Predicting Counter-Strike Win Rates Using Player Statistics*

Shuyang Qiu

April 19, 2024

Counter-Strike player statistics were used to create a model to predict their team's win rate. Predictors used include player performance metrics such as average damage per round and overall statistics such as total number of rounds played. We found some average damage per round is negatively correlated with win rates while other metrics were positively correlated. This model could be improved on to allow bookmakers to create more efficient markets, or allow coaches to make better roster decisions.

1 Introduction

Modelling and predicting results of competitive sports is a potentially lucrative endeavor and can help provide insights for competitors, team managers, coaches, and others. Sports betting is a multi-billion dollar industry employing hundreds of data analysts to create better models and increase market efficiency. On the other hand, sharp bettors are creating their own models to beat the odds and make fortunes. Meanwhile, teams and coaches can use models to create better rosters and find the optimal training for their players.

Sports and other physical activities such as horse racing have been the primary focus of bookmakers but esports betting is becoming more popular and esports markets are being offered by many major bookmakers such as Pinnacle and Bet365. Since esports are new compared to traditional sports, there is relatively little research and the models are of lower quality.

CS (Counter-Strike) is a 5v5 competitive multiplayer game where one team defends a bombsite while the other attempts to plant and detonate a bomb. Professional *CS* teams compete in tournaments where teams play each other, where the number of maps won determine the winner of the matchup. In this paper, we created a model predict the estimand, map win rate,

*Code and data are available at: https://github.com/shuyangqiu/cs_stats.

using various individual statistics such as K/D (kill/death ratio), ADR (average damage per round), and the total number of rounds played.

We found that some metrics such as K/D have a positive correlation with win rates while other metrics such ADR have negative correlations with win rate.

Section 2 explains the source of the data, how it was collected, how it was cleaned and analyzed, and visualizations of the data. Section 3 presents the model used to fit the dataset, the predictors used, and a justification for the model. Section 4 explains the results of the analysis with visualizations. Section 5 provides discussion on the results of this paper as well as weaknesses and next steps.

2 Data

The data used in this paper was provided by HLTV.org (Better Collective 2024), a website that provides *CS* news, information about tournaments, past matches, and player statistics. Existing datasets for this information are rare and their measurement criteria are not clear, so the dataset used for this paper was constructed. Player statistics and match data were scraped from HLTV.org using Python (Python Software Foundation 2020) and the `selenium` (Software Freedom Conservancy 2024) and `BeautifulSoup` (L. Richardson 2024) libraries.

The win/loss data used is from all matches played between October 16, 2023 and April 17, 2024 in *CS:2 (Counter-Strike: 2)* tournaments with prize pools of at least $100,000. The player data used are lifetime statistics and include matches played in both *CS:GO (Counter-Strike: Global Offensive)* and *CS:2*. The lifetime statistics also include data from matches not used in the win/loss data.

The raw data was then cleaned and analyzed using statistical programming language R (R Core Team 2022) and the `tidyverse` (Wickham et al. 2019), `janitor` (Firke 2023), `arrow` (N. Richardson et al. 2024), `stanarm` (Brilleman et al. 2018), `knitr` (Xie 2014), `modelsummary` (Arel-Bundock 2022), `marginaleffects` (Arel-Bundock 2024), and `here` (Müller 2020) libraries.

The final dataset consists of 445 players with a cumulative 9855 maps played across 22 tournaments. For each player, the dataset contains the number of tournament maps won and lost. A map is won if their team wins 13 of the first 24 rounds or if they win 4 out of 6 rounds in overtime. The dataset also contains the number of kills, deaths, HS% (headshot percentage), ADR, and the total number of rounds played. These statistics are measured over the player's professional career. HS% is measured as the proportion of kills where the killing blow is a headshot and ADR is measured as the total damage done to enemies divided by the number of rounds played.

Figure 1 shows the distribution of the number of tournament maps played while Figure 2 and Figure 3 show the distributions of the number of wins and losses respectively. The median number of maps played is 12 and teams typically play a minimum of 4 maps per tournament (2 best of 1s and a best of 3 elimination match), indicating most players have short tournament runs or are not invited. This can also be seen in Figure 3 where the most common number of maps won are 0 to 1, indicating those players have been knocked out early in every tournament they played.

