

Promotions and Exits

Tate Huffman

2/1/2022

Overview

This is part of a series of RMarkdown files that will break down the code contained in the document at the end of the fall. This file in particular will combine the recalculated information for promotions and exits into one dataframe in order to more accurately calculate a true application of the O'Flaherty and Siow model.

Exits, Promotions, and Total Movement

In this file, the code to isolate exit and promotion movement has been separated into their own code chunks. For batters and pitchers, it's determined whether a player ever exits, and in turn the percentage of the eligible player pool that exits in each season, up to and including ten seasons. It is found that about 8.2% of all players do not exit after ten seasons, and instead continue their careers, whether in the major leagues or in the minors.

```
# Batter exits

n_bat <- bat_performance %>% # number of unique batters meeting the PA threshold
  select(Name) %>%
  pull() %>%
  unique() %>%
  length()

exit_distr_bat <- bat_performance %>%
  group_by(Name) %>%
  summarise(n_year = n_distinct(Year),
            ever_exit = if_else(max(exit) == 1, 1, 0)) %>% # whether player ever exits
  group_by(ever_exit, n_year) %>%
  summarise(n_players = n_distinct(Name)) %>%
  ungroup() %>%
  # adds row for players who never exit (i.e., have 10+ year careers)
  add_row(ever_exit = 0,
          n_year = 10,
          n_players = sum(pull(select(filter(., ever_exit == 0), n_players)))) %>%
  slice(8:18) %>%
  mutate(pct = n_players / n_bat,
         pos = 'Batters')

# Pitcher exits

n_pitch <- pitch_performance %>% # number of unique pitchers
  select(Name) %>%
  pull() %>%
```

```

unique() %>%
length()

exit_distr_pitch <- pitch_performance %>% # distribution of player exit times
  group_by(Name) %>%
  summarise(n_year = n_distinct(Year),
            ever_exit = if_else(max(exit) == 1, 1, 0)) %>% # whether player ever exits
  group_by(ever_exit, n_year) %>%
  summarise(n_players = n_distinct(Name)) %>%
  ungroup() %>%
  # adds row for players who never exit (i.e., have 10+ year careers)
  add_row(ever_exit = 0,
          n_year = 10,
          n_players = sum(pull(select(filter(., ever_exit == 0), n_players)))) %>%
  slice(7:17) %>%
  mutate(pct = n_players / n_pitch,
         pos = 'Pitchers')

# Merging the batter and pitcher dataframes
# This has the combined info for all qualified players

n_total <- n_bat + n_pitch

exit_distr <- exit_distr_bat %>%
  bind_cols(exit_distr_pitch) %>%
  mutate(n_year = n_year...2,
         n_players = n_players...3 + n_players...8,
         pct = n_players / n_total,
         ever_exit = ever_exit...1) %>%
  select(n_year, n_players, pct, ever_exit)

```

For promotions, similar work is done, but with multiple promotions possible in a single season, the probabilities of a promotion sum to greater than 1, so the percent of players promoted in a given year is divided by the average number of promotions experienced by a given player in order to normalize probabilities.

```

# Batter promotions

promote_distr_bat <- bat_performance %>%
  group_by(yr_unique) %>%
  summarize(n_players = sum(promotion),
            pct = n_players / n_bat)

# Pitcher promotions

promote_distr_pitch <- pitch_performance %>%
  group_by(yr_unique) %>%
  summarize(n_players = sum(promotion),
            pct = n_players / n_pitch)

# Weighting the percent promoted by the average number of promotions
# This is to prevent probabilities from surpassing 1

performance <- bat_performance %>%
  bind_rows(pitch_performance)

```

```

n_promotions <- performance %>%
  group_by(Name) %>%
  summarize(promotions_total = sum(promotion)) %>%
  select(promotions_total) %>%
  pull() %>%
  mean()

# Combined promotions

promote_distr <- bat_performance %>%
  group_by(yr_unique) %>%
  summarize(n_players = sum(promotion)) %>%
  bind_cols(pitch_performance %>%
    group_by(yr_unique) %>%
    summarize(n_players = sum(promotion))) %>%
  mutate(n_years = yr_unique...1,
         n_players = n_players...2 + n_players...4,
         pct = (n_players / n_total) / n_promotions) %>% # weighting by total promotes
         # pct = (n_players / n_total)) %>%
  select(n_years, n_players, pct)

```

The exit and promotion information is then combined into one table in the form of the O'Flaherty and Siow model and saved as a CSV file, as are the individual tables for exits and promotions.

```

# Compiling data into yearly exit/promotion table

player_mvmt <- exit_distr %>%
  slice(1:10) %>% # ignoring the 8.2% that play for over ten years
  select(pct) %>%
  bind_rows(promote_distr %>% select(pct)) %>%
  mutate(# pct = pct / sum(. $pct), # normalizing to sum to 1 - probably not accurate?
         year = rep(seq(1:10), 2),
         exit = c(rep(T, 10), rep(F, 10)),
         cell = seq(1:20))

# Writing this info to CSV files

exit_distr %>%
  write_csv('../Data/Clean/exit_distr.csv')

promote_distr %>%
  write_csv('../Data/Clean/promote_distr.csv')

player_mvmt %>%
  write_csv('../Data/Clean/mvmt_distr.csv')

```