Note for Grading: In alignment with the ACM Proceedings format, the *Literature* component of the assignment is integrated into the *Related Work* and *Discussion* sections. All other assignment components are presented as distinct sections in this paper.

Predicting Outcomes in Connect 4: A Configuration-Based Approach

Michael Chan, Dhruv Khatri
University of California, San Diego
mac008@ucsd.edu, dkhatri@ucsd.edu

ABSTRACT

This paper explores the application of machine learning models to classify Connect 4 board states as wins or losses for a given player. The problem of classifying intermediate board states is computationally non-trivial due to nonlinear dependencies between board positions and outcomes. Logistic Regression, Random Forest, and Support Vector Machine (SVM) models were tested against a baseline using feature sets that emphasized positional importance, such as ownership of critical rows and columns. The SVM model demonstrated superior performance, achieving the highest accuracy and balanced error rate (BER) while maintaining precision and recall. These results highlight the importance of capturing nonlinear relationships in strategic tasks, as simpler models struggled to generalize. The work demonstrates the utility of advanced classification models for understanding and predicting outcomes in structured decision-making games.

CCS CONCEPTS

· Artificial intelligence → Machine learning → Supervised learning

KEYWORDS

game analysis, predictive modeling, board game strategies, classification, feature engineering

1. INTRODUCTION

Connect 4 is a two-player turn-based game in which colored tokens are dropped into a 6 row by 7 column grid, with the objective of matching four tokens in a row horizontally, vertically, or diagonally. While simple to explain, Connect 4 involves surprising strategic depth; players improve their chance of winning by securing optimal board positions and forming patterns that maximize winning combinations. As a board game, Connect 4 strikes a unique balance between simplicity and complexity, making it an intriguing subject for computational research. Despite being a solved game (the first player can guarantee a win with perfect play), research continues to explore Connect 4 through solvers and predictive tasks. This paper's contribution is by addressing one such task: classifying board states as wins or losses for a given player. We evaluate the performance of several machine learning models, identifying the most effective approach for this classification problem.

2. RELATED WORK

2.1 Existing Datasets

The dataset used in this paper is sourced from the University of California, Irvine (UCI) Machine Learning Repository [8]. It consists of 67,557 unforced positions on the Connect 4 board after 8 moves. Another dataset originating from Kaggle has also appeared in studies, containing 376,641 board positions which each represent a completed game of Connect 4 [3]. While no additional datasets for the game are commonly cited in the literature, researchers have developed methods to generate datasets through the random simulation of Connect 4 games and use of solvers to determine outcomes and other game properties [4].

2.2 Previous Models

The application of AI to board games, including Connect 4, has historically focused on creating game solvers or highly proficient game-playing models [9]. Approaches such as the neural network system Neural Connect 4 have demonstrated the efficacy of supervised learning methods like backpropagation for training models [6]. Temporal difference (TD) learning with eligibility traces has also been explored, achieving improvements in training efficiency and gameplay performance [7]. Additionally, the Minimax algorithm with Alpha-Beta pruning remains a cornerstone technique for solving Connect 4, while Monte Carlo Tree Search (MCTS) is recognized for its ability to plan strategically by considering the long-term consequences of moves [2, 5].

More closely related to the current study's focus, simpler classification tasks such as predicting game outcomes from board configurations or estimating the number of moves until game completion have also been explored. For example, Kim et. al [2019] employed logistic regression, support vector machines with linear kernels, and gradient boosting machines (GBM) to predict game outcomes using a combined dataset of UCI, Kaggle, and generated board states. These approaches highlight the potential for machine learning techniques to analyze and predict patterns in Connect 4 gameplay.

3. DATASET

3.1 Dataset Description

The dataset contains a comprehensive set of all board configurations in Connect 4 that satisfy the conditions:

1. Exactly 8 turns have been played, or each player (denoted as X and O) has placed 4 pieces.
2. The game is in-play; neither player has won.
3. The move in the following turn is not forced.

