# Individual Written Proposal

Franklin Zeng

DSCI100: Introduction to Data Science

November 14, 2025

**Data Summary:**

The main data files are *players.csv* and *sessions.csv*. They are attached to the GitHub repository under the *data* folder. The following are observations related to both files. General issues applying to both data files are listed below:

- Missing ages and inconsistent gender reports (this can likely be explained due to users preferring not to report their gender, which is fine)
- Certain sessions may include very short or overlapping durations, which may interfere with tuning
- External factors such as time zones and network speeds are not recorded.

*players.csv*:

- Observations: 196 | Variables: 7
- Quantitative Summary (in means):
  - Played_hours: 22.54 hours
  - Age: 24.32 years
- Variable Summary:

| **Variable** | **Type** | **Description** | **Potential Issues + Notes** |
|---|---|---|---|
| experience | categorical | Player skill level: beginner, amateur, regular, veteran, pro | Needs to be encoded for prediction models |
| subscribe | logical | Whether the user is subscribed to a game-related newsletter or not | Is the response variable for question 1 of the outline |
| hashedEmail | character | Unique user identifier: hashed/encoded for privacy | Is used to join datasets - link data between *players.csv* and *sessions.csv* to |

| | | | |
|---|---|---|---|
| | | | the user |
| played_hours | numeric | Total time played in hours | N/A or zero values indicate inactive users |
| name | character | Player name - randomized for privacy | Not necessary for modelling - there for quality of life viewing |
| gender | categorical | Player gender | Needs to be grouped for analysis |
| Age | numeric | Player age in years | Possible for missing or inconsistent entries to exist |

*sessions.csv:*

- Observations: 1535 | Variables: 5
- Variable Summary:

| **Variable** | **Type** | **Description** | **Notes and Potential Issues** |
|---|---|---|---|
| hashedEmail | character | Unique player ID; common with *players.csv* in order to merge data | Must match *players.csv* for data merging |
| start_time | character | Session start time | Needs to be converted to date/time format |
| end_time | character | Session end time | Needs to be converted to date/time format |
| original_start_time | numeric | Unix timestamp for start time | Alternative time format; backup |

| | | | |
|---|---|---|---|
| original_end_time | numeric | Unix timestamp for end time | Alternative time format; backup |

- Derived variable(s)
    - *duration_mins = end_time - start_time* (in minutes)
    - Application above can also be applied to *original_start_time* and *original_end_time*

---

**Questions**

- Broadly speaking, what player characteristics and behaviors are generally predictive of subscription to a game-related newsletter among users of the Minecraft server application?
- Can the following variables predict newsletter subscription among users of the Minecraft server application:
    - *Experience*
    - *Played_hours*
    - *Age*
    - *Gender*
- The rationale for the above questions are as follows:
    - Variable *subscribe* is a binary variable, greatly suitable for predictive models applied in the course.
    - Player demographics such as age, gender,  and engagement are explanatory variables.

- Data from sessions are summarizable into metric at the user level: total/mean duration, number of log-ons. This is helpful for modelling at the individual level.

---

**Explanatory Data Analysis and Visualization**

- Process Outline:
    1. Load datasets, analyze them visually for errors.
    2. Convert all timestamps to date/time for easier processing.
    3. Calculate mean values of quantitative values in *players.csv*
    4. Visualize explanatory variables:
        a. Histograms for *played_hours* and *age.*
        b. Bar plots for *experience* and *gender.*
        c. Box plots for *played_hours* and *experience.* (Note that *played_hours* is under both histogram and box plot for convenience of analysis and visual analysis).
- Expected Outcomes:
    - Whether high engagement and/or experienced players will subscribe to a game-related newsletter.
    - Identify and process/remove data issues, outliers, and non-relevant data.
    - Create visualizations in the form of graphs for ease of visualization and for convenience for predictive analysis.
- Code for this section is included under *groupprojindividual/dsci100groupproj/writtenproposal/WrittenProposalCode.ipynb*
    - The code will be outlined as such:

        - Load libraries
        - Load datasets from .csv files
            - (Inspect data for ease of visualization)
        - Summarize quantitative variables in *players.csv*
        - Wrangle data from play sessions into tidy data
        - Summarize play session data to player level
        - Merge sessions data into player data
        - Create plots for visualization:
            - Histograms
            - Bar plots
            - Box plots
    - The code will be as such:

```r
library(tidyverse); library(janitor); library(lubridate)
```

```r
library(readr)

players <- read_csv("data/players.csv")
sessions <- read_csv("data/sessions.csv")

players
sessions
```

```r
mean_played_hours <- mean(players$played_hours, na.rm = TRUE)
mean_played_hours <- round(mean_played_hours, 2)
mean_played_hours

mean_age <- mean(players$Age, na.rm = TRUE)
mean_age <- round(mean_age, 2)
mean_age

sessions_tidy <- sessions |>
  mutate(
    start_time = as_datetime(original_start_time, origin = "1970-01-01")
  )

sessions_tidy <- sessions_tidy |>
  mutate(
    end_time = as_datetime(original_end_time, origin = "1970-01-01")
  )

sessions_tidy <- sessions_tidy |>
  mutate(
    duration_mins = as.numeric(difftime(end_time, start_time, units = "mins"))
  )

player_sessions_summary <- sessions_tidy |>
  group_by(hashedEmail) |>
  summarise(
    total_sessions = n()
  )

player_sessions_summary <- player_sessions_summary |>
  left_join(
    sessions_tidy |>
      group_by(hashedEmail) |>
      summarise(total_duration = sum(duration_mins, na.rm = TRUE)),
    by = "hashedEmail"
  )

player_sessions_summary <- player_sessions_summary |>
  left_join(
    sessions_tidy |>
      group_by(hashedEmail) |>
      summarise(mean_duration = mean(duration_mins, na.rm = TRUE)),
    by = "hashedEmail"
  )

players_full <- players |>
  left_join(player_sessions_summary, by = "hashedEmail")

ggplot(players_full, aes(x = played_hours)) +
  geom_histogram(binwidth = 5, fill = "blue", color = "white") +
  labs(title = "Distribution of Played Hours",
       x = "Played Hours", y = "Count")

ggplot(players_full, aes(x = Age)) +
  geom_histogram(binwidth = 2, fill = "green", color = "white") +
  labs(title = "Age Distribution of Players",
       x = "Age", y = "Count")

ggplot(players_full, aes(x = experience)) +
  geom_bar(fill = "purple") +
  labs(title = "Player Experience Levels",
       x = "Experience", y = "Count")

ggplot(players_full, aes(x = experience, y = played_hours)) +
  geom_boxplot(fill = "orange") +
  labs(title = "Played Hours by Experience Level",
       x = "Experience", y = "Played Hours")

ggplot(players_full, aes(x = gender)) +
  geom_bar(fill = "pink") +
  labs(title = "Gender Distribution of Players",
       x = "Gender", y = "Count")
```

---

## Methodology and Proposed Plan

- The preferred method for analysis is logistic regression. However, an alternative will be provided utilizing K-NN and linear regression.
    - Logistic regression is best due to its use in modelling binary outcomes - in this case *subscribe*.
    - We are assuming that observations are independent + linearity of logit for continuous variables.
    - Problems with this method include its sensitivity to multicollinearity. It will also struggle to visualize non-linear trends.
    - Alternative ways to strengthen the prediction are to utilize tree-based outcomes like decision trees for backup. Note that if this is used cross-validation for hyper-parameter tuning will be needed.
    - The data processing flow will be as follows:
        - Splitting the data in a 70% training and 30% testing split.
        - The categorical variables will be encoded.