

# GFS: Goaltender Fatigue Score

Raymond Romaniuk<sup>1</sup>

<sup>1</sup>Brock University, PhD Intelligent Systems & Data Science Student  
Email: rromaniuk31@gmail.com

August 22, 2024

## 1 Introduction

When listening to fans and pundits compare different sports, one topic that inevitably comes up is which sport is the hardest to play and subsequently which position is the most difficult. Reading articles online there's a couple of positions that show up in almost every list like an NFL quarterback, or an MLB catcher. However, in my biased opinion from years of playing and coaching the position, I believe that the most difficult and consequential position in sports is that of an NHL goaltender.

Aside from the physical requirements of the position, like great hand-eye coordination, flexibility and foot speed, goaltenders also need to be strong mentally. These factors contribute to making goaltending the most volatile position in the sport. Goalies can go from an all-star to bottom of the league from season to season with no obvious reason to explain the regression. This is part of the reason it is imperative that teams find the right guy for the job because you can't necessarily win a game with good goaltending, but you can certainly lose it with bad goaltending.

For these reasons, and many more, one of my main personal goals is to "solve goaltending". Is that possible? As a goaltender, turned data nerd I'd love to think so. However, it is surely not, in the sense that there will never be an end to end solution ensuring no goal is ever scored. Ultimately, I feel that goaltending is an under researched area due to its volatility and perceived difficulty. I see this as an opportunity to expose a potential goaltending market inefficiency.

The first step in uncovering this inefficiency will be to create a measure to gauge a goaltenders fatigue levels, called the Goaltender Fatigue Score (GFS), and give teams another variable when trying to optimize goaltender deployment.



Figure 1: Connor Ingram in his sweet Bauer Hyperlite DigiPrint "Desert Nights" setup and Karel Vejmelka in his beautiful Brian's Optik 3 gear.

## 2 Data

The most important part of any good data analysis is the data, poor data produces poor results. This project contains data coming from multiple sources, which are outlined below, along with the data engineering and categorical encoding that was completed during the data preprocessing.

### 2.1 Data Sources

The first and main source of data used for this project is the yearly shot data provided by MoneyPuck [2]. The data spans ten seasons from 2014-15 to 2023-24 and contains information about each missed shot, shot on goal and goal scored. Each season contains more than 105,000 shots (except 2020-21 due to COVID), before any data engineering is done to clean it up for analysis, so overall there are approximately 1.1 million observations. This dataset contains information such as where the shot occurred on the ice (coordinates, angle, distance), time elapsed since previous event, type of shot, shooters name, goalies name and shift length for offensive/defensive players.

To supplement the data from MoneyPuck I also leveraged the public NHL API to access the dates for each game in the MoneyPuck dataset to make it possible to average specific statistics over time. In the future there is an opportunity to utilize this data source more fully, however the MoneyPuck dataset contains plenty of columns that are pre-calculated, so the time savings outweigh the potential for a few extra data points from the NHL API.

To make this project possible a dataset I needed that was not readily available was the distances between arenas, including those used in the NHL's Global Series. This meant I needed to spend a fair amount of time collecting this information and ensuring that it could be easily joined with all of the other data at hand. Having this information will help shed light on how travel may impact goaltender fatigue.

### 2.2 Data Engineering

The dataset is quite clean relative to the usual state of raw data, though there are still a few things that had to be done to optimize it for machine learning. The first exercise I explored was developing a model to predict the expected goals (xGoal) of each shot. Since this project revolves around goaltending, missed shots are not within the scope of this project, so I dropped all shots that missed the net. I also accounted for the change in team abbreviation (ex. N.J to NJD), since it wasn't consistent across years and would lead to inconsistent categorical data encoding. There are a couple players with identical names, so I distinguished Sebastian Aho (CAR) from Sebastian Aho (NYI) and Matt Murray (TOR) from Matt Murray (DAL (now NSH)). Finally, there were 13 variables that did not appear for all years and some rows with missing data, so for simplicity I dropped them, since after this step I still had approximately 800,000 data points.

### 2.3 Categorical Encoding

Since the data has a number of categorical columns containing valuable information it's necessary to encode them, so that I can leverage information like shot type, goalie and shooter. The first naive attempt I made to encode these categorical columns was one-hot encoding, however it was quickly obvious that this would not work. One-hot encoding entails creating a column for every possible category in the categorical column, so if the categorical column contains five categories then there will be five one-hot encoded columns with one of these five columns containing a 1 indicating that category is represented in that row (the remaining four columns would contain 0's for that row). Since one of the categorical columns in the data is *shooterName* and in the past ten years there have been 2000 shooters, one-hot encoding led to an additional 2000 columns added to the dataset. This caused the data to expand to over 6GB, which couldn't be handled efficiently locally in the Jupyter Notebook I was working in, due to its memory limit.