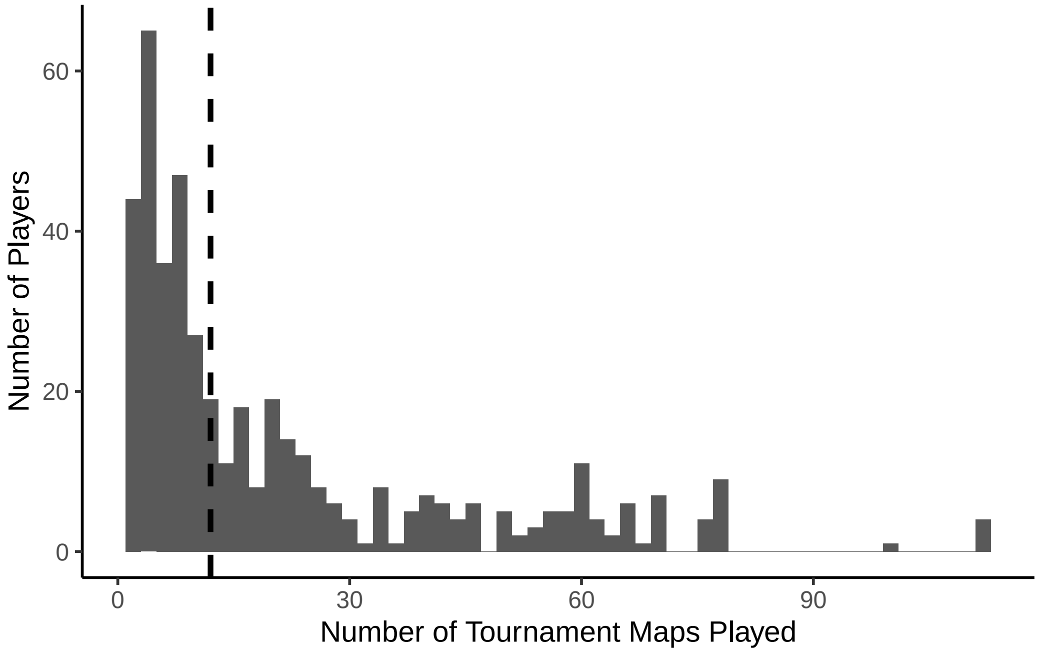

Figure 1: Number of tournament maps played with median

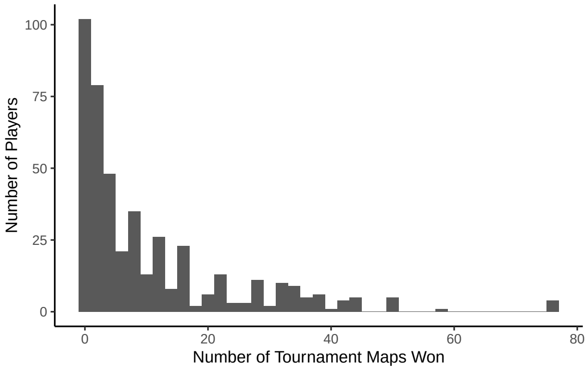

Figure 2: Number of tournament maps won

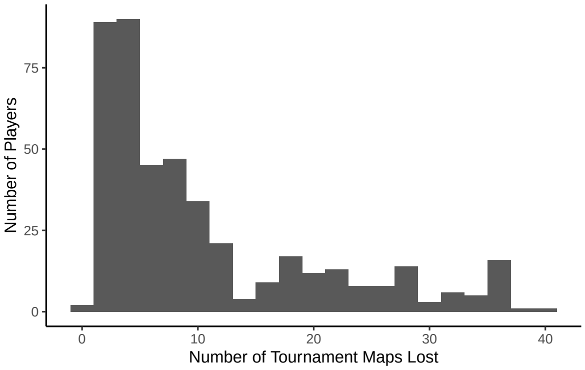

Figure 3: Number of tournament maps lost

Figure 4 shows the distribution of Kill/Death ratios. The median is slightly above the average of 1.0, indicating players in tournaments with significant prize pools are better than the average professional player, as expected. Figure 5 and Figure 6 shows the distributions of lifetime kills and deaths respectively.

