Homework 03 Data Cleaning, Normal Distributions

Due by 11:59pm, Friday, 2.7.25

S&DS 230/530/ENV 757

(1) More on List Manipulation *(18 points - 3 points each).*

(1.1) Make an object called `myList` that contains the following elements (in order):

- A matrix with the integers 1 through 36 that has six rows, filled by row
- A list that contains
  - A vector with the text "Pianoforte" and "Gerbera Daisy"
  - A vector with the integers 9 through 13
- The integers 1 through 10

You should be able to make this object in a single line of code.

Use `[]`, `[[]]`, `[,]` notation to answer parts b) through f).

(1.2) Make an object called `ans1` that is the third row of the matrix contained in `myList`.

(1.3) Make an object called `ans2` that is the sum of the 6th column of the matrix contained in `myList`.

(1.4) Make an object called `ans3` that is the sum of EACH column of the matrix contained in `myList` *(use the `apply()` function or check out `colSums()`).*

(1.5) Make an object called `ans4` that is the single element of `myList` that you can play.

(1.6) Make an object called `ans5` that is the third element of the second element of the second element of `myList` converted to characters.

Get the results of each of your objects you created above (i.e. get them to show up in your knitted file by typing their names or putting the code line that creates each object in parentheses).

```
myList <- list(matrix(1:36, nrow = 6, ncol = 6, byrow=TRUE),
               list(c("Pianoforte", "Gerbera Daisy"), c(9:13)),
               1:10)
myList
```

```
## [[1]]
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    2    3    4    5    6
## [2,]    7    8    9   10   11   12
## [3,]   13   14   15   16   17   18
## [4,]   19   20   21   22   23   24
## [5,]   25   26   27   28   29   30
## [6,]   31   32   33   34   35   36
##
```

```
## [[2]]
## [[2]][[1]]
## [1] "Pianoforte"    "Gerbera Daisy"
##
## [[2]][[2]]
## [1]  9 10 11 12 13
##
##
## [[3]]
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
ans1 <- myList[[1]][3, ]
ans1
```

```
## [1] 13 14 15 16 17 18
```

```
ans2 <- sum(myList[[1]][, 6])
ans2
```

```
## [1] 126
```

```
ans3 <- apply(myList[[1]], 2, sum)
ans3
```

```
## [1]  96 102 108 114 120 126
```

```
ans4 <- myList[[2]][[1]][1]
ans4
```

```
## [1] "Pianoforte"
```

```
ans5 <- as.character(myList[[2]][[2]][3])
ans5
```

```
## [1] "11"
```

(2) Normal Quantile Plots and the Binomial Distribution *(20 points, 3 points each, part (2.5) is 5 points).*

You may recall from your Intro Statistics course that a binomial distribution looks like a normal distribution if np > 10 and n(1-p) > 10 (i.e. as long as the average number of successes and failures are both larger than 10). Recall that n is the number of trials, and p is the probability of success for each Bernoulli trial. *As an example, flip a coin 30 times, count the number of heads. n=30, p=.5, np = 15 > 10 and n(1-p) = 15 > 10, so the distribution should be approximately normal).*

You are going to make six normal quantile plots that simulate 127 random observations from binomial distributions with p = 0.3 and various values of n.

(2.1) Install the `car` package. This will allow you use the `qqPlot()` function. Load this package.

(2.2) Make a vector called `vec` that is powers of 10 for powers 0 through 5. The one caveat is that you need to use the`**` operator which reads as 'to the power of' *(i.e. `2**3` is 8).* Show what is contained in `vec`.

(2.3) Use the `par()` function to set up your plot region to show 6 plots on a page. Go learn about the `mfrow` option in par to create a plot that was two rows and three columns.

(2.4) Use the `rbinom()` function to generate 6 random binomial observations, each with 17 trials, and with p=0.6. You may need to type `?rbinom` to get the syntax for this function. Store the result in an object called `vec2` and show what is contained in `vec2`.

(2.5) Write a loop that repeatedly creates a normal quantile plot for 127 random samples each from a binomial distribution with p = 0.3 and n equal to the 6 values stored in `vec`. A few plot details: * Use the `qqPlot` function. * Make the graph points red solid dots (`pch = 19`). * Make the boundary lines blue (use `col.lines`) * Make a main graph title that pastes the text "127 Binomial Samples, N =" to the corresponding value from `vec`.

```
library("car")
```

```
## Loading required package: carData
```

