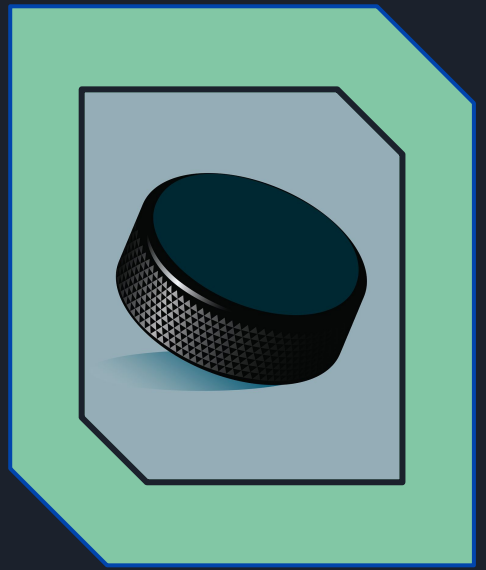
Predictive Modeling in the National Hockey League

Logan Schmitt

NHL Outcomes are Hard to Predict

Many Random Events per Game

- Unpredictable bounces
  - Bad “puck luck”
- Relatively low-scoring sport
  - One goal can change a game
- High-speed sport
  - Lots of deflections and collisions



NHL Outcomes are Hard to Predict

"Weak Link" Sport

- Superstars only play 25-40% of the game
- Depth players have to contribute
- A team is only as good as its weakest link

NHL Prediction Stakeholders

Who would want to predict an NHL season outcome?

NHL Team Officials

- Evaluate single player's impact on team's projected performance
- Understand what metrics make a playoff team

Sports Betting World

- Sportsbooks set team playoff odds based on projections
- Bettors can find value in favorable odds after roster moves

The business question:

Can I use 2021-22 statistics to predict 2022-23 playoff teams?

And do the predictions generalize to other seasons?

Exploratory Phase

- Data Collection & Wrangling
- Exploratory Data Analysis

Data Wrangling

Can I collect data to answer the business question?

Process

1. Created a web scraping function to pull player and team statistics from Hockey Reference
2. Accounted for new teams and teams whose names had changed
3. Aggregated player statistics by team to create a snapshot of each roster for each season
4. Adjusted shortened seasons due to lockout and COVID to standard 82 game season length

Data Wrangling

	Player	Age	Pos	GP	G	A	PTS	+/-	PIM	S	S%	OPS	DPS	PS	Team	Playoffs	Season
0	Brian Gionta	27.0	RW	82	48	41	89	18.0	46	291	16.5	8.5	2.9	11.4	NJD	1	2006
1	Scott Gomez	26.0	C	82	33	51	84	8.0	42	244	13.5	6.9	2.4	9.2	NJD	1	2006
2	Jamie Langenbrunner	30.0	RW	80	19	34	53	-1.0	74	243	7.8	2.6	2.0	4.6	NJD	1	2006
3	Brian Rafalski	32.0	D	82	6	43	49	0.0	36	126	4.8	2.9	5.7	8.5	NJD	1	2006
4	Patrik Eliáš	29.0	LW	38	16	29	45	11.0	20	142	11.3	3.9	1.4	5.2	NJD	1	2006

