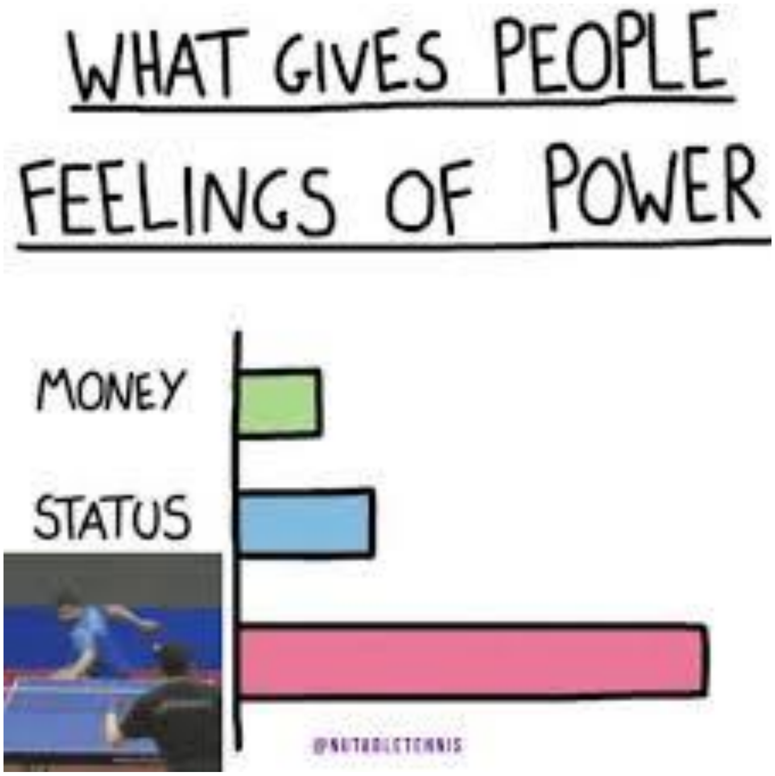
COMP4702 Machine Learning Assignment

Table Tennis Player Swing Talent Identification Analysis

Eileen Ip | 47451063 | Words: 4988



Contents

Introduction....3

Benefits....3

Dataset Information....4

Exploratory Data Analysis....4

Variables....6

Feature Engineering....7

Analysis Plan....7

Dataset Assumptions:....7

Evaluation Metrics Scores and Charts....8

Tuning Hyperparameters....8

Analysis Plan 1: Principal Component Analysis....8

Analysis Plan 2: Random Forest Model....9

Analysis Plan 3: Light Gradient Boosting Model....10

Analysis Plan 4: Deep Neural Network....11

Model Implementation....13

Model 1: Principal Component Analysis....13

Model 2: Random Forest Model....14

General Model....14

Optimised Model....15

Model 3: Light Boosting Model....16

General Model....16

Optimised Model....17

Model 5: Deep Neural Network....19

General Model....19

Optimised Model....20

Limitations....21

Future Work....22

Conclusion....22

Appendix....23

Introduction

Building on previous research that explored the use of machine learning models to predict player attributes such as age, gender, experience years, handedness, and swing type. The report introduces a new focus of classifying player talent based on table tennis swings. Leveraging the same dataset, the objective is to automate the talent identification process. This project was initiated at the request of stakeholders seeking to automate player selection to encourage their gambling habits and funds by betting on their best players.

The report details the following steps:

1. The dataset explored using a heatmap to identify the key features for creating a binary target variable (to identify talent).
2. A principal component analysis was applied to reduce the dimensionality of the dataset.
3. Three models were trained to determine a suitable automation to capture this dataset. The models include:
    - Random Forest Classifier
    - Light Gradient Boosting Classifier
    - Deep Neural Networks
4. Discussed in Analysis Plan, each model performed a basic model with default parameters and an optimised model using a range of parameters tuned with random search cross-validation (RandomisedSearchCV). Several evaluation metrics and visualisations are used.
5. Concludes with an analysis of limitations, proposed future research directions and a summary of key insights.

Benefits

This project offers several key advantages:

- Replacing subjective human judgment with machine learning models ensures fair and data driven player selections. This removes personal biases, and inconsistency often seen in manual scouting.
- By analysing the subtle patterns, the models can uncover high potential players who might be overlooked by traditional evaluation methods. This leads to a more comprehensive understanding of player capability.
- Accurate identification of skilled players creates a competitive edge. Data driven selection enables better team composition and game strategies.
- Concentrate resources on athletes with the highest likelihood of success, ensuring efficient processes and better return on investment in talent development.
- The automation significantly reduces reliance on large scouting teams or time-consuming trials, cutting operational costs and speeding up the recruitment process.

Dataset Information

The dataset comprises 97354 rows and 50 columns of swing information collected via 9-axis sensors that captured detailed movements. No missing values were present. Unknown categorical values were deemed irrelevant, as talent classification was decided based on physical sensory data.

