Halloween Candy Data

Stewart Donaldson

15/11/2019

1. Introduction to the dataset

The Boing Boing Halloween Candy Survey has been running since 2014 and is designed to find out what is the top preferred confectionary is during trick or treat season.

For the analysis I looked at the data collected from 2015, 2016 and 2017, which can be found in these files below:

`boing-boing-candy-2015.xlsx`

`boing-boing-candy-2016.xlsx`

`boing-boing-candy-2015.xlsx`

2. Assumptions

1. Any questions not related to candy where excluded from the data set as it was assumed these weren't relevant to the purpose of the analysis (i.e what was the most preferred candy)
2. Free form questions such as "Please list any items not included above that give you JOY" were excluded from the data. The reasons for this were:

- Time contraints pulling out specific candy from the free form answers
- The assumption was that the questions asked covered most candy in the market

3. Items like "Real Housewives of Orange County Season 9 Blu Ray", while not strictly candy, were left in the data set as they appeared to be asked in conjuction with the candy questions and appeared some what relevant to the analysis.

4. Steps to clean the data

As each data set was different in terms of the naming of columns and the order in which the columns were presented, it was decided that the best approach was to clean each data set seperately and pull each into a unified format before joining them all together.

Each step below was combined into a pipe `%>%` to make the code **easier to understand**, **reproducible** and **help with processing power.**

Pre cleaning checks

- Checking the dimensions

```
dim(boing_boing_2017)
```

```
## [1] 2460  120
```