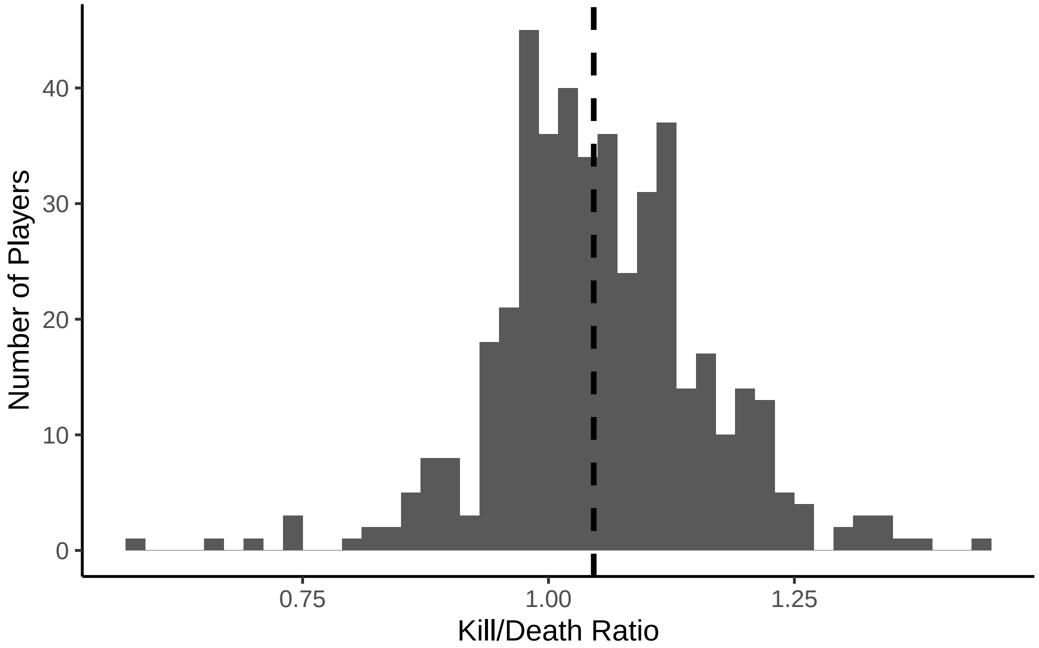

Figure 4: Kill/Death ratios with median

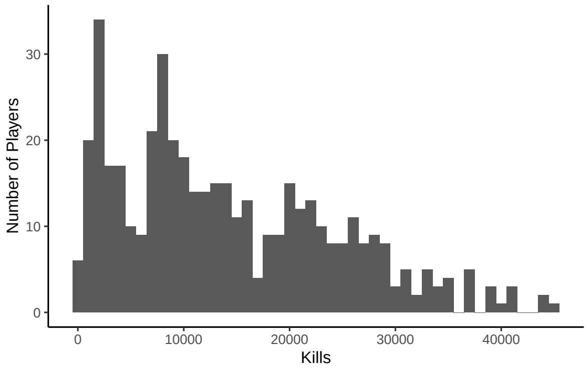

Figure 5: Number of kills

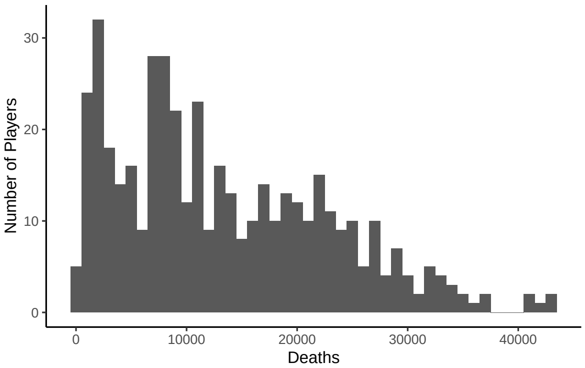

Figure 6: Number of deaths

Figure 7 shows the distribution of ADR. Each player has 100 health and on average, less than 5 enemies will be killed each round, so we expect the ADR to be less than 100.

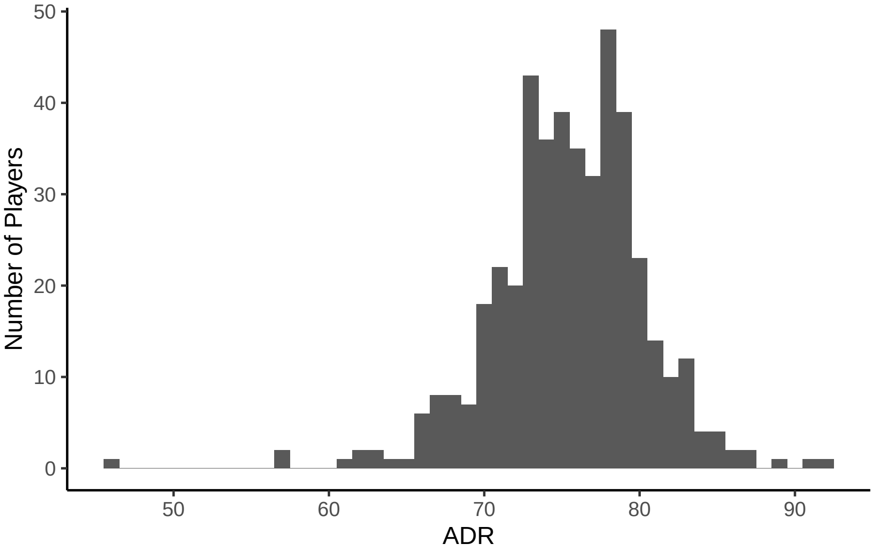

Figure 7: Distribution of ADR

Figure 8 shows the distribution of headshot percentages. A professional *CS* team typically has 4 riflers and 1 AWPer[1], where riflers aim for the head to do more damage while AWPers are more likely to aim for the chest since it is a larger area and they will kill in a single shot regardless. This could explain why the players are distributed with a left skew.

