

# Predicting Round and Game Winners in CSGO

Allen Rubin

[allen2rubin@gmail.com](mailto:allen2rubin@gmail.com)

## Abstract

Win probabilities have become a staple on scoreboards in physical sports such as baseball and basketball. Esports, or competitive video games with sponsored teams and major audiences, typically lack this detailed statistical analysis, beyond bare-bones metrics and commentator intuition. However, the advantage of esports in their tendency to have a central record of every game event makes them ripe for statistical analysis through machine learning. Previous research has covered popular video game genres such as MOBAs, and has found success in predicting game winners most of the time. Counterstrike: Global Offensive (CSGO) is an esports that is unique in its round and game-based nature, allowing researchers to examine how short and long-term decisions can interplay in competitive environments. We introduce a dataset of CSGO games To assess factors such as player purchasing decisions and individual scores, we introduce 3 round and game win probability models. Finally, we evaluate the performances of the models. We successfully predict winners in the majority of cases, better than the map average baseline win statistics.

## 1. Introduction

Enjoying massive growth over the past few years, the global esports industry has a predicted 2021 revenue of over 1 billion dollars and a live-streaming audience of almost 730 million [1]. While traditional athletic competitions such as football and basketball have rapidly expanded data collection of team and player performance for improved matchup analysis, esports still lag behind. In physical sports, the boundaries of data analysis have been expanded through increased measurements of participant actions, with tools such as player-worn monitors and video replays. On the front of data availability, esports are uniquely advantaged in that all game information, including player decisions, are recorded on a central server.

Many multiplayer video games, such as Counter-Strike: Global Offensive (CSGO), log all game events to a demo file. However, without proper frameworks offered by game developers to capture and examine these events, the greatest limiting factor in esports analytics is “the lack of easily accessible and clean esports data” [2]. As CSGO demo files are more accessible and easier to parse than demo files from other professional video games, CSGO represents an obvious stepping stone for growing the field of esports analytics. However, even with this data, it can be difficult for human analysts to quickly predict the outcome of a match. They have to consider many different variables, including the score, equipment, and balance of every player in a match. Through the development of models to predict winners based on these gameplay intricacies, we can inform sponsors of the best teams to support, increase audience engagement in the flourishing fundraising practice of charity streams, more accurately determine match odds in real-time for the growing industry of esports betting, and also isolate areas of improvement for individual team strategies. In this paper, we will present our methods for collecting demo files, parsing them for match data, and creating different models. Our models are trained using features such as player scores, equipment values, balances, and loadouts to predict winners on round and game levels.

Section 2 will describe the ruleset of CSGO, explaining how it is played. Section 3 will briefly discuss related works and contain some explanations for the decisions we made with this research. Section 4 will cover data collection and parsing. Section 5 will cover our different models and how they were adjusted. Section 6 will be about our experimental results. Finally, in section 7, we will address our limitations, avenues for future work, and conclude the paper.

## **2. Counter-Strike: Global Offensive**

### **2.1 Background**

Counter-Strike: Global Offensive (CSGO), released in 2012 and currently developed by Valve Corporation, is the latest in a series of video games pitting virtual terrorists against counter terrorists since 1999. A classic CSGO professional match comprises two teams playing to win 16 rounds first. If they both reach 15, an overtime process takes place. This eliminates ties as a game-end condition and ensures a winner for every match.

### **2.2 Match Overview**

At game start, teams are chosen to be the Terrorists (T) or the Counter-Terrorists (CT). They then play a game “half,” or 15 rounds, with this setup, and then switch sides. Each round has a time limit of 1 minute 55 seconds. Teams hold different avenues to victory based on their current side.

### **2.3 Win Conditions**

Every round, all players begin with 100 health points (HP) and are eliminated for the rest of the round if their HP drops to 0. Gunfire and grenade damage are the primary reasons for decreases in HP, yet players can also suffer from environmental damage, such as from falls.

Beginning with the T side, their most straightforward round win condition is completely eliminating the CT team. Failing to do this, Ts can still win by planting a bomb at one of two possible sites, and defending it until a 40-second timer counts down and the bomb explodes. Every round, a random T player spawns holding the bomb in their inventory, and they drop it unarmed on death.

