Class10

Arman Farahani (PID A17497672)

Today is Halloween, an ole 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.name=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
```

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

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

```
[1] 38
```

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

```
candy["Haribo Happy Cola", "winpercent"]
```

```
[1] 34.15896
```

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
```

```
top.candy <- candy[candy$winpercent > 50,]
top.candy[top.candy$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 folks like using the skimer package and its `skim()` function.

```
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

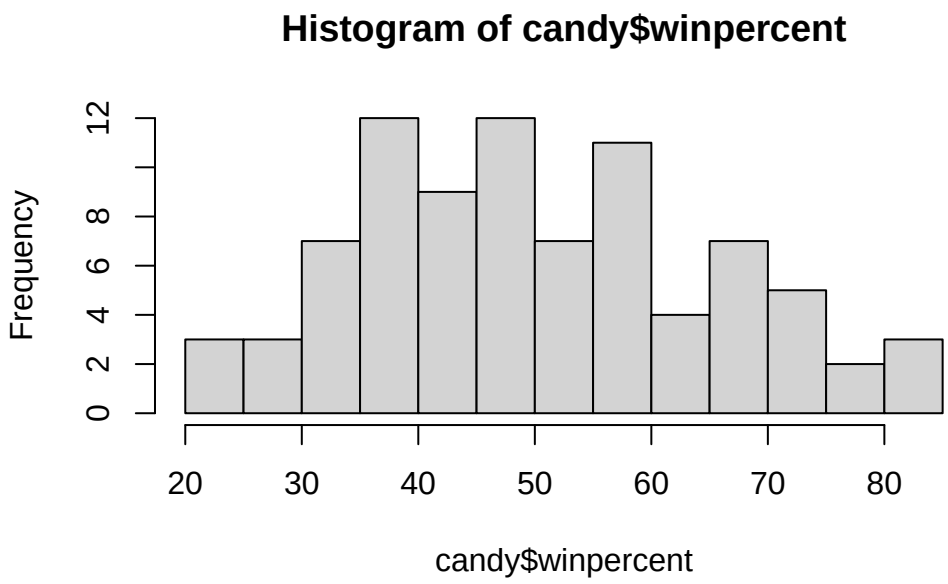
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
chocolate	0	1	0.44	0.50	0.00	0.00	0.00	1.00	1.00	
fruity	0	1	0.45	0.50	0.00	0.00	0.00	1.00	1.00	
caramel	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
peanutyalmondy	0	1	0.16	0.37	0.00	0.00	0.00	0.00	1.00	
nougat	0	1	0.08	0.28	0.00	0.00	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	

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.

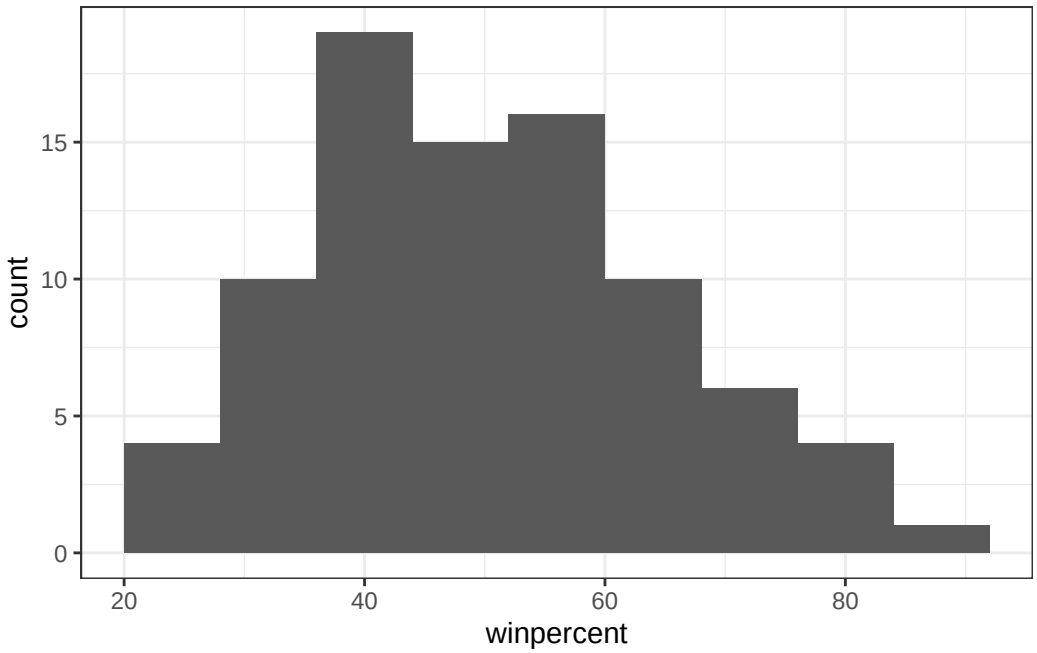
Q8. Plot a histogram of winpercent values

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 
```

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)
```

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

```
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
```

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?

