Advanced Programming 2025

Analyzing Performance Drivers Across Golf's Major Championships: A Multi-Method Econometric and Machine Learning Approach

Final Project Report

Luciana Piña Strzelecki
luciana.pinastrzelecki@unil.ch
Student ID: 20382701

January 11, 2026

Abstract

This project analyzes which performance skills drive success in the four golf Major Championships: The Masters, PGA Championship, U.S. Open and The Open Championship. The study combines econometric and machine learning models to examine scoring outcomes and top-25% finishes, using tournament data from DataGolf covering the period between 2020 and 2025. Linear and logistic regression models are used to interpret how performance metrics relate to total score and leaderboard success, in general and by tournament. Random Forest and XGBoost classifiers are applied to predict top-25% finishes, with SHAP values providing insights into feature importance. The results show strong and consistent performance across approaches: the linear regressions achieve R^2 values between 0.81 and 0.94, the logistic regressions attain ROC-AUC scores greater than 0.99 and the machine learning models reach ROC-AUC scores between 0.95 and 0.98. Strokes gained approach, putting, and around-the-green appear to be the most important skills across methods, with small differences across Majors. Overall, the results suggest that balanced skill profiles are essential to succeed in Major Championships and that combining econometric interpretation with machine learning prediction provides clear insights.

Keywords: data science, Python, machine learning, SHAP, econometric models, golf analytics, strokes gained, sports analytics

Contents

1 Introduction

Background and motivation

Golf is an individual sport played in professional tours, the most important being the one from the U.S., the PGA Tour. This project focuses on the four Major Championships, the most prestigious tournaments in men's professional golf.

The Four Major Championships

- **The Masters:** Played every year at Augusta National (Georgia USA). Very exclusive and small, golfers can only assist if invited or if they fulfill certain criteria. Played in April.
- **PGA Championship:** Played in different U.S. courses each year. Limited to professional golfers (mainly PGA tour players, Major winners, and top-ranked players). Played in May.
- **U.S. Open:** Played in rotating difficult courses across the USA. Called Open because any player (professional or amateur) can try to qualify. Played in June.
- **The Open Championship:** Played on coastal courses in the UK and Ireland. Players from all major professional tours from around the world can qualify. Played in July.

Each Major has unique characteristics that reward different skills. This project aims to analyze which skills matter the most at each tournament.

To be able to understand this project, it is essential to introduce how scoring works. In golf, lower scores mean better performance. The score represents the number of strokes per round, each round has 18 holes and tournaments consist of four rounds. The expected number of strokes for a course is called "par" (typically 70–72). Scores are reported relative to par (for example, -2 means two strokes better than expected). This project uses total tournament scores.

Broadly speaking, performance metrics fall into two groups: traditional metrics such as driving distance and accuracy (percentage of fairways hit), and strokes gained metrics, which measure the number of strokes a golfer gains or loses relative to the field average in areas like putting, approach play and around the green [8]. Positive strokes gained means the golfer performs better than the field. All metrics are described in more detail later in the report.

Problem statement

While the main objective in golf is straightforward: achieving the lowest possible score, different factors contribute to finishing near the top of the leaderboard, and some may vary across tournaments. Since each Major has its own characteristics, performance patterns may matter more in one tournament than in another. Therefore, this paper aims to identify which performance metrics drive success overall and in each of the four Major Championships.

Objectives and goals

This project aims to quantify the relationship between performance metrics and scoring using econometric models, identify tournament specific skill importance across the four majors, build predictive models to identify top-25% finishers, provide interpretable insights using SHAP values and validate the results across statistical and machine learning frameworks.

Report organization

The report first reviews the related work and background. It then describes the data, feature engineering and modeling methodology. Next, the results are presented and discussed together with their implications, challenges and limitations. Finally, the report concludes with a summary and directions for future work.

2 Literature Review

Most existing research in golf analytics is mainly exploratory and statistical, though more recently machine learning has been applied. Nevertheless, these papers focus on predicting different outcomes such as earnings, match results or winning scores, rather than identifying which performance skills drive success at Major Championships.

A fundamental concept of modern golf analytics are the strokes gained metrics, introduced by Mark Broadie [2, 1], which demonstrate that traditional statistics such as fairways hit fail to provide consistent skill measures because they aren't comparable across different game aspects. Strokes gained resolves this by expressing all performance components relative to a benchmark, allowing off-the-tee, approach, short game, and putting to be evaluated on a common scale.

Several studies use strokes gained for prediction but with different objectives. DataGolf [4] applies logistic regression to match-play outcomes, finding that strokes gained putting and driving predict match winners with 60% accuracy, though using only these two variables. Li [7] predicts annual FedExCup earnings using machine learning models on PGA Tour data, but earnings reflect season-long performance rather than tournament-specific skill importance. DataGolf [3] models tournament outcomes via OLS regression on adjusted scoring averages, though without detailed skill variables.

Two machine learning studies are closer in approach but still differ in focus. Klassen [6] applies regression models and XGBoost to predict golfer scores with temporal validation, achieving an AUC of around 0.60. Wiseman [9] predicts winning scores using tree-based models (R^2 between 0.42 and 0.59), but focuses on betting objectives using event variables like course par and prize money rather than player performance metrics.

While existing literature validates strokes gained as a performance measure and demonstrates that regression and machine learning work well for golf analytics, previous studies differ in their targets and scope. This project addresses a gap in the literature by identifying which skills matter most in Major Championships and how their importance varies across tournaments.

3 Methodology

3.1 Data Description

Source and collection method

The data was collected from the DataGolf website [5] (Scratch Plus subscription) using the detailed tournament stats pages filtered by Major and year. CSV files were manually downloaded for each Major Championship: The Masters, PGA Championship, U.S. Open and The Open Championship, covering the period from 2020 to 2025, as API access to historical data required a higher tier subscription.

The raw data includes player identifiers and outcome variables (name, finishing position, total score and round 4 score), along with a set of performance variables and strokes gained metrics.

Size and characteristics

The initial sample sizes (before data cleaning) for each Major between 2020 and 2025 are as follows (in player numbers): PGA Championship: 935 player observations (all years available), The U.S. Open: 924 observations (all years available), The Masters: 448 observations (2020 unavailable) and The Open Championship: 624 observations (2020 and 2021 unavailable). The total initial sample size is 2,931 observations.

Performance variables

As mentioned earlier in the report, golf performance metrics can be grouped into two categories: The first group consists of traditional metrics including average driving distance, driving accuracy (fairways hit percentage), greens in regulation (percentage of holes reaching green in expected number of strokes), proximity from the fairway, proximity from the rough, scrambling (percentage of times saving par after missing the green), great shots and poor shots.

The second group consists of strokes gained metrics, which measure performance relative to the field average, where positive values indicate above average performance. These include strokes gained off the tee (driver performance), approach game (quality shots into the green), around the green (short game performance: chips, bunker shots...) and putting. Composite strokes gained metrics are also included such as strokes gained total (sum of all SG components), tee to green (off the tee, approach and around the green) and ball striking (off the tee and approach).

Data quality issues

Although the dataset was extracted from a reliable source, there is one main constraint: the availability of data varies across tournaments and years. The Masters data is missing for 2020 and The Open Championship data is missing for 2020 and 2021 (2020 is missing because it was cancelled). Additionally, some performance variables are unavailable for the U.S. Open in 2022.

3.2 Approach

Algorithms used

This project combines econometric models and machine learning models to analyze golf performance. The econometric models are mainly used to interpret and explain which skills drive performance, while the machine learning models are used to capture non-linear relationships and predict which players finish in the top of the leaderboard given their performance.

