Extra credit

INFO 2950 - Spring 2023

Lili Mkrtchyan

5/8/23

Setup

Load packages and data:

```
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.4.2     v purrr   1.0.0
v tibble  3.2.1     v dplyr   1.1.2
v tidyr   1.2.1     v stringr 1.5.0
v readr   2.1.3     v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(stringr)
library(survival)
library(viridis)
```

```
Loading required package: viridisLite
```

```
library(dplyr)

survivalists <- readr::read_csv(
  'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-01-
```

```
Rows: 94 Columns: 16
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (10): name, gender, city, state, country, reason_tapped_out, reason_cate...
dbl  (5): season, age, result, days_lasted, day_linked_up
lgl  (1): medically_evacuated

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.
```

```
loadouts <- readr::read_csv(
  'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-01-
```

```
Rows: 940 Columns: 6
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (4): version, name, item_detailed, item
dbl (2): season, item_number

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.
```

```
episodes <- readr::read_csv(
  'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-01-
```

```
Rows: 98 Columns: 11
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (4): version, title, quote, author
dbl  (6): season, episode_number_overall, episode, viewers, imdb_rating, n_r...
date (1): air_date

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.
```

```
seasons <- readr::read_csv(
  'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2023/2023-01-
```

```
Rows: 9 Columns: 8
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (3): version, location, country
dbl  (4): season, n_survivors, lat, lon
date (1): date_drop_off

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.
```

Extra credit

Data Introduction and Cleaning

There are 4 data sets used as part of this Tidy Tuesday: survivalists, loadouts, episodes, and seasons.

```
survivalists_clean <- survivalists |>
  select(season, name, age, gender, result,
         days_lasted, medically_evacuated,
         reason_category, profession)

exis_professions <- survivalists_clean |>
  select(profession) |>
  group_by(profession) |>
  summarize()

# Diving the professions into categories
wilderness_survival <- exis_professions |>
  filter(str_detect(profession, 'Survival|Wild|Primitive|Skil|Outdoor')) |>
  mutate(category = 'Wilderness/Survival')

business <- exis_professions |>
  filter(str_detect(profession, 'Manager|Accountant|Commercial')) |>
  mutate(category = 'Business')

construction_eng <- exis_professions |>
  filter(str_detect(profession, 'Engineer|Construct|Electr|Build|Operat|Make')) |>
  mutate(category = 'Construction/Engineering')

sci_nature <- exis_professions |>
```

```
  filter(str_detect(profession, 'Nature|Sci|Bio|Env|Agr|Anthro')) |>
  mutate(category = 'Science/Nature')

military_law <- exis_professions |>
  filter(str_detect(profession, 'US|Army|Navy|Enforcement')) |>
  mutate(category ='Military/Law')

health <- exis_professions |>
  filter(str_detect(profession, 'Herbalist|Physician|Psychotherapist')) |>
  mutate(category = 'Health/Fitness')

media <- exis_professions |>
  filter(str_detect(profession, 'Photo|Writer|Media|Video|Author')) |>
  mutate(category = 'Media')

# Creating a unified dataframe with the professions and respective categories
prof_categories <- rbind(wilderness_survival,
                         business, construction_eng, sci_nature,
                         military_law, health, media)

# Delete the miscategorized rows
prof_categories <- prof_categories[-31,]

# Add professions that were left out from defined categories
all_prof_categories <- merge(exis_professions, prof_categories, all.x = TRUE)
all_prof_categories[is.na(all_prof_categories)] <- 'Other'

# Merge the two data frames to include profession categories
survivalists_clean <- merge(survivalists_clean, prof_categories, by = "profession")

# Merge original data frame to include the number of
#participants that have a certain profession per category
temp_graph <- survivalists_clean |>
  select(result, category) |>
  group_by(category, result) |>
  summarize(count = n())
```

```
`summarise()` has grouped output by 'category'. You can override using the
`.groups` argument.
```