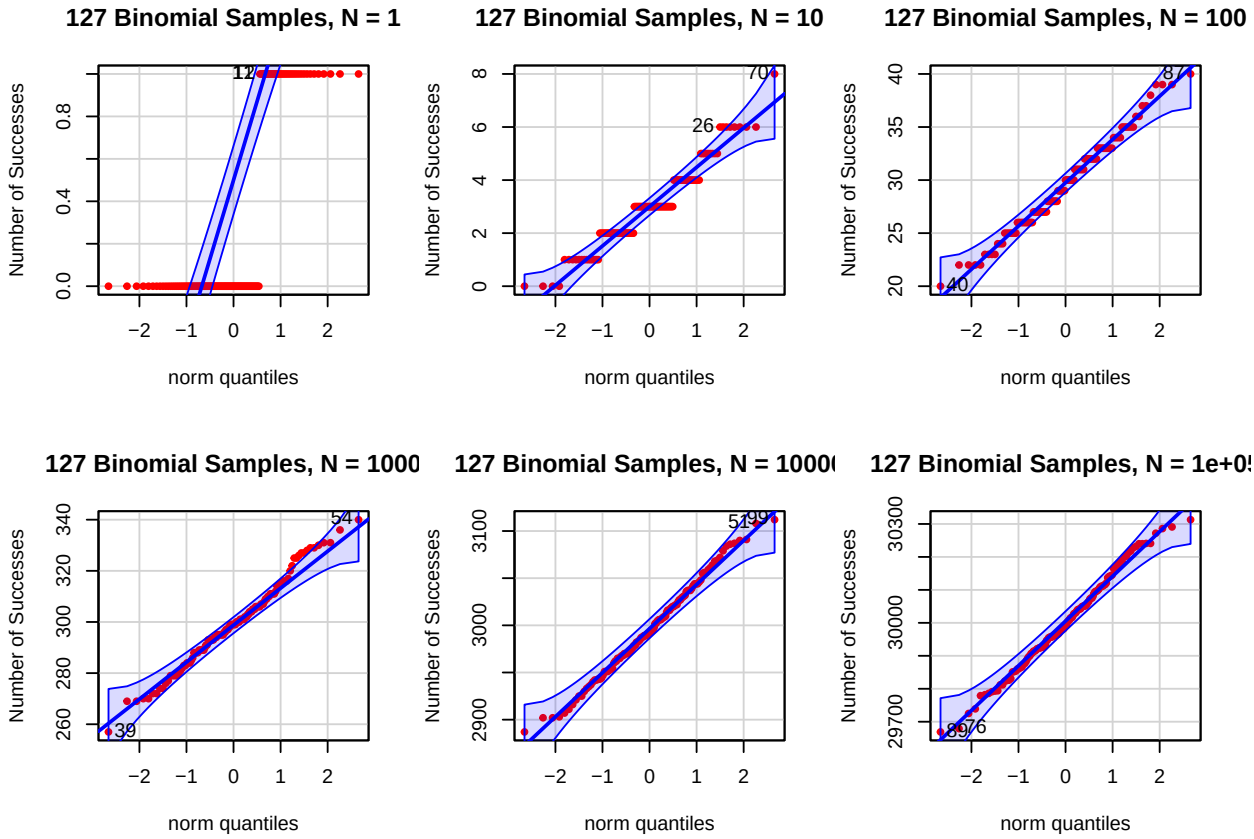
```
vec <- 10**(0:5)
vec
```

```
## [1] 1e+00 1e+01 1e+02 1e+03 1e+04 1e+05
```

```
par(mfrow=c(2, 3))
vec2 <- rbinom(6, 17, 0.6)
vec2
```

```
## [1] 11  7 12 10  7  7
```

```
for (i in 1:6) {
  qqPlot(rbinom(127, vec[i], 0.3),
         distribution = "norm",
         pch=19,
         col='red',
         col.lines='blue',
         main=paste("127 Binomial Samples, N =", vec[i]),
         ylab="Number of Successes")
}
```



(2.6) Take a look at the normal quantile plots. For what value of n do the graphs seem to be approximately normally distributed? Is this consistent with what you expect? Write two complete sentences to answer these questions.

The graphs seem approximately normally distributed for N >= 100 since for those N do the samples seem to have a linear relationship with the theoretical values of a normal distribution. This is consistent with what I would expect since for p=0.3, np and n(1-p) are greater than 10 only when n is greater than 34.

(3) Favorite food and Data Cleaning *(62 points. Parts 3.2 through 3.5, 2 pts each, other values listed below).*

This is data generated by former students. I simply asked "What is your favorite food?". You can get the data HERE.

Your goal is similar to what we did with the question "What animal would you like to be?" in Class 5: clean this variable, make a barplot, and discuss the results.

(3.1) *(1 pt)* Read in the data to a new object called `food`.

(3.2) Get a sense of the dataset - dimensions, variable names, look at the first few rows.

(3.3) Convert `food` to a single vector that is just the first column (literally, replace `food` with `food$Food`). if you need to, convert this value to character.

