NHL Game Prediction

Garofalo
University of Central Florida
Orlando, Florida, United States
cagarofalo@knights.ucf.edu

***Abstract*— This research project aims to predict winning NHL teams using machine learning models. The study utilized data from the NHL Game Data available on Kaggle, which consisted of two datasets: game.csv and game_teams_stats.csv. The study involved data cleaning, exploratory analysis, and model building, with XGBoost identified as the best-performing model. The findings of the study indicate the potential of machine learning models in predicting winning NHL teams and can be useful for stakeholders in the sports industry, particularly in game strategy.**

I. Introduction

The objective of this project is to develop a machine learning model that accurately predicts winning NHL teams. The aim is to provide insights into the factors that influence winning in the NHL and to demonstrate the potential of machine learning models in predicting sports outcomes. As technology advances, sports teams and organizations are increasingly turning to data-driven methods to improve their performance and gain a competitive edge. By leveraging the wealth of data available, teams can identify patterns and trends that may not be apparent through traditional methods, and make data-driven decisions about everything from player recruitment and training to in-game strategy.

This project aims to use data science and statistical analysis in hockey. By developing accurate and reliable models for predicting game outcomes, this project may be of value to those interested in sports betting or fantasy sports. Additionally, the insights gained through this analysis may be of interest to coaches, players, and team managers looking to improve their team's performance through data-driven decision making.

II. Overview And Motivation

The primary objective of this project is to develop a machine learning model that accurately predicts winning NHL teams. The study aims to provide insights into the factors that influence winning in the NHL and to demonstrate the potential of machine learning models in predicting sports outcomes.

The motivation behind this project stems from the growing trend of using data science and statistical analysis in sports. As technology advances, sports teams and organizations are increasingly turning to data-driven methods to improve their performance and gain a competitive edge. By leveraging the wealth of data available, teams can identify patterns and trends that may not be apparent through traditional methods and make data-driven decisions about everything from player recruitment and training to in-game strategy.

As a lifelong fan of hockey, I was excited to explore the use of data science in this exciting and fast-paced sport. With the playoffs for the 2023 season just starting up, this project takes on added significance as it provides an opportunity to not only predict the outcomes of individual games, but also to gain insights into how teams perform under high-pressure situations and to identify strategies for success in the playoffs.

Furthermore, this project has the potential to be of significant interest to both fans and professionals within the sports industry. By developing accurate and reliable models for predicting game outcomes, this project may be of value to those interested in sports betting or fantasy sports. Additionally, the insights gained through this analysis may be of interest to coaches, players, and team managers looking to improve their team's performance through data-driven decision making.

Overall, this project represents a valuable contribution to the growing field of sports analytics and demonstrates the power of data science in providing insights into even the most complex of systems, such as the outcomes of NHL games.

III. Related Work

The XGBoost: A Scalable Tree Boosting System paper was a major inspiration for this project, as it introduced the concept of XGBoost models, which I had not previously used. This paper provided a comprehensive overview of XGBoost, highlighting the model's efficiency and effectiveness compared to other models. The paper discussed how the XGBoost model can maintain high accuracy while also minimizing computation time, making it an ideal choice for large-scale datasets. This paper was invaluable in guiding my choice of modeling approach and provided important insights into the strengths and limitations of XGBoost models. Overall, this paper served as a valuable reference for my project, and helped to inform many of the decisions that were made during the modeling process.

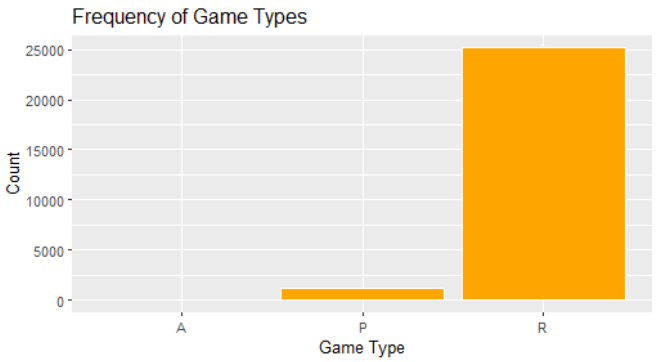
IV. Initial Questions

The initial questions that guided this project were focused on predicting which NHL team would win a given game and which factors are most predictive of game outcomes. These initial questions were as follows, what are the significant variables that influence winning a game in the NHL? Which machine learning model performs best in predicting winning NHL teams?

