Class 9

AUTHOR
Jo Bautista

```
candy_file <- read.csv("https://raw.githubusercontent.com/fivethirtyeight/data/master/candy-power

head(candy_file)
```

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

#A1
# Number of different candy types
num_candy_types <- nrow(candy_file)
num_candy_types
```

```
[1] 85
```

```
#A2
# Number of fruity candy types
num_fruity_candies <- sum(candy_file$fruity)
num_fruity_candies
```

```
[1] 38
```

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

```
[1] 81.64291
```

```
#Q3. What is your favorite candy in the dataset and what is it’s winpercent value?
#Q4. What is the winpercent value for “Kit Kat”?
#Q5. What is the winpercent value for “Tootsie Roll Snack Bars”?

#A3.
candy_file["Haribo Gold Bears", ]$winpercent
```

```
[1] 57.11974
```

```
#A4.
candy_file["Kit Kat", ]$winpercent
```

```
[1] 76.7686
```

```
#A5.
candy_file["Tootsie Roll Snack Bars", ]$winpercent
```

```
[1] 49.6535
```

```
library("skimr")
```

```
Warning: package 'skimr' was built under R version 4.3.3
```

```
skim(candy_file)
```

Data summary

Name	candy_file
Number of rows	85
Number of columns	12
_____	
Column type frequency:	
numeric	12
_____	
Group variables	None

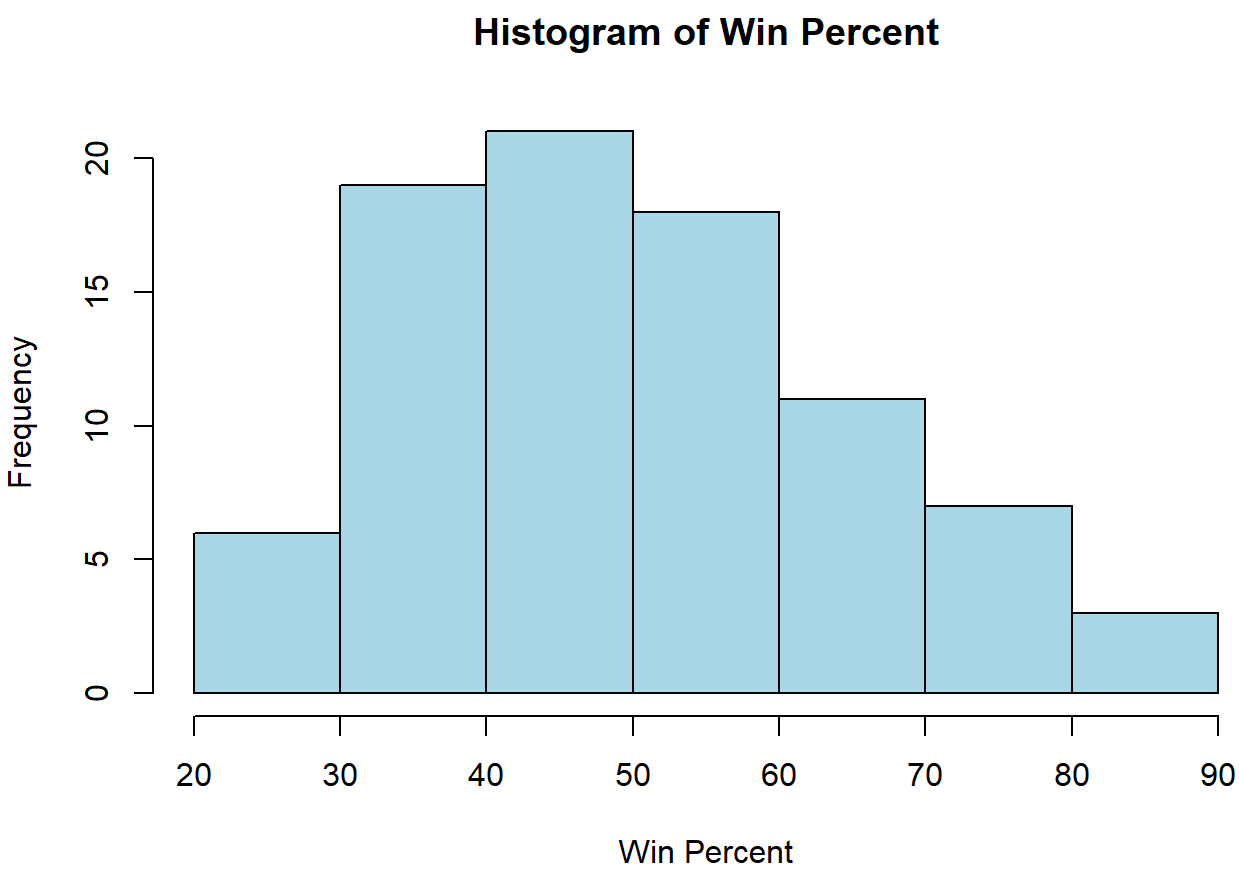
Variable type: numeric

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	▆▁▁▁▁▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
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 otl
#Q7. What do you think a zero and one represent for the candy$chocolate column?


#A6.
#Most variables are on the 0-1 scale.  However, the final column ("hist") displays a small repres

#A7.
# 0 represents candies that do not contain chocolate.  1 represents candies that do contain choch
```

```
# Q8:  Plot a histogram of winpercent values.
# A8:
hist(candy_file$winpercent, main = "Histogram of Win Percent", xlab = "Win Percent", col = "lightl
```




```
# Q9: Is the distribution of winpercent values symmetrical?
# A9: Check for symmetry (visually through the histogram and calculate skewness).
skewness <- mean((candy_file$winpercent - mean(candy_file$winpercent))^3) / sd(candy_file$winperc
print(paste("Skewness:", skewness))
```

```
[1] "Skewness: 0.320676060892139"
```

```
# Q10: Is the center of the distribution above or below 50%?
# A10: Calculate the mean of winpercent.
mean_winpercent <- mean(candy_file$winpercent)
print(paste("Mean Win Percent:", mean_winpercent))
```

```
[1] "Mean Win Percent: 50.3167638117647"
```

```
# Q11: On average is chocolate candy higher or lower ranked than fruit candy?
# A11: Mean winpercent for chocolate and fruity candies
mean_chocolate <- mean(candy_file$winpercent[as.logical(candy_file$chocolate)])
mean_fruity <- mean(candy_file$winpercent[as.logical(candy_file$fruity)])
print(paste("Mean Win Percent (Chocolate):", mean_chocolate))
```

```
[1] "Mean Win Percent (Chocolate): 60.9215294054054"
```

```r
print(paste("Mean Win Percent (Fruity):", mean_fruity))
```

```
[1] "Mean Win Percent (Fruity): 44.1197414210526"
```

```r
# Q12: Is this difference statistically significant?
# A12: Conduct a t-test
t_test_result <- t.test(candy_file$winpercent[as.logical(candy_file$chocolate)],
                        candy_file$winpercent[as.logical(candy_file$fruity)])
print(t_test_result)
```

```
	Welch Two Sample t-test

data:  candy_file$winpercent[as.logical(candy_file$chocolate)] and
candy_file$winpercent[as.logical(candy_file$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
```

```r
# Q13: What are the five least liked candy types in this set?
# A13: Using base R:
least_liked_candies <- head(candy_file[order(candy_file$winpercent), ], n = 5)
print("Five Least Liked Candy Types:")
```

```
[1] "Five Least Liked Candy Types:"
```

```r
print(least_liked_candies)
```

```
                   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
```

```
# Q14: What are the top 5 all time favorite candy types out of this set?
# A14: Using base R:
top_favorite_candies <- head(candy_file[order(-candy_file$winpercent), ], n = 5)
print("Top 5 Favorite Candy Types:")
```

```
[1] "Top 5 Favorite Candy Types:"
```