Econometric models

The pooled linear regression is used to explain total score across all Majors and separate linear regressions per Major are estimated to compare skill importance across tournaments. The pooled logistic regression is used to model the probability of finishing in the top 25% of the four tournaments in general and an extended version with interaction terms is added to show how the importance of the performance variables change per Major Championship.

Machine Learning models

The Random Forest and XGBoost classifiers are applied to predict top-25% finishers. These models are chosen because they can capture complex relationships while allowing us to understand which variables are the most important, using feature importance and SHAP values.

Data preprocessing steps

There are a few steps to follow before modeling. First, players who didn't complete all four rounds, because they missed the cut (only a certain number of players continue after the first two rounds), withdrew or were disqualified (CUT, WD or DQ), are removed to ensure comparable performance. Second, the U.S. Open 2022 is excluded due to missing performance variables. Third, a year column is added to each Major's dataset. Finally, all Majors and years are combined into a single dataset, and a Major column is added so tournaments can be distinguished.

Before running the models, an exploratory analysis was carried out to understand the distribution of the variables, detect outliers and observe relationships between performance metrics and scoring. This step helped validate the cleaning process and provided intuition for later modeling.

For the linear regression models, the target variable is the total tournament score. For the logistic regression and machine learning classification models, the target variable is a binary indicator identifying players finishing in the top 25% of the leaderboard within each tournament (by Major and year), based on the distribution of total scores.

Only performance variables are used for modeling. Player identifiers, finishing position and round scores are excluded to avoid data leakage. Year is used only for grouping and validation. All performance variables are standardized so that metrics measured on different scales can be compared. Composite strokes gained metrics (SG total, tee to green and ball striking) are excluded to avoid multicollinearity, as they are combinations of other variables already included.

Model architecture

The econometric models are estimated using standardized inputs in linear or logistic regressions, which allows a direct interpretation of the model coefficients. Pooled models assume common effects across Majors, while the per-Major regressions and interaction terms make it possible to see how the importance of each skill changes from one tournament to another.

The machine learning models are tree-based classifiers that combine multiple decision trees in order to capture non-linear relationships between the performance variables and the tournament outcomes (top 25% finishers). Furthermore, the best performing model results are complemented with a SHAP value analysis to better understand the contribution of each variable.

Evaluation metrics

The linear regression models are evaluated using R^2, adjusted R^2 and Root Mean Squared Error (RMSE) and the logistic regression using accuracy, precision, recall, F1 score and ROC-AUC.

For the machine learning models, a temporal validation approach is used: data from 2020 to 2024 is used for training, while 2025 data is held out for testing to avoid data leakage. These models are evaluated using test accuracy, overfitting gap, precision, recall, F1 score and ROC-AUC. Confusion matrices are also presented.

3.3 Implementation

Programming languages and libraries

The project is implemented in Python 3.11. The environment is managed with Conda via `environment.yml`, with Python package dependencies specified in `requirements.txt`. The main libraries used are `pandas` and `numpy` for data handling, `matplotlib` and `seaborn` for visualizations, `scikit-learn` for model training and evaluation, `statsmodels` for the econometric models, `xgboost` for the XGBoost classifier, `scipy` for mathematical and statistical functions and `shap` for computing SHAP values for machine learning model interpretation.

System architecture and key code components

The project is organized as a modular pipeline, where each step of the analysis is handled by a separate Python script inside the `src/` folder. The process is executed through the file `main.py`. The workflow runs in the following order: data loading, exploratory analysis, feature engineering, econometric models, machine learning models, evaluation and results. The outputs generated at each step are saved in the `results/` folder.

The preprocessing steps described in Section 3.2 are implemented through the scripts inside the `src/data_preparation/` folder. These scripts take the raw CSV files from the `data/raw/` folder, apply the filtering and modifications (as removing players who didn't finish, excluding the 2022 U.S. Open and adding the year and Major columns) and save one dataset per Major including all years, as well as a combined dataset with all Majors and years.

`data_loader.py` loads these processed datasets from the `data/processed/` folder, `exploratory.py` runs descriptive and exploratory analysis, and `feature_engineering.py` prepares the features, standardizes variables and creates the top-25% target. `The Econometric_models.py` handles the linear and logistic regression models. The `ML_models.py` implements the Random Forest and XGBoost classifiers, and computes the SHAP values. `evaluation.py` computes the performance metrics of all models and saves the results, `visualization.py` generates the figures and finally, `main.py` coordinates the entire pipeline so that each component can be modified or re-run independently without changing the rest of the code.

Project structure

The project is executed from `main.py`. All code is inside `src/`, which contains modules for data preparation, exploratory analysis, feature engineering and the modeling pipeline (`models/` for econometric and ML models). Datasets are stored in `data/` (raw and processed files) and all outputs are saved in `results/`. A detailed folder layout is provided in Appendix B.

Example code snippet 1: Top 25 Target

This function creates the binary target variable for classification models. For each tournament (by Major and year), it calculates the 25th percentile of total score and marks players at or below that threshold as top-25% finishers.

```
def create_top25_target(df):
    """Create binary target indicating top 25% finish per tournament."""
    df = df.copy()
    df["tournament_25th_percentile"] = (df.groupby(["major", "year"])["total_score"]
          .transform(lambda x: x.quantile(0.25)))
    df["top_25"] = (df["total_score"] <= df["tournament_25th_percentile"]).astype(int)

    return df
```

Example code snippet 2: SHAP by Major

This function computes SHAP values separately for each Major Championship to analyze tournament specific feature importance. For each Major, it filters the test set, computes SHAP values and returns the mean absolute SHAP contribution.
Note that the full implementation includes error handling, SHAP output handling and table construction, this snippet shows only the core logic for clarity.

```
def compute_shap_per_major(model, X_test):
    """Compute SHAP values separately for each Major to identify tournament-specific
    feature importance."""
    explainer = shap.TreeExplainer(model)
    per_major_results = {}

    majors = ["The Masters", "PGA Championship", "The Open Championship", "US Open"]

    for major in majors:
        major_col = f"major_{major}"
        major_mask = X_test[major_col] == 1
        X_test_major = X_test[major_mask]

        shap_values_major = explainer.shap_values(X_test_major)
        mean_abs_shap = np.abs(shap_values_major).mean(axis=0)

        per_major_results[major] = {"shap_values": shap_values_major,
        "shap_importance": mean_abs_shap}

    return per_major_results
```

4 Results

4.1 Experimental Setup

Hardware specifications

The project runs locally via `main.py` and the full workflow runs on a standard CPU only laptop.

Software versions

The pipeline was run in Python 3.11. The environment matches the versions specified in `requirements.txt`, with pandas $\geq$ 2.0.0, numpy $\geq$ 1.24.0, matplotlib $\geq$ 3.7.0, seaborn $\geq$ 0.12.0, scikit-learn $\geq$ 1.3.0, xgboost $\geq$ 2.0.0, statsmodels $\geq$ 0.14.0, scipy $\geq$ 1.11.0 and shap $\geq$ 0.42.0.

Hyperparameters

The econometric models are not tuned, as they are used only for interpretation. The machine learning models are tuned using GridSearchCV with 5-fold cross-validation and ROC-AUC as the selection criterion. Random Forest is tuned over max depth, minimum samples per split and leaf, and feature subsampling, while XGBoost is tuned over max depth, child weight, subsample and column subsample parameters. The random state is fixed at 42 to ensure reproducibility.

