Payoff Distribution: Finite vs. Infinite Games

Hana Kwon

2024-11-20

```
library(tidyverse)
```

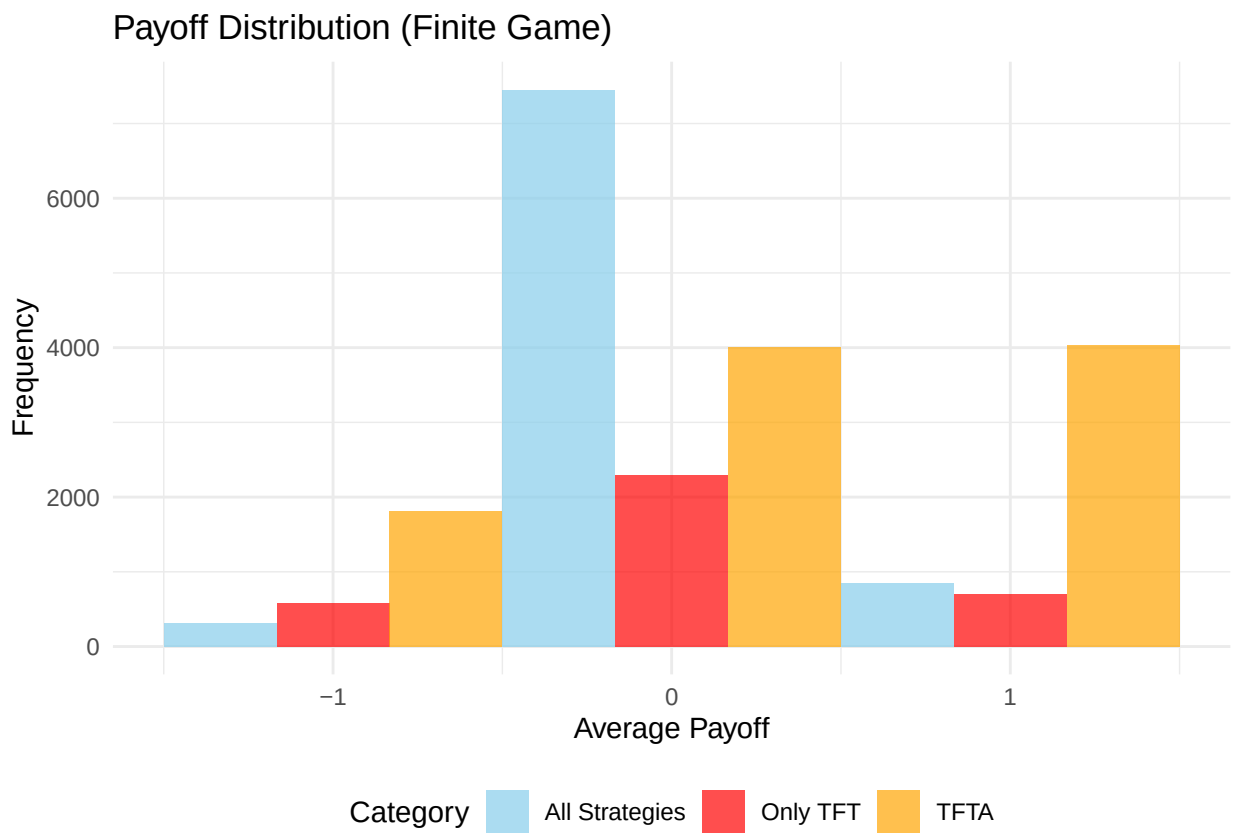
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
```

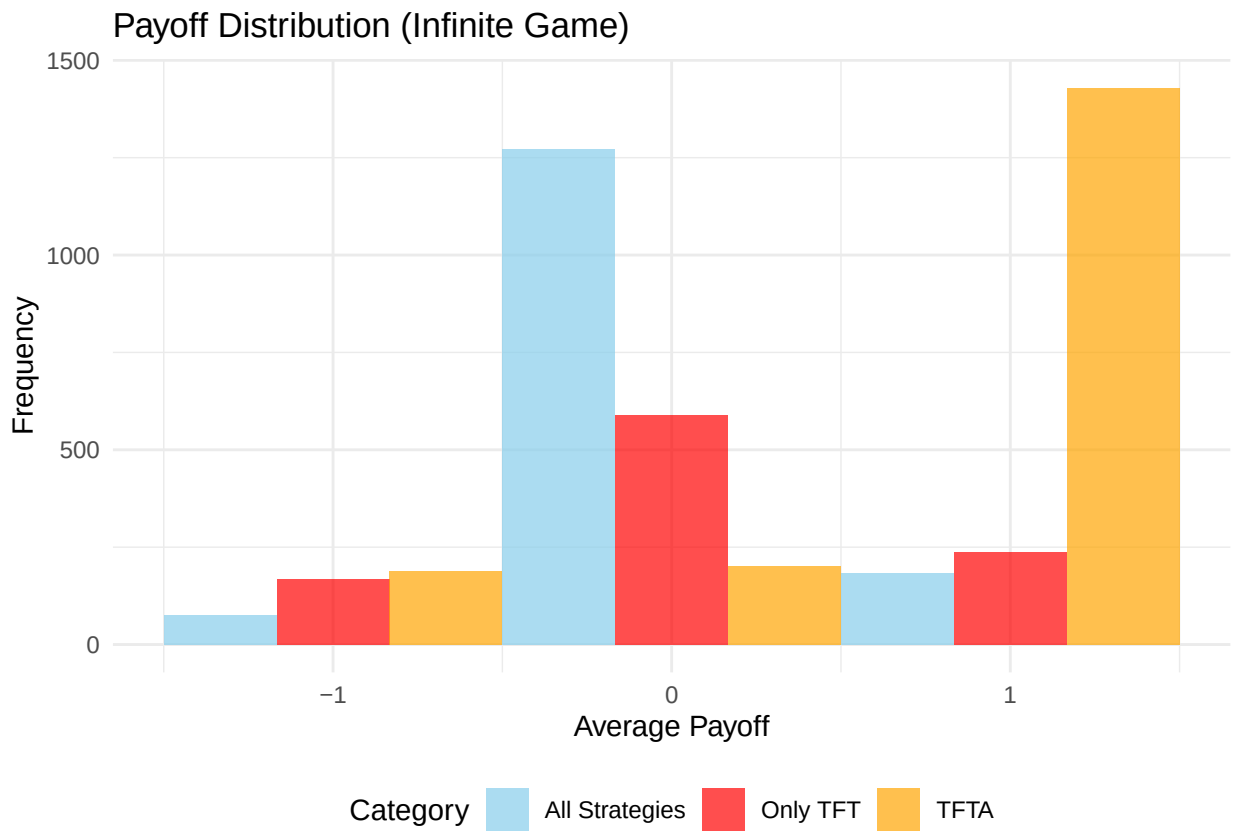
```
# Load the data
data <- read.table("~/Columbia Dropbox/Kwon Hana/Prof. W.Bentley MacLeod & Hana Kwon/Data/Embrey_2018a_
                   header = TRUE, sep = "\t", stringsAsFactors = FALSE)
data0 <- read.table("~/Columbia Dropbox/Kwon Hana/Prof. W.Bentley MacLeod & Hana Kwon/Data/Embrey_2018a_
                    header = TRUE, sep = "\t", stringsAsFactors = FALSE)
data <- bind_rows(data, data0) %>% select(-paper, -order)