The CT side can also directly achieve round victory through completely eliminating the T team, as long as the bomb has not yet been planted. If the bomb has been planted, CTs can win by defusing it, a process that takes 10 or 5 seconds, depending on whether the CT player has bought the specialized “defuse kit” tool. This process can be interrupted if the defusing player is eliminated in its duration. The final way to win as CTs is to stay alive until the round timer expires, without the bomb being planted.

## 2.4 In-Game Economy

All players have personal balances, which they use to purchase weapons, like guns and grenades, and equipment, such as armor and helmets, at the beginning of every round. Buy menu items have specific prices, yet teams have access to slightly different buy menus depending on if they are a T or CT. Ts have access to slightly cheaper and more damaging guns, while CTs have more accurate yet more expensive guns. This is meant to compensate for the common and expected gameplay scenario of Ts attacking a site held by CTs, who have a defender's advantage.

Every beneficial act a player commits, such as a successful elimination of an opponent, grants that player an increase to their balance. Team acts, such as bomb plants, defuses, and round victories, give a reward to every member on the team. When a team loses several rounds in a row, they get an increasing bonus with every loss to help improve the team economy and their chances of catching back up to the opposition. These loss bonuses are still typically less than the victory rewards for the winning team.

There are many options available for loadout construction for every player, and balancing weapon viability and cost is important with limited funds. Thus, team economies represent a major area of focus in optimizing decision making for teams and players alike. With the added capability of dropping weapons for other teammates, there is an incredibly large set of possibilities for economic decisions. Hence, we limit our analysis of economies and loadouts in this paper to one snapshot for every round, leaving closer scrutiny for future work.

## 2.5 Game Summary

In every round of the game, two teams independently decide how to spend their in-game money on in-game loadouts. Human players make these decisions based on the current game situation (represented by team scores), their own equipment values (the net worth of all the equipment they currently hold), and their balance (how much money each individual player holds in their personal bank).

Working together, team members may choose from a variety of strategies. "Full saving", or avoiding making new purchases, means likely throwing the round but ensuring adequate funds to buy equipment the next. "Full buying", or purchasing the best equipment, is likely to lead to round victory. The end goal is always winning the full game, so sacrificing a round in order to start building up momentum with better equipment in the next rounds is a legitimate strategy. We take these economic decisions in tandem with metrics of player performance measured by their scores in order to build a round and game winner prediction model.

## 3. Related Work

MOBAs, or multiplayer online battle arenas, such as DOTA2 have been a popular esports for data analysts to conduct research with, with a strong focus on predicting the winner [3]. This interest is present in both industry and academia, and our paper seeks to further expand its scope in CSGO, a different type of esports (first person shooter).

Previous research found that the effects on match outcome from map variance was less important than the quantity of data provided to a winner prediction model [4]. This is why, instead of building separate models for every map, we decided to ignore the map as a feature.

Previous work tends to focus on predicting winners in CSGO at the round level. Our paper introduces game winners as a feature in the dataset in order to also predict winners at the game level. This is because throughout the game, teams may decide to intentionally tank their chances of winning a certain round in order to increase their loss bonus and save money for better equipment in the next round. In order to take this possible long-term strategy into account, we evaluate game winners alongside round winners.

## **4. Dataset**

### **4.1 Source**

As esports are played through a server and thus expected to record a ground truth for events in every game, one could be led to believe that capturing data should be trivial. Unfortunately, tools for acquiring data are surprisingly limited. In absence of their availability offered by the game developer, we turn to community tools.

First, we chose 45 recent “maps” played at the professional level from September 1st, 2021 to December 5th, 2021. Each professional match usually consists of a best-of-three-map system. Each map counts as a game, and corresponds to one demo file. Each game averages slightly less than thirty rounds, so our total data set comprised 1229 rounds.

