

Mini Candy Project

Ebru Robinson

Table of contents

Winpercent vs Pricepercent 15

```
read.csv("candy-data.csv")
```

competitor-name	choco-late	fruitycaram	peanutyal-candy	nougat	crispedrice-wafer	hardbar	ricebar	sug-pluribuscent	arper-	pri-ceper-	win-per-	
100 Grand	1	0	1	0	0	1	0	1	0	0.732	0.860	66.97173
3 Musketeers	1	0	0	0	1	0	0	1	0	0.604	0.511	67.60294
One dime	0	0	0	0	0	0	0	0	0	0.011	0.116	32.26109
One quarter	0	0	0	0	0	0	0	0	0	0.011	0.511	46.11650
Air Heads	0	1	0	0	0	0	0	0	0	0.906	0.511	52.34146
Almond Joy	1	0	0	1	0	0	0	1	0	0.465	0.767	50.34755
Baby Ruth	1	0	1	1	1	0	0	1	0	0.604	0.767	56.91455
Boston Baked Beans	0	0	0	1	0	0	0	0	1	0.313	0.511	23.41782
Candy Corn	0	0	0	0	0	0	0	0	1	0.906	0.325	38.01096
Caramel Apple Pops	0	1	1	0	0	0	0	0	0	0.604	0.325	34.51768
Charleston Chew	1	0	0	0	1	0	0	1	0	0.604	0.511	38.97504
Chewey Lemonhead	0	1	0	0	0	0	0	0	1	0.732	0.511	36.01763
Fruit Mix Chiclets	0	1	0	0	0	0	0	0	1	0.046	0.325	24.52499
Dots	0	1	0	0	0	0	0	0	1	0.732	0.511	42.27208

competitor-name	choco-late	fruitycaram	peanutyal-	hardy	nougat	crispedrice-	wafer	hardbar	pluribus	sug-ar	pri-cep-	win-per-
										per-	cent	cent
Dum Dums	0	1	0	0	0	0	1	0	0	0.732	0.034	39.46056
Fruit Chews	0	1	0	0	0	0	0	0	1	0.127	0.034	43.08892
Fun Dip	0	1	0	0	0	0	1	0	0	0.732	0.325	39.18550
Gobstopper	0	1	0	0	0	0	1	0	1	0.906	0.453	46.78335
Haribo	0	1	0	0	0	0	0	0	1	0.465	0.465	57.11974
Gold Bears												
Haribo	0	0	0	0	0	0	0	0	1	0.465	0.465	34.15896
Happy Cola												
Haribo Sour	0	1	0	0	0	0	0	0	1	0.465	0.465	51.41243
Bears												
Haribo	0	1	0	0	0	0	0	0	1	0.465	0.465	42.17877
Twin												
Snakes												
Hershey's	1	0	0	0	0	0	0	0	1	0.127	0.093	55.37545
Kisses												
Hershey's	1	0	0	0	0	1	0	1	0	0.430	0.918	62.28448
Krackel												
Hershey's	1	0	0	0	0	0	0	1	0	0.430	0.918	56.49050
Milk												
Chocolate												
Hershey's	1	0	0	0	0	0	0	1	0	0.430	0.918	59.23612
Special												
Dark												
Jawbusters	0	1	0	0	0	0	1	0	1	0.093	0.511	28.12744
Junior	1	0	0	0	0	0	0	0	1	0.197	0.511	57.21925
Mints												
Kit Kat	1	0	0	0	0	1	0	1	0	0.313	0.511	76.76860
Laffy Taffy	0	1	0	0	0	0	0	0	0	0.220	0.116	41.38956
Lemonhead	0	1	0	0	0	0	1	0	0	0.046	0.104	39.14106
Lifesavers	0	1	0	0	0	0	0	0	0	0.267	0.279	52.91139
big ring												
gummies												
Peanut	1	0	0	1	0	0	0	0	1	0.825	0.651	71.46505
butter												
M&M's												
M&M's	1	0	0	0	0	0	0	0	1	0.825	0.651	66.57458
Mike & Ike	0	1	0	0	0	0	0	0	1	0.872	0.325	46.41172
Milk Duds	1	0	1	0	0	0	0	0	1	0.302	0.511	55.06407
Milky Way	1	0	1	0	1	0	0	1	0	0.604	0.651	73.09956