Exploratory Data Analysis

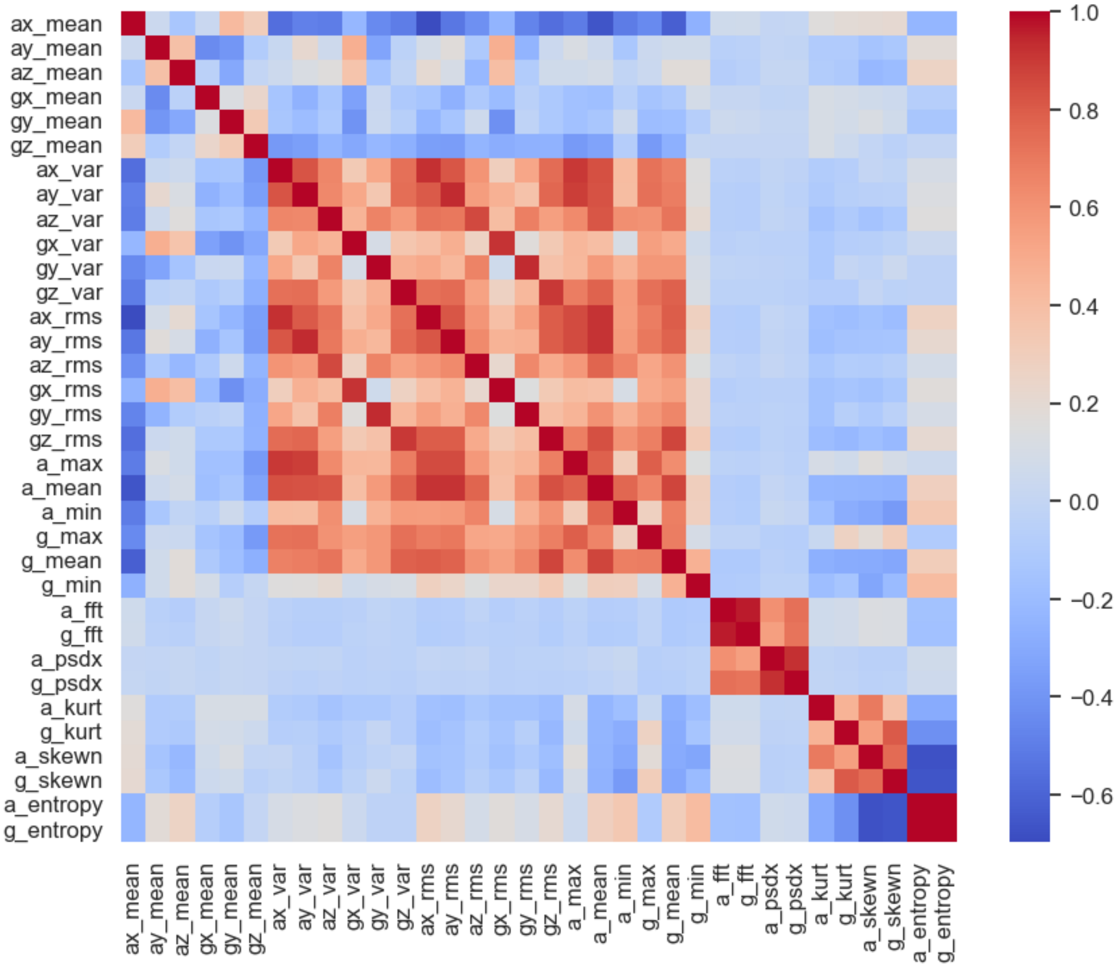


Figure 1: Correlation Matrix of the sensory parts of the dataset.

Why did you pick the specific chart?

Since the dataset does not contain a predefined target variable for talent, a correlation matrix was chosen to explore the relationships among the numerical features. This matrix is an important tool for identifying patterns, understanding variable interactions, and guiding feature selection for subsequent modelling steps.

What are the insights found from the chart?

Strong positive correlations:

- Variance, root mean square, max, mean, and min values from both accelerometer and gyroscope show strong positive correlations.
- Fourier transform and spectral density metrics from both sensors also are highly correlated.
- Kurtosis and skewness from the two sensors exhibit strong positive relationships.

Strong negative correlations:

- The x-axis mean from the accelerometer shows significant negative correlations with variance, RMS, max, mean, and min values from both sensors.
- Entropy and skewness from both sensors demonstrate negative correlations.

Weak correlation variables:

- y and z-axis mean from the two sensors have a weak correlation to the other variables.

Will the gained insights help creating a positive business impact?

These findings help create the binary target variable for talent and ensuring that redundant variables are handled appropriately. The strong correlations between sensor metrics (variance, RMS, etc) are used as key performance indicators for talent. This reduces reliance on subjective bias and the risk of costly mis-hires.

In addition, the weak correlation of y and z-axis mean from the accelerometer and gyroscope suggests that some variables are redundant. Therefore, the report will conduct a dimensionality reduction technique to optimise the computational costs and improve interpretability.

Variables

The sensory features used for this project:

- ax_mean: mean of x axis from accelerometer
- ay_mean: mean of y axis from accelerometer
- az_mean: mean of z axis from accelerometer
- gx_mean: mean of x axis from gyroscope
- gy_mean: mean of y axis from gyroscope
- gz_mean: mean of z axis from gyroscope
- ax_var: variance of x axis from accelerometer
- ay_var: variance of y axis from accelerometer
- az_var: variance of z axis from accelerometer
- gx_var: variance of x axis from gyroscope
- gy_var: variance of y axis from gyroscope
- gz_var: variance of z axis from gyroscope
- ax_rms: root mean square of x axis from accelerometer
- ay_rms: root mean square of y axis from accelerometer
- az_rms: root mean square of z axis from accelerometer
- gx_rms: root mean square of x axis from gyroscope
- gy_rms: root mean square of y axis from gyroscope
- gz_rms: root mean square of z axis from gyroscope
- a_max: maximum value from accelerometer
- a_mean: mean value from accelerometer
- a_min: minimum value from accelerometer
- g_max: maximum value from gyroscope
- g_mean: mean from gyroscope
- g_min: minimum value from gyroscope
- a_fft: fourier transform from accelerometer
- g_fft: fourier transform from gyroscope
- a_psdx: spectral density from accelerometer
- g_psdx: spectral density from gyroscope
- a_kurt: kurtosis from accelerometer
- g_kurt: kurtosis from gyroscope
- a_skewn: skewness from accelerometer
- g_skewn: skewness from gyroscope
- a_entropy: entropy values from accelerometer
- g_entropy: entropy values from gyroscope

