

Class 10: Halloween Mini-Project

Patricia Chen A16138722

```
#install.packages("webshot")
#webshot::install_phantomjs()
```

1. Importing candy data

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

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

	chocolate	fruity	caramel	peanut	almond	nougat	crisped	rice	wafer
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	sugar	percent	price	percent	win	percent
100 Grand	0	1	0		0.732		0.860	66.97	173
3 Musketeers	0	1	0		0.604		0.511	67.60	294
One dime	0	0	0		0.011		0.116	32.26	109
One quarter	0	0	0		0.011		0.511	46.11	650
Air Heads	0	0	0		0.906		0.511	52.34	146
Almond Joy	0	1	0		0.465		0.767	50.34	755

The functions `dim()`, `nrow()`, `table()` and `sum()` may be useful for answering the first 2 questions.

Q1. How many different candy types are in this dataset?

```
ncol(candy) # two ways
```

```
[1] 12
```

```
#dim(candy)
```

Answer: There are 12 different types of candy in the dataset.

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

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

```
[1] 38
```

Answer: There are 38 types of fruity candy type in the dataset.

2. What is your favorite candy?

```
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"	

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

```
[1] 81.64291
```

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

```
candy["Almond Joy", ]$winpercent
```

```
[1] 50.34755
```

Answer: My favorite candy in the data set is Almond Joy, and the winpercent value for it is 50.34755 percent.

Q4. What is the winpercent value for "Kit Kat"?

```
candy["Kit Kat", ]$winpercent
```

```
[1] 76.7686
```

Answer: The winpercent value for “Kit Kat” is 76.7686 percent.

Q5. What is the winpercent value for “Tootsie Roll Snack Bars”?

```
candy["Tootsie Roll Snack Bars", ]$winpercent
```

```
[1] 49.6535
```

Answer: The winpercent value for “Tootsie Roll Snack Bars” is 49.6535 percent.

Side-note: the `skimr::skim()` function

There is a useful `skim()` function in the `skimr` package that can help give you a quick overview of a given dataset. Let’s install this package and try it on our candy data.

```
#install.packages("skimr")
```

```
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	ratio	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		

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
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	

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

Answer: The columns that looks to be on a different scale to the majority of columns are the sugarpercent, pricepercent, and winpercent, which are in decimal values between 0 and 1. While the majority of other columns have categorical values that are exclusively either zero or one values.

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

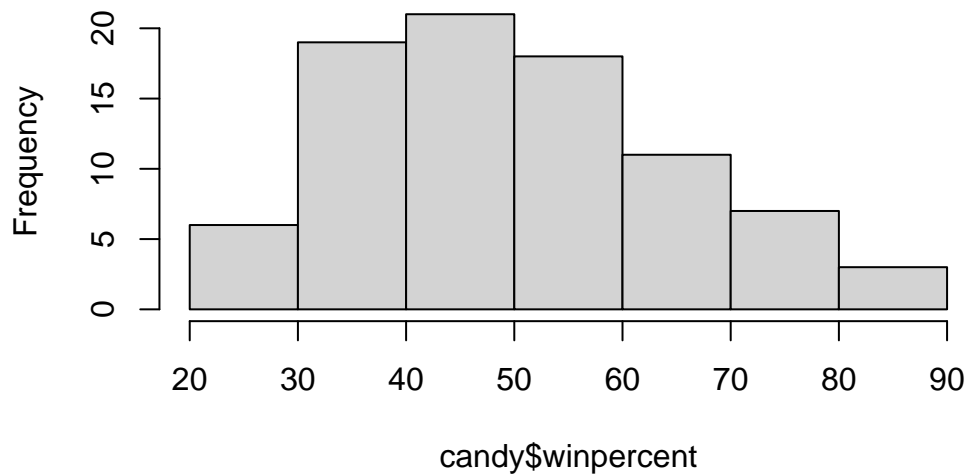
Answer: For the candy\$chocolate column, a zero means the absence of chocolate in the candy, and one represents the presence of chocolate in the candy.

Hint: look at the “Variable type” print out from the skim() function. Most variables (i.e. columns) are on the zero to one scale but not all. Some columns such as chocolate are exclusively either zero or one values.

Q8. Plot a histogram of winpercent values

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

Histogram of candy\$winpercent



Q9. Is the distribution of winpercent values symmetrical?

Answer: The distribution of winpercent values are not symmetrical and is skewed to the right (tail of the plot on the right side).