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

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

```
order(play)
```

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

```
head(play[ order(play)], 5)
```

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

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

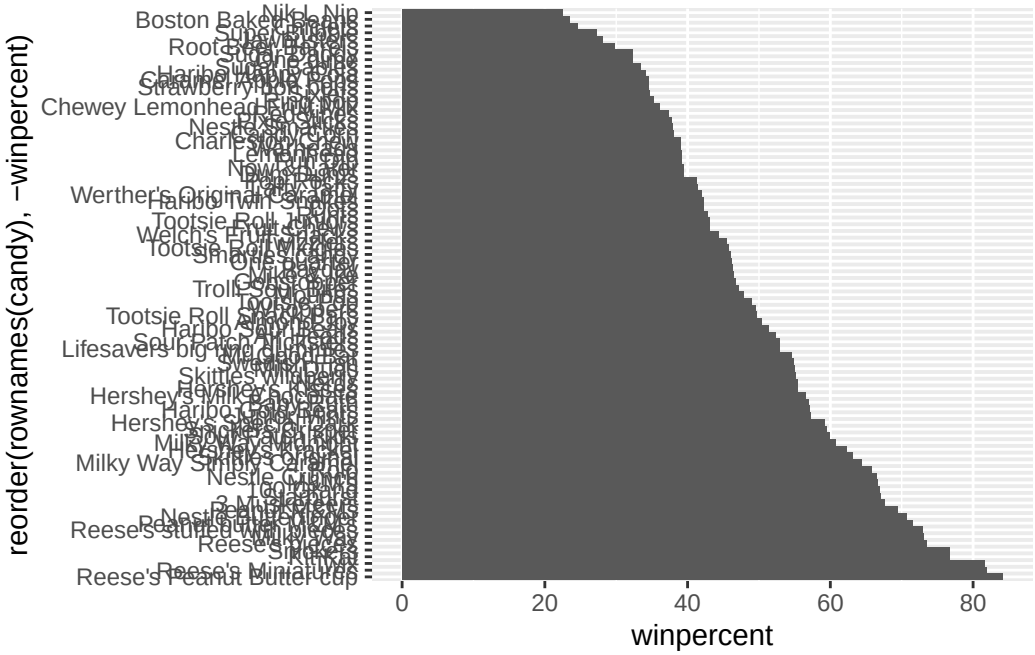
```
                  chocolate fruity caramel peanutyalmondy nougat
Nik L Nip                 0      1       0              0      0
Boston Baked Beans        0      0       0              1      0
Chiclets                  0      1       0              0      0
Super Bubble              0      1       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
```