The demo files came from HLTV.org, one of the most popular CSGO fan sites and the premier repository of professional game demo files. We used the Demo Downloader for HLTV offered by c9win.com [5] to expedite the demo file download process. Professional games were chosen for their consistency and to mitigate edge cases such as surrenders, player disconnects, abandons, or self-sabotage.

### **4.2 Parsing**

We used the open-source Node.js library offered by Saul Rennison [6] in order to systematically process our downloaded demo files. We logged the values of interest each round at the end of the buy period (the period by which all equipment should have been bought), as well as the round results. At the end of every demo file, we appended the team that won the game. After collection, we cleaned our data to ensure quality.

### **4.3 Data**

The following is a general structure of one sample in our dataset, each of which had 152 features and 2 outcomes.

15 attributes repeated across 10 players:

|           |         |            |             |        |
|-----------|---------|------------|-------------|--------|
| Team Type | Balance | Current EV | Round Spent | Score  |
| Armor?    | Helmet? | Defuser?   | Weapon      | Weapon |
| Weapon    | Weapon  | Weapon     | Weapon      | Weapon |

The current scores:

|             |
|-------------|
| team1 score |
| team2 score |

And finally, the 2 outcomes:

|              |
|--------------|
| Round Winner |
| Match Winner |

Most variable names are self-explanatory, but we'll cover them here in detail for clarity.

- Team Type - the side an individual player is on, meaning terrorist (T) or counterterrorist (CT). Different sides are exposed to different circumstances and objectives, as described in sections 2.3 and 2.4.
- Balance - funds in player account.
- Current Equipment Value - net worth of all items held by the player.
- RoundSpent - how much money the player has spent this round.
- Score - how many points the player has amassed so far into the game. Represents kills, assists, and successful objective plays.
- Armor? - presence of equipment that reduces incoming damage.
- Helmet? - presence of equipment that reduces incoming headshot damage.
- Defuser? - exclusive to CT side; reduces bomb defuse time.
- Weapon - all possible slots for different weapon combinations.
- Team1 Score - how many rounds team 1 has won so far.
- Team2 Score - how many rounds team 2 has won so far.
- Round winner - which team won the round.
- Match winner - which team won the match.

## 5. Model

### 5.1 Classification

Predicting a winner out of two teams based on a certain game state presents a classic binary classification problem. In this paper, we use and compare results from tree based classifiers and a neural network. We utilized the scikit-learn implementations of the random forest and multi-layer perceptron

classifiers, along with the `xgboost` package for our XGBoost implementation. We used a 70/30 train/test split.

## 5.2 Hyperparameter Tuning

Unless stated, we left the parameters set to their defaults.

### 1) Random Forest Classifier Tuning:

We set the number of estimators to 1000. We set max features to  $\log_2$ . We set max depth to 3.

### 2) XGBoost Classifier Tuning: We set the objective function to `binary:logistic`.

### 3) MLP Classifier Tuning: We set 10 hidden layers of 100 nodes each, and set max iterations to 10,000.

## 5.3 Evaluation Metrics

We evaluate the methods by calculating the AUROC curve, the accuracy, and the F1 score between model output and ground truth.

## 6. Experiments

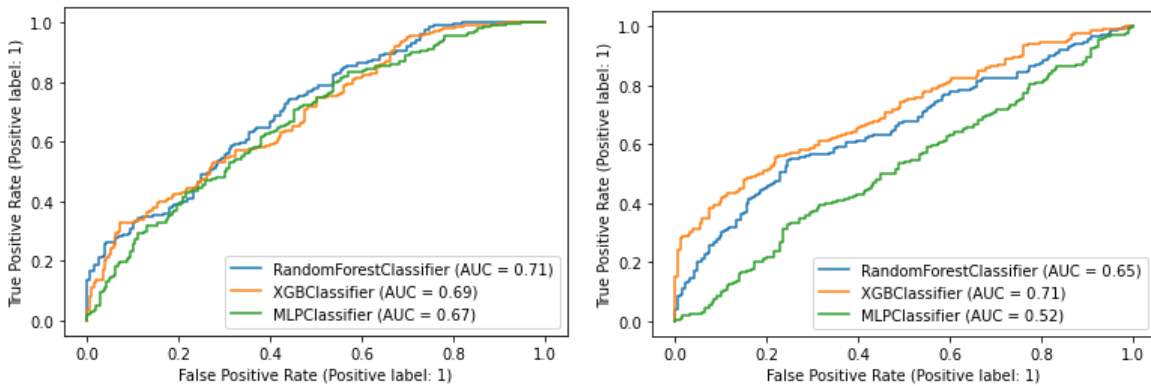


Figure 1: AUROC comparisons for RF, XGB, and MLP evaluating prediction of round and game winners, respectively.