Player statistics DataFrame after importing

Data Wrangling

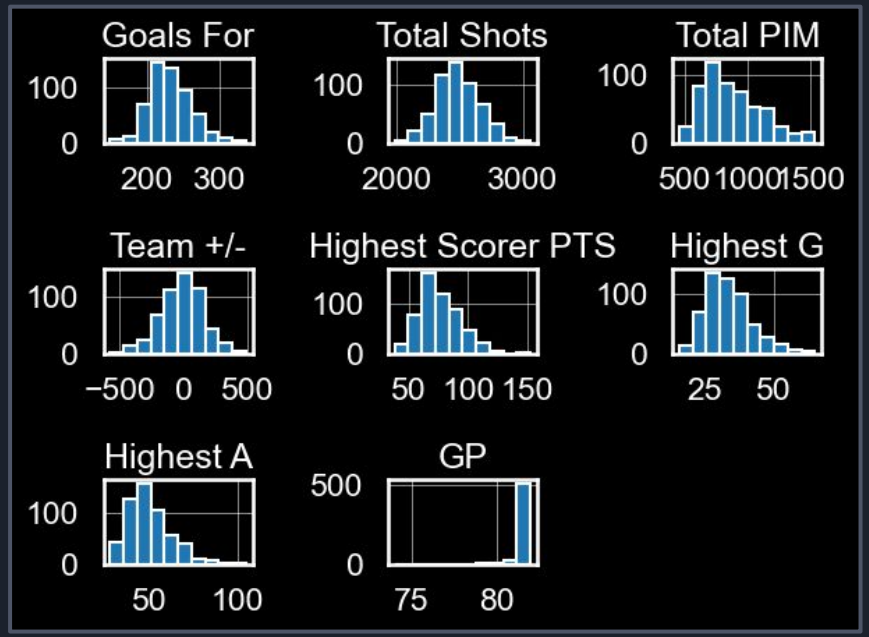
	Team	Season	Goals For	Total Shots	Total PIM	Avg Age	Team +/-	Sum OPS	Sum DPS	Highest Scorer PTS	Highest G	Highest A	GP	Playoffs
26	ANA	2006	251	2585	1445	28.2	126.0	39.3	43.4	90	40	51	82	1
17	ARI	2006	242	2318	1493	27.1	-135.0	36.0	32.1	66	30	41	82	0
2	BOS	2006	228	2511	1162	27.7	-70.0	30.8	32.7	73	31	43	82	0
11	BUF	2006	276	2510	1144	25.9	-8.0	47.5	40.0	73	30	51	82	1
25	CAR	2006	286	2553	1107	28.0	41.0	50.7	34.4	100	45	55	82	1

Team statistics DataFrame after aggregation

Exploratory Data Analysis

Process

1. Checked distributions of statistics
2. Explored differences in statistical measures between playoff and non-playoff teams
3. Checked for collinearity of features
4. Plotted teams along principal component axes