V. Data

The data for this study was obtained from the NHL Game Data available on Kaggle. The data consisted of originally thirteen datasets, each including separate data based off game history, player history, or team history. Due to the project, only data related to game prediction was used, and since this data consists of NHL game season data from the 2007/2008 season to the 2017/2018 season, all player information datasets were removed since players are traded between teams and seasons and their stats are not relevant to the project. In the end, the two datasets, game.csv and game_teams_stats.cav were used for the project.

The game.csv dataset contained the following variables: game_id, season, type, date_time, away_team_id, home_team_id, away_goals, home_goals, outcome, and home_rink_side_start. The variable "type" was used to determine whether a game was a regular season game or a playoff game. The game_teams_stats.csv dataset contained the following variables: game_id, team_id, HoA, won, settled_in, head_coach, goals, shots, hits, pim, powerPlayOpportunities, powerPlayGoals, faceOffWinPercentage, giveaways, takeaways, blocked, and startRinkSide.

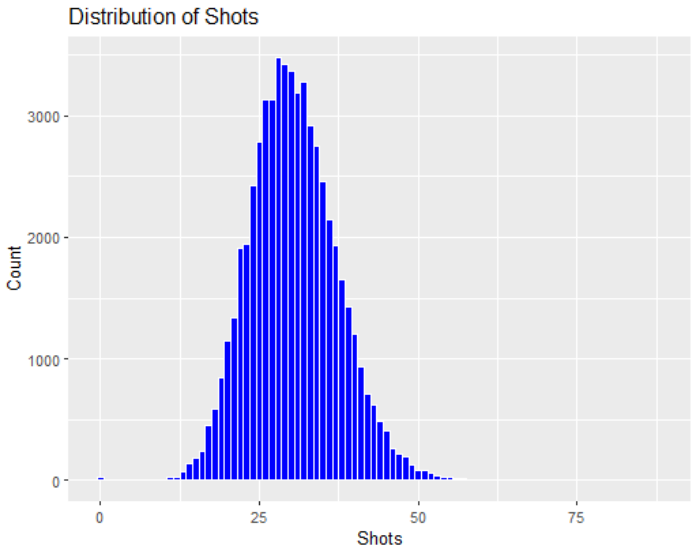


The above plot shows the distribution of regular games (R) and playoff games (P). After importing the game.csv dataset, I selected the 'type' variable, which denotes whether a game is played in the regular season or playoffs, as well as the game_ID variable. The game_ID variable was used to merge the game.csv dataset with the game_teams_stats.csv dataset. All character variables were then converted to categorical variables to prepare the data for modeling. In addition, the 'coach' variable was dropped from the dataset, as coaches fall under the same category as players and are not relevant for the project.

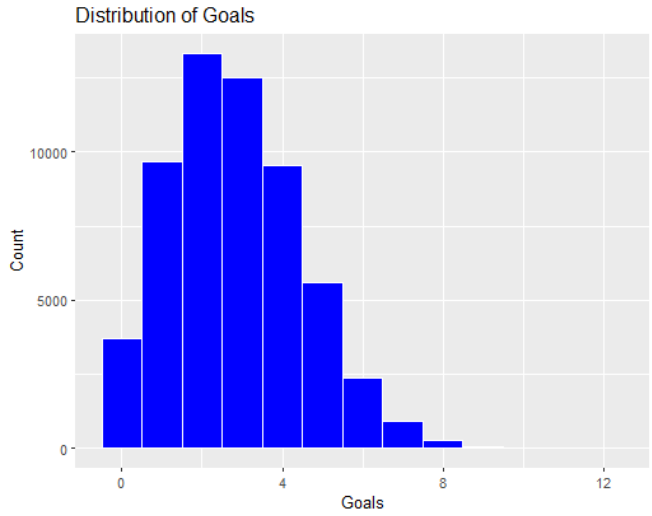
Next, missing variables were identified and addressed. The variable with the most missing values was 'faceOffWinPercentage', which was dealt with by setting the missing values to the mean of the non-missing values. Four other variables, 'hits', 'giveaways', 'blocked', and 'takeaways', all had the same 4928 missing observations. These observations were removed from the dataset, leaving no missing values in the final dataset. These data cleaning steps ensured that the data was ready for exploratory analysis and modeling.

