NHL Player Predictor

edX HarvardX: PH125.9x - Data Science: Capstone

Craig Hamlin

August 5, 2021

Contents

1 INTRODUCTION

Describe NHL, salary cap, aim of project (to gain potential insights into the player composition of successful teams) and plan of project (obtaining data from publicly available sources we run several models to gain the best accuracy)

2 Methods and Analysis

2.1 Data Wrangling and Cleaning

After importing the nhl player and team data with permission from hockeystatssupplier on github, the data was then cleaned and engineered to best suit the project. Of note: only players with a minimum of 20 games played per season were included in the player datasets. This minimum games threshold was important to implement in order to provide a larger sample size for player performance.

2.1.1 Download Player Data and coerce into useable dataframe

Player data consists of two separate dataframes. Each contain the similar columns such as player names, position, etc, however one dataframe contains salary cap based information like points/cap and the other dataframe contains other in-depth game based information like ice time and power play points. The two separate dataframes were modified with regex to create a more workable resource by removing dollar signs, trimming white space and removing accents from letters. Also, the features cap hit and point per cap hit were weighted to account for the salary cap inflation that occurs yearly. The two data frames were then joined to be utilized later alongside the team data.

Player	AGE	CAP.HIT	X..P	GP	TEAM	Pos	Season	G	A	P	PPP	TOI.GP
Jamie Benn	22	798523	12675	71	DAL	L	20112012	26	37	63	10	1084
John Tavares	21	1140747	14083	82	NYI	C	20112012	31	50	81	25	1234
Frans Nielsen	27	665435	14158	82	NYI	C	20112012	17	30	47	15	1047
Erik Karlsson	21	1109059	14218	81	OTT	D	20112012	19	59	78	28	1519
Adam Henrique	21	728810	14290	74	NJD	C	20112012	16	35	51	8	1090
Jordan Eberle	21	1109059	14593	78	EDM	R	20112012	34	42	76	20	1056

2.1.2 Download Team Data and coerce into useable dataframe

Team data consists of team performance data from years 2011-2020. The accumulated points of each team by year is displayed in a column within the dataframe. From this data a three tiered ranking system was developed which split teams into three groupings: 1 (top third percentile), 2 (middle third percentile), and 3 (bottom third percentile). Each team in each year now has a performance ranking attribute applied.

	PTS	Team	rank
26	80	Anaheim	1
8	102	Boston	3
20	89	Buffalo	2
18	90	Calgary	2
24	82	Carolina	1
11	101	Chicago	3

2.1.3 Combine the Player and Team Data

The final dataframe utilized in the models was created by combining the Player and Team dataframes. Several steps were needed to fit the two groupings together: city names in the team data needed to be converted to abbreviations to match the player data. The number of teams and team city names aren't exactly the same for all seasons so this needed to be addressed in the coding. The final step was to bind the two dataframes, convert variables to numeric where necessary and edit out any unwanted columns that would not be used as features in the model.

Player	AGE	CAP.HIT	X..P	GP	TEAM	Pos	Season	G	A	P	PPP	TOI.GP	rank
Jamie Benn	22	798523	12675	71	DAL	L	20112012	26	37	63	10	1084	elite
John Tavares	21	1140747	14083	82	NYI	C	20112012	31	50	81	25	1234	elite
Frans Nielsen	27	665435	14158	82	NYI	C	20112012	17	30	47	15	1047	elite
Erik Karlsson	21	1109059	14218	81	OTT	D	20112012	19	59	78	28	1519	elite
Adam Henrique	21	728810	14290	74	NJD	C	20112012	16	35	51	8	1090	elite
David Desharnais	25	1077372	17955	81	MTL	C	20112012	16	44	60	20	1104	elite