We begin by assessing the performances of our win probability models. We can see that by using a random forest model for the round-level and an XGBoost model for the game-level, we can predict winners in CSGO with AUC values of up to .71, better than the map average benchmark of .5303 found by previous researchers [2].

| Round Winner Prediction |               |               |
|-------------------------|---------------|---------------|
| Classifier Method       | Accuracy      | F1            |
| Random Forest           | <b>0.6407</b> | <b>0.6711</b> |
| XGBoost                 | 0.5970        | 0.6210        |
| Multi-layer Perceptron  | 0.6140        | 0.6223        |

Match Winner Prediction

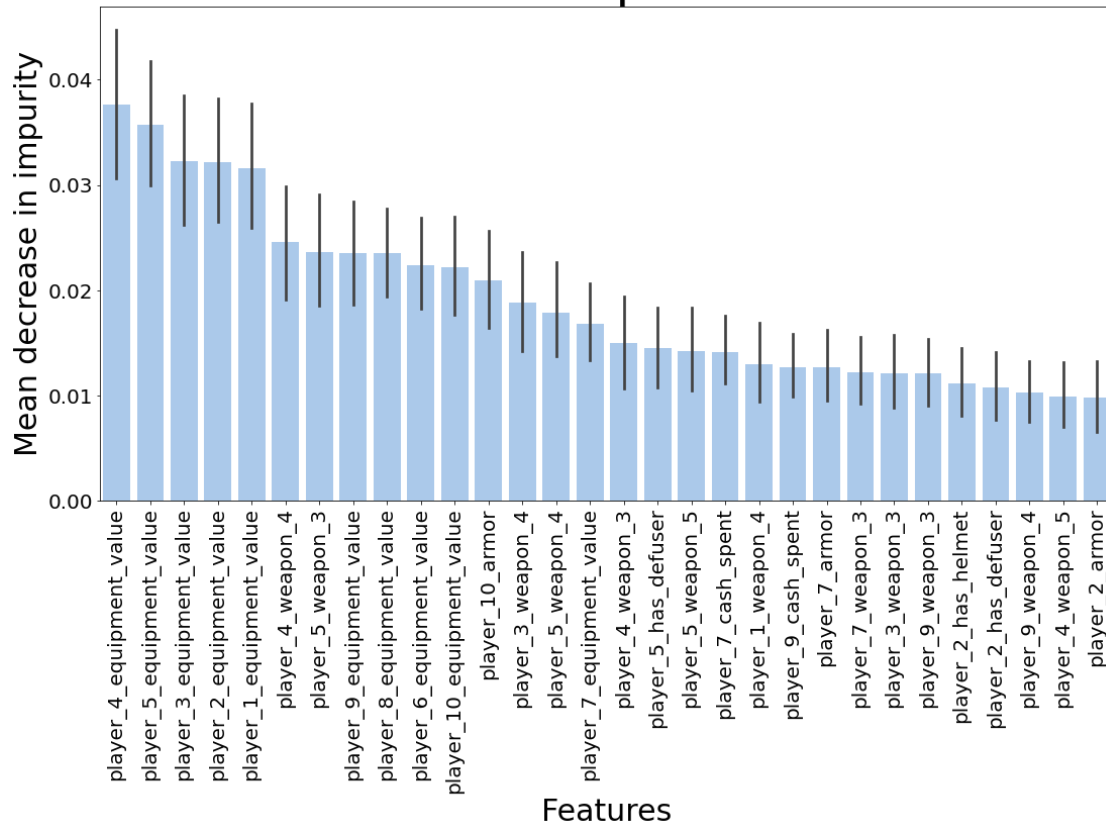
| Classifier Method      | Accuracy      | F1            |
|------------------------|---------------|---------------|
| Random Forest          | 0.5898        | 0.6841        |
| XGBoost                | <b>0.6237</b> | <b>0.6930</b> |
| Multi-layer Perceptron | 0.5242        | 0.5864        |

Table 1: **The results of different methods.** The Random Forest model worked better in predicting round winners, while the XGBoost model performed better in predicting game winners.

