**Analysis for Code Samples**

April 25, 2021

**Table of Contents**

[0 - Overview](#_wuy5dw9xn3wr) 1

[1 - The Relative Age Effect (relative\_age\_effect.py)](#_9l01cuzb1awu) 2

[2 - NHL Salary Prediction (salary\_prediction.py)](#_adxnizoi8jor) 5

[3 - KR Fantasy Hockey Goalie Pairings Helper (schedule.py)](#_5zwo8vsd2762) 10

# 0 - Overview

This repository contains 5 files written in Python code:

1. relative\_age\_effect.py — analyzes the relevance of the relative age effect among NHL players.
2. salary\_prediction.py — performs basic modelling techniques to predict player salaries given the statistics of NHL players from 2008-2019.
3. functions.py — includes custom functions used in the previous 2 files.
4. salary\_prediction\_old.py — the original script for salary\_prediction.py. I only included this file to show how my coding abilities have improved since last year. *Do not try to run it - it won’t compile.*
5. schedule.py — a script to identify how many times pairs of NHL teams play on different days to help goalie selection in fantasy drafts.

This repository also contains 2 additional files:

1. Read First: Analysis for Code Samples — this file you are reading
2. Non Overlapping Games for 2 NHL Teams (2020-2021) — the output for schedule.py

This code was initially developed in 2020 as “side projects” to help me develop my Python skills, particularly my familiarity with Pandas and other libraries.

After my internship as an actuarial co-op on the data science team at Aviva, my coding skills tremendously improved. This allowed me to greatly enhance the readability and efficiency of my code for these side projects and present them to you here.

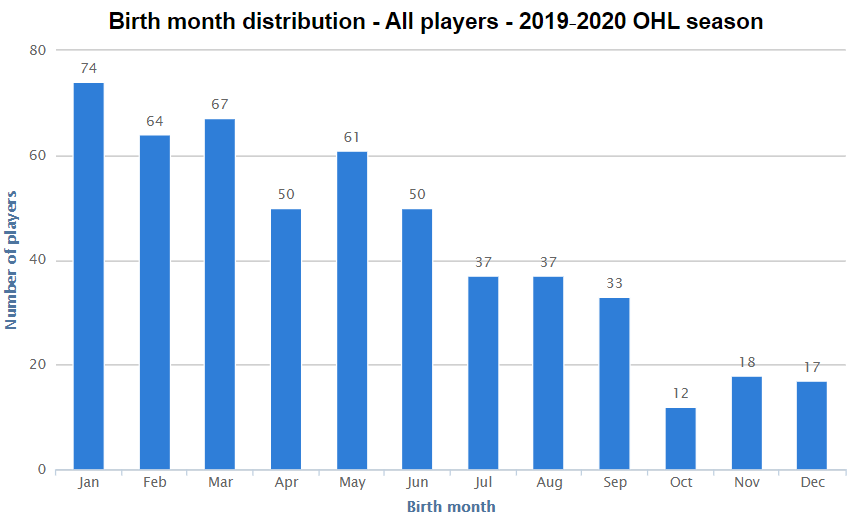
# 

# 1 - The Relative Age Effect (relative\_age\_effect.py)

**Introduction:**

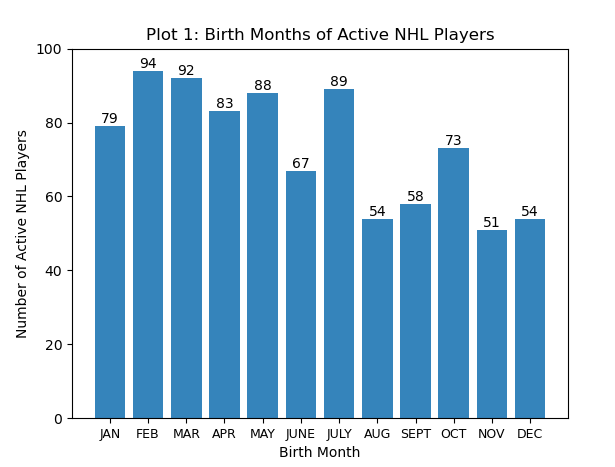
The Relative Age Effect (RAE) describes the “the phenomenon by which children born early in their year of birth perform more highly than children born later in the same cohort” (Frontiers in Psychology, 2018). Studies have shown that RAE has an impact on junior hockey players. That is, young players born in the first quarter of the year (January to March) are more likely to perform better than those born later because they are more emotionally and physically developed. This observation has been famously captured in Malcolm Gladwell’s book, *Outliers.*

The figure below, created by QuantHockey, provides evidence for this theory as it shows about 40% of all players in the 2019-2020 OHL season were born in the first quarter of the year.