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

Answer: The center of the distribution is below 50%.

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

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

mean(choco) > mean(fruity)
```

[1] TRUE

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

```
t.test(choco, fruity)
```

Welch Two Sample t-test

```
data: choco and fruity
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
```

Q12. Is this difference statistically significant?

Answer: According to the `t.test` function, the difference is statistically significant given a p-value of 2.87e-08, which is smaller than 0.05.

3. 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`

```
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	
	crisped	ricewafer	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					

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

	chocolate	fruity	caramel	peanuty	almondy	nougat
Nik L Nip	0	1	0	0	0	0
Boston Baked Beans	0	0	0	1	0	0
Chiclets	0	1	0	0	0	0
Super Bubble	0	1	0	0	0	0
Jawbusters	0	1	0	0	0	0
	crisped	ricewafer	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					

Answer: From the dplyr function, the five least liked candies are Nik L Nip, Boston Baked Beans, Chiclets, Super Bubble, and Jawbusters. I used the dplyr approach given the yielded results are more convenient to view and easier to understand.

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


```
candy %>% arrange(winpercent) %>% tail(5)
```

	chocolate	fruity	caramel	peanut	almond	nougat
Snickers	1	0	1		1	1
Kit Kat	1	0	0		0	0
Twix	1	0	1		0	0
Reese's Miniatures	1	0	0		1	0
Reese's Peanut Butter cup	1	0	0		1	0

	crisped	rice	wafer	hard	bar	pluribus	sugar	percent
Snickers			0	0	1	0		0.546
Kit Kat			1	0	1	0		0.313
Twix			1	0	1	0		0.546
Reese's Miniatures			0	0	0	0		0.034
Reese's Peanut Butter cup			0	0	0	0		0.720

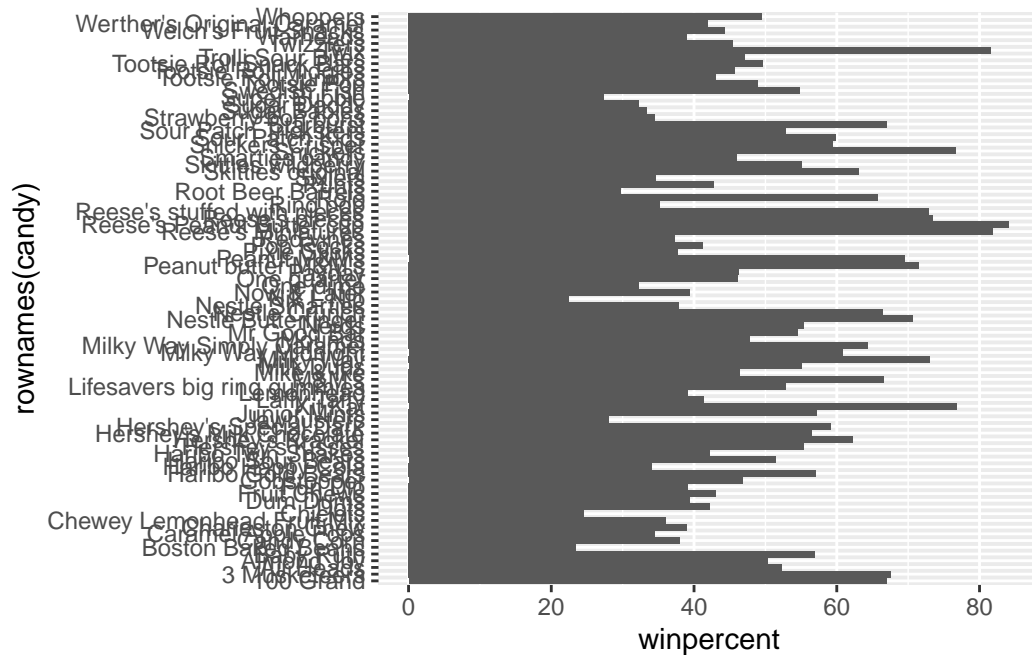
	price	percent	winpercent
Snickers	0.651	76.67378	
Kit Kat	0.511	76.76860	
Twix	0.906	81.64291	
Reese's Miniatures	0.279	81.86626	
Reese's Peanut Butter cup	0.651	84.18029	

Answer: The top five all time favorite candy types are Snickers, Kit Kat, Twix, Reese's Miniatures, and Reese's Peanut Butter cup.

