SIADS 696 Milestone II Project Report
NHL Playoff Predictions and Player Clustering

Tyler Sandy (sandytyl@umich.edu)
Alvin Kuo(alvnkuo@umich.edu)
Te Zhang (zte@umich.edu)

1 INTRODUCTION

1.1 Introduction and Motivation

The National Hockey League (NHL) represents a gold mine for data analysis, offering a field where data scientists can apply both supervised and unsupervised learning to extract profound insights. Supervised learning models delve into the predictive side of the sport, enabling teams to anticipate game outcomes, player performances, or potential injuries, thereby informing strategic decisions crucial for winning games or optimizing player drafts. They're indispensable for media outlets and betting platforms, too, enhancing fan engagement through more accurate forecasts and interactive experiences based on solid data-driven predictions. Conversely, unsupervised learning unveils the subtle, often hidden, patterns and relationships within the skater data, offering an uncharted map of intricate player and game dynamics. It identifies natural groupings among players or playing styles, providing teams with unexpected insights that can influence game strategy to player development and scouting.

Our overall goal is to utilize supervised and unsupervised learning to conduct the prediction and clustering to help with analysis. Since the prediction has been thought it's hard due to the dynamic and uncertainty every year from NHL alliances and divisions to the team levels. We wish to find a way to better analyze them and solve the prediction and clustering problems. It may bring nuance to analyze the NHL prediction and analysis for the team players. We believe the techniques could be utilized well and test the boundary of power of data science analysis.

1.2 Problem, Impact, and Finding

1.2.1 Unsupervised Learning

Our unsupervised learning goal in this section is to utilize multiple unsupervised learning models from dimension deduction, and modeling to the evaluation of NHL player stats to find out the best way to cluster them in a meaningful group. This will help further index or group the players which may directly support supervised learning or as an indirect vehicle to conduct prediction by index or scoring.

Due to the diverse nature of our high-dimensional data, our approach involves reconfiguring the data through dimensionality reduction techniques. This strategy aims to enhance our visualization capabilities, thereby facilitating a more profound exploration of the dataset. Our objectives encompass: 1) **Problem** - We're identifying the most important characteristics that contribute to a hockey player's performance. This involves analyzing various player statistics and recognizing those factors that are crucial in defining a player's role and effectiveness on the ice. 2) **Novel finding and Impact**- After categorizing players as either forwards or defensemen, we're analyzing the data further to identify subgroups of Defensemen and Forwards in All Situations, 5-on-5 regular time, power play 5-on-4, or 4-on-5 situations, even the extreme other situations(5-on-3, 3-on-5…). By using advanced sensitivity analysis techniques, we detect simplicity of 2-group could be the best solution and 5-on-4 could be perfect timing to classify the best for Defensemnpatterns to showcase their extraordinary capability from their original specialization, revealing the varied playing styles within each position. It shows that Defensemen are relatively easier to be clustered than Forwards. It's believed to lead to a fresh field for analysis of Defensemen.

1.2.2 Supervised Learning

Our supervised learning goal in this section is to train a supervised learning model on NHL regular season stats to predict the number of playoff games a team will win that season. This will help determine whether the strength of a team during the regular season translates into the NHL playoffs, and potentially point to areas of focus for evaluating the success of NHL teams.

The workflow involves data preparation, EDA to determine linear relationships and variable distribution, defining

model evaluation metrics, selecting models, hyperparameter tuning, feature importance, and failure analysis.

The main finding of the modeling process after conducting feature importance is that offensive metrics are weighted more heavily than defensive metrics in the model, which goes against the traditional idea that “defense wins championships.” Additionally, further tuning is required to help with predicting rare events such as teams progressing past the 2nd round of the playoffs.

2. RELATED WORK

2.1 The Evolution of Hockey - Tyler Sandy’s Milestone 1 Project

This project is an extension of Tyler Sandy’s Milestone 1 Project which analyzed historical trends at a team and player level using data from MoneyPuck.com. The main findings from this project were the following:

1. **Scoring trends** at both the team and player levels have fluctuated over time, which is an important consideration for this predictive modeling project
2. **The identification of data sources** from MoneyPuck and the NHL API helped streamline the data collection aspect of this project
3. **The NHL began tracking different statistics at different points** in its history. The biggest example we must consider for this project is that “advanced” statistics that utilize one ice location started being collected during the 2008 season, which is why Money Puck only goes back that far.

2.2 More References and Extension

We benefit from multiple references as follows: 1) Using Rare Event Classification Modelling to Predict the 2023 Stanley Cup Playoffs offers a fresh look toward playoff prediction by classification models for us to evaluate the supervised modeling progress. In NHL Game Prediction and Season Simulation(Della Baby, D. & Shi, M., 2022), it’s been explored with various supervised models to predict with an accuracy maximum of 0.55 for NHL 32-team winning probability - in provides a rough cornerstone for the level of difficulty and dynamics. Clustering NBA Players Based on Statistics(Smith, A., 2020) - offers a potentially successful path to cluster with the NBA player statistics to find the superstar. The tank effect highlighted from Better Talent and Rule Changes Have Resulted in More Goals in the NHL is highly echoed by our clustering result from this: the worse team may play even worse to get a better draft pick of next season as a motive to play worse in standing. It indicates the player could have a dramatic inclination to be identified as worse than expected in a tanking pattern to be clustered in groups.

3. DATA SOURCE

The data for our supervised learning comes from two sources: MoneyPuck.com and the NHL API. The combination of data from these two sources allows us to make predictions about the NHL playoffs. The explanation below will provide insight into how these datasets were gathered and joined together.

3.1 MoneyPuck

MoneyPuck provides in-depth analytics in an easily downloadable CSV format. For our supervised learning, we utilized the team stats dataset called teams.csv, which spans from 2008 to 2023, similar to the skater stats. Each of the 2,290 rows represents a season for a particular team in different game situations, such as having an extra player on the ice during a “power play” or missing a player while on a “penalty kill.” The dataset also includes the stats aggregated for all situations, which is what we will primarily use. These regular statistics include goalsFor and goalsAgainst, and advanced stats include xGoalsFor and xGoalsAgainst that take into account the location of on-ice events. There are 101 features in total.

For unsupervised models, we harness comprehensive skater data spanning from 2008 to 2023. Clients can select any season, for instance, 2022 to 2023, for analysis via our unsupervised models. The datasets are available for download as CSV files, updated on a seasonal or annual basis. To illustrate, our 2022 dataset encompasses 154 columns and 4,755 rows, aggregating a substantial around 700K data points. These columns capture statistics reflecting nearly every facet of performance for forwards and defensemen—the primary skater classifications in hockey. Recognizing the distinct nature of performances between these categories, we meticulously categorize them separately, as opposed to a combined format. This dedicated separation enhances the granularity of our analysis, particularly in improving clustering outcomes. Not too much missing data in the initial preprocessing. But

all the statistics data may need to be investigated which is better for various types of skaters.

Peter Tanner, the creator of the Money Puck website, confirmed that the downloadable data comes from the official NHL API. Although we could query the data from the API ourselves, downloading the data directly from Money Puck gives us access to a wide range of features we would otherwise need to calculate ourselves.

3.2 Official NHL API

The data from Money Puck is missing our target variable of playoff wins and additional information about the team's regular season, which is why we must utilize the NHL API. The API has an endpoint and has unofficial documentation available. From the API we return 3 datasets:

1. **playoffs.csv:** returns 1,026 rows and 8 columns for each team's playoff results for every NHL season (1917-2023), including our target variable of playoff_wins
2. **standings.csv:** returns 1,695 rows and 14 columns for each team's regular season results for NHL season (1917-2023)
3. **abbrevs.csv:** a bridge table that uses team abbreviations to join to teams.csv, and a combined season/team id to join onto playoffs.csv

4. FEATURE ENGINEERING

4.1 Unsupervised feature engineering

4.1.1 Steps

Our data preprocessing involves several critical steps:

1) ***Label Clarification and removal* of unnecessary data*:*** We begin by identifying and removing labels after reviewing every single column. We believe the team, player's name and ID, and specific season are not necessarily helpful for the clustering and getting them removed. The rest of them would be kept.
2) ***Handling Missing Data and necessary data manipulation:*** Each dataset has to identify and address any instances of missing information, but our data is relatively clean, with almost no missing data issues. For data manipulation, we also carefully added a column of face-off winning probability for the model to learn instead of purely the quantity of win or loss.
3) ***Normalization versus Standardization***: Selecting the appropriate scaling method is crucial. Initially, we were skeptical about using both normalization and standardization, considering the differing methodologies—normalization adjusts data within a specific range. In contrast, standardization centers the data, allowing for both positive and negative deviations. However, our experience has shown that normalization is more suitable for our datasets. This preference stems from the nature of performance data across an 82-game season, where extreme values, or outliers, could be less of a concern. We care more about relative strength. Consequently, normalization has consistently yielded superior results.
4) ***Strategic Grouping:*** We further refine our analysis by separating data for forwards only, defensemen only, and both. This move could be validated by improved clustering accuracy due to different types of skaters will absolutely have a specific range of statistics with their roles and responsibilities. Later we will heavily count on this analysis

Through these methods, datasets from Money Puck deliver precise, insightful, and valuable hockey analytics, offering users an unparalleled understanding of the game player and we

4.1.2 Complete List

The complete list in the index is a 5-column and 152-row table with all the original and newly added columns/features including the type and the decision to drop or keep. A snapshot as below:

4.2 Supervised feature engineering

4.2.1 Steps

Here are the steps for processing data for our supervised learning model:

1. Utilized the bridge table abbrevs.csv to connect teams.csv (with our dependent variables) to playoffs.csv (with our independent variable) as well as standings.csv (with additional dependent variables)
2. Manually removed any columns that would not have an impact on modeling, such as several representations of the team name and regular season games played, as all teams play the same amount of games (except for the 2020 season which was cut short due to the COVID-19 pandemic)
3. Declared the unique ID for each row to be the seasonTeamID

