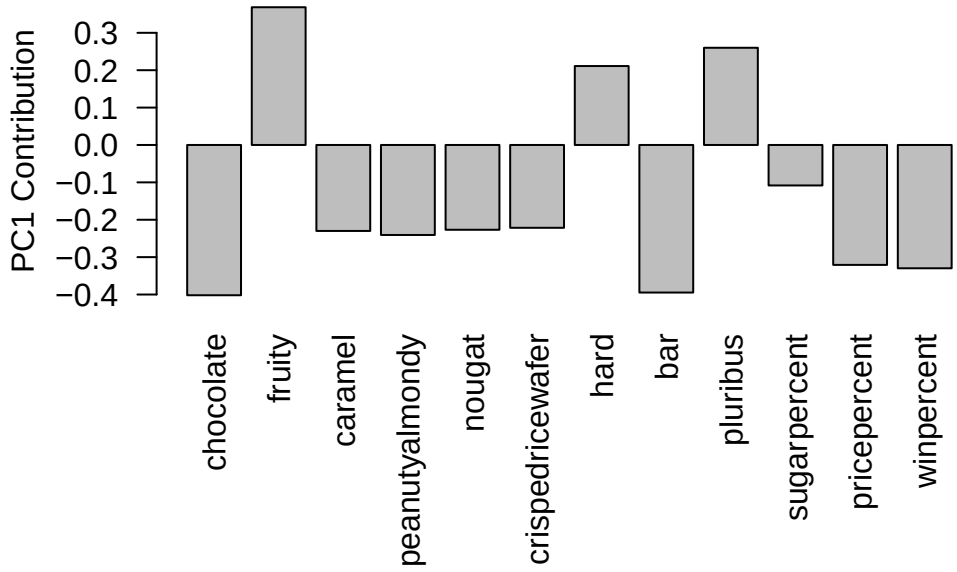
```
library(plotly)
```

```
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)
```

Lets take a look at the PCA loadings.

```
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?

The original variables that are picked up strongly by PC1 in the positive direction are `fruity`, `hard`, and `pluribus`. These make sense to me since fruity candies tend to be hard and tend to come as multiple small pieces of candy as opposed to one large hard bar that you can't bite through or fit into your mouth.