Untitled

Theodore Ho

1/17/2022

Data: Trails in San Francisco, CA.

Today's data comes from the Metropolitan Transportation Commission (MTC) Open Data Catalog an Open Data program managed by the MTC and the Association of Bay Area Governments to provide local agencies and the public with their data needs.

In this lab, we will focus on data about the existing and planned segments of the San Francisco Bay trail. The data is located in the *SFO_trails.csv* file located in the *data* folder. Use the code below to read in the .csv file and save it in the RStudio environment as a data frame called `trails`.

```
trails <- read_csv("data/SFO-trails.csv")
```

```
## Rows: 739 Columns: 12
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): county, city, surface, agency, status, year_cmplt, legend
## dbl (5): objectid, class, seg_num, length, SHAPE_Length
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

A full list of the variables in the dataset is available here. For today's analysis, we will primarily focus on the following variables:

`status`	Whether the trail is proposed or existing
`class`	Category for the trail segment (4 types)
`length`	Length of the trail segment in miles

Exercises

Write your answers in complete sentences and show all code and output.

Before doing any analysis, we may want to get quick view of the data. This is a useful thing to do after importing data to see if the data imported correctly. One way to do this, is to look at the actual dataset. Type the code below in the **console** to view the entire dataset.

```
View(trails)
```

Exploratory Data Analysis

1. Now that we've had a quick view of the dataset, let's get more details about its structure. Sometimes viewing a summary of the data structure is more useful than viewing the raw data, especially if the dataset has a large number of observations and/or rows. Run the code below to use the `glimpse` function to see a summary of the `trails` dataset.

   How many observations are in the `trails` dataset? How many variables?

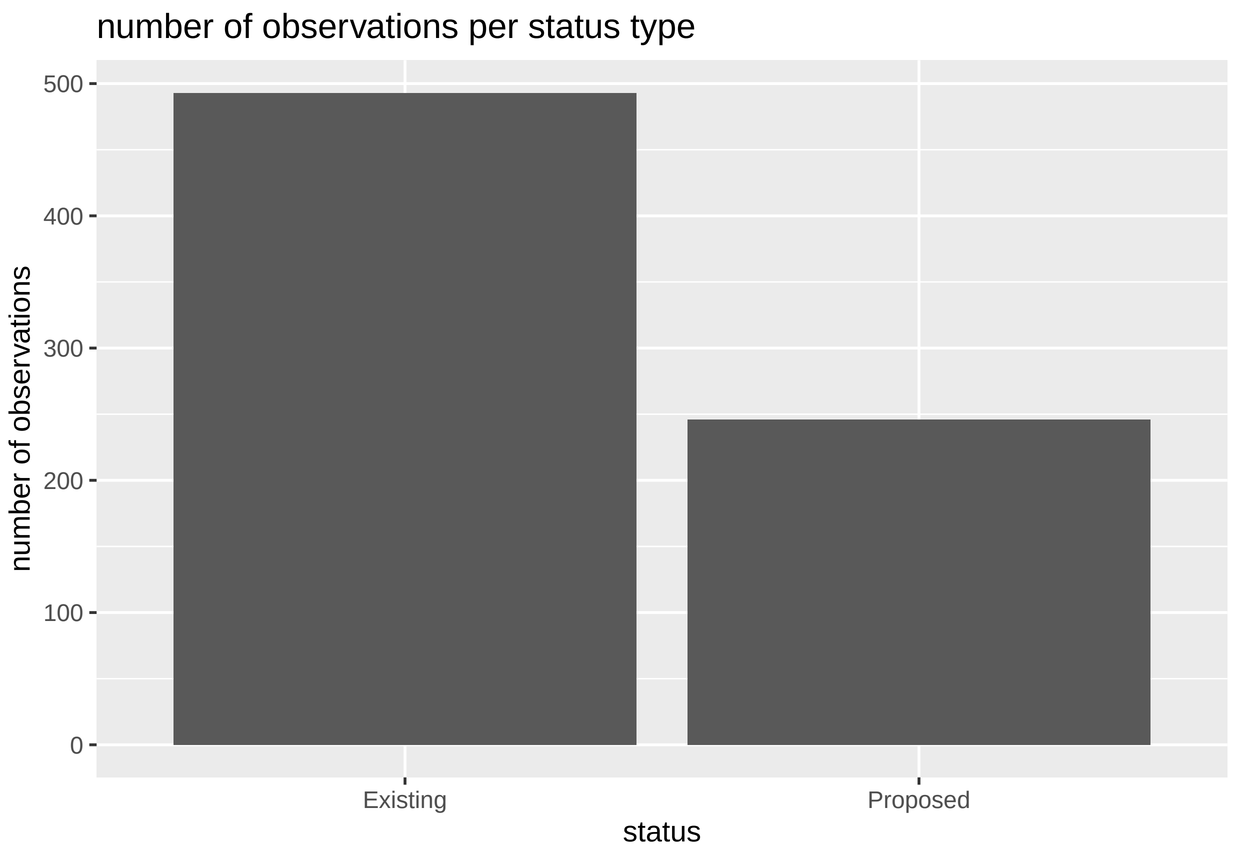
```
glimpse(trails)
```