To solve the issue I decided to use target encoding on the categorical variables. Target encoding entails replacing each category in the categorical column with a number derived from the target, goals. The major added benefit of using target encoding on a column with many categories is that it does not increase the size of the dataset, target encoding columns does not lead to many more columns like with one-hot encoding. Another feature I leveraged in the target encoding is smoothing.

Imagine 2024 sixth overall pick Tij Iginla begins his career in Utah with 15 goals in his first five games. Is it reasonable to believe that this production will sustain over time and 200 games into his career he will continue to score at this pace? Obviously not. Using smoothing on the encoded column accounts for this by ensuring categories with a small sample size have their encoded value adjusted toward the overall mean. This helps minimize potential outliers occurring in data points of players that have played few games.

### 3 Analysis

Now that the dataset has been cleaned and prepared for analysis, the first step was to build my own expected goals (xGoal) model, since MoneyPuck’s model accounts for missed shots. After the model has been trained and used to predict the expected goals of each shot, I will use these predictions to build out the Goaltender Fatigue Score (GFS).

#### 3.1 Expected Goals (xGoal) Model

The first thing I needed was a viable model to call a baseline and compare later models to. The model I decided to use as a baseline was a simple logistic regression, since it is one of the simpler models available and if a more computationally intensive model performs similarly then it may be advantageous to simply use logistic regression and save resources. After training a logistic regression model on the first nine seasons in the dataset I made predictions on this past season and, using mean squared error for the loss, obtained a loss of 0.0855. The second model I tested also leveraged logistic regression by ensembling the nine models trained on the nine years of training data and weighting the predictions made by each such that more recent years had more say in the final predictions. Ensembling the past 2,3,5 and 9 years each provided a slight improvement on the baseline with the best ensembled model using the past two years of data.

Moving away from logistic regression I tested the performance of Adaptive Boosting (AdaBoost), but quickly noticed the performance was significantly worse than the baseline. I then pivoted to what became the final model eXtreme Gradient Boosting (XGBoost).

To train the ideal XGBoost model I first performed k-fold cross-validation to select the two hyperparameters, learning rate and max depth, while ensuring the temporal structure of the data remained intact. This resulted in selecting a learning rate of 0.1 and max depth of 7. This model outperformed the other models, in addition to MoneyPuck on the testing data. Table 1 shows each model and their respective loss.

Table 1: Model Performance on Testing Data (lower loss is better)

| Model                         | Loss          |
|-------------------------------|---------------|
| <b>XGBoost</b>                | <b>0.0712</b> |
| MoneyPuck                     | 0.0784        |
| Ensembled Logistic Regression | 0.0849        |
| Logistic Regression           | 0.0855        |
| AdaBoost                      | 0.1390        |

#### 3.2 Goaltender Fatigue Score (GFS)

##### 3.2.1 Background

From 1995 to 2010 a total of 50 goalies played in more than 70 games in a single season, since then only eight goalies have done so, with the last goalie to do it being Cam Talbot in 2016-17 for the Edmonton Oilers. Over the years teams have started to deploy their star goaltenders less and less evidenced by Linus Ullmark winning the the Vezina in 2022-23 while only appearing in 49 games. Limiting goaltender appearances isn’t necessarily a bad idea for a team with a competent backup, as it ensures starters are not overworked and have energy left in the tank for a long playoff run. However in some cases a lack of starts for a starter may be the difference between a potential Stanley Cup run or completely missing the playoffs if a team is fighting for a wild card spot. So, in

the interest of keeping goaltenders healthy and ready for a playoff run, while also ensuring points aren't left on the board, I aim to devise a statistic to rate a goaltenders perceived fatigue levels to allow teams an extra data point in deciding whether a starter can handle extra starts or not.

### 3.2.2 Creating GFS

My metric, GFS, utilizes seven statistics that are derived from a goaltenders previous 28 days of work. These statistics are xGoals (xG) from my XGBoost model, games played (GP), games team has played (TGP), shots faced (SOG), shots missed (SM), distance travelled (DT) and number of rest days (RD). Using these statistics GFS aims to account for the action a goaltender has faced in the games he's played, while also accounting for travel between games. Long travel can negatively impact a bodies ability to recover, as was reported when John Klingberg flew with the Maple Leafs to Sweden this year and was placed on the LTIR shortly afterward with Sheldon Keefe saying the flight "didn't do him any favours" [1].

The proposed GFS is computed using the following formula:

$$GFS = 0.1 * \left( \frac{xG}{TGP} \cdot \frac{0.8 \cdot GP \cdot SOG}{TGP} + \frac{0.2 \cdot GP \cdot SM}{TGP} + \frac{DT}{RD} \right)$$

from the formula you can see that the aim is to use xGoals as a rate that helps indicate how difficult the shots faced by a goaltender were, more difficult shots naturally lead to a higher level of strain and greater fatigue afterward. The distance travelled by the goaltender is divided by the number of "days off" a goaltender has to recuperate from long travel. Figure 2 depicts the distribution of GFS over the past 10 seasons and Table 2 shows the top ten highest GFS's over the past ten seasons.