2.1.4 Create training and test sets

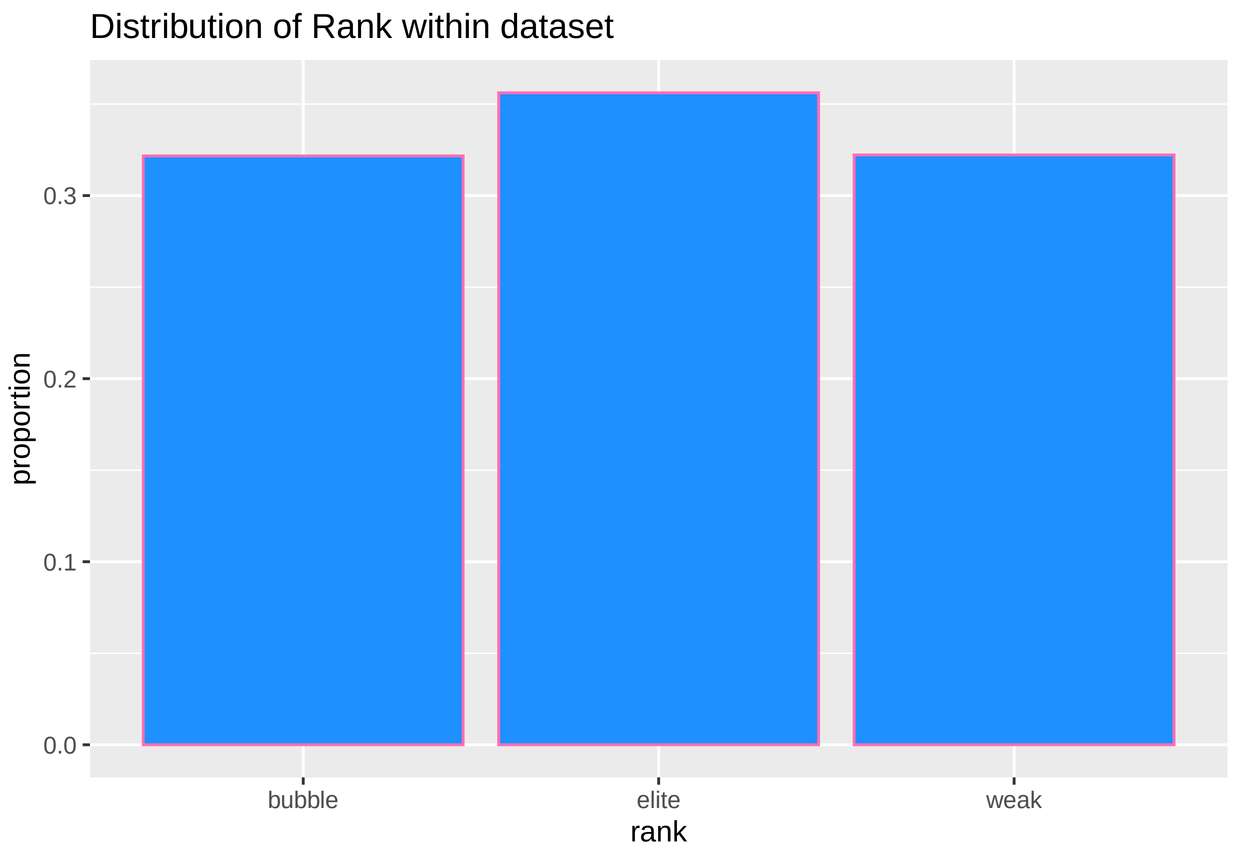
The player and team data obtained for this project covers the years 2011 until 2020. For modeling purposes the decision was made to designate the data from years 2011-2019 as the training set and the data from year 2020 as the test set. The training set consists of 5571 player values and the test set consists of 597 player values.

2.2 Exploratory Data Analysis

Exploratory Data Analysis is the initial process of analyzing your dataset to try and gain insight and obtain a summary of the relevant characteristics. The usage of graphs and summary statistics will be conducted in order to gain maximum insight into both the player and team data.

2.2.1 Distribution of Rank within Dataset

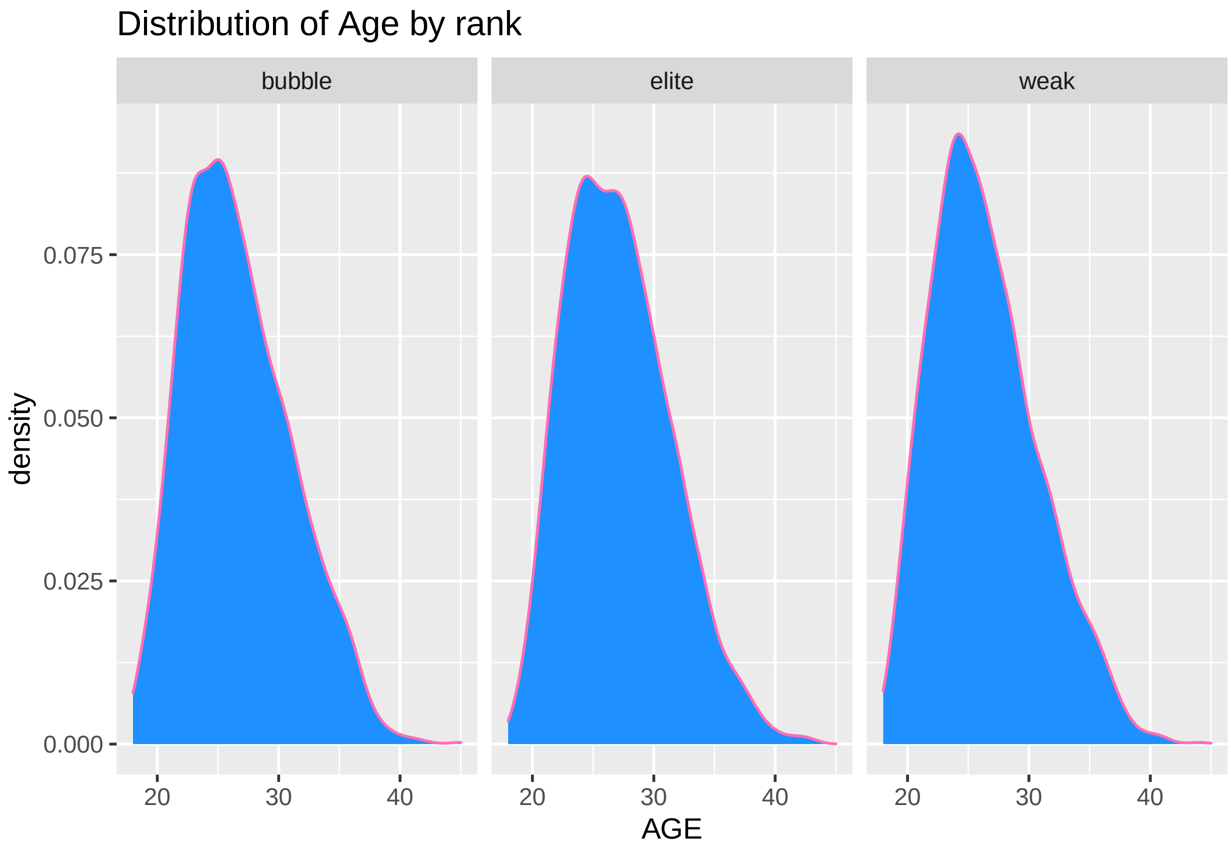
The data is separated into three rankings (Elite, Bubble, and Weak) which are based off of a 10 year sample of all players in the league and the team results for these seasons.