(3.4) Show the sorted unique values of `food`. Calculate how many unique values exist in `food`.

```
food <- read.csv("http://reuningscherer.net/S&DS230/data/food_230.csv")
head(food, 10)
```

```
##              Food
## 1   french fries.
## 2         Chicken
## 3  butter chicken
## 4          Ginger
## 5       Dal Bhaat
## 6       Thai food
## 7       chocolate
## 8         hot pot
## 9           sushi
## 10     Asian food
```

```
food <- as.character(food$Food)
sort(unique(food))
```

```
##   [1] "Albanian Food "
##   [2] "all kinds of delicious food"
##   [3] "amaretto dark chocolate"
##   [4] "any type of cheese"
##   [5] "Arepa"
##   [6] "Artichoke"
##   [7] "Asian cuisine"
##   [8] "asian food"
##   [9] "Asian food"
##  [10] "Bagel"
##  [11] "baguettes"
##  [12] "Bananas"
##  [13] "Blue Point Oyster"
##  [14] "brazilian chorizo pizzas"
##  [15] "Brazilian food."
##  [16] "Bread"
##  [17] "Burgers"
##  [18] "burritos"
##  [19] "Burritos"
##  [20] "butter chicken"
##  [21] "Cajun Fries"
##  [22] "cake"
##  [23] "Ceviche"
##  [24] "cheese"
##  [25] "Cheese"
##  [26] "Cheeseburgers"
##  [27] "cheez its"
##  [28] "chicken"
##  [29] "Chicken"
##  [30] "Chicken Malai Kabab"
##  [31] "Chicken parmesan and penne alla vodka"
##  [32] "chicken tenders"
##  [33] "Chicken Tikka and Naan"
##  [34] "Chicken Tikka Masala"
##  [35] "Chicken wings"
##  [36] "chinese"
##  [37] "Chinese food"
##  [38] "Chinese Food"
##  [39] "Chinese food is my favorite."
```

```
##  [40] "Chinese food "
##  [41] "chinese hotpot"
##  [42] "Chipotle"
##  [43] "chocolate"
##  [44] "chocolate chip cookies"
##  [45] "Chocolate\n\n \n\nA cat"
##  [46] "Cinnamon Rolls"
##  [47] "comfort food"
##  [48] "Cookies"
##  [49] "Corn"
##  [50] "Cottage Pie"
##  [51] "Crepes"
##  [52] "Curries of all sorts"
##  [53] "Curry"
##  [54] "Curry vindaloo"
##  [55] "Dal Bhaat"
##  [56] "Dandan noodles"
##  [57] "Delicious "
##  [58] "Dessert"
##  [59] "Donuts!"
##  [60] "Dried mangos"
##  [61] "dumplings"
##  [62] "Empanadas"
##  [63] "enchiladas"
##  [64] "Escargot in garlic butter"
##  [65] "Ethiopian Cuisine"
##  [66] "farofa"
##  [67] "Farofa "
##  [68] "Fish and chips"
##  [69] "Fish tacos"
##  [70] "Five guys' fries"
##  [71] "Flautas"
##  [72] "Freeze Dried"
##  [73] "french fries"
##  [74] "french fries."
##  [75] "French onion soup"
##  [76] "fried chicken"
##  [77] "Fried chicken"
##  [78] "Fried fish"
##  [79] "Fried okra"
##  [80] "Fried Rice"
##  [81] "fruit"
##  [82] "Fruit"
##  [83] "Fruits"
##  [84] "Ginger"
##  [85] "Gizzard (chicken) and vegetables (greens), Igbo dish"
##  [86] "gnocchi"
##  [87] "Good pizza"
##  [88] "Grilled chicken breast"
##  [89] "guacamole"
##  [90] "guacamole "
##  [91] "Gyros"
##  [92] "ham"
##  [93] "Hamburgers"
```

```
##  [94] "Hibachi"
##  [95] "Hot dog"
##  [96] "hot pot"
##  [97] "Hot pot"
##  [98] "hotpot"
##  [99] "I love Asian food, especially Chinese food, Thai food, and Sushi. \n\n "
## [100] "I love Korean cuisine."
## [101] "i love mexican food"
## [102] "ice cream"
## [103] "Ice cream"
## [104] "Ice Cream"
## [105] "ICE CREAM"
## [106] "Ice cream!!!"
## [107] "Ice cream. "
## [108] "indian"
## [109] "Indian"
## [110] "Indian food"
## [111] "Indian Food"
## [112] "Indian food is great."
## [113] "Indian Food "
## [114] "Indonesian food"
## [115] "Instant ramen"
## [116] "italian"
## [117] "Italian"
## [118] "italian food"
## [119] "Italian food"
## [120] "Italian Food"
## [121] "italian food - pasta"
## [122] "Italian food!"
## [123] "Japanese food"
## [124] "Japanese "
## [125] "Jollof Rice and Chicken"
## [126] "Jollof Rice with chicken "
## [127] "Kelewele, it's an African dish"
## [128] "Korean"
## [129] "Korean Barbeque"
## [130] "Korean BBQ"
## [131] "korean bbq!"
## [132] "korean bbq!!"
## [133] "korean food"
## [134] "Korean food"
## [135] "Korean Food"
## [136] "korean food "
## [137] "kosher Steak "
## [138] "lahmacun (turkish pizza)"
## [139] "Lasagna"
## [140] "Lasagna."
## [141] "Lasagne"
## [142] "Lebanese Food"
## [143] "mac and cheese"
## [144] "Macaroni and Cheese"
## [145] "mango"
## [146] "Meat"
## [147] "Mediterranean food "
```

```
## [148] "mediterranean "
## [149] "Mediterranean "
## [150] "Mexican"
## [151] "Mexican food"
## [152] "Mexican Food"
## [153] "Multigrain pancake"
## [154] "My favorite kind of food is pastries."
## [155] "nectarines"
## [156] "noodles"
## [157] "Noodles"
## [158] "noodles with broth"
## [159] "Noodles."
## [160] "palak paneer"
## [161] "pasta"
## [162] "Pasta"
## [163] "Patty Melt"
## [164] "Persian and Mediterranean"
## [165] "Pho"
## [166] "pizza"
## [167] "Pizza"
## [168] "Pizza!!!"
## [169] "platanos"
## [170] "Poke Bowl"
## [171] "Postickers"
## [172] "probably either casual diner fare, Italian, or shellfish"
## [173] "Puerto Rican food"
## [174] "ramen"
## [175] "Ramen"
## [176] "ramen\n\n "
## [177] "Ribs"
## [178] "rice with curry"
## [179] "Roast dinner "
## [180] "salmon"
## [181] "Salmon"
## [182] "salty"
## [183] "seafood"
## [184] "shahi paneer"
## [185] "sharp cheddar cheese"
## [186] "Shrimp curry"
## [187] "soup"
## [188] "Soup"
## [189] "South Asian Rice Dishes for example Biryani, Pulao, Mandi, Tahri "
## [190] "Spaghetti "
## [191] "Spicy"
## [192] "Spicy and flavorful food"
## [193] "spicy food "
## [194] "Spicy Hotpot"
## [195] "spicy tofu"
## [196] "steak"
## [197] "Steak"
## [198] "strawberries"
## [199] "Strawberries  "
## [200] "sundried tomatoes"
## [201] "sushi"
```

```
## [202] "Sushi"
## [203] "Sushi with an overwhelming amount of raw salmon"
## [204] "Swedish meatballs"
## [205] "Sweet chili Doritos"
## [206] "sweet potato"
## [207] "Tagine"
## [208] "Thai"
## [209] "thai food"
## [210] "Thai food"
## [211] "Thai Food"
## [212] "Thai food is my favorite kind of food "
## [213] "Thai food "
## [214] "Thai specifically beef pad see-ew and some rice."
## [215] "Tofu"
## [216] "udon noodle"
## [217] "Vietnamese food"
## [218] "Vietnamese spring rolls"
```

```
length(unique(food))
```

```
## [1] 218
```