4.2.2 Complete List

The appendix contains a complete list of the variables included in the final dataset

5. UNSUPERVISED LEARNING

5.1 Unsupervised Learning Methods

We utilize 3 different dimension techniques including PCA(Principal Component Analysis), t-SNE(t-distributed Stochastic Neighbor Embedding), and UMAP(Uniform Manifold Approximation and Projection). We also consider very different underlying mechanisms, (e.g. probabilistic, non-probabilistic, tree-based, instance-based) in 4 models including K-means, Hierarchical, GMM(Gaussian Mixture Model) & DBSCAN(Density-Based Spatial Clustering of Applications with Noise) as follows:

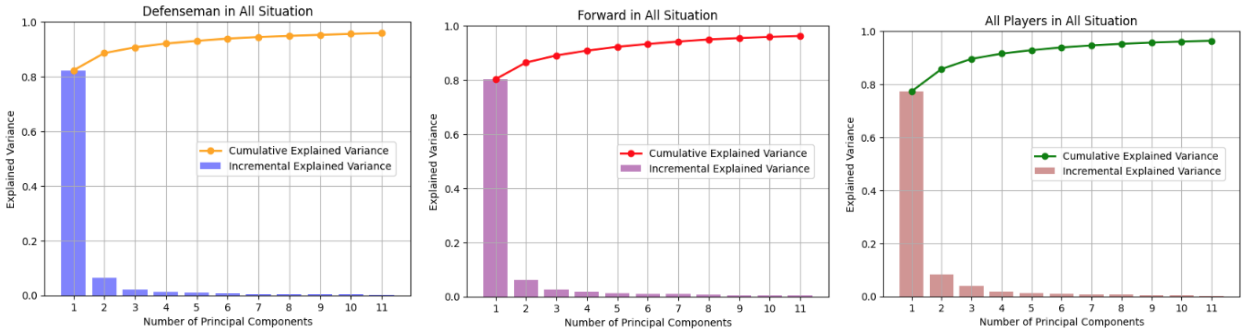
5.1.1 Dimension Reduction

5.1.1.1 PCA Analysis - Given our dataset's high dimensionality with maximum 150+ columns (if combining defenseman and forward), employing PCA is crucial. This technique helps us ascertain the optimal number of principal components that effectively capture the dataset's variance. Our comparative analysis unfolds as follows:

1) **Distinct Skater Types: Analyzing - All vs Forward vs. Defensemen** - PCA demonstrates enhanced efficacy when applied to distinct skater types separately rather than in combination. Remarkably, the cumulative explained variance sees a significant boost across all scenarios, particularly when the number of Principal Components (PC) is set to 2, with variances ranging from 0.7 to 0.9. Further exploration reveals that expanding PCs to 3, 5, or even 10 drives the cumulative variance above 0.9, showcasing PCA's suitability here.

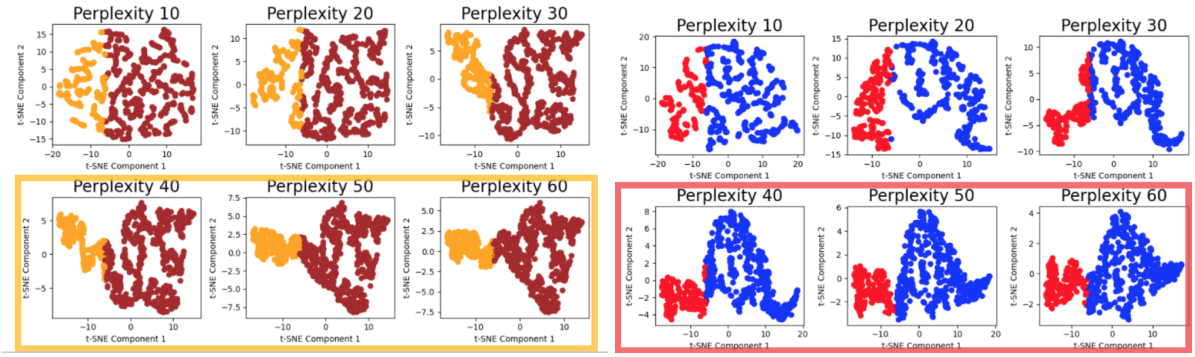
2) **Situational Analysis: All vs. 5-on-5 vs. 5-on-4 vs. 4-on-5 vs. Others** - From the EDA phase, we recognize that 5-on-5 play, accounting for roughly 80% of game time, represents the standard scenario. Conversely, 5-on-4 and 4-on-5 situations constitute around 5-10% each, while other scenarios are less frequent, under 5%. Despite the potential insights into players' performance during power plays (5-on-4 and 4-on-5), our dataset offers limited and highly imbalanced data in these categories. Interestingly, PCA yields the best performance with 5-on-5 data, guiding our decision to adopt 5-on-5 as the benchmark for subsequent analysis

Figure 1. ***PCA Comparisons Among All Players, Defensemen, and Forwards Types***



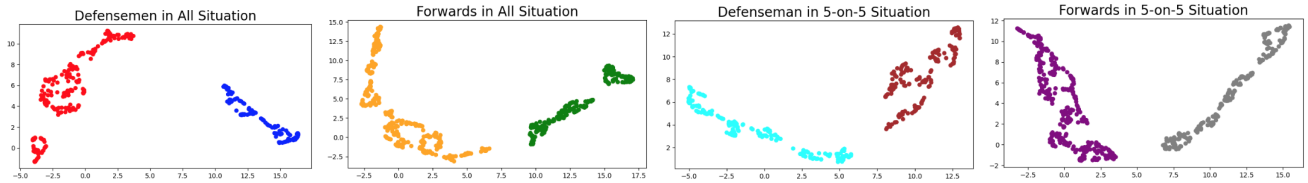
5.1.1.2 t-SNE Analysis - *In Figure 2,* We chose t-SNE after PCA because we want to harness PCA's ability to grab the most meaningful data and t-SNE's strong visualization skills. We tested different perplexity levels, from 10 to 60, to study the groupings. It became clear that *2 groups* are reasonable choices with *both Defensemen (Left) and Forwards (Right) Types under all situations*. In our repeated tests, the threshold works best in *perplexity levels of 40, 50, and 60*. Specifically, to leverage the best of t-SNE's visual strength a *threshold of -6* gave us a clear picture of the 2 groups right away as the best solution.

Figure 2. ***t-SNE Comparisons between Defensemen/Forwards Type/Various Perplexities***



5.1.1.3 UMAP Exploration - *In Figure 3,* for our UMAP analysis, we demonstrate to use of data that had already been processed through PCA, a decision driven by the high-dimensional nature of our original dataset, with the rationale of reducing noise and facilitating faster computations, not to mention improving visualization clarity and being more memory-efficient. From *Defenseman in All Situation (Left 1)* and *Forwards in All Situation(Left 2),* it easily clusters into *2 distinct groups.* This pattern was strengthened again when analyzing 5-on-5 scenarios - both *Defensemen (Right 2)* and *Forwards (Right 1)* give us a very clear 2-cluster result with default *n_neighbor = 15, min_dist = 0.1 (minimum distance) and learning_rate = 1.0*, and we can see all the *PCA, t-SNE, and UMAP* resonate in the same clustering direction.

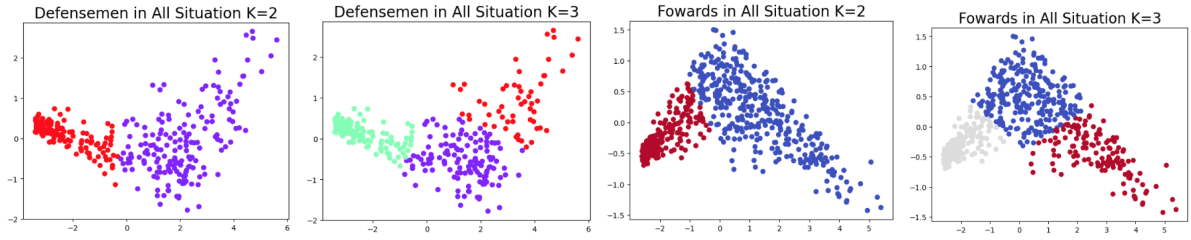
Figure 3. ***UMAP Comparisons between All/5-on-5 situations in Defensemen/Forwards Type***



5.1.2 Unsupervised Models

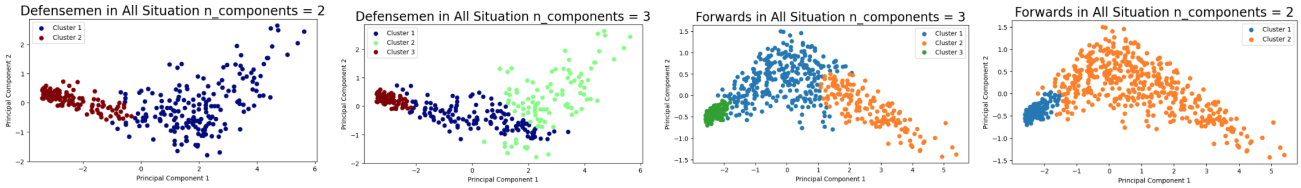
5.1.2.1 Non-probabilistic K-Means - Non-probabilistic unsupervised methods work with distances to group data, without providing a probability distribution that underlies these groupings. These are often used for their computational efficiency and ability to produce solid groupings. In *Figure 4*, We choose K-means clustering, a classic non-probabilistic method, dividing data into a fixed number of clusters based on spatial distance measures. K-means requires a specific K group pre-assigned. In *All Situations*, We tried *K=2* as suggested in *Defensemen (Left 1) and Forwards (Right 2)* after the 3-approach dimensionality reduction process and also K=3 for *Defensemen(Left 2)* then got the result as follows. All the results have been great with the default *init = k_means++ method, n_init = 10 Initiations, and the max_iter = 300 (maximum iterations)*. The Elbow Method and Silhouette Score will be introduced to evaluate later.

