

Class 10: Halloween Mini-Project

Sophia Wang (A16838155)

```
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
```

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

```
[1] 81.64291
```

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

Sour patch kids.

```
candy["Sour Patch Kids","winpercent"]
```

```
[1] 59.864
```

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)=="Tootsie Roll Snack Bars")|>
select(winpercent)
```

```

              winpercent
Tootsie Roll Snack Bars 49.6535
```

Q. Find fruity candy with a winpercent above 50%

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

	chocolate	fruity	caramel	peanut	almond	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

	crisped	rice	wafer	hard	bar	pluribus	sugar
Air Heads				0	0	0	0.906
Haribo Gold Bears				0	0	0	0.465
Haribo Sour Bears				0	0	0	0.465
Lifesavers big ring gummies				0	0	0	0.267
Nerds				0	1	0	0.848
Skittles original				0	0	0	0.941
Skittles wildberry				0	0	0	0.941
Sour Patch Kids				0	0	0	0.069
Sour Patch Tricksters				0	0	0	0.069
Starburst				0	0	0	0.151
Swedish Fish				0	0	0	0.604

	price	percent	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

```
top.candy <- candy[candy$winpercent>50,]
top.candy[top.candy$fruity==1,]
```

	chocolate	fruity	caramel	peanut	almondy	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

	crisped	rice	wafer	hard	bar	pluribus	sugar	percent
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

	price	percent	win	percent
Air Heads	0.511		52.34	146
Haribo Gold Bears	0.465		57.11	974
Haribo Sour Bears	0.465		51.41	243
Lifesavers big ring gummies	0.279		52.91	139
Nerds	0.325		55.35	405
Skittles original	0.220		63.08	514
Skittles wildberry	0.220		55.10	370
Sour Patch Kids	0.116		59.86	400

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 folks like using the `skier` package and its `skim()` function

```
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	

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.

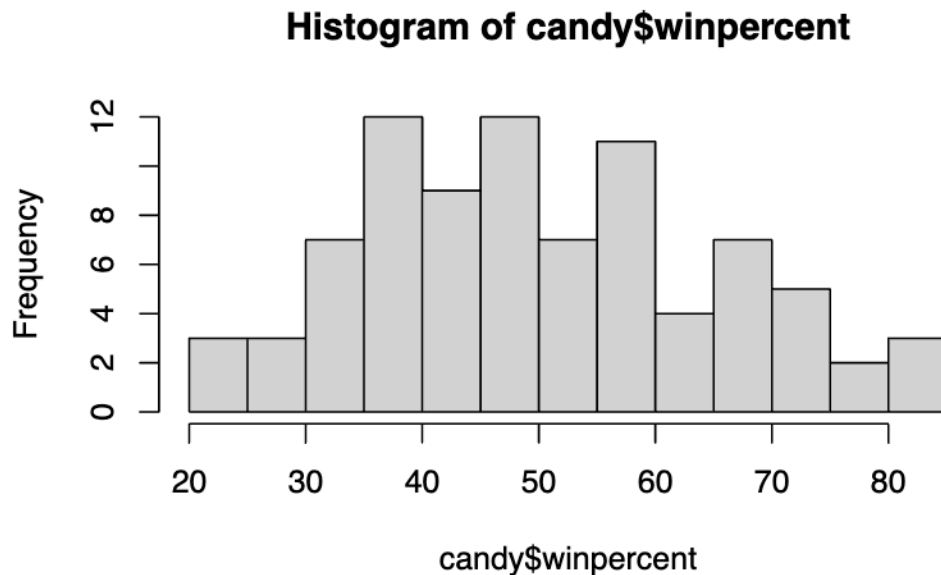
Q7. What do you think a zero and one represent for the `candy$chocolate` column?

A one means “yes that is a chocolate candy” and zero means “no that is not a chocolate candy”.