(3.5) Write a couple of sentences about what data cleaning issues you notice amoung the unique values of `food`.

The capitalization isn't consistent, for example, "thai food" and "Thai food" are counted as unique values. There is also inconsistent spacing and punctuation. Some answers are too specific and thus count as their own value, such as "noodles with broth" as opposed to just "noodles". Some people prefaced their answer with "I love" or something similar.

(3.6) Cleaning Part I *(8 pts)*: Clean the data using the following steps (in order):

- Convert data to lower case
- Find " or " and remove this and anything that follows.
- Find " and " and remove this and anything that follows.
- Find " food" and remove this AND anything that follows.
- Find " cuisine" and remove this AND anything that follows.
- Remove all special characters and punctuation - see class 6.
- Remove trailing spaces at the end of text

At each step, you'll probably want to check what unique values of `food` are left to make sure your functions are working correctly. By the time you finish, you should have 156 unique levels.

Your final two lines of code should again show the sorted unique values of food and the current number of unique values.

```
food <- tolower(food)
food <- sort(food)
food <- gsub(" or.*", "", food)
food <- gsub(" and.*", "", food)
food <- gsub(" food.*", "", food)
food <- gsub(" cuisine.*", "", food)
food <- gsub("[^0-9A-Za-z///' ]", "" , food)
food <- gsub(" $", "" , food)
sort(unique(food))
```

```
##   [1] "albanian"
##   [2] "all kinds of delicious"
##   [3] "amaretto dark chocolate"
##   [4] "any type of cheese"
##   [5] "arepa"
##   [6] "artichoke"
##   [7] "asian"
##   [8] "bagel"
##   [9] "baguettes"
##  [10] "bananas"
##  [11] "blue point oyster"
##  [12] "brazilian"
##  [13] "brazilian chorizo pizzas"
##  [14] "bread"
##  [15] "burgers"
##  [16] "burritos"
##  [17] "butter chicken"
##  [18] "cajun fries"
##  [19] "cake"
##  [20] "ceviche"
##  [21] "cheese"
##  [22] "cheeseburgers"
##  [23] "cheez its"
##  [24] "chicken"
##  [25] "chicken malai kabab"
##  [26] "chicken parmesan"
##  [27] "chicken tenders"
##  [28] "chicken tikka"
##  [29] "chicken tikka masala"
##  [30] "chicken wings"
##  [31] "chinese"
##  [32] "chinese hotpot"
##  [33] "chipotle"
##  [34] "chocolate"
##  [35] "chocolate chip cookies"
##  [36] "chocolatea cat"
##  [37] "cinnamon rolls"
##  [38] "comfort"
##  [39] "cookies"
##  [40] "corn"
##  [41] "cottage pie"
##  [42] "crepes"
##  [43] "curries of all sorts"
##  [44] "curry"
##  [45] "curry vindaloo"
##  [46] "dal bhaat"
##  [47] "dandan noodles"
##  [48] "delicious"
##  [49] "dessert"
##  [50] "donuts"
##  [51] "dried mangos"
##  [52] "dumplings"
##  [53] "empanadas"
##  [54] "enchiladas"
```

```
##  [55] "escargot in garlic butter"
##  [56] "ethiopian"
##  [57] "farofa"
##  [58] "fish"
##  [59] "fish tacos"
##  [60] "five guys' fries"
##  [61] "flautas"
##  [62] "freeze dried"
##  [63] "french fries"
##  [64] "french onion soup"
##  [65] "fried chicken"
##  [66] "fried fish"
##  [67] "fried okra"
##  [68] "fried rice"
##  [69] "fruit"
##  [70] "fruits"
##  [71] "ginger"
##  [72] "gizzard chicken"
##  [73] "gnocchi"
##  [74] "good pizza"
##  [75] "grilled chicken breast"
##  [76] "guacamole"
##  [77] "gyros"
##  [78] "ham"
##  [79] "hamburgers"
##  [80] "hibachi"
##  [81] "hot dog"
##  [82] "hot pot"
##  [83] "hotpot"
##  [84] "i love asian"
##  [85] "i love korean"
##  [86] "i love mexican"
##  [87] "ice cream"
##  [88] "indian"
##  [89] "indonesian"
##  [90] "instant ramen"
##  [91] "italian"
##  [92] "japanese"
##  [93] "jollof rice"
##  [94] "jollof rice with chicken"
##  [95] "kelewele it's an african dish"
##  [96] "korean"
##  [97] "korean barbeque"
##  [98] "korean bbq"
##  [99] "kosher steak"
## [100] "lahmacun turkish pizza"
## [101] "lasagna"
## [102] "lasagne"
## [103] "lebanese"
## [104] "mac"
## [105] "macaroni"
## [106] "mango"
## [107] "meat"
## [108] "mediterranean"
```

```
## [109] "mexican"
## [110] "multigrain pancake"
## [111] "my favorite kind of"
## [112] "nectarines"
## [113] "noodles"
## [114] "noodles with broth"
## [115] "palak paneer"
## [116] "pasta"
## [117] "patty melt"
## [118] "persian"
## [119] "pho"
## [120] "pizza"
## [121] "platanos"
## [122] "poke bowl"
## [123] "postickers"
## [124] "probably either casual diner fare italian"
## [125] "puerto rican"
## [126] "ramen"
## [127] "ribs"
## [128] "rice with curry"
## [129] "roast dinner"
## [130] "salmon"
## [131] "salty"
## [132] "seafood"
## [133] "shahi paneer"
## [134] "sharp cheddar cheese"
## [135] "shrimp curry"
## [136] "soup"
## [137] "south asian rice dishes for example biryani pulao mandi tahri"
## [138] "spaghetti"
## [139] "spicy"
## [140] "spicy hotpot"
## [141] "spicy tofu"
## [142] "steak"
## [143] "strawberries"
## [144] "sundried tomatoes"
## [145] "sushi"
## [146] "sushi with an overwhelming amount of raw salmon"
## [147] "swedish meatballs"
## [148] "sweet chili doritos"
## [149] "sweet potato"
## [150] "tagine"
## [151] "thai"
## [152] "thai specifically beef pad seeew"
## [153] "tofu"
## [154] "udon noodle"
## [155] "vietnamese"
## [156] "vietnamese spring rolls"
```

```
length(unique(food))
```

```
## [1] 156
```