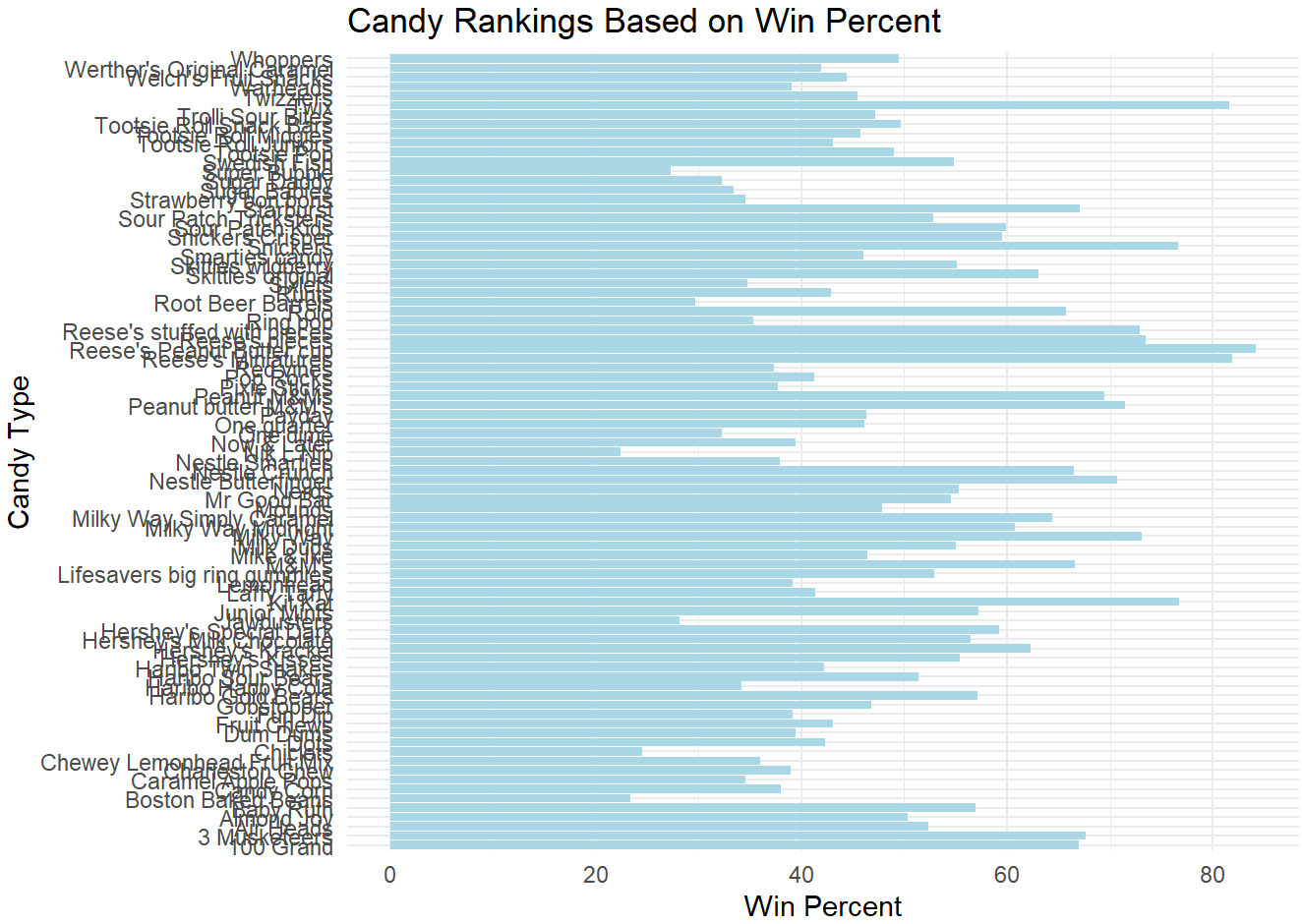
```
print(top_favorite_candies)
```

```
                         chocolate fruity caramel peanutyalmondy 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
                         crispedricewafer hard bar pluribus sugarpercent
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
```

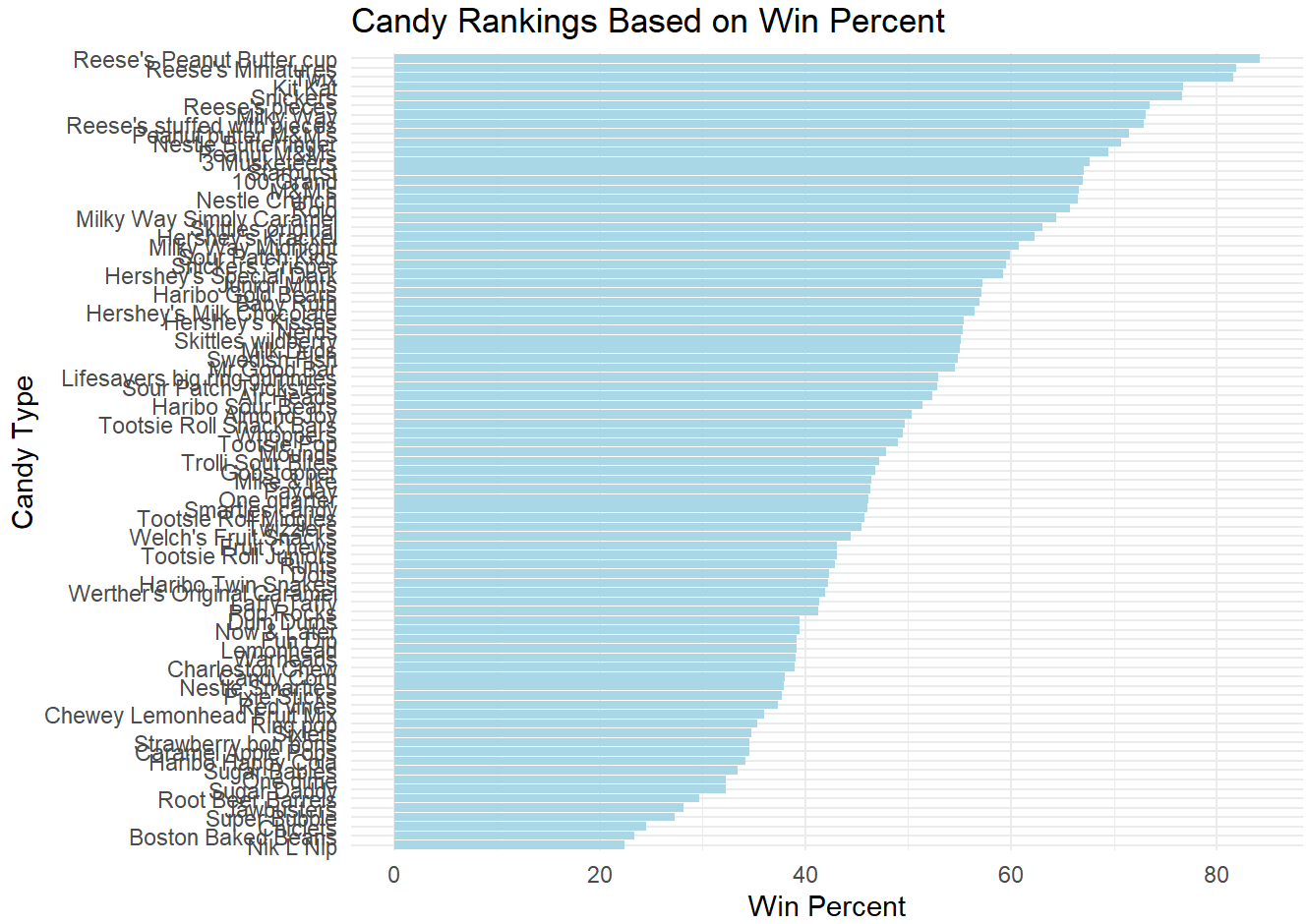
```
#Q15. Make a first barplot of candy ranking based on winpercent values.

library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.3.3
```