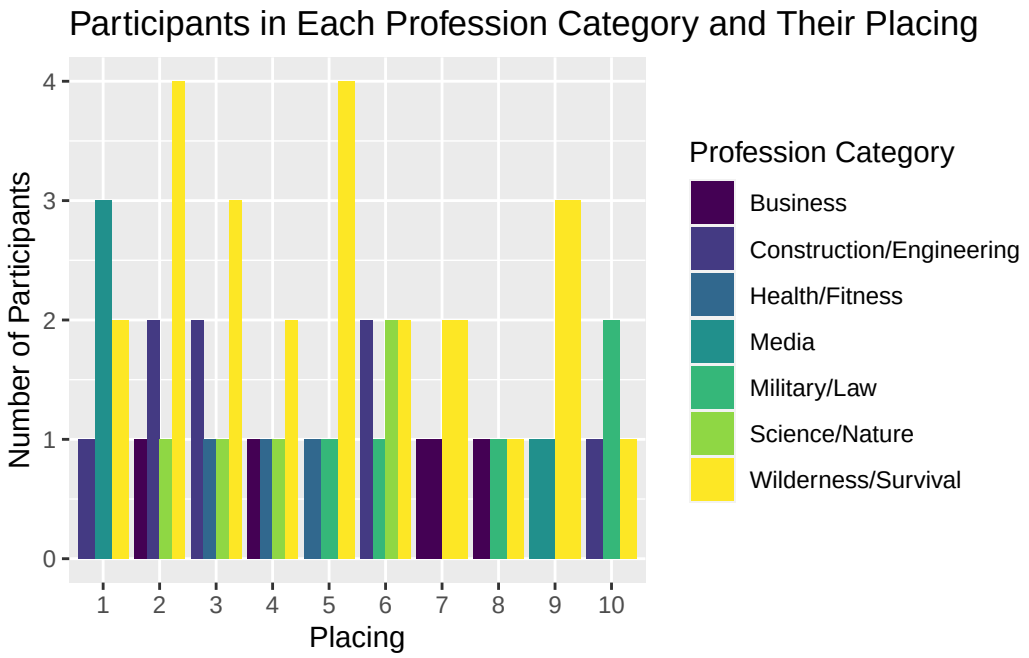
In the cleaning process of survivalists data frame I have selected the relative columns: `season`, `name`, `age`, `gender`, `result`, `days_lasted`, `medically_evacuated`, `reason_category`, and

`profession`. For answering research questions relating to participants' profession, I have grouped the existing professions together with common key words, such as all the professions that have "survival" and "wild" into a profession category group called "Wilderness/Survival". I have chosen the key words and the grouping of the profession categories based on how similar those professions can be considered in the scope of this game. An upside for this method is that I didn't have to select and place all the professions by hand one-by-one, however, a downside for this method is that some of the professions that would usually end up in a certain profession category would be left out because they don't have the key root/word that I have used to group the professions into profession groups. For that reason and for all the professions that are unique and do not fall under my defined categories, I have also created the group "other" for anything that does not fit into any other profession category. Therefore, I have 7 profession categories (Wilderness/Survival, Business, Construction/Engineering, Science/Nature, Military/Law, Health/Fitness, Media) and one additional group called "Other" for the professions that are outside of these defined profession categories.

