Supplementary Material

Trevor Greissinger

Data scraping/loading

This code should not be run as the scraping process takes a very long time.

```
drafts <- as.character(c(2008:2017))
draft.data <- get_drafts("nhl entry draft", drafts)


seasons <- c("2007-2008","2008-2009","2009-2010","2010-2011","2011-2012","2012-2013",
             "2013-2014","2014-2015","2015-2016","2016-2017")
ncaa.teams <- get_teams("ncaa", seasons)
ncaa.teams.1 <- ncaa.teams[1:150,]
ncaa.teams.2 <- ncaa.teams[151:300,]
ncaa.teams.3 <- ncaa.teams[301:450,]
ncaa.teams.4 <- ncaa.teams[451:600,]


ncaa.players.1 <- my_get_player_stats_team(ncaa.teams.1)
ncaa.players.2 <- my_get_player_stats_team(ncaa.teams.2)
ncaa.players.3 <- my_get_player_stats_team(ncaa.teams.3)
ncaa.players.4 <- my_get_player_stats_team(ncaa.teams.4)
all.ncaa.players <- bind_rows(ncaa.players.1, ncaa.players.2,
                              ncaa.players.3, ncaa.players.4)
```

```
ncaa.drafted <- all.ncaa.players %>%
  filter(name %in% draft.data$name) %>%
  arrange(name, team) %>%
  select(-(games_played_playoffs:plus_minus_playoffs))
```

Cleaning

Some players in the NCAA who weren't drafted had the same name as a player who was drafted. These have been filtered out of the data.

```
bad.player <- c("Brian Cooper","Nick Larson","Ryan Jones","Chris Wagner")
bad.team <- c("Lake Superior State Univ.","UMass-Lowell","Univ. of Minnesota",
              "Miami Univ. (Ohio)","Sacred Heart Univ.")


ncaa.drafted.clean <- all.ncaa.players %>%
  filter(name %in% draft.data$name,
         !(name %in% bad.player & team %in% bad.team),
         !(name == "Ryan Collins" & team == "UMass-Lowell"),
         !(name == "Chris Brown" & team == "Sacred Heart Univ."),
         !(name == "Chris Brown" & team == "Boston College"),
         games_played >= 10)
```

Scraping career statistics

```
indiv.players <- ncaa.drafted.clean %>%
  group_by(name, team) %>%
  mutate(school.year = row_number()) %>%
  ungroup() %>%
  filter(school.year == 1) %>%
  select(-school.year)
```

Again, this scraping code should not be run.

```
career.stats.1 <- indiv.players[1:125,]
career.stats.2 <- indiv.players[126:250,]
career.stats.3 <- indiv.players[251:375,]
career.stats.4 <- indiv.players[376:518,]


stats.1 <- my_get_player_stats_individual(career.stats.1)
stats.2 <- my_get_player_stats_individual(career.stats.2)
stats.3 <- my_get_player_stats_individual(career.stats.3)
stats.4 <- my_get_player_stats_individual(career.stats.4)
all.stats <- bind_rows(stats.1, stats.2, stats.3, stats.4)
```

Tidying career professional statistics

```
tidied.career <- all.stats %>%
  unnest() %>%
  select(-(games_played_playoffs:team_url),
         -(games_played_playoffs_:plus_minus_playoffs_),
         -(shot_handedness:age)) %>%
  filter(league_ %in% c("AHL","NHL")) %>%
  select(-(league:plus_minus))

pro.stats <- tidied.career %>%
  group_by(name, league_) %>%
  summarize(pro.gp = sum(games_played_),
            pro.gpg = sum(goals_) / pro.gp,
            pro.apg = sum(assists_) / pro.gp,
            pro.ppg = sum(points_) / pro.gp,
            pro.PIMpg = sum(penalty_minutes_) / pro.gp,
            pro.plusminus = sum(plus_minus_)) %>%
  filter(pro.gp > 20)

ahl.stats <- tidied.career %>%
  filter(league_ == "AHL") %>%
  group_by(name) %>%
  summarize(ahl.gp = sum(games_played_),
            ahl.gpg = sum(goals_) / ahl.gp,
            ahl.apg = sum(assists_) / ahl.gp,
            ahl.ppg = sum(points_) / ahl.gp,
            ahl.PIMpg = sum(penalty_minutes_) / ahl.gp,
            ahl.plusminus = sum(plus_minus_),
            ahl.seasons = n()) %>%
  filter(ahl.gp > 20)

nhl.stats <- tidied.career %>%
```

```
  filter(league_ == "NHL") %>%
  group_by(name) %>%
  summarize(nhl.gp = sum(games_played_),
            nhl.gpg = sum(goals_) / nhl.gp,
            nhl.apg = sum(assists_) / nhl.gp,
            nhl.ppg = sum(points_) / nhl.gp,
            nhl.PIMpg = sum(penalty_minutes_) / nhl.gp,
            nhl.plusminus = sum(plus_minus_),
            nhl.seasons = n()) %>%
  filter(nhl.gp > 20)


joined.stats <- full_join(ahl.stats,
                          nhl.stats) %>%
  replace(is.na(.),0) %>%
  mutate(pro.seasons = ahl.seasons + nhl.seasons)
```

```
## Joining, by = "name"
```

