p8105_hw2_brm2150

Brooklynn McNeil

2024-09-24

This is an R markdown document for the homework assignment #2 for the p8105 Data Science 1 Class.

Load Packages

Problem 1

Read and clean the data; retain line, station, name, station latitude / longitude, routes served, entry, vending, entrance type, and ADA compliance. Convert the entry variable from character (YES vs NO) to a logical variable (the ifelse or case_match function may be useful)

```
nyc_transit_df =
  read_csv("data/NYC_Transit_subway_Entrance_And_Exit_Data.csv", col_types = cols(Route8 = "c", Route9 =
  janitor::clean_names() |>
  select(
    line, station_name, station_latitude, station_longitude,
    starts_with("route"), entry, exit_only, vending, entrance_type,
    ada) |>
  mutate(entry = ifelse(entry == "YES", TRUE, FALSE)) |>
  pivot_longer(
    route1:route11,
    names_to = "route_num",
    values_to = "route")
```

This code chunk finds the distinct A train stations and then out of those which ones are ADA compliant.

```
nyc_transit_df |>
  filter(route == "A") |>
  select(station_name, line) |>
  distinct()
```

```
## # A tibble: 60 x 2
##    station_name                  line
##    <chr>                         <chr>
##  1 Times Square                  42nd St Shuttle
##  2 125th St                      8 Avenue
##  3 145th St                      8 Avenue
##  4 14th St                       8 Avenue
##  5 168th St - Washington Heights 8 Avenue
##  6 175th St                      8 Avenue
##  7 181st St                      8 Avenue
##  8 190th St                      8 Avenue
```

```
##  9 34th St                       8 Avenue
## 10 42nd St                       8 Avenue
## # i 50 more rows
```

```
nyc_transit_df |>
  filter(route == "A", ada == TRUE) |>
  select(station_name, line) |>
  distinct()
```

```
## # A tibble: 17 x 2
##    station_name                  line
##    <chr>                         <chr>
##  1 14th St                       8 Avenue
##  2 168th St - Washington Heights 8 Avenue
##  3 175th St                      8 Avenue
##  4 34th St                       8 Avenue
##  5 42nd St                       8 Avenue
##  6 59th St                       8 Avenue
##  7 Inwood - 207th St             8 Avenue
##  8 West 4th St                   8 Avenue
##  9 World Trade Center            8 Avenue
## 10 Times Square-42nd St          Broadway
## 11 59th St-Columbus Circle       Broadway-7th Ave
## 12 Times Square                  Broadway-7th Ave
## 13 8th Av                        Canarsie
## 14 Franklin Av                   Franklin
## 15 Euclid Av                     Fulton
## 16 Franklin Av                   Fulton
## 17 Howard Beach                  Rockaway
```