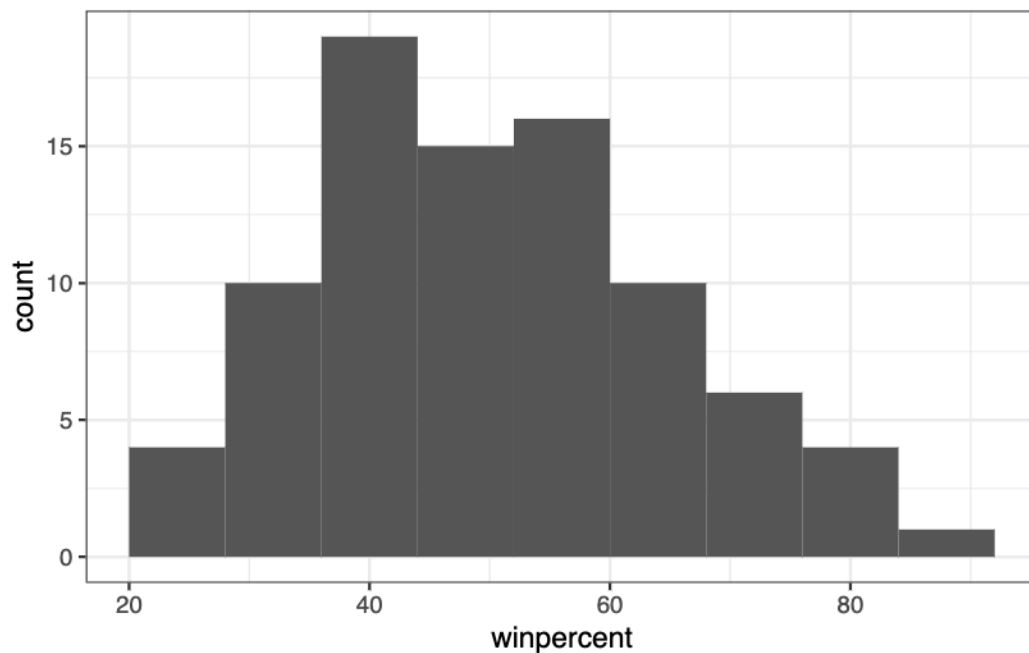
Q8. Plot a histogram of `winpercent` values

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_bw()
```



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

Below 50%

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

```
#choco.candy <- candy[as.logical(candy$chocolate),]$winpercent
choco.candy <- candy|>
  filter(chocolate==1)
summary(choco.candy$winpercent)
```

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

On average chocolate candy is higher ranked than fruit candy

Q12. Is this difference statistically significant?

```
t.test(choco.candy$winpercent,fruit.candy$winpercnet)
```

One Sample t-test

```
data: choco.candy$winpercent
t = 28.926, df = 36, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 56.65009 65.19297
sample estimates:
mean of x
 60.92153
```

Yes, the difference is significant.

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

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

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

```
order(play)
```

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



```
play[order(play)]
```

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

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

	chocolate	fruity	caramel	peanut	almond	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	rice	wafer	hard	bar	pluribus	sugar	percent
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,decreasing=T),],5)
```

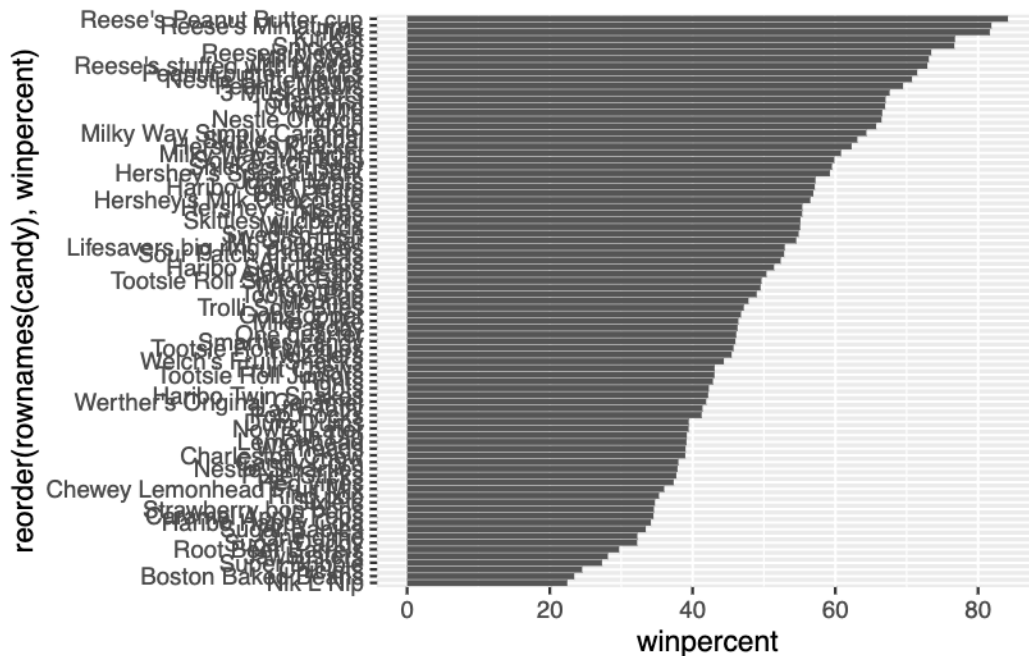
	chocolate	fruity	caramel	peanut	almond	nougat		
Reese's Peanut Butter cup	1	0	0		1	0		
Reese's Miniatures	1	0	0		1	0		
Twix	1	0	1		0	0		
Kit Kat	1	0	0		0	0		
Snickers	1	0	1		1	1		
	crisped	rice	wafer	hard	bar	pluribus	sugar	percent
Reese's Peanut Butter cup			0	0	0	0	0.720	
Reese's Miniatures			0	0	0	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			

