p8105_hw2_pz2334

Puyuan Zhang

2024-09-29

Problem 2

```
mr_wheel_df =
  read_excel("gbb_datasets/202309 Trash Wheel Collection Data.xlsx", sheet = "Mr. Trash Wheel", range =
  janitor::clean_names() |>
  drop_na(dumpster) |>
  mutate(
    sports_balls = as.integer(round(sports_balls)),
         trash_wheel = "Mr. Trash Wheel",
         year = as.numeric(year))
```

```
professor_wheel_df =
  read_excel("gbb_datasets/202309 Trash Wheel Collection Data.xlsx", sheet = "Professor Trash Wheel", ra
  janitor::clean_names() |>
  drop_na(dumpster) |>
  mutate(trash_wheel = "Professor Trash Wheel",
         sports_balls = NA)
p_names <- names(professor_wheel_df)
```

```
gwynnda_wheel_df =
  read_excel("gbb_datasets/202309 Trash Wheel Collection Data.xlsx", sheet = "Gwynnda Trash Wheel", rang
  janitor::clean_names() |>
  drop_na(dumpster) |>
  mutate(trash_wheel = "Gwynnda Trash Wheel",
         sports_balls = NA,
         glass_bottles = NA)
```

```
combined_trash_wheel <- bind_rows(mr_wheel_df, professor_wheel_df)
combined_trash_wheel <- bind_rows(combined_trash_wheel, gwynnda_wheel_df)
```

```
n_obs <- nrow(combined_trash_wheel)
n_vars <- ncol(combined_trash_wheel)
tot_weight_p <- combined_trash_wheel |>
  filter(trash_wheel == "Professor Trash Wheel") |>
  summarise(total_weight = sum(weight_tons, na.rm = TRUE))
cig_june2022_g <- combined_trash_wheel |>
  filter(trash_wheel == "Gwynnda Trash Wheel", year == 2022, month == "June") %>%
  summarise(total_butts = sum(cigarette_butts, na.rm = TRUE))
```

The dataset, after cleaning and combining data from Mr. Trash Wheel, Professor Trash Wheel, and Gwynnda, includes 845 observations and 15 variables. The variables in this dataset are dumpster, month, year, date, weight_tons, volume_cubic_yards, plastic_bottles, polystyrene, cigarette_butts, glass_bottles, plastic_bags, wrappers, homes_powered, trash_wheel, sports_balls, which represent important metrics such as the weight of trash collected (in tons), the number of cigarette butts, and the number of sports balls. Among all these variables, we create trash_wheel to identify the origin of this row of data.

During combining and cleaning process, we noticed that the Professor Trash Wheel dataset did not contain any values for the sports balls variable. Additionally, the variable "year" in this dataset was originally stored as a character type, which differed from the other datasets. So, we changed the data type for "year" to numeric. Also, we add the sports balls variable, setting its values to NA to represent its missing.

Similarly, in the Gwynnda dataset, there was no value for both sports balls and glass bottles. So, we added these variables with NA values to the Gwynnda dataset to maintain uniformity across all datasets.

For the available data in the combined dataframe, we can easily extract specific measurements of interest. For instance, the total weight of trash collected by Professor Trash Wheel amounts to 216.26. Similarly, the total number of cigarette butts collected by Gwynnda in June 2022 is 18120.

Problem 3

```r
bakers =
  read_csv("gbb_datasets/bakers.csv") |>
  janitor::clean_names() |>
  mutate(baker = word(baker_name, 1),
  baker = if_else(baker == "Jo", "Joanne", baker))
bakes =
  read_csv("gbb_datasets/bakes.csv") |>
  janitor::clean_names() |>
  mutate(baker = if_else(baker == '"Jo"', "Joanne", baker))
results =
  read_csv("gbb_datasets/results.csv", skip = 2) |>
  janitor::clean_names() |>
  drop_na(result)
```

```r
combined_df = left_join(results, bakes, by = c("baker", "episode", "series"))
combined_df = left_join(combined_df, bakers, by = c("baker", "series"))
```

```r
unmatched_bakers <- anti_join(bakers, combined_df, by = c("baker", "series"))
unmatched_bakes <- anti_join(bakes, combined_df, by = c("baker", "series", "episode"))
unmatched_results <- anti_join(results, combined_df, by = c("baker", "series", "episode"))
```

```r
combined_df = combined_df |>
  select(-baker)
c_names <- names(combined_df)
```

We quickly review the datasets during the cleaning process at beginning, we can find out that the 'result' variable in results.csv is empty for rows where the baker are OUT in the previous episode. So, we drop rows where 'result' is NA.

After initial review, all three datasets included variables related to series and baker names. However, bakers.csv lists full names. We create a helper variable called 'baker' to facilitate merging the datasets. After the merging process, anti_join used in checking process indicates the inconsistency with a baker named "Jo," who was absent in the other two datasets. Go back to the original datasets, we found that “Jo” is referred to “Joanne” in results.csv. So, we change this baker’s name from “Jo” to “Joanne” in the other datasets to ensure a successful merge.

After merging, we noticed redundancy with two columns referring to the bakers’ names. So I removed the helper variable (the ‘baker’ column which contained first names) to simplify the dataset. The final dataset contains variable series, episode, technical, result, signature_bake, show_stopper, baker_name, baker_age, baker_occupation, hometown.

```
winners <- combined_df |>
  filter(series %in% 5:10, result %in% c("WINNER", "STAR BAKER")) |>
  select(series, episode, result, baker_name)
```

After creating and viewing this datasets, we can easily check the winner of each season and the star baker in each episode. It is surprising that the winner in season 10 has never been a star baker in any of the episode.

```
reviews =
  read_csv("gbb_datasets/viewers.csv") |>
  janitor::clean_names()
head(reviews, 10)
```

```
## # A tibble: 10 x 11
##    episode series_1 series_2 series_3 series_4 series_5 series_6 series_7
##      <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
##  1       1     2.24     3.1      3.85     6.6      8.51     11.6     13.6
##  2       2     3        3.53     4.6      6.65     8.79     11.6     13.4
##  3       3     3        3.82     4.53     7.17     9.28     12.0     13.0
##  4       4     2.6      3.6      4.71     6.82    10.2      12.4     13.3
##  5       5     3.03     3.83     4.61     6.95     9.95     12.4     13.1
##  6       6     2.75     4.25     4.82     7.32    10.1      12       13.1
##  7       7    NA        4.42     5.1      7.76    10.3      12.4     13.4
##  8       8    NA        5.06     5.35     7.41     9.02     11.1     13.3
##  9       9    NA       NA        5.7      7.41    10.7      12.6     13.4
## 10      10    NA       NA        6.74     9.45    13.5      15.0     15.9
## # i 3 more variables: series_8 <dbl>, series_9 <dbl>, series_10 <dbl>
```

```
mean_1 <- mean(reviews$series_1, na.rm = T)
mean_5 <- mean(reviews$series_5, na.rm = T)
```

The average viewership in Season 1 is 2.77, and average in Season 5 is 10.0393.