Figure 4. ***K-Means Comparisons among All situations in Defensemen/Forwards When K = 2 or 3***



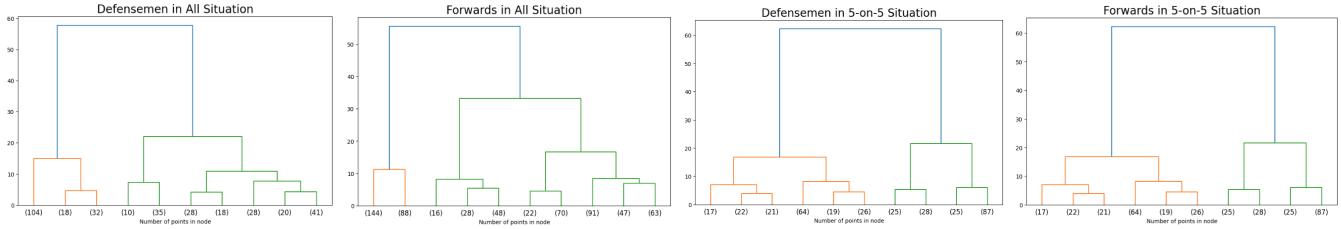
5.1.2.2 Probabilistic GMM - Probabilistic unsupervised methods involve statistical models that infer the probability distributions underlying the data. They attempt to understand the hidden and generative structures that give rise to the data and can provide a measure of uncertainty regarding the model's findings. In Figure 5, we demonstrate in All Situations, *Defensemen when n_componenets = 2 (Left 1)* and *n_components = 3 (Left 2)*. These are used when it's important to understand the uncertainty or likelihood of different outcomes, often in complex data environments with clusters shown. They can be computationally intensive. We also show in All Situations, *Fowards when n_componenets = 2 (Right 2)* and *n_components = 3 (Right 1)*. GMM is a common example where the data is assumed to come from multiple Gaussian distributions, and the algorithm tries to learn these distributions. We specifically tried many hyperparameters and we are satisfied with *default covariance_type = full (covariance matrix), max_iter = 100 (max. iterations), and tol = 0.001 (tolerance)*. The BIC (Bayesian Information Criterion) and AIC (Akaike Information Criterion) evaluation for GMM specifically will be addressed later.

Figure 5. GMM Comparisons *among Defensemen/Forwards When n_component = 2 or 3*



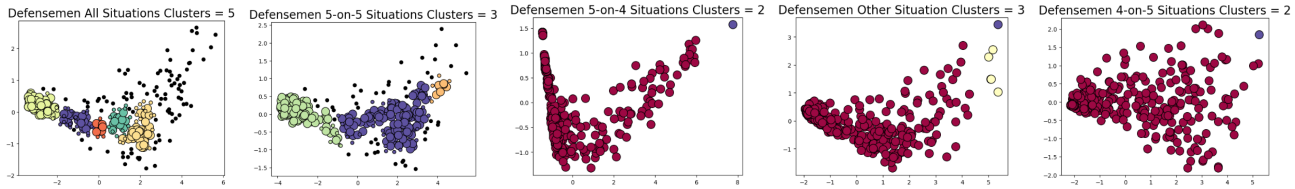
5.1.2.3. Tree-Based Hierarchical: These methods involve hierarchical structures for the data and are used in unsupervised learning to create a tree of clusters. Overall, they're less about individual instance predictions and more about understanding the data structure. These are used for hierarchical clustering with the goal to create a cluster tree that may be viewed at different scales, and for anomaly detection. Hierarchical clustering, which creates a tree of clusters by iteratively merging or splitting groups of data points. In *Figure 6*, we chose the *agglomerative method*. Even if we specifically choose p = 10 (numbers of clusters), it's easily seen the long arm indicates that 2 is the best cluster choice if *any threshold is 30+* in *Defenseman in All Situations (Left 1)*, *Defensemen in 5-on-5 Situation (Right 2)*, and *Forwards in 5-on-5 Situation (Right 1)*. The only case that needs *threshold 40+ is Forwards in All Situations(Left 2*). Overall it's very consistent in every test, so far the 2-group has the best clustering results. We pick the hyperparameter *truncate_mode = lastp (last p)* to benefit from easy-visualized data under that branch or cluster.

Figure 6.Hierarchical Comparisons *between All/5-on-5 Situations in Defensemen/Forwards Types*



5.1.2.4 Instance-Based DBSCAN: Instance-based learning in an unsupervised context refers to methods that cluster or organize data based on similarity measures between individual instances without prior learning from labeled responses. They don't form a generalized model but work by comparing new data points to specific instances in the existing data. We pick DBSCAN, a clustering algorithm that groups together points that are closely packed together, marking points that lie alone in low-density regions as outliers. Grid search is the core operation here to get the best Silhouette Score result as our pick for clusters. We set up *eps* (max. distance between samples) between 0.1 and 3, and a *minimum sample* between 1 and 10. In *Figure 7*, DBSCAN gave us five groups in *Defensemen All Situations (Left 1)* with *eps = 0.3* and minimum sample 5. Also 3 groups in Defensemen 5-on-5 Situation (Left 2) with eps = 0.3 and minimum sample 5 again. With the best evaluation results from the silhouette score, it looks like the results of 5-on-4, 4-on-5 or Other Situations are not impressive. In the chart, the black points are outliers. DBSCAN identifies a portion of the outliers in the first 2scenario.

Figure 7.DBSCAN Comparisons *among Defensemen All/5-on-5/5-on-4/4-on-5/Other Situations*

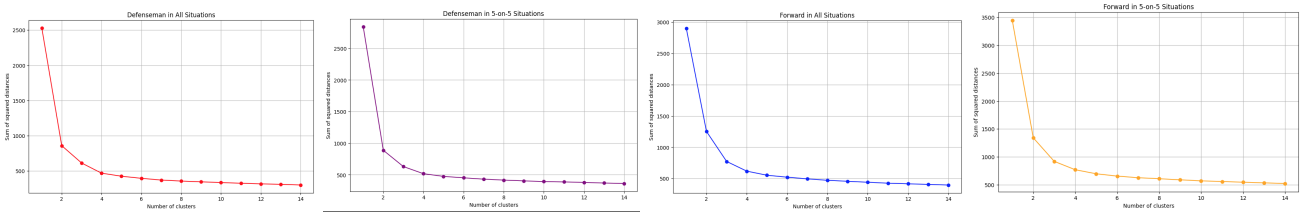


5.2.Unsupervised Learning Evaluations

Here we introduced 4 different unsupervised learning evaluations: Elbow Method, Sillouhette Score, and BIC/AIC.

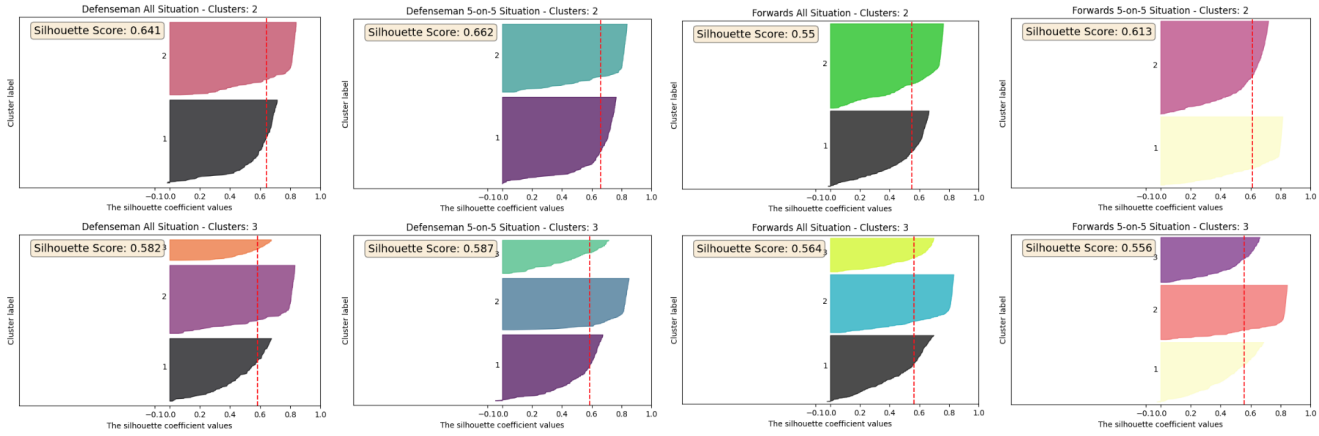
5.2.1 Elbow Method: Primarily used to determine the optimal number of clusters in K-means clustering. It involves plotting the explained variation as a function of the number of clusters and picking the "elbow" of the curve as the number of clusters to use. This elbow point is where the rate of decrease sharply changes, reflecting an optimal cluster number beyond which additional clusters don't explain sufficient variance. Intuitive and easy to implement. The "elbow" may not always be clear or easy to identify, making it somewhat subjective. In *Figure 8*, We are lucky to test the K-mean and run through all situations. We get 2 is sharply steep going down, and then 3 has a very significant slowdown. We believe 2 is the best group choice throughout various combinations.