Accumulate NCAA career statistics

```
ncaa.careers <- ncaa.drafted.clean %>%
  group_by(name) %>%
  summarize(position = unique(position)[1],
            seasons = n(),
            ncaa.gp = sum(games_played),
            ncaa.gpg = sum(goals) / ncaa.gp,
            ncaa.apg = sum(assists) / ncaa.gp,
            ncaa.ppg = sum(points) / ncaa.gp,
            ncaa.PIMpg = sum(penalty_minutes) / ncaa.gp,
            ncaa.plusminus = sum(plus_minus)) %>%
  mutate(ncaa.seasons = factor(seasons, levels = 1:4),
         position = if_else(position %in% c("RW/C","LW/C","C/W","C/LW","C","F",
                                            "RW","LW","C/RW","LW/RW","RW/LW","W/C"),
```

```
                              "F",
                              "D")) %>%
  select(-seasons)
```

Model selection

```
regsubsets(gp.per.season ~ position + ncaa.seasons + ncaa.plusminus + ncaa.PIMpg + ncaa.ppg + ncaa.gp,
           data = full.data,
           method="exhaustive") %>%
  summary()
```

```
## Subset selection object
## Call: regsubsets.formula(gp.per.season ~ position + ncaa.seasons +
##     ncaa.plusminus + ncaa.PIMpg + ncaa.ppg + ncaa.gp, data = full.data,
##     method = "exhaustive")
## 8 Variables  (and intercept)
##                Forced in Forced out
## positionF          FALSE      FALSE
## ncaa.seasons2      FALSE      FALSE
## ncaa.seasons3      FALSE      FALSE
## ncaa.seasons4      FALSE      FALSE
## ncaa.plusminus     FALSE      FALSE
## ncaa.PIMpg         FALSE      FALSE
## ncaa.ppg           FALSE      FALSE
## ncaa.gp            FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          positionF ncaa.seasons2 ncaa.seasons3 ncaa.seasons4
## 1  ( 1 ) " "       " "           " "           " "
## 2  ( 1 ) "*"       " "           " "           " "
## 3  ( 1 ) "*"       "*"           " "           " "
## 4  ( 1 ) "*"       "*"           "*"           " "
## 5  ( 1 ) "*"       "*"           "*"           " "
```

```
## 6  ( 1 ) "*"       "*"           "*"           "*"
## 7  ( 1 ) "*"       "*"           "*"           "*"
## 8  ( 1 ) "*"       "*"           "*"           "*"
##          ncaa.plusminus ncaa.PIMpg ncaa.ppg ncaa.gp
## 1  ( 1 ) " "            " "         "*"       " "
## 2  ( 1 ) " "            " "         "*"       " "
## 3  ( 1 ) " "            " "         "*"       " "
## 4  ( 1 ) " "            " "         "*"       " "
## 5  ( 1 ) "*"            " "         "*"       " "
## 6  ( 1 ) " "            " "         "*"       "*"
## 7  ( 1 ) "*"            " "         "*"       "*"
## 8  ( 1 ) "*"            "*"         "*"       "*"
```

The six-variable model using `ncaa.seaasons`, `position`, `ncaa.gp`, and `ncaa.ppg` for predictors is chosen to be the best model. `ncaa.gp` is excluded from the actual fit because of high collinearity with `ncaa.seasons`.