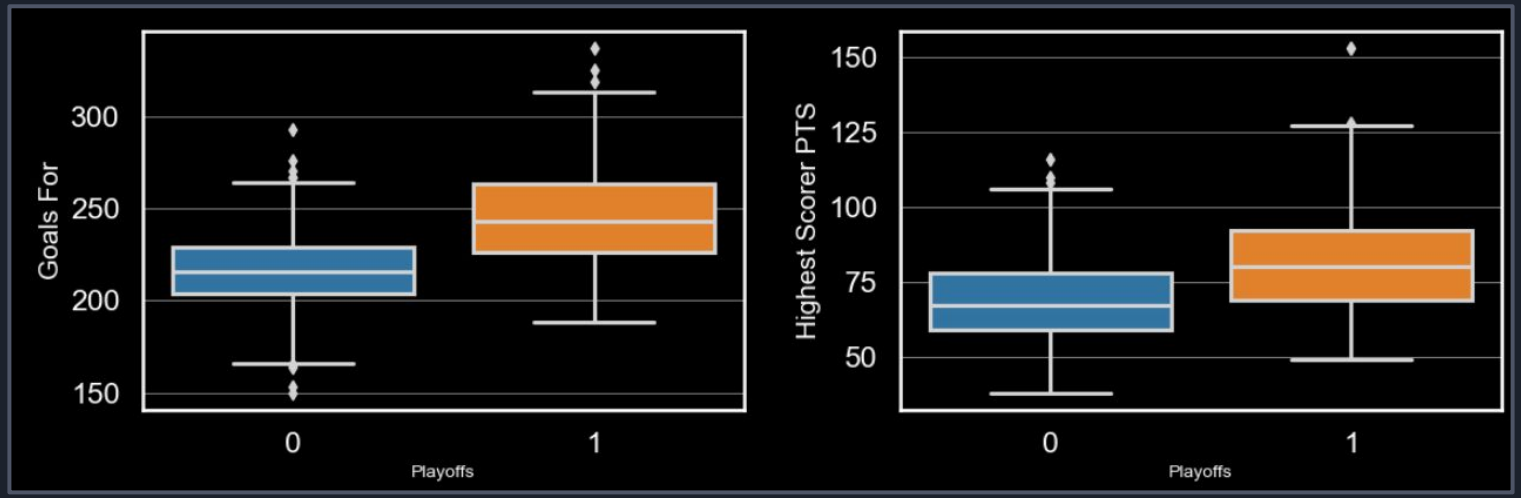


Histograms showing distributions of each statistic

Exploratory Data Analysis

Explored Playoff vs Non-Playoff Team Stats

- Each plot showed a clear divide between playoff and non-playoff teams

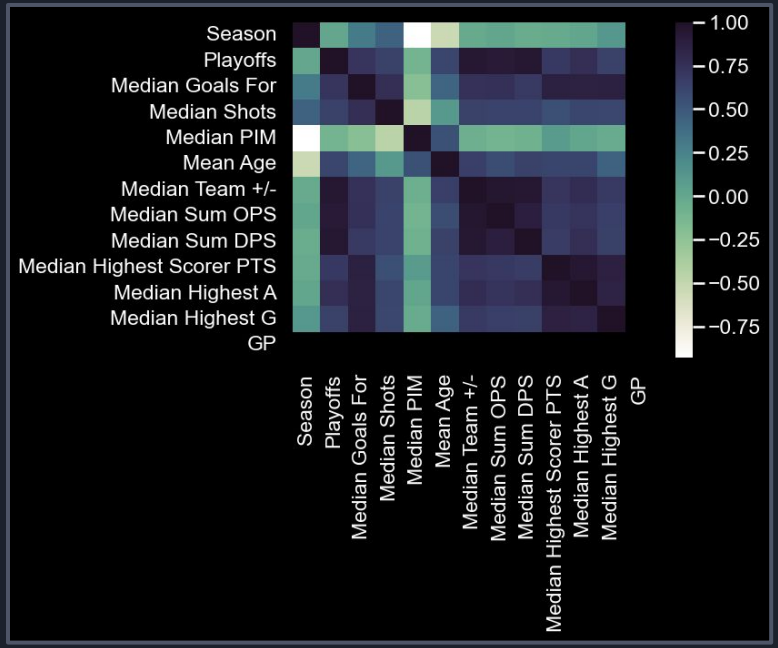


Boxplots of goals for and highest scorer points among playoff and non-playoff teams

Exploratory Data Analysis

Checked for Multicollinearity

- The point share statistics were correlated with +/-
- Higher scoring teams correlated with having higher scoring individuals

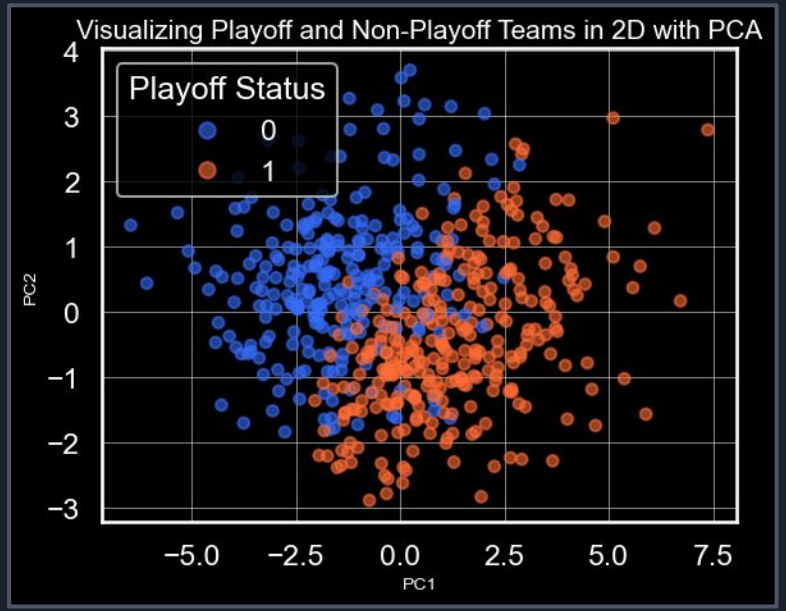


Correlation heatmap of features

Exploratory Data Analysis

Principal Component Analysis

- 2 components explained about 60% of variance
- 9 components explained 100% of variance
- Plotting teams along 2 component axes revealed overlapping clusters



Playoff and non-playoff teams plotted along principal component axes

Modeling Phase

- Feature Engineering
- Model Selection

Feature Engineering

How do I prepare the data to model my problem?