Figure 8. Elbow Comparisons *between All/5-on-5 situations in Defenseman/Forward Types*



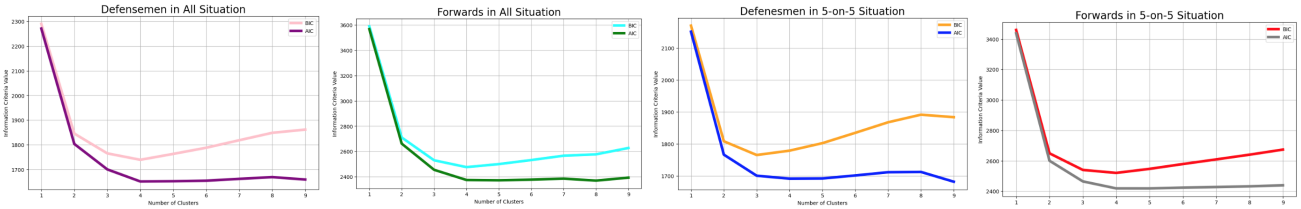
5.2.2 Sillouhette Score: The silhouette score uses the mean intra-cluster distance and the mean nearest-cluster distance for each sample, averaging the score over all samples. A higher silhouette score indicates that the object is better matched to its own cluster and poorly matched to neighboring clusters. This method provides a clear metric for cluster cohesion and separation, which can aid in validating the consistency within clusters and the adequacy of the clustering. It can be computationally intensive for large datasets. In *Figure 9*, We measured the K-mean and got 0.5~0.7 above for most of group 2 and got lower scores like 0.4 to 0.6 in group 3. We're very confident about the mostly moderate score for group 2 to prove that 2 clusters are better than 3.

Figure 9. Silhouette Comparisons *between All/5-on-5 Situations in Defenseman/Forward Types*



5.2.3 BIC and AIC: The Bayesian Information Criterion (BIC) and the Akaike Information Criterion (AIC) balance model fit with complexity, penalizing models with more parameters to prevent overfitting. Lower BIC or AIC values indicate a better model, but they don't guarantee the best predictive accuracy. BIC applies a stricter penalty than AIC, and neither provides an absolute measure of quality, only a relative comparison across models. The optimal model isn't solely determined by the lowest AIC or BIC, as other performance metrics also matter. In *Figure 10*, we always see the BIC jumps higher than AIC. BIC penalizes model complexity more heavily than AIC, making it more stringent in selecting models with fewer parameters, especially as sample sizes increase. Like the *Elbow method* concept, we look for the *turning point* with the sharpest drop, and then the slope changes to not so steep. We learned from all the 4 *GMM* models we're measuring, that the best model would be the *2-cluster* for all.

Figure 10. BIC/AIC Comparisons *between All/5-on-5 in Defensemen/Forwards Type*



5.3 Overall results

5.3.1 Justification for choice of evaluation metrics

Our nature of the data is high-dimensions. We did all the possible 3 approaches and decided to use PCA-transformed data afterward. After 4 different clustering methods, we still mostly get very consistent results. After most of the models we have, are all inclined to suggest the 2-group is the best choice in multiple scenarios no matter Defensemen or Forwards, in All or 5-on-5 Situations. The most obvious results are the Elbow-concept Elbow and BIC/AIC all indicate the same results and turning point of the 2-cluster. We also get the quantitative score support from Silhouette Score pointing to the 2-group best solutions.

5.3.2 Final Report and Comparison

For finding the optimal number of clusters, we used the Elbow plot and silhouette plot for K-Means. AIC/BIC for GMM. We visualize picking the clusters from the hierarchical dendrogram. They all came to the same number of clusters as mentioned above. Though DBSCAN may give us a somewhat inconsistent indication, we still believe in all the signs from various kinds of angles to cluster them into two groups.

5.4 Sensitivity Analysis

We conduct repetitive sensitive analysis among all the 3 distinctive dimensionality reduction approaches, 4 different clustering methods, and 3 various measurements among 1) 2 Types of transformation of data: Normalization or Standardized Data 2) 5 Situations - All, 5-on-5, 5-on-4, 4-on-5 and Others Situations, 3) 2 Types of Skaters/Players - Defenseman and Forward 4) Number of Clusters - If pre-assigned doable like 2 or 3 groups.

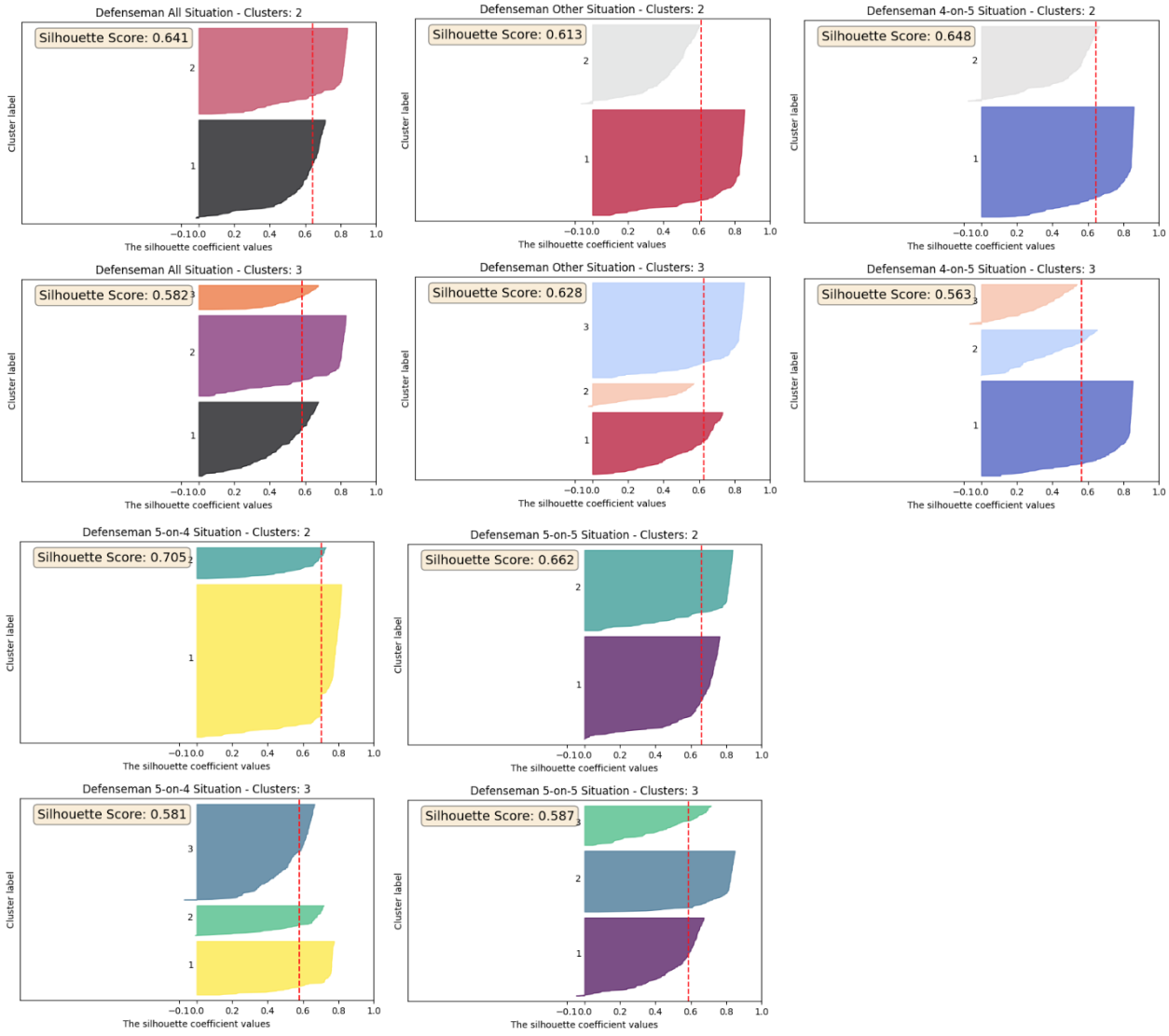
In *Figure 11*, Here comes a very special scenario: "Changing 5 *Sitautions with 2 or 3 Clusters Under Defenseman Type in K-mean Clustering"*. Firstly. we got a very significant direction *Defenseman's clustering universally performs better than Forward* previously(Due to Forward may have Left Wing, Center, and Right Wing 3 different sub-groups with a little bit more diversity) so we investigate *Defenseman* only here. Secondly, while we're testing the sensitivity analysis among different numbers of clusters from K-mean, the *2-cluster always has a higher silhouette score than the 3-cluster.* (2-cluster ranges from 0.641 to 0.705 while 3-cluster ranges from 0.581 to 0.628). Then here we got the best score among the 5 situations in a 5-on-4 situation (2-cluster 0.705).

Why 5-on-4 power play situation stand out as the best data for us to classify defenseman so well? During a 5-on-4 power play, this defenseman's team has five players, and the opposing team has four players. Defensemen in this situation often have specific responsibilities, such as maintaining control of the puck in the offensive zone, setting up plays, or taking shots from the point. How a defenseman performs in these roles can be very telling of their skills and style of play. While defensemen are traditionally more focused on preventing goals, now they often have more opportunities to participate in the offense during a power play to showcase their extra capability of scoring opportunities (e.g., through assists, shots on goal, or actual goals) - those can separate more offensively skilled

defensemen from those who are primarily defensive.

After 5-on-4, the 5-on-5 which is the normal 80-85% of the situations when there are 5 players each on-ice, performs as the second best score (2-cluster 0.662), and in another power play 4-on-5 situation comes after (2-cluster 0.648). For the defensemen, the all-situation (which includes every 5-on-5, 5-on-4, 4-on-5, and other situations) is eventually weaker than the mentioned above (2-cluster 0.641). Then the worse one comes as the "Other Situation" (mixed with all the 5-on-3, 5-on-2, 3-on-5, 2-on-5 extreme cases)(2-cluster 0.613) if we try to compare apples to apples. It makes sense since it mingles all the rare and outlier situations that may defeat the learning of our supervised model.