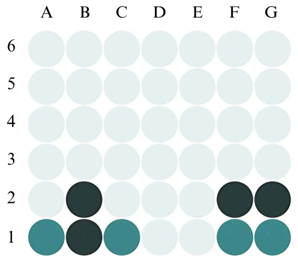


Fig 1: An example board

There are a total of 67,557 instances that meet these criteria. An example board is shown to the left **(Figure 1)**. Rows are denoted by letters (A-G) and columns by numbers (1–6), where A6 represents the upper-left and G1 the lower-right.

The features of the dataset are categorical, corresponding to the 42 positions on the board. Each position can take one of three values: *x* if player X has placed a piece, *o* if player O has placed a piece, or *b* if the position is empty. The target variable represents the game outcome relative to player X, the starting player, and is categorized as *win*, *loss*, or *draw*. In total, 44,473 boards (65.8%) result in a win for player X, 16,635 boards (24.6%) result in a loss for player X, and 6,449 boards (9.6%) result in a draw.

3.2 Exploratory Analysis

An initial analysis of the dataset's patterns and properties must consider both the mechanics of Connect 4 gameplay and the board constraints imposed by the dataset. First, the rules governing how pieces drop to the lowest available position introduce spatial and positional biases, emphasizing the importance of bottom-row positions which often serve as the foundation for future moves. Patterns of adjacent pieces or multiple pieces in a given line are also critical to reach the objective of aligning 4 pieces horizontally, vertically, or diagonally.

In addition to these mechanics, the constraints specified for the dataset impose further structure on the board states. The requirement that each board must represent a game with 8 moves ensures that games are relatively early in progress, further prioritizing the importance of lower positions. Moreover, the conditions that the board is in-play and that the next move is not forced exclude the presence of certain configurations, such as three-in-a-row patterns in any direction.

In consideration of these factors, our exploratory analysis aims to assess positional biases and spatial relationships across outcome conditions. To examine positional bias, we analyzed the occupancy rates of each position across all board configurations. **Figure 2** illustrates these rates, with darker red hues indicating positions more frequently occupied by either player and lighter hues less frequently occupied. As expected, positions in the lowest rows are most often filled, while positions in the highest rows are rarely occupied. Interestingly, heterogeneity is also observed within rows, and C1 stands out as the most frequently occupied position. This pattern likely reflects the strategic value of central columns, which offer more opportunities for forming winning alignments in multiple directions. Additional breakdowns of occupancy by outcome are provided in **Supplementary Figures 1, 2.**

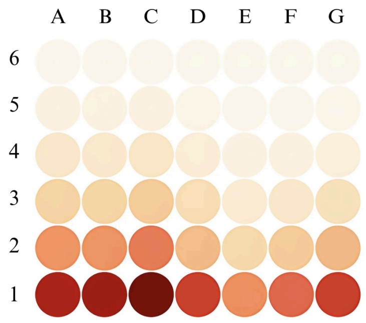
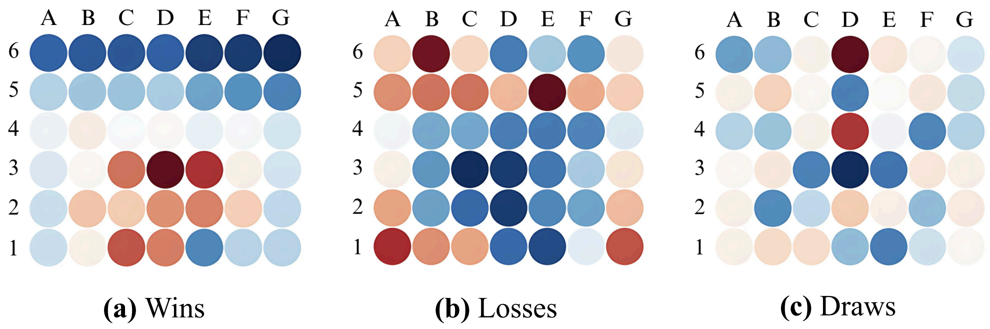


Fig 2: Occupancy status by position



(a) Wins

(b) Losses

(c) Draws

Fig 3: Fold change of *x* (red) with respect to *o* (blue) across outcome conditions

After identifying the center column and bottom row as key regions of interest, we calculated the mean number of pieces placed by players X and O in these areas, stratified by game outcomes. In the center column, both players placed pieces more frequently on boards where player X achieved a win compared to boards resulting in a loss or draw (**Table 1a**). An inverse trend was observed in the bottom row: on average, both players placed 0.2-0.4 fewer pieces in boards where player X won, relative to losses or draws (**Table 1b**).