(3.7) Cleaning Part II *(10 pts)*: A few quick random cleaning items:

Clean up the following types of food (in order) - one line of code per type of food. In each case, deal with misspellings, modifiers ("shrimp curry" vs just "curry"), two words ('hot pot' instead of 'hotpot'), plurals, etc.

- hotpot
- curry
- lasagna
- noodles
- cookies
- chocolate
- cheese
- steak
- sushi
- fries (french, cajun, five guys' should all be 'fries')
- ramen
- tofu
- burgers (of any kind)
- soup
- anything containing 'delicious' just call 'delicious'

When you're finished, you should have 130 unique values.

Your final two lines of code should again show the sorted unique values of food and the current number of unique values.

```
food <- gsub(".*hot\\s?pot.*", "hotpot", food)
food <- gsub(".*curr.*", "curry", food)
food <- gsub("lasagne", "lasagna", food)
food <- gsub(".*noodle.*", "noodles", food)
food <- gsub(".*cookies.*", "cookies", food)
food <- gsub(".*chocolate.*", "chocolate", food)
food <- gsub(".*cheese\\b", "cheese", food)
food <- gsub(".*steak.*", "steak", food)
food <- gsub(".*sushi.*", "sushi", food)
food <- gsub(".*fries.*", "fries", food)
food <- gsub(".*ramen.*", "ramen", food)
food <- gsub(".*tofu.*", "tofu", food)
food <- gsub(".*burgers.*", "burgers", food)
food <- gsub(".*soup.*", "soup", food)
food <- gsub(".*delicious.*", "delicious", food)
sort(unique(food))
```

```
##   [1] "albanian"
##   [2] "arepa"
##   [3] "artichoke"
##   [4] "asian"
##   [5] "bagel"
##   [6] "baguettes"
##   [7] "bananas"
##   [8] "blue point oyster"
##   [9] "brazilian"
##  [10] "brazilian chorizo pizzas"
##  [11] "bread"
##  [12] "burgers"
```

```
##  [13] "burritos"
##  [14] "butter chicken"
##  [15] "cake"
##  [16] "ceviche"
##  [17] "cheese"
##  [18] "cheez its"
##  [19] "chicken"
##  [20] "chicken malai kabab"
##  [21] "chicken parmesan"
##  [22] "chicken tenders"
##  [23] "chicken tikka"
##  [24] "chicken tikka masala"
##  [25] "chicken wings"
##  [26] "chinese"
##  [27] "chipotle"
##  [28] "chocolate"
##  [29] "cinnamon rolls"
##  [30] "comfort"
##  [31] "cookies"
##  [32] "corn"
##  [33] "cottage pie"
##  [34] "crepes"
##  [35] "curry"
##  [36] "dal bhaat"
##  [37] "delicious"
##  [38] "dessert"
##  [39] "donuts"
##  [40] "dried mangos"
##  [41] "dumplings"
##  [42] "empanadas"
##  [43] "enchiladas"
##  [44] "escargot in garlic butter"
##  [45] "ethiopian"
##  [46] "farofa"
##  [47] "fish"
##  [48] "fish tacos"
##  [49] "flautas"
##  [50] "freeze dried"
##  [51] "fried chicken"
##  [52] "fried fish"
##  [53] "fried okra"
##  [54] "fried rice"
##  [55] "fries"
##  [56] "fruit"
##  [57] "fruits"
##  [58] "ginger"
##  [59] "gizzard chicken"
##  [60] "gnocchi"
##  [61] "good pizza"
##  [62] "grilled chicken breast"
##  [63] "guacamole"
##  [64] "gyros"
##  [65] "ham"
##  [66] "hibachi"
```

```
##  [67] "hot dog"
##  [68] "hotpot"
##  [69] "i love asian"
##  [70] "i love korean"
##  [71] "i love mexican"
##  [72] "ice cream"
##  [73] "indian"
##  [74] "indonesian"
##  [75] "italian"
##  [76] "japanese"
##  [77] "jollof rice"
##  [78] "jollof rice with chicken"
##  [79] "kelewele it's an african dish"
##  [80] "korean"
##  [81] "korean barbeque"
##  [82] "korean bbq"
##  [83] "lahmacun turkish pizza"
##  [84] "lasagna"
##  [85] "lebanese"
##  [86] "mac"
##  [87] "macaroni"
##  [88] "mango"
##  [89] "meat"
##  [90] "mediterranean"
##  [91] "mexican"
##  [92] "multigrain pancake"
##  [93] "my favorite kind of"
##  [94] "nectarines"
##  [95] "noodles"
##  [96] "palak paneer"
##  [97] "pasta"
##  [98] "patty melt"
##  [99] "persian"
## [100] "pho"
## [101] "pizza"
## [102] "platanos"
## [103] "poke bowl"
## [104] "postickers"
## [105] "probably either casual diner fare italian"
## [106] "puerto rican"
## [107] "ramen"
## [108] "ribs"
## [109] "roast dinner"
## [110] "salmon"
## [111] "salty"
## [112] "seafood"
## [113] "shahi paneer"
## [114] "soup"
## [115] "south asian rice dishes for example biryani pulao mandi tahri"
## [116] "spaghetti"
## [117] "spicy"
## [118] "steak"
## [119] "strawberries"
## [120] "sundried tomatoes"
```

```
## [121] "sushi"
## [122] "swedish meatballs"
## [123] "sweet chili doritos"
## [124] "sweet potato"
## [125] "tagine"
## [126] "thai"
## [127] "thai specifically beef pad seeew"
## [128] "tofu"
## [129] "vietnamese"
## [130] "vietnamese spring rolls"
```

```
length(unique(food))
```

```
## [1] 130
```