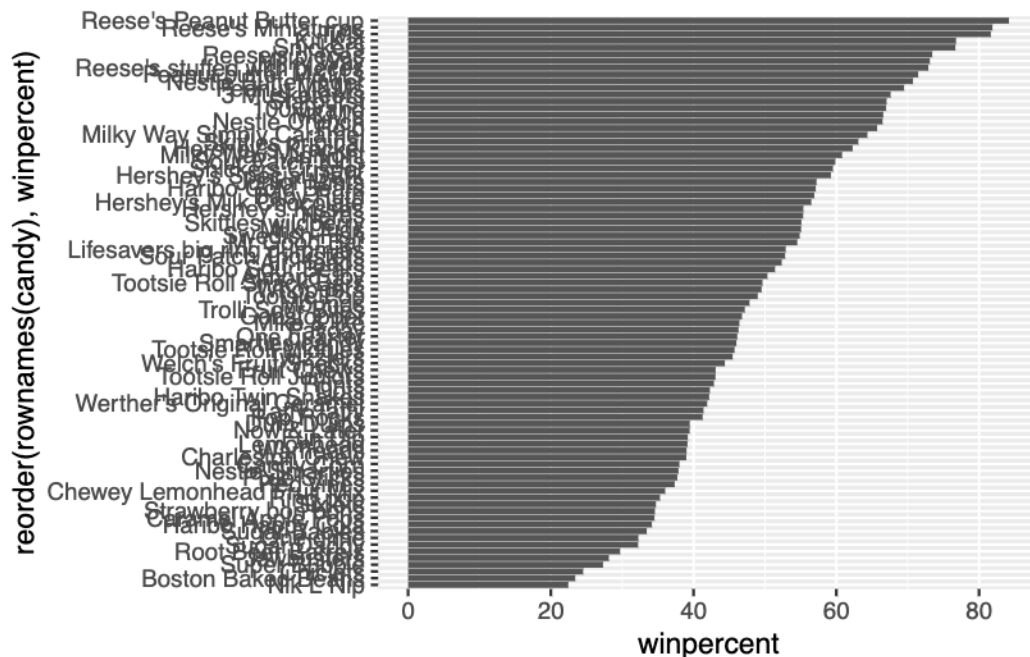
Q15. Make a first barplot of candy ranking based on winpercent values. Q16. This is quite ugly, use the `reorder()` function to get the bars sorted by winpercent?

