Prediction Compilation

Kyle Joecken

This R Markdown file will create a table containing all our predicted data and store it in our PosgreSQL database.

Games Played

Recall that we decided upon the model

$$x \sim 24 + \frac{4}{9}y + \frac{3}{20}z,$$

where x is the number of predicted games played in the upcoming season, y is the number of games played in the previous season and z is the number of games played two seasons prior. Equivalently (and perhaps more legibly), we could say that

$$x \sim 73 - \frac{4}{9}\bar{y} - \frac{3}{20}\bar{z},$$

where $\bar{y} = 82 - y$ now represents games *missed* in the previous season (and $\bar{z}$ is similarly defined). We use the first, more computationally straightforward version to begin construction of our table. For skaters that did not play in 2012-2013, we replace their `NA` with the median value for the season, as the data are heavily skewed. We also must scale the season up, as a lockout meant only 48 games were played.

```
byskater <- reshape(skaterstats, timevar = "season",
                        idvar = "nhl_num", direction = "wide")
byskater <- subset(byskater, !is.na(games_played.2014))
byskater$games_played.2013[is.na(byskater$games_played.2013)] <-
      median(byskater$games_played.2013, na.rm = TRUE)
byskater$games_played.2013 <- 82/48 * byskater$games_played.2013
skatpred15 <- as.data.frame(cbind(byskater$nhl_num,
                                  round(I(24 + 4/9 * byskater$games_played.2014 +
                                          3/20 * byskater$games_played.2013))))
names(skatpred15) <- c("nhl_num", "games_played")
```

Time On Ice

Having modelled situational TOI per game, we can now project TOI by predicting TOI/game and multiplying by projected games.

Our models are:

$$x_{es} \sim 75 + \frac{7}{10}y_{es} + \frac{1}{5}z_{es}$$
$$x_{pp} \sim 9 + \frac{13}{20}y_{pp} + \frac{1}{5}z_{pp}$$
$$x_{sh} \sim 18 + \frac{13}{20}y_{sh} + \frac{3}{20}z_{sh}$$

Again, we choose to assign the median 2013 situational TOI/game to players that did not play in 2013, as the data are ever so slightly left skewed.

```
byskater$es_toi_pg.2013 <- byskater$es_toi.2013 / byskater$games_played.2013
byskater$es_toi_pg.2013[is.na(byskater$es_toi_pg.2013)] <-
      median(byskater$es_toi_pg.2013, na.rm = TRUE)
byskater$pp_toi_pg.2013 <- byskater$pp_toi.2013 / byskater$games_played.2013
byskater$pp_toi_pg.2013[is.na(byskater$pp_toi_pg.2013)] <-
      median(byskater$es_toi_pg.2013, na.rm = TRUE)
byskater$sh_toi_pg.2013 <- byskater$sh_toi.2013 / byskater$games_played.2013
byskater$sh_toi_pg.2013[is.na(byskater$sh_toi_pg.2013)] <-
      median(byskater$es_toi_pg.2013, na.rm = TRUE)
byskater$es_toi_pg.2014 <- byskater$es_toi.2014 / byskater$games_played.2014
byskater$pp_toi_pg.2014 <- byskater$pp_toi.2014 / byskater$games_played.2014
byskater$sh_toi_pg.2014 <- byskater$sh_toi.2014 / byskater$games_played.2014
skatpred15$es_toi <- round((75 + 7/10 * byskater$es_toi_pg.2014 +
                              1/5 * byskater$es_toi_pg.2013) *
                             skatpred15$games_played)
skatpred15$pp_toi <- round((9 + 13/20 * byskater$pp_toi_pg.2014 +
                              1/5 * byskater$pp_toi_pg.2013) *
                             skatpred15$games_played)
skatpred15$sh_toi <- round((18 + 13/20 * byskater$sh_toi_pg.2014 +
                              3/20 * byskater$sh_toi_pg.2013) *
                             skatpred15$games_played)
```

Upload

Once the data have been computed and compiled, we simply add the table to our existing database. As we may be doing this repeatedly, we'll check for and if necessary delete the table before writing.

```
if (dbExistsTable(conn, "skatpred15")) {
      dbRemoveTable(conn, "skatpred15")
}
dbWriteTable(conn, "skatpred15", skatpred15)
dbDisconnect(conn)
```