We are able to predict round and match winners correctly the majority of the time. For the round-level, our Random Forest model performs better across every metric, while for the game-level, our XGBoost model performs better across every metric.

### Round-Level

## Feature importances



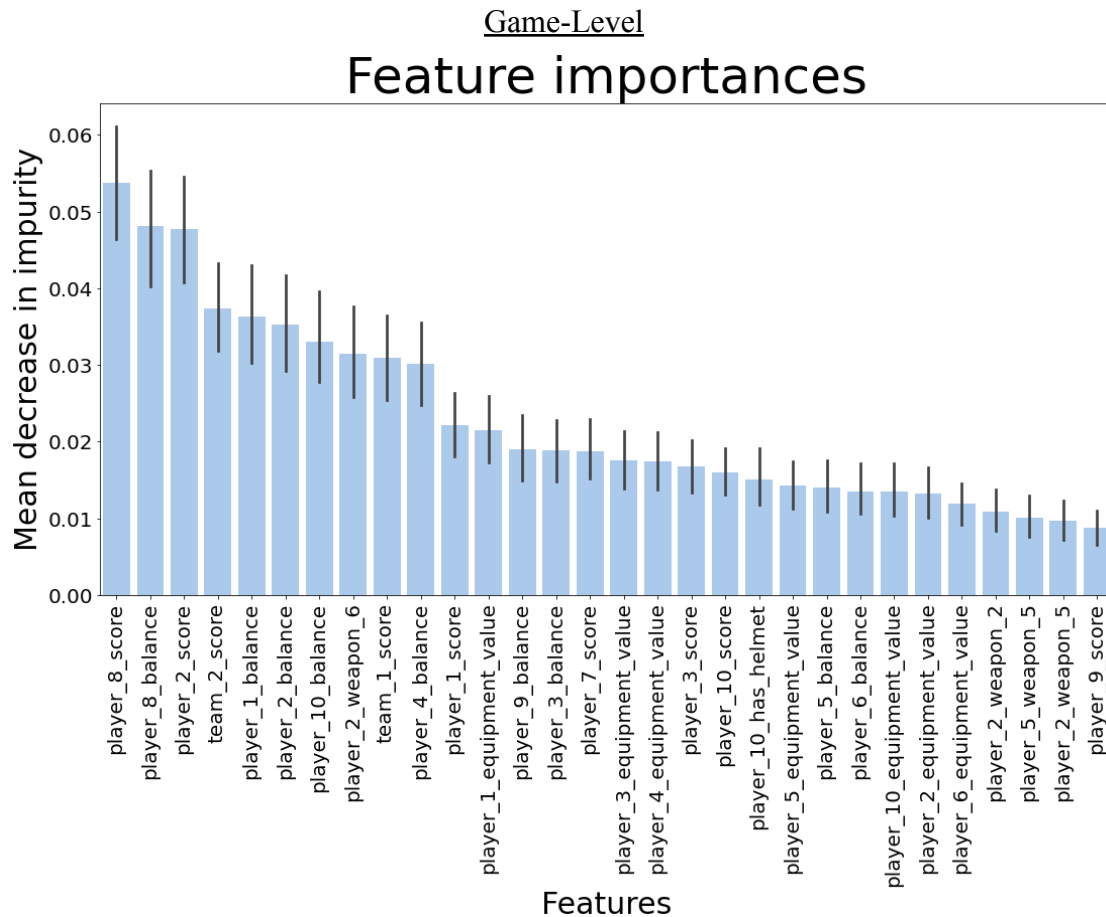


Figure 2: **Feature Importances.** Features were ranked by how much they decreased Gini impurity between parent and daughter nodes in the Random Forest Model.