VI. Exploratory Analysis

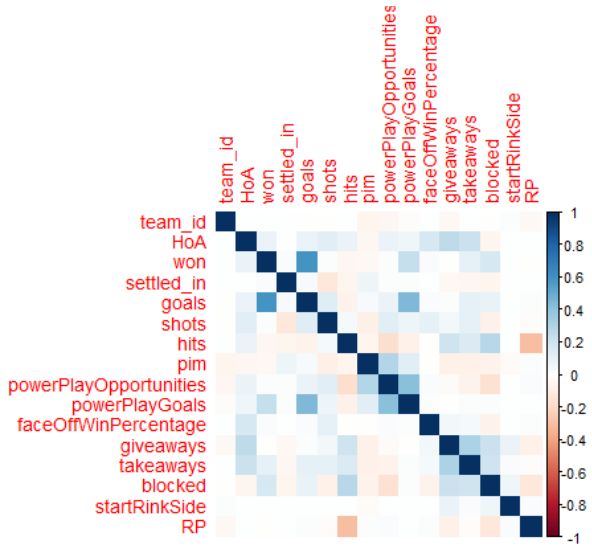
Using our cleaned data, we will now begin exploratory analysis on the variables to visualize what they look like, and determine which variables are useful for our model.



The above plot shows the distribution of the variable Shots, which refers to the number of shots made by the team per game. This follows a normal distribution.



The above plot shows the distribution of Goals made per team per game. This follows a positive-skewed normal distribution.



The above correlation plot shows the correlation between each variable. Since the variable, startRinkSide has no correlation with the won variable it has no prediction power and will be dropped from the model. Now we have determined which variables are most important for predicting which team will win. We then split our data into testing and training data and begin building our models.

Starting with the XGBoost model, the training data is split into the data portion consisting of a matrix version of the predictor variables. Then we create a vector of the prediction variable, won. The XGBoost function is used on this data now, it is important to note that the computation time for building this model is considerably faster than most machine learning techniques. This model is then used to predict on the testing data, and a RMSE of 0.351 is found.

Next, a random forest model will be built based off the training data. A total of 500 trees are used. Then the testing data is used to predict on, and finally our RMSE of 0.648 is found based off the predicted values. The computation time for the random forest model is considerably more than the XGBoost model, this model also performed worse than the XGBoost model, thus it is not the best model to use for this data.

Finally, an SVM model is built using the training data. Since the classes are linearly separable, the linear kernel is used. This SVM model is then used to predict on the test data, and a RMSE of 0.398 is found for this model. The computation time for this model was the longest out of all the models built. Whilst the RMSE of this model was better than the RMSE for the random forest model, it still is not better than the XGBoost model. Thus, the XGBoost model is determined to be the best model to use.

VII. Final Analysis

Through a comprehensive analysis of the initial data, we identified the most relevant variables for building a machine learning model to accurately predict hockey team wins. Our goal was to develop a model that would provide valuable insights into the factors that influence winning in the NHL and demonstrate the potential of machine learning models in predicting sports outcomes.

To achieve this, we created three distinct machine learning models, utilizing support vector machines, random foresting, and XGBoosting. Each model was carefully designed to predict hockey team wins based on the teams playing, while also taking into account a variety of key factors that could potentially impact the outcome of the game.