Figure 11. Sensitivity Analysis *of Silhouette Score by Defensemen Type for 5 Situations when K=2 or 3*



6. SUPERVISED LEARNING

6.1 Supervised Learning Methods

The nature of our dataset and our desired output impacted the models and evaluation metrics that we selected. We explored several models from 3 distinct model families outlined below:

- **Linear Regression** - effective for predicting continuous outputs and easily explainable
- **Lasso Regression** - can help with feature selection by shrinking the weight of less relevant features
- **Ride Regression** - similar to Lasso but uses L2 regularization to mitigate multicollinearity
- **Decision Tree** - effective for capturing non-linear trends in the data

- **Random Forest** - similar to a decision tree, but multiple trees may help with overfitting
- **Extreme Gradient Boosting (XGBoost)** - also helps with overfitting, may increase accuracy, and has feature importance score
- **Support Vector Machine (SVM) with RBF Kernel** - may increase accuracy compared to tree models at expense of explainability with feature importance score

6.2 Supervised Learning Data Preparation

We used an 80/20 train/test split for our data. We opted for this method instead of splitting based on a range of seasons, such as 2008-2020 for training and 2021-2023 for testing, because a randomly selected 80/20 split is not only straightforward but may also help capture changes in the league stats over time, such as goal scoring levels (per Tyler Sandy's Milestone 1 Project)

Additionally, we performed MinMax scaling on the dataset. Although this is typically not beneficial for tree-based methods such as Random Forest that do not rely on the scales of features, we did it to potentially improve the performance of other models that we will evaluate, such as Lasso and Ridge regression.

6.3 Supervised Learning Evaluation

To evaluate our models, we chose Mean Absolute Error (MAE) because it explains the difference between the model's predictions and the actual values. It uses the same unit of measurement as our target variable so the results are easily explainable. For example, we can say that our predictions were off by *X* wins on average.

6.4 Supervised Learning Model Selection

We performed 5-fold cross-validation on baseline versions of these models without hyperparameter tuning, and the results are shown in Figure 12 below. Part D of the appendix also includes additional metrics that we measured model performance with, such as Median Absolute Error, Mean Square Error, Root Mean Square Error, and R-squared. This is to establish which models are the most effective at addressing our problem without any additional tuning. As a result, we selected a Random Forest Regressor (RF). Although RF had only the 3rd best MAE, we chose it for our model for multiple reasons:

1. It has a relatively high MAE compared to other models.
2. Although SVM had the highest mean MAE across the 5-fold validation, it had a relatively large standard deviation, indicating there is variability in how well the model performs.
3. Lasso regression had the 2nd best average MAE. However, the multicollinearity in the dataset means we would need to perform principal component analysis (PCA) to run the model, and that would be at the expense of being able to perform feature analysis.
4. The nature of random forest modeling allows us to bypass the feature selection stage on the roughly 100 features in the dataset.
5. Being an ensemble method that utilizes multiple decision trees, it helps mitigate overfitting. Highlighted by the significantly better MAE compared to a regular decision tree.
6. We can calculate feature importance to extract additional insights from the model.

6.5 Hyperparameter Tuning and Sensitivity Analysis

Figure 12: Supervised Model Scoring

	Model	MAE Mean	MAE Std
0	Linear Regression	5.851294	0.612683
1	Lasso	3.703534	0.299376
2	Ridge	3.849031	0.231197
3	Random Forest	3.731200	0.342672
4	Decision Tree	4.652605	0.471521
5	Support Vector Machine	3.526994	0.550275

After selecting Random Forest as our model of choice, we ran GridSearchCV hyperparameter tuning to achieve the best MAE. We used the combinations of hyperparameters listed below in Figure 12. For an explanation of Random Forest model hyperparameters, reference Appendix D.

Figure 13: Random Forest Model Parameter Grid

```
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}
```

Figure 13 shows that even the best combination of hyperparameters led to a drastically worse result than a non-tuned model. We ran 5-fold cross-validation on 320 fit combinations resulting in 1,600 fits in total.

Figure 14: Random Forest Model Hyperparameter Tuning

	param_n_estimators	param_max_depth	param_min_samples_split	param_min_samples_leaf	MAE Mean	MAE Std
52	200	10	10	4	5.887300	4.235713
79	200	20	10	4	5.888543	4.249804
106	200	30	10	4	5.888543	4.249804
25	200	None	10	4	5.888543	4.249804
53	300	10	10	4	5.891043	4.009769
80	300	20	10	4	5.891892	4.019766
26	300	None	10	4	5.891892	4.019766
107	300	30	10	4	5.891892	4.019766
100	200	30	2	4	5.900038	4.202026
22	200	None	5	4	5.900038	4.202026

6.6 Tradeoff Analysis

As a result of this analysis, there are 2 main tradeoffs that we can identify:

1. **Accuracy vs explainability:** the Random Forest model did not achieve the best MAE, but the straightforward and interpretable way to measure feature importance gave it the edge and is why it was selected
2. **Speed vs accuracy:** running dimensionality reduction techniques such as PCA could potentially increase the speed of running the models, which would be beneficial given that we are trying to evaluate multiple models at a time. However, this comes at the cost of being able to explain how each feature impacted the model, which is why we opted not to perform PCA

6.7 Evaluation - Overall Model Performance

Overall, the model was able to predict the number of playoff games a team would win within **3.515** games. Keeping in mind the structure of the NHL playoffs, 4 wins are needed to win a series. This means the model typically predicts within 1 round how many rounds a team will win. Additional evaluation metrics are included in Part D of the appendix.

6.8 Feature Importance Analysis

After running the final model, we performed a feature importance analysis to get an understanding of which features are contributing to the success of the model, and which are not. Ablation analysis would be costly to run due to our large feature set, so we calculated feature importance using the Gini Impurity Index. This index measures how much each feature contributes to the reduction of impurity (probability of misclassifying a data point) during the splitting process for decision trees. Full tables with the values for each feature are included in the appendix.

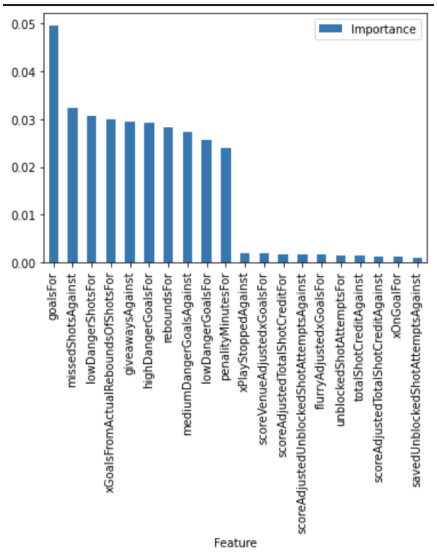
The model appears to weigh offensive metrics more heavily than defensive metrics as seen by goalsFor, giveawaysAgainst (turnovers by the other team), reboundsFor, and several other scoring-related metrics. goalsAgainst, on the other hand, was ranked 27th with an importance score of 0.013359. 0.013359

The model also learned that hitsAgainst is an important feature, which is unintuitive. This is likely because teams

that possess the puck more are likely to score and win more, and having the puck more opens up the opportunity to be hit by the other team more. Surprisingly, xGoalsAgainst had one of the lowest importance scores in the model, further showcasing that the model does not value defensive metrics highly.

The model reduced the importance of several features such as xPlayStoppedAgainst and scoreVenueAdjustedxGoalsAgainst. This makes sense, as these features and several others on this list do not necessarily reflect anything positive or negative about a team's performance and may be too specific and uncommon to have a major impact.

Figure 15: Top 10 and Bottom 10 Contributing Features to Random Forest Model

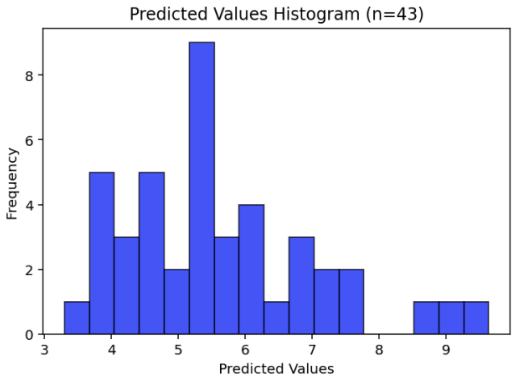


6.9 Failure Analysis

By analyzing some of the individual predictions of the model, we can evaluate cases where the model has failed. There are 3 that we will highlight.

Figure 16 below shows the distribution of predictions our model made. One error that stands out is the model didn't predict that any team would win the minimum of 0 or the maximum of 16 games. However, at least 1 team wins 16 games and is the champion champion in each playoff. This is likely due to the "class imbalance" in the training set (keeping in mind we approached this as a regression problem), where most teams only win enough games to progress to the 2nd round of the playoffs (between 4 and 7 games), and no further.

Figure 16: Random Forest Model Prediction Distribution



To mitigate this issue in future modeling, we could experiment with alternative train/test split methods that take into account the class balance, such as oversampling to increase the representation of the minority class of winning more than 5-6 games.

Because of this, the model incorrectly predicted that several teams would win fewer games than they ultimately would. Figure 17 below highlights the top 5 instances where a team overperformed the model's expectations. The prime example is the 2017 Stanley Cup champion Pittsburgh Penguins, who were expected to win 5.09 games according to the model

Figure 17: Top 5 Teams Overperforming Prediction

teamSeasonID	season_id	team.name	Actual	Predicted	diff	absoluteDiff	predicted_roundReached	actual_roundReached
52017	20162017.0	Pittsburgh Penguins	16.0	5.09	10.91	10.91	Round 2	Champion
42020	20192020.0	Philadelphia Flyers	10.0	4.00	6.00	6.00	Round 2	Round 3
32012	20112012.0	New York Rangers	10.0	4.02	5.98	5.98	Round 2	Round 3
242015	20142015.0	Anaheim Ducks	11.0	5.24	5.76	5.76	Round 2	Round 3
32015	20142015.0	New York Rangers	11.0	5.45	5.55	5.55	Round 2	Round 3

As previously mentioned, the model rarely predicts first-round exits or advancement past the 2nd round. In the examples below, the model over-predicted how many games would be won. One example is the 2021 St Louis Blues, who did not win a single game in the playoffs.