Q15. Make a first barplot of candy ranking based on winpercent values. HINT: Use the `aes(winpercent, rownames(candy))` for your first ggplot like so:

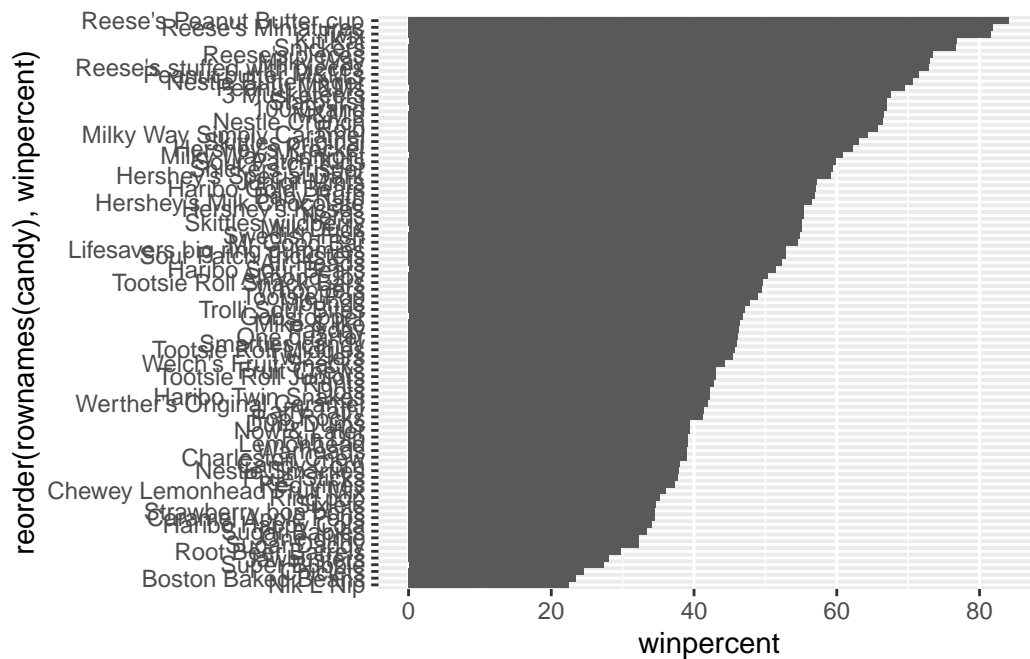
```
library(ggplot2)

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



Q16. This is quite ugly, use the `reorder()` function to get the bars sorted by winpercent? HINT: You can use `aes(winpercent, reorder(rownames(candy), winpercent))` to improve your plot.

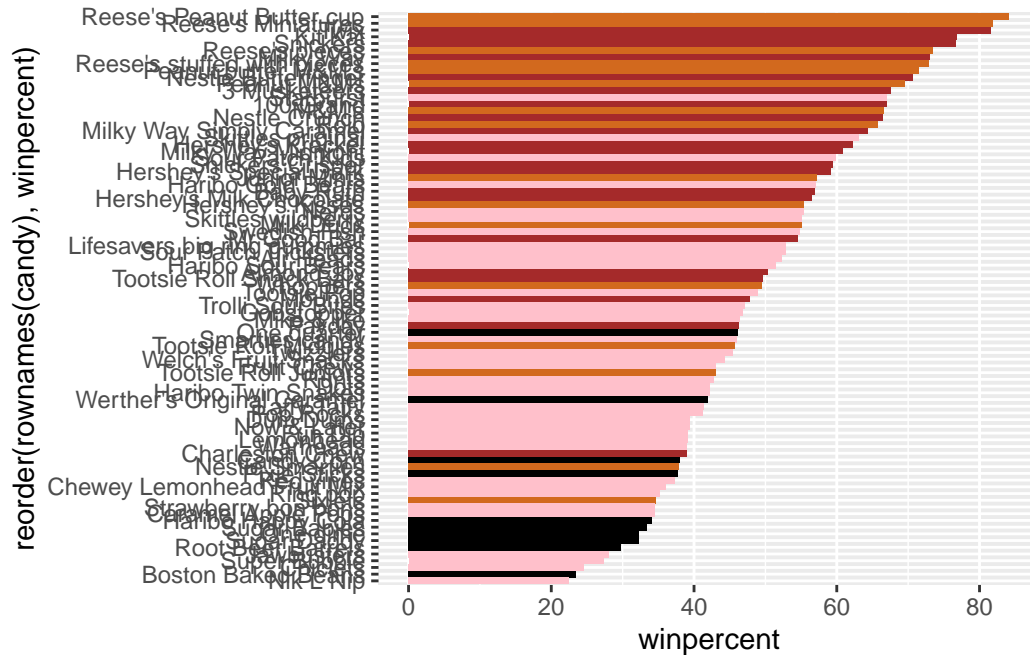
```
ggplot(candy) +
  aes(winpercent, reorder(rownames(candy), winpercent)) +
  geom_col()
```



Time to add some useful color

```
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?