```
## Rows: 739
## Columns: 12
## $ objectid     <dbl> 2952, 2953, 2954, 2955, 2956, 2957, 2958, 2959, 2960, 296~
## $ county       <chr> "Marin", "Marin", "Marin", "San Mateo", "San Mateo", "San~
## $ city         <chr> "Novato", "Novato", "San Rafael", "Brisbane", "S San Fran~
## $ surface      <chr> NA, NA, NA, NA, "paved", NA, "paved", NA, NA, NA, NA, NA,~
## $ class        <dbl> 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 2, ~
## $ agency       <chr> "Caltrans", "Sonoma-Marin Area Rail Transit", "San Rafael~
## $ status       <chr> "Proposed", "Proposed", "Proposed", "Proposed", "Existing~
## $ seg_num      <dbl> 9002, 9009, 9024, 2001, 2010, 1032, 2047, 2042, 2089, 206~
## $ length       <dbl> 3.20483759, 2.21318493, 1.47142826, 1.24527351, 0.5966338~
## $ year_cmplt   <chr> NA, NA, NA, NA, "2009", NA, NA, NA, NA, NA, NA, NA, "2014~
## $ legend       <chr> "Planned Bay Trail", "Planned Bay Trail", "Planned Bay Tr~
## $ SHAPE_Length <dbl> 0.052688281, 0.034781330, 0.022816134, 0.018364298, 0.009~
```

2. Before conducting statistical inference (or eventually fitting regression models), we need do some exploratory data analysis (EDA). Much of EDA consists of visualizing the data but it also includes calculating summary statistics for the variables in our dataset. Let's begin by examining the distribution of `status` with a data visualization and summary statistics.

   - What is a type of graph that's appropriate to visualize the distribution of `status`? Fill in the `ggplot` code below to plot the distribution of `status`. Include informative axis labels and title on the graph.
   - Then, calculate the proportion of observations in each category of `status` by completing the code below.

```
ggplot(data = trails, aes(x = status)) +
  geom_bar() +
  labs(x = "status",
       y = "number of observations",
       title = "number of observations per status type")
```



```
trails %>%
  count(status) %>%
  mutate(proportion = n / sum(n))
```

```
## # A tibble: 2 x 3
##   status       n proportion
##   <chr>    <int>      <dbl>
## 1 Existing   493      0.667
## 2 Proposed   246      0.333
```

3. Since we want to analyze characteristics for trails in the Bay Area, we will just use data from currently existing trails for the remainder of the analysis. Complete the code below to use the `filter` function to create a subset consisting only of trails that currently exist and have a value reported for `length`. Assign the subset the name `current_trails`. (*Hint: There should be 493 observations in current_trails.*)