**Do birth months impact your ability to make the NHL?**

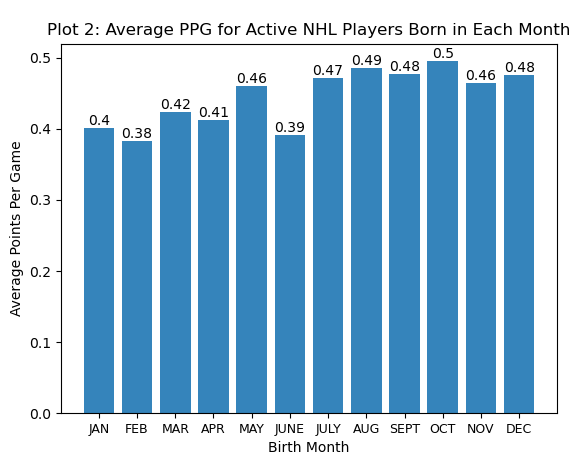
I tried answering the above question by creating a bar chart showing the birth months of all active NHL players. Below is my recreation of the QuantHockey chart but with NHL players.



The bar chart is slightly skewed to the right since there are more active players in the NHL born in the first half of the year. That said, the data is more evenly distributed than the OHL players in Figure 1.1. By inspection, one can notice that the proportion of players born in Oct/Nov/Dec has drastically increased from the OHL.

**Do birth months impact your performance in the NHL?**

I also wanted to investigate whether birth months impact performance in the NHL. For simplicity, performance in this case is measured by points per game.



As you can see, the trend is relatively flat. The later months tend to have slightly higher PPGs but this can be due to a smaller sample size. Therefore, I can’t conclude that NHL players score more points if they are born in the first quarter.