Answer: According to the plot, the worst ranked chocolate candy is Sixlets.

Q18. What is the best ranked fruity candy?

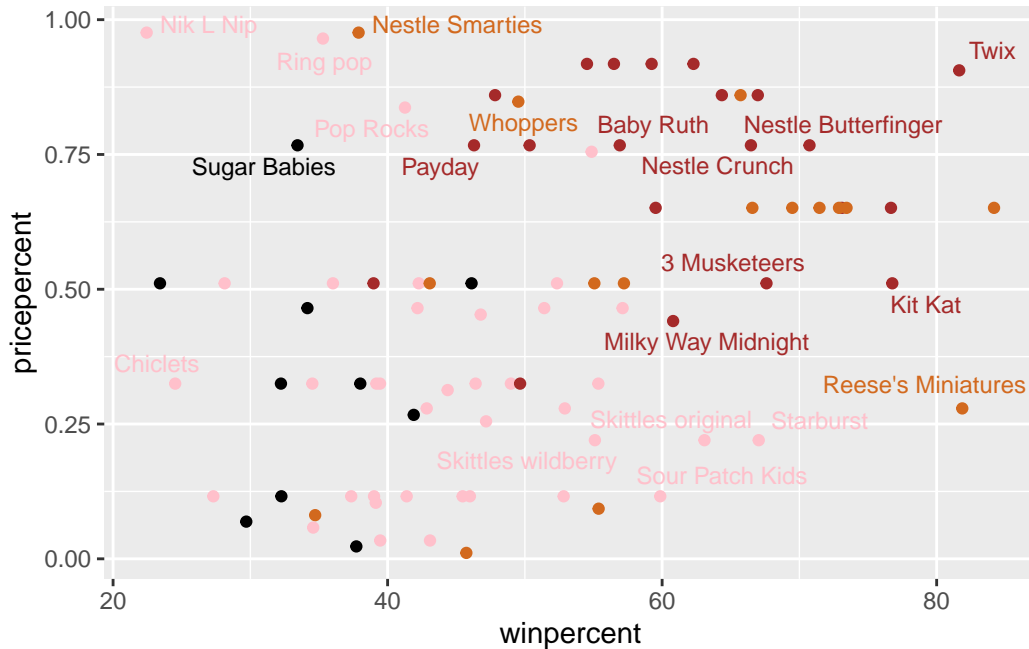
Answer: According to the plot, the best ranked fruity candy is Starbust.

4. Taking a look at pricepercent

```
library(ggrepel)

# How about a plot of price vs win
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=my_cols) +
  geom_text_repel(col=my_cols, size=3.3, max.overlaps = 5)
```

Warning: ggrepel: 65 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?

```
ord <- order(candy$pricepercent, decreasing = FALSE)
head( candy[ord,c(11,12)], n=5 )
```

	pricepercent	winpercent
Tootsie Roll Midgies	0.011	45.73675
Pixie Sticks	0.023	37.72234
Dum Dums	0.034	39.46056
Fruit Chews	0.034	43.08892
Strawberry bon bons	0.058	34.57899

Answer: The Tootsie Roll Midgies is the highest ranked candy type 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? Hint: To see which candy is the most expensive (and which is the least expensive) we can order() the dataset by pricepercent.

