HTHSCI 1MO3 Final Project

Javid Husseynly

04/14/2021

Introduction

The deployment of NHL players has always been a highly-debated topic. Some coaches are accused of over-trusting experienced veterans, while some are targeted for giving young players too many chances.

Thanks to the internet, we have access to vast amounts of data relating to player deployment and performance from many years past. Today, we take a look at data from three different seasons, 1999-00, 2009-10 and 2019-20, to attempt to see how coaches have fared in the last twenty years.

Specifically, we will ask:

- Do coaches have unjust biases in player deployment?
- How do players fare given their deployment?
- Are coaches heding in the right direction and making improvements?

We get our data from hockey-reference.com, a publicly-available site that scrapes its data from official NHL game reports.

Data Wranging Plan

Iteration 1

Phase 1

- Read in .csv files
- Remove unnecessary columns
- Rename columns to be more clear
- Convert age to a factor column
- Separate 'Player' column
  - Remove player name
  - Send player id to its own 'id' column
- Join basic and advanced statistics dataset for 2019

Phase 2

```
library(tidyverse)
library(magrittr)
library(reshape2)
library(patchwork)
```

```
#Reading in .csv files
basic1999 <- read_csv("basic1999.csv", skip = 1)
basic2009 <- read_csv("basic2009.csv", skip = 1)
basic2019 <- read_csv("basic2019.csv", skip = 1)
advanced2019 <- read_csv("advanced2019.csv", skip = 1)
glimpse(basic1999)
```

```
## Rows: 838
## Columns: 23
## $ Rk     <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ~
## $ Player <chr> "Antti Aalto\\aaltoan01", "Bryan Adams\\adamsbr01", "Greg Adams~
## $ Age    <dbl> 24, 22, 36, 25, 20, 21, 28, 35, 26, 27, 24, 24, 29, 33, 28, 36,~
## $ Tm     <chr> "MDA", "ATL", "PHX", "TOR", "BUF", "BOS", "PHX", "CGY", "DAL", ~
## $ Pos    <chr> "C", "LW", "LW", "C", "RW", "D", "LW", "D", "D", "RW", "D", "C"~
## $ GP     <dbl> 63, 2, 69, 52, 65, 3, 82, 41, 4, 57, 59, 37, 82, 55, 24, 77, 9,~
## $ G      <dbl> 7, 0, 19, 5, 16, 0, 10, 4, 0, 21, 1, 10, 43, 2, 3, 20, 0, 12, 0~
## $ A      <dbl> 11, 0, 27, 8, 18, 0, 17, 6, 0, 38, 3, 18, 41, 6, 8, 16, 1, 18, ~
## $ PTS    <dbl> 18, 0, 46, 13, 34, 0, 27, 10, 0, 59, 4, 28, 84, 8, 11, 36, 1, 3~
## $ `+/-`  <dbl> -13, -1, -1, -7, -4, -3, -3, -3, 1, 11, -5, 5, 10, -3, -3, -20,~
## $ PIM    <dbl> 26, 0, 14, 39, 41, 0, 36, 12, 0, 28, 102, 20, 48, 4, 8, 30, 4, ~
## $ PS     <dbl> 0.7, -0.1, 4.5, 0.4, 3.7, -0.2, 1.8, 2.1, 0.2, 7.1, 0.7, 2.9, 1~
## $ EV     <dbl> 6, 0, 14, 5, 14, 0, 9, 2, 0, 15, 1, 7, 27, 1, 2, 12, 0, 12, 0, ~
## $ PP     <dbl> 1, 0, 5, 0, 2, 0, 1, 1, 0, 4, 0, 3, 11, 0, 1, 8, 0, 0, 0, 7, 1,~
## $ SH     <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 2, 0, 0, 5, 1, 0, 0, 0, 0, 0, 0, 1, ~
## $ GW     <dbl> 1, 0, 0, 0, 2, 0, 1, 1, 0, 0, 0, 1, 2, 0, 0, 3, 0, 2, 0, 4, 4, ~
## $ EV_1   <dbl> 11, 0, 20, 7, 12, 0, 17, 4, 0, 27, 3, 9, 32, 5, 4, 7, 1, 16, 0,~
## $ PP_1   <dbl> 0, 0, 7, 0, 6, 0, 0, 1, 0, 9, 0, 9, 8, 0, 3, 9, 0, 2, 0, 8, 0, ~
## $ SH_1   <dbl> 0, 0, 0, 1, 0, 0, 0, 1, 0, 2, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, ~
## $ S      <dbl> 102, 1, 129, 70, 128, 2, 107, 37, 6, 164, 24, 66, 260, 57, 31, ~
## $ `S%`   <dbl> 6.9, 0.0, 14.7, 7.1, 12.5, 0.0, 9.3, 10.8, 0.0, 12.8, 4.2, 15.2~
## $ TOI    <dbl> 830, 22, 1171, 644, 854, 57, 1035, 885, 44, 1069, 840, 797, 179~
## $ ATOI   <time> 13:11:00, 10:57:00, 16:58:00, 12:23:00, 13:09:00, 18:57:00, 12~
```

```
#Remove unnecessary columns
basic1999 %<>% select(Player,Age,GP,PTS,TOI)
basic2009 %<>% select(Player,Age,GP,PTS,TOI)
basic2019 %<>% select(Player,Age,GP,PTS,TOI)
advanced2019 %<>% select(Player,'CF%', 'oZS%')
```

```
#Rename columns to be more clear
basic1999 %<>%
  rename(
    age = Age,
    games_played = GP,
    points = PTS,
    time_on_ice = TOI
  )
```

```r
basic2009 %<>%
  rename(
    age = Age,
    games_played = GP,
    points = PTS,
    time_on_ice = TOI
  )

basic2019 %<>%
  rename(
    age = Age,
    games_played = GP,
    points = PTS,
    time_on_ice = TOI
  )

advanced2019 %<>%
  rename(
    'corsi%' = 'CF%',
    'offensive_zone_start_%' = 'oZS%'
  )
```