competitor-name	choco-late	fruitycaram	peanutyal-	crispedrice-	arper-	sug-	pri-	win-
	choco-late	fruitycaram	peanutyal-	crispedrice-	arper-	sug-	pri-	win-
Milky Way	1	0	1	0	1	0	0	0.732
Midnight							0.441	60.80070
Milky Way	1	0	1	0	0	0	0	0.965
Simply Caramel							0.860	64.35334
Mounds	1	0	0	0	0	0	0	0.313
Mr Good Bar	1	0	0	1	0	0	0	0.313
Nerds	0	1	0	0	0	1	1	0.848
Nestle Butterfinger	1	0	0	1	0	0	0	0.604
Nestle Crunch	1	0	0	0	0	1	0	0.313
Nik L Nip	0	1	0	0	0	0	1	0.197
Now & Later	0	1	0	0	0	0	1	0.220
Payday	0	0	0	1	1	0	0	0.465
Peanut M&Ms	1	0	0	1	0	0	1	0.593
Pixie Sticks	0	0	0	0	0	0	1	0.093
Pop Rocks	0	1	0	0	0	1	0	0.604
Red vines	0	1	0	0	0	0	1	0.581
Reese's Miniatures	1	0	0	1	0	0	0	0.034
Reese's Peanut Butter cup	1	0	0	1	0	0	0	0.720
Reese's pieces	1	0	0	1	0	0	1	0.406
Reese's stuffed with pieces	1	0	0	1	0	0	0	0.651
Ring pop	0	1	0	0	0	1	0	0.965
Rolo	1	0	1	0	0	0	1	0.860
Root Beer Barrels	0	0	0	0	0	1	0	0.069
Runts	0	1	0	0	0	1	0	0.872
Sixlets	1	0	0	0	0	0	1	0.220
							0.279	42.84914
							0.081	34.72200

competitor-name	choco-late	fruitycaram	peanutyalmond	crispedrice-nougat	rice-wafer	hardbar	aribuscent	sug-cep-	pri-cep-	win-
								per-	per-	per-
Skittles original	0	1	0	0	0	0	0	1	0.941	0.220
Skittles wildberry	0	1	0	0	0	0	0	1	0.941	0.220
Nestle Smarties	1	0	0	0	0	0	0	1	0.267	0.976
Smarties candy	0	1	0	0	0	0	1	0	0.267	0.116
Snickers	1	0	1	1	1	0	0	0	0.546	0.651
Snickers	1	0	1	1	0	1	0	1	0.604	0.651
Crisper										
Sour Patch Kids	0	1	0	0	0	0	0	1	0.069	0.116
Sour Patch Tricksters	0	1	0	0	0	0	0	1	0.069	0.116
Starburst	0	1	0	0	0	0	0	1	0.151	0.220
Strawberry bon bons	0	1	0	0	0	0	1	1	0.569	0.058
Sugar Babies	0	0	1	0	0	0	0	1	0.965	0.767
Sugar Daddy	0	0	1	0	0	0	0	0	0.418	0.325
Super Bubble	0	1	0	0	0	0	0	0	0.162	0.116
Swedish Fish	0	1	0	0	0	0	0	1	0.604	0.755
Tootsie Pop	1	1	0	0	0	0	1	0	0.604	0.325
Tootsie Roll Juniors	1	0	0	0	0	0	0	0	0.313	0.511
Tootsie Roll Midgies	1	0	0	0	0	0	0	1	0.174	0.011
Tootsie Roll Snack Bars	1	0	0	0	0	0	0	0	0.465	0.325
Trolli Sour Bites	0	1	0	0	0	0	0	1	0.313	0.255
Twix	1	0	1	0	0	1	0	1	0	0.546
Twizzlers	0	1	0	0	0	0	0	0	0.220	0.116
Warheads	0	1	0	0	0	0	1	0	0.093	0.116

competitor-name	choco-late	fruity	caram-el	peanuty-almond	yalmond	nougat	crispedrice-nougat	rice-wafer	arbar	per-hardbar	sug-pluribus	arcent	pri-ceper-	win-per-
Welch's Fruit Snacks	0	1	0	0	0	0	0	0	1	0	0.313	0.313	44.37552	
Werther's Original Caramel Whoppers	0	0	1	0	0	0	0	1	0	0	0.186	0.267	41.90431	
	1	0	0	0	0	1	1	0	0	1	0.872	0.848	49.52411	

```
candy_file <- "candy-data.csv"
```

```
candy = read.csv(candy_file, row.names = 1)
head(candy)
```

	choco-late	fruity	caram-el	peanuty-almond	yalmond	nougat	crispedrice-wafer	rice-hard bar	arpluribus	sugcent	arper-cent	pri-ceper-	win-per-	
100 Grand	1	0	1	0	0	0	1	0	1	0	0.732	0.860	66.97173	
3 Musketeers	1	0	0	0	1	0	0	0	1	0	0.604	0.511	67.60294	
One dime	0	0	0	0	0	0	0	0	0	0	0.011	0.116	32.26109	
One quarter	0	0	0	0	0	0	0	0	0	0	0.011	0.511	46.11650	
Air Heads	0	1	0	0	0	0	0	0	0	0	0.906	0.511	52.34146	
Almond Joy	1	0	0	1	0	0	0	0	1	0	0.465	0.767	50.34755	

```
library(flextable)
flextable::flextable(head(candy))
```

chocolate	fruity	caramel	peanuty-almondy	nougat	crispedrice-wafer	hard	bar	pluribus
1	0	1	0	0	1	0	1	0
1	0	0	0	1	0	0	1	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	1	0

```
library(dplyr)
candy |>nrow()
```