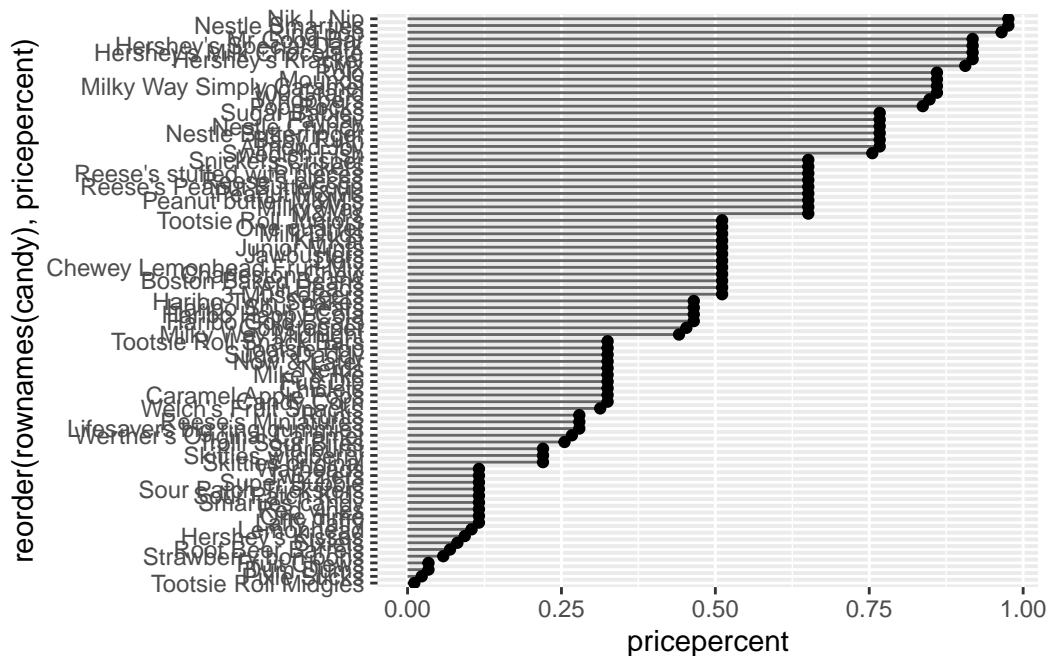
```
ord <- order(candy$pricepercent, decreasing = TRUE)
head( candy[ord,c(11,12)], n=5 )
```

	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

Answer: The top 5 most expensive candy types in the dataset are Nik L Nip, Nestle smarties, Ring pop, Hershey's Krackel, and Hershey's Milk Chocolate. Nik L Nip is the least popular among them.

Optional. 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()`.

```
# Make a lollipop chart of pricepercent
ggplot(candy) +
  aes(pricepercent, reorder(rownames(candy), pricepercent)) +
  geom_segment(aes(yend = reorder(rownames(candy), pricepercent),
                    xend = 0), col="gray40") +
  geom_point()
```



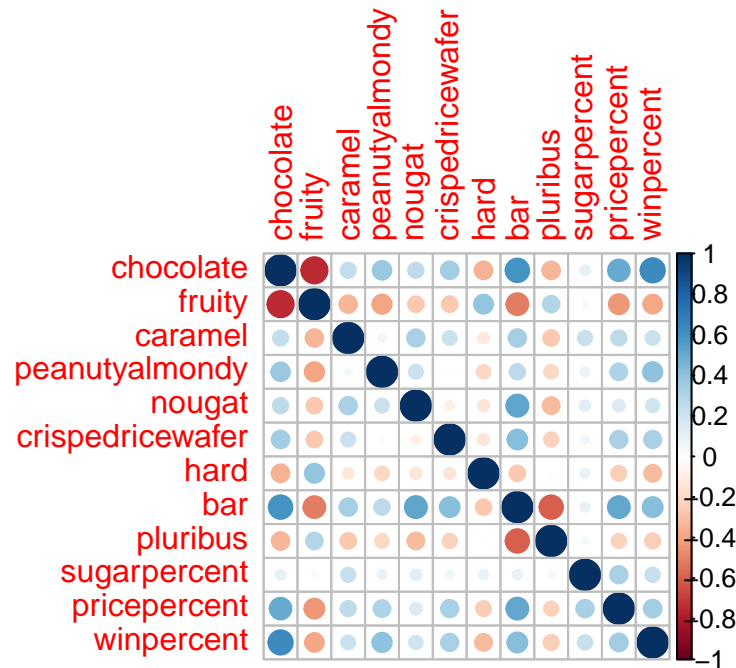
One of the most interesting aspects of this chart is that a lot of the candies share the same ranking, so it looks like quite a few of them are the same price.

5 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)?

Answer: The fruity and chocolate variables are anti-correlated.

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

Answer: The two pairs of variables bar and chocolate; and winpercent and chocolate are positively correlated.