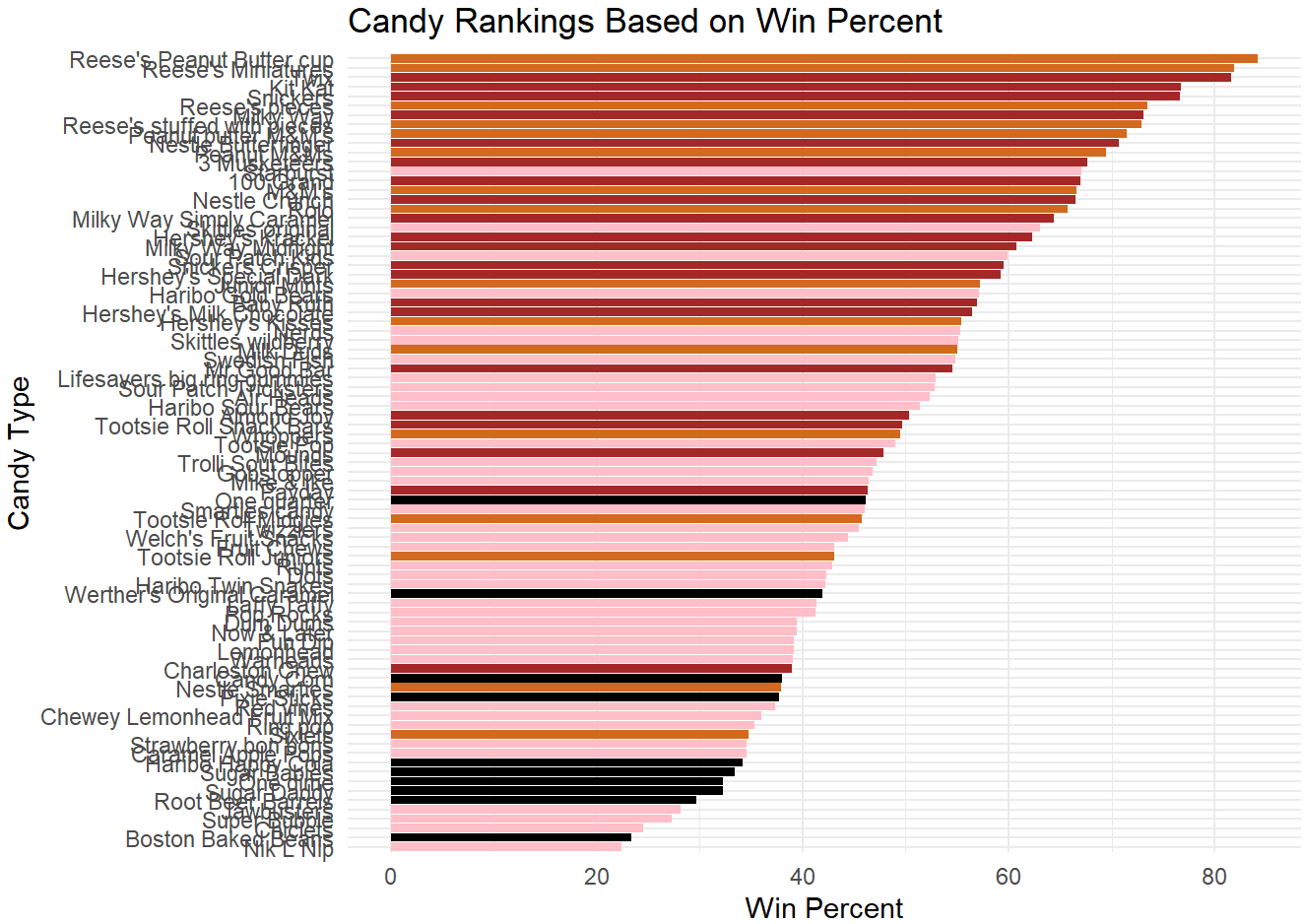
```
#A15.
ggplot(data = candy_file, aes(x = winpercent, y = rownames(candy_file))) +
  geom_bar(stat = "identity", fill = "lightblue") +
  labs(title = "Candy Rankings Based on Win Percent", x = "Win Percent", y = "Candy Type") +
  theme_minimal()
```




```
#Q16. This is quite ugly, use the reorder() function to get the bars sorted by winpercent?
#A16.
ggplot(data = candy_file, aes(x = winpercent, y = reorder(rownames(candy_file), winpercent))) +
  geom_bar(stat = "identity", fill = "lightblue") +
  labs(title = "Candy Rankings Based on Win Percent", x = "Win Percent", y = "Candy Type") +
  theme_minimal()
```




```
#Time to add some useful color:

# Create a color vector initialized to black
my_cols <- rep("black", nrow(candy_file))

# Overwrite colors based on candy type
my_cols[as.logical(candy_file$chocolate)] <- "chocolate"
my_cols[as.logical(candy_file$bar)] <- "brown"
my_cols[as.logical(candy_file$fruity)] <- "pink"

# Create the bar plot using my_cols for fill
ggplot(candy_file) +
  aes(x = winpercent, y = reorder(rownames(candy_file), winpercent)) +
  geom_col(fill = my_cols) +
  labs(title = "Candy Rankings Based on Win Percent", x = "Win Percent", y = "Candy Type") +
  theme_minimal()
```




```
# Q17: Worst ranked chocolate candy
worst_ranked_chocolate <- candy_file[candy_file$chocolate == 1, ]
worst_chocolate <- worst_ranked_chocolate[which.min(worst_ranked_chocolate$winpercent), ]
print("Worst Ranked Chocolate Candy:")
```

```
[1] "Worst Ranked Chocolate Candy:"
```

```
print(worst_chocolate)
```

```
        chocolate fruity caramel peanutyalmondy nougat crispedricewafer hard
Sixlets         1      0       0               0      0                 0    0
        bar pluribus sugarpercent pricepercent winpercent
Sixlets   0        1         0.22        0.081     34.722
```

```
# Q18: Best ranked fruity candy
best_ranked_fruity <- candy_file[candy_file$fruity == 1, ]
best_fruity <- best_ranked_fruity[which.max(best_ranked_fruity$winpercent), ]
print("Best Ranked Fruity Candy:")
```

```
[1] "Best Ranked Fruity Candy:"
```

```
print(best_fruity)
```