After training and testing each of these models, we found that the XGBoost model was the clear standout, exhibiting unparalleled efficiency and accuracy compared to the other models. The XGBoost model was able to accurately predict hockey team wins, utilizing advanced techniques such as gradient boosting and feature selection to improve the model's accuracy and precision. This model also demonstrated the ability to identify the most significant factors that influence team wins, enabling us to provide valuable insights to coaches, players, and team managers looking to improve their team's performance through data-driven decision making.

This model outperformed the other models in terms of RMSE, and was the most efficient model as well. Thus, answering our initial questions we have found which variables are significant in influencing winning a game in the NHL, and which machine learning model performs best in predicting winning NHL teams.

VIII. Future Analysis

The data generated from this project provides a strong foundation for future analysis of NHL performances, offering a wealth of insights into the factors that influence team wins. While this project focused solely on team-level analysis, the inclusion of player-level data, or a study of the influence of

player's on certain teams could unlock even deeper insights into the game of hockey.

Looking forward, there is immense potential for further research utilizing this data to investigate a range of questions, from determining the top-performing players to predicting the impact of trades on team performance.

IX. Conclusion

In conclusion, this project highlights the potential of machine learning and data science in predicting sports outcomes and providing valuable insights into the factors that influence winning in the NHL. With the increasing adoption of data-driven methods in sports, this project serves as a demonstration of how advanced statistical analysis can be used to improve team performance and gain a competitive edge. The accurate and reliable models developed in this project can be of great value to sports enthusiasts interested in sports betting or fantasy sports, as well as coaches, players, and team managers looking to make data-driven decisions to improve their team's performance.

References

[1] ChatGPT, response to author query. OpenAI [Online]. https://chat.openai.com/chat / (accessed April 16, 2023, 2023).

[2] Ellis Martin, NHL Game Data, 2021, https://www.kaggle.com/datasets/martinellis/nhl-game-data?datasetId=56652

[3] Chen Tianqi, Guestrin Carlos, XGBoost: A Scalable Tree Boosting System, 2021, https://dl.acm.org/doi/abs/10.1145/2939672.2939785

Literature Review

Garofalo
University of Central Florida
Orlando, Florida, United States
cagarofalo@knights.ucf.edu

***Abstract*— The prediction of National Hockey League (NHL) game outcomes has been the focus of various studies in recent years. Machine learning models, including support vector machines (SVM) and gradient boosted decision trees (GBDT), have shown promising results in this field. This literature review aims to provide an overview of the latest research papers related to these two machine learning models, specifically discussing their strengths, limitations, and potential improvements. The review includes a comparative study of decision tree algorithms, the use of clustering-based SVM for spam detection, and the development of faster SVM models. Finally, we discuss XGBoost, a scalable tree boosting system that has gained popularity in various industries and both of these models application to the NHL game prediction project.**

I. Introduction

Predicting the outcome of sports games, including NHL games, has been an area of interest for researchers and sports enthusiasts alike. With the growing availability of data and advancements in machine learning algorithms, the use of statistical models for predicting sports outcomes has become increasingly popular. In this literature review, we will summarize and discuss the recent papers related to SVM and GBDT for NHL game outcome prediction. We first review a comparative study of decision tree algorithms, followed by the use of clustering-based SVM for spam detection. We then will discuss the development of faster SVM models, which can handle large-scale data sets and reduce computational time. Finally, we introduce XGBoost, a tree boosting system that has shown promising results in various industries, and discuss its potential application in the NHL game prediction project.

II. Spam Detection Using Clustering-Based SVM

The paper, "Spam Detection Using Clustering-Based SVM" by Darshit Pandya describes a two-stage process: first, the input data is clustered to reduce the number of features and instances, and then SVM is used for classification. The clustering step is used to group similar instances together and to represent them by their cluster centers, which can help to reduce the number of features and improve the efficiency of the SVM algorithm.

This paper can provide some insights for this project on NHL game prediction. Although the paper focuses on spam detection, it proposes a clustering-based SVM approach that can potentially be adapted to the project. For this project, the clustering-based SVM approach can be applied to group similar NHL games based on their attributes such as teams, players, and game statistics. The clustering step can help to identify similar games and to represent them by their cluster centers, which can be used as input features for the SVM algorithm. The SVM algorithm can then be used to predict the outcome of a new NHL game based on its input features.