6. Principal Component Analysis

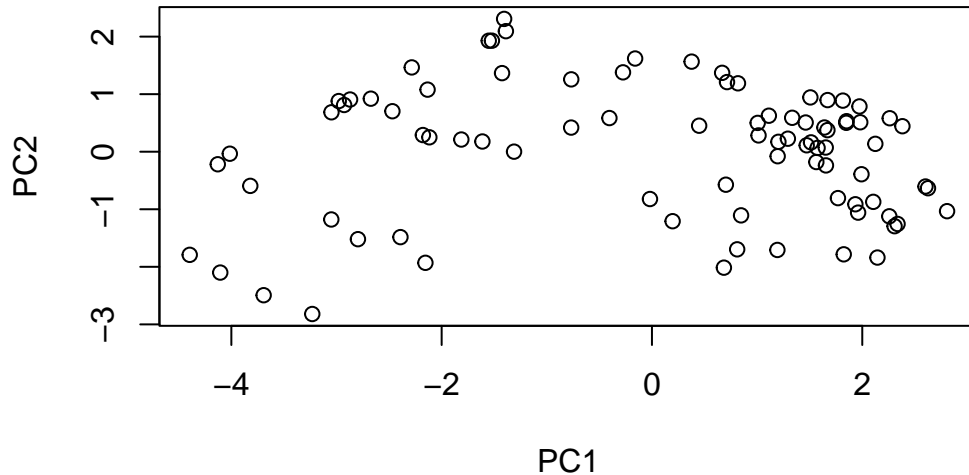
```
pca <- prcomp(candy, scale=TRUE)
summary(pca)
```

Importance of components:

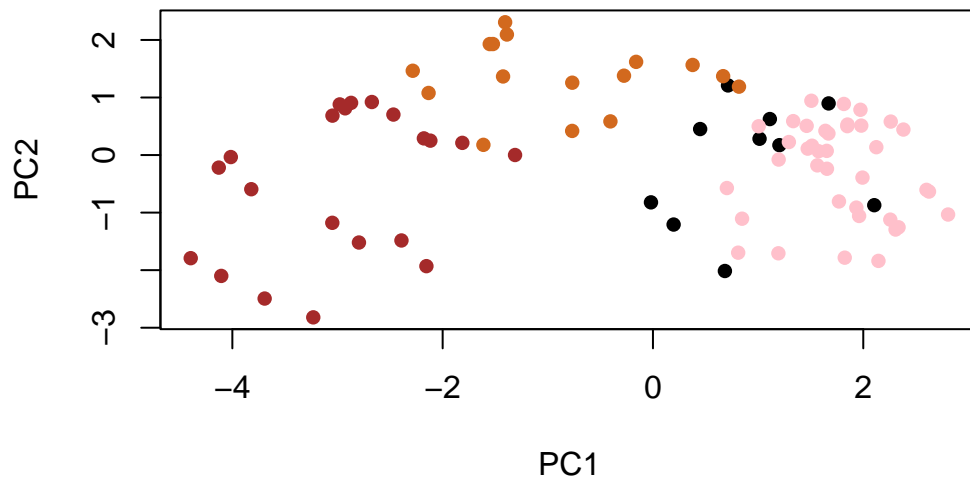
	PC1	PC2	PC3	PC4	PC5	PC6	PC7
Standard deviation	2.0788	1.1378	1.1092	1.07533	0.9518	0.81923	0.81530
Proportion of Variance	0.3601	0.1079	0.1025	0.09636	0.0755	0.05593	0.05539
Cumulative Proportion	0.3601	0.4680	0.5705	0.66688	0.7424	0.79830	0.85369