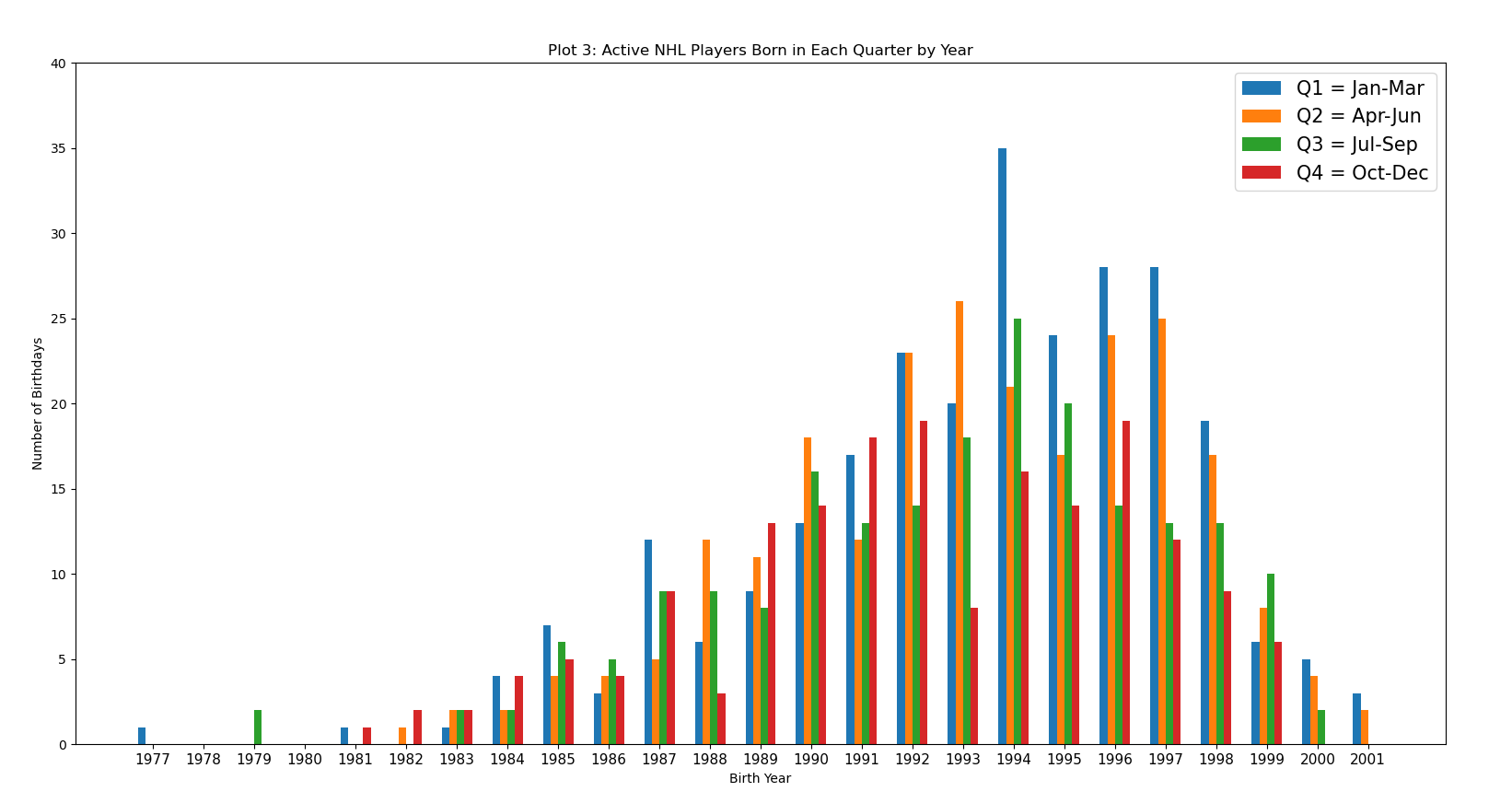
**Comparing within the same birth year:**

Recall that the definition of the Relative Age Effect focuses on the performance between two players in the *same cohort*.

Therefore, we shouldn’t be comparing the birth months in the NHL as a whole.