[1] 85

```
win <- candy$winpercent
win.mean <- mean(win)
round(win.mean)
```

[1] 50

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)
```

[1] 38

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

My favorite candy is Snickers.I can find its popularity as follows:

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

```
[1] 76.67378
```

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("skimr")
skim(candy) |> as.data.frame() |> kable()
```

skim_type	skim_varincomplete	miss_rate	completeness	nu-meric.mean	nu-meric.sd	nu-meric.p0	nu-meric.p25	nu-meric.p50	nu-meric.p75	nu-meric.p100	nu-meric.hist
numerical	chocolate	0	1	0.43529404987379000000	0.000000	0.000000	1.000000	1.000000	1.000000	1.000000	
numerical	late										
numerical	fruity	0	1	0.44705885001400000000	0.000000	0.000000	1.000000	1.000000	1.000000	1.000000	
numerical	caramel	0	1	0.16470593731102000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	peanutyalmond	0	1	0.16470593731102000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	nougat	0	1	0.08235292765302000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	crispedrice-wafer	0	1	0.08235292765302000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	hard	0	1	0.17647063834805000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	bar	0	1	0.24705884338609000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
numerical	pluribus	0	1	0.51764705026500000000	0.000000	1.000000	1.000000	1.000000	1.000000	1.000000	

skim_type	skim_varin	miss-ing	com-plete_rate	nu-meric.mean	nu-meric.sd	nu-meric.p0	nu-meric.p25	nu-meric.p50	nu-meric.p75	nu-meric.p100	nu-meric.hist
nu-meric	sug-aper-	0	1	0.4786470.282770	0.9011000.22000	0.46500	0.732	0.98800			
	cent										
nu-meric	pri-cep-	0	1	0.4688820.285730	0.6011000.25500	0.46500	0.651	0.97600			
	cent										
nu-meric	winper-	0	1	50.3167638.714392445349.1410617.829759.864	84.18029						
	cent										

```
skim(candy)
```

Table 5: 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_miss-ing	com-plete_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	
peanutyaly-	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
mondy										
nougat	0	1	0.08	0.28	0.00	0.00	0.00	0.00	1.00	
crispedrice-	0	1	0.08	0.28	0.00	0.00	0.00	0.00	1.00	
wafer										
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_vari- able	n_miss- ing	com- plete_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?

Yes. The variable winpercent is on a different scale than most others, because it ranges from about 22 to 84, whereas most variables are either 0 or 1. The variables sugarpercent and pricepercent are also continuous but remain within 0–1, so winpercent stands out the most.

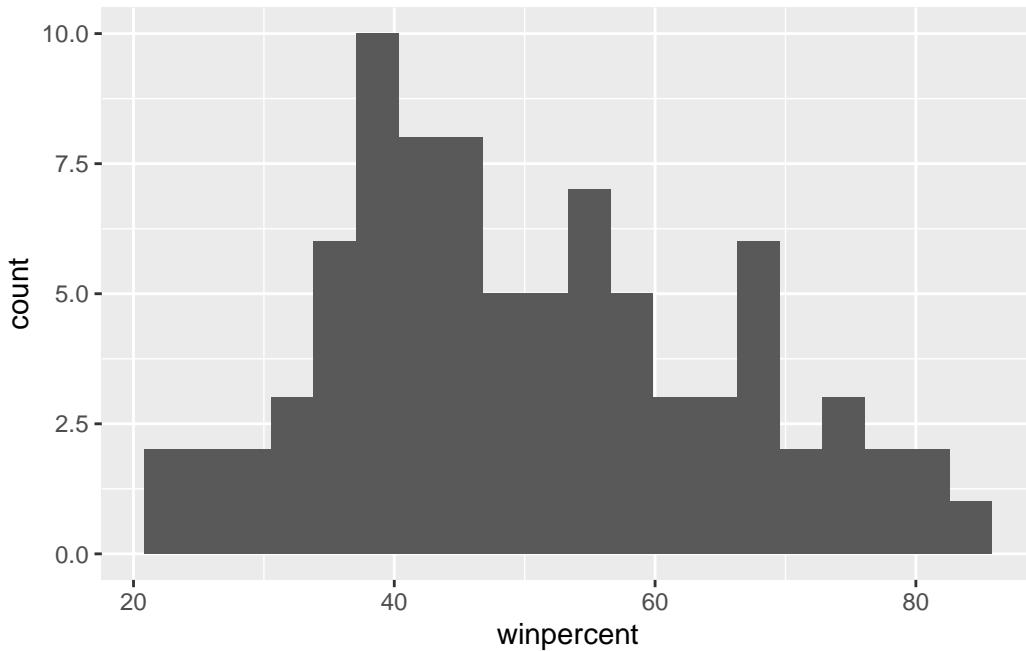
Q7. What do you think a zero and one represent for the candy\$chocolate column?

