DSC 412 - Final Project Report

Student Name: Bruno Perdigao Lobo

Student ID: bperdig

Email: bperdig@ncsu.edu

November 26, 2024

Dead By Daylight Predictive Killer Tier List Placement Model

Abstract

This project explores the use of machine learning models to predict the performance of different killers in the video game Dead By Daylight, using a dataset consisting of in-game statistics such as kill rates, speeds, and power features. The goal was to create models that could predict a killer's placement in community power rankings, or tier lists, based on these attributes, ultimately helping players make more informed decisions about character selection. Various machine learning algorithms, including decision trees, logistic regression, and neural networks, were trained on the data, and their performance was evaluated using cross-validation techniques.

The dataset was carefully curated from multiple sources, including community-contributed match outcomes and official game statistics, and was augmented to include relevant features such as power attributes. Despite mixed results across the models, they consistently outperformed random guessing, showcasing the potential of machine learning to analyze complex in-game data. The project highlighted the challenges of ensuring model generalization and the importance of careful data preprocessing and hyperparameter tuning.

The findings of this project suggest that while further refinement is needed, machine learning models can offer valuable insights into multiplayer game dynamics. Future work could involve the inclusion of additional features, model improvements, and collaboration with players or game developers to enhance the models' real-world applicability.

Background

Dead by Daylight (DBD) is a popular multiplayer survival horror game developed by Behaviour Interactive, featuring asymmetric gameplay where one player takes on the role of a killer and the other players take on the role of survivors. A key aspect of the game is the variety of killers, each with distinct abilities, playstyles, and performance characteristics. As new killers are introduced, players and the community often rely on tier lists to rank them based on their effectiveness, which is determined by factors such as killer abilities, map design, survivor behavior, and other dynamic gameplay mechanics. However, tier lists can be subjective and heavily influenced by player experience and the meta-game. Predicting a killer's tier placement ahead of time, particularly before players have fully familiarized themselves with the character, presents a significant challenge.

This project aims to address this challenge by leveraging machine learning techniques to predict a killer's placement within tier lists based on an analysis of their inherent mechanics. Through a custom dataset, this project focuses on certain features of each killer, such as their abilities, kill rates, speeds, and other mechanical characteristics, to build predictive models. The goal is to determine an effective method to predict a killer's base performance prior to player feedback and meta shifts by analyzing these characteristics alone. The project also seeks to explore the feasibility of such predictions for newly released killers, providing insights into potential performance expectations right as the killer is introduced.

Tier lists in gaming communities typically serve as rankings that categorize characters or elements based on their perceived strength, utility, or balance. These lists are often subjective and can evolve over time based on community experience, gameplay patches, and shifting metas. However, creating an accurate and data-driven predictive model for tier placement has been a relatively underexplored area, especially when applied to Dead by Daylight, where gameplay mechanics play a complex role in shaping outcomes. By using machine learning techniques to analyze and predict these tier rankings, this project contributes to an ongoing conversation in predictive gaming models, aiming to provide more objective and early-stage assessments of new characters as they are introduced.

Data Analysis

In this project, the primary goal was to analyze and predict the tier placement of different killers in *Dead by Daylight* based on a variety of features. To achieve this, I explored several machine learning models and evaluated their performance on the dataset.

The dataset consists of multiple features, such as “Kill Rate”, “Speed”, and various indicators of a killer's power mechanics, including “Power Chase”, “Power Control”, and “Insta-Down”. For each killer, these features were measured along with their “Placement” in a tier list. The “Placement” is the target variable we aim to predict, where each killer is assigned to one of five categories (D to S).

The analysis began with a focus on understanding the relationship between features and the Placement. Key insights were drawn from the following:

- “Skill Ceiling” and “Average Max Speed” showed high information gain when analyzed, suggesting that a killer's ability to perform optimally (“Skill Ceiling”) and the average maximum speed of their lethal means during a match (“Avg Max Spd”) are significant indicators of their performance in the game.
- “Power Chase”, a feature indicating how useful a killer's power is during a chase, didn’t exhibit strong correlations with the “Placement”. This may be due to the assumption that all killers have effective powers for chasing, which could explain the lack of variability in this feature.

To assess which features are most relevant for predicting “Placement”, I calculated entropy and information gain for the features in the dataset. These methods measure the uncertainty of a feature and its ability to reduce uncertainty about the target variable (“Placement”). Features with higher information gain are considered more influential in predicting the outcome. By calculating information gain for each feature, it became clear that “Skill Ceiling” and “Average Max Speed” were the most predictive features, while others, like “Power Chase”, contributed less to the overall model.

The next step involved exploring various machine learning models, which allowed me to test how well these features could be used to predict a killer's "Placement". Given the randomness inherent in small datasets, I decided to experiment with a variety of models to showcase their comparative performance, rather than selecting one final model. This included models like:

- Logistic Regression: A simple yet effective baseline.
- Decision Trees: Offer interpretability, allowing me to see how different features split the data.
- Random Forests: An ensemble method that reduces the risk of overfitting by averaging multiple decision trees.
- Support Vector Classifier (SVC) and Gradient Boosting: More complex models that may handle non-linear relationships better.
- Neural Networks: Allows for potentially capturing complex patterns within the data at the expense of being more computationally intensive.