4.2 Exploratory Data Analysis

The dataset has 1,384 player-tournament observations across the four Majors and around 28% of players fall into the top-25% group (consistent with the target used later). The scoring patterns differ by tournament: The Open Championship has the lowest winning scores, while the U.S. Open is the toughest, since it has the highest scores.

Looking at how performance metrics relate to scoring, the correlation heatmap (Figure A1) shows that better scores are mainly linked to greens in regulation (GIR), scrambling and SG Approach (excluding composite metrics). To put these patterns into context, Figures A2 and A3 compare how the same metrics vary across tournaments. The standardized z-score plot highlights the differences in average skill profiles between Majors and the boxplots show the spread and variability within each event. Finally, when comparing top-25% players with the rest of the field (Figure A4), the largest positive gaps appear in SG Approach, SG Putting and GIR, while poor shots have the strongest negative effect. Therefore, these are the metrics that most clearly separate the highest performing golfers from the rest.

4.3 Econometric Models Results

This subsection presents the performance evaluation and results of the econometric models, which are mainly used for interpretation. The main goal of these models is to understand which are the performance metrics that are most strongly associated with scoring and finishing in the top of the leaderboard, both overall and across the four Major Championships.

Performance evaluation:

Table 1: Linear Regression Models Performance

Model	R^2	Adj. R^2	RMSE (strokes)	N
Pooled Linear Regression	0.811	0.809	3.118	1384
The Masters Linear Regression	0.920	0.916	1.593	272
PGA Championship Linear Regression	0.805	0.800	2.958	466
U.S. Open Linear Regression	0.904	0.901	1.990	337
The Open Championship Linear Regression	0.935	0.932	1.868	309

All per-Major models explain better scoring than the pooled model, showing that the skill effects differ across tournaments.

Table 2: Logistic Regression Models Performance

Model	Accuracy	Precision	Recall	F1	ROC-AUC
Pooled Logistic Regression	0.965	0.953	0.924	0.938	0.996
Logistic Regression (Interactions)	0.970	0.971	0.924	0.947	0.997

Interaction terms improve predictive performance, confirming tournament specific effects.

Results:

Table 3: Linear Regression Models: Top 3 Performance Metrics by Coefficient Magnitude

Model	Top 3 Metrics
Pooled Linear Regression	Greens in regulation, SG Putting, Scrambling
The Masters Linear Regression	SG Putting, SG Around-the-Green, SG Approach
PGA Championship Linear Regression	Greens in regulation, SG Putting, SG Around-the-Green
U.S. Open Linear Regression	SG Putting, Driving distance, SG Approach
The Open Championship Linear Regression	SG Putting, Greens in regulation, Driving distance

All coefficients are significant at the 5% level ($p < 0.05$). For the variables listed in the table, coefficients are negative, meaning better performance is associated with lower score. The pooled and per-Major coefficient visualizations are shown in Figures A5 and A7.

Table 4: Logistic Regression Models: Top 3 Predictors by Coefficient Magnitude (Pooled Model and Model with per-Major interaction terms).

Model	Top 3 Predictors
Pooled Logistic Regression	SG Putting, SG Approach, SG Around-the-Green
The Masters	SG Putting, SG Approach, SG Around-the-Green
PGA Championship	SG Approach, SG Putting, SG Around-the-Green
U.S. Open	SG Around-the-Green, SG Putting, SG Approach
The Open Championship	SG Approach, SG Putting, SG Around-the-Green

All predictors are significant at the 5% level ($p < 0.05$). For variables in the table, coefficients are positive, meaning better performance increases the probability of finishing in the top 25%. The pooled and interaction-term coefficient visualizations are shown in Figures A6 and 1.

4.4 Machine Learning Results

This subsection includes the evaluation and interpretation of the results of the machine learning models. The goal is to compare predictive accuracy across models and to identify the most influential performance metrics, including SHAP values for the best performing model.

Performance evaluation:

Table 5: Machine Learning Models Performance

Model	Accuracy	Precision	Recall	F1	ROC-AUC	Overfitting Gap
Random Forest	0.810	0.930	0.460	0.615	0.949	0.130
XGBoost	0.909	0.920	0.793	0.852	0.978	0.088
Baseline	0.720	–	–	–	–	–

Both models clearly outperform the 72% baseline, which corresponds to always predicting that a player doesn't finish in the Top-25%. XGBoost clearly outperforms Random Forest here. The confusion matrices for both models can be found in Figure A8.

Results:

Table 6: Machine Learning Models: Top 3 Predictors by Feature Importance and SHAP Values

Model	Top 3 Predictors
Random Forest (Feature Importance)	SG Approach, SG Putting, SG Off-the-Tee
XGBoost (Feature Importance)	SG Approach, SG Putting, Poor Shots
XGBoost (SHAP)	SG Approach, SG Putting, SG Around-the-Green

SHAP values are computed for the best performing model (XGBoost). Higher values indicate a greater contribution to the probability of finishing in the top 25%.
The corresponding feature importance visualization can be found in Figure 3.

Table 7: XGBoost Model: Top 3 Predictors by SHAP Values per Major

Major	Top 3 Features (SHAP value)
The Masters	SG Approach, SG Putting, SG Around-the-Green
PGA Championship	SG Approach, SG Putting, SG Around-the-Green
U.S. Open	SG Approach, SG Putting, SG Around-the-Green
The Open Championship	SG Putting, SG Approach, SG Around-the-Green

SHAP values are computed per Major (XGBoost model). Higher values indicate features that contribute more to the probability of finishing in the top 25% within that Major.
The per-Major SHAP visualization that corresponds to these results is shown in Figure 2.

4.5 Visualizations

Econometric models

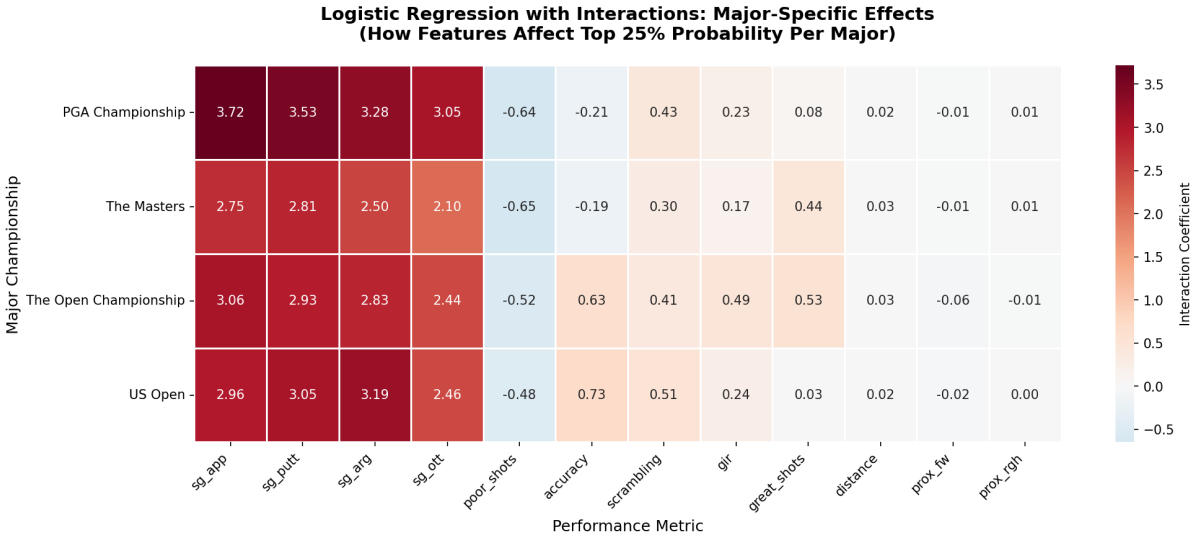

Figure 1: Logistic regression with interaction terms. Values show Major specific effects on the probability of finishing in the Top-25%, positive coefficients indicate a higher probability.