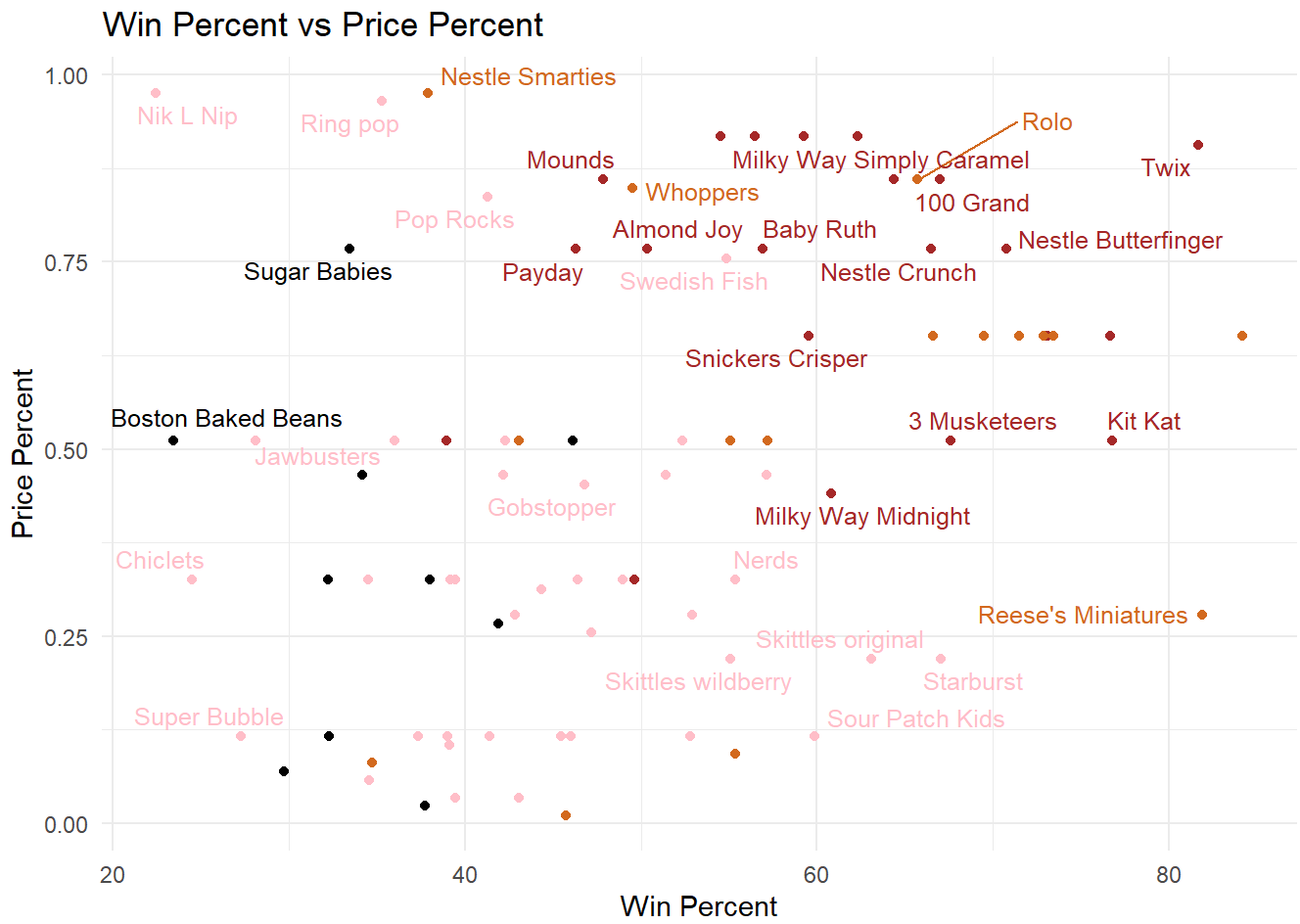
```
          chocolate fruity caramel peanutyalmondy nougat crispedricewafer hard
Starburst         0      1       0              0      0                0    0
          bar pluribus sugarpercent pricepercent winpercent
Starburst   0        1        0.151         0.22   67.03763
```

```
library(ggrepel)
```

```
Warning: package 'ggrepel' was built under R version 4.3.3
```

```
# Create a color vector for candy types
my_cols <- rep("black", nrow(candy_file))
my_cols[as.logical(candy_file$chocolate)] <- "chocolate"
my_cols[as.logical(candy_file$bar)] <- "brown"
my_cols[as.logical(candy_file$fruity)] <- "pink"

# Plot winpercent vs pricepercent
ggplot(candy_file) +
  aes(x = winpercent, y = pricepercent, label = rownames(candy_file)) +
  geom_point(col = my_cols) +
  geom_text_repel(col = my_cols, size = 3.3, max.overlaps = 5) +
  labs(title = "Win Percent vs Price Percent", x = "Win Percent", y = "Price Percent") +
  theme_minimal()
```

```
Warning: ggrepel: 53 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. of
#A19.
#Calculate the ratio of winpercent to pricepercent
candy_file$bang_for_buck <- candy_file$winpercent / candy_file$pricepercent

# Find the candy with the highest bang for buck
best_value_candy <- candy_file[which.max(candy_file$bang_for_buck), ]
print("Candy with the highest winpercent for the least money:")
```

```
[1] "Candy with the highest winpercent for the least money:"
```

```
print(best_value_candy)
```

```
                     chocolate fruity caramel peanutyalmondy nougat
Tootsie Roll Midgies         1      0       0              0      0
                     crispedricewafer hard bar pluribus sugarpercent
Tootsie Roll Midgies                0    0   0        1        0.174
                     pricepercent winpercent bang_for_buck
Tootsie Roll Midgies        0.011   45.73675      4157.886
```

```
#Q20. What are the top 5 most expensive candy types in the dataset and of these which is the leas
#A20.
#Order the dataset by pricepercent in decreasing order
ord <- order(candy_file$pricepercent, decreasing = TRUE)
top_expensive_candies <- candy_file[ord, ][1:5, ]

# Find the least popular candy among the top 5 most expensive
least_popular_among_expensive <- top_expensive_candies[which.min(top_expensive_candies$winpercent
print("Top 5 most expensive candy types:")
```

```
[1] "Top 5 most expensive candy types:"
```

```
print(top_expensive_candies)
```

```
                        chocolate fruity caramel peanutyalmondy nougat
Nik L Nip                       0      1       0              0      0
Nestle Smarties                 1      0       0              0      0
Ring pop                        0      1       0              0      0
Hershey's Krackel               1      0       0              0      0
Hershey's Milk Chocolate        1      0       0              0      0
                        crispedricewafer hard bar pluribus sugarpercent
Nik L Nip                               0    0   0        1        0.197
Nestle Smarties                         0    0   0        1        0.267
Ring pop                                0    1   0        0        0.732
Hershey's Krackel                       1    0   1        0        0.430
Hershey's Milk Chocolate                0    0   1        0        0.430
                        pricepercent winpercent bang_for_buck
Nik L Nip                      0.976   22.44534      22.99728
Nestle Smarties                0.976   37.88719      38.81884
Ring pop                       0.965   35.29076      36.57073
Hershey's Krackel              0.918   62.28448      67.84802
Hershey's Milk Chocolate       0.918   56.49050      61.53649
```

```
print("Least popular candy among the top 5 most expensive:")
```