Results

```r
#Convert age to a factor column
basic1999 %<>% mutate(age = as_factor(age))
basic2009 %<>% mutate(age = as_factor(age))
basic2019 %<>% mutate(age = as_factor(age))
```

```r
#Separate 'Player' column
basic1999 %<>% separate(Player, sep="\\\\", into=c(NA, "id"))
basic1999 %$% table(id) %>% .[.>1]
```

```
## named integer(0)
```

- The uid for our tibbles can be **id**
- We can perform our join on the **id** column

```r
#Do the same for the other tibbles (output hidden to save space):
basic2009 %<>% separate(Player, sep="\\\\", into=c(NA, "id"))
basic2019 %<>% separate(Player, sep="\\\\", into=c(NA, "id"))
advanced2019 %<>% separate(Player, sep="\\\\", into=c(NA, "id"))

basic2009 %$% table(id) %>% .[.>1]
basic2019 %$% table(id) %>% .[.>1]
advanced2019 %$% table(id) %>% .[.>1]
```

```
#Join basic and advanced statistics dataset for 2019, checking dimensions before and after
basic2019 %>% dim_desc()
```

```
## [1] "[883 x 5]"
```

```
advanced2019 %>% dim_desc()
```

```
## [1] "[883 x 3]"
```

```
all2019 <- left_join(basic2019, advanced2019, by="id")
all2019 %>% dim_desc()
```

```
## [1] "[883 x 7]"
```

Iteration 2

Phase 1

- Check factor levels on 'age'
- Check for NA Values
- Check for strange #'s
- Create total points, TOI columns
- Group by age and create % of team points and % of league TOI columns

```
#Check factor levels on 'age'
basic1999 %>%
  pull(age) %>%
  levels()
```

```
##  [1] "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29" "30" "31" "32"
## [16] "33" "34" "35" "36" "37" "38" "39"
```

```
#Do the same for the other tables (output hidden to save space):
basic2009 %>%
  pull(age) %>%
  levels()

all2019 %>%
  pull(age) %>%
  levels()
```

```
#Check for NA Values
#Check for strange #'s
basic1999 %>%
  summary()
```

```
##       id                  age        games_played       points
##  Length:838         24     : 87   Min.   : 1.00   Min.   : 0.00
##  Class :character   23     : 77   1st Qu.:23.00   1st Qu.: 3.00
```

```
##  Mode  :character   26     : 68   Median :58.00   Median :13.00
##                     25     : 67   Mean   :49.28   Mean   :20.17
##                     22     : 64   3rd Qu.:74.00   3rd Qu.:31.00
##                     27     : 63   Max.   :83.00   Max.   :96.00
##                     (Other):412
##   time_on_ice
##  Min.   :   1.0
##  1st Qu.: 238.2
##  Median : 826.0
##  Mean   : 806.2
##  3rd Qu.:1266.0
##  Max.   :2389.0
##
```

```
#Output hidden to save space
basic2009 %>%
  summary()
```