Machine learning models

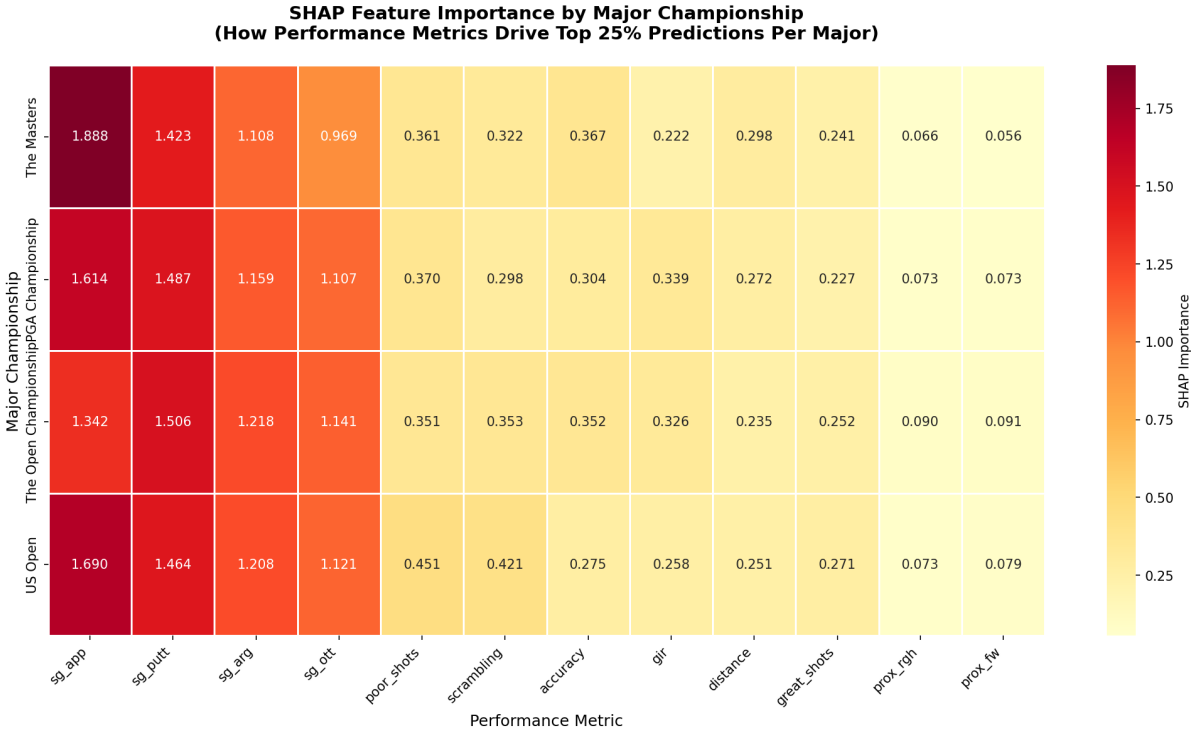

Figure 2: Per-Major SHAP feature importance (XGBoost model). Values show how much each performance metric contributes to predicting top-25% finishes in each Major Championship.

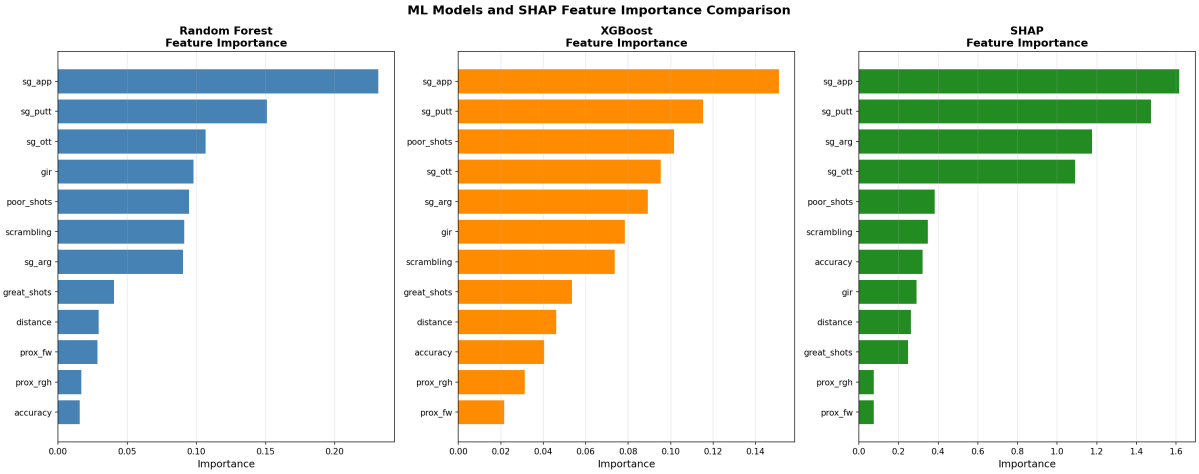

Figure 3: Random Forest, XGBoost and SHAP feature importance comparison for predicting top-25% finishers. The three methods consistently highlight SG Approach and SG Putting as the most influential skills.

5 Discussion

5.1 What Worked Well

The econometric models performed surprisingly well, exceeding initial expectations. The linear regressions achieved R^2 values between 0.81 and 0.94, while the logistic regressions achieved ROC-AUC scores between 0.996 and 0.997. These results substantially exceed the project proposal's conservative estimates (R^2 between 0.3-0.5 and ROC-AUC > 0.7), which assumed greater noisiness in golf data.

All per-Major linear regression models outperformed the pooled model, with higher R^2 and lower RMSE values. Similarly, the logistic regression with interaction terms performed better than the pooled logistic model, with equal or higher scores across all evaluation metrics. Taken together, these results suggest that skill effects differ across tournaments.

For the linear regressions, Greens in Regulation, SG Putting, and Scrambling are the main contributors to explaining total score in the pooled model. However, the per-Major models emphasize different combinations: although SG Putting remains consistent, Greens in Regulation, Driving distance, SG Approach and SG Around-the-Green appear as recurring factors.

For the logistic regressions, SG Putting, SG Approach, and SG Around-the-Green are the top three predictors of finishing in the top 25%. The interaction model shows that these same skills are important across all Majors, but not always in the same order, indicating that while these matter everywhere, the most rewarded skill changes by tournament.

The machine learning models also performed strongly, exceeding the initial ROC-AUC target of 0.7 from the project proposal, and both Random Forest and XGBoost outperformed the baseline model (always predicting non-top-25%). XGBoost delivered the best results overall, with higher recall, F1 and ROC-AUC, and a smaller overfitting gap. Its moderate overfitting gap (8.8%) remains acceptable given the noisy nature of golf data and the limited temporal test window.

The XGBoost feature importance and SHAP results align closely with the logistic regression findings. Approach play, putting and short-game performance appear repeatedly as the most influential skills. Furthermore, Poor Shots appears among the top three features in XGBoost, highlighting the importance of avoiding mistakes to finish strongly.