# Arrange and process the data
sorted_data <- data %>%
  arrange(session, id, supergame, round) %>%
  left_join(
    data %>% select(session, supergame, round, id = oid, ocoop = coop),
    by = c("session", "supergame", "round", "id")
  ) %>%
  mutate(
    TFT = ifelse(round == 1, 1, lag(ocoop)),
    G0 = ifelse(round == 1, 1, ifelse(lag(ocoop) == 0, 0, ifelse(lag(coop) == 0, 0, 1))),
    r = 1.0, s = -1.0, t = 1.5, p = 0.0,
    payoff = case_when(
      coop == 1 & ocoop == 1 ~ r,
      coop == 1 & ocoop == 0 ~ s,
      coop == 0 & ocoop == 1 ~ t,
      coop == 0 & ocoop == 0 ~ p
    ),
    category = case_when(
      TFT == 1 & G0 == 0 ~ "Only TFT",
```

```
      TFT == 1 ~ "TFTA",
      TRUE ~ "All Strategies"
    )
  )
```

```
# Filter and collapse Finite Game data
finite_data <- sorted_data %>%
  filter(horizon < 10) %>%
  group_by(session, id, supergame, category) %>%
  summarise(avg_payoff = mean(payoff, na.rm = TRUE), .groups = "drop")

# Histogram for Finite Game
generate_histogram <- function(data, title) {
  ggplot(data, aes(x = avg_payoff, fill = category)) +
    geom_histogram(binwidth = 1, position = "dodge", alpha = 0.7) +
    scale_fill_manual(values = c("All Strategies" = "skyblue", "Only TFT" = "red", "TFTA" = "orange")) +
    labs(
      title = title,
      x = "Average Payoff",
      y = "Frequency",
      fill = "Category"
    ) +
    theme_minimal() +
    theme(legend.position = "bottom")
}

finite_plot <- generate_histogram(
  finite_data,
  "Payoff Distribution (Finite Game)"
)
finite_plot
```




```
# Filter and collapse Infinite Game data
infinite_data <- sorted_data %>%
  filter(horizon == 10) %>%
  group_by(session, id, supergame, category) %>%
  summarise(avg_payoff = mean(payoff, na.rm = TRUE), .groups = "drop")

# Histogram for Infinite Game
infinite_plot <- generate_histogram(
  infinite_data,
  "Payoff Distribution (Infinite Game)"
)
infinite_plot
```




```
ggsave("Finite_Game_Payoff_Distribution.pdf", plot = finite_plot, width = 10, height = 6)
ggsave("Infinite_Game_Payoff_Distribution.pdf", plot = infinite_plot, width = 10, height = 6)
```