Process

1. Create a stat line for each player in a given season
2. Aggregate statistics for players on updated teams
3. Add features for top 3 goal scorers and top 3 assist earners
4. Scale features
5. Create train/test split over time

Feature Engineering

	Team	Season	G	S	PIM	Avg Age	+/-	OPS	DPS	g_1	g_2	g_3	a_1	a_2	a_3	PTS%
0	ANA	2007	213	2075	1231	28.8	59.0	38.1	35.2	0.500	0.415	0.275	0.625	0.622	0.610	0.671
1	ANA	2008	233	2410	1284	30.3	115.0	41.9	47.6	0.585	0.309	0.305	0.697	0.684	0.603	0.622
2	ANA	2009	170	1970	1337	28.9	61.0	26.6	39.9	0.462	0.414	0.312	0.753	0.431	0.423	0.555
3	ANA	2010	223	2350	1452	27.1	20.0	43.6	33.1	0.484	0.415	0.410	0.815	0.549	0.523	0.543
4	ANA	2011	213	2063	1103	29.0	-36.0	38.3	25.4	0.500	0.432	0.329	0.758	0.598	0.465	0.604

Feature engineered DataFrame before scaling

Feature Engineering

Modified Standard Scaling

- Some seasons see higher scoring rates due to rule and technology changes.
- Scaling features across all seasons would imply that there were many outliers in high-scoring seasons.
- Scaling features across each individual season paints a clearer picture of which teams were best in a given season.

Feature Engineering

Train/Test Split

- Time series data should not be shuffled randomly
- Train on season 1, test on season 2
- Train on seasons 1 and 2, test on season 3
- Repeat for all training seasons
- Evaluate testing metrics

Model Selection

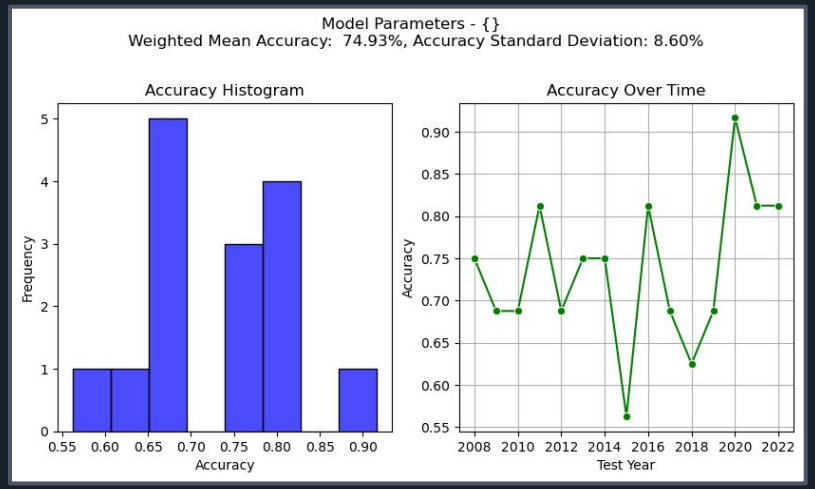
How will I know if the model is any good?