A zero indicates that the candy does not contain chocolate, and a one indicates that it does contain chocolate.

Q8. Plot a histogram of winpercent values

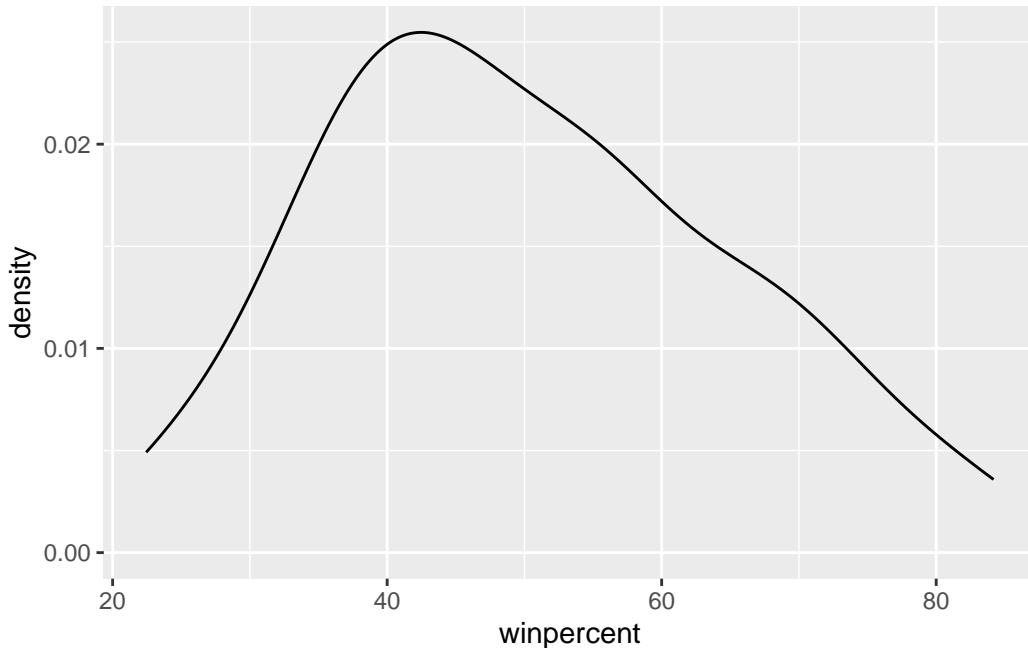
```
library(ggplot2)

ggplot(candy)+ aes(winpercent)+geom_histogram(bins=20)
```



Q9. Is the distribution of winpercent values symmetrical?

```
ggplot(candy)+ aes(winpercent)+geom_density()
```



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

Above

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

[1] 50.31676

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

```
# 1. Find all chocolate candies
choc inds <- as.logical(candy$chocolate)
choc.candy <- candy[choc inds, ]

# 2. Extract the winpercent values
choc.win <- choc.candy$winpercent

# 3. Find the mean winpercent for chocolate candies
```

```
choc.mean <- mean(choc.win)

# 4. Do the same for fruity candies
fruit inds <- as.logical(candy$fruity)
fruit.candy <- candy[fruit inds, ]

# 5. Extract winpercent values for fruity candies
fruit.win <- fruit.candy$winpercent

# 6. Find the mean winpercent for fruity candies
fruit.mean <- mean(fruit.win)

# 7. Compare the two means
choc.mean
```

[1] 60.92153

```
fruit.mean
```

[1] 44.11974

Q12. Is this difference statistically significant?