	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], xlab="PC1", ylab="PC2")
```



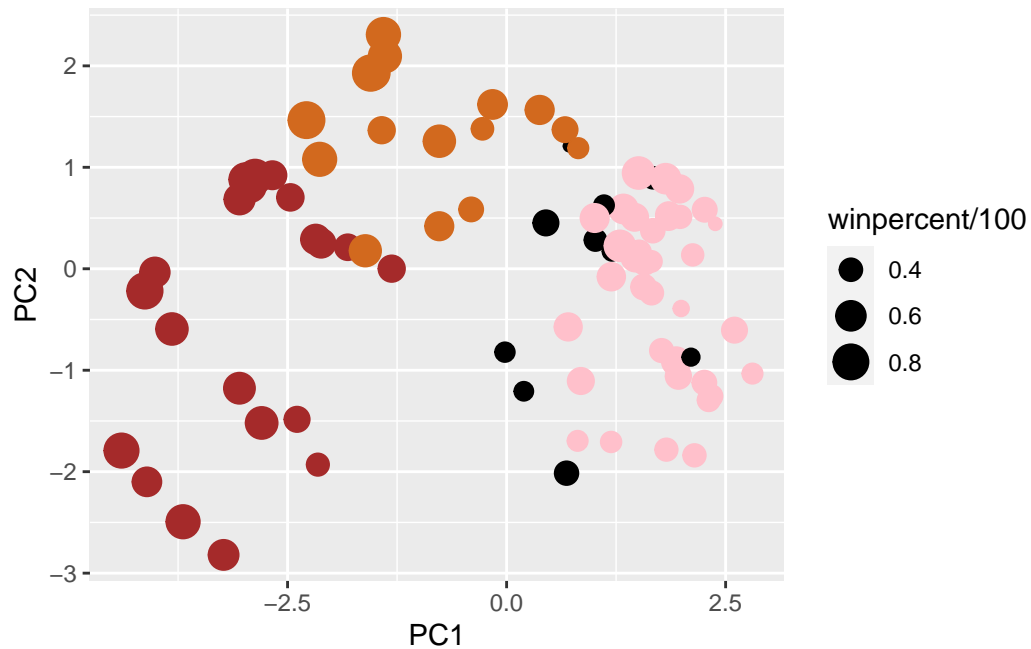
```
plot(pca$x[,1:2], col=my_cols, pch=16)
```

```
# Make a new data-frame with our PCA results and candy data  
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
```



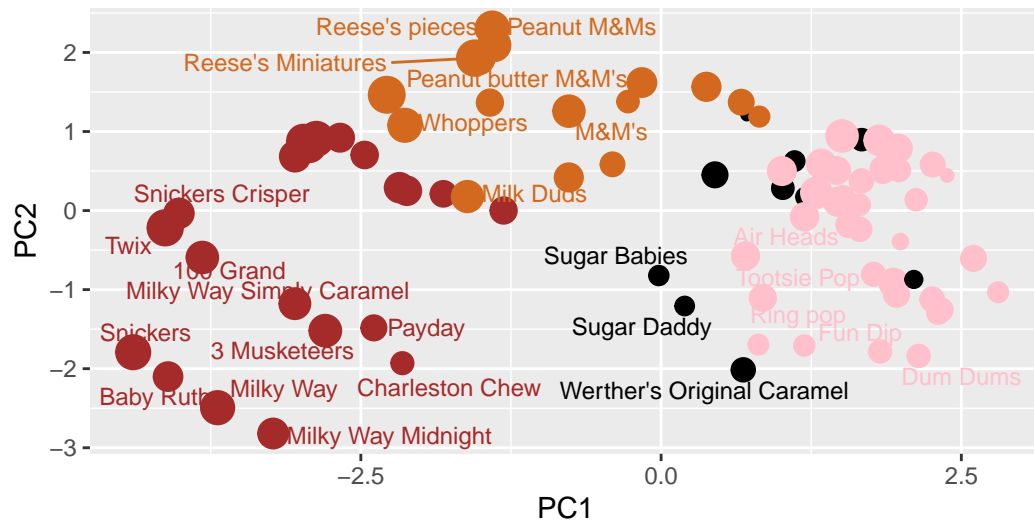
```
library(ggrepel)

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")
```

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

Halloween Candy PCA Space

Colored by type: chocolate bar (dark brown), chocolate other (light brown),



Data from 538