Table 1: A summary of column and row statistics

(a) Mean number of pieces in center column

	Win	*Loss*	*Draw*
x	0.639	0.393	0.404
o	0.871	0.445	0.513

(b) Mean number of pieces in bottom row

	Win	*Loss*	*Draw*
x	1.984	2.332	2.129
o	1.719	2.116	1.956

Finally, we directly compared the distribution of *x* and *o* pieces across win, loss, and draw outcomes. For each position, the proportion of *x* and *o* pieces was calculated across all boards for a given outcome. The fold change of *x* relative to *o* (*x*/*o*) was then determined and mapped to a color gradient. Positions where *x* is strongly preferred are shown in red, while those dominated by *o* are shown in blue. This analysis revealed stark differences in spatial dominance between outcomes. Boards where player X wins consistently show *x* controlling the bottom-central region from C1 to F3 (**Figure 3a**). Conversely, boards where player X loses show the same central region dominated by *o* while *x* is relegated to the outer corners (**Figure 3b**). For draw outcomes, no discernible spatial pattern is observed, suggesting a balanced distribution of pieces (**Figure 3c**).

4. PREDICTIVE TASK

4.1 Task Definition

We define our predictive task as classifying whether a given 8-play Connect 4 board state will lead to a win or loss for player X. The task is relevant for analyzing gameplay strategies and offering insights into the patterns that lead to victory. By strictly focusing on win/loss outcomes, the model mirrors the strategic mindset of players who aim to either secure a win or avoid a loss. Of note, draw scenarios are excluded from the remainder of this study for two primary reasons: first, draws typically result from evenly matched gameplay or suboptimal moves, offering limited insight into decisive strategies. Second, training the model on win/loss outcomes aligns with players' primary goals, as draws are rarely a preferred outcome.

4.2 Evaluation Metrics

To evaluate the validity of model predictions, all models are trained and tested using a random 90-10 train-test split. Performance is assessed using a range of standard metrics, namely accuracy, precision, recall, and F1 score, to provide a comprehensive understanding of each model's strengths and weaknesses.

4.3 Feature Engineering

Guided by insights from the exploratory data analysis, our main objective is to engineer features that encode board centrality, individual positions, and row-column relationships. As the dataset already provides pre-cleaned positional data for each board square, our analysis does not need a preprocessing pipeline. While preprocessing would be required to expand the dataset further, such as by simulating additional board configurations, the size of the dataset does not present an immediate issue for our analysis.

The first feature we engineered was the middle-bottom D1 position, identified as a pivotal position for creating winning combinations. This motivated the development of a comprehensive feature vector encompassing all board positions and their occupancy status. Initially, we mapped the values *x*, *o*, and *b* to numeric values {0, 1, 2}, but transitioned to one-hot encoding to avoid ranking biases between players and empty spaces. We further enhanced the feature set by incorporating counts of pieces each player owned in the bottom row and middle column. These were encoded as numeric values representing the number of pieces owned by each player. Additional features, such as the number of three-in-a-row sequences and diagonal relationships, were explored but ultimately excluded. Three-in-a-row features proved inconclusive due to the unforced nature of board positions, while diagonal relationships were deprioritized at this early stage of play, where fewer pieces are available for stacking.

4.4 Baseline Models

We define a baseline model for this task which classifies a game as a win for player X if their chip is in the D1 position on the board. In literature, this bottom-middle position is often noted as the most valuable position of the board, as it can be used to create the most winning combinations. This feature serves as a simple and intuitive benchmark for evaluating the performance of more advanced models.

5. MODEL DESIGN

The classification of Connect 4 game outcomes, we argue, is best-suited for a Support Vector Machine (SVM) model using the Radial Basis Function (RBF) kernel. Our exploratory analysis identified spatial features, specifically the positions of each player's pieces on the board, as critical to predicting game outcomes. Given the interdependence of the 42 board positions, the model must accommodate non-linear relationships. The RBF kernel is particularly effective in this context, as it projects the data into higher-dimensional feature space, allowing the model to find non-linear decision boundaries to separate the classes. Moreover, SVM models are designed to handle large feature vectors of diverse types, including one-hot encodings and numerical values.