Let's do a barplot of winpercent values

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



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



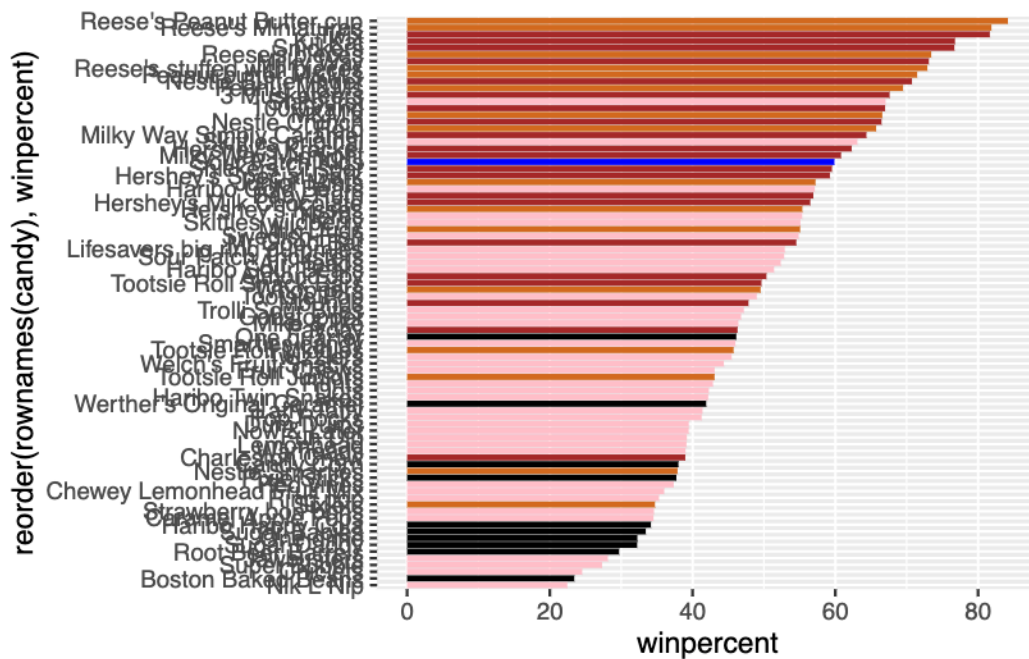
I want a more custom color scheme where I can see both chocolate and bar and fruity etc. all from the one plot. To do this we can roll our own color vector...

```
#Place holder 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)=="Sour Patch Kids"] <- "blue"
mycols
```

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

```
[73] "pink"      "pink"      "pink"      "chocolate" "chocolate" "brown"
[79] "pink"      "brown"     "pink"      "pink"      "pink"      "black"
[85] "chocolate"
```

```
ggplot(candy) +
  aes(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?

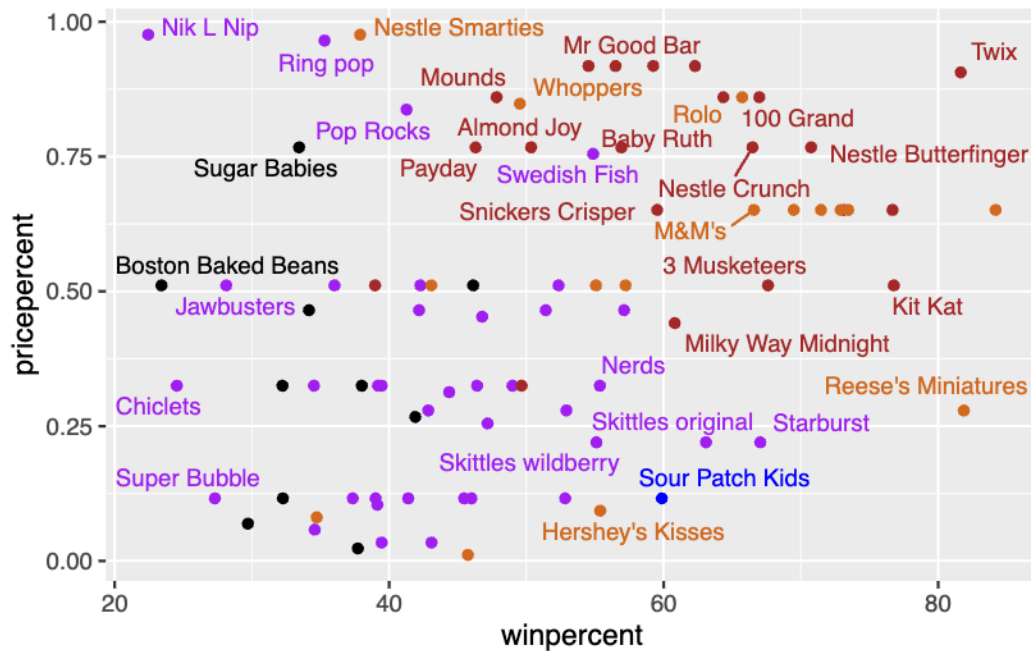
Starburst

```
mycols[as.logical(candy$fruity)]<-"purple"
mycols[row.names(candy)=="Sour Patch Kids"] <- "blue"
```

Add label,ggrepel

```
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?

```
tail(candy[order(candy$pricepercent),0],5)
```