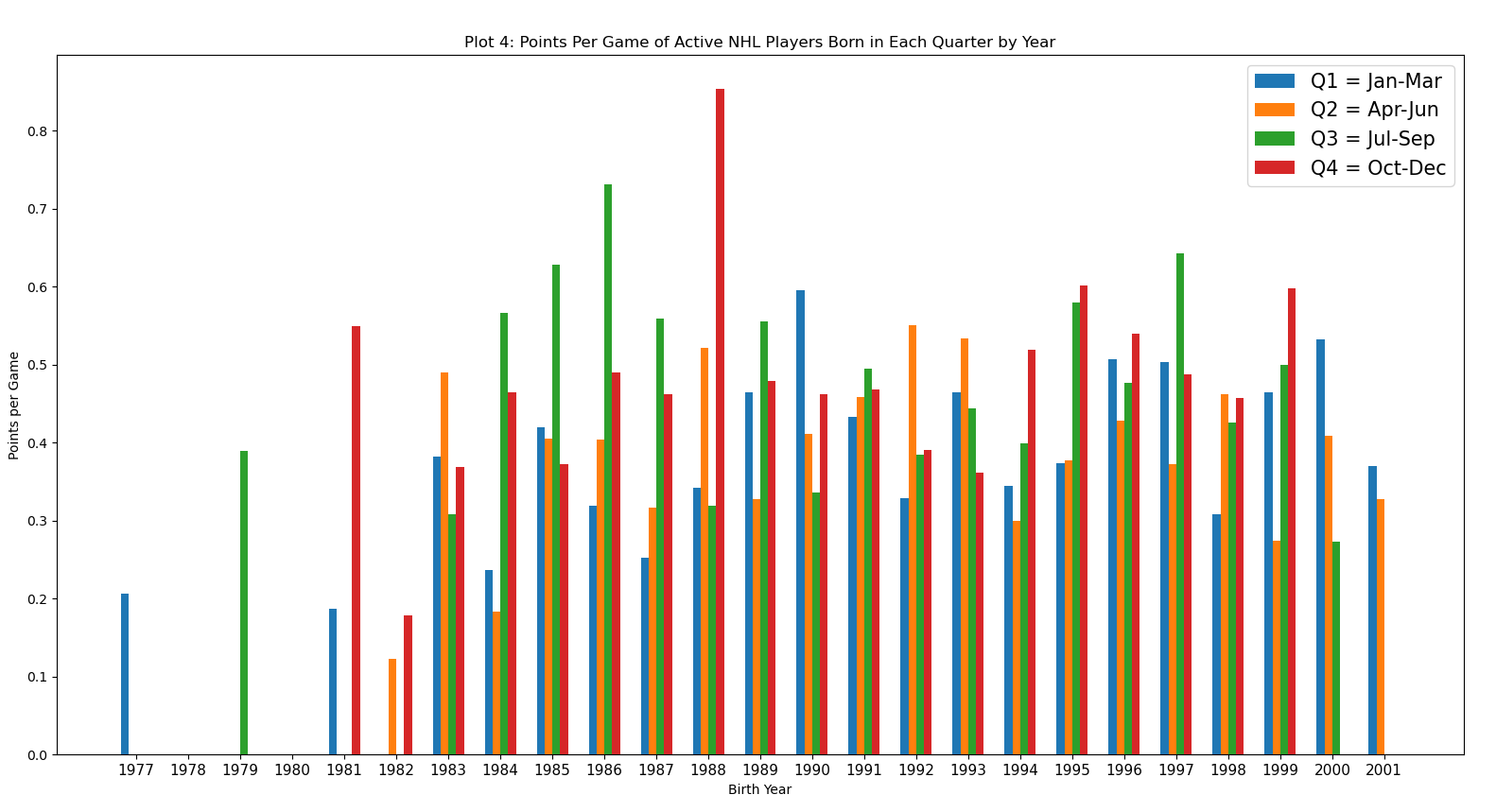
Instead, we should be comparing the birth months in the NHL *by year of birth*. We know that the Relative Age Effect is not relevant for a 27 year old born in January and a 20 year old born in December, but the above figures do not clearly make that distinction. As a result, I have remodelled the NHL data to create two additional figures.

By visualizing the data with the grouped bar chart below, it is easier to see the impact of the Relative Age Effect on all active NHL players and for *how long*.



For all but one year between 1994 and 2001, there are more players born in Q1 than in any other quarter. Moreover, in the same interval, there are the least amount of players born in Q4 in every year except for 1996.

For players born earlier than 1994, however, the data shows a different story. There are only 3 birth years out of 17 in which Q1 is the distinct leader for NHL players.



When it comes to PPG by quarter, the figure above shows that there is still no real trend among birth quarters, even for the youngest players.

**Small caveat: draft cutoff**

It’s important to note that the NHL restricts entry into the draft for players who turn 18 years old by September 15th. This means that all 18-year-olds who are drafted in the NHL are born in the first three quarters of the year. Therefore, due to the nature of the rules, players born earlier in the same year automatically have an advantage of making it into the NHL.

**References:**

<https://www.quanthockey.com/>

<https://www.frontiersin.org/articles/10.3389/fpsyg.2018.01091/full>

# 2 - NHL Salary Prediction (salary\_prediction.py)

**Introduction:**

The inspiration for this script came from my grade 12 statistics course. We did a project where we used points per game to predict the salaries of basketball players. I wanted to expand on this project and apply it to a hockey context.

**Initial methodology:**

* Web scrape for the salaries of all NHL players from online sites
* Use a variety of statistical categories from the 2018-2019 season to build a linear regression model to predict salaries
  + Split data by 80/20 for training/testing
* Measure the accuracy of the model with R2 and mean absolute error (MAE)

I won’t elaborate on the code specifics - you can read the code for that.

**Possible features for model:**

['Signed Age', 'GP', 'G', 'A', 'P', '+/-', 'PIM', 'P/GP', 'EVG', 'EVP', 'PPG', 'PPP', 'SHG', 'SHP', 'OTG', 'GWG', 'S', 'S%', 'TOI/GP', 'FOW%']

**Initial model results (your results may vary due to random sampling):**

|  |  |  |
| --- | --- | --- |
| **Features** | **Mean Absolute Error** | **R-Squared** |
| ['G', 'A', 'TOI/GP', '+/-', 'PPP', 'GP'] | 1.21M  (on avg, off by $1.21M) | 0.877  (max value 1) |

Overall, I felt that the greatest area for improvement was reducing the MAE to a number below $1M.