The bargraph above shows that the highest proportion of the training data is elite and the lowest proportion is weak. We have divided our team standings equally by percentile to create a non biased representation of team so why are we seeing a higher proportion of players that play for elite rather than weak teams? The answer would be found in the selection criteria for the dataset, in particular the minimum games played selector in the player data. As our cut off level for games played is 20 this would indicate that weak teams have more players with under 20 games played per season than elite or bubble teams. From this interpretation we can deduce that elite teams keep a roster of players that remains fairly constant through an entire ~80 game season. Weak teams, on the other hand, tend to have a roster that isn't firmly set and could feature young prospects and fill in players where the expectation level and number of games played is lower on average.

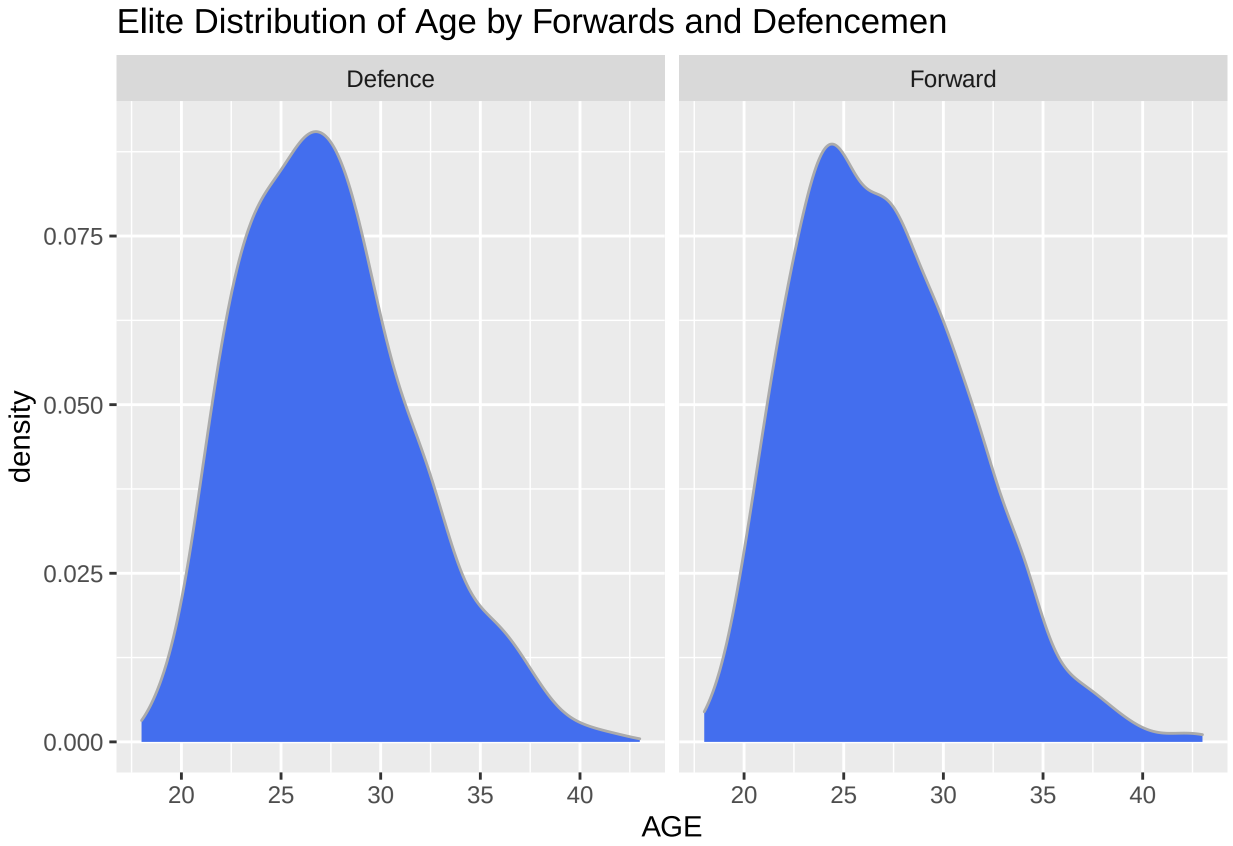
2.2.2 Distribution of Player Age by Team Rank

In order to search for noticeable difference between ranks we will select the player age feature and visualize the three separate distributions.



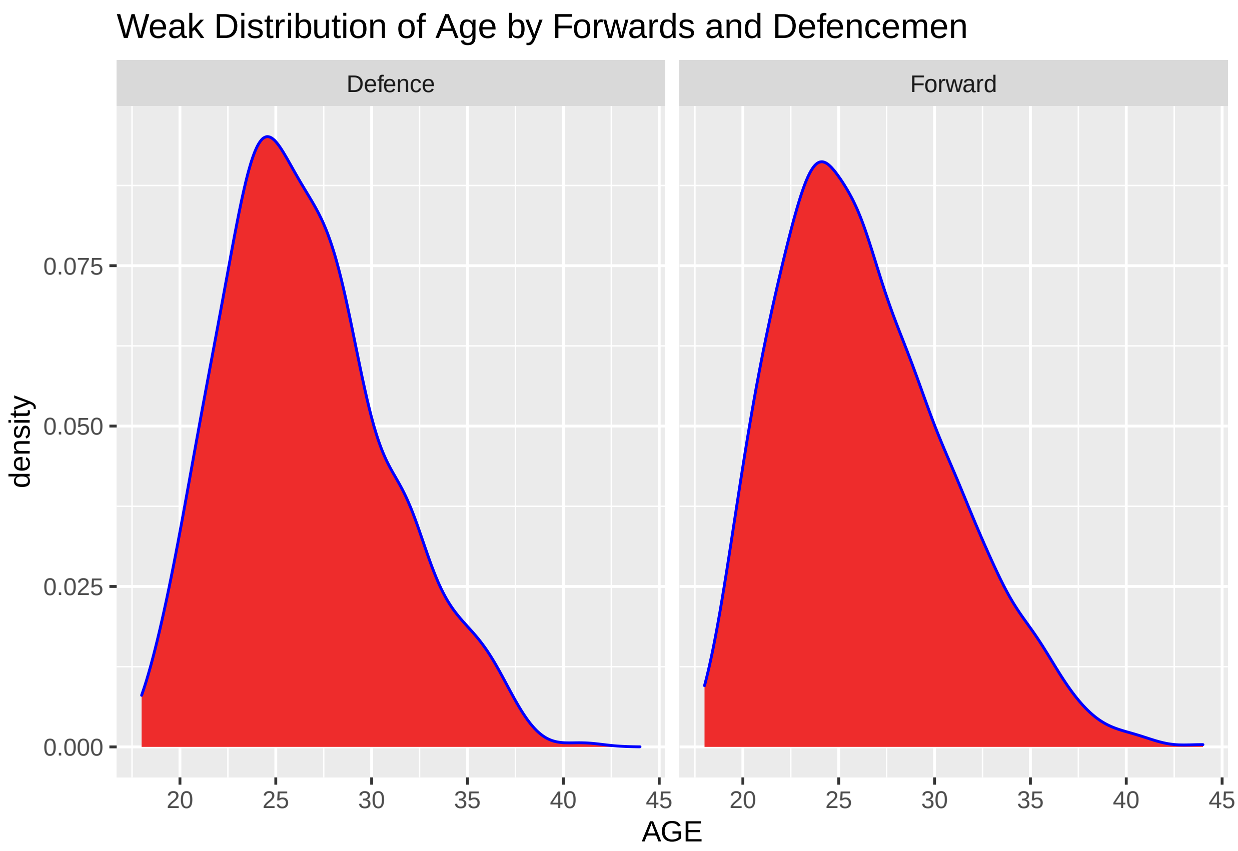
We see in the density plots expressed above that there is visible difference between the team rankings in accordance to player age distribution. The age distribution of the weak teams shows a steep peak that is centered at approximately 24 years of age. The bubble teams age distribution peaks at 25 years of age. The elite teams have bimodal age distribution peaks of 24.5 and 27 years of age. Doing some math it appears that the these bimodal values are tied to Position. The highest portion of players for elite forward positions (C,LW,RW) fall within one year difference around the first peak of 24.5 while th highest proportion of players for elite defence position (D) fall with in one year difference of the second peak 27 years. This phenomenon can be seen in the following density plot example.

2.2.3 Elite Distribution of Age by Forwards and Defencemen



The visualization above confirms that there is a distinct bimodal distribution of age for the elite teams. The distinction between age and position is quite clear for an elite team. How might this look for the weakly ranked teams?