There is strong statistical evidence ($p < 0.001$) that chocolate candies are, on average, significantly more popular than fruity candies.

```
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
```

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

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

	choco- late	fruity caram- el	peanutyal- mond	almond- nougat	crispedrice- wafer	rice- hard wafer	arper- bar	sug- pluribus	pri- cent	ceper- cent	win- percent
Nik L	0	1	0	0	0	0	0	1	0.197	0.976	22.44534
Nip	0	0	0	1	0	0	0	1	0.313	0.511	23.41782
Boston Baked Beans	0	1	0	0	0	0	0	1	0.046	0.325	24.52499
Chiclets	0	1	0	0	0	0	0	0	0.162	0.116	27.30386
Super Bubble	0	1	0	0	0	0	0	0	0.093	0.511	28.12744
Jaw- busters	0	1	0	0	0	0	1	1	0.720	0.651	84.18029

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

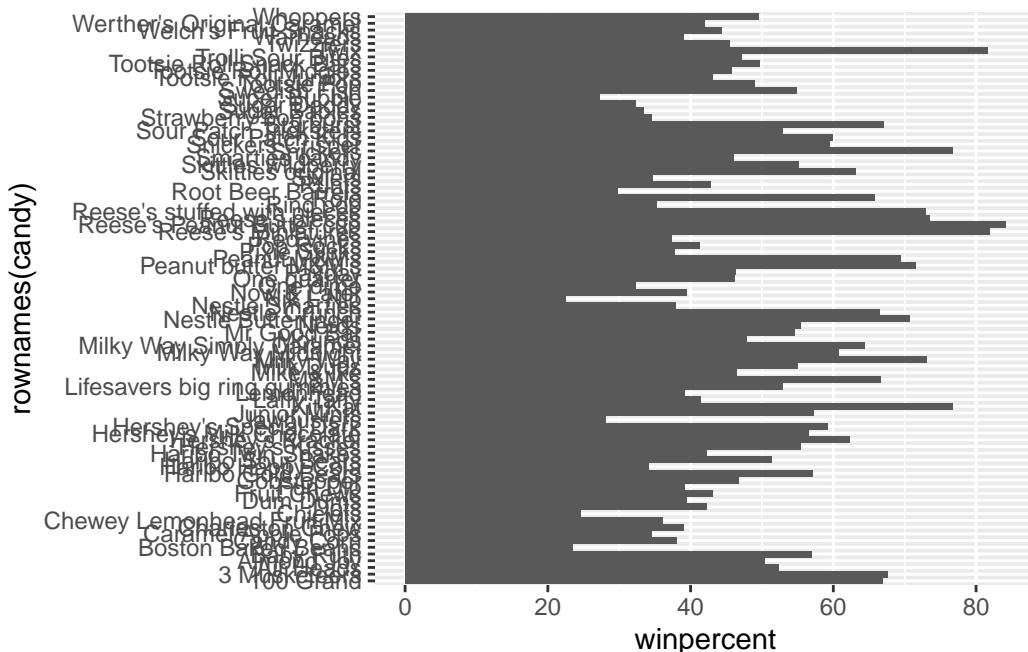
```
head(candy[order(-candy$winpercent), ], n = 5)
```

	choco- late	fruity caram- el	peanutyal- mond	almond- nougat	crispedrice- wafer	rice- hard wafer	arper- bar	sug- pluribus	pri- cent	ceper- cent	win- percent
Reese's Peanut Butter cup	1	0	0	1	0	0	0	0	0.720	0.651	84.18029
Reese's Miniatures	1	0	0	1	0	0	0	0	0.034	0.279	81.86626
Twix	1	0	1	0	0	1	0	1	0	0.546	0.906
Kit Kat	1	0	0	0	0	1	0	1	0	0.313	0.511
Snickers	1	0	1	1	1	0	0	1	0	0.546	0.651

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

```
library(ggplot2)

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

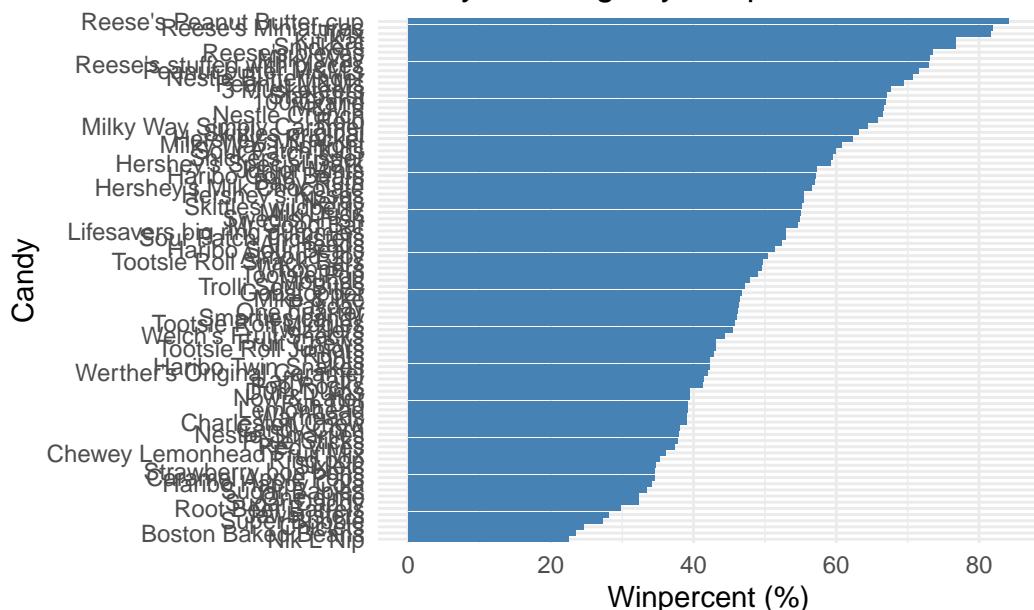


Q16. This is quite ugly, use the `reorder()` function to get the bars sorted by `winpercent`?

