Data Exploration: Cooperation

Yanxi Fang

September 16, 2021

The Prisoner's Dilemma Simulator

Setup

As instructed, the Shiny app was run and the `prisoners_dilemma_data.csv` file was compiled and saved.

```
# read and modify data
pd_data <- read_csv("prisoners_dilemma_data.csv") %>%
  mutate(winner = case_when(
    score1 > score2 ~ player1,
    score1 < score2 ~ player2,
    score1 == score2 ~ "tie"
  ))
```

Question 1

How do the strategies you chose rank against each other based on the number of wins? How do they rank based on the number of points won? Discuss the patterns you see here as they relate to what you read from Axelrod. Keep in mind the concepts of niceness, retaliation, forgiveness, and clarity.

```
# original, provided code to generate `player_data_long`
player1_data <- pd_data %>% select(player = player1, score = score1, opponent = player2)
player2_data <- pd_data %>% select(player = player2, score = score2, opponent = player1)
player_data_long <- bind_rows(player1_data, player2_data)

# additional code to rearrange `player_data_long`
# this code sums up the scores corresponding to each strategy
player_data_scores <- player_data_long %>%
  group_by(player) %>%
  summarize(sum(score))
print(player_data_scores)
```

```
## # A tibble: 6 x 2
##   player      `sum(score)`
##   <chr>              <dbl>
## 1 backstabber        1576.
## 2 cooperator         1266.
## 3 cycler              931.
```

```
## 4 foolMeOnce         1577.
## 5 random             1075.
## 6 titfortat          1391.
```

```
# rank strategies by number of wins, not total score
table(pd_data$winner)
```

```
##
## backstabber      cycler  foolMeOnce      random         tie
##           5           2           2           3           3
```

To provide context: for the "tournament," I chose six strategies: `backstabber`, `cooperator`, `cycler`, `foolMeOnce`, `random`, and `titfortat`.

Based on the number of wins, the `backstabber` strategy is dominant with 5 wins, which is 2 more wins than the second-best strategy, `random`. Based on the total score won, the `foolMeOnce` strategy is dominant, with 1576.64 points, while `backstabber` is in a very close second with score of 1576.16 points.

These results are interesting in light of Axelrod's claim that the `titfortat` strategy is the most successful, although obviously, this comparison of just 6 strategies is likely far from the actual tournament that Axelrod ran to generate his results. Nonetheless, the patterns observed here make sense: both the `backstabber` and `foolMeOnce` strategies employ strong enforcement mechanisms (the third – or second – defect leads to permanent defection, respectively), showing a small degree of forgiveness but simultaneously a strong degree of retaliation. The fact that these strategies prevailed is also consistent with Axelrod's finding that niceness doesn't lead to a high-performing strategy. Finally, with regard to clearness, both strategies are laid out clearly, but `foolMeOnce` is somewhat easier to code and understand (there is only one condition) than `backstabber`, a strategy that always stipulates defection during the last two rounds.