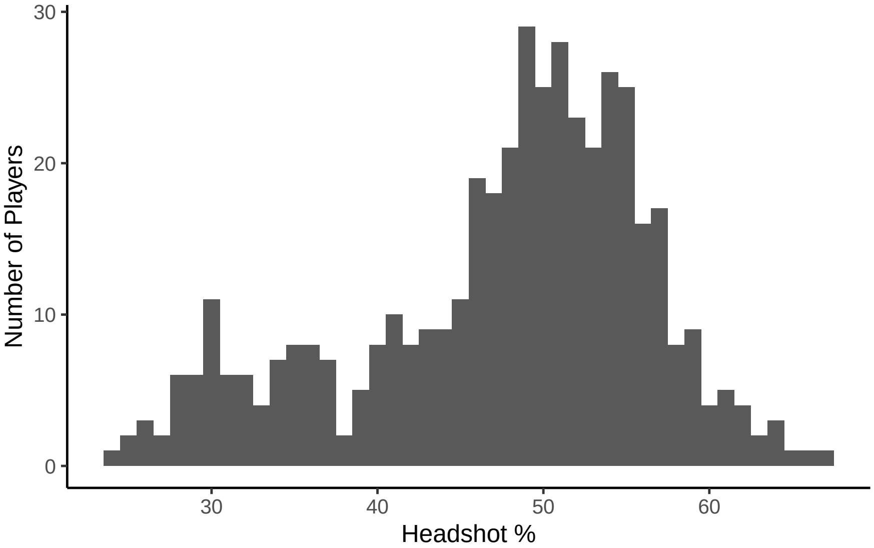

Figure 8: Distribution of Headshot Percentage

[1] An AWP is a powerful but expensive sniper rifle, capable of killing in one shot to the chest

Figure 9 shows the total number of rounds played.

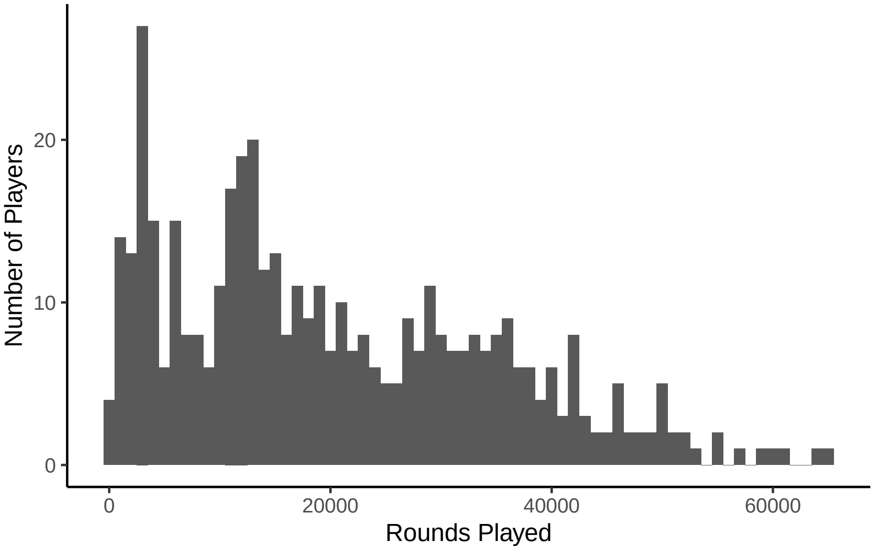

Figure 9: Number of rounds played

3 Model

3.1 Model set-up

The regression model used was:

$$
\begin{aligned}
y_i|\pi_i, &\sim \text{Bern}(\pi_i) && (1)\\
\text{logit}(\pi_i) &= \beta_0 + \beta_1 \times \text{K/D}_i + \beta_2 \times \text{ADR}_i + \beta_3 \times \text{HS }\%_i + \beta_4 \times \text{Rounds played}_i && (2)\\
\beta_0 &\sim \text{Normal}(0, 2.5) && (3)\\
\beta_1 &\sim \text{Normal}(0, 2.5) && (4)\\
\beta_2 &\sim \text{Normal}(0, 2.5) && (5)\\
\beta_3 &\sim \text{Normal}(0, 2.5) && (6)\\
\beta_4 &\sim \text{Normal}(0, 2.5) && (7)
\end{aligned}
$$

where y_i is 1 if player i will win a map and 0 if they will lose. K/D_i is the kill/death ratio of the player, ADR_i is the player's ADR, HS % is the player's headshot percentage, and Rounds played_i is the total number of rounds played by player i.