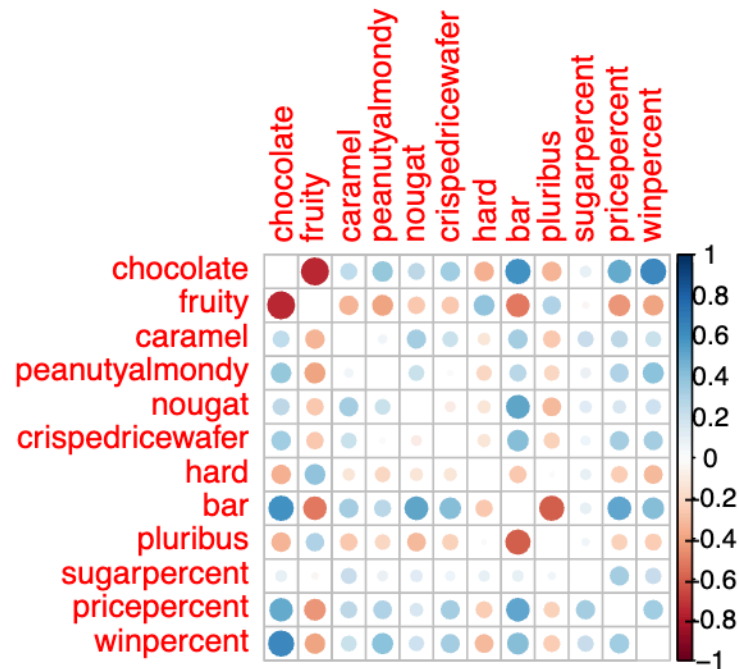
data frame with 0 columns and 5 rows

Hershey's Special Dark, Mr Good Bar, Ring pop, Nik L Nip, Nestle Smarties. The least popular one is Nik L Nip.

```
library(corrplot)
```

corrplot 0.95 loaded

```
cij <- cor(candy)  
corrplot(cij,diag=F)
```



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

Chocolate and fruity.