SVM using RBF kernel prediction is defined as:

$$f(X_i) = \Phi(X_i).\, w + b \qquad [1]$$

where

$$w = \sum_{i=1}^{j} y_i \alpha_i \Phi(X_{i,train}) \qquad [1]$$

Model optimization is achieved through a combination of efficient feature engineering and hyperparameter tuning. Initially, the model is tested using the default features provided in the dataset, where each board position is represented as a one-hot encoded variable. Based on insights from the exploratory analysis, additional features are then extracted to improve model performance. After maximizing the features extracted from the dataset, hyperparameters are optimized to further improve accuracy. A grid search is conducted over a range of values for C, while keeping the kernel and gamma values constant. The resulting optimal set of parameters is then used to train and execute the model.

Despite the high dimensionality of the feature vectors due to one-hot encoding representations of the board, no scalability issues emerged during this process. The computational cost of both the model prediction and grid search for tuning did not create noticeable problems. Overfitting the model, however, remained a concern, as the board configurations in the dataset do not generalize to Connect 4 board states at various stages of the game. Acknowledging this, we narrow the intended scope of our model to predicting outcomes from early state data. Another major issue we faced with preliminary attempts was an inexplicably poor accuracy rate, which we were eventually able to attribute to sample boards which resulted in a draw outcome. After careful consideration and for the reasons outlined in Section 4.1, we decided the draw state was not relevant to our classification task and opted to remove draw outcomes from the dataset.

In our design process, we evaluated four different models for comparative purposes: the baseline model outlined in Section 4.4, the SVM model discussed earlier, a Logistic Regression model, and a Random Forest model.

Defined as:

$$f(X_i) = \frac{1}{1+e^{-(b_0+b_1\cdot x_1+\cdots+b_n\cdot x_n)}}$$ [1],

Logistic Regression was selected for its simplicity and ability to provide probabilistic classification. However, its assumption of a linear relationship between features and outcomes limits its effectiveness for this task, which involves complex, non-linear patterns.

In contrast, the Random Forest model, represented by the equation:

$$f(X_i) = \hat{C}_B(X_i) = \text{majority vote}\{\hat{C}_b(X_i)\}_{b=1}^{B}$$ [1],

is specifically suited to accommodate non-linear relationships and can manage many features without a great risk of overfitting. However, this model requires careful hyperparameter tuning and is very reliant on the quality of feature engineering.

This selection of models represents a range of strengths and weaknesses that we believe is necessary to provide a comprehensive evaluation of Connect 4 classification, and will help to verify and contextualize the performance of our proposed SVM model across multiple measures.

6. EXPERIMENTAL RESULTS

The performance of each model was assessed on its accuracy, balanced error rate (BER), precision, recall, and F1 score. **Table 2** displays each of these metrics, ranking the models in order of increasing accuracy: baseline, logistic regression, random forest, and SVM.

The baseline model performed the worst, with an accuracy of 0.3594 and a BER of 0.4834. Its high precision of 0.9298 but low recall of 0.0391 suggests that while winning predictions were highly accurate, the model struggled to correctly predict losses. Notably, the baseline model predicted losses for 5940 out of 6111 games in the test set, which likely inflated its precision **(Figure 4a)**. Both the logistic regression and random forest models showed significant improvement over the baseline. The logistic regression model, fine-tuned to an accuracy of 0.8079, demonstrated balanced performance with a precision of 0.8689 and recall of 0.8373. The random forest model achieved a similar accuracy of 0.8207 but showed reduced precision for winning outcomes at 0.7966, although recall increased to 0.9806. The SVM model outperformed all of the alternatives, achieving the highest accuracy of 0.8715 and the lowest BER of 0.1435. It also had a precision of 0.9043 and recall of 0.9023, indicating a reliably high number of correct positive and negative predictions (**Figure 4b**).