```
current_trails <- trails %>%
  filter(status == "Existing", !is.na(length))
```

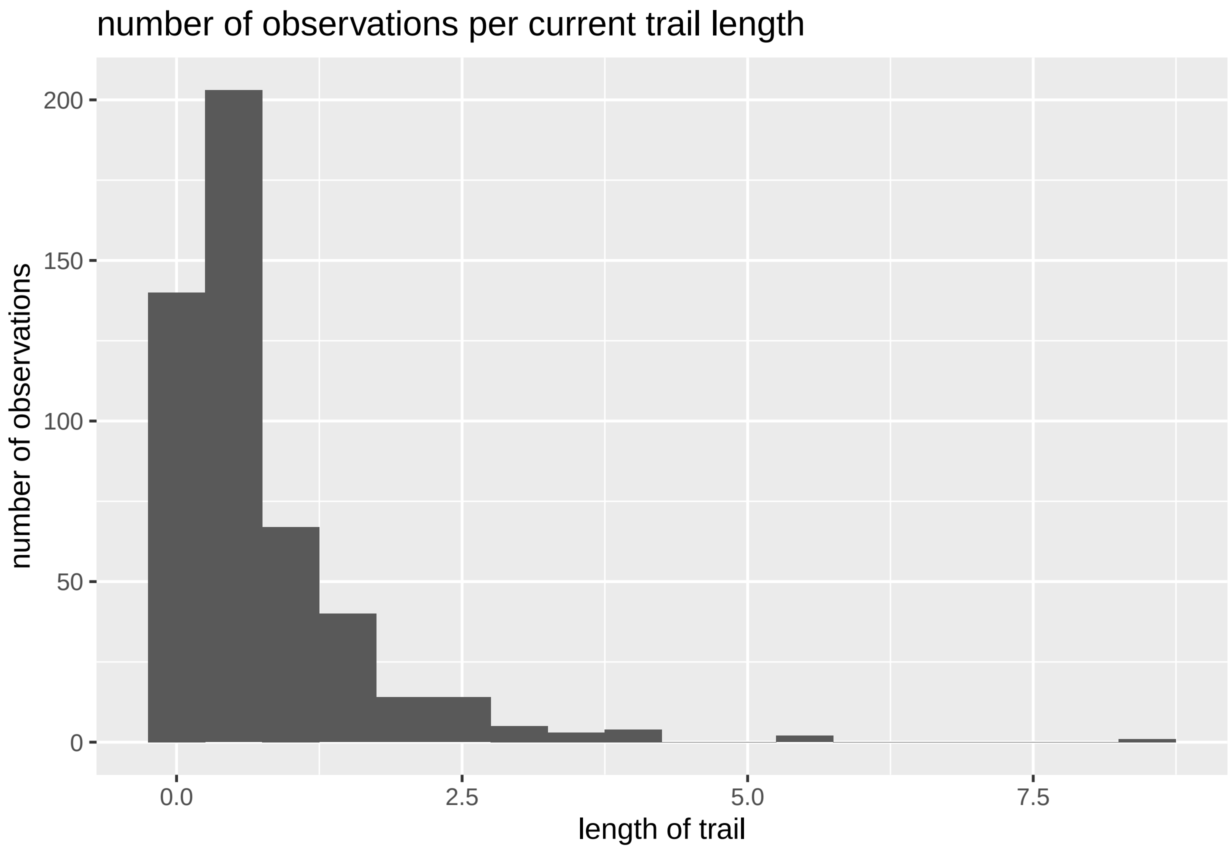
This is a good place to knit, commit, and push changes to your remote lab-01 repo on GitHub. Be sure to write an informative commit message (e.g. "Completed exercises 1 - 3"), and push every file to GitHub by clicking the checkbox next to each file in the Git pane. After you push the changes, the Git pane in RStudio should be empty."

Use `current_trails` for Exercises 4 - 7.

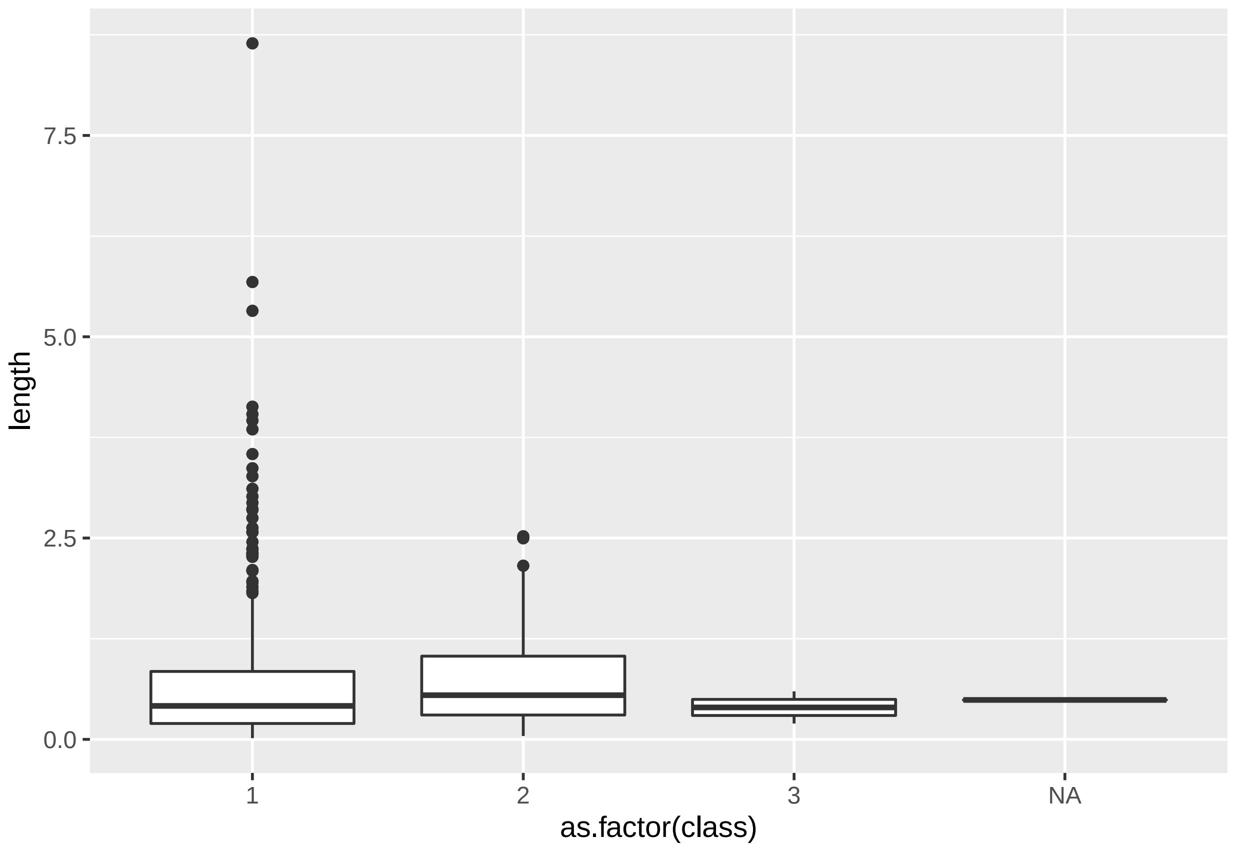
4. Let's examine the distribution of `length`. One important part of EDA is creating data visualizations to see the shape, center, spread, and outliers in a distribution. Data visualizations are also useful for examining the relationship between multiple variables. There are a lot of ways to make data visualizations in R; we will use the functions available in the `ggplot2` package.

   Make a graph to visualize the distribution of `length`. Include an informative title and axis labels.