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

Chocolate and winpercent.

```
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		

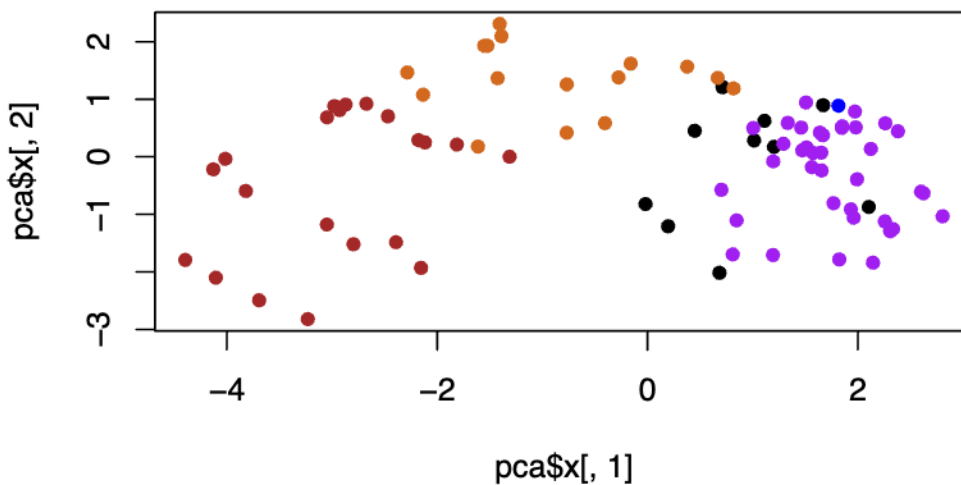
```
pca <- prcomp(candy, scale=T)
summary(pca)
```

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

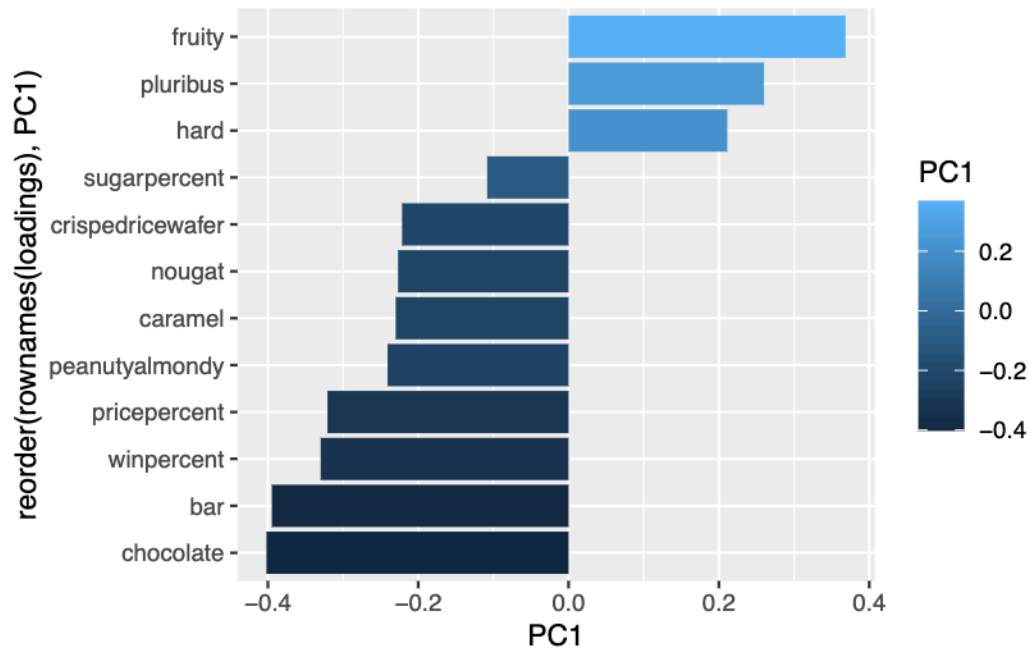
	PC8	PC9	PC10	PC11	PC12
Standard deviation	0.74530	0.67824	0.62349	0.43974	0.39760
Proportion of Variance	0.04629	0.03833	0.03239	0.01611	0.01317
Cumulative Proportion	0.89998	0.93832	0.97071	0.98683	1.00000

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



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

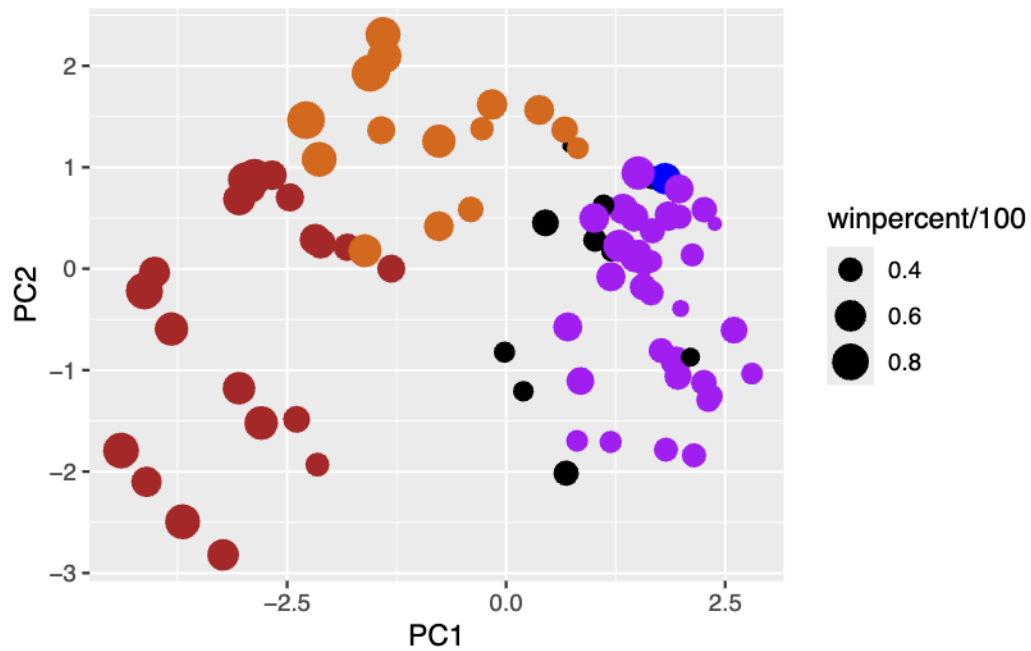
```
loadings <- as.data.frame(pca$rotation)
ggplot(loadings)+
  aes(PC1,reorder(rownames(loadings),PC1),fill=PC1)+
  geom_col()
```