**Changes to initial methodology:**

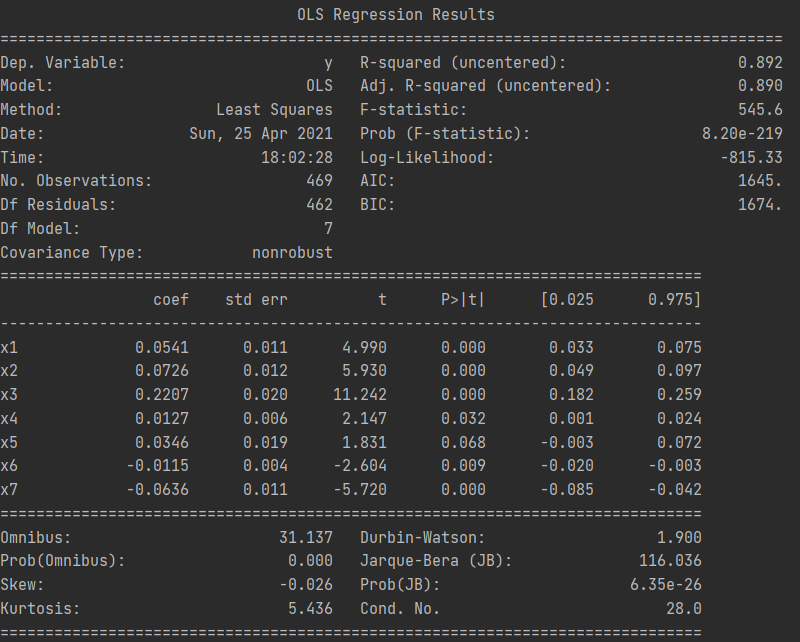
* Use the stats from the player’s “signing” year rather than the 2018-2019 season
  + By only using the 2018-2019 stats, the model would be quite inaccurate for players like Jeff Skinner who have degraded since their big contract signing.
  + The one caveat is that not all seasons have been 82 games long - so for simplicity, we will consider the “signing year” as the most recent 82 game season.
* Include “signed age” as a feature for the model
  + Players should be paid less when they sign as an older player (ie. Thornton/Spezza/Simmonds for the Leafs).

**Second model results:**

Overall, the results of the model were significantly better!

|  |  |  |
| --- | --- | --- |
| **Features** | **Mean Absolute Error** | **R-Squared** |
| ['G', 'A', 'TOI/GP', '+/-', 'PPP', 'GP', 'Signed Age'] | 0.976M  (0.23M decrease) | 0.892  (0.015 increase) |

Below is an output of the regression results by statsmodels (Python library).



G

A

TOI/GP

+/-

PPP

GP

Signed Age

The main item that I want to comment on is the green box showing the coefficients and various statistics for every model feature.

* The “coef” column shows the change in salary for every unit increase for every feature.
  + For example, for every minute of ice time per game, the model suggests that a player will earn $0.2207M more.
  + In my opinion, it is expected that a player should get paid more if they score more goals/assists/ppp and get greater ice time/plus-minus.
  + When it comes to GP and Signed Age, the coefficients are actually negative. I can rationalize why salaries may decrease as signing age increases — most players sign their “big contract” in their early-mid twenties. Meanwhile, old players (Spezza/Thornton/Simmonds) are more likely to sign shorter term and smaller contracts when they are past prime. It’s not intuitive why GP would have a negative coefficient. This could be a result of outliers in the data or a weak trend that players who get paid more tend to get injured more (maybe because of more ice time?).
* The [0.025, 0.975] columns show the endpoints for the 95% confidence interval for the coefficients. In brief, the 95% confidence interval says that “between these two endpoints, there is a 95% that the \*true\* coefficient for this feature is within this range”.
  + When I see the confidence interval, the main thing I am looking for is if the interval contains ZERO.
  + If the confidence interval contains zero, there is a high probability that the impact of the feature is very minimal on salary.
  + **PPP** is the only feature with a confidence interval that includes 0, and **+/-** and **GP** each have an endpoint extremely close to 0.
    - This suggests that PPP, +/-, and GP may not be the best features to include in a model. Using my hockey knowledge, this intuitively makes sense since PPP are in general pretty rare, +/- is strongly dependent on the team’s performance and not the individual, and playing games won’t be a huge difference maker in salary once a player reaches a certain level.

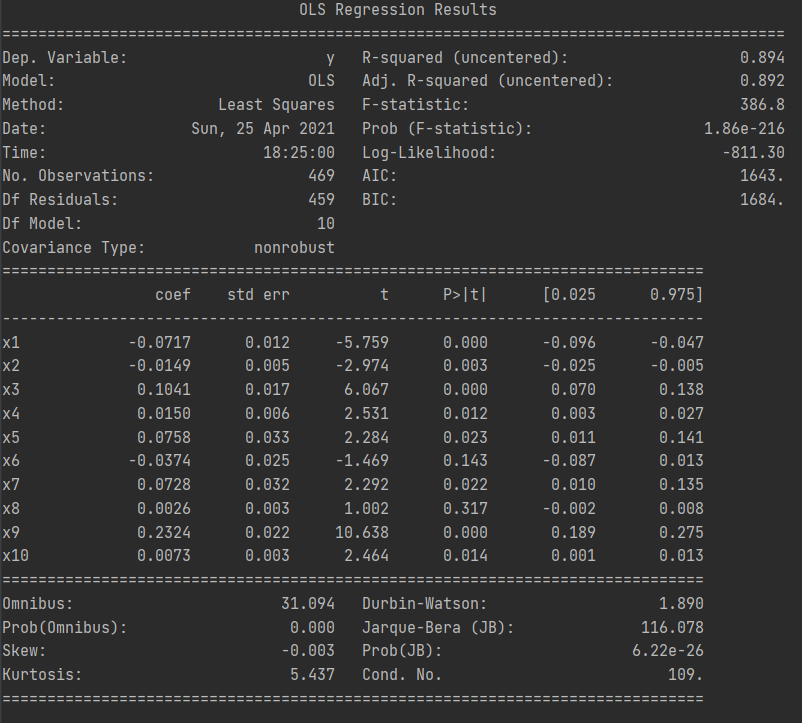
**Best possible model:**

For fun, I wanted to see which combination of features would result in a linear regression model with the lowest possible MAE.

The methodology here would be to loop through every combination of the possible features and record their MAE in a dictionary. Then I took the minimum MAE and its key to reveal the best possible combination of features.

**Best possible model results:**

|  |  |  |
| --- | --- | --- |
| **Features** | **Mean Absolute Error** | **R-Squared** |
| ['Signed Age', 'GP', 'A', '+/-', 'EVG', 'EVP', 'PPG', 'S', 'TOI/GP', 'FOW%'] | 0.961M | 0.894 |



Signed Age

GP

A

+/-

EVG

EVP

PPG

S

TOI/GP

FOW%

* Note that the best possible model’s MAE is still 0.961M which is not far off from the MAE for Model #2 above.
* There seems to be a good amount of over fitting with this model as well. +/-, EVP, S, and FOW% are all looking like very dodgy features due to their 95% confidence interval containing or being close to 0.

While I do call this the “best possible model”, I understand that this model does not include categorical/qualitative features (ie. position, team) nor interactions (ie. being a forward and having a high shooting percentage).

In conclusion, even though the best possible model has the lowest MAE, I would prefer a model in which its coefficients have a higher guaranteed impact on salary (such as model #2 but without PPP, +/-, and GP).

# 3 - KR Fantasy Hockey Goalie Pairings Helper (schedule.py)

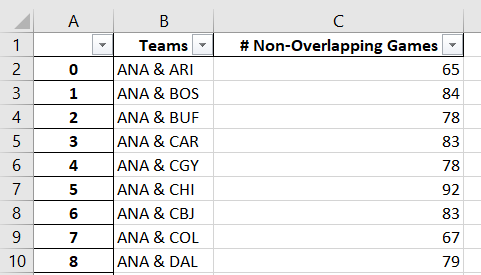
**Introduction:**

For the KR fantasy pool this year, we are only allowed to have 1 starting goalie each night. However, to win as many games as possible and have some insurance for injuries, most participants decided to draft two goalies.