	Accuracy	*BER*	*Precision*	*Recall*	*F1*
Baseline	0.3594	0.4834	0.9298	0.0391	0.0751
Logistic Regression	0.8079	0.2065	0.8689	0.8373	0.8528
Random Forest	0.8207	0.2579	0.7966	0.9806	0.8791
SVM	0.8715	0.1435	0.9043	0.9023	0.9033

Table 2: Model validation statistics

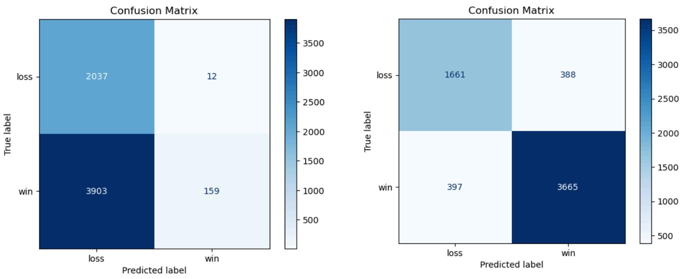


Fig 4: Confusion matrices for (a) baseline and (b) SVM model

The results underscore the importance of selecting features that capture the positional and relational nuances of the board. The baseline model performs poorly, as it relies solely on the ownership of the middle bottom position and cannot account for scenarios where this feature does not determine the outcome. In contrast, the incorporation of more complex feature representations such as ownership of the bottom row and middle column significantly enhances performance across all other models, as evidenced by the feature coefficients in logistic regression (**Supplemental Figure 3**). Here, the high magnitude of the coefficients in significant regions reflects the strong linear relationship between these features and the likelihood of a win or loss.

Despite utilizing the same feature representations, the Logistic Regression and Random Forest models fell short compared to the proposed SVM model, which achieved the highest accuracy along with robust precision and recall. Its success stems from the model's ability to understand non-linear relationships between features and outcomes. While logistic regression performed well due to effective feature engineering, its assumption of linearity limited its ability to fully model these relationships. Similarly, the random forest model excelled in predicting wins but struggled with precision due to class imbalance and overshadowing of key engineered features by positional data. Overall, the SVM model’s ability to adapt to non-linear patterns makes it best suited for the Connect 4 prediction task, outperforming simpler models that fail to account for the intricate dependencies in game states.

7. DISCUSSION

7.1 Comparison to other approaches

Our developed model is unique in that it is constructed using only early-stage game data. Similar approaches have been explored using a combined dataset of UCI, Kaggle, and generated data to train logistic regression and SVM models, achieving a maximum accuracy of 0.844 [4]. However, the models in that study also incorporate draw outcomes, which may account for their lowered accuracy. The most successful approach has been the use of neural networks to predict game outcomes, as deep learning models can capture the task's complexity and all potential outcomes. But, due to the data size and computational resources required, this approach falls outside the scope of the current project.

7.2 Limitations

Our analysis has confirmed that position-based dependencies are key factors that influence game outcomes in Connect 4, with the prioritization of the middle column and bottom row being an effective strategy for increasing the likelihood of a winning outcome. However, the data used in this study is limited to the first 4 moves made by each player, restricting the number of patterns which can be extrapolated by the model. Additionally, we assume that each player made the most optimal move at each turn to reach the game outcome, which may not always reflect real gameplay. These limitations could be addressed by gathering more comprehensive data, either from published datasets of real games or by generating games using AI.

8. CONCLUSION AND FUTURE WORK

Based on model results, Connect 4 prediction is well suited for machine learning and AI applications. We have identified position based dependencies which contribute to game outcomes. Combined with grouped positions of significance as features, we are able to train an effective machine learning classifier. Through model training and fine tuning, we have proven the SVM classifier is the most effective at early stage game prediction, trumping Logistic Regression and Random Forest classifiers.

This work has implications for puzzle solving and game theory based applications. Early stage game prediction is a difficult task due to the limited input and multitude of possibility outcomes. Our model results set a precedent for approaching these tasks, first focusing on the data in each game state and then extrapolating and engineering diverse features based on correlation to different outcomes. This research can be extended to incorporate all game stages as patterns in more developed games may inform early stage predictions. Additionally, this data and feature set can be translated into an generative AI model for Connect 4 game generation. Using each piece as a token, optimal moves leading to winning outcomes can be generated from early game states, improving the ability of current AI models.