```
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)#+
  #geom_text()

p
```

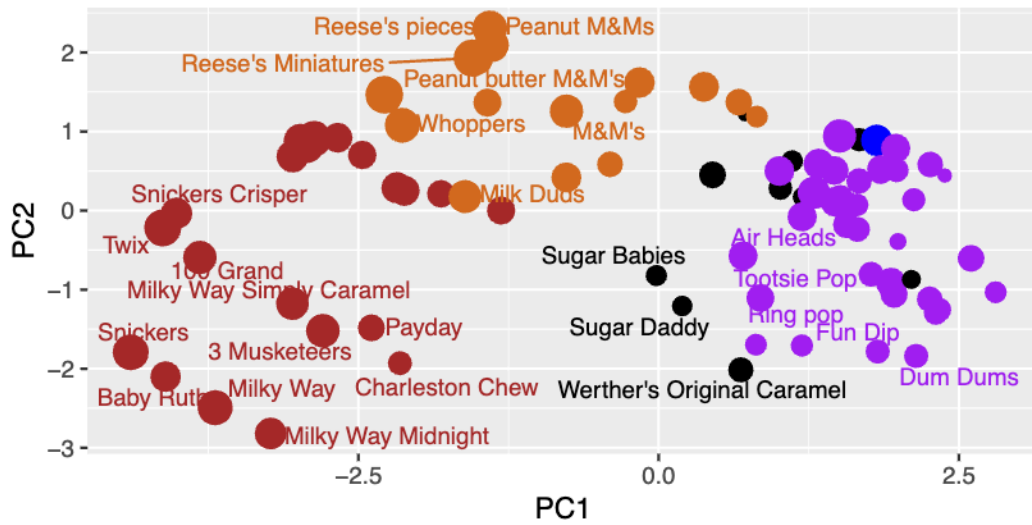


```
p + geom_text_repel(size=3.3, col=mycols, max.overlaps = 7) +
  theme(legend.position = "none") +
  labs(title="Halloween Candy PCA Space",
        subtitle="Colored by type: chocolate bar (dark brown), chocolate other (light brown),",
        caption="Data from 538")
```

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

Halloween Candy PCA Space

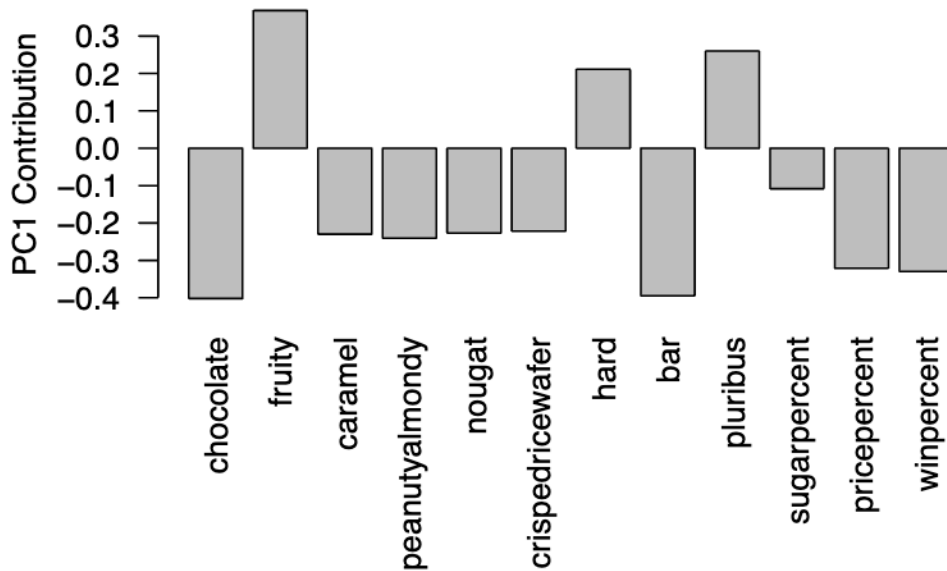
Colored by type: chocolate bar (dark brown), chocolate other (light brown),



Data from 538

```
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?

Fruity, hard, and pluribus are picked up strongly by PC1 in the positive direction. Yes these make sense because these three variables are shown to be correlated in the corrplot.

```
pca$rotation[,1]
```

chocolate	fruity	caramel	peanutyalmondy
-0.4019466	0.3683883	-0.2299709	-0.2407155
nougat	crispedricewafer	hard	bar
-0.2268102	-0.2215182	0.2111587	-0.3947433
pluribus	sugarpercent	pricepercent	winpercent
0.2600041	-0.1083088	-0.3207361	-0.3298035