Looking at the SHAP results by tournament (Table 7): The Masters, the PGA Championship and U.S. Open all share the same top three performance metrics, in the same order: SG Approach, SG Putting and SG Around-the-Green. However, The Open Championship shows a slightly different order, with SG Putting moving ahead of SG Approach as the most important skill. This is surprising, since The Masters would seem the most likely Major for putting to dominate, given its reputation for very demanding greens. However, the stronger role of putting at The Open Championship is consistent with the nature of links-style courses (traditional coastal courses commonly used in the UK and Ireland), which tend to have firm greens exposed to wind.

5.2 Comparison to Existing Literature

The results support Broadie's [2, 1] strokes gained framework, which proposes these metrics as a more comparable measure of skill than traditional statistics. In this project, the most influential variables across all models are primarily strokes gained metrics, consistent with DataGolf [4] and Li [7], which found similar predictive power.

Compared to prior ML studies, this project achieves stronger predictive performance. Klassen [6] achieves AUC around 0.60 for score prediction, while Wiseman [9] reaches R^2 between 0.42-0.59 for winning scores. In contrast, the machine learning models in this project reach ROC-AUC scores of 0.95 and 0.98, and the linear regressions achieve R^2 values between 0.81 and 0.94.

A key reason for this difference is that these studies predict different types of outcomes. Prior work focused on exact scores or betting outcomes for individual players or targets, which are highly sensitive to course conditions, weather, and randomness. In contrast, this project uses linear regressions to identify which metrics drive scoring and classification models to predict top-25% finishes. This target is more stable, as it reflects relative performance within a tournament rather than exact scores, which tend to vary more with conditions and randomness.

Additionally, the data structure contributes to the stronger performance. This project combines strokes gained and traditional metrics, focuses only on Major Championships (maintaining a homogeneous context) and uses tournament-level performance variables directly linked to outcomes. These choices explain why the models perform better than the ones in prior work.

5.3 Challenges and Limitations

The main limitation was data availability. Missing years for The Masters (2020) and The Open Championship (2020–2021), together with unavailable variables for the 2022 U.S. Open, reduced sample sizes and may have weakened some tournament-specific estimates (especially for The Masters, with 272 observations versus 466 for the PGA Championship). Class imbalance (28% top-25% vs 72% rest) also affected the machine learning models, leading Random Forest to achieve high precision but low recall. XGBoost handled the imbalance better but still showed a moderate overfitting gap (8.8%), which is more likely related to noisy performance data and the limited temporal test window. Additionally, evaluating on a single test year (2025) helps avoid leakage but may not fully reflect performance stability.

The analysis focuses only on measurable performance variables and doesn't account for contextual factors such as weather, course setup differences or psychological pressure, which can't be captured through performance statistics. The top-25% threshold is only one possible definition of strong performance and different cutoffs could lead to different patterns. The study examines only the four Major Championships, so results may not fully generalize to other tournaments or Tours. Finally, the models identify statistical relationships rather than causal effects, meaning the coefficients should not be interpreted as the outcome of directly improving a specific skill.

6 Conclusion and Future Work

6.1 Summary

This project successfully identified which performance skills drive success in Major Championships, combining econometric interpretability with machine learning predictive power. The models achieve strong explanatory and predictive performance.

The results consistently identify three main skills: SG Approach, SG Putting, and SG Around-the-Green. These skills appear as the top predictors across logistic regressions, Random Forest, XGBoost and SHAP analysis. Taken together, the convergence across these approaches provides strong evidence that these skills are universally important across all Major Championships. While tournament-specific factors exist (notably, putting becomes more important than approach play at The Open Championship), these represent minor variations in emphasis rather than fundamentally different skill profiles.

This suggests that performing well in Major Championships requires being strong across several areas, not just exceptional in one. Great driving can't compensate for weak putting, and excellent short-game work can't fully offset poor approach play. The convergence of findings across econometric and machine learning methods reinforces that balanced skill development, particularly in approach play, putting and short game, is essential to succeed at Major Championships.

6.2 Future Directions

Future work could extend this analysis in several directions. On the methodological side, adding more tournament years, especially for The Masters and The Open Championship, would strengthen per-Major estimates. Evaluating the models across more seasons would improve stability and give a clearer view of how well the results generalize. Handling class imbalance more explicitly (for example through re-weighting or resampling) could improve recall, particularly for Random Forest.

The scope of the study could also be broadened. Extending the dataset beyond the four Majors to all PGA Tour events or other professional tours would allow testing whether the same skill patterns hold in different competitive contexts. This would reveal whether Majors reward a distinct performance profile or if the findings generalize across tournament types. Further work could also explore alternative targets such as top-10%, winning or making the cut and examine whether different skills become more important at these performance levels.

Furthermore, incorporating contextual and player-level information (such as weather conditions, course setup, field strength, player experience or recent form) would capture how situational factors influence performance and improve both interpretability and predictive accuracy.

Beyond the methodological contributions, this work also has practical applications. For players and coaches, the results suggest where training time may be most beneficial: prioritizing approach play, putting and short-game, which are most strongly associated with top performance. For broadcasters, tracking these metrics during the early rounds could help identify the players most likely to finish near the top of the leaderboard by the end of the tournament, adding more reliability to live analysis.

References

[1] Mark Broadie. *Assessing Golfer Performance on the PGA TOUR*. https://business.columbia.edu/sites/default/files-efs/pubfiles/4996/assessing_golfer_performance.full.pdf. Columbia Business School working paper. 2012.

[2] Mark Broadie. *Assessing Golfer Performance on the PGA TOUR (Strokes Gained)*. https://columbia.edu/~mnb2/broadie/Assets/strokes_gained_pga_broadie_20110408.pdf. Columbia University working paper. 2011.

[3] DataGolf. *A Predictive Model of Tournament Outcomes on the PGA TOUR*. Accessed: 2025-12-23. 2017. URL: https://datagolfblogs.ca/a-predictive-model-of-tournament-outcomes-on-the-pga-tour/.

[4] DataGolf. *Match Play – Data Exploration Exercise*. Accessed: 2025-12-23. 2016. URL: https://datagolfblogs.ca/tag/logistic-regression/.

[5] DataGolf Analytics. *DataGolf Historical Tournament Statistics*. https://datagolf.com/historical-tournament-stats. Player performance and strokes gained statistics for PGA Tour events and Major Championships. 2025. (Visited on 01/06/2026).

[6] Brandon Klassen. *Using Machine Learning to Predict Professional Golf Performance*. https://studylib.net/doc/26285044/using-machine-learning-to-predict-professional-golf-performance. Bachelor thesis, Brock University. 2019.

[7] Kevin Li. *PGA Tour Exploratory Data Analysis and Machine Learning Predictive Model*. Accessed: 2025-12-23. Northwestern Sports Analytics Group. 2022. URL: https://sites.northwestern.edu/nusportsanalytics/2022/03/01/pga-tour-exploratory-data-analysis-and-machine-learning-prediction/.

[8] Kyle Schomer. *How Is Strokes Gained Calculated?* Golfshot. July 17, 2023. URL: https://golfshot.com/blog/how-is-strokes-gained-calculated.

[9] Oisin Wiseman. *Using Machine Learning to Predict the Winning Score of Professional Golf Events on the PGA*. https://norma.ncirl.ie/2493/1/oisinwiseman.pdf. Master's thesis, National College of Ireland. 2016.