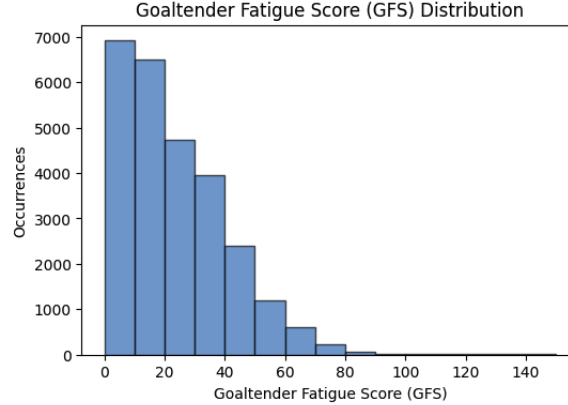


Figure 2: Distribution of Goaltender Fatigue Score (GFS) over the past 10 seasons.

Table 2: Top 10 Goaltender Fatigue Scores

|    | Name             | Date         | GFS    |
|----|------------------|--------------|--------|
| 1  | Sergei Bobrovsky | May 22, 2023 | 124.13 |
| 2  | Sergei Bobrovsky | May 24, 2023 | 119.87 |
| 3  | Igor Shesterkin  | Jun 1, 2022  | 115.00 |
| 4  | Mike Smith       | Dec 23, 2016 | 114.16 |
| 5  | Sergei Bobrovsky | May 20, 2023 | 110.70 |
| 6  | Mike Smith       | Dec 21, 2016 | 109.32 |
| 7  | Igor Shesterkin  | Jun 9, 2022  | 108.09 |
| 8  | Igor Shesterkin  | Jun 7, 2022  | 107.89 |
| 9  | Igor Shesterkin  | Jun 11, 2022 | 105.19 |
| 10 | Igor Shesterkin  | May 30, 2022 | 104.08 |

Looking at this list it seems to be dominated by Sergei Bobrovsky and Igor Shesterkin during their spectacular playoff runs. In Bobrovsky's case on top of playing every game for Florida in

the 28 day window, he faced more than 35 shots in 7 of Florida's 11 games, including a 63 save 4OT game that contained over six xGoals. Shesterkin on the other hand had 12 games with save percentages above .925 while averaging 41 shots against per game. Naturally, these performances are almost impossible in the regular season, since starters will not play every game and those games can't exceed 65 minutes limiting the shots a goaltender can face.

### 3.3 xGoal & GFS

Now that I've calculated GFS I can try inserting it into the training of the XGBoost xGoal model to see if it can improve the models performance. Following the same k-fold cross-validation procedure provided a new model with negligible change in performance.

One would assume that GFS should help improve the models predictions, since a fatigued goalie should be less effective. However, it is obvious looking at Table 2 why adding GFS didn't provide any improvement. The goaltenders with the highest GFS, who are perceived to be the most fatigued, are all putting together unbelievable performances and are at the top of their game, so high GFS cannot easily be equated with decreased performance as one might expect from a fatigue statistic.

Although the Goaltender Fatigue Score didn't have an impact on the xGoals models performance, I still maintain that the overall premise has merit and something similar to how MLB teams track pitcher fatigue could be developed. Naturally there is a limited amount of public data that could help create such a statistic, but I feel that with the addition of some health metrics, like those from Whoop [3], we may be able to better understand how fatigue impacts goaltenders.

## 4 Future Considerations

There are a handful of possible improvements that I could make to this work, including adding some variation to how I encoded the categorical variables, enhancing the MoneyPuck data since there are some obvious improvements that could be made to the structure and omissions that could be added, and testing the performance of a hand crafted neural network.

In addition there are many other avenues to be pursued in the field of goaltending, like computer vision projects to better understand how efficient a goaltenders movement and positioning is when tracking pucks. Or leveraging the AWS tracking data to generate novel insights on goaltenders and determine potential areas of improvement.

If any of this sounds compelling and Utah HC is open to academic collaboration on PhD research please don't hesitate to reach out by email, rromaniuk31@gmail.com.

## 5 Acknowledgements

Big thanks to Steve, one of my Lead Developer's at work, for his encouragement throughout the project and advice when I ran into issues, and my academic supervisor Billy for letting me put off the work I was doing with him for a couple weeks to complete this project. Lastly, thank you to everyone at Utah HC who put this competition together.

## 6 References

- [1] David Alter. *"I don't think the long flight did him any favours": Maple Leafs' John Klingberg chances of playing first game in ...* URL: <https://thehockeynews.com/nhl/toronto-maple-leafs/latest-news/i-dont-think-the-long-flight-did-him-any-favours-maple-leafs-john-klingberg-chances-of-playing-first-game-in-stockholm-sweden-low>.
- [2] MoneyPuck. *MoneyPuck Data*. 2024. URL: <https://moneypuck.com/data.htm>.
- [3] Whoop. *Whoop*. 2024. URL: <https://www.whoop.com/ca/en/>.