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

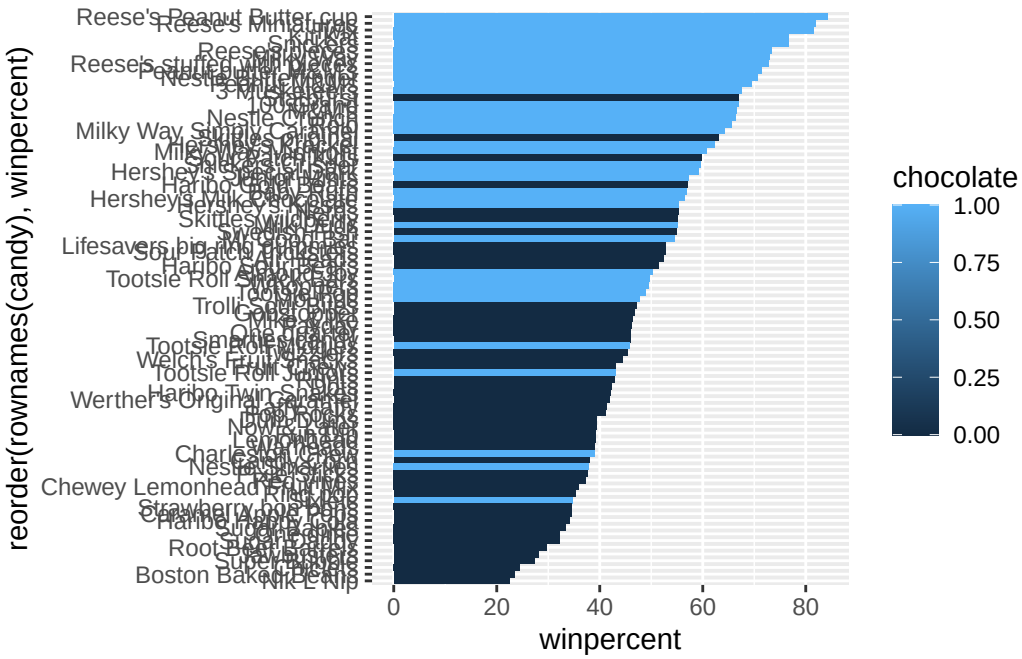
```
[1] 10  5  2
```

Let's do a barplot of winpercent values

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




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



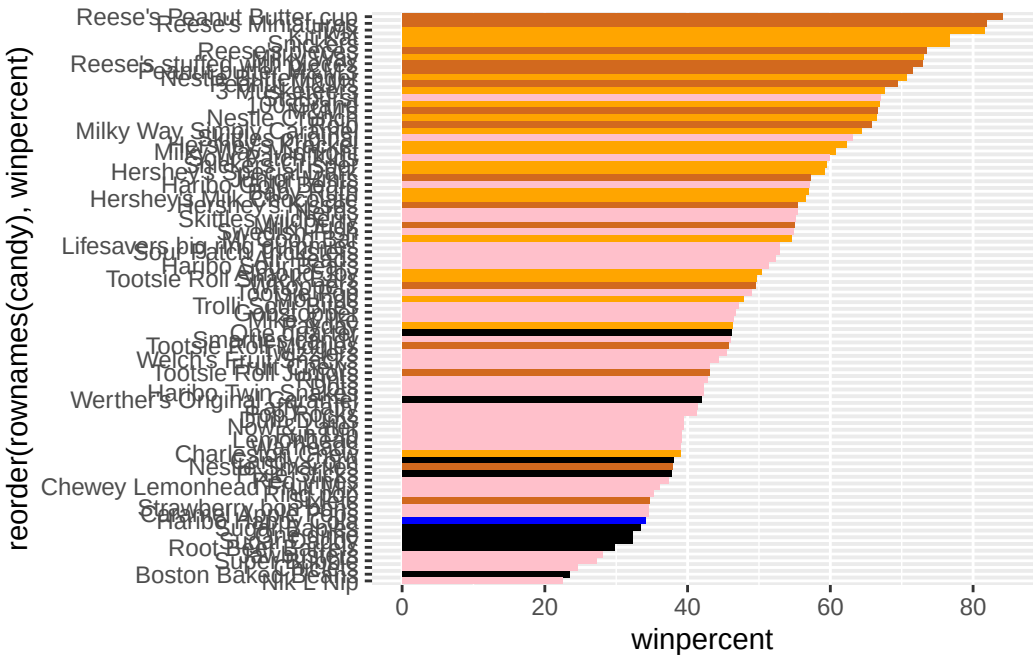
I want a more custom color scheme where I can see both chocolate and bar and fruity etc. all from the one plot. To do this we can roll our own color vector...

```
# Place holder color vector
mycols <- rep("black", nrow(candy))
mycols[rownames(candy) == "Haribo Happy Cola"] <- "blue"

mycols[as.logical(candy$chocolate)] <- "chocolate"
mycols[as.logical(candy$fruity)] <- "pink"
mycols[as.logical(candy$bar)] <- "orange"

#Use blue for your favorite candy!


mycols
```

```
 [1] "orange"    "orange"    "black"     "black"     "pink"      "orange"
 [7] "orange"    "black"     "black"     "pink"      "orange"    "pink"
[13] "pink"      "pink"      "pink"      "pink"      "pink"      "pink"
[19] "pink"      "blue"      "pink"      "pink"      "chocolate" "orange"
[25] "orange"    "orange"    "pink"      "chocolate" "orange"    "pink"
[31] "pink"      "pink"      "chocolate" "chocolate" "pink"      "chocolate"
[37] "orange"    "orange"    "orange"    "orange"    "orange"    "pink"
```

```
[43] "orange"    "orange"    "pink"      "pink"      "orange"    "chocolate"
[49] "black"     "pink"      "pink"      "chocolate" "chocolate" "chocolate"
[55] "chocolate" "pink"      "chocolate" "black"     "pink"      "chocolate"
[61] "pink"      "pink"      "chocolate" "pink"      "orange"    "orange"
[67] "pink"      "pink"      "pink"      "pink"      "black"     "black"
[73] "pink"      "pink"      "pink"      "chocolate" "chocolate" "orange"
[79] "pink"      "orange"    "pink"      "pink"      "pink"      "black"
[85] "chocolate"
```