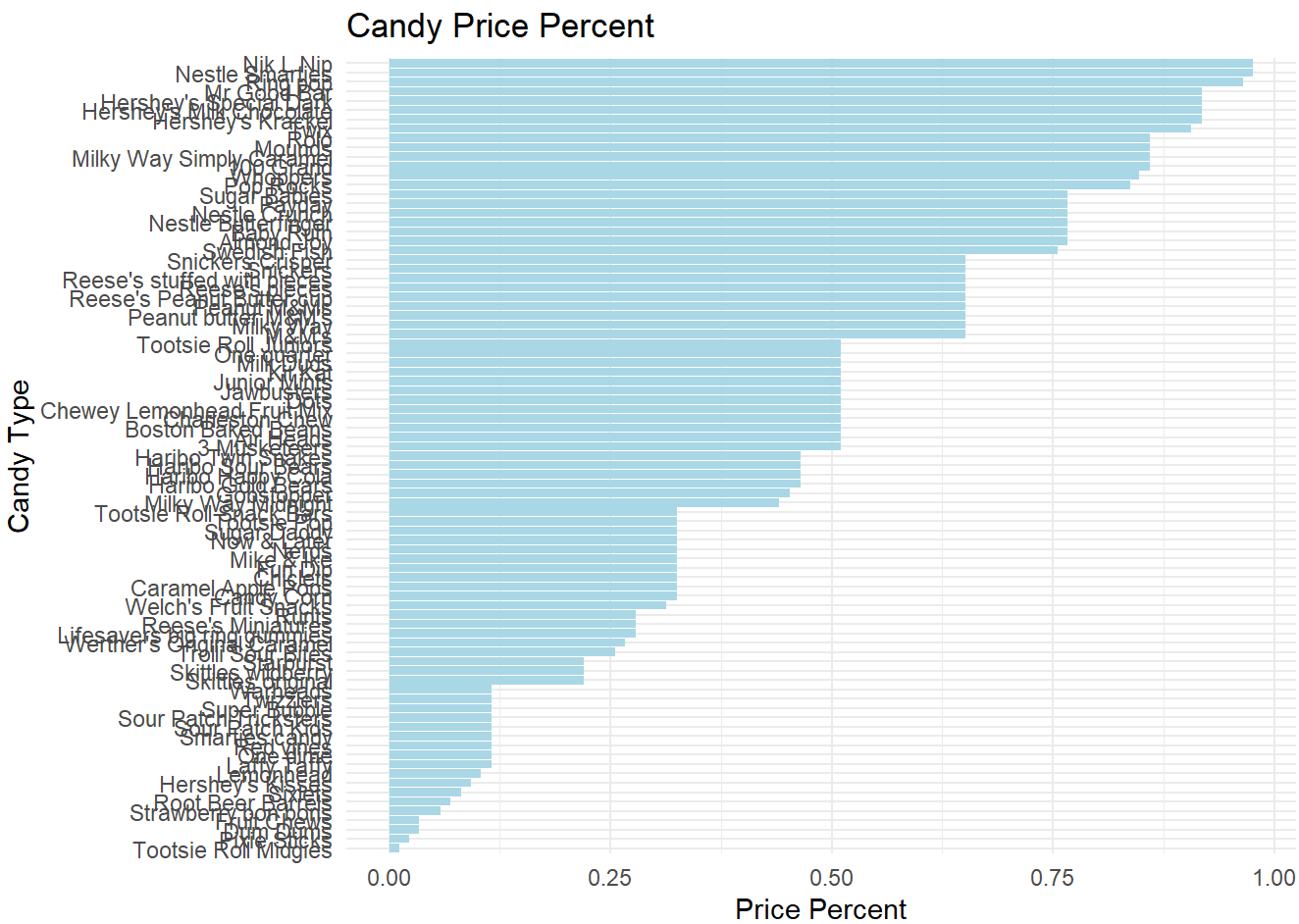
```
[1] "Least popular candy among the top 5 most expensive:"
```

```
print(least_popular_among_expensive)
```