Profession and Placing

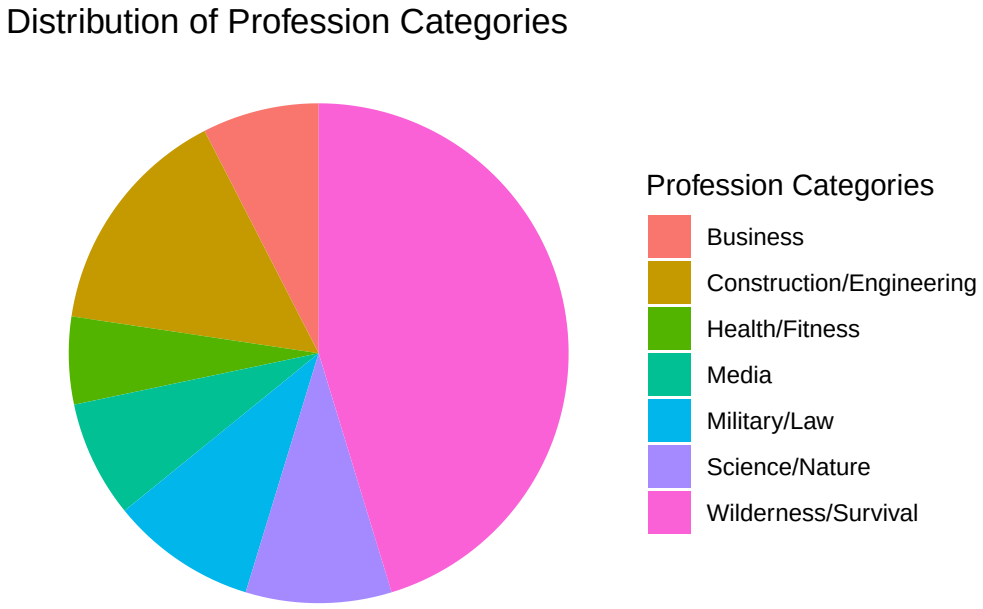
In this section I am visualizing the relationship between how participants performed and the correlation (if any) between their profession category.

```
temp_graph <- temp_graph |>
  mutate(result = factor(result, levels = as.character(1:10)))

ggplot(temp_graph, aes(x = result, y = count, fill = category)) +
  geom_col(position = "dodge") +
  scale_fill_viridis(discrete = TRUE, option = "D") +
  labs(title = "Participants in Each Profession Category and Their Placing",
       y = "Number of Participants",
       x = "Placing",
       fill = "Profession Category")
```



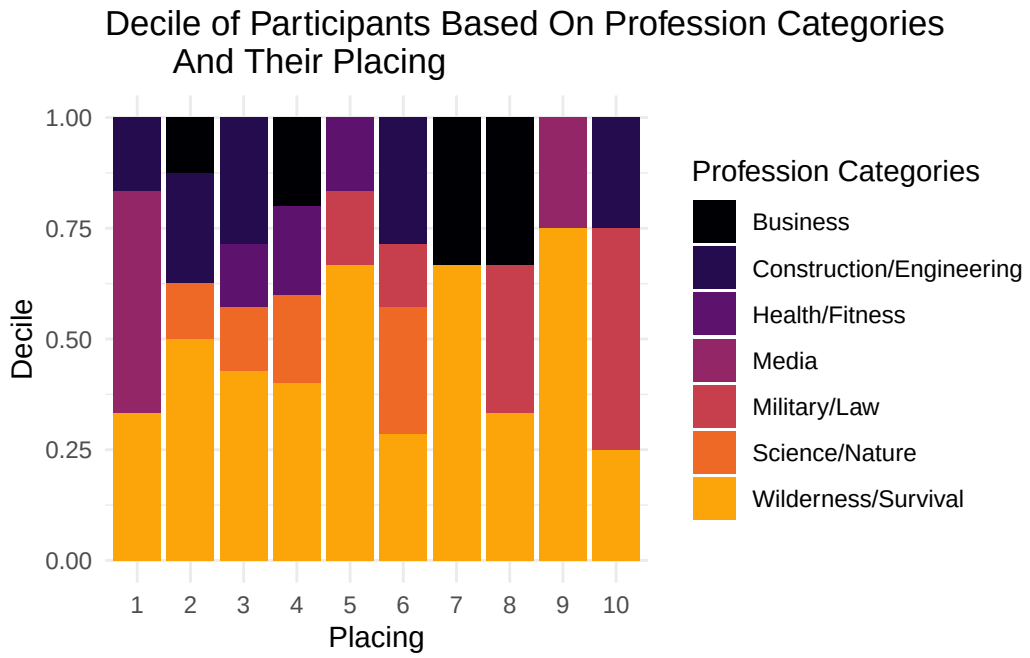
This graph conveys information about how many participants from given profession groups have achieve a certain placing in the game throughout all the seasons. If we follow the graph, we would interpret the results as participants who work for the Media field have won the most times. This said, this graph is biased in a way that it does not account for uneven distribution of certain profession groups. The graph also shows that participants who have worked in fields related to Wilderness and Survival have dominated throughout all positions, which also could be a result of uneven number of participants who have professions relating to that field and have an unfair number advantage over other profession categories.

```
category_count <- survivalists_clean |>
  group_by(category) |>
  summarise(count = n()) |>
  mutate(percentage = count / sum(count) * 100)

ggplot(category_count, aes(x = "", y = count, fill = category)) +
  geom_col(width = 1) +
  coord_polar(theta = "y") +
  theme_void() +
  labs(title = "Distribution of Profession Categories") +
  guides(fill = guide_legend(title = "Profession Categories"))
```