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



Q17. What is the worst ranked chocolate candy?

Boston Baked Beans

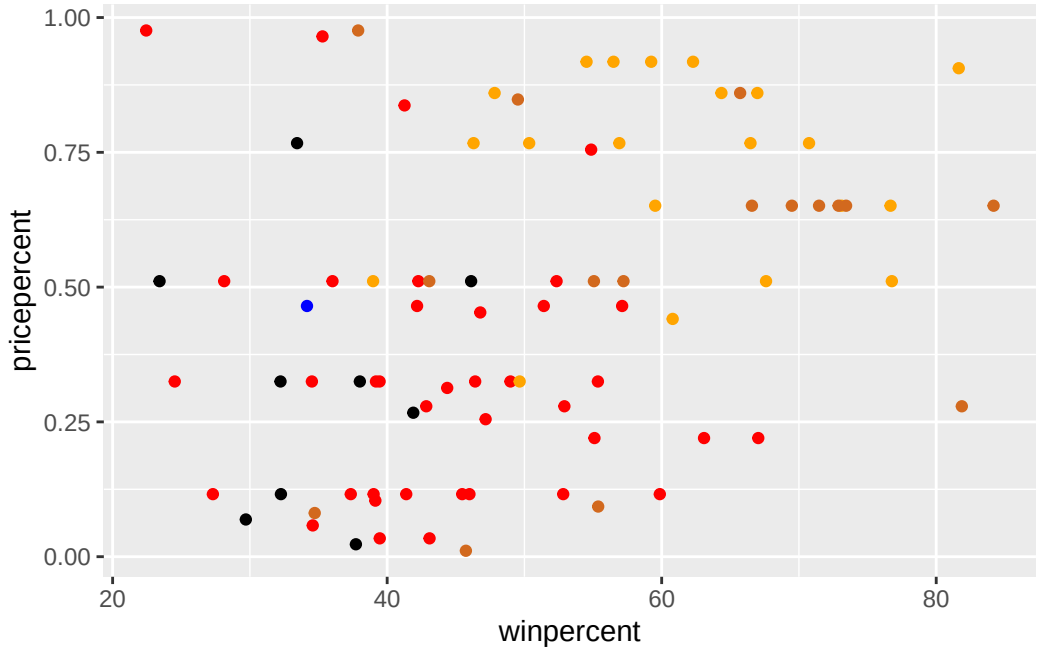
Q18. What is the best ranked fruity candy?