- Checking the column names

```
names(boing_boing_2017)
```

```
##   [1] "Internal ID"
##   [2] "Q1: GOING OUT?"
##   [3] "Q2: GENDER"
##   [4] "Q3: AGE"
##   [5] "Q4: COUNTRY"
##   [6] "Q5: STATE, PROVINCE, COUNTY, ETC"
##   [7] "Q6 | 100 Grand Bar"
##   [8] "Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)"
##   [9] "Q6 | Any full-sized candy bar"
##  [10] "Q6 | Black Jacks"
##  [11] "Q6 | Bonkers (the candy)"
##  [12] "Q6 | Bonkers (the board game)"
##  [13] "Q6 | Bottle Caps"
##  [14] "Q6 | Box'o'Raisins"
##  [15] "Q6 | Broken glow stick"
##  [16] "Q6 | Butterfinger"
##  [17] "Q6 | Cadbury Creme Eggs"
##  [18] "Q6 | Candy Corn"
##  [19] "Q6 | Candy that is clearly just the stuff given out for free at restaurants"
##  [20] "Q6 | Caramellos"
##  [21] "Q6 | Cash, or other forms of legal tender"
##  [22] "Q6 | Chardonnay"
##  [23] "Q6 | Chick-o-Sticks (we don’t know what that is)"
##  [24] "Q6 | Chiclets"
##  [25] "Q6 | Coffee Crisp"
##  [26] "Q6 | Creepy Religious comics/Chick Tracts"
##  [27] "Q6 | Dental paraphenalia"
##  [28] "Q6 | Dots"
##  [29] "Q6 | Dove Bars"
##  [30] "Q6 | Fuzzy Peaches"
##  [31] "Q6 | Generic Brand Acetaminophen"
##  [32] "Q6 | Glow sticks"
##  [33] "Q6 | Goo Goo Clusters"
##  [34] "Q6 | Good N' Plenty"
##  [35] "Q6 | Gum from baseball cards"
##  [36] "Q6 | Gummy Bears straight up"
##  [37] "Q6 | Hard Candy"
##  [38] "Q6 | Healthy Fruit"
##  [39] "Q6 | Heath Bar"
##  [40] "Q6 | Hershey's Dark Chocolate"
##  [41] "Q6 | Hershey’s Milk Chocolate"
##  [42] "Q6 | Hershey's Kisses"
##  [43] "Q6 | Hugs (actual physical hugs)"
##  [44] "Q6 | Jolly Rancher (bad flavor)"
##  [45] "Q6 | Jolly Ranchers (good flavor)"
##  [46] "Q6 | JoyJoy (Mit Iodine!)"
##  [47] "Q6 | Junior Mints"
##  [48] "Q6 | Senior Mints"
##  [49] "Q6 | Kale smoothie"
##  [50] "Q6 | Kinder Happy Hippo"
##  [51] "Q6 | Kit Kat"
##  [52] "Q6 | LaffyTaffy"
##  [53] "Q6 | LemonHeads"
##  [54] "Q6 | Licorice (not black)"
##  [55] "Q6 | Licorice (yes black)"
##  [56] "Q6 | Lindt Truffle"
##  [57] "Q6 | Lollipops"
##  [58] "Q6 | Mars"
##  [59] "Q6 | Maynards"
##  [60] "Q6 | Mike and Ike"
##  [61] "Q6 | Milk Duds"
##  [62] "Q6 | Milky Way"
##  [63] "Q6 | Regular M&Ms"
##  [64] "Q6 | Peanut M&M’s"
##  [65] "Q6 | Blue M&M's"
##  [66] "Q6 | Red M&M's"
##  [67] "Q6 | Green Party M&M's"
##  [68] "Q6 | Independent M&M's"
##  [69] "Q6 | Abstained from M&M'ing."
##  [70] "Q6 | Minibags of chips"
##  [71] "Q6 | Mint Kisses"
##  [72] "Q6 | Mint Juleps"
##  [73] "Q6 | Mr. Goodbar"
##  [74] "Q6 | Necco Wafers"
##  [75] "Q6 | Nerds"
##  [76] "Q6 | Nestle Crunch"
##  [77] "Q6 | Now'n'Laters"
##  [78] "Q6 | Peeps"
##  [79] "Q6 | Pencils"
##  [80] "Q6 | Pixy Stix"
##  [81] "Q6 | Real Housewives of Orange County Season 9 Blue-Ray"
##  [82] "Q6 | Reese’s Peanut Butter Cups"
##  [83] "Q6 | Reese's Pieces"
##  [84] "Q6 | Reggie Jackson Bar"
##  [85] "Q6 | Rolos"
##  [86] "Q6 | Sandwich-sized bags filled with BooBerry Crunch"
##  [87] "Q6 | Skittles"
##  [88] "Q6 | Smarties (American)"
##  [89] "Q6 | Smarties (Commonwealth)"
##  [90] "Q6 | Snickers"
##  [91] "Q6 | Sourpatch Kids (i.e. abominations of nature)"
##  [92] "Q6 | Spotted Dick"
##  [93] "Q6 | Starburst"
##  [94] "Q6 | Sweet Tarts"
##  [95] "Q6 | Swedish Fish"
##  [96] "Q6 | Sweetums (a friend to diabetes)"
##  [97] "Q6 | Take 5"
##  [98] "Q6 | Tic Tacs"
##  [99] "Q6 | Those odd marshmallow circus peanut things"
## [100] "Q6 | Three Musketeers"
## [101] "Q6 | Tolberone something or other"
## [102] "Q6 | Trail Mix"
## [103] "Q6 | Twix"
## [104] "Q6 | Vials of pure high fructose corn syrup, for main-lining into your vein"
## [105] "Q6 | Vicodin"
## [106] "Q6 | Whatchamacallit Bars"
## [107] "Q6 | White Bread"
## [108] "Q6 | Whole Wheat anything"
## [109] "Q6 | York Peppermint Patties"
## [110] "Q7: JOY OTHER"
## [111] "Q8: DESPAIR OTHER"
## [112] "Q9: OTHER COMMENTS"
## [113] "Q10: DRESS"
## [114] "...114"
## [115] "Q11: DAY"
## [116] "Q12: MEDIA [Daily Dish]"
## [117] "Q12: MEDIA [Science]"
## [118] "Q12: MEDIA [ESPN]"
## [119] "Q12: MEDIA [Yahoo]"
## [120] "Click Coordinates (x, y)"
```

- Checking the structure of the date

```
view(boing_boing_2017)
```

1. Cleaning column names

After loading in the data the first step was the clean the column names.

The reason for this was to make it easier for the next step when it came to deselecting the columns I didn't want and using the prepopulate option to return the name when I pressed the tab key. Not doing this would have resulted in a lot of time being wasted copying and pasting the names and putting them into quotation marks to prevent R from picking these characters up as code syntax.

```
# Cleaning column names in boing boing 2015 data
boing_boing_2015_clean_names <- clean_names(boing_boing_2015)
```

2. Removing the columns that are not needed for the analysis

The next step was removing the columns that were not relevant to the analysis questions being asked.