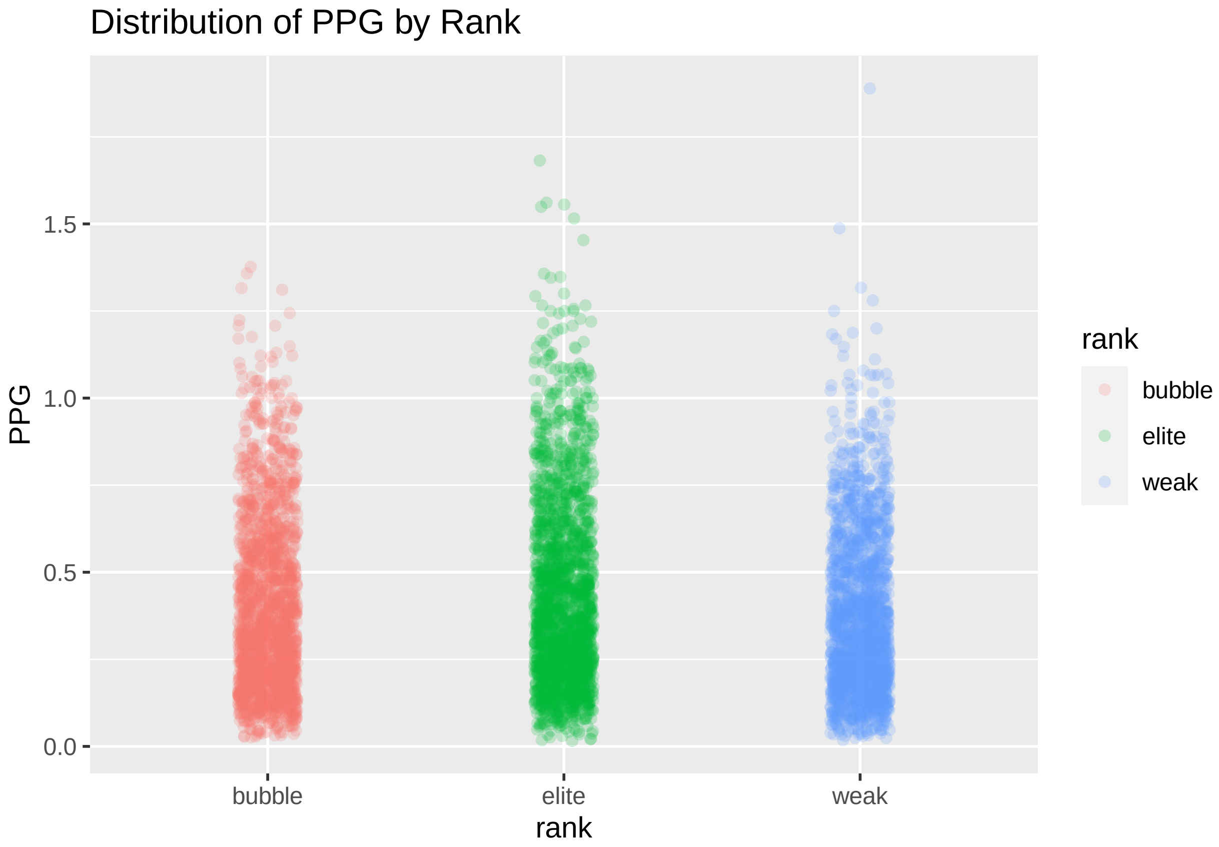
2.2.4 Weak Distribution of Age by Forwards and Defencemen



As we can see in the distribution the age separation for the two positions is not nearly as distinct as with the elite teams. From this information we can infer a correlation between elite teams and age distinction of players by position. An elite team is will generally have its highest distribution of forwards around 24.5 years of age and its highest distribution of defencemen around 27 years of age;

2.2.5 Distribution of Points per Game determined by Rank

As the objective of hockey is to score more points than the other team in order to win the match, a player with a high point per game total will be valuable to team success. By using a jitter plot we can see the differences in data between the three rankings of teams.