Feature Engineering

To classify whether a player is talented, new features were engineered based on insights from extensive research of talent selection criteria. Two key attributes consistently emerged as indicators of talent: speed and control. Based on the features in the dataset, four features were created:

- Total accelerometer magnitude and total gyroscope magnitude: The acceleration and gyroscope magnitudes are likely to capture the speed of player movements. They are calculated using the vector magnitude of mean acceleration across all three axes (x, y, z).
- Motion consistency of accelerometer and gyroscope: the motion consistency metrics are likely to capture the control of the player's movements. They are derived by combining the variance and RMS to measure the player's movements consistency.

These features were normalised to a scale of 0 and 1 to ensure equal weighting when evaluating talent. A composite metric, performance score, was created by combining the normalised features with weights (25% each). The binary target variable for talent classification is created by setting the top 15% scores as 1 and the others as 0. To address the class imbalance, the dataset was resampled using SMOTE to generate additional synthetic samples to ensure 30% of the data represent talented swings. This was done to improve model performance without introducing overfitting, which can occur with excessive oversampling.

Analysis Plan

Dataset Assumptions:

- No systematic bias in how sensors were attached to different players.
- No significant fatigue effects or abnormal performance days included.
- The top 30% cutoff in performance score is assumed to reasonably reflect as a talented swing because this binary classification may oversimplify a talent spectrum.
- The weighting assigned to each component in the performance score assumed equal importance because it simplifies validation against actual performance outcomes.
- Talent is reflected purely on sensory data and not the categorical factors
- Talent is assumed to remain relatively stable over time because accounting for players who may improve dramatically with training would introduce complexity beyond the scope of this analysis.

Evaluation Metrics Scores and Charts

- Precision ($\frac{TP}{TP+FP}$): Proportion of classified talents that are actually talented.
- Recall ($\frac{TP}{TP+FN}$): Proportion of talents correctly identified.
- Accuracy ($\frac{TP+TN}{TP+TN+FP+FN}$): Overall correct prediction rate. The train and test dataset accuracy were calculated to evaluate the model's generalisability.
- ROC AUC Score: To show the model's ability to distinguish negative and positive classes across different threshold. As the primary metric, it is particularly useful in evaluating performance with class imbalance, which is essential when both false positives and false negatives carry high costs.
- Confusion Matrix: Show true negatives (non-talented swings), false positives (incorrect selection), false negatives (missed talents) and true positives (talented swings)
- Precision and Recall Curve: Shows the trade-off between precision and recall which is informative for this unbalanced dataset.
- Training and validation Comparison Chart: Used to monitor potential overfitting or underfitting by evaluating training and validation scores side by side.

Tuning Hyperparameters

All models were optimised using RandomisedSearchCV (random sampling of parameters combinations) with stratified k-fold cross-validation. This is chosen for its computational efficiency, effectiveness in high dimensional parameter spaces and great with unbalanced classes to maintain group proportions in each fold as the original dataset. The method uses 5-fold cross-validation ensures robust evaluation without the risk of overfitting.

Analysis Plan 1: Principal Component Analysis

Model: Principal Component Analysis (PCA) is an unsupervised dimensionality reduction technique that transforms the features into a new set of uncorrelated variables (principal components, PCs) ordered by how much variance they capture.

Purpose: The data has 30+ features. Based on the correlation matrix, many features are highly correlated while others are weakly corelated or redundant. Therefore, the use of the PCA helps reduce computational costs for resource-intensive models and helps visualise high dimensional data.

Method:

1. The dataset is standardised by subtracting the mean and dividing by the standard deviation for each feature, ensuring all features contribute equally.
2. A covariance matrix is created to understand the relationships between features.
3. Eigenvalues and eigenvectors are derived from the covariance matrix to identify directions of maximum variance.
4. The original data is projected onto the eigenvectors to form the principal components.

Key Parameters:

- Number of principal components: Number of components retained is determined based on how much cumulative variance they explain.

Assumptions:

- Assume PCs are linear combinations of the original features because nonlinear relationships may be lost in dimensionality.
- Assume directions with highest variance are most relevant for talent identification to be able to interpret the data.

Analysis/Visualisations:

- 2D PCA Plot: Visualises how data points are distributed along the first two principal components, highlighting clustering or separation.
- Explained Variance Plot: Shows the cumulative variance explained as more components are added. PCs are retained until 95% of total variance is captured.
- Scree Plot: Displays the individual variance explained by each component, helping to identify the point of diminishing returns.
- Feature Importance Analysis: Highlights the original features contributing most strongly to the top principal components

Analysis Plan 2: Random Forest Model

Model: A Random Forest Model (RFM) is an ensemble method that builds a collection of decision trees during training. Each tree contributes a prediction, and the final output is determined by majority voting across all trees.

Purpose: The RFM is well suited because it handles nonlinear relationships well and is robust to outliers/noise in the data. The model is also less prone to overfitting than single decision trees.