3.2 Model justification

The outcome of any map played is binary, so a logistic regression model was chosen.

The outcome of each round can often be decided by the outcome of 1v1 duels. Thus, the outcome of a map can be decided by many 1v1 duels. K/D is the number of enemies killed on average per death, indicating the ability for a player to win fights. Since rounds can be won by killing all enemies, we expect K/D to be positively correlated with win rates.

Doing damage to enemies can result in kills by the player or make it easier for their teammates to kill the enemies. Doing more damage in a round is clearly beneficial for winning the round and thus the map, so we expect ADR to be positively correlated with win rates.

HS% is a metric of the player's mechanical skill, which should be part of determining their ability to win fights. Then similar to K/D, we should also expect HS% to be positively correlated with win rates.

The number of rounds played can be used to determine the experience of a player, with more experienced players having played more rounds. Additionally, better players should play in the professional scene more, so they should have more rounds played in total. Given these factors, we expect the number of rounds played to be positively correlated with win rates.

4 Results

Figure 10 shows the relationship between K/D and map win rates with a line provided by the linear regression model. The overall relationships in this figure and following figures are difficult to see using only data points due to the number and distribution. However, the regression line suggests there is a positive correlation between K/D and win rate, as expected when constructing the model.

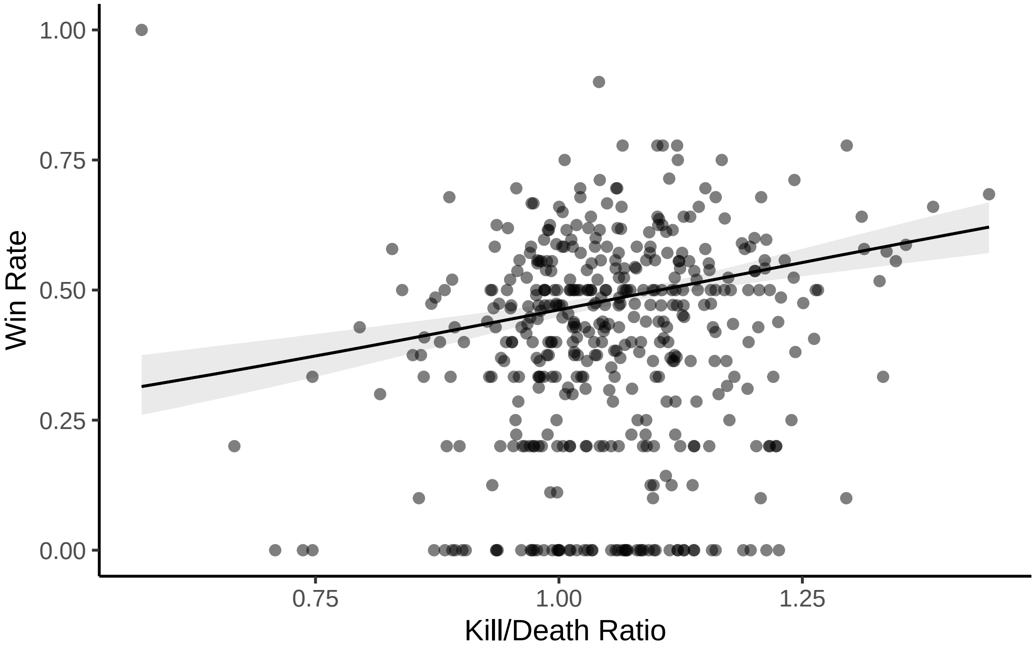

Figure 10: Win rate by kill/death ratio with regression line

Figure 11 shows the relationship between HS% and map win rates with a line provided by the linear regression model. The regression line suggests there is a positive correlation between HS% and win rate, as expected when constructing the model.