Reeses Miniatures

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

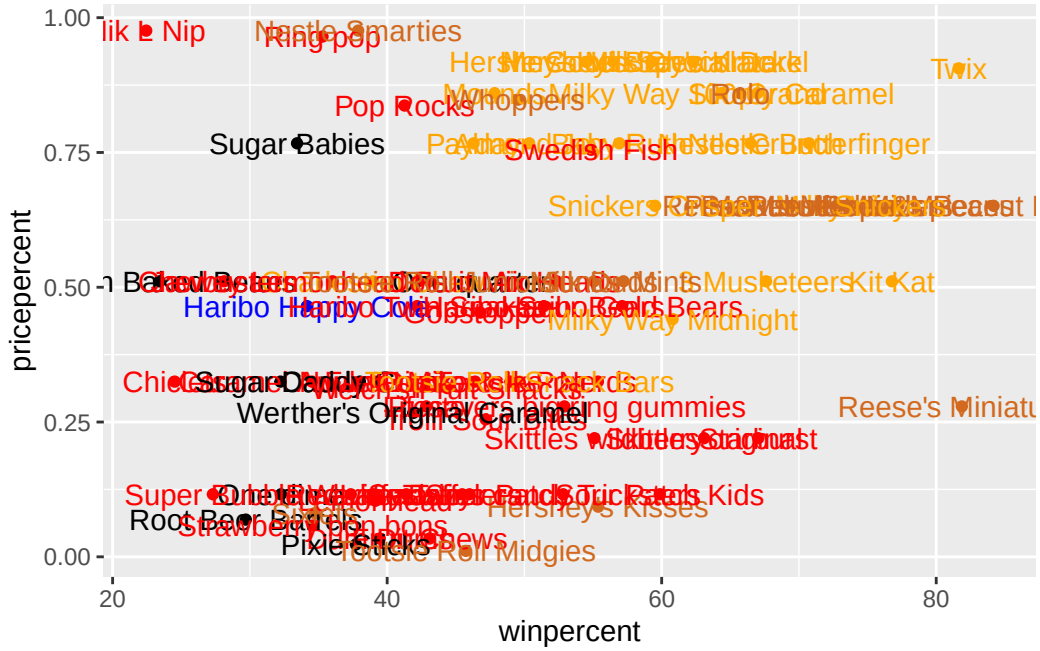
```
mycols[as.logical(candy$fruity)] <- "red"
```

```
ggplot(candy) +
  aes(winpercent, pricepercent) +
  geom_point(col=mycols)
```



Add labels

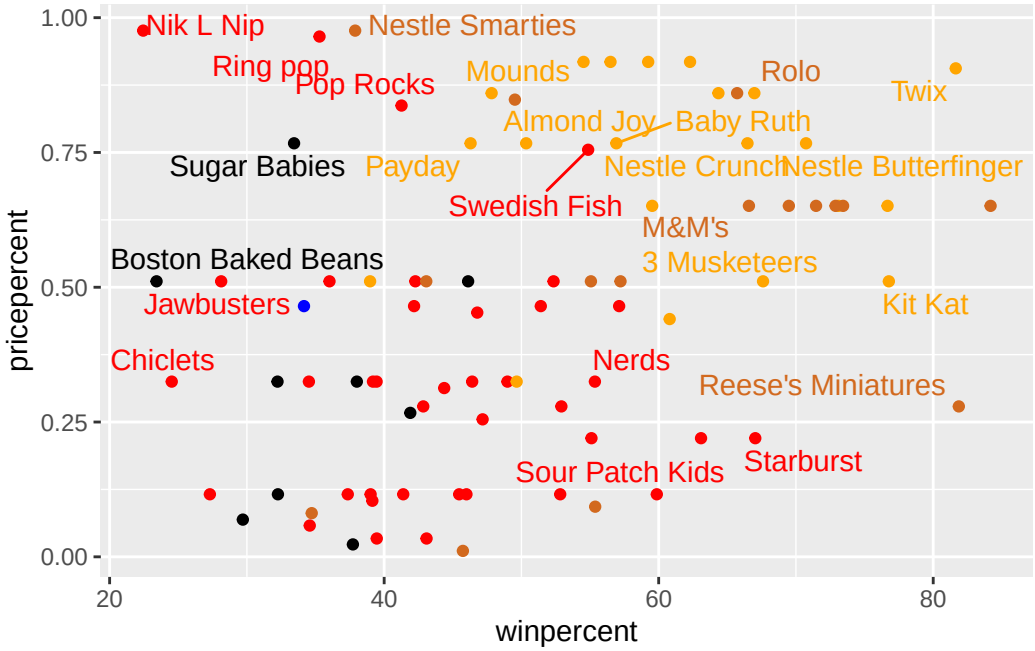
```
ggplot(candy) +
  aes(winpercent, pricepercent, label=rownames(candy)) +
  geom_point(col=mycols) +
  geom_text(col=mycols)
```



Make the labels non-overlapping

```
library(ggrepel)

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