Method:

1. Use bootstrap aggregation to create replacements from randomly select multiple subsets of the training dataset.
2. For each split in the tree building process, only a random subset of features is used for splitting.
3. Grows each decision tree to the maximum depth, creating multiple decision trees.
4. Each tree makes its own prediction on the class (talented swing or not), the final prediction is determined by majority vote.

Key Parameters

- Number of Estimators: Number of trees in the forest, the higher values generally enhance the performance however increases computation costs and time.
- Min Sample Leaf: Minimum number of samples needed to be at a leaf node. Higher values generally enhance the performance however increases computation costs and time.
- Max Features: Maximum Number of features used at each split, can be either sqrt, log2 or none.
- Max Depth: Maximum tree depth for each decision tree. Controls overfitting.
- Min Samples Split: Minimum number of samples needed to split a node.
- Bootstrap: Enables bootstrap samples to be used when building trees, can be either true or false.
- Criterion: Metric used for split quality evaluation, can be either gini or entropy.

Assumptions:

- Features have some degree of independence because the RF to handle multicollinearity better than linear models when the features have some correlations.
- Assuming bootstrap samples adequately represent true distribution of the swings. This is important because of classifying talent across different genders, ages and experience.
- Assumes the dataset is reasonably balanced because RF performs better with balanced datasets.

Analysis Plan 3: Light Gradient Boosting Model

Model: The Light Gradient Boosting Model (LGBM) is a high-performance gradient boosting framework that builds decision trees using leaf-wise growth with advanced optimisations.

Purpose: This is a highly efficient model for large datasets compared to the traditional gradient boosting models. The model also handles imbalanced classes well and works effectively with nonlinear data. LGBM has built in regularisation that helps mitigate overfitting.

Method:

1. Unlike level-wise methods, LGBM grows the tree by splitting the leaf with the largest loss reduction, improving convergence and accuracy.
2. Uses gradient based one side sampling (GOSS) to retain data instances with large gradients and randomly samples those with smaller gradients to reduce computation.
3. Employs exclusive feature bundling (EFB) to combine mutually exclusive sparse features to reduce the dimensionality without losing information.
4. Each new tree focuses on fixing the residuals of the previous ensemble.
5. Final predictions are made by aggregating trees, where more accurate trees have greater influence.

Key Parameters:

- Number of estimators: Total number of boosting iterations. Higher values allow the model to capture more complex patterns however may result in overfitting.
- Max Depth: Maximum depth of individual trees. Controls model complexity and prevents overfitting.
- Learning Rate: Controls the step size during optimisation. Smaller values increase accuracy and generalisation but require more iterations to learn.
- Min Child Samples: Minimum number of data points required in a leaf.
- Regularisation Alpha: L1 regularisation term on leaf weights. Promotes sparsity in feature usage.
- Regularisation Lambda: L2 regularisation term on leaf weights. Promotes smoother models.

Assumptions:

- Features have some degree of independence because LBGM is robust to sensor features with some correlations, however, extreme multicollinearity can affect feature importance.
- Assumes relationships between features and target labels are generally monotonic because the features used to calculate talent is correlated.
- Assumes that the SMOTE balanced dataset maintains the true distribution and does not introduce significant noise or artifacts.

Analysis Plan 4: Deep Neural Network

Model: A Deep Neural Network (DNN) is a type of neural network that consists of multiple hidden layers that apply successive nonlinear transformations to input data.

Purpose: DNN is selected for its ability to capture intricate patterns and automatically learns feature representations. The model also minimises overfitting and generalises well with high dimensional data and nonlinear relationships.

How it works?

1. Creates input layer to receive pre-processed feature vectors as input.
2. Creates three dense hidden layers with decreasing neurons (64 →32→16). Each layer applies to linear transformation (weights × inputs + bias), nonlinear activation and regular dropouts.
3. Creates output layer with single neuron of sigmoid activation to produces probability between 0-1 for binary classification.
4. Trained by repeating the following steps for the number of epochs:
    - Forward propagation where data flowing through the network to produce prediction.
    - Lose calculations where binary cross entropy loss is used to compare the predicted output with the true label.
    - Backpropagation where gradients are calculated and weights are updated via the Adam optimiser to minimise loss.

Key Parameters

- Optimiser: Adaptive Moment Estimation (Adam) for efficient gradient descent.
- Dropout Rate: Portions the input units into randomly set to 0 during training to prevent overfitting by breaking up co-adaptions.
- Hidden Layer Sizes: [64, 32, 16] to progressively reduce dimensionality.
- Activation: nonlinear function such as ReLU for hidden layers and Sigmoid for output layers.
- Learning rate: Controls the step size in weight updates.
- Batch Size: Number of training samples per update step; impacts training speed and convergence.
- Epochs: Total number of complete passes through the training dataset.

Assumptions:

- Neurons are organized in layers where each layer is fully connected to the next, and information flows only forward.
- All data enters through the input layer and all predictions exit via the output layer.
- Each neuron has an associated weight and bias, and they are adjusted during training to optimise performance.
- All neurons in hidden layers use the same activation function (ReLU), and a sigmoid function is used for binary output.
- Assumes the model can learn relevant patterns from training data and apply them effectively to unseen data.

Model Implementation

Model 1: Principal Component Analysis

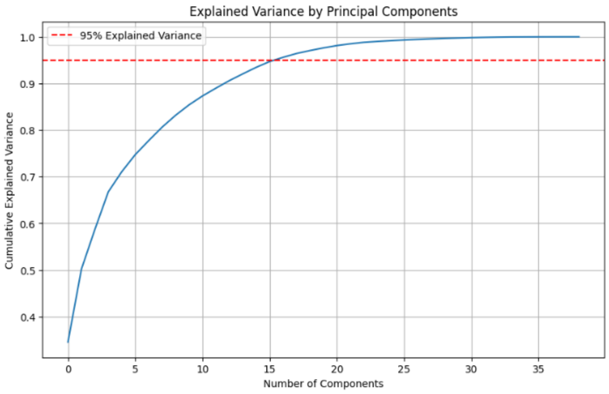


Figure 2: Explained Variance Plot by Principal Components

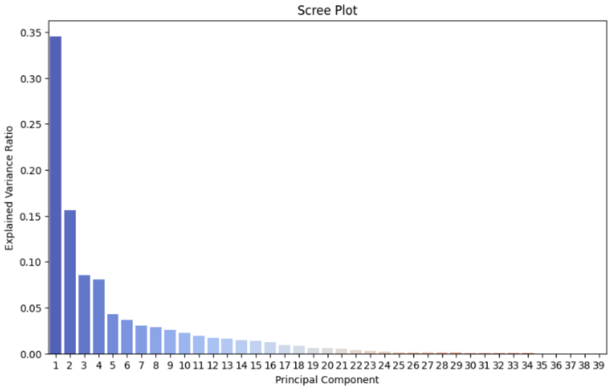


Figure 3: Scree Plot

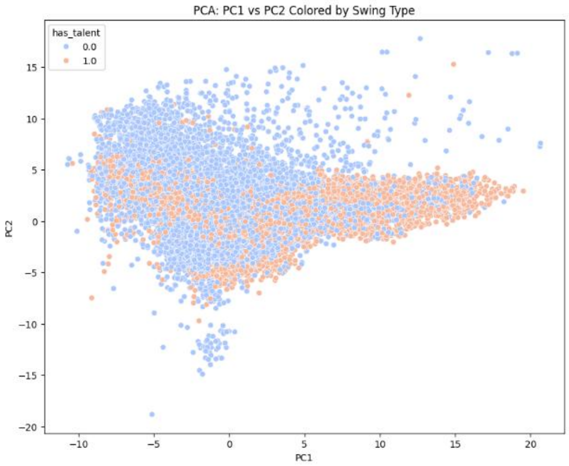


Figure 4: PC1 vs PC2 Comparison

The number of principal components retained is determined by capturing 95% of the cumulative explained variance. This approach balances dimensionality reduction by preserving the most informative features while discarding less relevant ones. Figure 2 shows 16 components is ideal as supported with Figure 3, which illustrates that beyond 16 components contribute minimal variance and are largely redundant. Figure 4, demonstrates the binary target variable is well distributed across the two components, indicating the reduction process maintains the integrity of class distinctions and appropriate for modelling.

Top 13 Features from 16 PCs

1. Overall mean from accelerometer
2. Entropy from gyroscope
3. Spectral Density from gyroscope
4. Kurtosis from accelerometer
5. Root mean square of y axis from gyroscope
6. Mean of z axis from gyroscope
7. Mean of x axis from gyroscope
8. Kurtosis from gyroscope
9. Motion consistency from accelerometer
10. Spectral Density from accelerometer
11. Mean of z axis from accelerometer
12. Min of z axis from gyroscope
13. Mean of y axis from accelerometer

Model 2: Random Forest Model

General Model

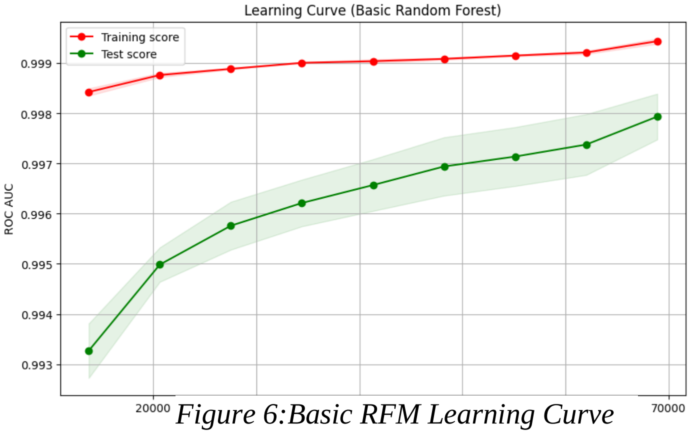


Figure 6:Basic RFM Learning Curve

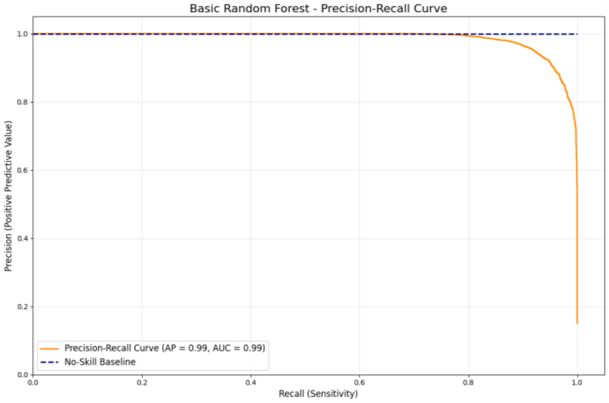


Figure 5: Basic RRM Precision-Recall Curve

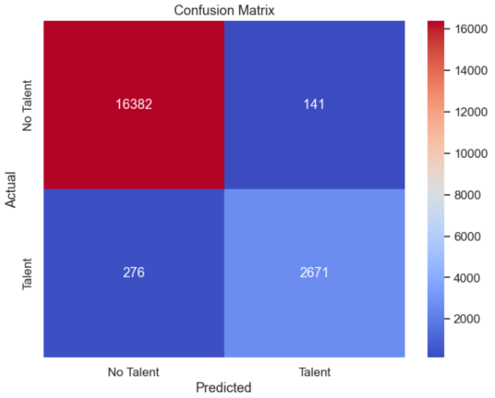


Figure 7: Basic RFM Confusion Matrix

Table 1: Basic RFM Metrics	
Test Accuracy	97.86%
Train Accuracy	98.78%
Precision	96.66%
Recall	94.89%
F1 Score	95.75%
ROC AUC Score	99.65%