Each model was trained and tested with the same dataset to evaluate their performance. However, due to the small size of the dataset and the inherent randomness in the results, there was no single model that consistently outperformed the others. The results were mixed, with accuracies fluctuating significantly across different runs. This variability highlights the challenges of working with a relatively sparse dataset and reinforces the idea that more data would be beneficial for improving model performance.

In this project, the key takeaway was not to identify a "best" model, but rather to showcase a variety of machine learning techniques and how they compare in terms of their ability to predict Placement. The mixed results underscore the influence of randomness in machine learning, especially with a dataset of this size. Even though no model provided a perfect prediction, the accuracy of around 50% (considering 5 possible outcomes) is still a notable achievement compared to random guessing, which would yield only 20% accuracy.

Data Augmentation

The data underwent some modifications to make it suitable for machine learning modeling. While the dataset was not "dirty", and no missing values or glaring inconsistencies were present, as the dataset was handcrafted by me for this project, some steps were necessary to ensure consistency, facilitate easier analysis, and enable effective model training. Below, I detail the key data transformations and augmentations made during the process:

- Categorical Variables: For features such as “Power Chase”, “Power Ctrl”, and “Insta-Down", which are categorical (Yes/No), I ensured they were encoded as binary values (1 for Yes, 0 for No). This transformation allowed for better handling by machine learning models, which typically expect numerical inputs. This step is necessary because machine learning algorithms generally do not interpret text categories directly, and encoding them ensures the models can process these features effectively.
- Scaling Numerical Features: Numerical features like “Kill Rate” and “Avg Max Spd” were standardized to have zero mean and unit variance. This was achieved using “StandardScaler” from the “sklearn” library. Scaling ensures that features with larger ranges (e.g., Kill Rate) do not dominate those with smaller ranges (e.g., Avg Min Spd), leading to better model performance.

The data was derived from various sources to provide a comprehensive set of features for the models:

- Kill Rate values were collected from nightlight.gg, a community-driven website that aggregates match outcome data and provides kill statistics for each killer.
- Speeds were sourced from the official game wiki, which documents detailed statistics about each killer's movement capabilities.
- Power-related features (e.g., Power Chase, Power Ctrl) were gathered through a combination of personal gameplay experience and insights from prominent content creators within the game's community.
- Placement data, reflecting how each killer performs in terms of tier rankings, was sourced both from personal experience and other community-driven data points.

This combination of data from multiple sources ensured a well-rounded dataset, capturing both objective in-game statistics and subjective performance insights.

Model Selection

The models selected for this project were chosen primarily to showcase a variety of machine learning algorithms and their performance on the dataset. This approach was used to explore how different models would handle the same data, providing different sets of outcomes. Given the randomness inherent in machine learning, as well as the diminished size of available data, no single model emerged as clearly superior, so the goal was to test a range of models and observe how each performed under similar conditions.

The models used were selected based on personal experience and familiarity with the algorithms learned throughout the course. Instead of choosing a single model, the selection was made to cover a spectrum of popular techniques, allowing for a comparison between simpler models and more complex models. This allows for insights into which models are better suited for certain kinds of prediction tasks based on the data features.
In terms of building the models, I did not develop them from scratch. Instead, I leveraged existing model architectures available through machine learning libraries. These models are widely used in the machine learning community and are well-documented, making them ideal for comparison. I applied standard implementations of these algorithms with minor adjustments to hyperparameters in order to optimize their performance.

Training Methodology

The models in this project were trained using the data prepared in the earlier steps. After the data was collected and augmented, it was split into training and testing sets. Specifically, 80% of the data was allocated for training, while the remaining 20% was held out for testing. This division allowed for both model training and performance evaluation, ensuring that the results were not biased by the same data that was used for training. The test set remained untouched during training to preserve its integrity for model evaluation.

For training the models, I relied on standard machine learning techniques implemented in Scikit-learn. The training process involved fitting each selected model to the feature set, which included variables such as kill rates, speeds, and power features. These features were used to predict the performance of each killer in the game.

In terms of hyperparameter tuning, I focused on using entropy and information gain calculations to tune parameters. These metrics helped in identifying which features were most informative for predicting a killer's placement and allowed for optimization of the models' decision-making processes.

To avoid issues like overfitting or underfitting, I used cross-validation during hyperparameter tuning, which helped assess how the model would generalize to unseen data. Specifically, I used k-fold cross-validation to split the training data into multiple subsets, training the model on different combinations of these subsets and validating it on the remaining ones. This process helped ensure that the model was not overly reliant on any specific partition of the data.

Results

To evaluate the performance of the models, I used many different metrics. The focus on accuracy, though additional metrics such as precision, recall, and F1 score were also considered.