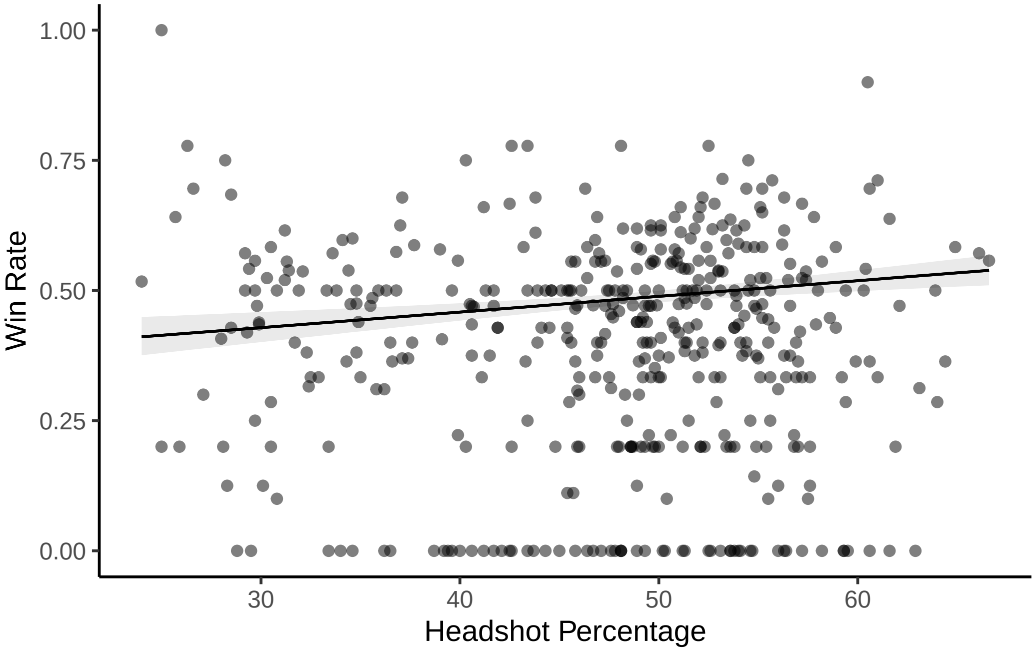

Figure 11: Win rate by headshot percentage with regression line

Figure 12 shows the relationship between ADR and map win rates with a line provided by the linear regression model. Interestingly, the regression line suggests there is a negative correlation between ADR and win rate. This appears to be counterintuitive since more damage is generally good. An explanation for this could be overaggressive players deal more damage but leave their teams in a worse position, leading to lower win rates. Another explanation could be "baiting", where a player waits for their team members to distract enemies in order to have easier kills, allowing them to stay alive, inflating their personal stats but reducing the chance for their team to win.

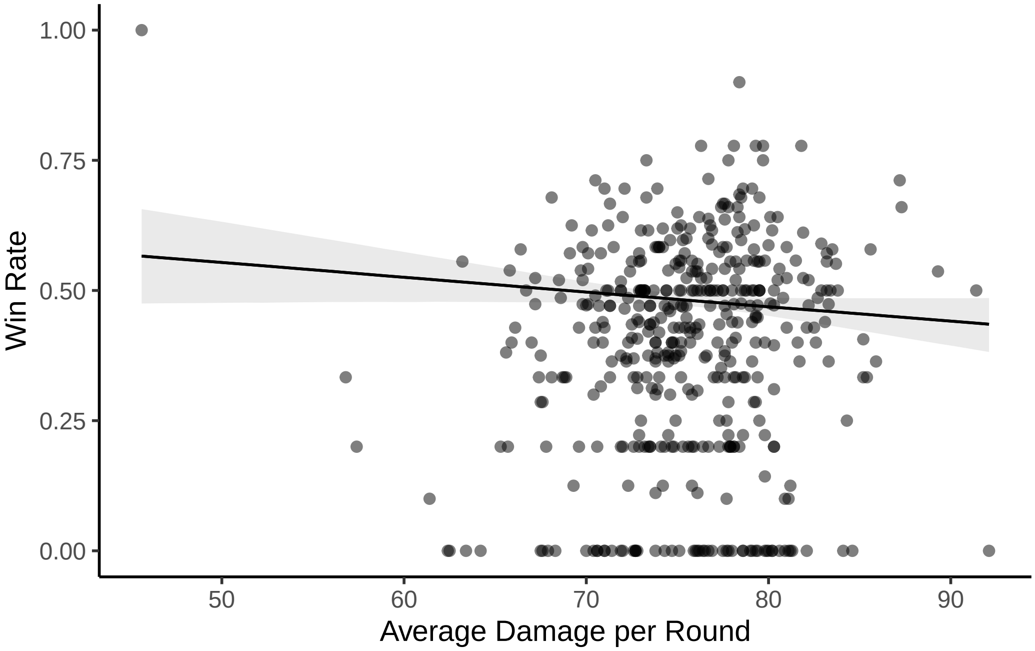

Figure 12: Win rate by ADR with regression line

Figure 13 shows the relationship between total number of rounds played and map win rates with a line provided by the linear regression model. The regression line suggests there is a positive relationship between total rounds played and win rate, as expected when constructing the model.