Model Used and its Performance

The basic RFM was configured with 100 estimators and min 1 leaf sample, which produced high performance across all evaluation metrics. This indicates strong predictive performance for identifying talent, supported by Figure 5-6 and Table 1. With 97.86% test accuracy and 98.78% train accuracy indicates good generalisation, minimal overfitting occurring and maintained high overall correctness. The 96.66% precision means the model correctly identifies talented players in nearly all positive predictions, while 94.89% recall indicates it successfully captures almost all actual talented players. These findings are supported with Figure 6. The F1 score of 95.75% reflects a balanced and robust trade-off between precision and recall. Figure 5 and the ROC AUC score of 99.65% reveals the RFM's near perfect ability to distinguish between talented and non-talented swings. As shown in Figure 7, the high true negative and true positive further confirms the model's effectiveness in accurate classifications. These results suggest the basic RFM has effectively learned meaningful patterns to identify player talent with high reliability, making it suitable for player selection decisions.

Optimised Model

The key range for each parameter includes 50-500 estimators, max depth of 3-20, min samples split of 2-20, min sample leaf 1-20, the max features, bootstrap and criterion. The optimised model outperforms the basic version, as evidenced by Figure 8-10 and Table 2.

The best parameters found:

- bootstrap: False
- criterion': 'gini'
- max depth: 19
- max features: 'log2'
- min samples leaf: 1
- min samples split: 17
- number of estimators: 303

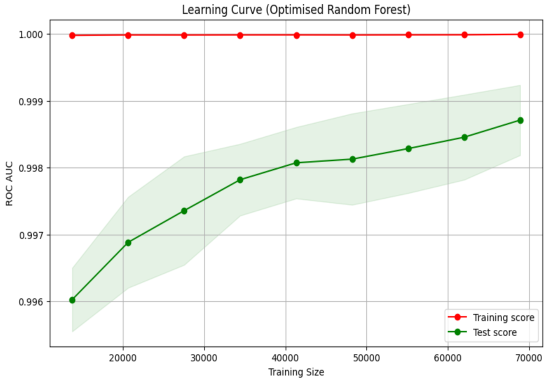


Figure 9: Optimised RFM Learning Curve

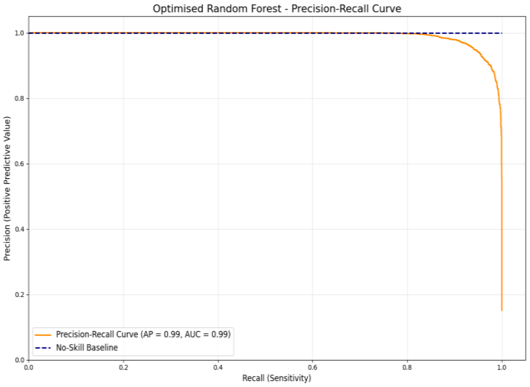


Figure 8: Optimised RFM Precision-Recall Curve

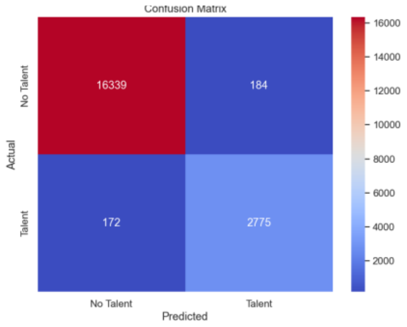


Figure 10: Optimised RFM Confusion Matrix

Table 2: Optimised RFM Metrics

Test Accuracy	98.17%
Train Accuracy	99.89%
Precision	96.37%
Recall	96.52%
F1 Score	96.45%
ROC AUC Score	99.77%

Performance and the Business Impact of the Model Used

The optimised RFM delivers greater performance than the basic RFM. The 98.17% test accuracy indicates slightly higher accuracy than basic RFM, which reduces costly selection errors. The generalisation is slightly greater than base model which suggests minimal overfitting occurred. Precision increased to 96.37%, with only 3.63% of players predicted as talented are false positives, reducing resource on underqualified athletes. Recall rose to 96.52%, ensuring only 3.48% of actual talented players are missed, which is critical for maintaining a competitive advantage. The F1 score (96.45%) confirms balanced and consistent performance, while 99.77% ROC AUC establishes the model's great ability to discriminate talent in imbalanced data.

Overall, the optimised RFM contributing high reliability in data driven decisions, which offers significant business value of creating a competitive advantage, reducing subjective bias in selections, maximising team potential and minimising budget costs.

Model 3: Light Boosting Model

General Model

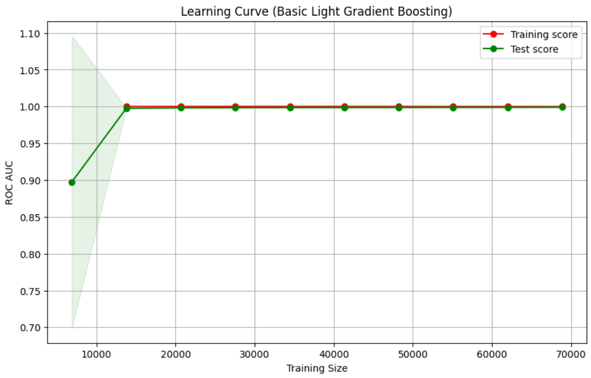


Figure 12: Basic LGBM Leaning Curve

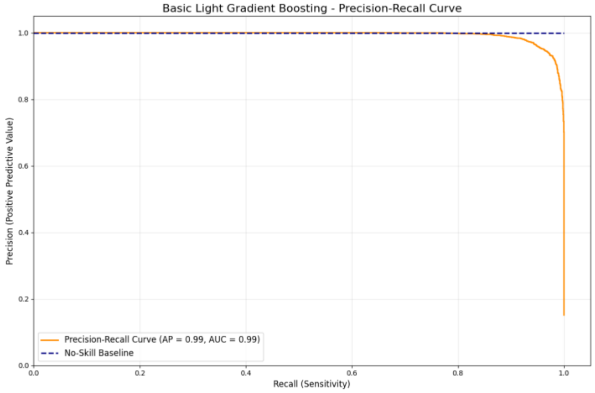


Figure 11: Basic LGBM Precision-Recall Curve

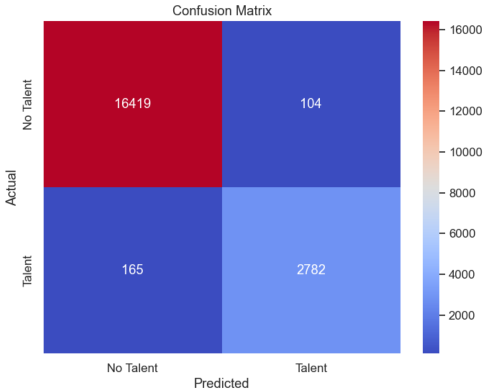


Figure 13: Basic LGBM Correlation Matrix

Table 3: Base LGBM

Test Accuracy	98.28%
Train Accuracy	99.42%
Precision	97.70%
Recall	96.89%
F1 Score	97.29%
ROC AUC Score	99.85%

Model Used and its Performance

The basic LGBM was constructed with 100 estimators and a learning rate of 0.1. It outperforms the basic RFM, as supported by the Figure 11-13 and Table 3. With a test accuracy of 98.28% and train accuracy of 99.42%, the model shows strong predictive capabilities while maintaining higher generalisation. The near perfect ROC AUC score of 99.85% and Figure 11 reveals exceptional discrimination ability, inferring the model can almost flawlessly separate talented from non-talented swings. Figure 13 confirms the LBGM accurately identifies most cases of talented and non-talented cases. The low false positive (misclassified non-talent) and false negative (missed talent) suggest high performance. This balance is confirmed by the F1 score of 97.29% and Figure 12, which harmonises the precision (97.70%) and recall (96.89%) metrics. The precision score indicates that when the model predicts talent correctly 97.70% of the time, while the recall shows the model captures 96.89% of the true talents. These metrics collectively demonstrate that the basic LGBM has learned meaningful patterns to reliably identify talented players with very few misclassifications.

Optimised Model

The key parameters ranges include number of estimators from 50 to 1000, number of leaves 20 to 150, learning rate from 0.01 to 0.3, min child samples from 5 to 100, max depth from 3 to15, reg alpha o to1, reg lambda 0 to 1, subsample from 0.6 to 0.4, colsample by tree got 0.5 and is unbalanced. The optimised model outperforms the basic version as demonstrated by the Figure 14-16 and Table 4.

The best parameters found:

- Colsample bytree: 0.6644885149016018
- Is unbalance: False
- Learning rate: 0.197306214440138
- max depth: 11
- min child samples: 10
- number of estimators: 845
- number of leaves: 47
- reg alpha: 0.2184404372168336
- reg_lambda: 0.4165099478703662
- subsample: 0.9533121035675474