- Convert predictions to playoff seeds, score the precision of those picks
  - Model scores will be low if just predicting points percentage
  - The goal is to capture relative performance, not absolute
- Choose models with better average scores, with more weight to more recent seasons
  - Recent season predictions are supported by more training data and may better capture modern trends
- Grid search through hyperparameters, then choose from top models

Model Selection

Ordinary Least Squares Linear Regression Model

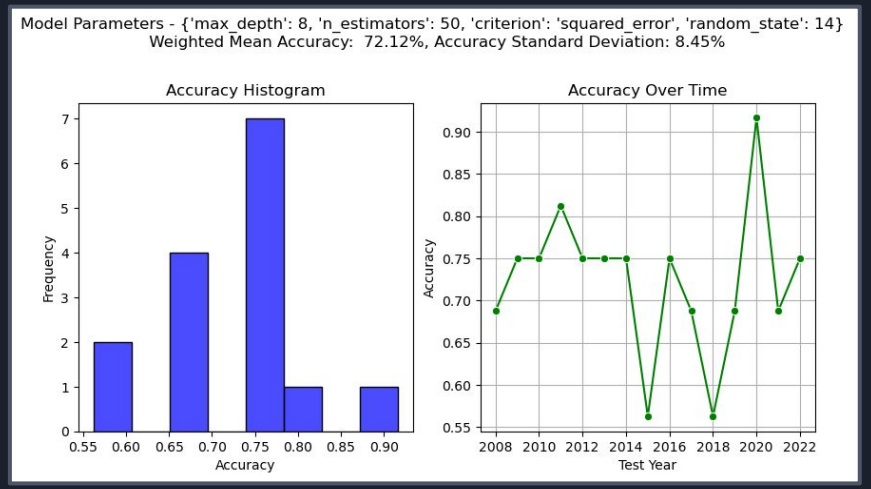
- No hyperparameters to tune
- Near 75% weighted average accuracy
- Excellent performance in past 3 years



Model Selection

Random Forest Regression Model

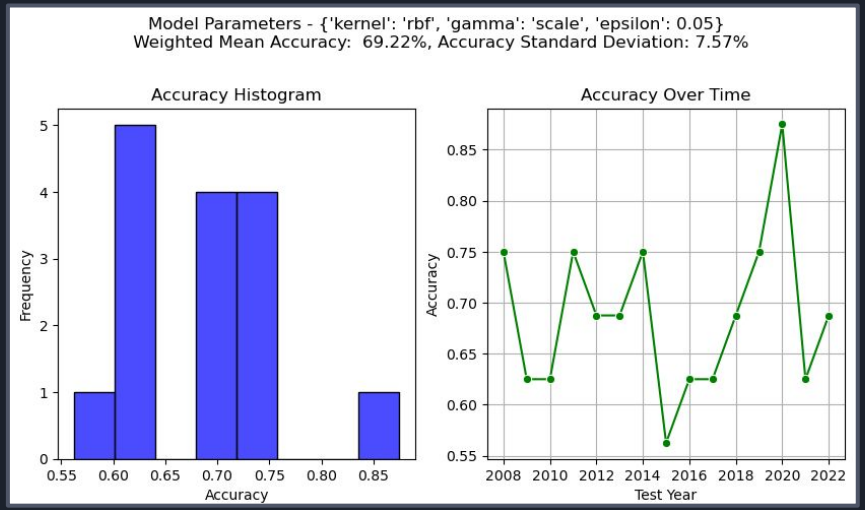
- Tuned 3 hyperparameters
- Just over 72% weighted average accuracy
- Hits 75% frequently, but has some bad years



Model Selection

Support Vector Regression Model

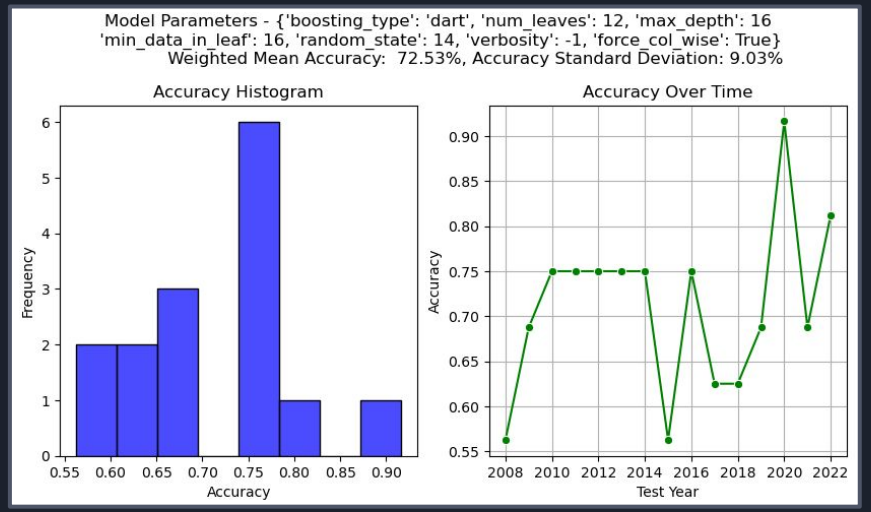
- Tuned 3 hyperparameters
- Just under 70% weighted average accuracy
- Lots of fluctuation