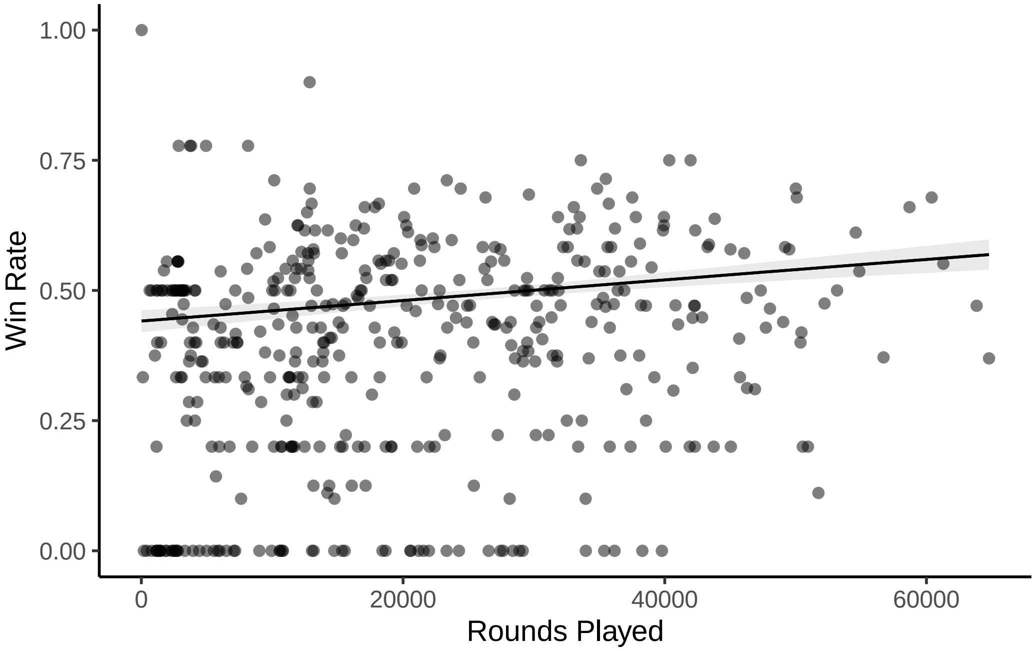

Figure 13: Win rate by total rounds played with regression line

Table 1 shows the model coefficients. Note this is not standardized, so the coefficient for K/D is large since a ± 0.1 change in K/D is a large difference whereas the coefficient for number of rounds is tiny since a change of ± 1 round played is practically nothing.

Figure 14 shows the confidence intervals for the model predictors.

Table 1: Player win rate based on K/D, HS%, ADR, and rounds played

	Tournament Map Win Rate
(Intercept)	−1.495 446
	(0.376 549)
kill_death_ratio	1.463 514
	(0.263 105)
hs	0.012 106
	(0.003 094)
adr	−0.011 318
	(0.006 217)
rounds	0.000 008
	(0.000 001)
Num.Obs.	445
R2	0.963
Log.Lik.	−1018.597
ELPD	−1026.6
ELPD s.e.	24.4
LOOIC	2053.2
LOOIC s.e.	48.9
WAIC	2053.1
RMSE	0.22

5 Discussion

5.1 Impacts of an individual on team performance

As seen in Section 4, some of the predictors we chose, K/D, HS%, and total rounds played, are positively correlated with win rate whereas ADR is unexpectedly negatively correlated with win rate. In a game where killing all members of the enemy team is a win condition, it is clear that individual metrics such as K/D are an important part of a team's overall performance. However, it is also obvious that a team's performance can not purely be explained by an individual's stats.

CS is a team game, every individual has 4 teammates they must work with to win. But even considering every numerical stat for all team members is not sufficient to fully determine a team's success. Similar to physical sports, there are too many other factors to build a perfect model based only on hard numbers. Every round of *CS* has many tactical decisions that must be made. Poor strategies and bad shotcalling can ruin a team's chances at winning while good decision making and communication can make a player invaluable to a team.

For example, Finn "karrigan" Andersen currently plays on FaZe, a top ranking team having just won a large tournament, IEM Chengdu 2024. On paper, karrigan is worse than the rest of his team and many of his competitors with a 0.89 K/D and 43.8% HS%, but he is the IGL (in game leader) for FaZe. As IGL, he provides strategies and makes decisions for the rest of his team, allowing them to more effective. Many other top *CS* teams also have IGLs, who will typically be older and have more experience, but have worse stats.

In summary, individual player stats can be a helpful way to predict their impact on a team's performance, but it never tells the full story. This is applicable to *CS* as well as physical sports and other esports.

5.2 Weaknesses and next steps

As stated in Section 5.1, *CS* is a team game and the current model attempts to predict a team's win rate based only on individual's stats. This limits the predictive power of the model and a next step would be to include the remaining teammate's statistics in the model. This would allow us to predict win rates based on all the members of a team and would also allow us to see how players with different stats interact and effect the team.

A team's win rate does not exist in a vacuum, it must play against another team. Currently, the model is not very useful as it only gives an overall win rate. A much more useful model would give the win rate for a match against another team. A next step would be to include the statistics of the enemy team's players in the model. This would allow us to see some of the interactions between teams and bring the model much closer to something bookmakers and professional sports bettors would use.