```
library(ggplot2)

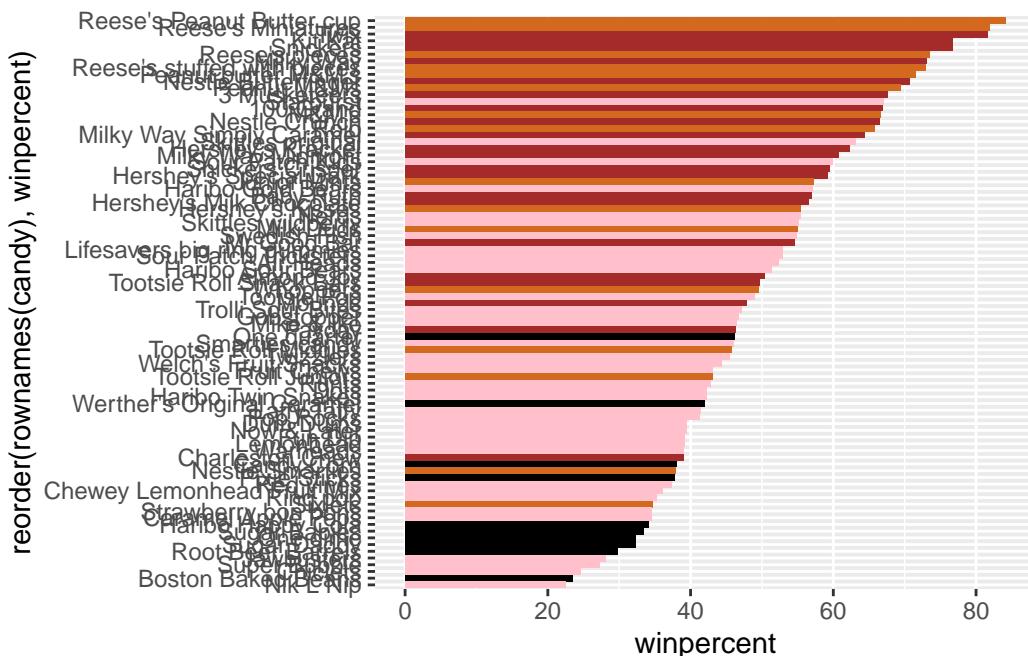
ggplot(candy) +
  aes(x = winpercent, y = reorder(rownames(candy), winpercent)) +
  geom_col(fill = "steelblue") +
  labs(
    title = "Candy Rankings by Winpercent",
    x = "Winpercent (%)",
    y = "Candy"
  ) +
  theme_minimal()
```

Candy Rankings by Winpercent



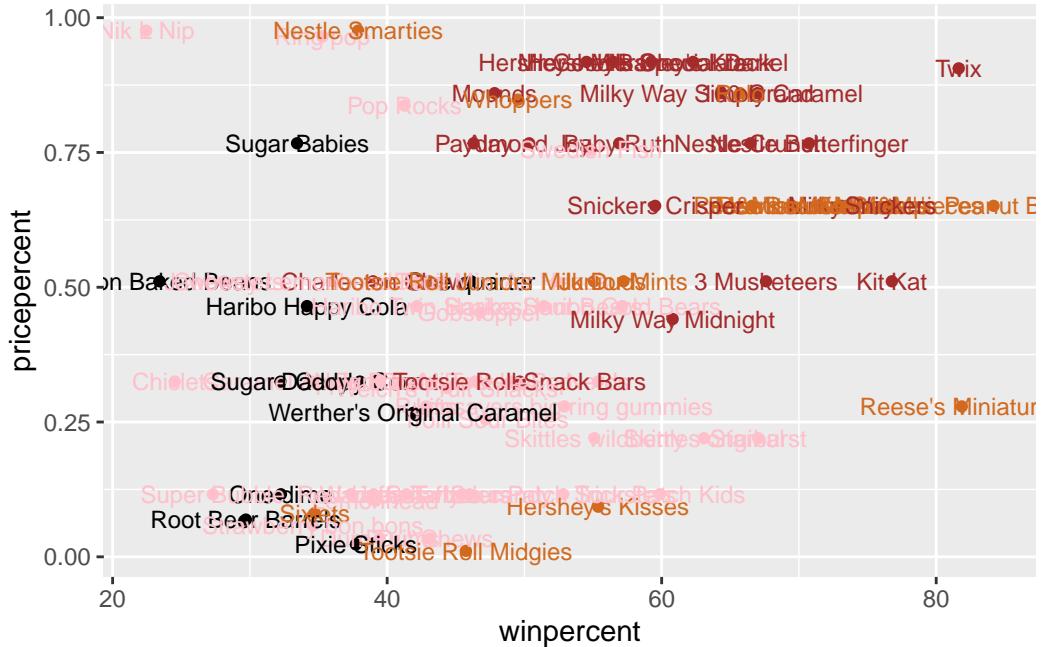
```
my_cols=rep("black", nrow(candy))
my_cols[as.logical(candy$chocolate)] = "chocolate"
my_cols[as.logical(candy$bar)] = "brown"
my_cols[as.logical(candy$fruity)] = "pink"