Model Selection

LightGBM Regression Model

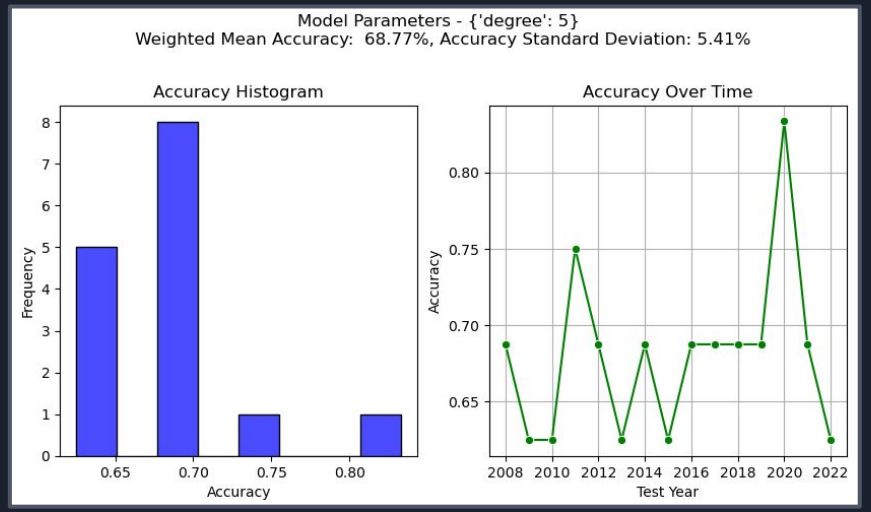
- Tuned 4 hyperparameters
- Over 72% weighted average accuracy
- Slight improvement over Random Forest



Model Selection

Polynomial Regression Model

- Tried degrees 2 through 7
- Just under 69% weighted average accuracy at degree 5
- Poor fit for data, rarely exceeding 70% accuracy



Model Selection

- Linear Regression performed well
  - Elastic net regression and Bayesian ridge regression (not shown) were identical to linear regression
- LightGBM Regression performed nearly as well
  - Random Forest Regression was slightly worse than LightGBM
- Support Vector Regression was all over the place
- Polynomial Regression was worst

Results

- Ensemble Model Results
- Future Prediction

Individual Model Results

How did the models perform when predicting on the unseen test data?

2022-23 was the holdout test season

- Linear Regression picked 10/16 playoff teams correctly - 62.5%
- LightGBM Regression picked 12/16 playoff teams correctly - 75%
- Support Vector Regression picked 12/16 playoff teams correctly - 75%

Ensemble Model Results

What if the models worked together to pick teams?

Ensemble Approach

- Each model predicts points percentage
- The predictions are scaled for each model and averaged for each team
- The average predictions are converted to playoff seeds and scored
- The ensemble model correctly identified 12/16 playoff teams in 2022-23

Future Prediction

What teams will make the 2023-24 playoffs according to the ensemble model?

Eastern Conference		Western Conference	
Atlantic Division	**Metropolitan Division**	**Central Division**	**Pacific Division**
Boston Bruins	New Jersey Devils	Colorado Avalanche	Edmonton Oilers
Tampa Bay Lightning	New York Rangers	Dallas Stars	Vegas Golden Knights
Toronto Maple Leafs	New York Islanders	Minnesota Wild	Los Angeles Kings
Buffalo Sabres	Carolina Hurricanes	Winnipeg Jets	Seattle Kraken