REFERENCES

[1] R. Alkhanbouli, A. Al-Aamri, M. Maalouf, K. Taha, A. Henschel and D. Homouz, "Analysis of Cancer-Associated Mutations of POLB Using Machine Learning and Bioinformatics," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 21, no. 5, pp. 1436-1444, Sept.-Oct. 2024, doi: 10.1109/TCBB.2024.3395777.

[2] H. Baier and M. H. M. Winands, "Monte-Carlo Tree Search and minimax hybrids," 2013 IEEE Conference on Computational Intelligence in Games (CIG), Niagara Falls, ON, Canada, 2013, pp. 1-8, doi: 10.1109/CIG.2013.6633630.

[3] “Connect-4 Game Dataset.” [Online]. Available: https://kaggle.com/tbrewer/connect-4.

[4] Kim, Luke, et al. *Applying Machine Learning to Connect Four*, 2019, cs229.stanford.edu/proj2019aut/data/assignment_308832_raw/26646701.pdf.

[5] Polak, E. (1989). Basics of Minimax Algorithms. In: Clarke, F.H., Dem’yanov, V.F., Giannessi, F. (eds) Nonsmooth Optimization and Related Topics. Ettore Majorana International Science Series, vol 43. Springer, Boston, MA. https://doi.org/10.1007/978-1-4757-6019-4_20

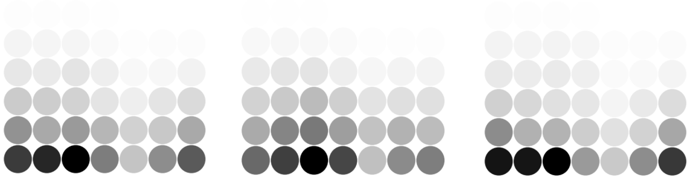
[6] M. O. Schneider and J. L. Garcia Rosa, "Neural Connect 4 - a connectionist approach to the game," VII Brazilian Symposium on Neural Networks, 2002. SBRN 2002. Proceedings., Pernambuco, Brazil, 2002, pp. 236-241, doi: 10.1109/SBRN.2002.1181482.

[7] M. Thill, S. Bagheri, P. Koch and W. Konen, "Temporal difference learning with eligibility traces for the game connect four," 2014 IEEE Conference on Computational Intelligence and Games, Dortmund, Germany, 2014, pp. 1-8, doi: 10.1109/CIG.2014.6932870.

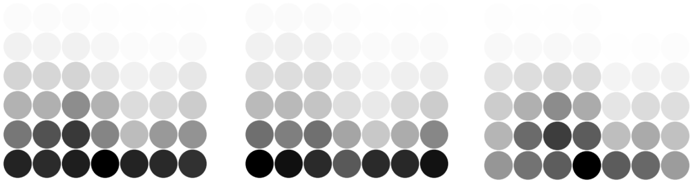
[8] “UCI Machine Learning Repository: Connect-4 Data Set.” [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Connect-4.

[9] B. Xia, X. Ye and A. O. M. Abuassba, "Recent Research on AI in Games," 2020 International Wireless Communications and Mobile Computing (IWCMC), Limassol, Cyprus, 2020, pp. 505-510, doi: 10.1109/IWCMC48107.2020.9148327.

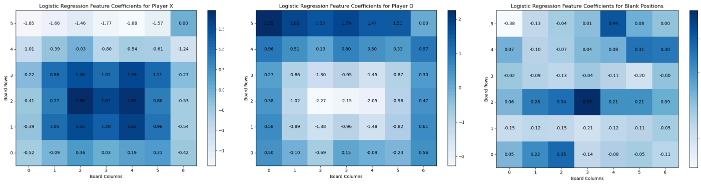
APPENDIX



Supplemental Figure 1: Average board occupancy of *x* for all boards (left), wins (center), and losses (right)



Supplemental Figure 2: Average board occupancy of *o* for all boards (left), wins (center), and losses (right)



Supplemental Figure 3: Coefficients of Logistic Regression model in each position for Player X (left), Player O (center), and Blanks (right)