- All numbers are reasonable
- No NA Values

```
all2019 %>%
  summary()
```

```
##       id                 age       games_played       points
##  Length:883         23     : 94   Min.   : 1.00   Min.   :  0.00
##  Class :character   25     : 90   1st Qu.:20.00   1st Qu.:  4.00
##  Mode  :character   24     : 78   Median :53.00   Median : 14.00
##                     26     : 78   Mean   :44.11   Mean   : 19.51
##                     27     : 75   3rd Qu.:66.00   3rd Qu.: 31.00
##                     29     : 66   Max.   :71.00   Max.   :110.00
##                     (Other):402
##   time_on_ice         corsi%       offensive_zone_start_%
##  Min.   :   1.0   Min.   :  7.70   Min.   :  0.00
##  1st Qu.: 233.0   1st Qu.: 46.50   1st Qu.: 46.02
##  Median : 782.0   Median : 49.70   Median : 50.25
##  Mean   : 727.9   Mean   : 48.92   Mean   : 49.63
##  3rd Qu.:1133.5   3rd Qu.: 52.30   3rd Qu.: 54.67
##  Max.   :1846.0   Max.   :100.00   Max.   :100.00
##                                    NA's   :1
```

However, in our joined 2019 data, we have one missing value in **offensive_zone_start%**. We will be grouping by age and averaging this value later, so single imputation by mean should suffice.

```
mean_val <- all2019 %>%
  group_by(age) %>%
  pull(`offensive_zone_start_%`) %>%
  mean(na.rm = TRUE)
all2019 %<>%
  mutate(`offensive_zone_start_%` = if_else(is.na(`offensive_zone_start_%`), !!mean_val, `offensive_zone
```

We can now consider our data tidy and get to work:

```
#Create total points, total TOI columns
basic1999 %<>%
  mutate(total_points = sum(points),
         total_toi = sum(time_on_ice))
basic2009 %<>%
  mutate(total_points = sum(points),
         total_toi = sum(time_on_ice))
all2019 %<>%
  mutate(total_points = sum(points),
         total_toi = sum(time_on_ice))
```

```
#Group by age and create % of total points and % of total TOI columns
basic1999 %<>%
  group_by(age) %<>%
  summarise('points%' = sum(points)/max(total_points) * 100,
            'toi%' = sum(time_on_ice)/max(total_toi) * 100,)

basic2009 %<>%
  group_by(age) %<>%
  summarise('points%' = sum(points)/max(total_points) * 100,
            'toi%' = sum(time_on_ice)/max(total_toi) * 100,)

all2019 %<>%
  group_by(age) %<>%
  summarise('points%' = sum(points)/max(total_points) * 100,
            'toi%' = sum(time_on_ice)/max(total_toi) * 100,
            'offensive_zone_start_%' = mean(`offensive_zone_start_%`),
            'corsi%' = mean(`corsi%`))
```

Our data is now complete and ready for plotting:

```
glimpse(all2019)
```

```
## Rows: 23
## Columns: 5
## $ age                      <fct> 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 2~
## $ `points%`                <dbl> 0.28448676, 1.04505341, 2.03785416, 4.8014398~
## $ `toi%`                   <dbl> 0.3257795, 0.9233531, 1.9996391, 5.0318467, 5~
## $ `offensive_zone_start_%` <dbl> 51.90000, 52.89091, 54.95769, 50.75873, 50.51~
## $ `corsi%`                 <dbl> 43.95000, 49.77273, 49.14615, 46.27460, 49.61~
```