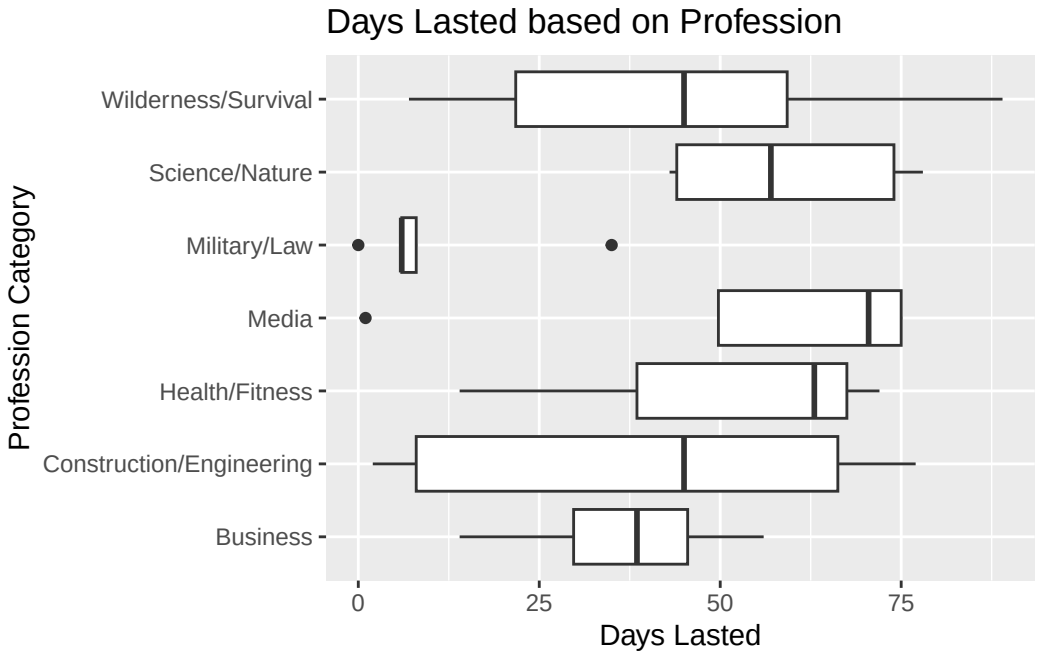
The pie-chart above proves the skewness of the initial bar charts that attempted to show the correlation between participants' profession category and their placing in the game. As assumed, the participants that have professions related to Wilderness and Survival dominate all the other profession groups. This makes sense, given the context of the show. This having said, a normalized stacked bar chart would remove the bias as it normalizes the frequency and does take into consideration the fact that there are different number of participants in each different profession group.

```
temp_graph <- temp_graph |>
  mutate(result = factor(result, levels = as.character(1:10)))


ggplot(temp_graph, aes(x = result, y = count, fill = category)) +
  geom_bar(position = "fill", stat = "identity") +
  scale_fill_viridis_d(option = "B", end = 0.8) +
  theme_minimal() +
  labs(title = "Decile of Participants Based On Profession Categories
       And Their Placing", x = "Placing", y = "Decile",
       fill = "Profession Categories")
```



From this normalized frequency bar chart we can deduce that the most participants who have won the game throughout the considered seasons had professions related to Media, Wilderness/Survival, and Construction/Engineering accordingly. It is also important to deduce that the choice of participants have not been random and even throughout the seasons, and as such, we cannot confidently say that there is a strong relationship between placing and any profession category.

```
ggplot(survivalists_clean,
       mapping = aes(x = days_lasted,
                     y = category))+
  geom_boxplot() + labs(title="Days Lasted based on Profession",
                        x = "Days Lasted", y = "Profession Category")
```



In addition to the previous analysis where I investigated the relationship between participants' profession category and their placing in the game, I have created a box-and-whiskers graph that shows how many days (usually) participants last based on their profession category.