A Additional Figures

A.1 Exploratory Data Analysis

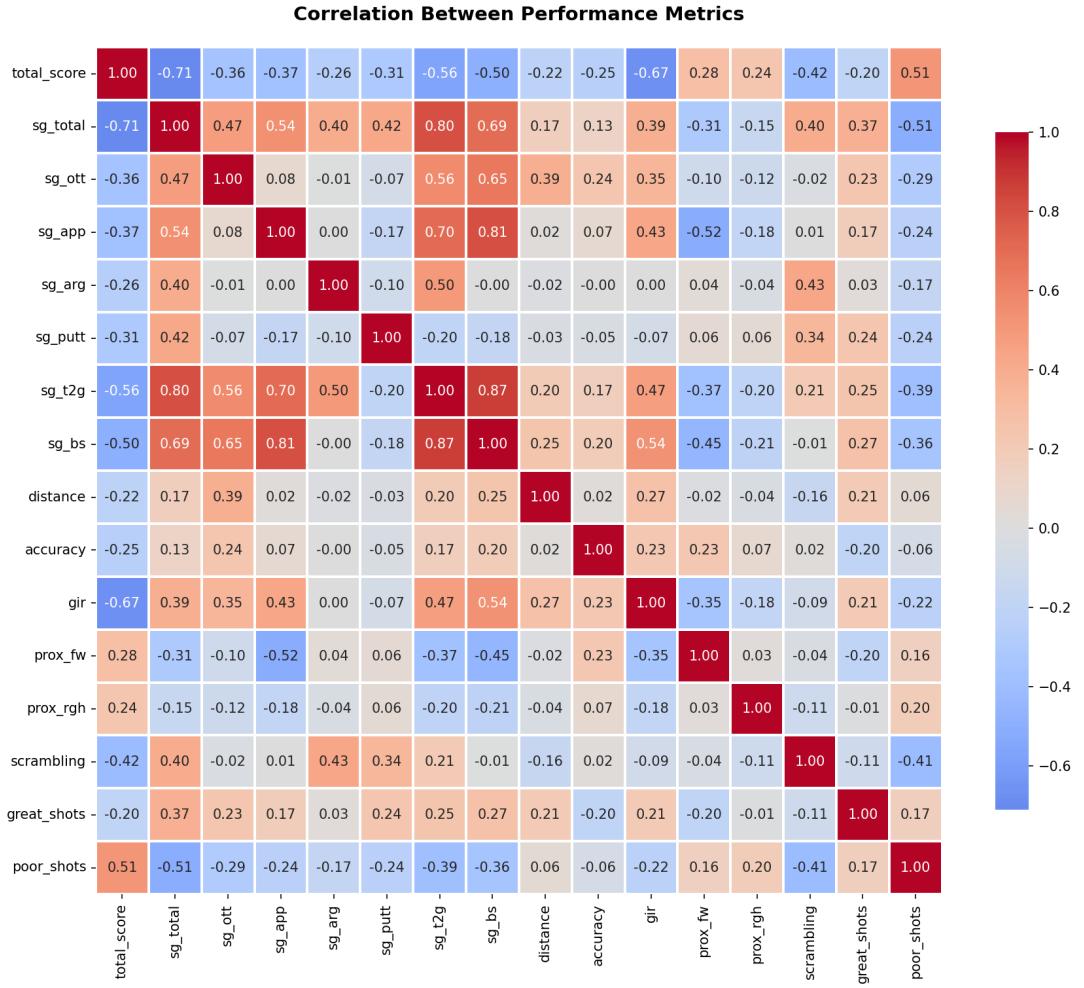

Figure A1: Correlation heatmap between performance metrics and total score

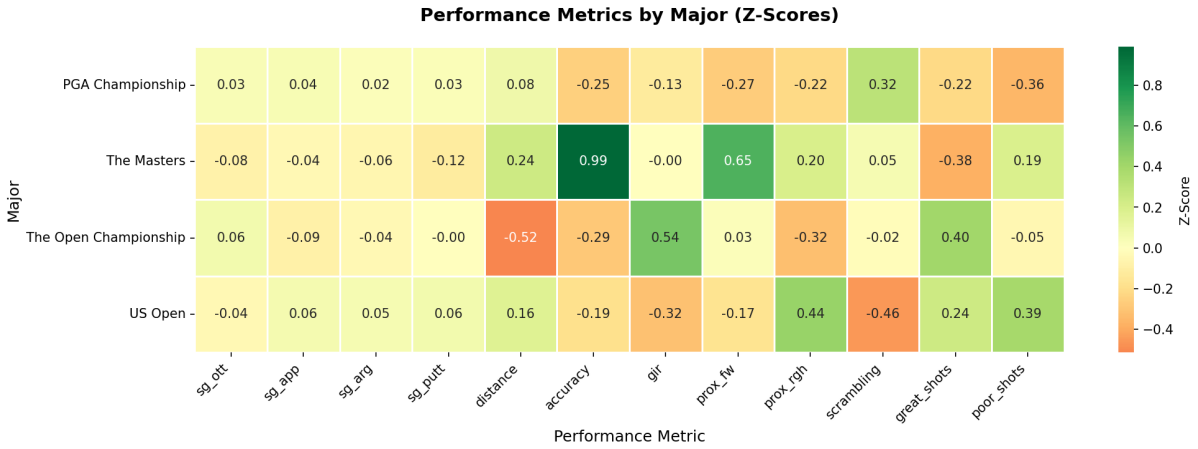

Figure A2: Standardized performance metrics by Major Championship

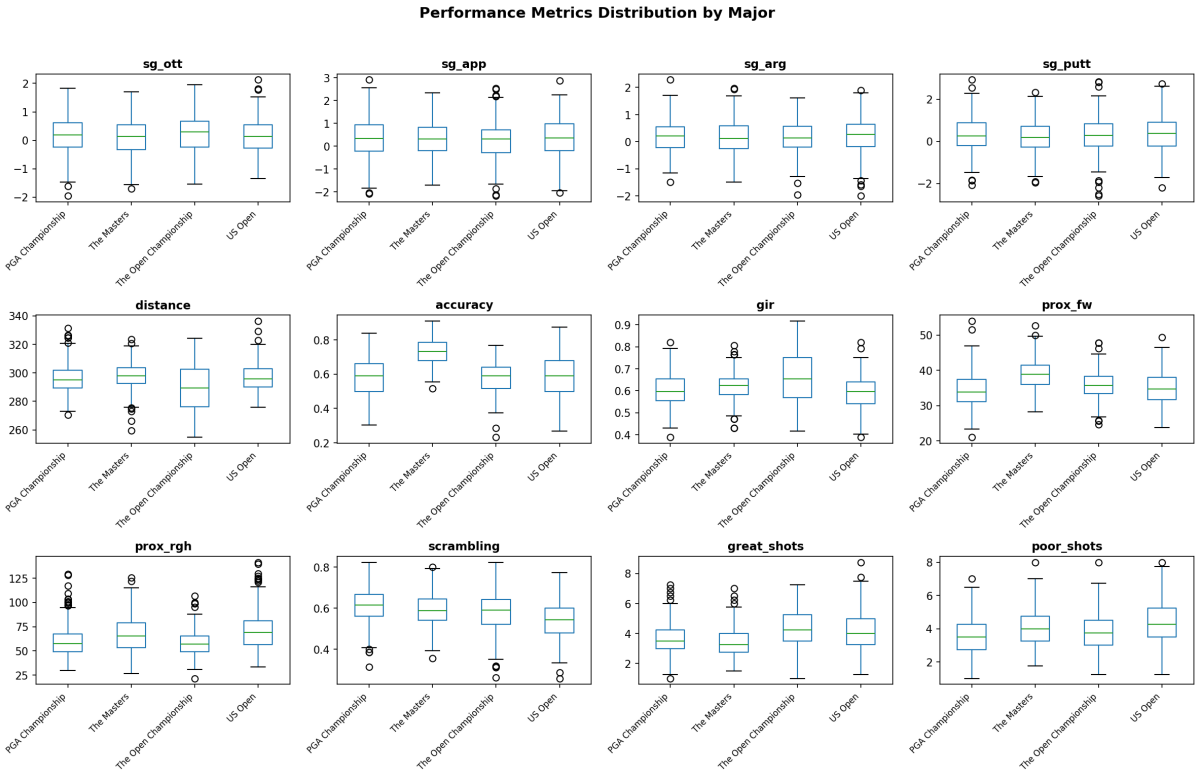

Figure A3: Standardized performance metrics by Major Championship

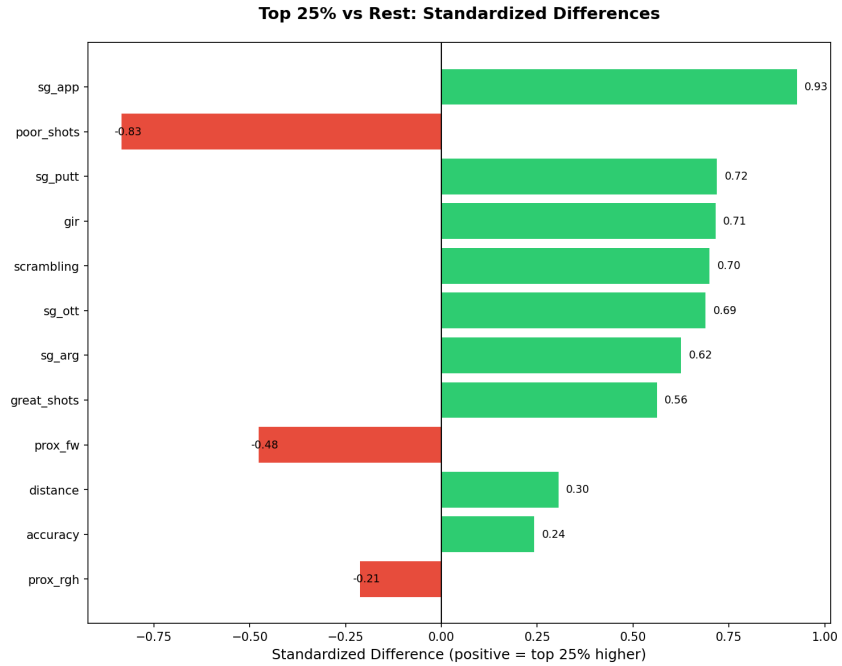

Figure A4: Standardised differences between Top-25% players and the rest of the field