Problem 2

Load Trash Wheel Data from excel and combine Mr. Trash Wheel, Professor Trash Wheel, and Gwynda Trash Wheel.

```
mr.trashwheel_df =
  read_excel("data/202409 Trash Wheel Collection Data.xlsx", sheet = "Mr. Trash Wheel", range = "A2:N653
  janitor::clean_names() |>
  mutate(sports_balls = round(sports_balls)) |>
  mutate(sports_balls = as.integer(sports_balls)) |>
  mutate(year = as.double(year)) |>
  mutate(trash_wheel = "mr")

mr.trashwheel_df
```

```
## # A tibble: 651 x 15
##    dumpster month  year date                weight_tons volume_cubic_yards
##       <dbl> <chr> <dbl> <dttm>                    <dbl>              <dbl>
##  1        1 May    2014 2014-05-16 00:00:00        4.31                 18
##  2        2 May    2014 2014-05-16 00:00:00        2.74                 13
##  3        3 May    2014 2014-05-16 00:00:00        3.45                 15
##  4        4 May    2014 2014-05-17 00:00:00        3.1                  15
```

```
##  5        5 May    2014 2014-05-17 00:00:00        4.06                18
##  6        6 May    2014 2014-05-20 00:00:00        2.71                13
##  7        7 May    2014 2014-05-21 00:00:00        1.91                 8
##  8        8 May    2014 2014-05-28 00:00:00        3.7                 16
##  9        9 June   2014 2014-06-05 00:00:00        2.52                14
## 10       10 June   2014 2014-06-11 00:00:00        3.76                18
## # i 641 more rows
## # i 9 more variables: plastic_bottles <dbl>, polystyrene <dbl>,
## #   cigarette_butts <dbl>, glass_bottles <dbl>, plastic_bags <dbl>,
## #   wrappers <dbl>, sports_balls <int>, homes_powered <dbl>, trash_wheel <chr>
```

```
prof.trashwheel_df =
  read_excel("data/202409 Trash Wheel Collection Data.xlsx", sheet = "Professor Trash Wheel", range = "A
  janitor::clean_names() |>
  mutate(trash_wheel = "prof")

prof.trashwheel_df
```

```
## # A tibble: 118 x 14
##    dumpster month     year date                weight_tons volume_cubic_yards
##       <dbl> <chr>    <dbl> <dttm>                    <dbl>              <dbl>
##  1        1 January   2017 2017-01-02 00:00:00        1.79                 15
##  2        2 January   2017 2017-01-30 00:00:00        1.58                 15
##  3        3 February  2017 2017-02-26 00:00:00        2.32                 18
##  4        4 February  2017 2017-02-26 00:00:00        3.72                 15
##  5        5 February  2017 2017-02-28 00:00:00        1.45                 15
##  6        6 March     2017 2017-03-30 00:00:00        1.71                 15
##  7        7 April     2017 2017-04-01 00:00:00        1.82                 15
##  8        8 April     2017 2017-04-20 00:00:00        2.37                 15
##  9        9 May       2017 2017-05-10 00:00:00        2.64                 15
## 10       10 May       2017 2017-05-26 00:00:00        2.78                 15
## # i 108 more rows
## # i 8 more variables: plastic_bottles <dbl>, polystyrene <dbl>,
## #   cigarette_butts <dbl>, glass_bottles <dbl>, plastic_bags <dbl>,
## #   wrappers <dbl>, homes_powered <dbl>, trash_wheel <chr>
```

```
gwyn.trashwheel_df =
  read_excel("data/202409 Trash Wheel Collection Data.xlsx", sheet = "Gwynnda Trash Wheel", range = "A2
  janitor::clean_names() |>
  mutate(trash_wheel = "gwyn")

gwyn.trashwheel_df
```

```
## # A tibble: 263 x 13
##    dumpster month   year date                weight_tons volume_cubic_yards
##       <dbl> <chr>  <dbl> <dttm>                    <dbl>              <dbl>
##  1        1 July    2021 2021-07-03 00:00:00        0.93                 15
##  2        2 July    2021 2021-07-07 00:00:00        2.26                 15
##  3        3 July    2021 2021-07-07 00:00:00        1.62                 15
##  4        4 July    2021 2021-07-16 00:00:00        1.76                 15
##  5        5 July    2021 2021-07-30 00:00:00        1.53                 15
##  6        6 August  2021 2021-08-11 00:00:00        2.06                 15
##  7        7 August  2021 2021-08-14 00:00:00        1.9                  15
```

```
##  8        8 August  2021 2021-08-16 00:00:00        2.16                15
##  9        9 August  2021 2021-08-16 00:00:00        2.6                 15
## 10       10 August  2021 2021-08-17 00:00:00        3.21                15
## # i 253 more rows
## # i 7 more variables: plastic_bottles <dbl>, polystyrene <dbl>,
## #   cigarette_butts <dbl>, plastic_bags <dbl>, wrappers <dbl>,
## #   homes_powered <dbl>, trash_wheel <chr>
```

```
trashwheel_df =
  bind_rows(mr.trashwheel_df,prof.trashwheel_df, gwyn.trashwheel_df)

trashwheel_df
```

```
## # A tibble: 1,032 x 15
##    dumpster month  year date                weight_tons volume_cubic_yards
##       <dbl> <chr> <dbl> <dttm>                    <dbl>              <dbl>
##  1        1 May    2014 2014-05-16 00:00:00        4.31                18
##  2        2 May    2014 2014-05-16 00:00:00        2.74                13
##  3        3 May    2014 2014-05-16 00:00:00        3.45                15
##  4        4 May    2014 2014-05-17 00:00:00        3.1                 15
##  5        5 May    2014 2014-05-17 00:00:00        4.06                18
##  6        6 May    2014 2014-05-20 00:00:00        2.71                13
##  7        7 May    2014 2014-05-21 00:00:00        1.91                 8
##  8        8 May    2014 2014-05-28 00:00:00        3.7                 16
##  9        9 June   2014 2014-06-05 00:00:00        2.52                14
## 10       10 June   2014 2014-06-11 00:00:00        3.76                18
## # i 1,022 more rows
## # i 9 more variables: plastic_bottles <dbl>, polystyrene <dbl>,
## #   cigarette_butts <dbl>, glass_bottles <dbl>, plastic_bags <dbl>,
## #   wrappers <dbl>, sports_balls <int>, homes_powered <dbl>, trash_wheel <chr>
```

The trash wheel in these 3 locations collected on average 2201 plastic bottles per dumpster. The trash wheel data were collected between 2014, 2024. Professor Trash Wheel collected a total weight of 246.74 tons of trash. The total number of cigarette butts that was collected by Gwynda in June of 2022 was 1.812×10^4.