```
Warning: ggrepel: 61 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

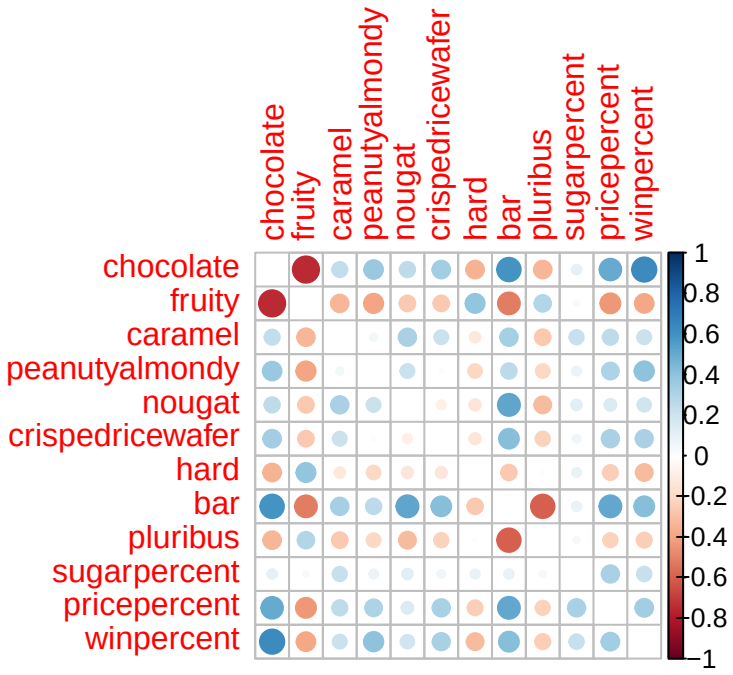


Q19. Which candy type is the highest ranked in terms of winpercent for the least money - i.e. offers the most bang for your buck?

```
library(corrplot)
```

```
corrplot 0.95 loaded
```

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

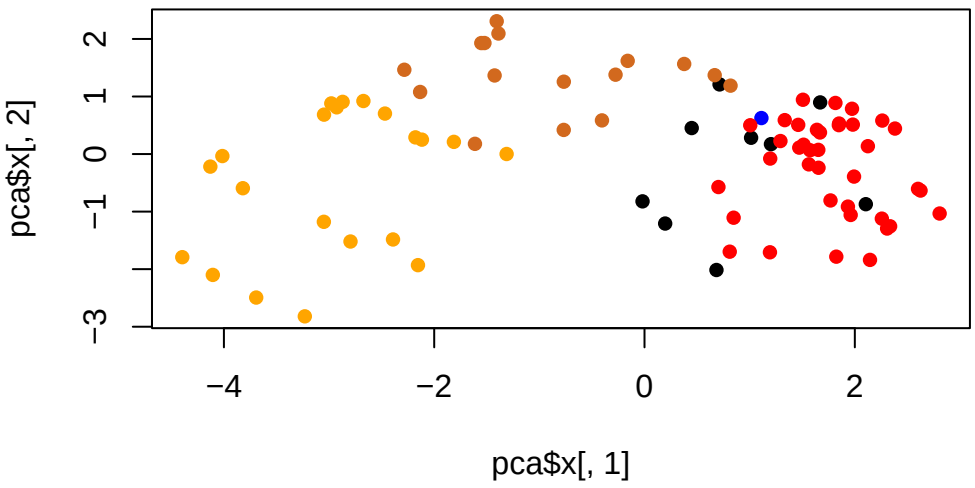


Principal Component Analysis

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

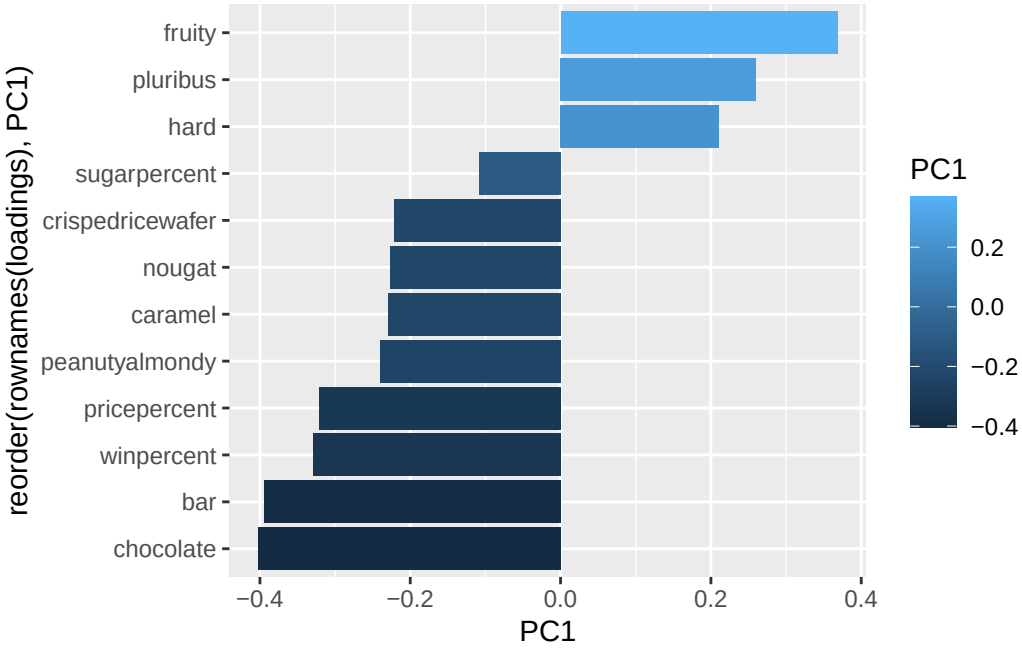
```
Importance of components:
                          PC1    PC2    PC3     PC4    PC5     PC6     PC7
Standard deviation     2.0788 1.1378 1.1092 1.07533 0.9518 0.81923 0.81530
Proportion of Variance 0.3601 0.1079 0.1025 0.09636 0.0755 0.05593 0.05539
Cumulative Proportion  0.3601 0.4680 0.5705 0.66688 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)
```



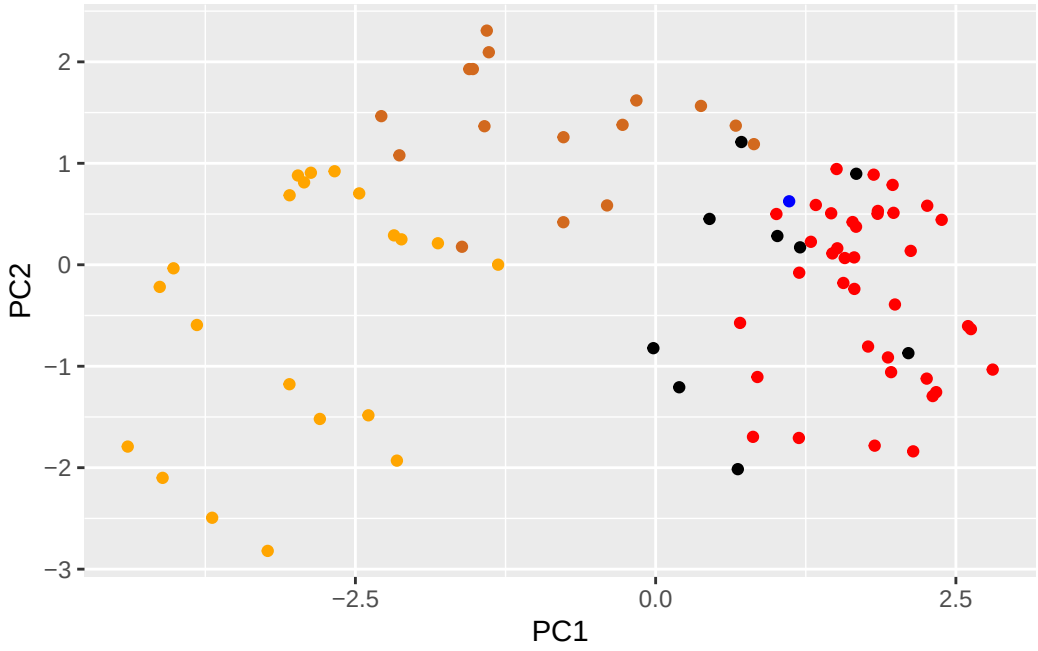
How do the original variables (columns) contribute to the new PCs. I will look at PC1 here

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

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



Let's make a nicer score plot with ggplot. Again I need a data.frame with all the stuff I want (PC results and candy data) for my plot as input

```
pc.results <- cbind(candy, pca$x)

ggplot(pc.results) +
  aes(PC1, PC2, label=rownames(pc.results)) +
  geom_point(col=mycols)
```




```
  geom_text_repel(col=mycols)
```

```
geom_text_repel: parse = FALSE, na.rm = FALSE, box.padding = 0.25, point.padding = 1e-06, mi
stat_identity: na.rm = FALSE
position_identity
```

```
  labs(title="Candy Space via PCA")
```

```
$title
[1] "Candy Space via PCA"

attr(,"class")
[1] "labels"
```

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