(3.8) Cleaning Part III *(8 pts)*: Cleaning types of cuisine.

Clean up the following types of cuisine (in order) - in this case, you'll want to make a vector called `searchvec` that contains the types of cuisine. Then create a loop following the example in Class 5 to replace all the modifiers for each cuisine type so that you ultimately end up with cleaned up versions of each cuisine type. Use not more than 5 lines of code.

The cuisine types (in order) are * asian * chinese * vietnamese * italian * indian * thai * mexican * brazilian * korean

(there are other types of cuisine, but they don't require cleaning).

When you're finished, you should have 120 unique values.

Your final two lines of code should again show the sorted unique values of food and the current number of unique values.

```
searchvec <- c("asian", "chinese", "vietnamese", "italian",
               "indian", "thai", "mexican", "brazilian", "korean")
for (i in 1:length(searchvec)) {
  food <- gsub(paste0(".*", searchvec[i] ,".*"), searchvec[i], food)
}
sort(unique(food))
```

```
##   [1] "albanian"                    "arepa"
##   [3] "artichoke"                   "asian"
##   [5] "bagel"                       "baguettes"
##   [7] "bananas"                     "blue point oyster"
##   [9] "brazilian"                   "bread"
##  [11] "burgers"                     "burritos"
##  [13] "butter chicken"              "cake"
##  [15] "ceviche"                     "cheese"
##  [17] "cheez its"                   "chicken"
##  [19] "chicken malai kabab"         "chicken parmesan"
##  [21] "chicken tenders"             "chicken tikka"
##  [23] "chicken tikka masala"        "chicken wings"
##  [25] "chinese"                     "chipotle"
##  [27] "chocolate"                   "cinnamon rolls"
##  [29] "comfort"                     "cookies"
##  [31] "corn"                        "cottage pie"
```

```
##  [33] "crepes"                         "curry"
##  [35] "dal bhaat"                      "delicious"
##  [37] "dessert"                        "donuts"
##  [39] "dried mangos"                   "dumplings"
##  [41] "empanadas"                      "enchiladas"
##  [43] "escargot in garlic butter"      "ethiopian"
##  [45] "farofa"                         "fish"
##  [47] "fish tacos"                     "flautas"
##  [49] "freeze dried"                   "fried chicken"
##  [51] "fried fish"                     "fried okra"
##  [53] "fried rice"                     "fries"
##  [55] "fruit"                          "fruits"
##  [57] "ginger"                         "gizzard chicken"
##  [59] "gnocchi"                        "good pizza"
##  [61] "grilled chicken breast"         "guacamole"
##  [63] "gyros"                          "ham"
##  [65] "hibachi"                        "hot dog"
##  [67] "hotpot"                         "ice cream"
##  [69] "indian"                         "indonesian"
##  [71] "italian"                        "japanese"
##  [73] "jollof rice"                    "jollof rice with chicken"
##  [75] "kelewele it's an african dish" "korean"
##  [77] "lahmacun turkish pizza"         "lasagna"
##  [79] "lebanese"                       "mac"
##  [81] "macaroni"                       "mango"
##  [83] "meat"                           "mediterranean"
##  [85] "mexican"                        "multigrain pancake"
##  [87] "my favorite kind of"            "nectarines"
##  [89] "noodles"                        "palak paneer"
##  [91] "pasta"                          "patty melt"
##  [93] "persian"                        "pho"
##  [95] "pizza"                          "platanos"
##  [97] "poke bowl"                      "postickers"
##  [99] "puerto rican"                   "ramen"
## [101] "ribs"                           "roast dinner"
## [103] "salmon"                         "salty"
## [105] "seafood"                        "shahi paneer"
## [107] "soup"                           "spaghetti"
## [109] "spicy"                          "steak"
## [111] "strawberries"                   "sundried tomatoes"
## [113] "sushi"                          "swedish meatballs"
## [115] "sweet chili doritos"            "sweet potato"
## [117] "tagine"                         "thai"
## [119] "tofu"                           "vietnamese"
```

```
length(unique(food))
```

```
## [1] 120
```

(3.9) *(15 pts)* Following the example from Class 05, display a dataframe of the sorted tabular results of `food` to see how many individuals prefer each kind of food.