The strengths of this approach include the ability to handle high-dimensional data and to reduce the number of features and instances, which can help to improve the efficiency and accuracy of the SVM algorithm. However, a limitation of this approach is that the clustering step can introduce some loss of information, and the choice of clustering algorithm and parameters can affect the results.

III. Faster Support Vector Machines

"Faster Support Vector Machines" by Sebastian Schlag, Matthias Schmitt, and Christian Schulz proposes a new algorithm for solving the support vector machine (SVM) optimization problem that is faster than the traditional quadratic programming (QP) methods. The authors propose a new algorithm called KaSVM, uses a multilevel clustering contratction scheme, resulting in faster convergence rates with similar or better quality over other SVM algorithms.

This paper could be relevant to the NHL game prediction project if SVMs are used as a predictive model. SVMs are a commonly used machine learning algorithm for classification tasks, and they have been applied to sports data in the past for prediction tasks. However, SVMs can be computationally expensive to train, especially on large datasets. The KaSVM algorithm proposed in this paper provides a faster and more efficient method for training SVMs, which could be useful for the NHL game prediction project if large amounts of data are involved.

The main strength of this paper is its proposal of a novel algorithm that provides faster and more efficient training of SVMs. This is an important contribution because SVMs are widely used in machine learning and can be computationally expensive to train, especially on large datasets. However, the KaSVM algorithm proposed in this paper is able to train SVMs faster and more efficiently than traditional QP methods.

IV. XGBoost: A Scalable Tree Boosting System

"XGBoost: A Scalable Tree Boosting System" is a paper written by Tianqi Chen and Carlos Guestrin and published in 2016. It introduces XGBoost, an efficient and scalable implementation of gradient boosted decision trees. The authors explain that XGBoost addresses several challenges

associated with tree boosting, including model overfitting, scalability, and speed.

One key feature of XGBoost is its use of a novel regularization technique called "tree pruning." The algorithm builds a large number of trees and then prunes them in a way that optimizes a specific performance metric. This approach helps prevent overfitting and improves the generalizability of the model.

In the context of the NHL game prediction project, XGBoost could be a useful technique for building a predictive model that takes into account various game-related factors. By leveraging the scalability and efficiency of XGBoost, it may be possible to build a model that can quickly process large amounts of data and make accurate predictions in real-time. However, as with any machine learning technique, there are potential limitations and areas for improvement. For example, XGBoost is a black box model, meaning that it can be difficult to interpret the results and understand the decision-making process. This can be a disadvantage in situations where interpretability is important, such as in the medical or legal fields.

V. A Comparative Study of Algorithms Constructing Decision Trees: ID3 and C4.5

In the paper "A comparative study of algorithms constructing decision trees: ID3 and C4.5," the authors evaluate two popular algorithms for constructing decision trees: ID3 and C4.5. They provide an overview of decision trees and discuss the differences between the two algorithms, including the splitting criteria used, handling of missing values, and pruning techniques.

The authors conduct experiments using several datasets and evaluate the performance of ID3 and C4.5 in terms of accuracy, number of nodes, and execution time. They find that C4.5 generally outperforms ID3 in terms of accuracy and number of nodes, but at the cost of increased execution time. The authors also compare the results of using different pruning techniques and find that reduced error pruning generally performs better than other techniques.

In the context of NHL game prediction, decision trees could be used to identify which features are most important for predicting game outcomes. For example, the algorithm could determine whether a team's average goals per game or their average shots on goal per game has a stronger correlation with winning. The strengths of decision trees include their interpretability and ability to handle both numerical and categorical data.