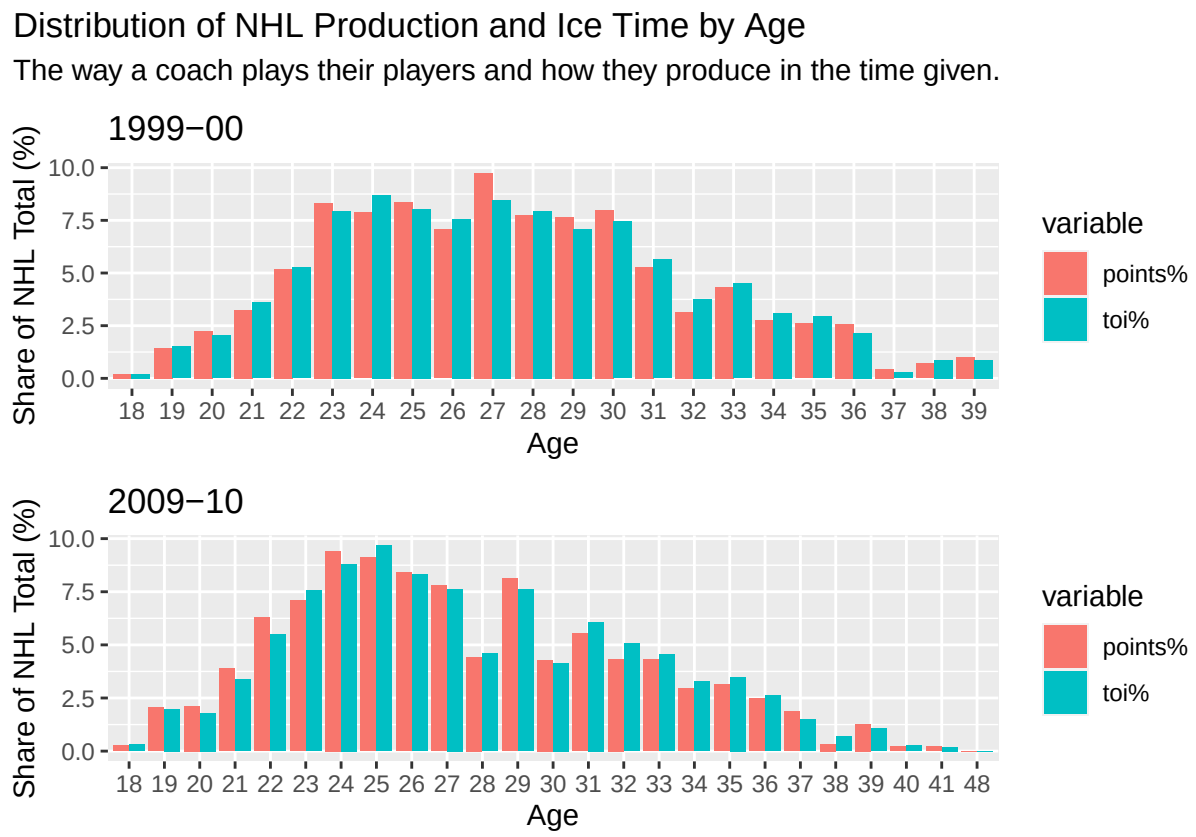
```
#output hidden to save space
glimpse(basic1999)
glimpse(basic2009)
```

```
basic1999_melt <- basic1999 %>%
  melt(id=c("age"))
basic2009_melt <- basic2009 %>%
  melt(id=c("age"))
```

```
dist1999 <- basic1999_melt %>%
  ggplot(aes(fill=variable,y=value,x=age)) +
  geom_bar(position='dodge',stat='identity') +
  labs(title='1999-00') +
  ylab("Share of NHL Total (%)") +
  xlab("Age")

dist2009 <- basic2009_melt %>%
  ggplot(aes(fill=variable,y=value,x=age)) +
  geom_bar(position='dodge',stat='identity') +
  labs(title='2009-10') +
  ylab("Share of NHL Total (%)") +
  xlab("Age")

dist1999 / dist2009 + plot_annotation(title="Distribution of NHL Production and Ice Time by Age", subti
```