```
ggplot(data = current_trails) +
  geom_histogram(mapping = aes(x = length), binwidth = 0.5) +
  labs(x = "length of trail",
       y = "number of observations",
       title = "number of observations per current trail length")
```



```
ggplot(data = current_trails, mapping = aes(x = as.factor(class), y = length)) +
  geom_boxplot()
```



See Section 7.3.1 “Visualizing Distributions” or the ggplot2 reference page for details and example code.

5. Next, fill in the code below to use the `summarise` function to calculate various summary statistics for the variable `length`. You can use the summarise reference page for more information about the function and example code.

```
current_trails %>%
  summarise(min = min(length),
            q1 = quantile(length, .25),
            median = median(length),
            q3 = quantile(length, .75),
            max = max(length),
            iqr = IQR(length),
            mean = mean(length),
            std_dev = sd(length)
            )
```

6. Describe the distribution of `length`. Your description should include comments about the shape, center, spread, and any potential outliers. Use the graph from Exercise 4 and relevant summary statistics from Exercise 5 in your description.

7. We want to limit the analysis to trails that are more likely intended for day hikes, rather than multi-day hikes and camping. Therefore, let’s remove the extreme outliers from the data for this analysis and only consider those trails that are 5 miles or shorter.

```
current_trails %>% filter(length <= 5)
```

```
## # A tibble: 490 x 12
##    objectid county  city  surface class agency  status seg_num length year_cmplt
##       <dbl> <chr>   <chr> <chr>   <dbl> <chr>   <chr>    <dbl>  <dbl> <chr>
##  1     2956 San Ma~ S Sa~ paved       1 SSF Pl~ Exist~    2010 0.597  2009
##  2     2958 San Ma~ Burl~ paved       1 Burlin~ Exist~    2047 0.0754 <NA>
##  3     2964 San Ma~ Menl~ paved       1 Menlo ~ Exist~    2091 0.353  2014
##  4     2965 San Ma~ East~ <NA>        1 Midpen~ Exist~    2092 0.571  <NA>
##  5     2966 Alameda San ~ paved       2 San Le~ Exist~    4044 0.422  <NA>
##  6     2969 Alameda Oakl~ paved       1 Oaklan~ Exist~    4078 0.155  2009
##  7     2970 Alameda Alam~ paved       2 Alamed~ Exist~    4063 0.208  2011
##  8     2975 Alameda Alam~ paved       1 Alamed~ Exist~    4120 0.209  2006
##  9     2976 Alameda Alam~ paved       2 Alamed~ Exist~    4108 0.820  <NA>
## 10     2978 Alameda Alba~ <NA>        1 East B~ Exist~    4164 0.936  2020
## # ... with 480 more rows, and 2 more variables: legend <chr>,
## #   SHAPE_Length <dbl>
```