Figure 18: Top 5 Teams Underperforming Prediction

teamSeasonID	season_id	team.name	Actual	Predicted	diff	absoluteDiff	predicted_roundReached	actual_roundReached
192013	20122013.0	St. Louis Blues	2.0	6.61	-4.61	4.61	Round 2	Round 1
192021	20202021.0	St. Louis Blues	0.0	5.37	-5.37	5.37	Round 2	Round 1
222021	20202021.0	Edmonton Oilers	0.0	5.58	-5.58	5.58	Round 2	Round 1
132021	20202021.0	Florida Panthers	2.0	7.62	-5.62	5.62	Round 2	Round 1
232013	20122013.0	Vancouver Canucks	0.0	5.76	-5.76	5.76	Round 2	Round 1

In addition to addressing the class imbalance with our train/test split methods, we can look to add additional data to the model with the goal of improving performance. New data could include important factors in playoff wins such as the team's first-round opponent and their record against them during the regular season. Additionally, we could capture the experience of the roster. It is often said that veteran players with more experience in the league are a driving factor for winning in the playoffs, so the model could account for that. Finally, the team's stats and Win/Loss record near the end of the regular season would highlight whether they are performing well before entering the playoffs, and this may be predictive of playoff success.

7. DISCUSSION

7.1. Unsupervised

7.1.1 Learning

We learned about the sophisticated relationships among various approaches from dimension reduction, clustering methods compabaitlibitliy, and suitable evaluation methods accordingly. They are not purely models with fancy visualizations but are full of *actionable insights* behind *appropriate setups, hyperparameter finetuning,* and *thoughtful application-oriented algorithms*. To streamline every one of them successfully with cohesive logic could be challenging but also a harvest from solid stories and applications.

7.1.2 Suprise

The most surprising part of our results is the score ranging from 0.5 to 0.6 Sillouhette Score. We only expect very mild like 0.1~0.3 potential results since we are worried that 2 groups are too rough or simple clustering which may lead to very low Sillouhette Socre. However, it seems our strategy from one-by-one statistics features engineering and choice of normalization (standardization did not generate a better one) with comprehensive pipelines from dimensionality reduction, model selections/trials to scrutinized evaluations, The 2-cluster K-Mean proves the best Sillouhette Socre when Defensmen 5-on-4 Situation could generate the highest 0.602 for us - which could be almost classified stronger than moderate.

7.1.3 Challenge

One of the most challenging parts is the GMM since we chose to normalize but not standardize the data before PCA - likely the data is not favored by GMM to learn. We had very low expectations to get any meaningful interpretation from GMM and AIC/BIC evaluation, but eventually not bad. At least we get AIC showing some elbow turning point. Another one of the most challenging parts is the DBSCAN. We barely get insightful clusters from smaller subsets (5-on-4, 4-on-5, and others), but we still managed to draw the conclusions from Defensemen All Situations and 5-on-5 with identified 5 and 3 clusters.

7.1.4 Future extension

We see huge potential with future extensions after our structured and comprehensive unsupervised learning fruitful results as follows:

1. **Score/Indexing for Prediction** - Since most of the clustering indicates the 2 groups as the best bet. If we could figure out they are probably the higher and lower-performance ones after checking the key score/shot-related statistics. It's doable with the labels and weight/score for the team player. Say 2021-2022 is the season we generate the score/index afterward, we could test if the score/index could contribute to the 2022-2023 prediction with the roster list. It's doable with the measurement by RMSE or a similar approach to either analyze the standing or the winner probability.
2. **Multiple-season analysis** - Since we start from one season, the NHL has a 100+ year history. It could be worth watching the longer-term clustering results to see how they perform over time. (Say still simpler clustering results like 2 or 3 or more sophisticated groups with distinction.
3. **Looking for the Current/Next NHL Superstar** - like the article we mentioned before, utilize the Altair "Tooltip" to demonstrate and compare the Superstar or the top layers grouping position to look for the next Superstar from the chart.

7.2 Supervised

7.2.1 Learning

We learned several things from the supervised learning portion of this project:

1. Offensive statistics during the NHL regular season are better predictors of playoff success than defensive metrics, going against the commonly held belief that "defense wins championships."
2. The metrics that had the highest impact on the model weren't advanced stat, but simply goalsFor. This indicates that regular stats can be effective at predictions along with more advanced stats
3. Model evaluation scores aren't the only important aspect of machine learning projects. In many cases, the results need to be interpretable. How we selected a random forest over a higher performing SVM model because the random forest allowed us to determine feature importance and learn more for future analysis

7.2.2 Surprise

There were several surprises from this process:

1. As mentioned in 7.2.1 Learning section, offensive metrics were weighted more heavily in the model.
2. The model predicted 4 times that a team would amass enough wins to the 3rd round of the playoffs. In 2/4 of these cases, the model was correct, and both actually made it to the Stanley Cup finals
3. HitsAgainst was weighted heavily in the model, indicating that there is a correlation between the number of times the team is hit and how many games they win. This doesn't make sense intuitively and is likely due to confounding factors. Teams that have possession of the puck are able to be hit. Thus, if a team has more pucks, they will likely be hit more. If a team has possession of the puck more often, they are more likely to score and win more. That is how this variable is incorrectly connected to wins, and should potentially be removed in future modeling.

7.2.3 Challenge

Our team faced several challenges during the supervised learning process:

1. We initially set out to predict NHL standings before we pivoted to predicting playoff success. We were unable to make these predictions because data availability did not match our proposed methodology. The NHL API did not adequately track the historical tenure of players on each team, therefore we couldn't gather training data to use for predicting standings based on the strength of the roster.
2. We set out to gather additional data beyond the initial set of features. This took up a lot of time early on in the project lifecycle that could have been spent on refining the modeling methodology with the data that we already had. These efforts to acquire additional data mostly came up short.

7.2.4 Future extension

With additional time and resources, there are several things we can do to potentially increase the performance of our supervised learning model:

1. As mentioned previously, including additional data, such as playoff matchups, playoff seed, record near the end of the season, features for players on the team such as combined regular season and playoff experience, coaching tenure, etc.
2. Additionally, we could extend to more seasons of training data pre-2008, with the tradeoff of not having advanced stats available before the 2008 season

8. ETHICAL CONSIDERATIONS

8.1 Unsupervised

While clustering players based on performance statistics has merits in understanding play styles and strengths, there is a risk it could lead to unfair profiling or discrimination of certain player groups. For example, clustering may reveal declines in older players' stats. Teams could use this to avoid signing or trading for veterans perceived as being past their prime. However, some veterans remain highly skilled and effective despite aging. Blanket assumptions could deprive them of opportunities and fair salaries they still deserve.

To mitigate this, analysts should contextualize performance clusters and not solely rely on them for decisions. Consider individual circumstances and track records. Have personal conversations to assess capabilities. Implement policies to ensure fair treatment regardless of age or cluster profiles. The insights should inform, not dictate decisions.

8.2 Supervised

The predictive models estimating playoff success could inflate expectations and pressure for teams and players. If a team is predicted to go far, failing to meet that bar could be seen as a disappointment, even if their season was reasonably successful. Players may feel they let the fans down.

To avoid this, stakeholders should remember predictions have inherent uncertainty. Outcomes depend on many complex factors. Success is not black and white. Transparent communication is key - be open about model limitations, and keep perspective even if projections are beat. Focus praise and criticism on actual performance, not pre-season estimates.

In both cases, ethical risks arise when data-driven insights are not properly contextualized. Thoughtful policies and transparent communication can help uphold fairness and perspective. The models should inform human decisions, not replace them.

Statement of Work

Table 1: Statement of the Work

Tyler Sandy	Alvin Kuo	Te Zhang
1) Proposal writing, 2) Data source identification and gathering, 3) Supervised modeling and evaluation, 4) Report writing, 5) Statement of work, 6) NHL data domain expert	1)Organizing and presenting stand-up content, 2) Unsupervised modeling and evaluation, 3) Data pipeline construction, 4) Report structure and Project Management, 5) Report writing, 6) References	1) Supervised learning for a possible direction, 2) Proofreading, 3) Ethical consideration

APPENDIX

Appendix A - References

[1] Using Rare Event Classification Modelling to Predict the 2023 Stanley Cup Playoffs (Josselyn, A., 2023) DataDrivenInvesetor, Medium.

[2] NHL Game Prediction and Season Simulation(Della Baby, D., & Shi, M., 2022), Master's thesis, Department of Computer and Information Science, Linköping University

[3] Clustering NBA Players Based on Statistics(Smith, A., 2020) Medium.

[4] Better Talent and Rule Changes Have Resulted in More Goals in the NHL: An Analysis of the Increase in NHL Goal Scoring From 2011 to 2023(Ngo, J., 2023), Telling Stories with Data.

[5] Tyler Sandy's The Evolution of Hockey History: An Analysis of Historical Scoring Trend in the NHL(Dougall W., Sandy T., and Fery Bl., 2022), Milestone I, School of Information, University of Michigan, Ann Arbor

Appendix B - Hockey Terminology

Tabel B1 - Hockey Terminology Explanation

Term	Explanation
Corsi	Corsi measures the total number of shot attempts (shots on goal, missed shots, and blocked shots) generated by a team or player during a game. It provides a broader view of a team's or player's overall offensive and possession performance, including all shot attempts, whether blocked or not.
Drawn	For example penalty drawn means a player's ability to draw penalties from opponents, as mentioned earlier. Drawing penalties can disrupt the opposing team's game plan and create scoring opportunities.
Expected Goal (xG or xGoals)	It assesses every shot taken during a game, assigning a probability that the shot will result in a goal.
Fenwick	Fenwick, on the other hand, focuses specifically on unblocked shot attempts, which includes shots on goal and missed shots but excludes blocked shots. By excluding blocked shots, Fenwick emphasizes the shooting opportunities that were not impeded by defenders, making it a metric that often highlights a team's or player's ability to generate high-quality scoring chances.
flurry	Typically refers to a rapid and intense sequence of scoring opportunities or offensive activity around the opposing team's net. During a flurry, a team applies sustained pressure on the opposing defense and goaltender, leading to multiple shots on goal, rebound attempts, and scrambles in front of the net.
Giveaways	In the team's defensive zone" refers to instances where a player on the player's own team turns over the puck to the opposing team while they are in their own defensive zone
On the fly	A player is changing (substituting) during active play rather than during a stoppage in play
Penalty Minutes (PIM)	The total number of minutes a player has spent in the penalty box serving penalties during a game or over a season
Shift	A "shift" refers to the period of time during which a player is on the ice actively participating in the game. Players typically rotate on and off the ice during a game to stay fresh.