First, we can interpret the median of days lasted for each profession category. Media profession category seems to have the highest median number of days lasted in the show. This can also be affected by the fact that there were relatively few people who fall under this profession category and those people have performed well compared to participants who are identified with other profession categories. The unevenness of the number of participants in this group makes this result biased and not very reliable. However, given the limited number of participants, those participants have outperformed participants in other profession groups. Second in order are Health/Fitness and Science/Nature profession groups. This would be reasonable in the context of the show as it is assumed that participants with knowledge of health and fitness are more fit and ready to participate in challenges and participants knowledgible in science and nature can survive in wilderness. Next in order of days lasted come participants who are identified with Construction/Engineering and Wilderness/Survival. These results do not show much as the box is spread throughout almost all the grid, meaning there is no set performace for the whole group and participants differed significantly in their performance. Lastly, there are participants with professions in Business and Military/Law profession groups whose median and overall performance is shown to be the worst compared to other profession groups. This boxes are more accurate as there is not much variability/dispersion among all the participants and boxes are small. There are dots that indicate outliers. There are outliers for worse than the median performance for Military/Law and Media profession categories. And there is an

outlier that indicates a significantly better than the median performance for Military/Law profession category.

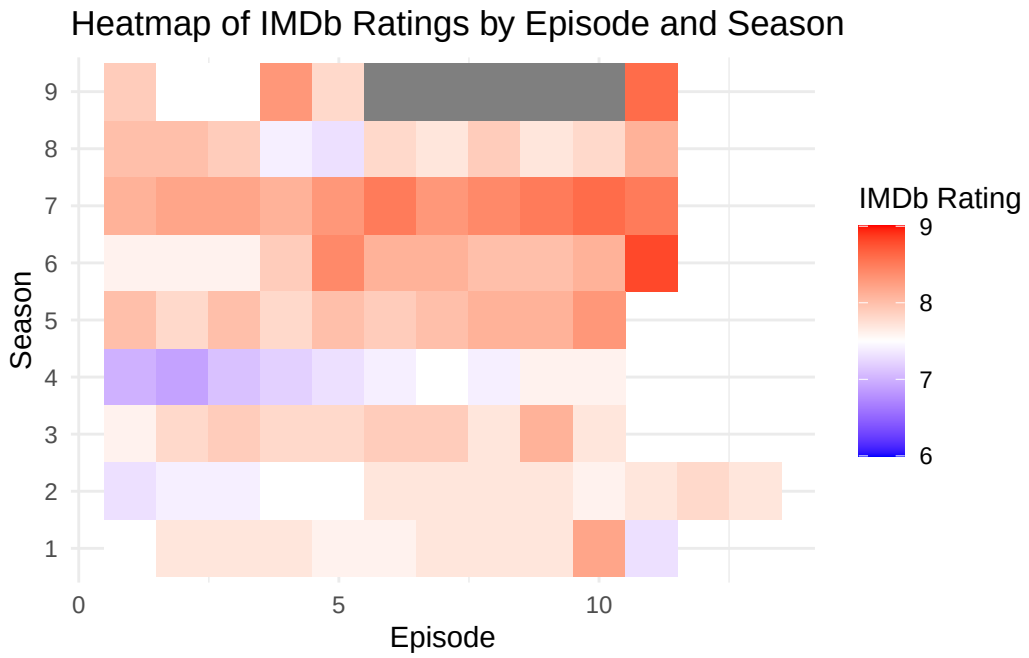
IMDB Rating

This section investigates the relationship between seasons, episodes, and IMDB ratings.

```
joined_data <- survivalists_clean |>
  inner_join(episodes, by = "season") |>
  select(season, name, medically_evacuated,
         episode_number_overall, episode, viewers,
         imdb_rating, reason_category)
```

```
Warning in inner_join(survivalists_clean, episodes, by = "season"): Detected an unexpected ma
i Row 1 of `x` matches multiple rows in `y`.
i Row 12 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