Problem 3

Importing and wrangling the data sets from the Great British Bakeoff show. To create the combined data set I took the following steps:

1. `read_csv` to import the data.
    1. some data sets needed `nas` to be specified.
    2. some data sets needed rows skipped at the beginning of the file.
2. `clean_names` to transform all of the variable names to snake case.
3. In `bakers_df` the `baker_name` variable needed to have the last name removed so it could be used to join with the , so I used `separate` and `select` to remove the last name.
4. I selected only the unique columns in `bakes_df` and `results_df` so that it didn't cause issues when joining the data sets together.
5. I then created a new data frame `gbb_df` that joined together the 3 data sets using the `full_join` function.I also made sure to join by `baker`, `series` and `episode` to make sure the data was aligned and no duplicates were made.

6. I also found that a few data was missing from the final data frame because it had missing values for all of the variables. This was checked using the `anti-join` function.
7. I used `tidyverse` and the `pipe` function to achieve this data wrangling.

It seems that "Jo" is being left out and is maybe named "Joanne" in other tables. I am choosing to not include them in the final table because it is unclear if they are the same person or not.

```
bakers_df =
  read_csv("data/bakers.csv") |>
  janitor::clean_names() |>
  separate(baker_name, into = c("baker", "last_name"), sep = " ")
```

```
## Rows: 120 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Baker Name, Baker Occupation, Hometown
## dbl (2): Series, Baker Age
##
## 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.
```

```
bakers_df
```

```
## # A tibble: 120 x 6
##    baker     last_name   series baker_age baker_occupation             hometown
##    <chr>     <chr>        <dbl>     <dbl> <chr>                        <chr>
##  1 Ali       Imdad            4        25 Charity worker               Saltley,~
##  2 Alice     Fevronia        10        28 Geography teacher            Essex
##  3 Alvin     Magallanes       6        37 Nurse                        Bracknel~
##  4 Amelia    LeBruin         10        24 Fashion designer             Halifax
##  5 Andrew    Smyth            7        25 Aerospace engineer           Derby / ~
##  6 Annetha   Mills            1        30 Midwife                      Essex
##  7 Antony    Amourdoux        9        30 Banker                       London
##  8 Beca      Lyne-Pirkis      4        31 Military Wives' Choir Singer Aldersho~
##  9 Ben       Frazer           2        31 Graphic Designer             Northamp~
## 10 Benjamina Ebuehi           7        23 Teaching assistant           South Lo~
## # i 110 more rows
```

```
bakes_df =
  read_csv("data/bakes.csv", na = "N/A") |>
  janitor::clean_names()
```

```
## Rows: 548 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Baker, Signature Bake, Show Stopper
## dbl (2): Series, Episode
##
## 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.
```

```
bakes_df
```

```
## # A tibble: 548 x 5
##    series episode baker     signature_bake                            show_stopper
##     <dbl>   <dbl> <chr>     <chr>                                     <chr>
##  1      1       1 Annetha   "Light Jamaican Black Cakewith Strawbe~ Red, White ~
##  2      1       1 David     "Chocolate Orange Cake"                   Black Fores~
##  3      1       1 Edd       "Caramel Cinnamon and Banana Cake"        <NA>
##  4      1       1 Jasminder "Fresh Mango and Passion Fruit Humming~ <NA>
##  5      1       1 Jonathan  "Carrot Cake with Lime and Cream Chees~ Three Tiere~
##  6      1       1 Lea       "Cranberry and Pistachio Cakewith Oran~ Raspberries~
##  7      1       1 Louise    "Carrot and Orange Cake"                  Never Fail ~
##  8      1       1 Mark      "Sticky Marmalade Tea Loaf"               Heart-shape~
##  9      1       1 Miranda   "Triple Layered Brownie Meringue Cake\~ Three Tiere~
## 10      1       1 Ruth      "Three Tiered Lemon Drizzle Cakewith F~ Classic Cho~
## # i 538 more rows
```

```
results_df =
  read_csv("data/results.csv", skip = 2, na = "NA") |>
  janitor::clean_names()
```

```
## Rows: 1136 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): baker, result
## dbl (3): series, episode, technical
##
## 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.
```

```
results_df
```

```
## # A tibble: 1,136 x 5
##    series episode baker     technical result
##     <dbl>   <dbl> <chr>         <dbl> <chr>
##  1      1       1 Annetha           2 IN
##  2      1       1 David             3 IN
##  3      1       1 Edd               1 IN
##  4      1       1 Jasminder        NA IN
##  5      1       1 Jonathan          9 IN
##  6      1       1 Louise           NA IN
##  7      1       1 Miranda           8 IN
##  8      1       1 Ruth             NA IN
##  9      1       1 Lea              10 OUT
## 10      1       1 Mark             NA OUT
## # i 1,126 more rows
```

```
gbb_df =
  results_df |>
  full_join(bakes_df, by = c("baker", "series","episode")) |>
  full_join(bakers_df, by = c("baker", "series"))

gbb_df
```

```
## # A tibble: 1,145 x 11
##    series episode baker   technical result signature_bake show_stopper last_name
##     <dbl>   <dbl> <chr>       <dbl> <chr>  <chr>          <chr>        <chr>
##  1      1       1 Annetha         2 IN     "Light Jamaic~ Red, White ~ Mills
##  2      1       1 David           3 IN     "Chocolate Or~ Black Fores~ Chambers
##  3      1       1 Edd             1 IN     "Caramel Cinn~ <NA>         Kimber
##  4      1       1 Jasmin~        NA IN     "Fresh Mango ~ <NA>         Randhawa
##  5      1       1 Jonath~         9 IN     "Carrot Cake ~ Three Tiere~ Shepherd
##  6      1       1 Louise         NA IN     "Carrot and O~ Never Fail ~ Brimelow
##  7      1       1 Miranda         8 IN     "Triple Layer~ Three Tiere~ Browne
##  8      1       1 Ruth           NA IN     "Three Tiered~ Classic Cho~ Clemens
##  9      1       1 Lea            10 OUT    "Cranberry an~ Raspberries~ Harris
## 10      1       1 Mark           NA OUT    "Sticky Marma~ Heart-shape~ Whithers
## # i 1,135 more rows
## # i 3 more variables: baker_age <dbl>, baker_occupation <chr>, hometown <chr>
```

```
anti_join(bakes_df, bakers_df, by = "baker")
```

```
## # A tibble: 8 x 5
##   series episode baker    signature_bake                              show_stopper
##    <dbl>   <dbl> <chr>    <chr>                                       <chr>
## 1      2       1 "\"Jo\"" Chocolate Orange CupcakesOrange and Card~ Chocolate a~
## 2      2       2 "\"Jo\"" Caramelised Onion, Gruyere and Thyme Qui~ Raspberry a~
## 3      2       3 "\"Jo\"" Stromboli flavored with Mozzarella, Ham,~ Unknown
## 4      2       4 "\"Jo\"" Lavender Biscuits                           Blueberry M~
## 5      2       5 "\"Jo\"" Salmon and Asparagus Pie                    Apple and R~
## 6      2       6 "\"Jo\"" Rum and Raisin Baked Cheesecake             Limoncello ~
## 7      2       7 "\"Jo\"" Raspberry & Strawberry Mousse Cake          Pain Aux Ra~
## 8      2       8 "\"Jo\"" Raspberry and Blueberry Mille Feuille       Mini Victor~
```

```
anti_join(bakes_df, results_df, by = "baker")
```

```
## # A tibble: 8 x 5
##   series episode baker    signature_bake                              show_stopper
##    <dbl>   <dbl> <chr>    <chr>                                       <chr>
## 1      2       1 "\"Jo\"" Chocolate Orange CupcakesOrange and Card~ Chocolate a~
## 2      2       2 "\"Jo\"" Caramelised Onion, Gruyere and Thyme Qui~ Raspberry a~
## 3      2       3 "\"Jo\"" Stromboli flavored with Mozzarella, Ham,~ Unknown
## 4      2       4 "\"Jo\"" Lavender Biscuits                           Blueberry M~
## 5      2       5 "\"Jo\"" Salmon and Asparagus Pie                    Apple and R~
## 6      2       6 "\"Jo\"" Rum and Raisin Baked Cheesecake             Limoncello ~
## 7      2       7 "\"Jo\"" Raspberry & Strawberry Mousse Cake          Pain Aux Ra~
## 8      2       8 "\"Jo\"" Raspberry and Blueberry Mille Feuille       Mini Victor~
```

```
anti_join(results_df, bakers_df, by = "baker")
```

```
## # A tibble: 8 x 5
##   series episode baker  technical result
##    <dbl>   <dbl> <chr>      <dbl> <chr>
## 1      2       1 Joanne        11 IN
## 2      2       2 Joanne        10 IN
## 3      2       3 Joanne         1 IN
```

```
## 4      2       4 Joanne         8 IN
## 5      2       5 Joanne         6 IN
## 6      2       6 Joanne         1 STAR BAKER
## 7      2       7 Joanne         3 IN
## 8      2       8 Joanne         1 WINNER
```