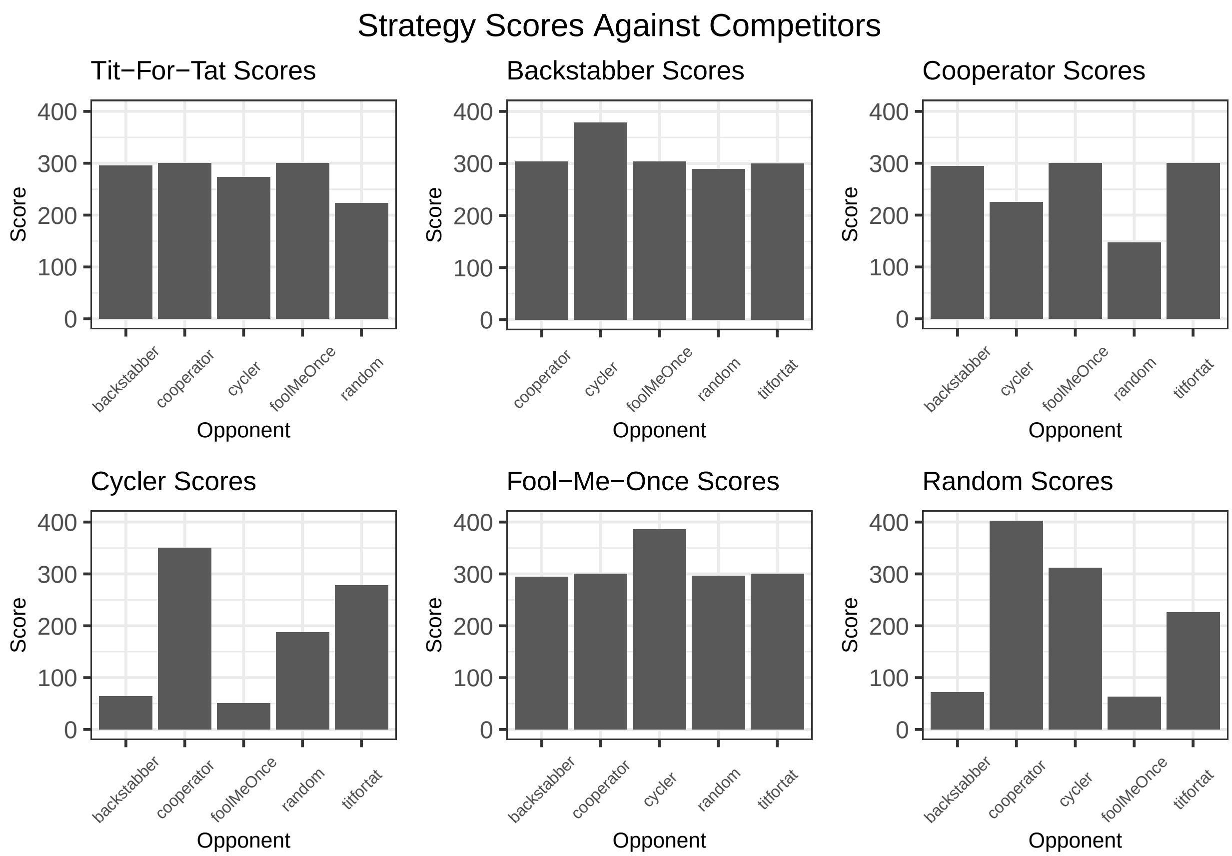
Question 3: Data Science Question

Create a plot, similar to the one above, that displays how each strategy fared against all the other strategies. Which strategies were most successful against which other ones? Comment on the patterns that you see.

```
# load additional package for graphing
library(gridExtra)

# plot for each of the 6 categories, against the remaining 5 other categories
graph1 <- player_data_long %>%
  filter(player == 'titfortat') %>%
  ggplot(aes(opponent)) +
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Tit-For-Tat Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

graph2 <- player_data_long %>%
  filter(player == 'backstabber') %>%
```

```r
  ggplot(aes(opponent)) +
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Backstabber Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

graph3 <- player_data_long %>%
  filter(player == 'cooperator') %>%
  ggplot(aes(opponent)) +
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Cooperator Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

graph4 <- player_data_long %>%
  filter(player == 'cycler') %>%
  ggplot(aes(opponent)) +
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Cycler Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

graph5 <- player_data_long %>%
  filter(player == 'foolMeOnce') %>%
  ggplot(aes(opponent)) +
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Fool-Me-Once Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

graph6 <- player_data_long %>%
  filter(player == 'random') %>%
  ggplot(aes(opponent)) +
```

```r
  geom_bar(aes(weight = score)) +
  ylim(0, 402) +
  xlab("Opponent") +
  ylab("Score") +
  ggtitle("Random Scores") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 6, angle = 45, vjust = 0.5),
        plot.title = element_text(size = 10),
        axis.title.x = element_text(size = 8), axis.title.y = element_text(size = 8))

grid.arrange(graph1, graph2, graph3, graph4, graph5, graph6, ncol = 3,
             top = "Strategy Scores Against Competitors")
```



The plots that display how each of the six strategies fared against the other five are shown above.

Based on the plots, it appears that `titfortat`, `backstabber`, and `foolMeOnce` were the three most consistent strategies in the sense that all three scored a large number of points against every one of their competitors; the other three strategies had much spottier performance records. More specifically, (as previously discussed in Q1), it seems that either `backstabber` or `foolMeOnce` was the most successful strategy, since both beat out even `titfortat` with scores that were consistently at or above 300. Meanwhile, it is clear that `titfortat` still has high scores against all competitors, but perhaps more importantly, the values of the scores seem more stable across each of the opponents, which may be the reason why Axelrod found it to be the most successful overall.

Another pattern that I see is that the "non-strategic" strategies (i.e. `cycler`, `random`, both of which do not respond directly to the opponent's actions) are not particularly successful in obtaining high scores, while the strategy representing niceness (i.e. `cooperator`) actually performed reasonably well, except against the "non-strategic" strategies. At the same time, as discussed in Q1, the more punitive (less nice) strategies did better as a whole, showing the value of punishment in enforcing the idea of the shadow of the future (i.e. punishment to ensure that the opponent starts cooperating, and thus valuing future rewards).