To test the models, I split the data into training and testing sets, reserving 20% of the data for testing. The models were trained on the 80% training set and then evaluated on the testing set to determine how well they performed on data they hadn't seen before. This allowed for an unbiased evaluation of the models and gave insight into how they might perform in real-world scenarios. Additionally, I used k-cross-validation to ensure that the models were robust and didn't overfit to the training data. This method helped further verify the models' ability to generalize to unseen data.

The output of the models was the predicted placement of each killer in the game, based on the features provided, such as kill rates, speed, and power features. The results were compared against the actual placements to determine the accuracy of the predictions.

As for whether or not the data analysis helped, the answer is yes. I was able to provide the models with meaningful information to predict a killer's performance in the game with a decent accuracy most of the time. The fact that the models were able to make relatively accurate predictions based on these features is evidence that the data analysis contributed significantly to the success of the models.

While the results across different models varied due to the inherent randomness in machine learning processes and the limitations of workable data, as there are only 36 killers in the game at this time, the data-driven approach allowed for meaningful insights into which features were most predictive of a killer's placement. This approach laid the groundwork for comparing the strengths and weaknesses of each model, with certain models performing better than others in some random instances, but none being universally superior. The data analysis, therefore, was integral in helping guide model selection and evaluation.

Future Work

Overall, I’m satisfied with how the models performed, as they generally exceeded the performance of random guessing and provided useful insights into predicting a killer's placement in the game. While the results were not perfect, they demonstrated the potential of using data-driven approaches to make more informed predictions. Given the inherent randomness in this kind of modeling and the relatively small dataset, the models performed well in capturing patterns that could aid decision-making or strategy development for players.

However, there are areas that can be improved. One notable area for enhancement would be exploring different models and techniques. While I explored a range of machine learning models, some models (like decision trees and random forests) could benefit from more specialized tuning, or perhaps integrating ensemble methods for better performance. Additionally, trying out more advanced deep learning approaches or experimenting with feature engineering might uncover more intricate patterns that current models didn’t fully capture.

Looking ahead, a potential next step could involve sharing the insights from this project with the broader community or stakeholders in the game’s ecosystem. By making this project accessible to players, developers, or analysts, the models could be used as a real-time resource to guide decision-making during gameplay. This could evolve into a tool or application where players could input their character's attributes and receive predictive feedback on placement or performance potential, further integrating data science into the gaming experience.

As for future projects, one direction could be to apply similar models to predict other game outcomes, such as the effectiveness of specific killer strategies or character balance. The techniques used in this project, including feature engineering and model comparison, could be directly applied to these areas.

While I am not currently planning another specific project of this nature, I find that this project has greatly expanded my understanding of machine learning, and I would be glad to apply these methods to other data-rich domains in the future.

Stakeholder Acknowledgements

While no specific stakeholders were directly involved in the creation of this project, there are several potential groups who could benefit from the insights and models developed here.

Potential stakeholders include players of the game, game developers, and the broader gaming community. Players could use the model to make more informed decisions about their in-game strategies and character choices, potentially improving their performance in matches. Game developers might find value in using the model to evaluate the balance of characters, fine-tune gameplay mechanics, or gather data-driven insights about player behavior. Content creators or analysts might also use the model to support discussions about strategies or share predictions with their audiences.

However, there are potential negative effects of using this model. One concern could be over-reliance on the model’s predictions, which might discourage players from experimenting with different strategies or enjoying the unpredictability of the game. Additionally, if the model is not robust or is misused, it could lead to flawed assumptions or unfair advantages in gameplay, undermining the intended balance or challenge of the game. Furthermore, if exposed to a wider audience, inaccurate predictions or misunderstandings of the model's limitations could damage trust in its usefulness.

Since no stakeholders were directly involved in the development or testing of this model, I am currently unable to share it with them for direct feedback. It would be useful, however, to involve potential stakeholders, such as players or developers, in the testing phase to refine the model and gather valuable input on its practical application.

Conclusion

This project aimed to explore the use of machine learning models in predicting the performance of killers in a DBD, based solely on game data. Through data collection and feature engineering, I was able to build a set of models that demonstrated promising results, outperforming random guesses and providing insights into the factors influencing killer performance.

While the results were mixed across different models, the overall takeaway is that machine learning can provide valuable support for in-game decision-making and strategy formulation. The models were evaluated based on their ability to predict character placement, and while there were some challenges in achieving consistent high performance, the models still offered a significant improvement over random selection.

The project also highlighted the importance of data quality and feature selection in the success of machine learning models, as well as the need for careful hyperparameter tuning and validation techniques. While the models performed adequately, there is room for further refinement, particularly in terms of improving the generalization ability and ensuring they account for the variability and randomness inherent in multiplayer games.

Looking forward, there are opportunities to build on this work by incorporating additional data sources, exploring more advanced models, and involving stakeholders such as players or developers in future testing.

Overall, this project provided valuable insights into the potential of machine learning in gaming analytics, and while there is still much to explore, it lays a solid foundation for future endeavors in this space.

References

"Dead by Daylight Stats & Icon Packs." *NightLight*, https://nightlight.gg/.

Dead by Daylight Wiki. https://deadbydaylight.fandom.com/wiki/Dead_by_Daylight_Wiki.