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



Winpercent vs Pricepercent

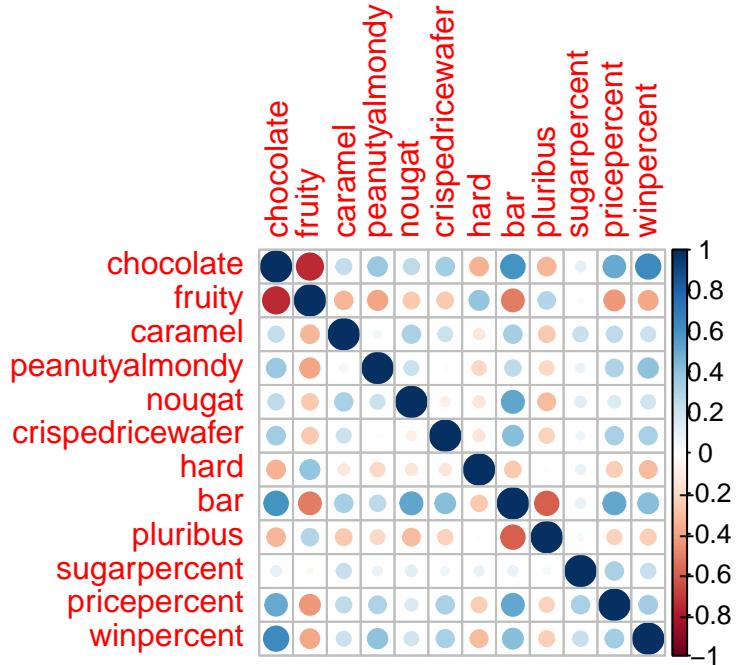
```
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
  geom_text(col=my_cols, size=3.3, max.overlaps = 5)
```



```
# ##corrplot
```

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

```
library(corrplot)
corrplot(cij)
```



##PCA

The main function of this on R is `prcomp()` and we want to set `scale=TRUE` here:

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

Importance of components:

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12
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					
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 look at the first main result figure- the “pc-plot” or PC1 vs PC2