```
#ggplot documentation for geom_tile()
ggplot(joined_data, aes(x = episode,
                        y = as.factor(season), fill = imdb_rating)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", mid = "white",
                       high = "red", midpoint = 7.5,
                       limits = c(6, 9), name = "IMDb Rating") +
  theme_minimal() +
  labs(title = "Heatmap of IMDb Ratings by Episode and Season",
       x = "Episode",
       y = "Season")
```



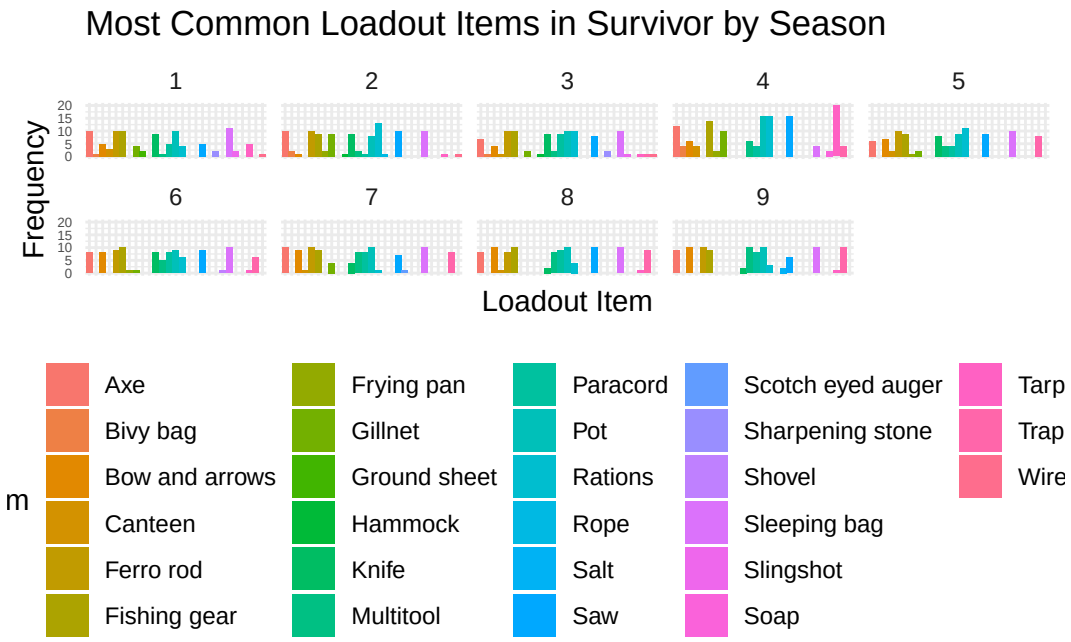
The heat map shows the overall IMDB rating of an episode in a season. From this graph we can deduce which episodes/seasons were the most liked by the viewers and which ones were the most disliked. The IMDB socre scale that is indicated by the color of a cell is limited between 6 and 9, as the lowest score present in the data is 6.9 and the highest score data is 8.8. The narrower the scale range is, the more contrast the graph will display, allowing us to compare the episodes and seasons more accurately.

From this heat map we can see that some seasons were particularly high-ranked while others were particularly disliked by the viewers. Season 4 displays the lowest IMDB rankings across all seasons for almost every episode, with the exception for the last 2 episodes were the rankings get neutralized. Season 7 on the other hand seems to be the most overall liked and highly ranked season of all seasons present in the data set. The gray color that is displayed for some of the episodes is season 9 indicates that the ranking is missing from the data set.

Loadouts Frequency and Significance

In this section I will be investigating the frequency of certain loadouts that participants would occasionally win during their stay in the game to help them with the game and make progress in their survival, and how effective and helpful these loadouts are in terms of assisting the participants to last longer (or even win the game).