These columns involved any free form questions and questions that had nothing to do with candy

```
# Cleaning boing boing 2015 data

boing_boing_2015_clean <- boing_boing_2015_clean_names %>%

# Removing all columns that are not needed for the analysis
  select(
    -timestamp,
    -please_leave_any_remarks_or_comments_regarding_your_choices,
    -please_list_any_items_not_included_above_that_give_you_joy,
    -please_list_any_items_not_included_above_that_give_you_despair,
    -guess_the_number_of_mints_in_my_hand,
    -betty_or_veronica,
    -check_all_that_apply_i_cried_tears_of_sadness_at_the_end_of,
    -that_dress_that_went_viral_early_this_year_when_i_first_saw_it_it_was,
    -fill_in_the_blank_taylor_swift_is_a_force_for,
    -what_is_your_favourite_font,
    -if_you_squint_really_hard_the_words_intelligent_design_would_look_like,
    -fill_in_the_blank_imitation_is_a_form_of,
    -which_day_do_you_prefer_friday_or_sunday,
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_jk_rowling,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_jk_rowling,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_malala_yousafzai,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_thom_yorke,
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_jj_abrams,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_jj_abrams,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_hillary_clinton,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_donald_trump,
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_bieber,
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_kevin_bacon,
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_francis_bacon_15
    -please_estimate_the_degree_s_of_separation_you_have_from_the_following_celebrities_beyonce,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_beyonce_knowles,
    -please_estimate_the_degrees_of_separation_you_have_from_the_following_folks_bruce_lee
  ) %>%
```

3. Reshaping data into longer format

Once the column names had been appropriately formatted and irrelevant columns removed, this allowed for the candy question columns to be converted into long format using `pivot_longer`

Here I specified the first and last columns I wanted to be reshaped using the argument `col = butterfinger:necco_wafers`

I then named this new column "questions" using the `names_to` argument.

And, finally created a new column for the results using the `values_to` argument and called the column "response"

```
# Reshaping data into longer format
  pivot_longer(
    col = butterfinger:necco_wafers,
    names_to = "questions",
    values_to = "response"
    ) %>%
```

4. Adding in columns to help join data sets together

As 2016 and 2017 both had a country and gender column, I decided to add this to the 2015 data so it made it easier to bind all three data sets together. I did this using the `add_column()` function.

I also added in a **"year"** column to help identify what data was connected to what survey year.

```
# Creating three new columns:
      #1) year - to identify what year each data is from when it is joined together
      #2) country - pre-empting the columns that will be needed when 2016 and 2017 data is joined (whic
      #3) gender - same as above pre_empting the gender data that will be joined on from the 2016 and 2
```

```
add_column(
    gender = NA,
    country = NA,
    year = 2015
    ) %>%
```

5. Renaming columns to better naming conventions

To ensure the column names where in line with the style guidelines covered in week one, I renamed the columns gender, age and country columns for all three data sets, as well as, the question that asked whether or not the respondent was going trick or treating.

```
rename(
    trick_or_treating = "are_you_going_actually_going_trick_or_treating_yourself",
    gender = "your_gender",
    age = "how_old_are_you",
    country = "which_country_do_you_live_in"
    ) %>%
```

6. Reshaping data into a unified format

Now that I had added in the gender, and country columns to the 2015 data set, I used the `select()` function to order my columns into the same order that the 2016 and 2017 data would look like once I removed all columns that weren't needed for the analysis.

The reason again for this was to make joing all three data sets simplier at the end.

```
# Get columns in the correct order as 2016 and 2017 data so the join will be easier
  select(
    trick_or_treating,
    gender,
    age,
    country,
    questions,
    response, year
    ) %>%
```

7. Formatting fields

Again to align to the style guidelines covered in week 1 and to cover off basic number formatting issues with the age column, I used a combination of `mutate()` and `str_to_lower()` or `as.integer` to fix these fields.

