EE860 - Final Term Project – Option 1

# Dataset:

The dataset, named "Chess End-Game: King-Rook versus King-Pawn on a7 (KRKPA7)," focuses on a specific chess endgame scenario involving White (King and Rook) against Black (King and a pawn on a7). Created by Alen Shapiro and managed by Rob Holte, the dataset was obtained from Peter Clark of the Turing Institute in 1989. It comprises 3196 instances with 36 attributes, including classes indicating whether White can win ("won") or cannot win ("nowin"). The class distribution reveals that in 52% of positions, White can win, while in 48%, White cannot win, based on the Black pawn's ability to advance safely.

# Abstract:

This research project focused on the binary classification of a chess dataset, specifically targeting the King-Rook versus King-Pawn on a7 (KRKP) scenario. Serving as a benchmark for diverse classification approaches, this dataset's enduring relevance in the field underscores its significance. Utilizing the pycaret library, different machine learning models are compared, with the decision tree classifier emerging as the optimal performer, achieving an accuracy rate of nearly 99%. Comprehensive performance evaluations, employing various matrices, highlighted the robustness of these models in discerning intricate patterns within the chess dataset. Subsequently, the selected model was implemented on the Google Cloud Platform (GCP), emphasizing the practicality and scalability of machine learning applications in cloud environments. This integration underscores the broader implications of deploying machine learning models in cloud-based solutions, reflecting the academic standard and the far-reaching potential of machine learning methodologies in analytical and scalable contexts.

# Literature Review:

The categorization of datasets into distinct classes has long been a persistent challenge, retaining its significance due to the ongoing pursuit of heightened precision and methodologies for classifying intricate datasets. The scenario involving King-Rook versus King-Pawn on a7 (KRKP) has served as a prevalent benchmark for evaluating diverse classification approaches and various machine learning models, underscoring its enduring relevance in the field. This literature review examines how different authors use the KRKP dataset to benchmark their studies on machine learning classification models, exploring various practices aimed at improving existing classification techniques.

Rajarajeswari & Somasundaram (2012) used KRKP dataset along with some other datasets to study the importance of feature selection to improve the accuracy of Data classification. Feature selection is an important pre-processing technique that can significantly improve the performance of the machine learning model and the selection of appropriate feature selection is crucial in the design of model. The research assesses the efficacy of ID3 (Iterative Dichotomiser 3), a decision tree-based algorithm, C4.5 (an enhanced iteration of ID3), and Naive Bayes classifiers (a probabilistic algorithm based on Bayes' theorem) across chosen datasets. This evaluation employs diverse feature selection methods, substantiated by comprehensive experiments and comparisons. The three models performed well under different feature selection methods and for the KRKP dataset, C4.5 and ID3 was found to have better performance in terms of accuracy as compared to Naïve Bayes classifier.

In another investigation(Dietterich, 1999), the KRKP dataset was employed to assess the effectiveness of various methods for ensembling decision trees in addressing classification problems. This approach involves the combination of distinct individual classification trees, with the ultimate decision determined through a voting mechanism based on the contributions of these individual trees. It is proven that ensembling considerably improves the accuracy (Breiman, 1996b; Kohavi & Kunz, 1997; Bauer & Kohavi, 1999; Maclin & Opitz, 1997). This particular study focused on evaluating the impact of randomization, bagging, and boosting on the performance of the C4.5 decision tree algorithm. Unlike bootstrapping and adaptive boosting, where the training algorithm is iteratively invoked with different training sets, bagging entails drawing samples from the training pool, repeating training with replacement. However, these methods are susceptible to the model's stability, indicating how a minor alteration in training can result in significant changes in the learned classifier. The study introduces the concept of randomization, wherein the decision regarding the splitting of a tree at an internal node is randomized, imparting resilience to model stability. This proposed technique was implemented within the C4.5 framework and compared against bagging and boosting. Experimental results conducted on the KRKP dataset and 32 other datasets demonstrated that boosting yielded better outcomes.