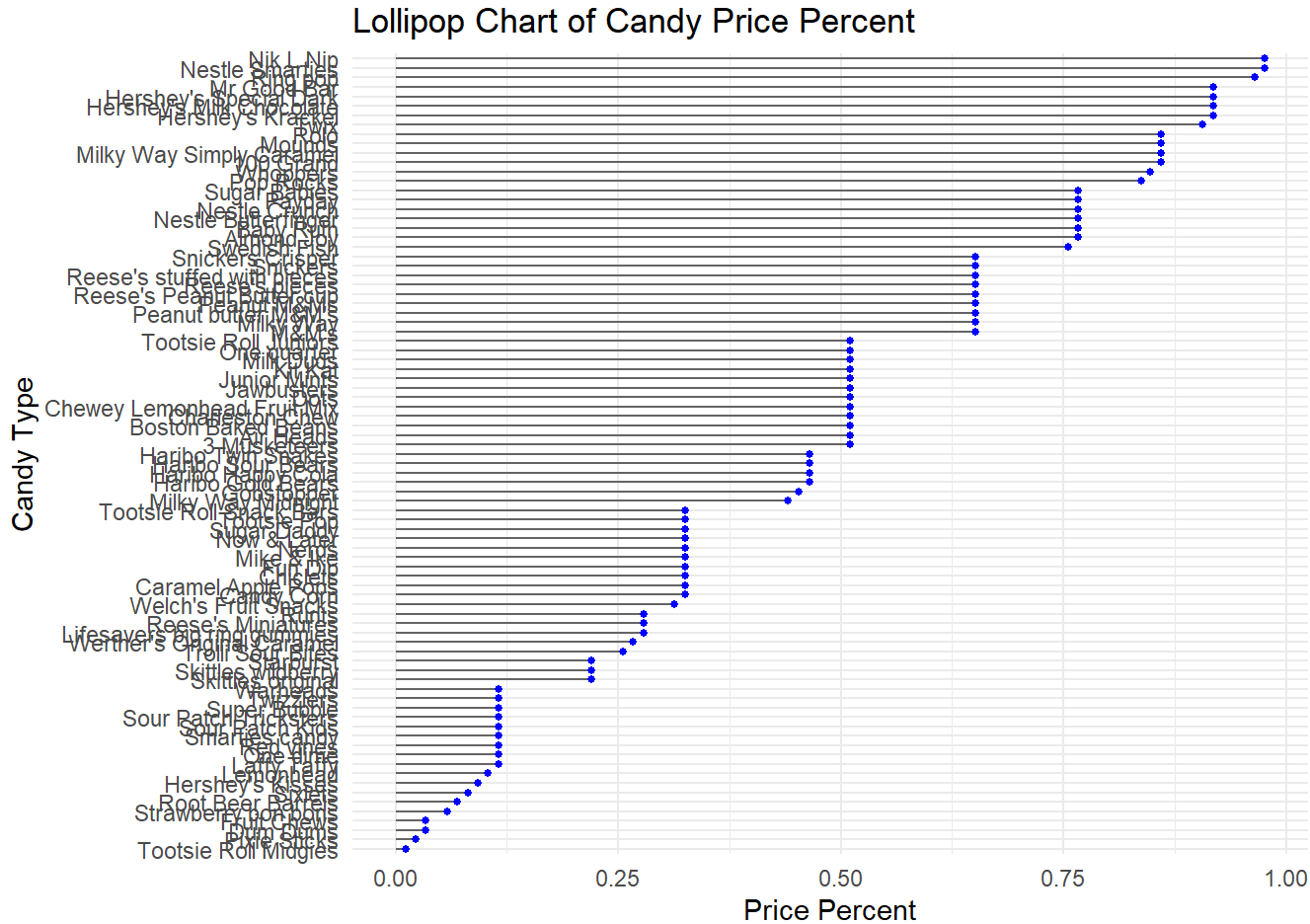
```
          chocolate fruity caramel peanutyalmondy nougat crispedricewafer hard
Nik L Nip         0      1       0              0      0                0    0
          bar pluribus sugarpercent pricepercent winpercent bang_for_buck
Nik L Nip   0        1        0.197        0.976   22.44534      22.99728
```

```
#Q21. Make a barplot again with geom_col() this time using pricepercent and then improve this ste

#A21.
```

```
# Create the initial bar plot for pricepercent
ggplot(candy_file) +
  aes(x = reorder(rownames(candy_file), pricepercent), y = pricepercent) +
  geom_col(fill = "lightblue") +
  labs(title = "Candy Price Percent", x = "Candy Type", y = "Price Percent") +
  theme_minimal() +
  coord_flip()  # Flipping coordinates for better visibility
```




```
# Make a lollipop chart of pricepercent
ggplot(candy_file) +
  aes(x = pricepercent, y = reorder(rownames(candy_file), pricepercent)) +
  geom_segment(aes(xend = 0, yend = reorder(rownames(candy_file), pricepercent)), color = "gray40
  geom_point(size = 1, color = "blue") +
  labs(title = "Lollipop Chart of Candy Price Percent", x = "Price Percent", y = "Candy Type") +
  theme_minimal()
```



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

#A22. Anti-Correlated Variables –> Look for pairs of variables that have a correlation coefficient close to -1 (negative values). These are the variables that are anti-correlated.

#bang_for_buck & pricepercent; pricepercent & winpercent

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

#A23. Most Positively Correlated Variables –> Look for pairs of variables with a correlation coefficient close to 1 (positive values). These indicate strong positive correlation.

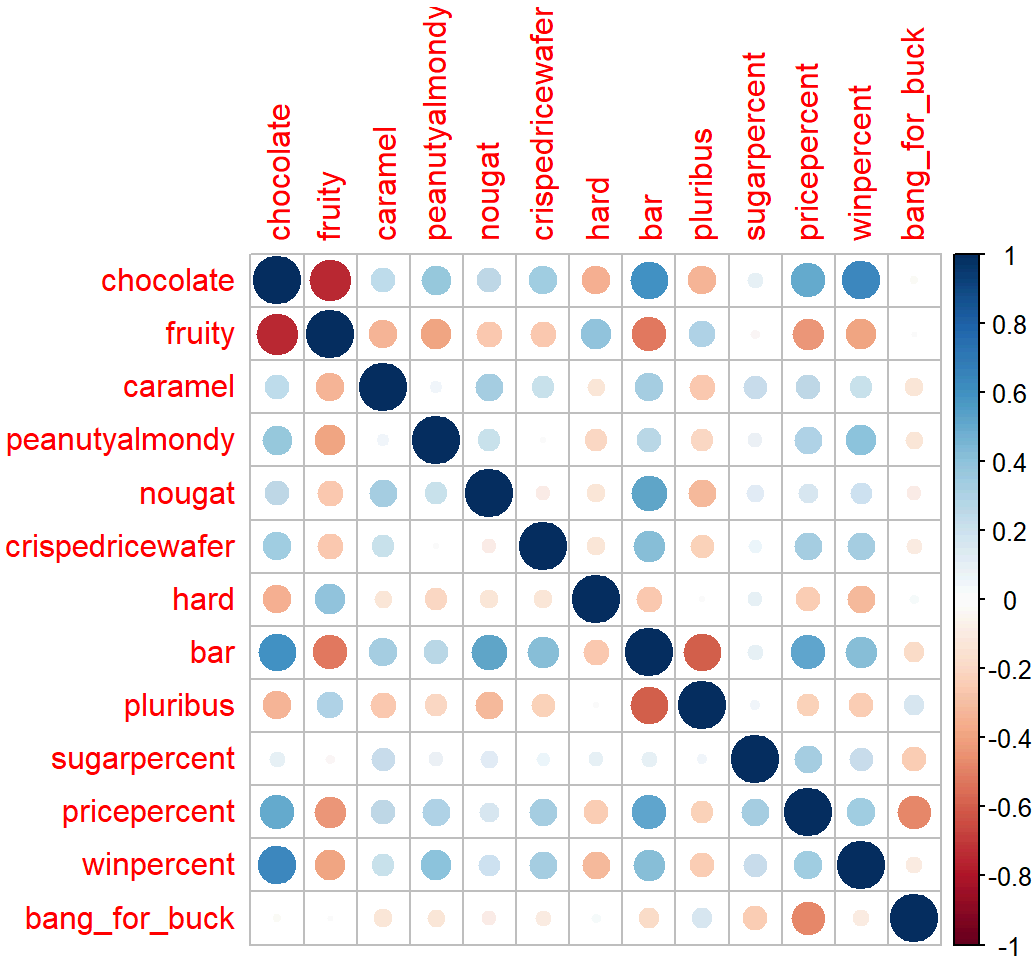
#fruity & chocolate; caramel & chocolate

```
library(corrplot)
```

```
Warning: package 'corrplot' was built under R version 4.3.3
```

```
corrplot 0.95 loaded
```