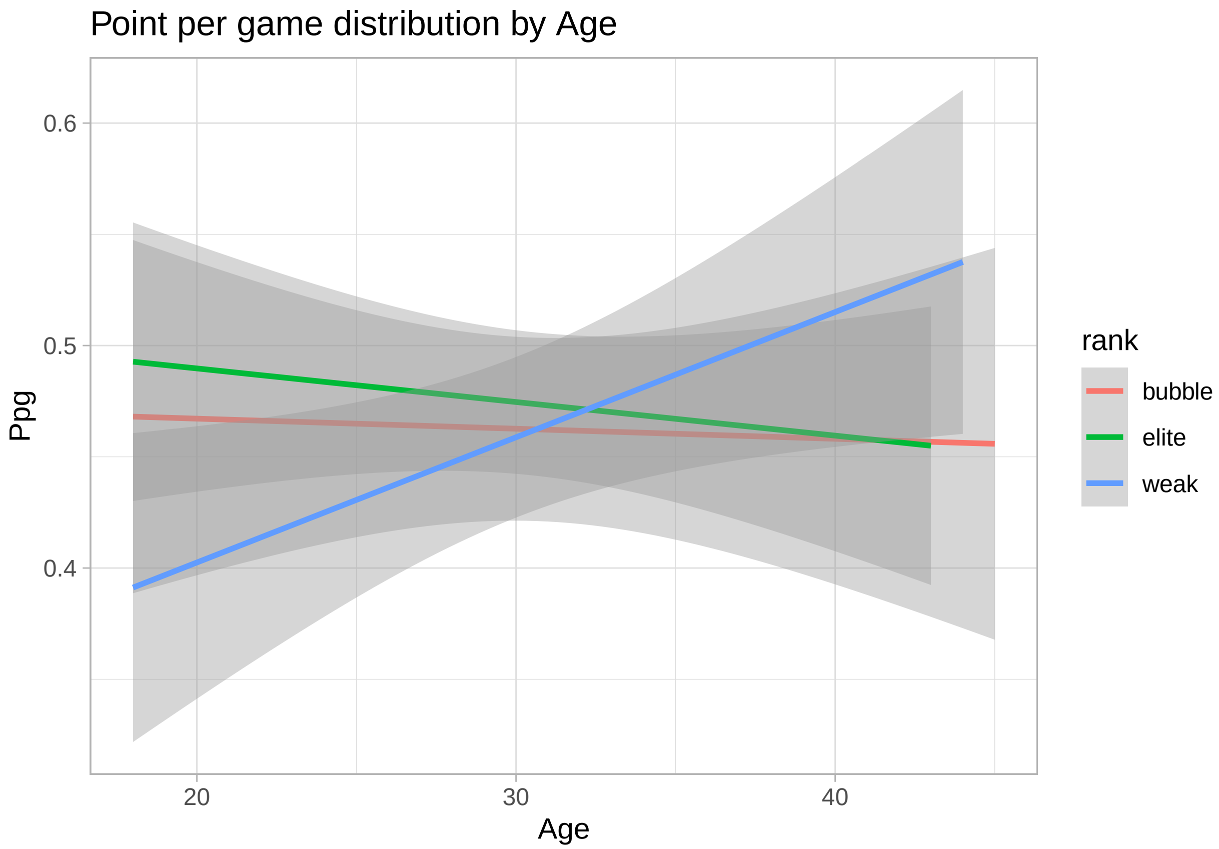


From the visual we can see that though the weak ranking contains the highest individual value for ppg, the elite column has a much higher overall total of ppg values. Summarizing the mean ppg per ranking shows elite with the highest average and weak with the lowest average.

rank	avg
elite	0.4344039
bubble	0.3996136
weak	0.3704804

2.2.6 Point distribution by Age

We have seen how elite teams have the highest mean age and seem most specific in the relevance of age distribution. How might the PPG be influence by Age for the three rankings?



As we can see from the linear interpretation of the data, weak teams are much more reliant upon older players to carry their scoring whereas the elite and bubble teams have a much more even distribution of scoring by age.

2.3 Model and Tuning Exploration

In this project we will utilize several models in an attempt to obtain the best accuracy. These models will consist of Classification and Regression Tree (rpart), Robust Linear Discriminate Analysis (Linda), and eXtreme Gradient Boosting (xgbTree). As there is considerable variability in the time it takes to run these different models we will include time in our results as a consideration of efficiency. The model will be performed on both the training set and test set in order to see if there is a significant difference between the accuracy obtained by the model for both datasets.

2.3.1 Step One: Guessing the accuracy

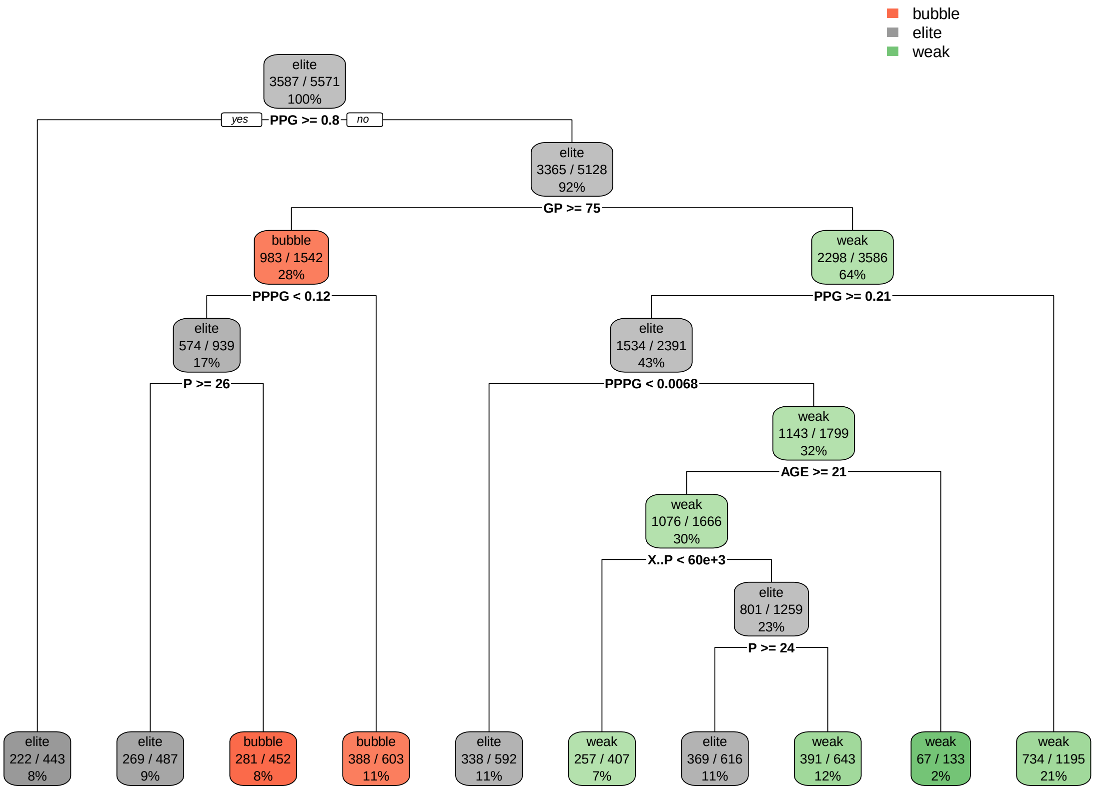
Guessing the accuracy allows us the ability to determine a baseline where results are obtained by simple random selection.