```
Filter the dataset to remove the extreme outliers. **Be sure to save the updated dataset, so you can us
```

This is a good place to knit, commit, and push changes to your remote lab-01 repo on GitHub. Be sure to write informative commit message (e.g. "Completed exercises 4 - 7"), and push every file to GitHub by clicking the checkbox next to each file in the Git pane. After you push the changes, the Git pane in RStudio should be empty."

8. Consider the distribution of `class`.

   - What are the values of `class` in the dataset? Show the code and output to support your answer.
   - What do you think is the most likely reason for the missing observations of `class`? In other words, what does a missing value of `class` indicate?

9. Complete the code below to impute (i.e. fill in) the missing values of `class` with the appropriate value. After that, eliminate all the observations from class = 3, since we are not going to use the. Then, display the distribution of `class` to check that the missing values were correctly imputed.

```
class_trails <- current_trails %>% filter(class !=3) %>%
  mutate(class = if_else(is.na(class),0,class))

ggplot(data = class_trails, aes(x = class)) +
  geom_bar() +
  labs(x = "class",
       y = "number of observations",
       title = "number of observations per class type")
```

10. Now that we've completed the univariate EDA (i.e. examining one variable at a time), let's examine the relationship between the length of the trail and its class variable. Make a graph to visualize the relationship between `length` and `class` and calculate the appropriate summary statistics. Include informative axis labels and title on your graph.

```
#ggplot(data = class_trails, mapping = aes(x = as.factor(class), y = length)) +
 # geom_boxplot()
```

11. Describe the relationship between `length` and `class`. In other words, describe how the distribution of `length` compares between trails that have different classes (1 = shared use bicycle and pedestrian path, 2 = bike lane, and 3 = bike route). Include information from the graph and summary statistics from the previous exercise in your response.

This is a good place to knit, commit, and push changes to your remote lab-01 repo on GitHub. Be sure to write informative commit message (e.g. "Completed exercises 8 - 11"), and push every file to GitHub by clicking the checkbox next to each file in the Git pane. After you push the changes, the Git pane in RStudio should be empty."

Statistical Inference

We'd like to use the data from the trails in SFO to make more general conclusions about trails in urban areas in California, United States. We will reasonably consider the trails in SFO representative of the trails in other urban areas in the West Coast of United States.

Over the next few questions, will use statistical inference to assess whether there is a difference in the mean length of trails that share use bicycle and pedestrian path (class = 1) and those that only have a bike line (class = 2).

12. The following conditions must be met when we conduct statistical inference on the difference in means between two groups. For each condition, specify whether it is met and a brief explanation of your reasoning.
    - **Independence**
    - **Sample Size**
    - **Independent Groups**
13. While we have observed a small difference in the mean length in trails with bike lanes (class = 2) and trials that share bikes with pedestrians (class = 1), let's assess if there is enough evidence to consider the difference "statistically significant" or if it appears to be due to random chance.

The null and alternative hypotheses are written in statistical notation below. State the hypotheses in words in the context of this analysis.

$$H_0 : \mu_1 - \mu_2 = 0 H_a : \mu_1 - \mu_2 \neq 0$$

14. Fill in the code below to use the `t.test` function to calculate the test statistic and p-value. Replace `response` with the variable we're interested in drawing conclusions about and `group_var` with the variable used to define the two groups.

```r
?t.test # to see the help page from the function
t.test(response ~ group_var, data = ________,
       alternative = "___________",
       conf.level = 0.99) #less, greater, or two.sided
```

15. Use the output from the previous exercise to answer the following:
    - Write the definition of the test statistic in the context of this analysis.
    - Write the definition of the p-value in the context of this analysis.
    - State your conclusion in the context of this analysis. Use a significance level of $\alpha = 0.01$.
16. Notice the confidence interval for the difference in mean trail length printed in the output from Exercise 14. Interpret this confidence interval in the context of this analysis.

You're done and ready to submit your work! Knit, commit, and push all remaining changes. You can use the commit message "Done with Lab 1!", and make sure you have pushed all the files to GitHub (your Git pane in RStudio should be empty) and thatall documents are updated in your repo on GitHub. Then submit the pdf for your assignment on Gradescope. Include your repo name, so I can check your commits.