Creating a table showing the winner of each episode in Season 5 through 10

```
gbb_df |>
  filter(result == "STAR BAKER"|result == "WINNER", series >=5) |>
  select(c(baker, series, episode, result)) |>
  arrange(series, episode) |>
  gt() |>
  tab_header(
    title = "Great British Bakeoff Episode Winners",
    subtitle = "Episode Winners from Season 5 through 10"
  )
```

Great British Bakeoff Episode Winners

Episode Winners from Season 5 through 10

baker	series	episode	result
Nancy	5	1	STAR BAKER
Richard	5	2	STAR BAKER
Luis	5	3	STAR BAKER
Richard	5	4	STAR BAKER
Kate	5	5	STAR BAKER
Chetna	5	6	STAR BAKER
Richard	5	7	STAR BAKER
Richard	5	8	STAR BAKER
Richard	5	9	STAR BAKER
Nancy	5	10	WINNER
Marie	6	1	STAR BAKER
Ian	6	2	STAR BAKER
Ian	6	3	STAR BAKER
Ian	6	4	STAR BAKER
Nadiya	6	5	STAR BAKER
Mat	6	6	STAR BAKER
Tamal	6	7	STAR BAKER
Nadiya	6	8	STAR BAKER
Nadiya	6	9	STAR BAKER
Nadiya	6	10	WINNER
Jane	7	1	STAR BAKER
Candice	7	2	STAR BAKER
Tom	7	3	STAR BAKER
Benjamina	7	4	STAR BAKER
Candice	7	5	STAR BAKER

Tom	7	6	STAR BAKER
Andrew	7	7	STAR BAKER
Candice	7	8	STAR BAKER
Andrew	7	9	STAR BAKER
Candice	7	10	WINNER
Steven	8	1	STAR BAKER
Steven	8	2	STAR BAKER
Julia	8	3	STAR BAKER
Kate	8	4	STAR BAKER
Sophie	8	5	STAR BAKER
Liam	8	6	STAR BAKER
Steven	8	7	STAR BAKER
Stacey	8	8	STAR BAKER
Sophie	8	9	STAR BAKER
Sophie	8	10	WINNER
Manon	9	1	STAR BAKER
Rahul	9	2	STAR BAKER
Rahul	9	3	STAR BAKER
Dan	9	4	STAR BAKER
Kim-Joy	9	5	STAR BAKER
Briony	9	6	STAR BAKER
Kim-Joy	9	7	STAR BAKER
Ruby	9	8	STAR BAKER
Ruby	9	9	STAR BAKER
Rahul	9	10	WINNER
Michelle	10	1	STAR BAKER
Alice	10	2	STAR BAKER
Michael	10	3	STAR BAKER
Steph	10	4	STAR BAKER
Steph	10	5	STAR BAKER
Steph	10	6	STAR BAKER
Henry	10	7	STAR BAKER
Steph	10	8	STAR BAKER
Alice	10	9	STAR BAKER
David	10	10	WINNER

In season 5 it looks like there was an upset because Richard won the last 3 episodes before the finale but Nancy took the cake *literally*. Nadiya powered through the last 3 episodes of season 6 to become the winner. Rahul looks like he started strong and then lost some steam, but came back to win the season in the end. Season 10 was probably the most surprising because the winner, David, was never crowned star baker in an episode before the finale.

Import, tidy, and wrangle the *viewers* data.

```
viewers_df =
  read_csv("data/viewers.csv", na = "NA") |>
  janitor::clean_names() |>
```

```
  pivot_longer(
    cols = series_1:series_10,
    names_to = "series",
    values_to = "viewership",
    names_prefix = "series_") |>
  drop_na()
```

```
## Rows: 10 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (11): Episode, Series 1, Series 2, Series 3, Series 4, Series 5, Series ...
##
## 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.
```

```
head(viewers_df, n=10)
```

```
## # A tibble: 10 x 3
##    episode series viewership
##      <dbl> <chr>       <dbl>
##  1       1 1            2.24
##  2       1 2            3.1
##  3       1 3            3.85
##  4       1 4            6.6
##  5       1 5            8.51
##  6       1 6           11.6
##  7       1 7           13.6
##  8       1 8            9.46
##  9       1 9            9.55
## 10       1 10           9.62
```

The average viewership for season 1 was 2.77 and for season 5 was 10.0393