One of the Random Forest algorithm's benefits is that it can identify feature importances through analyzing the total impurity decrease provided by each feature. By knowing which features are driving the model, teams can be more informed considering relevant aspects of their gameplay.

In Figure 2, we show the top feature importances in our round-level win probability model. From Figure 2, it is apparent that a team's equipment value is the lead determinant of a team's chances of winning a round. Intuitively, holding better equipment is a significant advantage for a team. Better equipment is generally more accurate, more damaging, and faster-firing, making it easier for a team to achieve every win condition. We also see that the specific weapons held are a strong factor for predicting which side wins a round. With game updates and balance changes to weapon stats, the impact of weapons chosen on match outcomes may also change, necessitating continuous training and additions to the dataset.

As for the game-level, we found that player scores and balances are much better indicators for their long-term win probability. Score, as a direct metric of player performance, and the balance, a direct reference to the economic trends of a player, must be directly considered when performing win probability analysis among 2 teams in a CSGO match.



## 7. Conclusion

For this paper, we constructed a round-based data set of 45 CSGO games. Compared to datasets from other relevant works[2][4][7], ours is quite small. However, the data was selected from high-quality professional games and manually cleaned to ensure consistency that is representative of most CSGO games. By automating data collection, parsing, and data cleaning, our dataset can be grown for future use.

With our model comparison, we found the Random Forest Classifier to be the most effective for predicting round winners, and the XGBoost classifier the most effective for predicting match winners. Using the tool for feature importance, we reinforced the intuitive belief that holding expensive and thus effective weapons is the single greatest predictor of team round performance, especially at the professional level. We also reinforced the expectation of economic and player performance trends being strong predictors of winners at the game-level. Our model can be expanded to include other features of interest to teams, such as positional values, in order to formulate effective attack and defense strategies.

In order to facilitate further research and analytics in their games, developers should offer their own tools for data collection. By quantitatively exploring our competitive forms of entertainment, we can find advantages for participants, increase audience engagement for sponsors, and inform bettors in real time.

## References

- [1] Reitkerk, R. (2021, December 3). Newzoo's Global Esports & Live Streaming Market Report 2021: Free version. Newzoo. Retrieved December 25, 2021, from <https://newzoo.com/insights/trend-reports/newzoos-global-esports-live-streaming-market-report-2021-free-version/>
- [2] Xenopoulos, P., Doraiswamy, H., & Silva, C. (2020). Valuing player actions in Counter-Strike: Global Offensive. 2020 IEEE International Conference on Big Data (Big Data). <https://doi.org/10.1109/bigdata50022.2020.9378154>
- [3] Hodge, V. J., Devlin, S., Sephton, N., Block, F., Cowling, P. I., & Drachen, A. (2019). Win prediction in multiplayer esports: Live professional match prediction. *IEEE Transactions on Games*, 13(4), 368–379. <https://doi.org/10.1109/tg.2019.2948469>
- [4] Xenopoulos, P., Coelho, B., & Silva, C. (2021). Optimal Team Economic Decisions in Counter-Strike. *arXiv [cs.GT]*. Opgehaal van <http://arxiv.org/abs/2109.12990>
- [5] Balderston, D. (2021, July 29). Demo downloader for HLTV. C9WIN. Retrieved December 8, 2021, from <https://c9win.com/demo-downloader/>
- [6] Rennison, S. (2021, December 1). Saul/demofile: Node.js library for Parsing Counter-Strike: Global Offensive Demo Files. GitHub. Retrieved December 7, 2021, from <https://github.com/saul/demofile>
- [7] Zeng, Y., Lei, D., Li, B., Jiang, G., Ferrara, E., & Zyda, M. (2020). Learning to Reason in Round-based Games: Multi-task Sequence Generation for Purchasing Decision Making in First-person Shooters. *arXiv [cs.AI]*. Opgehaal van <http://arxiv.org/abs/2008.05131>