Predictor	Accuracy..train.	Accuracy..test.
Guess Model	0.2498654	0.2629816

Using a system of guessing we obtained an accuracy of 0.263 for the training data and 0.263 for the test data.

2.3.2 Step Two: Regression Tree

Regression Tree is the result of a tunable learning algorithm known as rpart. Using the statistical method of recursive partitioning, the rpart algorithm splits data into simpler versions of itself which then continue to split until the expectation of the algorithm parameters are met. The end result of the algorithm is a regression tree which can aid in the classification of data. Rpart contains one tunable hyperparameter, the complexity parameter, which is set at a default value of 0.01. A tunegrid dataframe containing a sequence of values between 0 and 0.05 was utilized to replace the default value.



Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
Rpart Model	0.4047747	0.3919598	157.34

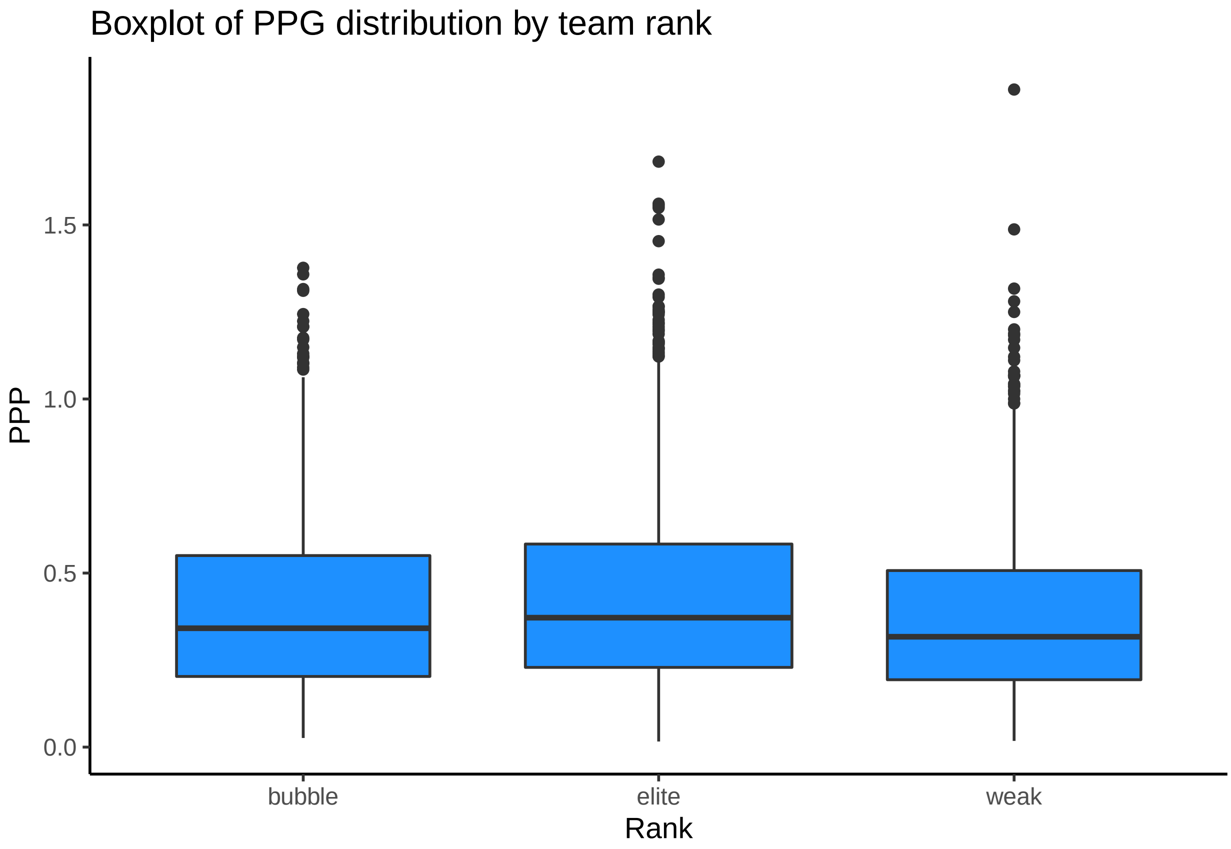
Each node shows the predicted rank, the predicted probability of a player being of that rank, and the percentage of observations in the node. The best tune of the complexity parameter was 0.004 there is a slight overfit as the train set performs better than the test set. For the training and test sets we see an accuracy of close to 40% which is much higher than the guessing model which is only around 25%. The accuracy obtained isn't particulary high but the nodes of the classification tree can provide insight on the data by determining which features were highlighted in the recursive modeling process. We see PPG, P, PPPG, GP, AGE, and X..P included as the relevant arguments in the decision flow of the classification tree. PPG appears to be important to all rankings while X..P is more relevant in defining the differences between the elite and weak ranking.