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

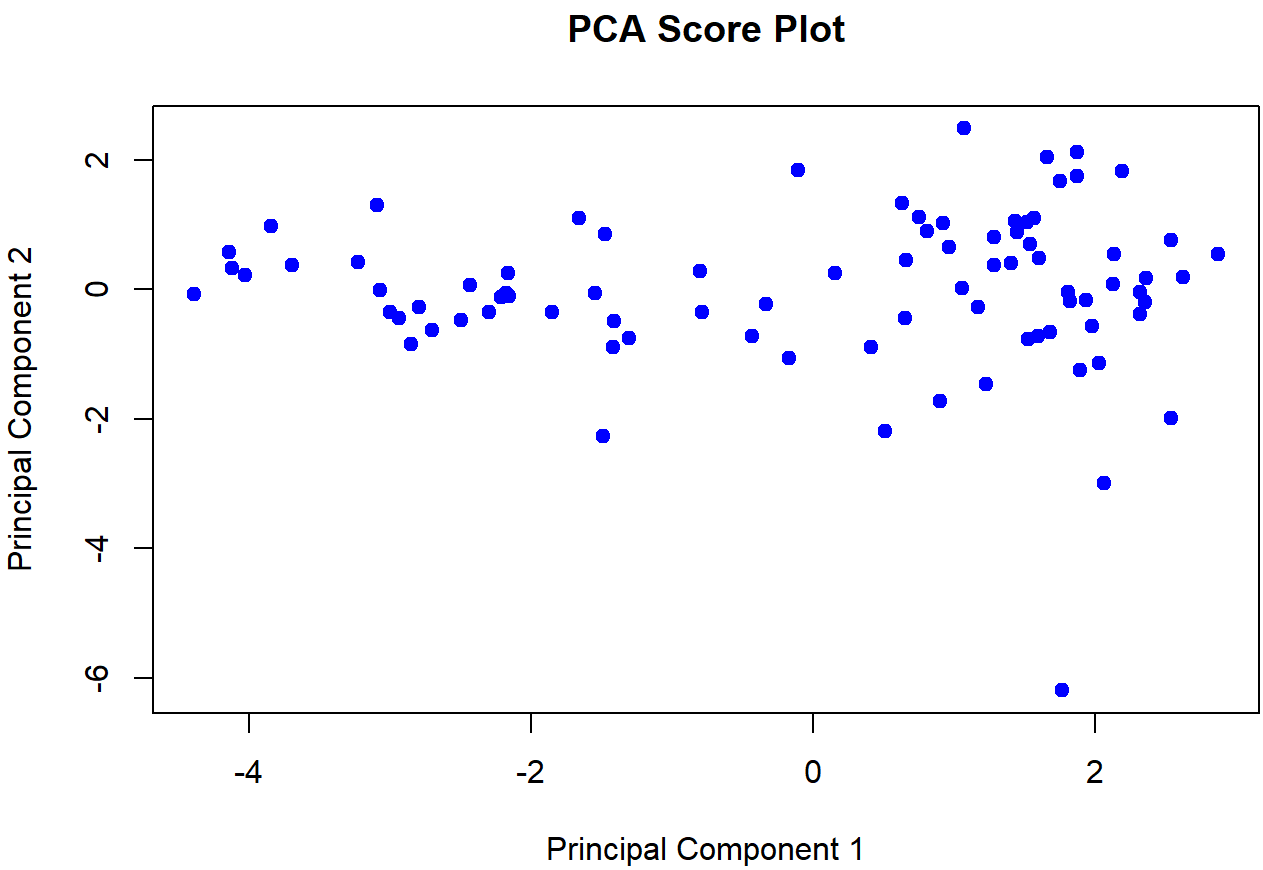



```
# Select relevant numeric columns for PCA
numeric_columns <- candy_file[, sapply(candy_file, is.numeric)]

# Perform PCA
pca <- prcomp(numeric_columns, scale = TRUE)

# Summary of the PCA
summary(pca)
```

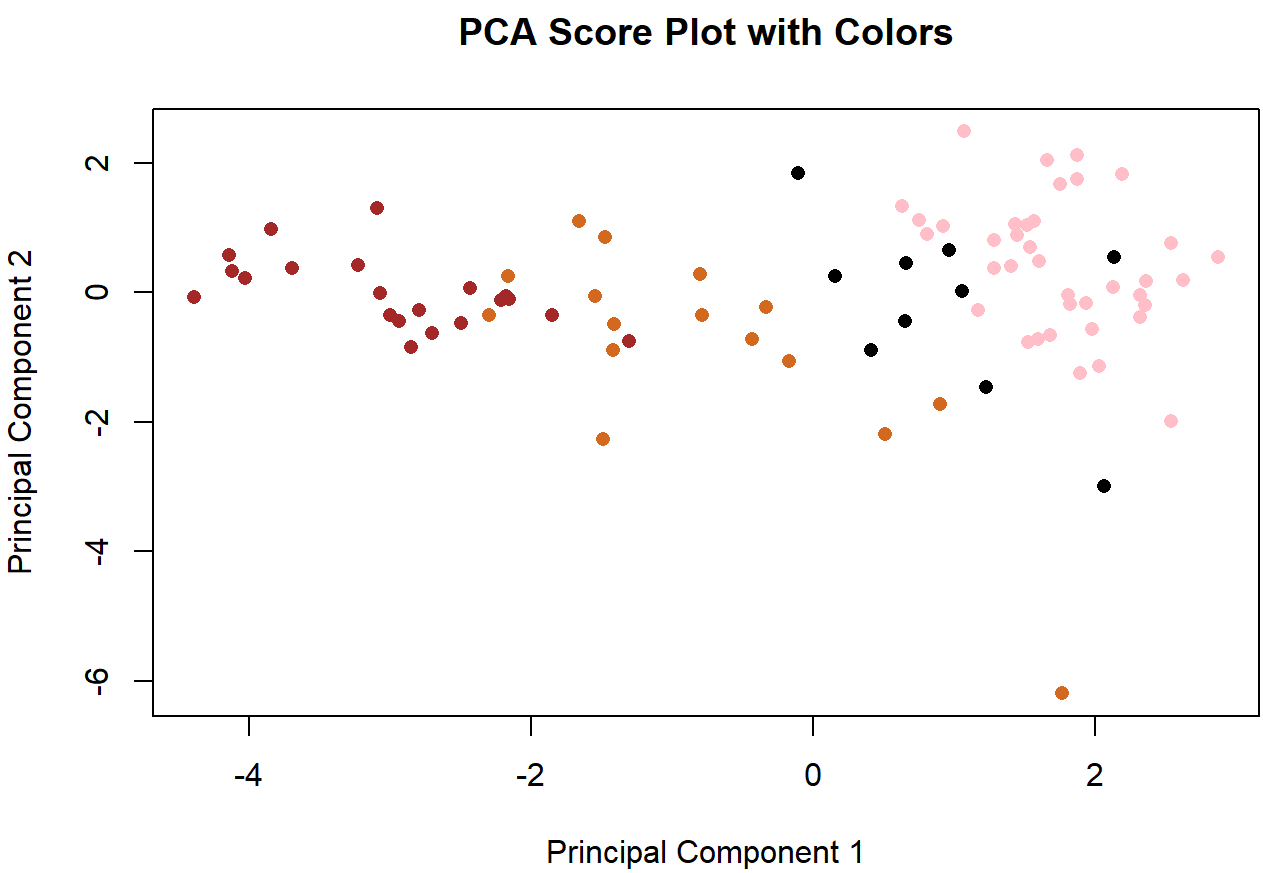
```
Importance of components:
                          PC1    PC2     PC3    PC4     PC5     PC6     PC7
Standard deviation     2.0938 1.2127 1.13054 1.0787 0.98027 0.93656 0.81530
Proportion of Variance 0.3372 0.1131 0.09832 0.0895 0.07392 0.06747 0.05113
Cumulative Proportion  0.3372 0.4503 0.54866 0.6382 0.71208 0.77956 0.83069
                           PC8     PC9    PC10    PC11    PC12    PC13
Standard deviation     0.78462 0.68466 0.66328 0.57829 0.43128 0.39534
Proportion of Variance 0.04736 0.03606 0.03384 0.02572 0.01431 0.01202
Cumulative Proportion  0.87804 0.91410 0.94794 0.97367 0.98798 1.00000
```

```
# Plot the scores for PC1 and PC2
plot(pca$x[, 1], pca$x[, 2],
     xlab = "Principal Component 1",
     ylab = "Principal Component 2",
```