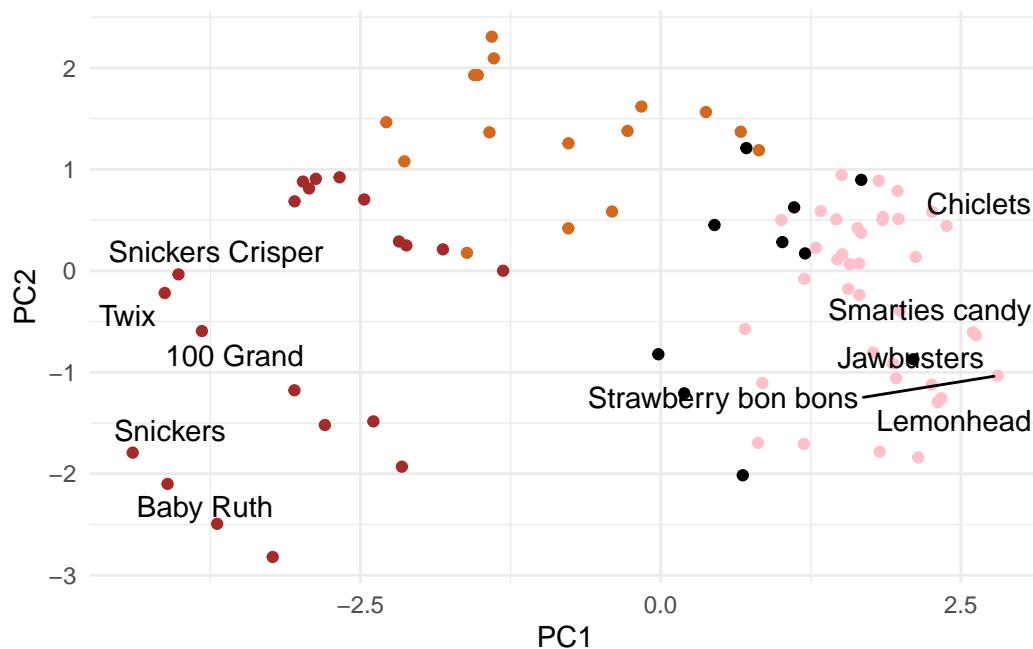
```
library(ggrepel)
# Find points to label
label_points <- as.data.frame(pca$x) %>%
  slice_max(PC1, n = 5) %>%
```

```

bind_rows(slice_min(as.data.frame(pca$x), PC1, n = 5))

ggplot(as.data.frame(pca$x), aes(PC1, PC2)) +
  geom_point(col = my_cols) +
  geom_text_repel(
    data = label_points,
    aes(label = rownames(label_points)),
    col = "black"
  ) +
  theme_minimal()

```



Don't forget about your variable "loadings" - how the original variables contribute to your new PC's

```

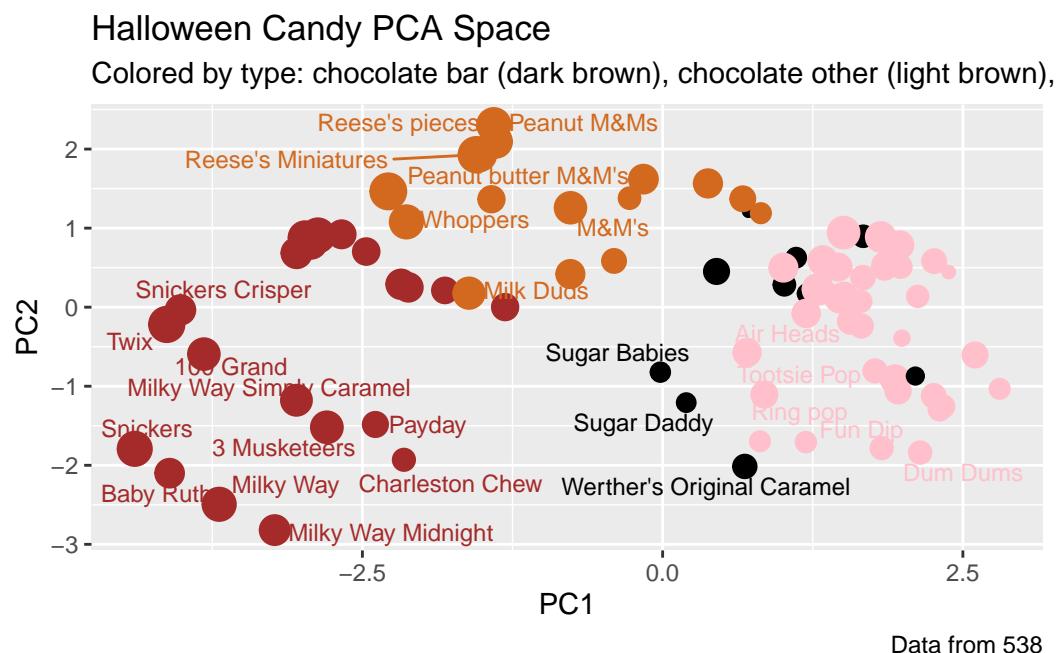
my_data <- cbind(candy, pca$x[,1:3])
p <- ggplot(my_data) +
  aes(x=PC1, y=PC2,
      size=winpercent/100,
      text=rownames(my_data),
      label=rownames(my_data)) +
  geom_point(col=my_cols)

```

```

p +
  geom_text_repel(size = 3.3, col = my_cols, 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"
  )

```



```

280
281  ````{r}
282  #/ label: interactive-plot
283  #/ eval: knitr:::is_html_output()
284  library(plotly)
285  plotly::ggplotly(p)
286  ````
```

