Quantile Regression for Fantasy Football Projections

Kyle Ciarkowski
Department of Statistics
University of Michigan
Ann Arbor, Michigan
kyledc@umich.edu

Arjun Mayur
Ross School of Business
University of Michigan
Ann Arbor, Michigan
amayur@umich.edu

I. Introduction

Fantasy football prediction has seen significant improvements over the past two decades, driven by advancements in predictive analytics and cutting edge data science techniques. The requirement are models that must both predict a player's likely outcome for a given game week but also quantify the uncertainty of that outcome. NFL player performance is highly volatile on a week-to-week basis, due to factors ranging from the quality of the gameday opponent, changes in team strategies, weather, etc. While predicting a single value for a given player provides a simple and easy to understand metric to quantify, it does fail to generate a full picture of the player's risk (floor) and upside (ceiling).

The primary goal of this project is to provide more context to a players prediction by developing a robust Quantile Regression Model for weekly NFL player performance, specifically targeting half-point PPR (Points Per Reception) fantasy points. This framework is based on generated reliable Prediction Intervals (PI) that achieve a target coverage of 80%, such that the true prediction value will fall in this prediction interval 80% of the time. This will provide fantasy managers with invaluable information to properly assess risk based on the needs of their team for a given week.

Quantile Regression utilized estimates specific points in the conditional distribution of the target variable, rather than focusing on just the mean. The training of three separate models for the $10^{th}, 50^{th},$ and 90^{th} quantiles established uncertainty bounds around the median prediction. A gradient boosting method was used as the primary modeling technique, with the support of quantile loss (i.e., pinball loss) needed for construction of the lower and upper quantiles and to help capture any non-linear relationships present in our detailed, play-by-play NFL data set.

The remainder of this paper will provide details in the data preparation steps, feature engineering and model construction evaluation needed to achieve target prediction interval coverage of 80%.

II. Methods

The task for this project was the use of a modeling approach that was able to handle the stochastic nature of NFL weekly performance data. Our methodology is centered around three areas; detailed play-by-play NFL data and appropriate filtering, feature engineering based on metrics relevant to predicting performance, while taking into account volume share and volatility, and the implementation of a non-parametric algorithm to help quantify uncertainty.

A. Data and Features

The model is built using historical data sourced from the NFL Play-by-Play (PBP) open source dataset found in the `nfl_data_py` package that contains data for every NFL play from 1999 to present day. For the purpose of this project, we used data from the 2015 through 2024 seasons, which provided a necessary quantity to calculate career averages (although we will address later that there are limitations to this) and enough data points to reliably train the model on. The resulting final aggregated data set consisted of 54,213 unique player-week observations.

After the data was extracted and aggregated, the next step was to define the target variable, `Y_target_points`, which was calculated using Half-PPR scoring rules (1). The PBP data also required filtering to ensure that only relevant offensive plays (pass, run, qb_kneel) were included, excluding penalties that don't have direct contribution to fantasy points and kicks, punts and defensive points. The decision was made for simplicity reasons to omit defensive and special teams from consideration in modeling due to the high degree of randomness and to prevent the model from being overly complex. Lastly, data imputation was performed for quarterback kneels to ensure that accurate negative points were assigned for any lost yards due to this type of play which assigns negative rushing yards to any player that kneels (although it is almost always a quarterback that performs this action).

B. Feature Engineering

The principal challenge we found was in predicting fantasy performance was the mitigation of data leakage, specifically as it pertains to features that contain information only know after the game's conclusion. As a result, all raw game outcome metrics, such as yards gained, touchdowns, EPA (Expected Points Added), and CPOE (Completion Percentage Over Expectation) were excluded from the final design matrix to prevent any potential leakage.

Because of the omission of several raw game outcome metrics, we needed to generate features to bridge this game to ensure predictions were adequate. The final feature set was constructed based on the following categories:

1) **Historical Performance and Trend**
   - Lagged Scores and Averages: `Y_lag_1` (previous week's score), `Y_roll_avg_3` (3 week rolling average) and `Y_cum_avg` (career cumulative average). We expect that these variables pick up on changes in trends but also are accounting for historical values since future performance is highly correlated with career historical performance.
   - Efficiency Metrics: `epa_roll_avg_3` (lagged 3 week average EPA), provides a historical metric of the player's efficiency that is independent of their fantasy points total.
2) **Player Workload and Opportunity**
   - `total_plays_involved` is used to represent the number of snaps a player was involved in during the given week and is clever way to generative strong predictive power without causing data leakage.
   - Also lagged target shared to help pick up on shifts in players involvement in an offense based on past performance.
3) **Volatility and Uncertainty** Also needed features to help capture a player's inherent volatility from week-to-week.
   - Rolling Absolute Error (MAE): `Y_MAE_roll_3` measures the average absolute deviation of a player's score relative to their own mean over the prior three weeks.
   - Spike Rate: `Y_Spike_Rate_5` calculates the percentage of games in the last five weeks were a player exceeded 20 points. 20 points was chosen as this is widely regarded as a very successful fantasy performance week for a given player.

C. Algorithm Selection and Training

Before passing the data through the model, any categorical features were one hot encoded and all numerical features were standard scaled to ensure they contributed equally to predictions. In addition to the those preprocessing steps, the data set was subjected to chronological split (90% train, 10% test). This was crucial as the time-series splitting of the data ensured that the model is always evaluated on future, unseen data, which simulates weekly performance predictions. Without this chronological split, results would be inflated by non-causal correlations as a result of temporal data leakage. Due to the data exhibiting heteroskedasticity (i.e., variance of the actuals increases as predictions increase), we needed a model that could account for the lack of convergence to a predicted average for higher performing players. In the subsequent sections we provide an overview of the model as well as the loss function it utilizes.

1) **HistGradientBoostingRegressor (HGBR) for Quantile Estimation**

   This algorithm is a tree-based ensemble learning that serves as the our model for this project. It is an optimized version of the standard Gradient Boosting Machine (GBM) and uses a histogram-based binning of continuous features. It works just as any gradient boosting method does by correcting the errors (residuals) of previous tree to gradually improve the overall prediction, however it bins continuous features values into discrete integers instead of evaluate split points for all data points like a GBM. This results in a large reduction in computation costs (even though this wasn't a concern for us with the smaller amount of data). The model is non-parametric in nature and allows for the model to capture complex, non-linear relationships that are common in NFL datasets.

2) **Quantile Loss (Pinball Loss)**

   It is common for standard or classic models to be optimized using Mean Square Error (MSE) or Mean Absolute Error (MAE) to prediction the conditional mean. For the context of this problem however, we need a different technique, this is where **Pinball Loss** is needed.

   Pinball loss penalizes errors differently baed on each quantile τ. For a given quantile τ, the Pinball Loss for an error ($e_i = y_i - \hat{y_i}$):

$$L_\tau(e_i) = \begin{cases} \tau \cdot e_i, & \text{if } e_i > 0 \\ (1-\tau)\cdot(-e_i), & \text{if } e_i \leq 0 \end{cases}$$

   Pinball loss is designed to be "asymmetric" whenever the quantile is not the median. For example in our case, when training the 90^{th} quantile model ($\tau = 0.90$), the loss function applies a higher penalty when the actual score is greater than the predicted score, therefore forcing the model to overestimate the prediction and push the line higher (in this case the ceiling).

This is core concept of our project, by training three separate HGBR models, each minimizing the Pinball Loss for a specific quantile, we construct an 80% prediction interval and our three trained models are:

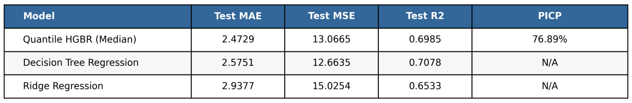
- $\tau = 0.10$ (Floor): Estimates the 10^{th} percentile.
- $\tau = 0.50$ (Median): Estimates the 50^{th} percentile.
- $\tau = 0.90$ (Ceiling): Estimates the 90^{th} percentile.

III. Evaluation and Analysis

The evaluation of the Quantile HistGradientBoostingRegressor (HGBR) model was performed with two metrics in mind; the accuracy of the median prediction mean absolute error (MAE) for comparison to simple benchmarking models and verification of the calibration of the prediction interval coverage probability (PICP).

A. *MAE Evaluation and Benchmarking*

The median model ($\tau = 0.50$) served as the best point-estimate for weekly, with it's performance measured by the MAE. The final HGBR achieved a MAE of 2.47 when evaluated on the held out test data from our chronological train/test split. The result was competitive with other methods as we see in Table 1 below and validates the feature engineering process as well as final features included in the final model form. The HGBR was benchmarked against a standard linear regression with L2 (Ridge) regularization, which served as simple linear model comparison and a standard decision tree regressor, which served as a simple non-linear model comparison.



Model	Test MAE	Test MSE	Test R2	PICP
Quantile HGBR (Median)	2.4729	13.0665	0.6985	76.89%
Decision Tree Regression	2.5751	12.6635	0.7078	N/A
Ridge Regression	2.9377	15.0254	0.6533	N/A

Fig. 1. Benchmarking comparison between HGBR and simple baseline models.

As detailed in the table above, the HGBR demonstrated better performance relative to both the ridge regression and the decsion tree models when in terms of MAE, but performed slightly worse than than the ridge regression in terms of mean squared error (MSE) and R^2. Here we can see that in terms of predicting a mean or median value for a player for a given week, all three models are pretty consistent and gives validity to our HGBR model with respect to making point-estimates. However, this is only one piece of the output to consider for the HGBR as we will see in the following section.

B. *Prediction Interval Calibration (PICP)*

The main metric for this project and one that determines success in prediction is the Prediction Interval Coverage Probability (PICP), which should meet a established target of 80% for the interval as defined by the 10^{th} and 90^{th} quantiles. This can be interpreted as when deploying the model, we expect 80 out of every 100 weekly player scores to fall between the predicted floor and ceiling. If the PICP were too low (e.g. 70%), then the model would be classified as been ”overconfident” and conversely if it were too high, it would be ”over conservative”, providing wide intervals that have little utility for the a fantasy player trying to determine an appropriate risk ranking for their players.

1) **Initial PICP Overconfidence**
   Early model runs revealed large overconfidence with initial performance on the test data set showing a PICP of about 70%. This early overconfidence meant that the predicted interval generated by the model was too narrow, incorrectly excluding the true output in $\sim 10\%$ of the player-weeks despite claiming 80% confidence. This overconfidence required model developers to better optimize the PICP convergence to the target set.

2) **PICP Convergence Optimization**
   To reduce the gap from 70% to the target of 80%, two approaches were used focusing on adjusting the prediction interval width:

   - Introduction of volatility features designed to better signal risk to each of the respective quantile models. `Y_MAE_roll_3` was added to proved a measure of the player's recent unpredictability and `Y_Spike_Rate_5` to capture a player's recent frequency of extreme outcomes (i.e., a high ceiling). Both of these features directly aid in informing the 10^{th} and 90^{th} models when to predict a wider interval to account for any added uncertaity.

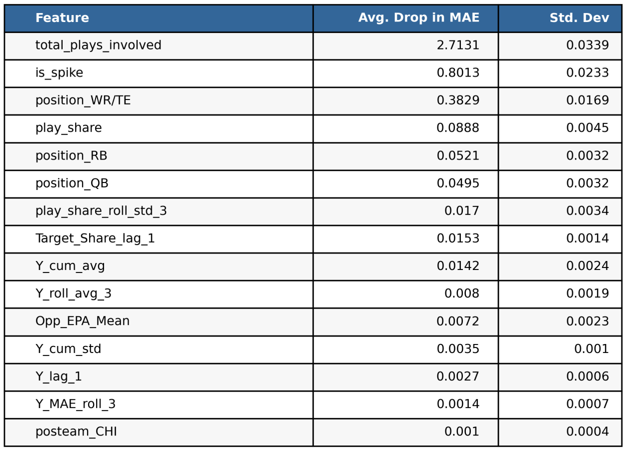
   - Hyperparameter tuning based on our final feature set to control `max_depth`, `learning_rate` and `max_iter` to generalize the models more broadly, preventing them from overfitting to narrower, specific features. By generalizing, this lead to predictions that were slightly more conservative, with wider and better-calibrated prediction intervals.

   The result of these efforts in increasing the PICP lead to the final PCP of $\sim 76\%$, yielding better calibration and converge as it relates to the project's primary objective.

C. *Feature Importance*

Although model accuracy and interval coverage are very important metrics for this project, an analysis of feature importance is also necessary. We employed Permutation Feature Importance (PFI) on the median model to determine the most critical predictive signals and validate their use in our feature engineering.

We decided to use PFI as it works well with any model form since this model isn't the usual classical approach. PFI is used to measure the change in a model's error when the values of a single feature are randomly shuffled across the test set. The way it works is by shuffling a feature, if this causes a large increase in the model's error, it signals that the model relies greatly on that feature for it's prediction, thus confirming it's high importance. Conversely, shuffling with little change can be used to help determine if a feature has little importance. It works well as a more robust measure of generalization importance because it measures the feature's impact on prediction.



Feature	Avg. Drop in MAE	Std. Dev
total_plays_involved	2.7131	0.0339
is_spike	0.8013	0.0233
position_WR/TE	0.3829	0.0169
play_share	0.0888	0.0045
position_RB	0.0521	0.0032
position_QB	0.0495	0.0032
play_share_roll_std_3	0.017	0.0034
Target_Share_lag_1	0.0153	0.0014
Y_cum_avg	0.0142	0.0024
Y_roll_avg_3	0.008	0.0019
Opp_EPA_Mean	0.0072	0.0023
Y_cum_std	0.0035	0.001
Y_lag_1	0.0027	0.0006
Y_MAE_roll_3	0.0014	0.0007
posteam_CHI	0.001	0.0004

Fig. 2. Permutation Feature Importance (PFI) for HGBR model.

The results in Figure 2 support that simple opportunity is the primary driver of fantasy output, as `total_plays_involved` features accounts for the largest share of explained variance in the data. We also see that volatility features `Y_MAE_roll_3` and `Y_cum_std` are lower for the median model but their impact is high on the other quantiles. Remember, these features are primarily used for the 10^{th} and 90^{th} quantiles and their function is not to predict a likely score but to ask as risk multiplier. They signal when when the conditional variance of the outcome is high. These features are also validated in the increase in the target PICP that was discussed in an earlier section. They are crucial for capturing the underlying heteroskedasticity in the data, allowing for intervals in the quantile models to widen with the player is high-risk and narrow when low-risk.

D. Visual Evidence of Heteroskedasticity

The most significant evidence of the model's success in captures the risk spread of players is provided by the visual analysis of the prediction intervals provided through positional quantile plots.

The plot below in Figure 3 shows the quantiles ability to capture most of the data. We see that the predicted median line is closely tracking the actual median line across the entire range of predicted scores and not displaying any systematic bias, as well as only large outliers below and above the 10^{th} and 90^{th} quantiles.

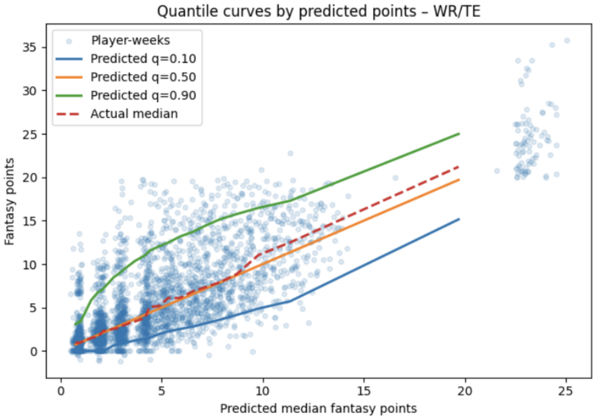


Fig. 3. Quantile Model performance for receiver positions.

The plot also confirm the presence and the model's successful learning of conditional variance. Here we see the distance between the 10^{th} and 90^{th} quantiles widens as the predicted median score increases for each position. This intuitively makes a lot of sense, as players are predicted to score more points, they have a much wider range of possible outcomes compared to a player predicted to only score a few points. We also notice that the interval width for receivers (WR/TEs) is wider reflecting the higher week-to-week volatility of receiving positions due to dependency of targets, game scheme or usage. This showcases the robustness of this model to learn pick up on these positional differences without explicit creation of features or volatility features specific to each position.

In summation, the analysis of this model confirms that the model provides consistent single-point estimates but also generates well-calibrated 80% prediction intervals, expanding a predictive task into a very useful risk-assessment tool for a fantasy players.

IV. Related Work

The current project builds on sports forecasting, machine learning and the quantifying of uncertainty in predictions. Our approach builds upon established methodologies but also diverges to address a specific overlooked gap in fan-based sports analytics.

A. NFL Forecasting and Performance Metrics

Early NFL analytics was focused only the core foundational metrics for player evaluation. Eventually there was a shift from these traditional statistics (e.g., player yards, attempts, etc.) to more advanced metrics that provided a significant lift in the predictive accuracy of player performance. One such metric that was even considered for use in our model was the development of expected points added (EPA) which enhanced play evaluation, provided additional context to player gains and increasing predictive power of models [1].

Most early predictive models still however are focused on minimizing a global error metric, which is typically MSE. Even inclusion of more generalized techniques, including lagging of player features, while establishing a good baseline for accuracy were limited by their linear nature and inability to account for higher volatility in weekly player outputs.

B. Introduction of Non-Linear and Ensemble Techniques

With advancement in machine learning, sports predictions move to more complex, non-learning techniques to utilizes methods such as Gradient Boosting Machines (GBM), including XGBoost and now even the introduction of deep learning to set an industry standard for maximizing predictive accuracy.

There are several recent publications to confirm the superior performance of tree-based methods over linear models in predicting sports outcomes [2]. There are examples supporting that apply GBM to football and basketball outcomes can lead to 10% to 20% improvements in MAE over their linear counterparts. Our decision to utilize HGBR is consistent with this methodology as it not only exploits gains in predictive power of linear models, but due to the novelty of our project as it does not minimizing standard MSE, but instead is utilized for it's ability in quantile estimations.

C. Use of Quantile Regression in Quantifying Uncertainty

The real value in this project has to do the with the expansion of point-estimates by leveraging advanced algorithms to help quantify conditional uncertainty.

The use of Quantile Regression (QR) can be traced back to Koenker and Bassett's 1978 paper which introduced the technique as more robust alternative to mean regression via the use of the pinball loss function [3]. QR has been successfully adopted to a range of fields requiring robust risk management such as energy demand forecast and financial risk modeling, however, it's applied use in the sports field, specifically as it relates to it's application in NFL fantasy forecasting remains much less common compared to mean regression. There have been other attempts to quantify uncertainty which have usually centered around using heteroskedastic linear models or Monte Carlo simulations but have to make general assumptions that residuals follow a normal distribution [4].

This however, is where our model distinguishes itself from those approaches. It is focused on calibration of a prediction interval (in our case, the PICP) and does not need to make any assumptions about underlying data, residuals, etc., as the model is non-parametric in nature. By including features that capture volatility, we are able to better capture an estimate of the risk inherent in player performance to capture most predictions 80% of the time as defined by our model.

V. Discussion and Conclusion

The development and use of the HGBR model successfully satisfied it's objective of providing consistent accuracy for the median and calibration of the prediction interval.

A. Validation of Quantile Regression (QR) Framework

The most important justification for the use of this model was the convergence of the PCP to the 80% target, achieving a final PICP of 76%. This validates the choice of using QR over traditional MAE models. This was accomplished by the training separate quantile models optimized using Pinball Loss, which provided a reliable measure of uncertainty reflects the true stochastic nature of NFL performance. The keys for providing these reliable uncertainty measures were the engineering of volatility features to provide necessary risk signals to the quantile models and generation of quantile models to better capture this volatility for data points further from the mean/median.

B. Performance and Features

As discussed in previous sections, the median prediction was consistent with other baseline methods with a similar MAE. The Permutation Feature Importance (PFI) provided insights into the predictive hierarchy of our features. It was found that `total_plays_involved` was by far the most significant predictor of fantasy performance. This supports the idea that opportunity in fantasy football is a large part of whether or not a player will be successful (i.e., if they aren't on the field, they can't score points). Meanwhile volatility metrics were key in determine the quantiles, specifically spikes in performance and cumulative average as anchor for the floor prediction.

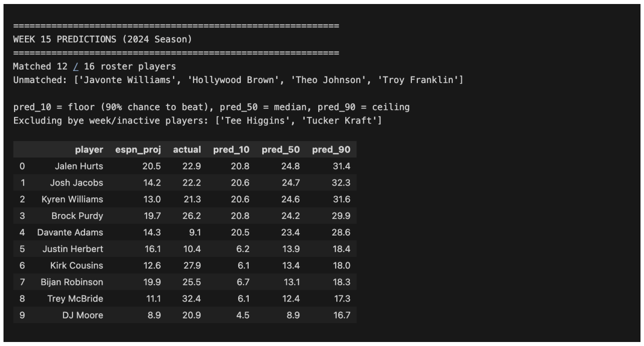
C. Heteroskedasticity

Through visualization, we were able to support the evidence of heteroskedasticity in the data and the model was able to learn that uncertainty is no not constant as it differentiated it between positions. The model is therefore able to perform true probabilistic forecasting by updating it's confidence level based on specific player, matchups and trends on a week-to-week basis.

D. Practical Application Use

Fantasy football players are constantly assessing risk/reward when it comes to whether or not they should start a player for the week, but no meaningful metric to help make this decision. Fantasy players are most concerned with **downside risk** and **upside risk**. Downside risk or in our case the Q_{10} (floor) helps identify players have a floor that is quite low despite a promising mean/median projection which may be helpful in standard fantasy leagues. Upside risk or in our case the Q_{90} (ceiling) highlights players with high upside can be more helpful in ”cash games”.

The table below highlights these points and allows the fantasy manager to think about it less in absolute terms (”Who will score the most points?”) and more in terms of (”Is the added ceiling worth the downside risk?”) after assessing the risk spread. Here we see the model achieves coverage most of the time as the actual score falls within the prediction interval.



```
============================================================
WEEK 15 PREDICTIONS (2024 Season)
============================================================
Matched 12 / 16 roster players
Unmatched: ['Javonte Williams', 'Hollywood Brown', 'Theo Johnson', 'Troy Franklin']

pred_10 = floor (90% chance to beat), pred_50 = median, pred_90 = ceiling
Excluding bye week/inactive players: ['Tee Higgins', 'Tucker Kraft']
```

	player	espn_proj	actual	pred_10	pred_50	pred_90
0	Jalen Hurts	20.5	22.9	20.8	24.8	31.4
1	Josh Jacobs	14.2	22.2	20.6	24.7	32.3
2	Kyren Williams	13.0	21.3	20.6	24.6	31.6
3	Brock Purdy	19.7	26.2	20.8	24.2	29.9
4	Davante Adams	14.3	9.1	20.5	23.4	28.6
5	Justin Herbert	16.1	10.4	6.2	13.9	18.4
6	Kirk Cousins	12.6	27.9	6.1	13.4	18.0
7	Bijan Robinson	19.9	25.5	6.7	13.1	18.3
8	Trey McBride	11.1	32.4	6.1	12.4	17.3
9	DJ Moore	8.9	20.9	4.5	8.9	16.7

Fig. 4. Comparison of Quantiles with ESPN projection showcasing risk spread for each player.

E. Limitations and Future Work

While the model achieved its primary goals, there were a few limitations that need to be fixed in the future, specifically as it relates to getting the PICP to converge more to 80%. Some potential avenues to explore would be:

- **Conditional Regularization**: Using different complexities (max depths) for the median versus the other quantile models to better strategically influence the width of the interval with the objective of better capturing more of the data.

- **Positional Specialization**: The current model uses a single matrix for all positions. A more advanced approach would be to train separate HGBR models for each position, allowing for feature weights and volatility parameters to be more specialized for each role.

- **Interaction Effects**: The model currently relies on the underlying tree structures to capture complex interactions. Future consideration would be to engineer interaction features to provide more direct signals to the HGBR model, potentially reducing any noise and improving our error metric.

F. Conclusion

This project successfully utilized the Quantile HistGradientBoostingRegressor (HGBR) in Python to expand on the standard practice of single-point estimation to create reliable probabilistic risk modeling. By generating additional features based on optimization prediction intervals the model achieved a level predictive accuracy consistent with standard baseline models but also the calibration required to create prediction intervals that contain our intended targe of 80% of the data. The final model provides a robust, data-driven framework for accessing the inherent risk in a player’s expected performance and offering fantasy players significant utility for making strategic decisions as relates to setting a lineup to meet their risk appetite.

VI. Reflections

The development the HGBR model for NFL fantasy forecasting provided valuable insights into advanced sports predictive analytics, model building and calibration and the challenges of working with high-dimensional, time-series data.

A. Key Takeaways

- The need for a more probabilistic approach to forecasting player performance. While there is a convenience to having only a mean-based prediction, in a highly volatile domain like NFL scoring, it makes it very difficult to make a well-informed decision. By constructing a PICP, this provides fantasy palayers with a far more information and ability to make a sound decision by accounting for risk.

- The creation of more advanced features in addition to base features included in a dataset. Even though the approach adopted is an advanced tree-based algorithm, it still suffered from substandard performance until features were created to account for workload, volatility and lags in performance.

B. Major Challenges

- Data leakage from inclusion of features that perfectly predict the target score. The large NFL data set used contained performance metrics that are directly used in the calculation of the target variable and as a result had to be omitted. In their place, lagged metrics were needed to help capture a player’s performance.

- Persistent tendency of the HGBR model to be overconfident (which is shown by the PICP being less than 80%). We tried hyperparameter tuning, the inclusion of several more engineered features, but still unable to raise the PICP closer to the goal target. As discussed in prior section, there may need to be inclusion of interaction terms and potentially separate models for each respective position.

In conclusion, a lot of good work was done to generate a model that adequately handles the inherent risk in NFL data. However, if we had more time I think there are plenty of opportunities to improve upon the existing framework. Such as generation of a more interactive decisioning tool by allowing player to set their risk level and therefore generate custom prediction intervals or even provide feedback on the best roster to start based on risk setting. In addition, as stated before, specialized models for each position as well.

Our team set out to find a novel approach to help fantasy players address risk in decision making and while this framework does check a lot of boxes, it has plenty of room to be improved and made much better in the future.

References

[1] Burke, E. and E. S. K. E. S. S. J. (2018). The Hidden Game of Football: On the Statistics of the NFL.

[2] Luo, J. (2020). Predicting NBA Player Efficiency Using Gradient Boosting. Journal of Sports Analytics, 6(3).

[3] Koenker, R. and Bassett, G. (1978). Regression Quantiles. Econometrica, 46(1).

[4] Gros, C., et al. (2021). Quantifying Predictive Uncertainty in Football Match Outcomes. Journal of Sports Sciences.