A.2 Econometric Models

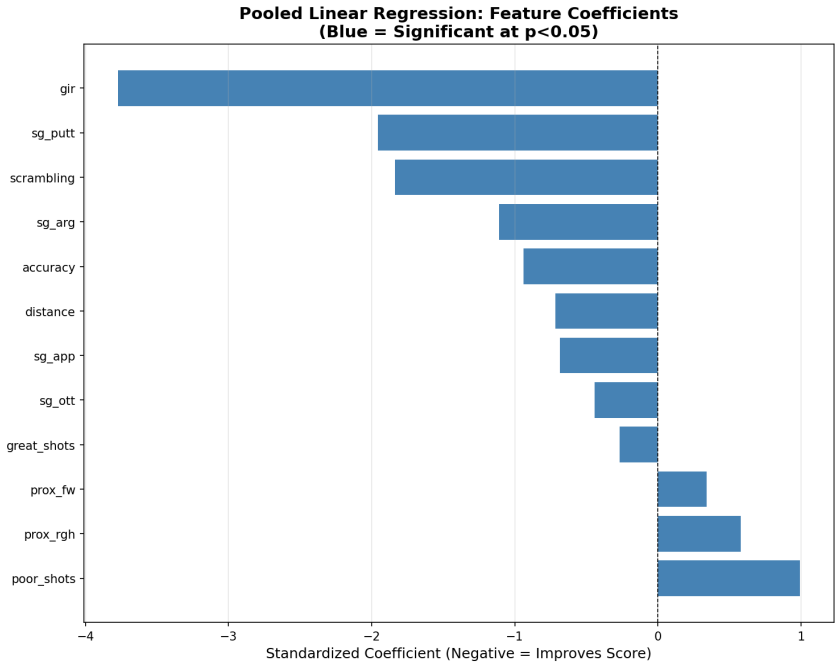

Figure A5: Pooled linear regression coefficients. Values show how each performance metric is associated with total score across all Majors, more negative coefficients indicate better scoring performance.

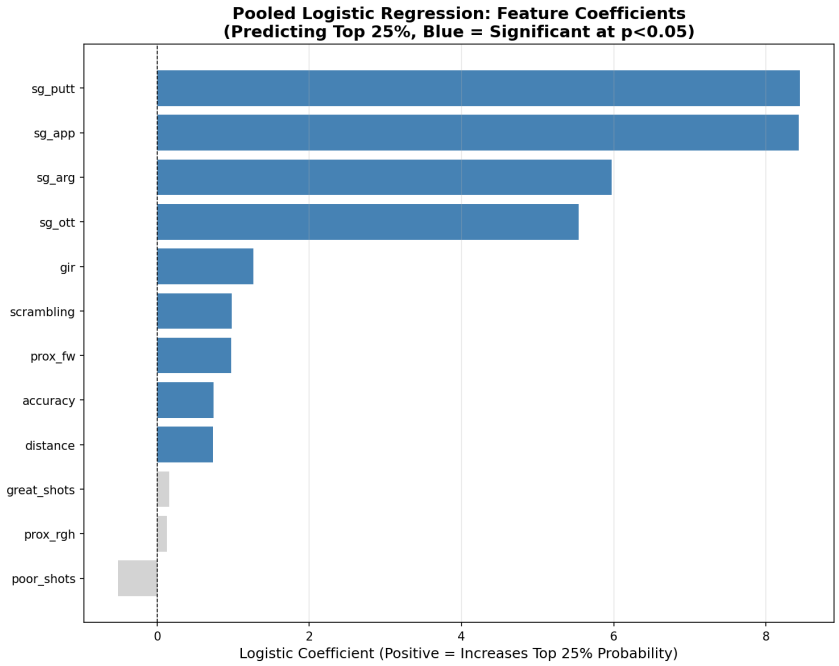

Figure A6: Pooled logistic regression coefficients for predicting Top-25% finishes. Positive coefficients indicate a higher probability of finishing in the Top-25%.