```r
loadout_summary <- loadouts |>
  group_by(item) |>
  summarise(frequency = n()) |>
  arrange(desc(frequency))

ggplot(loadouts, aes(x = item, fill = item)) +
  geom_bar() +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_text(size = 5, angle = 0),
        legend.position = "bottom"
        ) +
  labs(x = "Loadout Item",
       y = "Frequency",
       title = "Most Common Loadout Items in Survivor by Season",
       color= "Item") +
  facet_wrap(~season, ncol = 5, nrow = 2)
```



The graph above shows the frequency of loadouts given faceted by a season. In general, the loadouts seem to be distributed evenly across seasons with no outstanders. However, we can still observe that certain seasons distributed more or less of some loadouts compared across all the seasons. For example, in season 4, a significantly more number of tarps have been distributed compared to other seasons. Meanwhile, items such as wires were not popular in

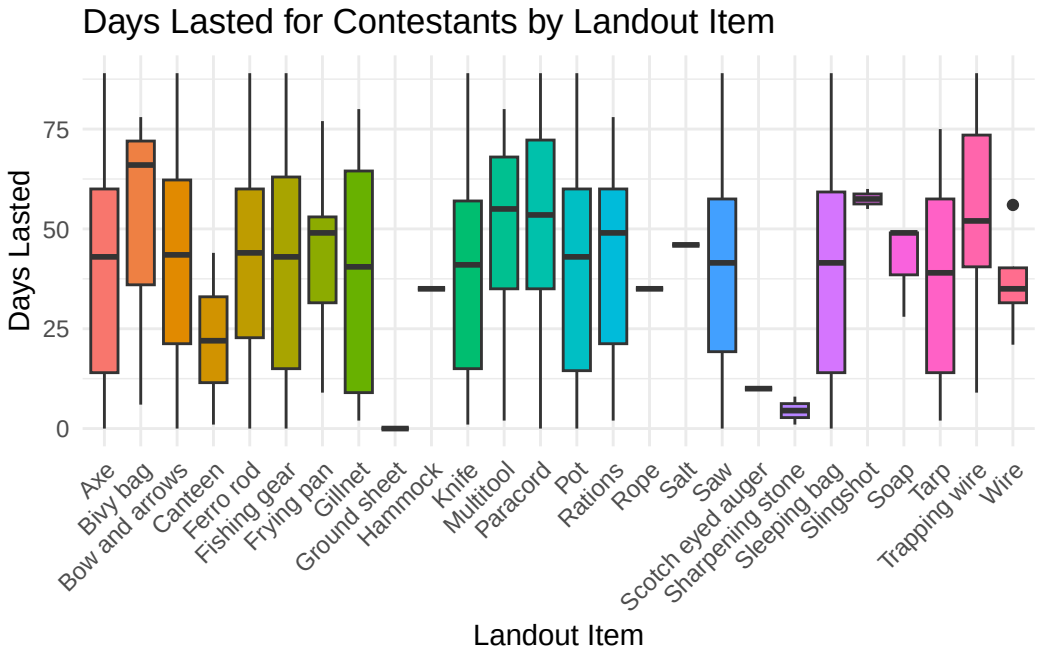
the earlier seasons of the show but gradually became more and more popular.

The following graph explores the relationship between the loadouts and how long the participants who received those loadouts stayed in the show.

```
participants_loadout <- survivalists_clean |>
  inner_join(loadouts, by = "name")
```

```
Warning in inner_join(survivalists_clean, loadouts, by = "name"): Detected an unexpected many
i Row 1 of `x` matches multiple rows in `y`.
i Row 291 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

```
ggplot(participants_loadout, aes(x = item,
                                 y = days_lasted, fill = item)) +
  geom_boxplot() +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "none") +
  labs(x = "Landout Item",
       y = "Days Lasted",
       title = "Days Lasted for Contestants by Landout Item")
```