Coverting all string fields to lower first was motivated by the fact that the country column was particularly messy with a number of country spellings including randomly placed capital letters. By coverting strings to lower case first this massively helped when it came to hardcoding different identifiers for grouping different spellings under a single country name.

```
# Tidying up the trick_or_treating, gender, age, country, and response columns so all numbers are whole

  mutate(
    trick_or_treating = str_to_lower(trick_or_treating),
    gender = str_to_lower(gender),
    age = as.integer(age),
    country = str_to_lower(country),
    response = str_to_lower(response)
    ) %>%
```

8. Grouping different country spellings/misspellings

I used a combination of `mutate()`, `str_replace_all`, `if_else()` and `Regex` to capture all the different variations of how respondents wrote the country they were from.

```
# Finding all the different variations of the country and grouping them by a single name
 mutate(country = str_replace_all(country, "usa(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "united s(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "u.s(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "merica(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "murica(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "ausa", "usa")) %>%
 mutate(country = str_replace_all(country, "a tropical island south of the equator", NA_character_)) %
 mutate(country = str_replace_all(country, "united kindom", "uk")) %>%
 mutate(country = str_replace_all(country, "neverland", NA_character_)) %>%
 mutate(country = str_replace_all(country, "this one", "usa")) %>%
 mutate(country = str_replace_all(country, "units states", "usa")) %>%
 mutate(country = str_replace_all(country, "the best one - usa", "usa")) %>%
 mutate(country = str_replace_all(country, "cascadia", "usa")) %>%
 mutate(country = str_replace_all(country, "the yoo ess of aaayyyyyy", "usa")) %>%
 mutate(country = str_replace_all(country, "somewhere", NA_character_)) %>%
 mutate(country = str_replace_all(country, "god's country", NA_character_)) %>%
 mutate(country = str_replace_all(country, "there isn't one for old men", NA_character_)) %>%
 mutate(country = str_replace_all(country, "eua", "usa")) %>%
 mutate(country = str_replace_all(country, "merca(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "españa", "spain")) %>%
 mutate(country = str_replace_all(country, "(.*)usa", "usa")) %>%
 mutate(country = str_replace_all(country, "see above", NA_character_)) %>%
 mutate(country = str_replace_all(country, "denial", NA_character_)) %>%
 mutate(country = str_replace_all(country, "usaa", "usa")) %>%
 mutate(country = str_replace_all(country, "^([1-9][0-9].*)$", NA_character_)) %>%
 mutate(country = str_replace_all(country, "one of the best ones", NA_character_)) %>%
 mutate(country = str_replace_all(country, "trump(.*)", "usa")) %>%
 mutate(country = str_replace_all(country, "the neatherlands", "netherlands")) %>%
 mutate(country = if_else(country == "us", NA_character_, country)) %>%
 mutate(country = str_replace_all(country, "^a$", NA_character_)) %>%
 mutate(country = str_replace_all(country, "^1$", NA_character_)) %>%
 mutate(country = if_else(country == "ud", NA_character_, country))
```

Using the `unique()` also helped identify all the misspellings and check which countries still needed rules to capture errors

```
unique(boing_boing_2017_clean$country)
```

9. Joining all three data sets together and writing to CSV

Once all three data sets were cleaned the final step was to join them all together using the `bind_row` function and assigning the final data set to the variable name **"boing__boing__data__combined"**

From here, I then wrote the final clean data set to a csv to then perform my analysis on

```
# Binding 2016 and 2017 data onto 2015 data
boing_boing_data_combined <- bind_rows(boing_boing_2015_clean, boing_boing_2016_clean, boing_boing_2017_
```

```
# Writing the final data file to csv
write_csv(boing_boing_data_combined, here("/3_clean_data/boing_boing_clean_data.csv"))
```

5. Next steps

1. rename candy that has "q6__" at the beginning

2. Look for duplicate candy names

3. Perform analysis to answer the following questions:

- What is the total number of candy ratings given across the three years. (number of candy ratings, not number of raters. Don't count missing values)
- What was the average age of people who are going out trick or treating and the average age of people 3. not going trick or treating?
- For each of joy, despair and meh, which candy bar revived the most of these ratings?
- How many people rated Starburst as despair?
- What was the most popular candy bar by this rating system for each gender in the dataset?
- What was the most popular candy bar in each year?
- What was the most popular candy bar by this rating for people in US, Canada, UK and all other countries?
- Any other interesting analyses or conclusions you come across.