```
library(plotly)
```

Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

last_plot

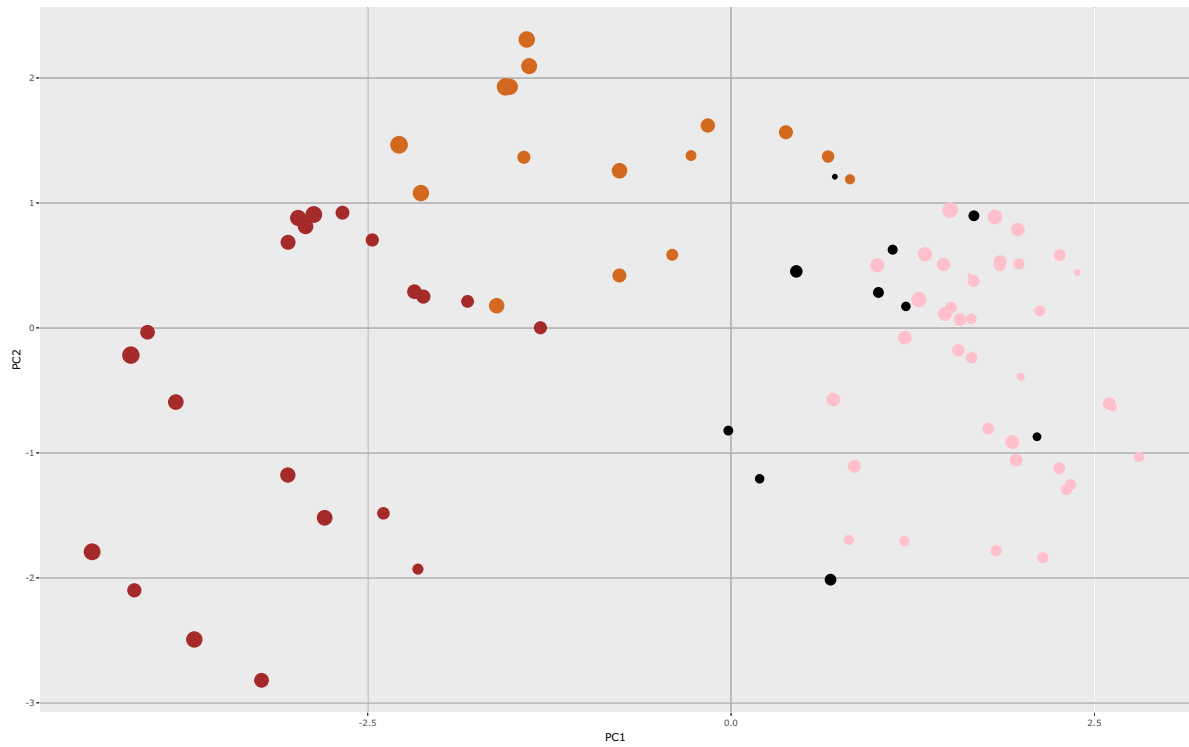
The following object is masked from 'package:stats':

filter

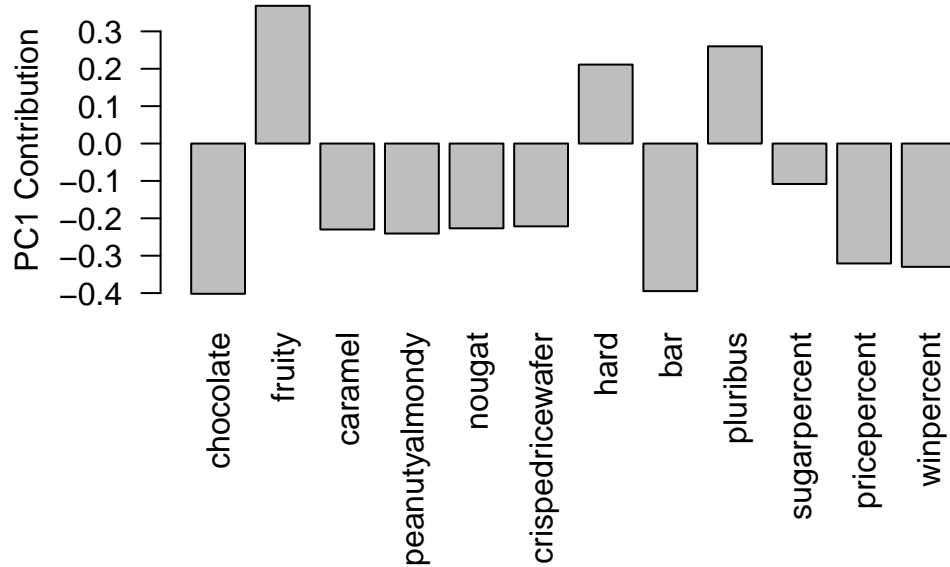
The following object is masked from 'package:graphics':

layout

```
ggplotly(p)
```



```
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? HINT. pluribus means the candy comes in a bag or box of multiple candies.

Answer: The original variables picked up strongly by PC1 in the positive direction are fruity, hard, and pluribus. These do make sense since fruity and hard candies usually do not contain chocolate. And other types of candies such as caramel and nougat are not hard candies and comes in individual packages rather than as pluribus in bag or box.