From here on, the decisions of how to clean and combine categories are yours! Any food that currently has a count of 3 or more should remain (you can add to these categories - for example, you could add 'lasagna'

to 'italian' or to 'pasta'). All other levels should be recoded or incorporated into a 'miscellaneous' food category. Points awarded based on thoughtfulness, effort, and quality/preciseness of your code.

Include your code below, and add comments where appropriate to describe the choices you make. You should have no more than 40 levels by the time you finish.

Display a dataframe of the sorted tabular results of `food` to see how many individuals prefer each kind of food AGAIN after you've finished your coding.

```
food <- gsub(".*fish.*|salmon", "fish", food)
food <- gsub(".*chicken.*", "chicken", food)
food <- gsub(".*pizza.*", "pizza", food)
food <- gsub(".*fried.*", "fried foods", food)
food <- gsub(".*fruit.*|strawberries|bananas|.*mango.*|oranges|berries|nectarines", "fruit", food)
food <- gsub(".*bread.*|.*cake.*|.*donut.*|.*bagel.*|.*baguette.*|.*cookies.*", "pastries", food)
food <- gsub("burritos", "mexican", food)
table1 <- data.frame(sort(table(food),decreasing=T))
table1
```

```
##                          food Freq
## 1                       sushi   24
## 2                       pizza   19
## 3                     chicken   15
## 4                      korean   14
## 5                        thai   14
## 6                     chinese   13
## 7                     mexican   12
## 8                       fruit   10
## 9                   ice cream   10
## 10                     indian    9
## 11                    italian    9
## 12                   pastries    9
## 13                      ramen    9
## 14                    noodles    8
## 15                      pasta    8
## 16                      steak    8
## 17                  chocolate    6
## 18                     hotpot    6
## 19                      asian    5
## 20                      curry    5
## 21                       fish    5
## 22                       soup    5
## 23                    burgers    4
## 24                     cheese    4
## 25                      fries    4
## 26                   japanese    3
## 27                    lasagna    3
## 28              mediterranean    3
## 29                        pho    3
## 30                      spicy    3
## 31                  brazilian    2
## 32                  delicious    2
## 33                     farofa    2
## 34                fried foods    2
## 35                  guacamole    2
```

```
## 36                         tofu    2
## 37                   vietnamese    2
## 38                     albanian    1
## 39                        arepa    1
## 40                    artichoke    1
## 41            blue point oyster    1
## 42                      ceviche    1
## 43                    cheez its    1
## 44                     chipotle    1
## 45                cinnamon rolls    1
## 46                      comfort    1
## 47                         corn    1
## 48                  cottage pie    1
## 49                       crepes    1
## 50                    dal bhaat    1
## 51                      dessert    1
## 52                    dumplings    1
## 53                    empanadas    1
## 54                   enchiladas    1
## 55    escargot in garlic butter    1
## 56                    ethiopian    1
## 57                      flautas    1
## 58                  freeze dried    1
## 59                        ginger    1
## 60                       gnocchi    1
## 61                         gyros    1
## 62                           ham    1
## 63                       hibachi    1
## 64                       hot dog    1
## 65                    indonesian    1
## 66                   jollof rice    1
## 67 kelewele it's an african dish    1
## 68                      lebanese    1
## 69                           mac    1
## 70                      macaroni    1
## 71                          meat    1
## 72          my favorite kind of    1
## 73                 palak paneer    1
## 74                   patty melt    1
## 75                      persian    1
## 76                     platanos    1
## 77                    poke bowl    1
## 78                    postickers    1
## 79                  puerto rican    1
## 80                          ribs    1
## 81                  roast dinner    1
## 82                         salty    1
## 83                       seafood    1
## 84                  shahi paneer    1
## 85                     spaghetti    1
## 86            sundried tomatoes    1
## 87            swedish meatballs    1
## 88          sweet chili doritos    1
## 89                 sweet potato    1
```

```
## 90                        tagine    1
```

```
for (i in 38:90){ # misc
  food <- gsub(paste0("^", as.character(table1[i, 1]), "$"), "other", food)
}
table1 <- data.frame(sort(table(food),decreasing=T))
table1
```

```
##             food Freq
## 1          other   53
## 2          sushi   24
## 3          pizza   19
## 4        chicken   15
## 5         korean   14
## 6           thai   14
## 7        chinese   13
## 8        mexican   12
## 9          fruit   10
## 10     ice cream   10
## 11        indian    9
## 12       italian    9
## 13      pastries    9
## 14         ramen    9
## 15       noodles    8
## 16         pasta    8
## 17         steak    8
## 18     chocolate    6
## 19        hotpot    6
## 20         asian    5
## 21         curry    5
## 22          fish    5
## 23          soup    5
## 24       burgers    4
## 25        cheese    4
## 26         fries    4
## 27      japanese    3
## 28       lasagna    3
## 29 mediterranean    3
## 30           pho    3
## 31         spicy    3
## 32     brazilian    2
## 33     delicious    2
## 34        farofa    2
## 35   fried foods    2
## 36     guacamole    2
## 37          tofu    2
## 38    vietnamese    2
```

(3.10) *(8 pts)* Final steps and a plot: You'll want to CAREFULLY follow the example in the code at the end of Class 05.