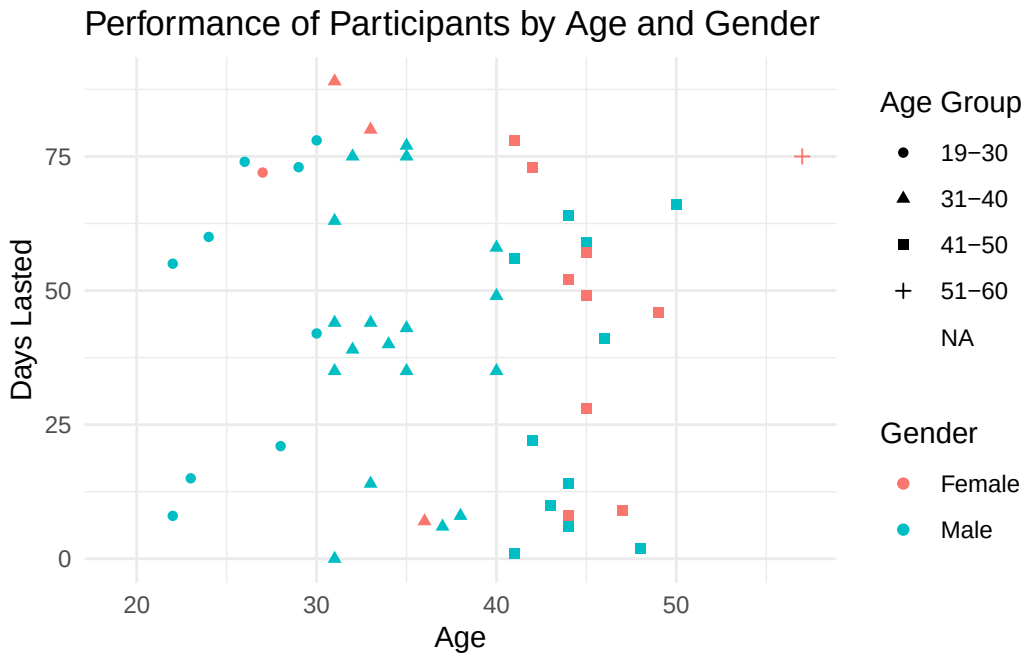
Each box represents a loadout item that was given to the participants and the box-plot shows the distribution of days lasted by the participants based on the loadout items they received. Almost all the box-plots are spread across all the y-axis, which shows a great disparity and variety in the results and does not allow us to conclude anything distinctive about those items. This means that the IQR is large and the data points in the data set are different from each other, meaning the days lasted for one participant who received the loadout can be drastically and significantly different from the days lasted of another participant who received the same loadout. Hence, this data would not allow us to make any unanimous decision about the correlation of the variables. However, there are some items that can be deduced not to be so helpful for the participant's longitude in the show, as the data shows that the median days lasted for the participants who received those items is very low. Those items are ground sheet, scotch eyed auger, and sharpening stone. Additionally, there are other items that would indicate a correlation between them and the days lasted. Those items are hammock, rope, salt, soap, slingshot, and wire. The box-plots of the latter items are smaller/shorter meaning their IQR is smaller which also indicates that there is less variability in the data and the data is more similar to each other and more homogeneous. This can be useful in concluding a correlation as there might be more probability that there is a correlation between a loadout and days lasted if all the results are similar.

Performance of Participants based on Age and Gender

This section explores the distribution of participants in terms of their age and gender and its possible relationship with their performance in the show.

```r
survivalists_clean <- survivalists_clean |>
  mutate(age_group = cut(age, breaks = c(19, 30, 40, 50, 60),
                         labels = c("19-30", "31-40", "41-50", "51-60")))

ggplot(survivalists_clean,
       aes(x = age, y = days_lasted,
           color = gender, shape = age_group)) +
  geom_point() +
  theme_minimal() +
  labs(x = "Age",
       y = "Days Lasted",
       title = "Performance of Participants by Age and Gender") +
  guides(color = guide_legend(title = "Gender"),
         shape = guide_legend(title = "Age Group"))
```

```
Warning: Removed 1 rows containing missing values (`geom_point()`).
```



The graph above shows a slight disproportion of the participants where there seems to be larger number of male participants and larger number of participants between 31 and 50. In terms of the relationship between the age/gender and the performance there seems to be no correlation as all the groups are displayed to perform equally well. This also means that to

succeed in the show participants cannot rely on their age or gender but participants would need other strengths such as strategy and alliances.

Citation

The data sets are taken from Tidy Tuesday initiative:

Thomas Mock (2022). Tidy Tuesday: A weekly data project aimed at the R ecosystem. https://github.com/rfordatascience/tidytuesday.