One major flaw are the lifetime statistics. These help provide some ways to measure a player's performance but many of these may be outdated, but are still included. Many players have been playing for years and their past performance may no longer be relevant due changes to the game itself or age and declining physical abilities. The model could likely be improved by limiting the lifetime statistics to more recent ones.

Finally, the current model uses only 4 predictors. Many other statistics could be added such as number of opening kills (first kill of the round) and opening deaths. As stated in Section 2 for the explanation of Figure 8, the HS% could be dependent on the weapons used, so weapons used could be included as an interaction variable.

Appendix

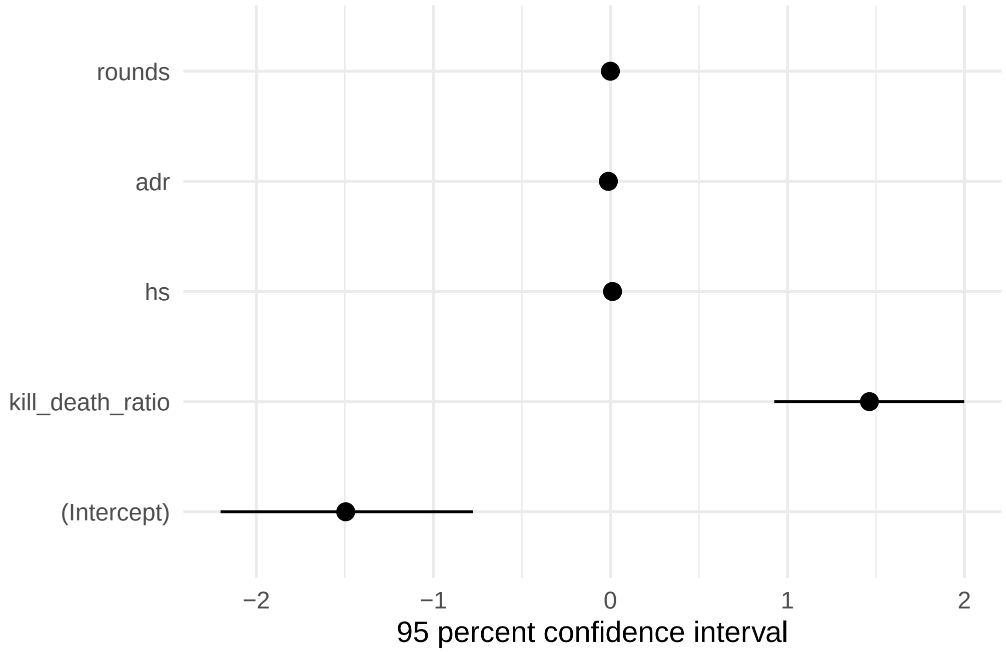

Figure 14: Confidence intervals for predictors win rate

References

Arel-Bundock, Vincent. 2022. “modelsummary: Data and Model Summaries in R.” *Journal of Statistical Software* 103 (1): 1–23. https://doi.org/10.18637/jss.v103.i01.

———. 2024. *Marginaleffects: Predictions, Comparisons, Slopes, Marginal Means, and Hypothesis Tests*. https://CRAN.R-project.org/package=marginaleffects.

Better Collective. 2024. *HLTV.org - the Home of Competitive Counter-Strike*. https://www.hltv.org/.

Brilleman, SL, MJ Crowther, M Moreno-Betancur, J Buros Novik, and R Wolfe. 2018. “Joint Longitudinal and Time-to-Event Models via Stan.” https://github.com/stan-dev/stancon_talks/.

Firke, Sam. 2023. *Janitor: Simple Tools for Examining and Cleaning Dirty Data*. https://CRAN.R-project.org/package=janitor.

Müller, Kirill. 2020. *Here: A Simpler Way to Find Your Files*. https://CRAN.R-project.org/package=here.

Python Software Foundation. 2020. *Python 3.9*. https://www.python.org/.

R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Richardson, Leonard. 2024. *Beautiful Soup 4.12.3*. https://www.crummy.com/software/BeautifulSoup/.

Richardson, Neal, Ian Cook, Nic Crane, Dewey Dunnington, Romain François, Jonathan Keane, Dragoș Moldovan-Grünfeld, Jeroen Ooms, Jacob Wujciak-Jens, and Apache Arrow. 2024. *Arrow: Integration to 'Apache' 'Arrow'*. https://CRAN.R-project.org/package=arrow.

Software Freedom Conservancy. 2024. *Python 4.19.0*. https://www.selenium.dev/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Xie, Yihui. 2014. "Knitr: A Comprehensive Tool for Reproducible Research in R." In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC.