Presentation Script
Weekly NFL Fantasy Football Point Forecasting Using Gradient Boosting Models

AAI 595 Applied Machine Learning
Stevens Institute of Technology

Overview

Total Duration: Approximately 7 minutes
Speakers: Three speakers rotating throughout

1 Slide 1: Title & Context

Estimated time: 45 seconds

Speaker 1

Good afternoon. Today we're presenting our project on weekly NFL fantasy football point forecasting using gradient boosting models.

The challenge is predicting how many fantasy points a player will score in their next game—week by week, not season-long. This is difficult because weekly performance is highly variable. A running back might score 20 points one week and 3 the next, even with similar usage.

Our goal was to build a machine learning system that beats simple heuristics like rolling averages. We used NFL play-by-play data from 2009 to 2016, engineered 51 time-aware features, and trained three regression models.

[Advance to Slide 2]

2 Slide 2: Why Weekly Fantasy Forecasting Is Hard

Estimated time: 50 seconds

Speaker 1

So why is this hard? Three main reasons.

First, extremely high week-to-week variance. A single game can swing wildly based on unpredictable events—a defensive breakdown, a fluky touchdown, bad weather.

Second, performance is context-dependent. Opponent strength, injuries, weather, and game script all affect outcomes in ways historical averages don't capture.

Third, limited data per player. Only 16 or 17 games per season—a small sample compared to other forecasting domains. For rookies or players changing teams, even less.

This means we need time-aware models that capture recent trends without overfitting—that's where machine learning comes in.

[Speaker 2 takes over]

[Advance to Slide 3]

3 Slide 3: Data & Pipeline Overview

Estimated time: 60 seconds

Speaker 2

Let me walk you through our data pipeline.

We started with the NFL play-by-play dataset from Kaggle, covering 2009 through 2016. This dataset contains every single play—every pass, rush, and sack—with detailed statistics like yards gained, touchdowns, and turnovers.

We filtered this down to fantasy-relevant plays and aggregated them to the player-week level. So instead of individual plays, each row represents one player's performance in one week. We then computed fantasy points directly using Half-PPR scoring rules—that's half a point per reception, which is a standard scoring format in fantasy football.

From there, we built 51 features. These include lag variables like "points scored last week," rolling averages over 3, 5, and 8 weeks to capture trends, season-to-date aggregates, and usage metrics like touches and targets. Critically, all features use only historical information. We never include current-week stats in the feature set—everything respects temporal ordering.

In total, we have about 40,000 player-week observations spanning eight seasons. This gives us a rich dataset for training time-series regression models.

[Advance to Slide 4]

4 Slide 4: Methodology: Features, Split, Baselines

Estimated time: 60 seconds

Speaker 2

Our methodology follows a rigorous time-aware evaluation framework to prevent data leakage.

On the left, you can see our feature types. Lag variables capture what happened most recently—last week's points, yards, and touchdowns. Rolling averages smooth out noise by averaging performance over the last 3, 5, or 8 weeks. Season-to-date aggregates track cumulative performance from the start of the season up through the previous week. We also engineered trend indicators to measure momentum—is a player improving or declining recently? And usage metrics quantify player role—how many carries, targets, or attempts they're getting.

Crucially, these features are strict historical-only. We never peek at the current week we're trying to predict.

On the right is our time-based split. We train on 2009 through 2014 data—six full seasons. We validate on 2015 to tune hyperparameters. And we test on 2016, a completely held-out season the model has never seen. This mimics real-world forecasting: you train on the past and predict the future.

We also built three simple baselines: using last week's score, a 3-week rolling average, and a 5-week rolling average. These represent typical heuristics fantasy managers actually use, so beating them is a meaningful benchmark.

[Speaker 3 takes over]

[Advance to Slide 5]

5 Slide 5: Results: Model Comparison

Estimated time: 60 seconds

Speaker 3

Now let's look at results.

This chart shows mean absolute error—the average prediction error in fantasy points—for all models on the 2016 test set.

The three baselines are on the right. "Last week" performs worst with an MAE over 5.6

points. The 3-week and 5-week rolling averages do better—around 4.9 and 4.8 points respectively. These are decent heuristics, but they leave a lot of room for improvement.

Our three machine learning models—Ridge Regression, Random Forest, and Gradient Boosting—all substantially outperform the baselines. Ridge, the simplest linear model with regularization, achieves an MAE of 3.19. Random Forest, an ensemble of decision trees, gets down to 2.69.

But the best model is Histogram Gradient Boosting, shown in green, with a test MAE of 2.62 fantasy points. That's 45% better than the best baseline.

To put that in context, the Gradient Boosting model also achieves an R-squared of 0.726, meaning it explains nearly 73% of the variance in weekly fantasy points. For a noisy, high-variance time series like this, that's a strong result.

[Advance to Slide 6]

6 Slide 6: What the Model Learned

Estimated time: 50 seconds

Speaker 3

One of the advantages of gradient boosting is interpretability. We can ask: what features does the model think are most important?

This chart shows the top 10 features by importance. The most predictive feature is the 3-week rolling average of fantasy points—recent sustained performance is the strongest signal. The 5-week and 8-week rolling averages are also near the top.

Notice that raw lag variables—like last week's score—rank lower. The model learned that smoothed trends are more informative than single-week spikes. A player who's consistently scored 15 points over the last month is more predictable than one who scored 30 last week but 5 the week before.

Our engineered rolling features are doing the heavy lifting.

[Advance to Slide 7]

7 Slide 7: Player-Level Example

Estimated time: 45 seconds

Speaker 3

Let's make this concrete with a player-level example.

This shows Aaron Rodgers' 2016 season—actual fantasy points in blue, model predictions in orange.

The model tracks the general trend well. When Rodgers has a consistent stretch, predictions stay close to reality. The model captures his typical performance range.

But look at the outliers. Week 3, Rodgers scores over 40 points—the model underestimates that. In unusually low weeks, it sometimes overestimates.

This illustrates a key limitation: the model smooths toward expected values. It's good at predicting typical performance but can't anticipate extreme outliers. That's an inherent challenge in weekly forecasting.

[Speaker 1 takes over]

[Advance to Slide 8]

8   Slide 8: Limitations & Wrap-Up

Estimated time: 60 seconds

Speaker 1

To wrap up, let's discuss what this system is good for and where it falls short.

This model is useful for weekly lineup decisions. If you're managing a fantasy team and deciding between two players, our system gives you a data-driven forecast based on recent trends and historical performance. It consistently outperforms naive heuristics in realistic forecasting scenarios—meaning it respects time ordering and only uses past information.

However, there are important limitations. The model doesn't incorporate contextual features like opponent strength, injury reports, or weather. Those factors clearly matter, but they're not in our current feature set. We also have missing Week 1 data for several seasons, which limits early-season predictions. And like any statistical model, it struggles with true outliers—games where something unusual happens that isn't predictable from history.

For future work, we'd want to add matchup-specific features, integrate injury and weather data, and explore ensemble approaches that combine multiple models. We might also look at multi-step forecasting—predicting not just next week, but the next several weeks.

Overall, this project demonstrates that machine learning can substantially improve weekly fantasy forecasting over simple baselines, while still remaining interpretable and practically useful.

[Pause for questions]

Thank you.