Appendix C - Unsupervised Data Schema

Data Source: [Money Puck]

Table C1 - Money Puck Player Data Schema: Dictionary and Decision

#	Column Name	Description	Type	Decision
1	corsiAgainstAfterShifts	Shot attempts the opposing team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to identify players who shift off instead of backchecking	all	keep
2	corsiForAfterShifts	Shot attempts the player's team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to give credit to give players who leave the ice when the puck is going towards the opposing team's zone	all	keep
3	faceoffsLost	Number of faceoffs the player has lost	forward	keep
4	faceoffsWon	Number of faceoffs the player has won	forward	keep
5	faceoffsWonPercentage	Percentage of faceoffs the player has won	all	new
6	fenwickAgainstAfterShifts	Unblocked Shot attempts the opposing team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to identify players who shift off instead of backchecking	all	keep
7	fenwickForAfterShifts	Unblocked Shot attempts the player's team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to give credit to give players who leave the ice when the puck is going towards the opposing team's zone	all	keep
8	games_played	Number of games played.	all	keep
9	gameScore	Game Score rating as designed by @domluszczyszyn	all	keep
10	I_F_blockedShotAttempts	Blocked shot attempts. The number of shot attempts a player has taken that were blocked by the opponent's team	forward	keep
11	I_F_dZoneGiveaways	Giveaways in the team's defensive zone	forward	keep
12	I_F_dZoneShiftEnds	Number of player's shifts that end with a defensive zone faceoff for the oncoming players	all	keep
13	I_F_dZoneShiftStarts	Number of defensive zone face-off shift starts a player had	all	keep
14	I_F_faceOffsWon	Number of faceoffs the player has won	forward	keep
15	I_F_flurryAdjustedxGoals	Flurry Adjusted Expected Goals.	all	keep
16	I_F_flurryScoreVenueAdjustedxGoals	Flurry and Score and Venued Adjusted xGoals	forward	keep
17	I_F_flyShiftEnds	Number of player's shifts that end on the fly	forward	keep
18	I_F_flyShiftStarts	Number of shift starts on the fly a player had	forward	keep
19	I_F_freeze	Puck freezes after a player's shots. The number of puck freezes by goalies after the player's unblocked shot attempts.	forward	keep
20	I_F_giveaways	Number of giveaways the player has given to other team	defenseman	keep
21	I_F_goals	Goals	forward	keep
22	I_F_highDangerGoals	Goals from high danger shots	forward	keep
23	I_F_highDangerShots	High Danger Shots (Higher than 20% xGoal Value)	forward	keep
24	I_F_highDangerxGoals	Sum of expected goals from high danger shots	forward	keep
25	I_F_hits	Number of hits the player has given	forward	keep
26	I_F_lowDangerGoals	Goals from low danger shots	forward	keep
27	I_F_lowDangerShots	Low danger shots (<8% xGoal value)	forward	keep
28	I_F_lowDangerxGoals	Sum of expected goals from low danger shots	forward	keep
29	I_F_mediumDangerGoals	Goals from medium danger shots	forward	keep
30	I_F_mediumDangerShots	Medium danger shots (Between 8% and 20% xGoal Value)	forward	keep

#	Column Name	Description	Type	Decision
31	I_F_mediumDangerxGoals	Sum of expected goals from medium danger shots	forward	keep
32	I_F_missedShots	Missed shots. Shots that aren't blocked but don't hit the net	forward	keep
33	I_F_neutralZoneShiftEnds	Number of player's shifts that end with a neutral zone faceoff for the oncoming players	forward	keep
34	I_F_neutralZoneShiftStarts	Number of neutral zone face-off shift starts a player had	forward	keep
35	I_F_oZoneShiftEnds	Number of player's shifts that end with an offensive zone faceoff for the oncoming players	forward	keep
36	I_F_oZoneShiftStarts	Number of offensive zone face-off shift starts a player had	forward	keep
37	I_F_penalityMinutes	Number of penalty minutes the player has received	forward	keep
38	I_F_playContinuedInZone	Number of times the play continues in the offensive zone after the player's shot besides an immediate rebound shot. This is proxied by another event happening in the zone after the shot (such as a hit, takeaway, etc) without any events outside of the zone happening inbetween and all the same players for both teams are still on the ice as they were for the original shot	forward	keep
39	I_F_playContinuedOutsideZone	Number of times the play goes outside the offensive zone after the player's shot.	forward	keep
40	I_F_playStopped	Number of times the play is stopped after shots for reasons other than the goalie freezing the puck, such as the puck going over the glass or a dislodged net.	forward	keep
41	I_F_points	Goals + Assists	forward	keep
42	I_F_primaryAssists	Primary Assists the player has received on teammates' goals	forward	keep
43	I_F_reboundGoals	Goals from rebound shot attempts	forward	keep
44	I_F_rebounds	Rebound shot attempts. These must occur within 3 seconds of a previous shot.	forward	keep
45	I_F_reboundxGoals	Expected Goal on rebound shots	forward	keep
46	I_F_savedShotsOnGoal	Number of the player's unblocked shots that were saved by the goalie	forward	keep
47	I_F_savedUnblockedShotAttempts	Number of the player's unblocked shots that were saved by the goalie or missed the net	forward	keep
48	I_F_scoreAdjustedShotsAttempts	Shot attempts adjusted for score and venue	forward	keep
49	I_F_scoreAdjustedUnblockedShotAttempts	Unblocked shot attempts adjusted for score and venue	forward	keep
50	I_F_scoreVenueAdjustedxGoals	Score and Venue Adjusted xGoals. Gives more credit to away teams and teams with large leads when they get an xGoal.	forward	keep
51	I_F_secondaryAssists	Secondary Assists the player has received on teammates' goals	forward	keep
52	I_F_shifts	Number of shifts a player had	forward	drop
53	I_F_shotAttempts	Shot attempts. Includes shots on goal, missed shots, and blocked shot attempts	forward	keep
54	I_F_shotsOnGoal	Shots on goal. Does not include shots that miss the net or blocked shots	forward	keep
55	I_F_takeaways	Number of takeaways the player has taken from opponents	forward	keep
56	I_F_unblockedShotAttempts	All shot attempts that weren't blocked	forward	keep
57	I_F_xFreeze	Expected puck freezes after shots. The expected number of puck freezes by the goalie after the player's unblocked shot attempts.	forward	keep
58	I_F_xGoals	Expected Goals.	forward	keep
59	I_F_xGoals_with_earned_rebounds	xGoals With Earned Rebounds. Also known as 'Created Expected Goals': Expected Goals of non-rebound shots + xGoals of xRebounds of all shots. This metric gives credit to the player that created the original shot, opposed to the player getting the rebound. See http://moneypuck.com/about.htm#xRebounds for more info	forward	keep
60	I_F_xGoals_with_earned_rebounds_scoreAdjusted	Score adjusted xGoals With Earned Rebounds	forward	keep

#	Column Name	Description	Type	Decision
61	I_F_xGoals_with_earned_rebounds_scoreFlurryAdjusted	Score and flurry adjusted xGoals With Earned Rebounds	forward	keep
62	I_F_xGoalsFromActualReboundsOfShots	Expected Goals from actual rebounds shots of player's shots.	forward	keep
63	I_F_xGoalsFromxReboundsOfShots	Expected Goals from Expected Rebounds of player's shots. Even if a shot does not actually generate a rebound, if it's a shot that is likely to generate a rebound the player is credited with xGoalsFromxRebounds	forward	keep
64	I_F_xOnGoal	Expected number of unblocked shot attempts that are expected to be a shot on goal (not miss the net) given the context (distance, situation, etc) they were taken from. This assumes the player has average shooting talent.	forward	keep
65	I_F_xPlayContinuedInZone	Expected number of times the play continues in the offensive zone after the player's shot besides an immediate rebound shot. This is proxied by another event happening in the zone after the shot (such as a hit, takeaway, etc) without any events outside of the zone happening inbetween and all the same players for both teams are still on the ice as they were for the original shot	forward	keep
66	I_F_xPlayContinuedOutsideZone	Expected number of times the play goes outside the offensive zone after the player's shot.	all	keep
67	I_F_xPlayStopped	Expected number of times the play is stopped after shots for reasons other than the goalie freezing the puck, such as the puck going over the glass or a dislodged net.	forward	keep
68	I_F_xRebounds	Expected Rebounds. The expected number of rebound shots generated from the player's unblocked shot attempts.	forward	keep
69	icetime	Ice time in seconds	all	keep
70	iceTimeRank	Rank of the player's ice time in a given game. Forwards and Defensemen are ranked seperately. 1 means the player got more icetime in the game than any other forward/D.	all	keep
71	OffIce_A_shotAttempts	See above	defenseman	keep
72	OffIce_A_xGoals	See above	defenseman	keep
73	offIce_corsiPercentage	Off Ice Shot Attempts For / (Off Ice Shot Attempts For + Off Ice Shot Attempts Against)	forward	keep
74	OffIce_F_shotAttempts	See above	forward	keep
75	OffIce_F_xGoals	See above	forward	keep
76	offIce_fenwickPercentage	Off Ice Unblocked Shot Attempts For / (Off Ice Unlbocked Shot Attempts For + Off Ice Unbloked Shot Attempts Against)	forward	keep
77	offIce_xGoalsPercentage	Off Ice xGoals For / (Off Ice xGoals For + Off Ice xGoals Against)	forward	keep
78	OnIce_A_blockedShotAttempts	See above	defenseman	keep
79	OnIce_A_flurryAdjustedxGoals	See above	defenseman	keep
80	OnIce_A_flurryScoreVenueAdjustedxGoals	See above	defenseman	keep
81	OnIce_A_goals	See above	defenseman	keep
82	OnIce_A_highDangerGoals	See above	defenseman	keep
83	OnIce_A_highDangerShots	See above	defenseman	keep
84	OnIce_A_highDangerxGoals	See above	defenseman	keep
85	OnIce_A_lowDangerGoals	See above	defenseman	keep
86	OnIce_A_lowDangerShots	See above	defenseman	keep
87	OnIce_A_lowDangerxGoals	See above	defenseman	keep
88	OnIce_A_mediumDangerGoals	See above	defenseman	keep
89	OnIce_A_mediumDangerShots	See above	defenseman	keep
90	OnIce_A_mediumDangerxGoals	See above	defenseman	keep