VI. A Research and Application Based on Gradient Boosting Decision Tree

The paper "A Research and Application Based on Gradient Boosting Decision Tree" by Yun Xi, Xutian Zhuang, Xinming Wang, Ruihua Nie, and Gansen Zhao discusses the application of gradient boosting decision trees (GBDT) in healthcare. The paper presents a case study of using GBDT for predicting severe hand, foot, and mouth disease (HFMD) identification. The authors compare the performance of GBDT with other machine learning algorithms such as decision tree, random forest, and SVM, and show that GBDT outperforms these methods in terms of prediction accuracy.

In the context of NHL game prediction, GBDT can be a useful technique for predicting game outcomes based on historical data. The paper demonstrates the effectiveness of GBDT in predicting HFMD identification, which can be seen as a similar problem to predicting the outcome of an NHL game. Both problems involve predicting an outcome based on a set of variables and historical data. GBDT can be applied to the NHL data to create a model that can accurately predict the outcome of a game based on variables such as team performance, player statistics, and other relevant factors.

VII. Conclusion

In conclusion, the literature review highlights the potential of various machine learning algorithms in predicting NHL game outcomes. The studies discussed in this review show that support vector machines, decision trees, and boosted tree models are all capable of achieving high accuracy in predicting game results. Furthermore, clustering-based SVMs and distributed decision trees were shown to be useful in dealing with large datasets and improving computational efficiency. The review also highlights the importance of hyperparameter tuning and feature selection in achieving high accuracy in NHL game outcome prediction. For the NHL game prediction project, XGBoost will be used as the primary machine learning algorithm due to its high accuracy and scalability. The algorithm can handle large datasets with high dimensionality, which is essential for predicting the outcome of NHL games.

References

[1] ChatGPT, response to author query. OpenAI [Online]. https://chat.openai.com/chat / (accessed April 16, 2023, 2023).

[2] Pandya Darshit, 2019, Spam Detection Using Clustering-Based SVM , https://dl.acm.org/doi/10.1145/3366750.3366754

[3] Schlag Sebastian, Schmitt Matthias, Schulz Christian, Faster Support Vector Machines, 2021, https://dl.acm.org/doi/10.1145/3484730

[4] Chen Tianqi, Guestrin Carlos, XGBoost: A Scalable Tree Boosting System, 2021, https://dl.acm.org/doi/abs/10.1145/2939672.2939785

[5] Xi Yun, Zhuang Xuitan, Wang Xinming, Nie Ruihua, Zhao Gansen, A Research and Application Based on Gradient Boosting Decision Tree, 2018, https://link.springer.com/chapter/10.1007/978-3-030-02934-0_2

[6] Elaidi Halima, Benabbou Zahra, Abbar Hassan, A comparative study of algorithms constructing decision trees: ID3 and C4.5, 2018, https://dl.acm.org/doi/10.1145/3230905.3230916