Sylvester & Chawla(2006) addresses the limitation in majority voting where each classifier's contribution is treated equally, regardless of its relative improvement. To enhance ensemble algorithm performance, genetic algorithms were employed to assign weights to each classifier, thereby enhancing the overall accuracy. Genetic algorithms, drawing inspiration from the evolution of biological populations, serve as computational models in this context. In the study, the process involves learning a set of classifiers, where the predictions of each classifier on a validation set become features for a new dataset. This new dataset is then utilized to train a meta-classifier, which learns a function based on the predictions of the base classifiers, ultimately making composite predictions. Stacking, as described, can mitigate the risk of overfitting. The implemented system, termed EVEN (EVolutionary ENsembles), employs a genetic algorithm for the meta-learning phase. Eleven datasets was used in the study where KRKP was one of them where the KRKP data classification achieved 99.25% accuracy using the method. The study utilized eleven datasets, including KRKP, where the classification of KRKP data achieved an accuracy of 99.25% employing the specified method.

# Feature Engineering:

All attributes were used for training as the data represents a sequence of moves in chess. Data was tested for duplicates or null values. All the 37 columns in the dataset were categorical. It was observed that one hot encoding was not performed in all the data. PyCaret employed normal label encoding (‘0’ and ‘1’) for all the columns that had only two entries (either ‘t’ or ‘f’) and one hot encoding was used on column ‘n’ where there were 3 categories (‘n’, ’w’ and ‘b’). This resulted in a total of 39 columns in the transformed dataset.

# Video Presentation:

[EE8603 - Machine Learning Project Presentation (loom.com)](https://www.loom.com/share/b9ed92175b2245a1a83fac95d1485790)

# GitHub Repository:

<https://github.com/dsatheesan/EE8603/tree/main>

# Performance Metric:

In a balanced binary classification scenario without emphasis on false positives or false negatives, accuracy and F1 score are suitable metrics for comparing models. The F1 score, being the harmonic mean of precision and recall, provides a balanced assessment, making it valuable when both types of errors are of equal concern.

# Experimental Results:

The best-performing model among the tested models appears to be the Decision Tree Classifier, achieving an impressive accuracy of 99.33%, an AUC of 99.3%, and high values across other crucial metrics such as recall, precision, F1 score, Kappa, and MCC. This indicates that the Decision Tree Classifier demonstrates exceptional predictive capabilities and a well-balanced trade-off between precision and recall. The model's ability to distinguish between classes is highlighted by its high AUC score of 99.3%, while its overall accuracy of 99.33% showcases its effectiveness in making correct predictions. Additionally, the Decision Tree Classifier exhibits robustness, as evidenced by its high F1 score, which considers both false positives and false negatives. The model's training time (TT) of 0.47 is also noteworthy, suggesting efficiency in the learning process. Overall, the Decision Tree Classifier emerges as the standout model in terms of both accuracy and comprehensive performance metrics.

The tuned Decision Tree model demonstrates strong performance with an accuracy of 94.32%, a high AUC of 98.99%, and balanced precision and recall at 94.62%. The F1 score is 94.32%, highlighting its effectiveness in handling false positives and false negatives. The model also exhibits a high Kappa coefficient of 88.65% and an MCC of 88.94%, indicating reliability and agreement with actual class labels. Overall, these metrics collectively underscore the model's effectiveness and robustness in the classification task.

Conclusions:   
The project focused on binary classification to predict the outcome of a chess game either a win or no win. The pycaret library was employed, resulting in a notable 99% accuracy rate. Also, the decision tree classifier outperformed other models, suggesting a potential sequential influence of move choices on the game's outcome. While more advanced classification methods using machine learning or deep learning frameworks like TensorFlow are available, the current investigation highlights the effectiveness of the pycaret library for this specific chess problem, yielding good results.