#	Column Name	Description	Type	Decision
91	OnIce_A_missedShots	See above	defenseman	keep
92	OnIce_A_reboundGoals	See above	defenseman	keep
93	OnIce_A_rebounds	See above	defenseman	keep
94	OnIce_A_reboundxGoals	See above	defenseman	keep
95	OnIce_A_scoreAdjustedShotsAttempts	See above	defenseman	keep
96	OnIce_A_scoreAdjustedUnblockedShotAttempts	See above	defenseman	keep
97	OnIce_A_scoreVenueAdjustedxGoals	See above	defenseman	keep
98	OnIce_A_shotAttempts	See above	defenseman	keep
99	OnIce_A_shotsOnGoal	See above	defenseman	keep
100	OnIce_A_unblockedShotAttempts	See above	defenseman	keep
101	OnIce_A_xGoals	See above	defenseman	keep
102	OnIce_A_xGoals_with_earned_rebounds	See above	defenseman	keep
103	OnIce_A_xGoals_with_earned_rebounds_scoreAdjusted	See above	defenseman	keep
104	OnIce_A_xGoals_with_earned_rebounds_scoreFlurryAdjusted	See above	defenseman	keep
105	OnIce_A_xGoalsFromActualReboundsOfShots	See above	defenseman	keep
106	OnIce_A_xGoalsFromxReboundsOfShots	See above	defenseman	keep
107	OnIce_A_xOnGoal	"On Ice Against" version of xOnGoal stat. Gives 'credit' to all of the opposing team's player on ice for the event.	defenseman	keep
108	onIce_corsiPercentage	On Ice Shot Attempts For / (On Ice Shot Attempts For + On Ice Shot Attempts Against)	all	keep
109	OnIce_F_blockedShotAttempts	See above	forward	keep
110	OnIce_F_flurryAdjustedxGoals	See above	forward	keep
111	OnIce_F_flurryScoreVenueAdjustedxGoals	See above	forward	keep
112	OnIce_F_goals	See above	forward	keep
113	OnIce_F_highDangerGoals	See above	forward	keep
114	OnIce_F_highDangerShots	See above	forward	keep
115	OnIce_F_highDangerxGoals	See above	forward	keep
116	OnIce_F_lowDangerGoals	See above	forward	keep
117	OnIce_F_lowDangerShots	See above	forward	keep
118	OnIce_F_lowDangerxGoals	See above	forward	keep
119	OnIce_F_mediumDangerGoals	See above	forward	keep
120	OnIce_F_mediumDangerShots	See above	forward	keep
121	OnIce_F_mediumDangerxGoals	See above	forward	keep
122	OnIce_F_missedShots	See above	forward	keep
123	OnIce_F_reboundGoals	See above	forward	keep
124	OnIce_F_rebounds	See above	forward	keep
125	OnIce_F_reboundxGoals	See above	forward	keep
126	OnIce_F_scoreAdjustedShotsAttempts	See above	forward	keep
127	OnIce_F_scoreAdjustedUnblockedShotAttempts	See above	forward	keep
128	OnIce_F_scoreVenueAdjustedxGoals	See above	forward	keep
129	OnIce_F_shotAttempts	See above	forward	keep
130	OnIce_F_shotsOnGoal	See above	forward	keep

#	Column Name	Description	Type	Decision
131	OnIce_F_unblockedShotAttempts	See above	forward	keep
132	OnIce_F_xGoals	See above	forward	keep
133	OnIce_F_xGoals_with_earned_rebounds	See above	forward	keep
134	OnIce_F_xGoals_with_earned_rebounds_scoreAdjusted	See above	forward	keep
135	OnIce_F_xGoals_with_earned_rebounds_scoreFlurryAdjusted	See above	forward	keep
136	OnIce_F_xGoalsFromActualReboundsOfShots	See above	forward	keep
137	OnIce_F_xGoalsFromxReboundsOfShots	See above	forward	keep
138	OnIce_F_xOnGoal	"On Ice For" version of xOnGoal stat. Gives credit to all of the team's players on the ice for the event, opposed to just the player who did the event	forward	keep
139	onIce_fenwickPercentage	On Ice Unblocked Shot Attempts For / (On Ice Unlbocked Shot Attempts For + On Ice Unbloked Shot Attempts Against)	forward	keep
140	onIce_xGoalsPercentage	On Ice xGoals For / (On Ice xGoals For + On Ice xGoals Against)	forward	keep
141	penalityMinutes	Number of penalty minutes the player has received	all	keep
142	penalityMinutesDrawn	Number of penalty minutes the player has drawn	all	keep
143	penalties	Number of penalties the player has received. Both majors and minors both count as '1'	all	keep
144	penaltiesDrawn	Number of penalties the player has drawn	all	keep
145	playerId	Unique ID for each player assigned by the NHL	var	drop
146	season	Starting year of the season. For example 2018 for the 2018-2019 season	var	drop
147	shifts	Number of shifts a player had	all	drop
148	shotsBlockedByPlayer	Number of shot attempts blocked by the player	defenseman	keep
149	situation	5on5 for normal play, 5on4 for a normal powerplay, 4on5 for a normal PK. 'Other' includes everything else: two man advantage, empty net, 4on3, etc. 'all' includes all situations	var	variable
150	timeOnBench	Amount of time the player has been on the bench for. (in seconds)	all	drop
151	xGoalsAgainstAfterShifts	xGoals the opposing team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to identify players who shift off instead of backchecking	all	keep
152	xGoalsForAfterShifts	xGoals the player's team gets between 1 and 5 seconds after the player has shifted off on the fly. Meant to give credit to give players who leave the ice when the puck is going towards the opposing team's zone	all	keep

Appendix D - Supervised Learning Additional Information

Figure D1 - Random Forest Model Hyperparameters

Feature	Explanation
n_estimators	the number of decision trees that comprise the ensemble
max_depth	how deep each decision tree can go
min_samples_split	the minimum number of samples needed to split a node (and create a new branch)
min_samples_leaf	the minimum number of samples needed to be a leaf node (an end of the decision tree)

Figure D2 - Model Selection

	Model	MAE Mean	MAE Std	MedAE Mean	MedAE Std	MSE Mean	MSE Std	RMSE Mean	RMSE Std	R2 Mean	R2 Std
0	Linear Regression	5.851294	0.612683	5.435720	0.827966	53.190523	13.547260	7.234544	0.922980	-1.685779	0.425098
1	Lasso	3.703534	0.299376	2.805215	0.571818	21.369379	4.164365	4.600506	0.452466	-0.081730	0.040584
2	Ridge	3.849031	0.231197	3.260387	0.247680	22.479429	3.212741	4.729194	0.337871	-0.155408	0.134491
3	Random Forest	3.731200	0.342672	3.196000	0.351332	21.810936	5.379588	4.643641	0.563798	-0.108586	0.091713
4	Decision Tree	4.652605	0.471521	3.400000	0.374166	39.156303	8.771702	6.414091	0.537287	-1.095943	0.524471
5	Support Vector Machine	3.526994	0.550275	2.423240	0.361504	23.743542	7.932863	4.804591	0.812062	-0.173762	0.201953

Figure D3: Final Random Forest Model Performance

	Metric	Value
0	MAE	3.515349
1	Median Absolute Error	4.060823
2	RMSE	0.047528
3	R2	3.530000

Figure D4 - Top and Bottom 10 Contributing Features

	Feature	Importance
0	goalsFor	0.046241
1	giveawaysAgainst	0.034400
2	reboundsFor	0.032412
3	xPlayContinuedOutsideZoneFor	0.032078
4	mediumDangerGoalsAgainst	0.028836
5	missedShotsAgainst	0.027146
6	mediumDangerGoalsFor	0.025213
7	highDangerGoalsFor	0.024901
8	lowDangerGoalsFor	0.024714
9	hitsAgainst	0.022115

	Feature	Importance
88	xPlayStoppedAgainst	0.001936
89	scoreVenueAdjustedxGoalsAgainst	0.001908
90	xPlayContinuedInZoneAgainst	0.001896
91	flurryAdjustedxGoalsAgainst	0.001864
92	scoreVenueAdjustedxGoalsFor	0.001734
93	totalShotCreditAgainst	0.001496
94	unblockedShotAttemptsFor	0.001483
95	xOnGoalAgainst	0.001346
96	xGoalsAgainst	0.001183
97	scoreAdjustedTotalShotCreditFor	0.000791

Figure D5: Supervised Learning Dataset Correlation Heatmap to Demonstrate Multicollinearity