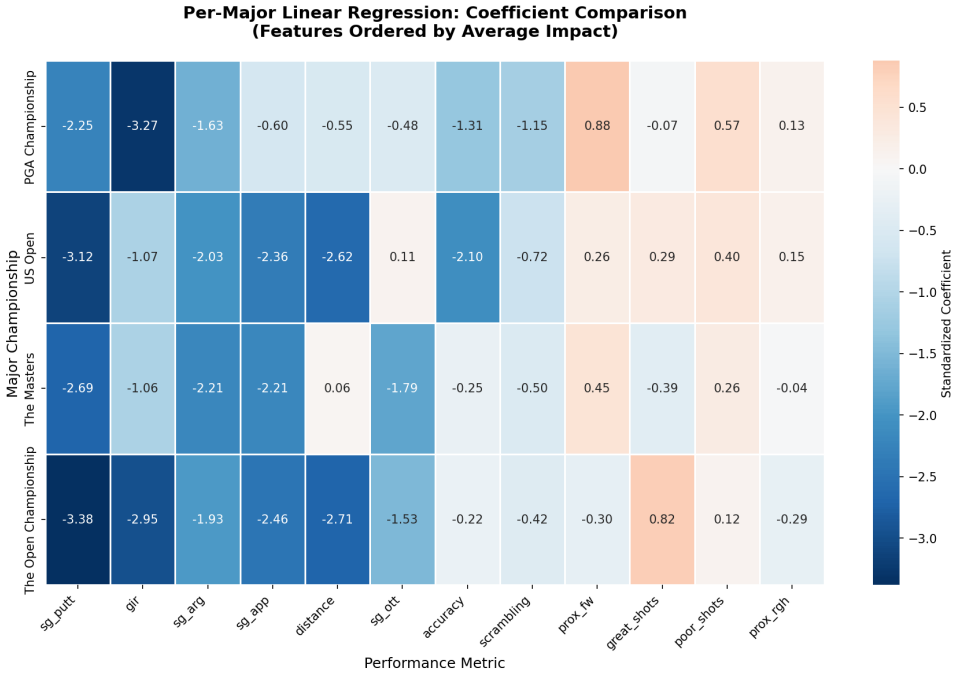

Figure A7: Per-Major linear regression coefficients, using standardized variables. Values show how each performance metric is associated with total score, where more negative coefficients indicate better scoring performance.

A.3 Machine Learning Models

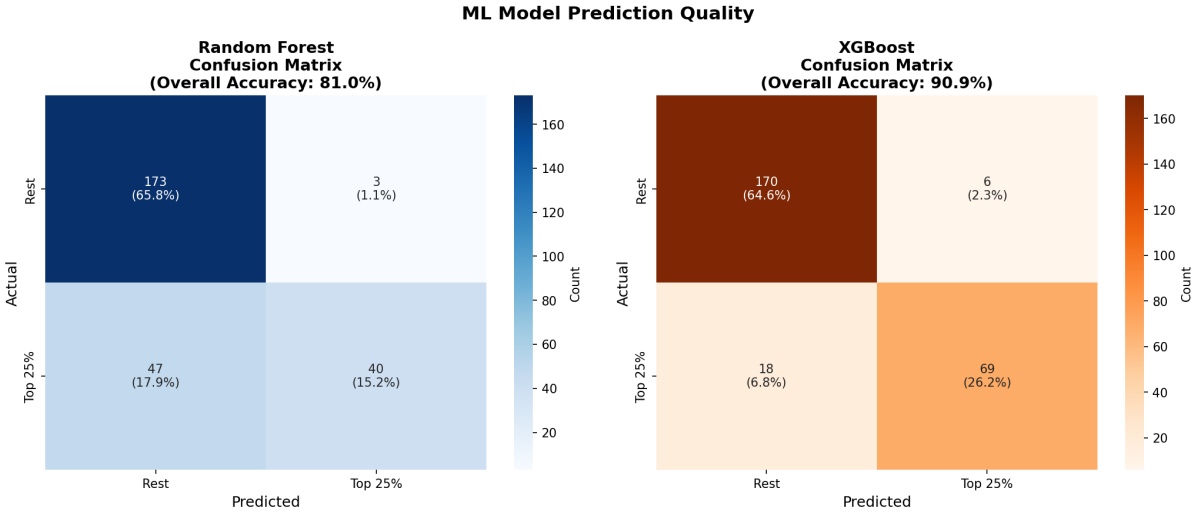

Figure A8: Confusion matrices for Random Forest and XGBoost on the 2025 test set, comparing predicted and actual Top-25% finishes.

B Code Repository

GitHub Repository: `https://github.com/pinaluciana/Golf_project`

Project structure

```
Golf_project/
   main.py                    # Main entry point
   environment.yml            # Conda dependencies
   requirements.txt           # Pip dependencies
   proposal.md                # Project proposal
   README.md                  # Setup and usage details
   AI_use.md                  # External and AI tools usage
   Golf_report_LSP.pdf        # This file
 src/                         # Source code
      data_loader.py          # Data loading
      exploratory.py          # Exploratory analysis
      feature_engineering.py # Feature preparation
      visualization.py        # Plotting functions
    data_preparation/         # Data preprocessing scripts
    models/                   # Econometric and ML models
       Econometric_models.py
       ML_models.py
       evaluation.py
 data/                        # Data directory
    processed/                # Combined datasets per major and all_majors_combined.csv
    raw/                      # Raw csvs per major and year
 results/                     # Outputs in csvs and figures
     1_Exploratory/
     2_Econometric_models/
     3_ML_models/
```

Installation instructions

The project was developed and tested on Python 3.11 using Conda. To set up the environment follow the next commands:

```
git clone https://github.com/pinaluciana/Golf_project
cd Golf_project
conda env create -f environment.yml
conda activate Golf_project
```

Alternatively, the dependencies can be installed using `pip install -r requirements.txt`. The main packages used include pandas, numpy, scikit-learn, xgboost, statsmodels, shap, matplotlib, and seaborn, with exact versions specified in `requirements.txt`.

How to Reproduce Results

Step 1: Data preprocessing Raw CSV files should be placed in their respective Major folders under `data/raw/`. The preprocessing scripts located in `src/data_preparation/` must be run to clean the raw data and combine tournament years for each Major, saving the processed datasets to `data/processed/`. Finally, `combine_all_majors.py` must be run to merge these files into `all_majors_combined.csv`, which is loaded by `data_loader.py` and used by `main.py`.

Step 2: Full analysis After preprocessing is complete, return to the project root and run: python `main.py`. This executes the full workflow: data loader, exploratory analysis, feature engineering, econometric models, machine learning models, evaluation and visualization.

Outputs and reproducibility All results are saved in the `results/` directory, with separate subfolders for exploratory analysis, econometric models and machine learning models. The random state is fixed at 42 and a temporal split is used (training: 2020–2024, testing: 2025), ensuring full reproducibility given the same environment and data.

Code maintenance and version control

The project uses Git for version control, hosted on GitHub. The code has a modular structure that allows independent updates without affecting other components. The code is structured in `src/` with separate files for each analysis, for instance updating a model requires changes only to the corresponding file in `src/models`. The `main.py` file remains stable across updates.

Formal unit tests were not implemented, as the project's focus was data analysis and modeling rather than software engineering. However, the pipeline includes validation steps for instance, `data_loader.py` verifies file paths exist before loading, `exploratory.py` checks for missing values and `evaluation.py` validates that model outputs match expected formats.

C Use of external tools

In order to develop this project's code and write the report, some external tools were used. However, these tools were merely used for support and editing and were not used to generate content.

Code development

Tools used: YouTube tutorials, library documentation, Claude and ChatGPT.

Used for: debugging in general, understanding error messages, explaining Python library documentation (for statsmodels, SHAP and xgboost), resolving library compatibility issues, understanding pandas operations, finding out the best practices for scikit-learn and GridSearchCV, formatting of the visualizations in matplotlib and seaborn, and understanding SHAP value computation and TreeExplainer output classification.

All logic, data preprocessing steps, feature engineering, model design, evaluation and code structure were developed independently.

Report writing

Tools used: Grammarly and ChatGPT.

Used for: rephrasing to improve clarity and readability, checking grammar and consistency, and formatting in LaTeX.

The research question, methodology, analysis, result interpretation and conclusions are original work. The external tools were only used to polish the report, not to generate any content or interpretations.