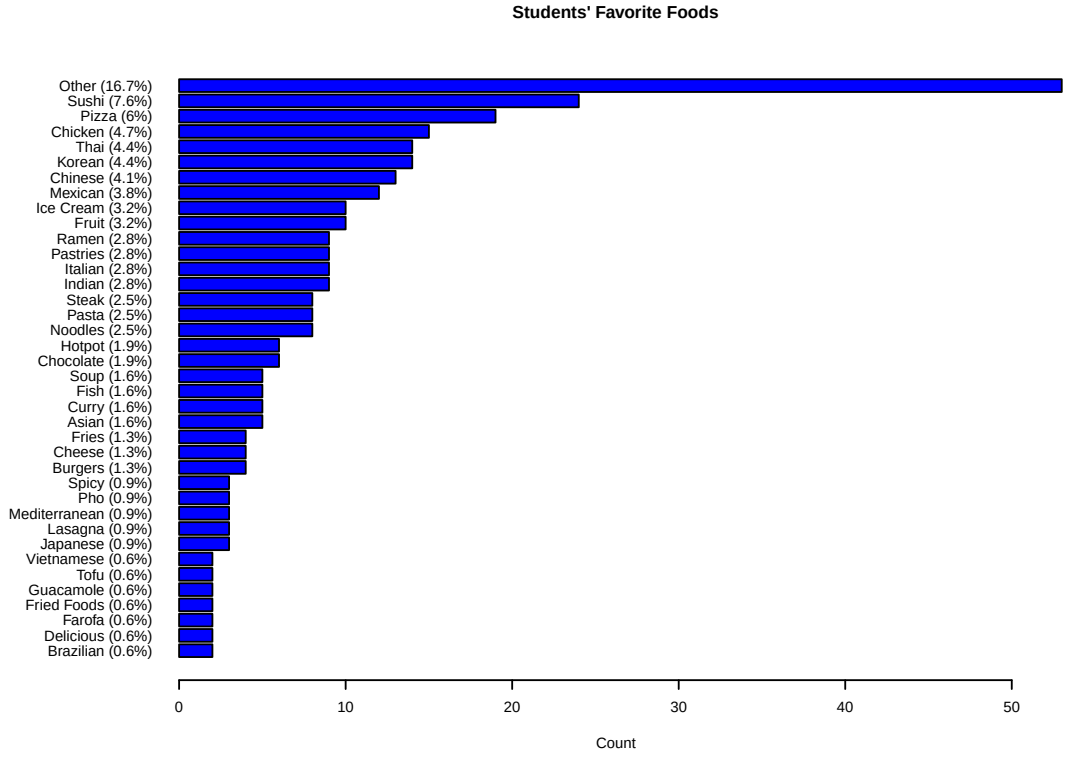
- Use the `toTitleCase()` function from the package `tools` to convert food to title case.
- Make an object called `finaltab` that is a table of your final vector `food`.

- Calculate percents, rounded to the nearest integer, for each food type. Save this as an object called `percents`.
- Change the names of `finaltab` to include a space and then the percents followed by a "%" in curved parentheses.
- Make a horizontal barplot of your final plot. Choose a nice bar color, adjust the left margins as necessary, give a main title and label the horizontal axis.

```
library(tools)
food <- toTitleCase(food)
finaltab <- sort(table(food), decreasing=FALSE)
par(mar = c(5, 9, 4, 2), cex = .6)
percents <- round(finaltab/sum(finaltab)*100, 1)
names(finaltab) <- paste0(names(finaltab)," (", percents, "%)")
finaltab
```

```
##     Brazilian (0.6%)     Delicious (0.6%)        Farofa (0.6%)
##                    2                    2                    2
##   Fried Foods (0.6%)     Guacamole (0.6%)          Tofu (0.6%)
##                    2                    2                    2
##    Vietnamese (0.6%)      Japanese (0.9%)       Lasagna (0.9%)
##                    2                    3                    3
## Mediterranean (0.9%)           Pho (0.9%)         Spicy (0.9%)
##                    3                    3                    3
##       Burgers (1.3%)        Cheese (1.3%)         Fries (1.3%)
##                    4                    4                    4
##         Asian (1.6%)         Curry (1.6%)          Fish (1.6%)
##                    5                    5                    5
##          Soup (1.6%)     Chocolate (1.9%)        Hotpot (1.9%)
##                    5                    6                    6
##       Noodles (2.5%)         Pasta (2.5%)         Steak (2.5%)
##                    8                    8                    8
##        Indian (2.8%)       Italian (2.8%)      Pastries (2.8%)
##                    9                    9                    9
##         Ramen (2.8%)         Fruit (3.2%)     Ice Cream (3.2%)
##                    9                   10                   10
##       Mexican (3.8%)       Chinese (4.1%)        Korean (4.4%)
##                   12                   13                   14
##          Thai (4.4%)       Chicken (4.7%)           Pizza (6%)
##                   14                   15                   19
##         Sushi (7.6%)        Other (16.7%)
##                   24                   53
```

```
par(mar = c(10, 10, 5, 10), cex = 0.5)
barplot(finaltab,
        horiz = T,
        las = 1,
        col = "blue",
        main = "Students' Favorite Foods",
        xlab = "Count")
```



(3.11) *(3 pts)* In no more than three sentences, discuss your process and results. Be sure to mention how many unique values of 'food' you started and ended with. Any surprises?

By coalescing together similar responses, we took 218 unique food values, and reduced them down to 38, including "other". It was surprising that people wrote essentially variations of the same type of food item. It was also a little surprising that people decided to put down the names of countries/cuisine rather than a specific food item.