The worst feeling for a fantasy hockey manager is having to pick the starting goalie when both goalies are playing on a given night. In this situation, you run the risk of starting a losing goalie while the other wins, and you don’t maximize the number of games that each goalie can play.

This led me to create schedule.py which counts the number of non-overlapping games for every pair of NHL teams.

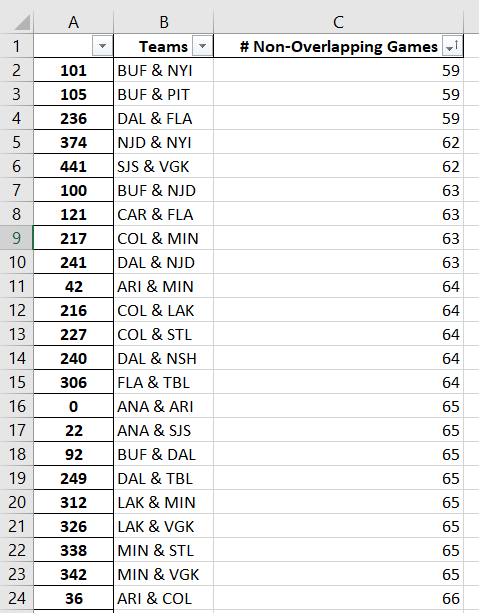
**Output:** A list of all 465 pairs of NHL teams and their number of non-overlapping games

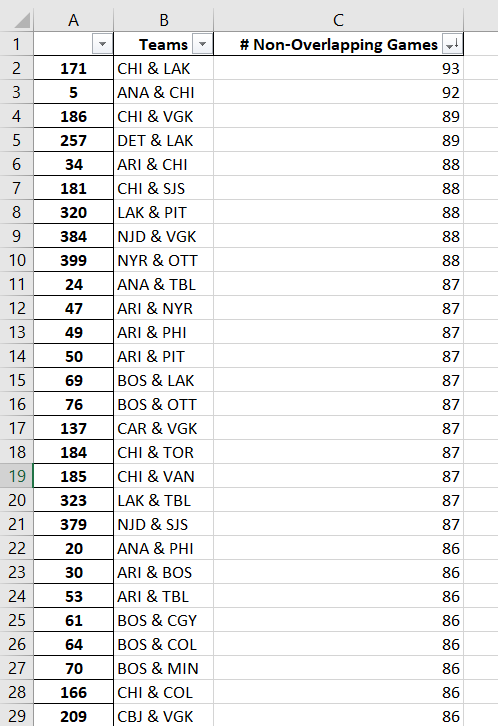


Note that the average number of non-overlapping games is 77.

The minimum is 59 and maximum is 93.

**Insights from this spreadsheet:**

By looking at the pairs with the fewest number of overlapping games, you’ll find some goalies that you shouldn’t pick together. The pairs at the bottom of the list shouldn’t come as a surprise since most of the teams are in the same division. But there are some exceptions such as BUF & DAL at 65.



This spreadsheet is only a “helper” and not a “cheat sheet” because the pairs with the highest number of overlapping games obviously don’t indicate the best goalie pairings for a variety of reasons:

1) One or two teams in the pair might just be a bad team overall and not win many games (ie. CHI & LAK)

2) There may not be a clear cut starting goalie for the team. (Korpisalo vs. Merzlikens, Fleury vs. Lehner, Grubauer vs. Francouz, Varlamov vs. Sorokin were all pairs where I couldn’t see a definite starting goalie at the start of season.)

Overall, this spreadsheet is better off highlighting combinations of goalies that you shouldn’t pick rather than ones you should pick.

In the end, I went with Korpisalo and Lehner - since the season projected CBJ and VGK to have 88 non overlapping games (now 86 due to scheduling changes).

Were they good picks?

*Not at all.*

Columbus is dead last in their division and Lehner was injured for quite a few weeks and became a definite backup to Fleury.

But at least it was a fun little project.