2.3.3 Step Two: Linda

While the rpart method contained one tunable hyperparameter (cp) there are many algorithms which perform without a tuning option, some examples include: Bayesian Generalized Linear Model, Gaussian Process, and Linear Discriminant Analysis. The benefit of these models with no tunable hyperparameters is that results are generally obtained quickly and simply. After testing several models with non-tunable parameters the highest accuracy was found using Robust Linear Discriminant Analysis (Linda).

Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
Linda model	0.3963382	0.4070352	157.7

In considering why Linda outperformed the other non tunable algorithms, the Linda model being robust indicates that it is resistant to outliers and thus minimizes error to create a higher accuracy.



As shown in the above boxplot there are noticable outliers in the feature of ppg. Outliers are also consistently evident in the featured variables of PPPG, X..P, CAP.HIT and PPP.

2.3.4 Step Four: xgbTree

As Linda does not implement tunable hyperparameters it would be good to find a model that is similarly robust but also tunable: an algorithm with tunable hyperparamters can produce model specific optimization when training the data. XGBoost and boosting in general are very sensitive to outliers.This is because boosting builds each tree on previous trees' residuals/errors. Outliers will have much larger residuals than non-outliers, so boosting will focus a disproportionate amount of its attention on those points

2.3.4.1 xgbTree with default parameters

Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
xgbTree - Default	0.4448034	0.4304858	154.33

The xgbTree model using the default parameters resulted in a significant increase in accuracy in both training and test sets over the Linda and rpart models. The default tuning parameters for the algorithm are: nrounds = 150, max_depth = 3, eta = 0.4, gamma = 0, colsample_bytree = 0.8, min_child_weight = 1 and subsample = 0.5.

2.3.4.2 xgbTree with tuned hyperparameters By tuning the hyperparameters we attempt to find the optimal levels for producing the highest accuracy. The first step of tuning was to alter the default number of rounds from 150 instead to a sequence from 100 to 1000 increased by intervals of 50. The learning rate was also expanded to include 0.1, 0.2 and 0.3 and the max tree depth count now included all whole numbers between 1 and 5.

Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
xgbTree - step1	0.6738467	0.4371859	695.67

We see a definite increase in the accuracy of the model once the initial hyperparameter tuning was performed. The best tune for learning rate was 0.1 and the best tune for maximum tree depth was 5.

2.3.4.3 tuning minimum child weight Moving further with the tuning, the best tune values of learning rate and max tree depth were added to a new tuning grid which also contained multiple values for minimum child depth.

Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
xgbTree - step2	0.6847963	0.4321608	200.97

We see a slight increase in the accuracy of the model once the initial hyperparameter tuning was performed. The training accuracy was higher but the test accuracy was lower.MAYBE? The best tune for minimum child weight in this model was 0.25

2.3.4.4 Tuning subsample ratio of columns and training instances In the next step, the best tune values of learning rate, max tree depth and minimum child weight were added to a new tuning grid which also contained multiple values for subsample ratio or columns and training observations. This process creates different random samples of the training data prior to growing trees which can help prevent overfitting.

Predictor	Accuracy (train)	Accuracy (test)	Time (secs)
xgbTree - step3	0.6939508	0.438861	568.92

We see a slight increase in the accuracy of the model after tuning these hyperparameters. The training accuracy was higher but the test accuracy was lower.MAYBE? The best tune for column subsample in this model was 0.6 and the best tune for subsample ratio of training observations was 0.5

3 Results

The table below shows our results through all explored versions of our model.

The ??? model listed, '???' created the best results for our accuracy.

Here is a list of the Variable Importance for our best model

This shows that three top variables really stand out(TOI.GP, X..P, P), our variables are middling (CAP.HIt, AGE, GP, PPP), one variable is low (PosD), while PosR and PosL are listed but the value is essentially negligible.

A team that is seeking to attain elite status should consider focusing resources and attention accordingly on these features. If you want your team to be elite a positive strategy would be to model ice time of players in the same way as other elite teams with an extra emphasis on Defence. You want to have an X..P similar to the elites and you can evaluate your players when offering salary contracts in that regard. and perhaps the most obvious and classic standard of Points is relevant

As the accuracy certainly isn't perfect we can't say for certain that a team will have more success if they focus on these areas, however we have found a significant correlation between team performance and these features. By studying the features of the elite teams, a performance expectation of the players can be more clearly defined. Further testing could be conducted between variables to gain more insight. For example, if a correlation between age and average ice time for elite teams vs weak teams is identified, a weight potentially could be applied to the model.

4 Conclusion

Stoked!