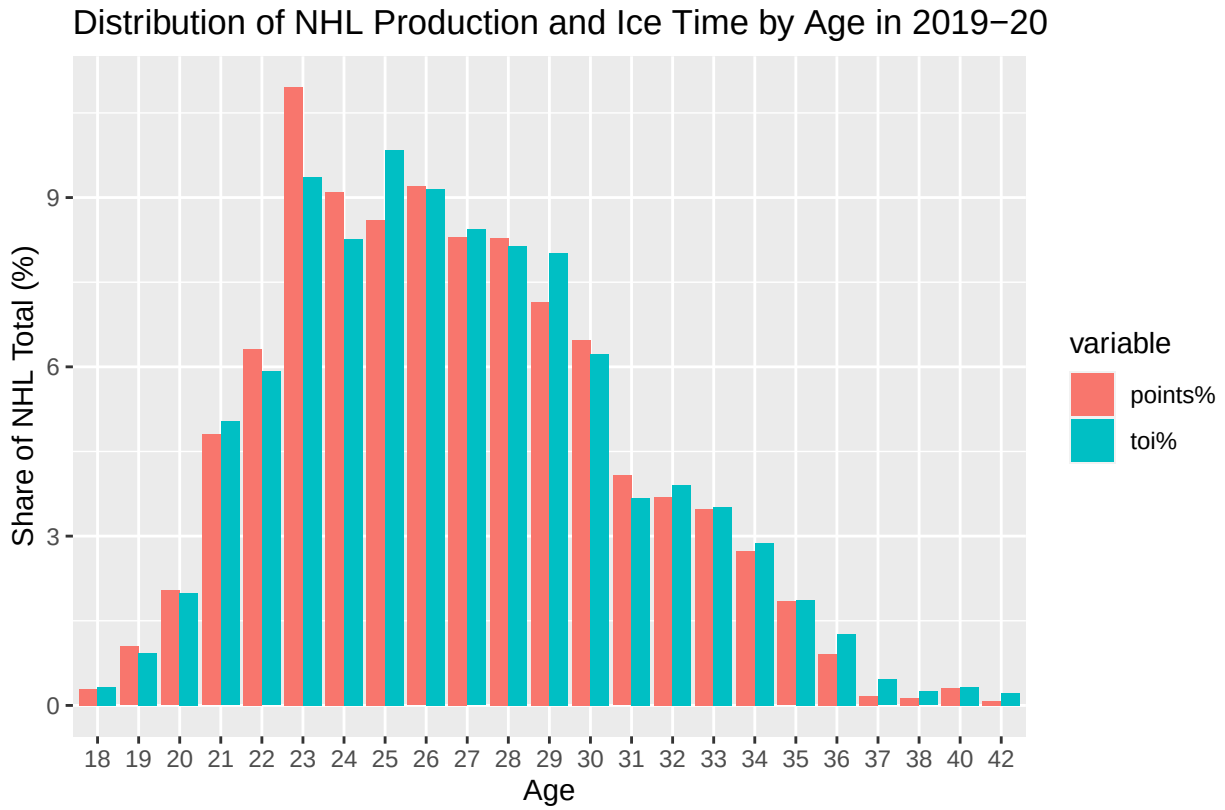
Here we compare the distribution of production (measured in points) and age in 1999 to 2009. In both years we see that players 31-35 were given a larger share of ice-time than their share of production. This points to common beliefs that coaches over-trust their veteran players.

In comparing the two seasons, we see minor changes. For example, coaches failed to adjust for the increase in talent from ages 19-22: in 2009 those younger players produced better than in 199, but we see players given less ice-time in proportion to their production. We can also see that in 2009, the gap between ice-time and production of veteran players aged 31-36 increased, indicating that coaches actually got worse in that regard.

```
all2019_melt <- all2019 %>%
  select(c("age","points%","toi%")) %>%
  melt(id=c("age"))

dist2019 <- all2019_melt %>%
  ggplot(aes(fill=variable,y=value,x=age)) +
  geom_bar(position='dodge',stat='identity') +
  labs(title='Distribution of NHL Production and Ice Time by Age in 2019-20', caption="Each age group's
  ylab("Share of NHL Total (%)") +
  xlab("Age")

dist2019
```