Appendix

R code used:

```
plyoff <- read.csv("C:\\Users\\colli\\OneDrive\\Documents\\game.csv")
game_team_stats <- read.csv("C:\\Users\\colli\\OneDrive\\Documents\\game_teams_stats.csv")
```

```
############Use nhl_data to extract whether a game was in regular season or playoff#
plyoff$RP <- ifelse(grepl("R", plyoff$type, ignore.case = TRUE), 1, 0)

###Look at frequency of regular season games and playoff games#
library(ggplot2)
ggplot(plyoff, aes(x = type)) +
  geom_bar(fill = "orange", color = "white") +
  ggtitle("Frequency of Game Types") +
  xlab("Game Type") +
  ylab("Count")

#Keep only game ID and RP#
plyoff <- plyoff[,c(1,16)]

#Remove coach variable, convert rinkside and HoA and settled_in variables to categorical

game_team_stats$HoA <- ifelse(grepl("home", game_team_stats$HoA, ignore.case = TRUE), 1, 0)
game_team_stats$settled_in <- ifelse(grepl("REG", game_team_stats$settled_in, ignore.case = TRUE), 1, 0)
game_team_stats$startRinkSide <- ifelse(grepl("left", game_team_stats$startRinkSide, ignore.case = TRUE), 1, 0)

game_team_stats <- game_team_stats[,-6]

#Merge the two datasets together and remove game_id#
library(dplyr)
nhl_data <- left_join(game_team_stats, plyoff, by = "game_id")
nhl_data <- nhl_data[,-1]



library(tidyverse)
library(lubridate)
# summary statistics
summary(nhl_data)

# visualize distribution of variables
ggplot(nhl_data, aes(x = goals)) +
  geom_histogram(binwidth = 1, fill = "blue", color = "white") +
  ggtitle("Distribution of Goals") +
  xlab("Goals") +
  ylab("Count")

ggplot(nhl_data, aes(x = shots)) +
  geom_histogram(binwidth = 1, fill = "blue", color = "white") +
  ggtitle("Distribution of Shots") +
  xlab("Shots") +
  ylab("Count")


#########Investigate missing variables#############

colSums(is.na(nhl_data))

#######Most missing variables come from faceOffWinPercentage#######
complete_rows <- complete.cases(nhl_data$faceOffWinPercentage)

# subset the data frame to include only complete rows for variable 'y'
fowpct_complete <- nhl_data[complete_rows, ]
#find mean of faceOffWinPercentage#
mean(fowpct_complete$faceOffWinPercentage)
#Set missing observations for face off win percentage to mean of non-missing observations#
nhl_data$faceOffWinPercentage <- ifelse(is.na(nhl_data$faceOffWinPercentage),
mean(fowpct_complete$faceOffWinPercentage),
nhl_data$faceOffWinPercentage)

##### hits, giveaways, blocked, and takeaways are all missing the same 4928 observations, these observations will be removed#
# Subset the data to include only the rows that have complete data for the variable of interest
nhl_data <- nhl_data[complete.cases(nhl_data[, "hits"]), ]

######Rechecking missing values, all are not missing now#######
colSums(is.na(nhl_data))

# investigate correlations between variables
cor(nhl_data[,c("goals", "won", "settled_in", "shots", "hits", "pim",
          "powerPlayOpportunities", "powerPlayGoals", "faceOffWinPercentage", "giveaways",
          "takeaways", "blocked", "startRinkSide", "RP")])

library(corrplot)

# Create a correlation matrix of your data
corr_matrix <- cor(nhl_data, use = "pairwise.complete.obs")

# Plot the correlation matrix using corrplot
corrplot(corr_matrix, method = "color")
######StartRinkSide has no correlation with won, will be dropped from model#
nhl_data <- nhl_data[, -15]



# Split data into training and testing sets

nhl_data$won <- as.numeric(nhl_data$won)


set.seed(941)
filterds <- sample(
  x = c("train","test"),
  size = nrow(nhl_data),
  replace = TRUE,
  prob = c(0.7,0.3)
)
part <- split(
  x = nhl_data,
  f = filterds
)
nhl_train <- as.data.frame(part$train)
```

```
nhl_test <- as.data.frame(part$test)


# Perform XGBoost on data
install.packages("xgboost")
library(xgboost)

X <- nhl_train[, c(1:2,4:15)]
Y <- as.numeric(nhl_train$won)
X <- as.matrix(X)
set.seed(941)

xgb_model <- xgboost(data = X, label = Y, nrounds = 100, objective =
"binary:logistic")
library(caret)
test_X <- as.matrix(nhl_test[, c(1:2,4:15)])
test_Y <- as.numeric(nhl_test$won)
    xgb_pred <- predict(xgb_model, newdata = test_X)


    # Random Forest
    library(randomForest)
    rf_model <- randomForest(won ~ ., data = nhl_train, ntree = 500)
    rf_pred <- predict(rf_model, newdata = nhl_test)

    # SVM
    library(e1071)
    svm_model <- svm(won ~ ., data = nhl_train, kernel = "linear")
    svm_pred <- predict(svm_model, newdata = nhl_test)

    # Calculate RMSE
    svm_rmse <- sqrt(mean((nhl_test$won - svm_pred)^2))
    rf_rmse <- sqrt(mean((nhl_test$won - rf_pred)^2))
xgb_rmse <- sqrt(mean((nhl_test$won - xgb_pred)^2))
```