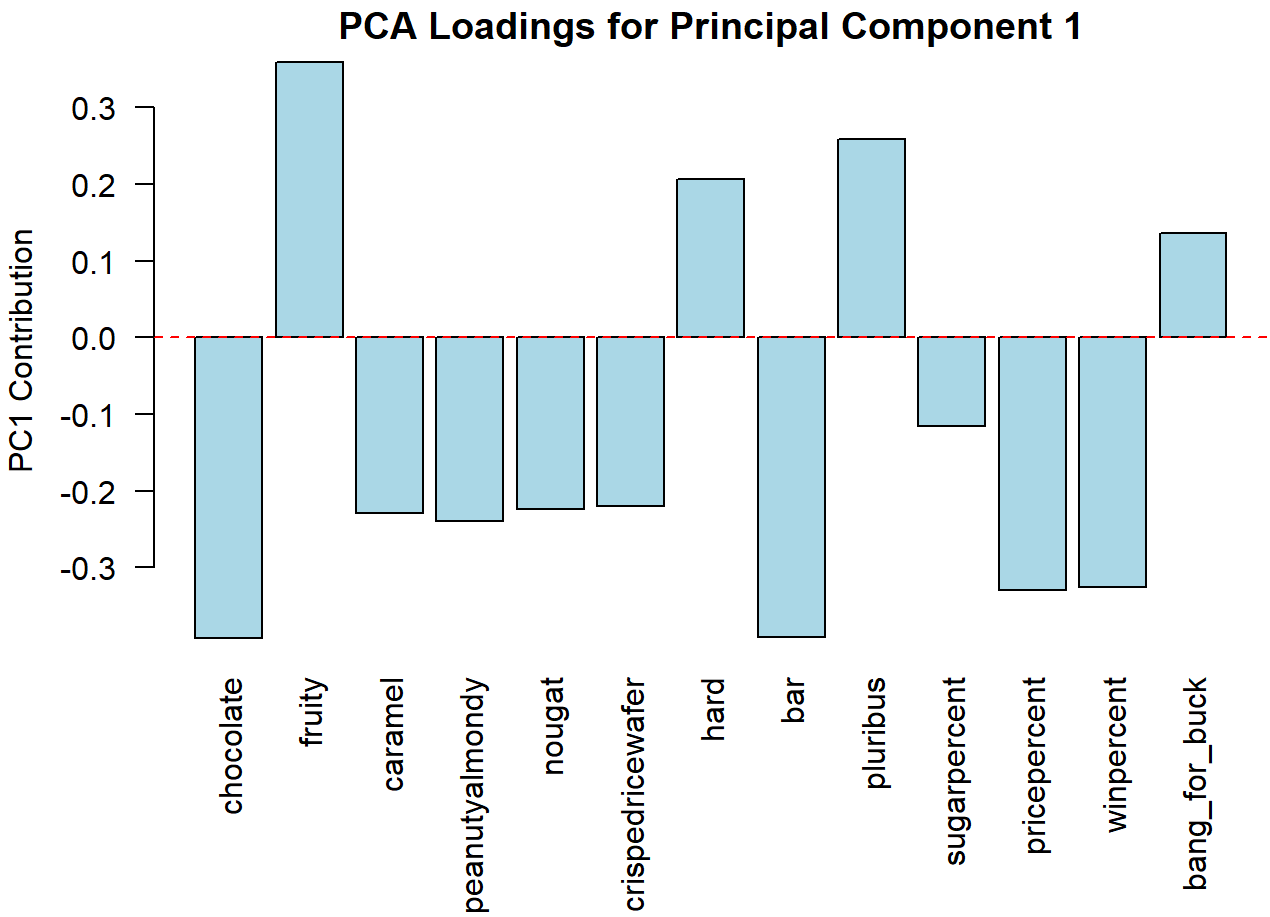
```
     main = "PCA Score Plot",
     pch = 19, col = "blue")
```




```
# Create the color vector based on candy types
my_cols <- rep("black", nrow(candy_file))
my_cols[as.logical(candy_file$chocolate)] <- "chocolate"
my_cols[as.logical(candy_file$bar)] <- "brown"
my_cols[as.logical(candy_file$fruity)] <- "pink"

# Plot the PCA scores for PC1 vs PC2 with colors
plot(pca$x[, 1], pca$x[, 2],
     col = my_cols,
     pch = 16,
     xlab = "Principal Component 1",
     ylab = "Principal Component 2",
     main = "PCA Score Plot with Colors")
```




```
# Select relevant numeric columns for PCA
numeric_columns <- candy_file[, sapply(candy_file, is.numeric)]

# Perform PCA with scaling
pca <- prcomp(numeric_columns, scale = TRUE)

# Set up the plotting parameters
par(mar = c(8, 4, 2, 2))  # Adjust margins to accommodate labels

# Create a bar plot of PC1 loadings
barplot(pca$rotation[, 1],
        las = 2,
        ylab = "PC1 Contribution",
        main = "PCA Loadings for Principal Component 1",
        col = "lightblue")

# Optionally, add a horizontal line at y = 0 for better visualization
abline(h = 0, col = "red", lty = 2)
```




```
# Select relevant numeric columns for PCA
numeric_columns <- candy_file[, sapply(candy_file, is.numeric)]

# Perform PCA with scaling
pca <- prcomp(numeric_columns, scale = TRUE)

# View the loadings for PC1
loadings_pc1 <- pca$rotation[, 1]
loadings_pc1
```

```
      chocolate           fruity          caramel    peanutyalmondy
     -0.3924439        0.3588085       -0.2293954       -0.2389173
         nougat crispedricewafer             hard              bar
     -0.2241826       -0.2195121        0.2059573       -0.3912663
       pluribus     sugarpercent     pricepercent       winpercent
      0.2590791       -0.1161206       -0.3299041       -0.3250778
   bang_for_buck
      0.1359085
```

```
#Q24. What original variables are picked up strongly by PC1 in the positive direction? Do these ma

#A24. fruity > pluribus > hard > bang_for_buck.  Yes, this makes sense.
```