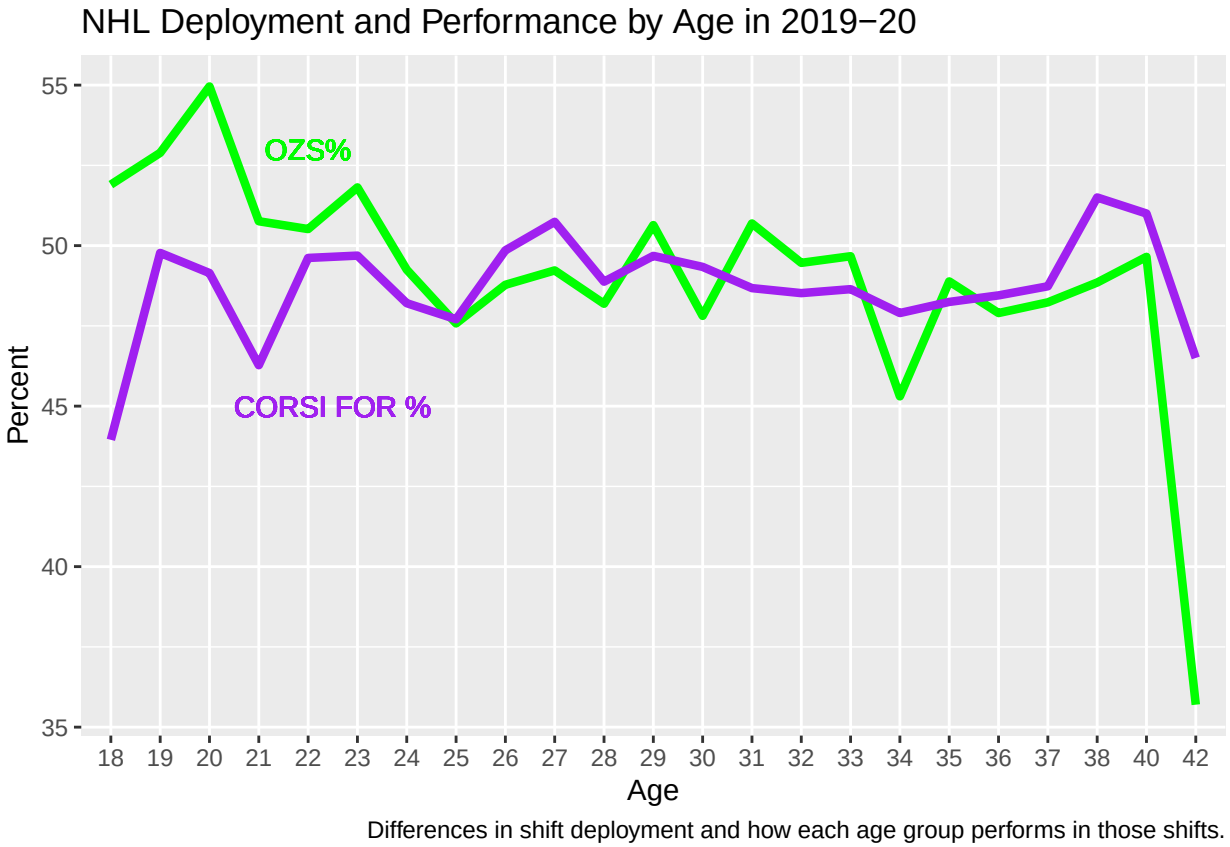


By 2019, it seems coaches have figured it out, for the most part. We still see a slight hint that they are still over-playing players age 32+, but the gap is not as well-pronounced.

One possible explanation for this is that coaches trust older players more defensively, which means they contribute in ways other than production. We can check how coaches have been deploying players with a statistic called offensive zone start % - the share of shifts the player starts in the offensive zone. Traditionally, a player with a lower OZS% would also have a lower CF%, because of the disadvantage of starting your shift in the defensive zone; and a low OZS% indicates that the coach trusts a player in his defensive end..

Next, we can measure how their team has performed with them on the ice with a relatively new statistic: corsi. Corsi attempts to measure possession - it simply represents shot attempts. If a player has a 55% corsi for %, that means their team had 55% of the shot attempts with them on the ice.

```
all2019 %>%
  ggplot(aes(x=age)) +
  geom_line(aes(y=`offensive_zone_start_%`), group=1, color='green', size=1.5) +
  geom_line(aes(y=`corsi%`), group=1, color='purple', size=1.5) +
  ylab("Percent") +
  xlab("Age") +
  labs(title="NHL Deployment and Performance by Age in 2019-20", caption="Differences in shift deployme
  geom_text(aes(x=5, y=53, label="OZS%"), color='green') +
  geom_text(aes(x=5.5, y=45, label="CORSI FOR %"), color='purple')
```



This graph shows us there may be merit to that explanation. Players 23 and under are getting significantly more shifts that start in the offensive zone, but their corsi for % does not reflect the advantage they are getting. Similarly, players 33+ see a decrease in offensive zone shift starts, yet they maintain steady possession.

Conclusion

The conclusion you reach depend on how you measure a player's success. If you believe that a player's most important contribution is his production, then our data would indicate that coaches have long-held a bias towards older players. They get more ice time than they give points, and that has been true going back to 1999, showing very little adjustment from coaches. Younger players have made more of their opportunities, producing at a higher clip than the ice time they receive, and older players have continued to get trust from their coach even when their production dips.

However, if you look at it through the scope of the team, you might value possession more. In that case, it is

shown that older players are given tougher shifts, starting in the defensive zone more often. Regardless, their team puts up equal, if not better, possession stats than the younger players getting fresh offensive starts.

A more definitive conclusion demands better measures of success. This could be viable with something like xGF% - a stat similar to CF% but with the improvement that each shot attempt is given more weight if it has a better chance of resulting in a goal. However, that is beyond the scope of this report.

References

Hockey-Reference (click for link.)