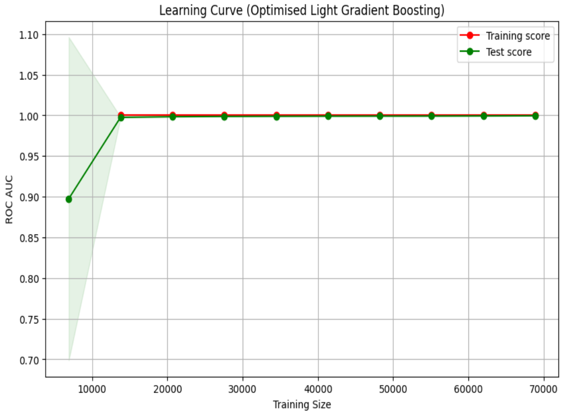


Figure 14: Optimised LGBM Learning Curve

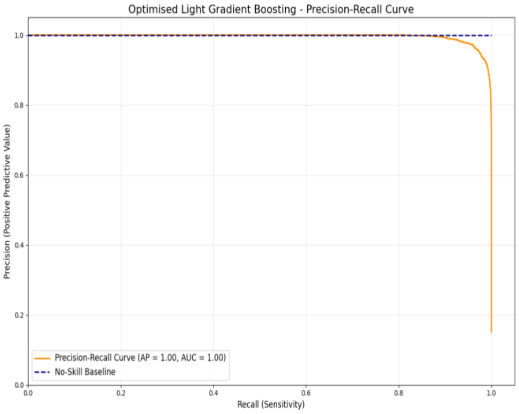


Figure 15: Optimised LGBM Precision-Recall Curve

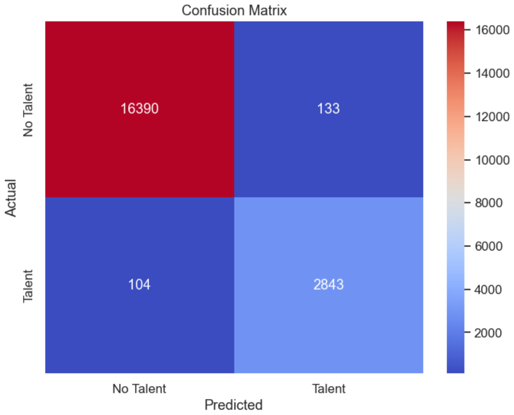


Figure 16: Optimised LGBM Correlation Matrix

Table 4: Optimised LGBM Metrics

Test Accuracy	98.78%
Train Accuracy	100.0%
Precision	97.45%
Recall	97.83%
F1 Score	97.64%
ROC AUC Score	99.89%

Performance and the Business Impact of the Model Used

The optimised LGBM model surpasses the optimised RFM in performance and computational runtime. The 98.78% test accuracy indicates the model correctly classifies nearly all swings, optimising selection efficiency and reducing costs. The 97.45% precision implies when the model identifies a player as talented, and a 2.55% of false positives, which critical for avoiding wasted resources on unqualified players. While recall guarantees the organisation would not overlook 97.83% potential athletes. The 97.64% F1 score confirms reliable balance between these precision/recall, while the 99.89% ROC AUC underscores outstanding discrimination power. Figure 16 reveals 104 false negatives (missed talents) and 133 false positives (incorrect selections), indicating minimal classification errors that translate directly to cost savings and effective talent identification.

These results provide business value by reducing subjective bias in decisions, optimising scouting efficiency, lowering expenses, and delivering high reliability in data driven decision making that provide a competitive advantage in talent acquisition.

Model 5: Deep Neural Network

General Model

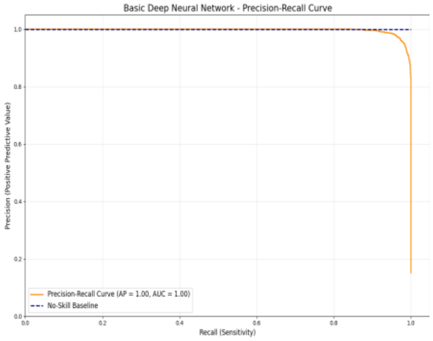


Figure 17: Basic DNN Precision-Recall Curve

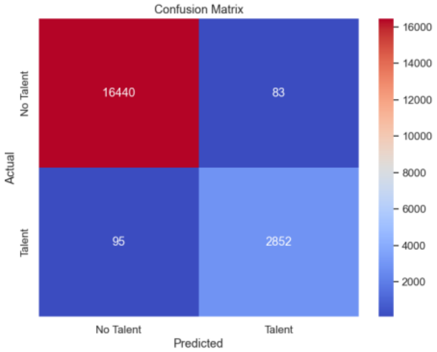


Figure 18: Basic DNN Correlation Matrix

Table 5: Basic DNN Metrics	
Test Accuracy	99.09%
Train Accuracy	99.35%
Precision	98.73%
Recall	98.14%
F1 score	98.22%
ROC AUC Score	99.94%

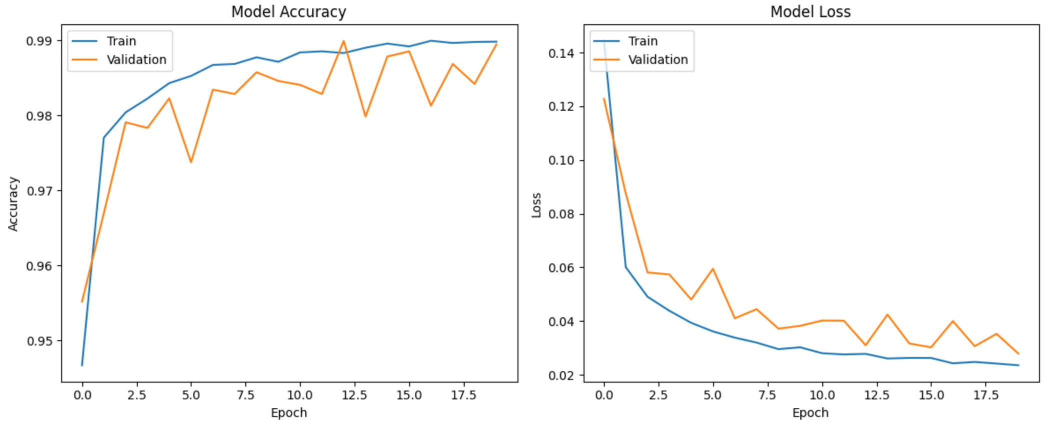


Figure 19: Basic DNN Training and Validation Comparison

Model Used and its Performance

The basic DNN was trained for 20 epochs and batch size 200 and demonstrates exceptional performance when compared to the other models, as shown in Figure 17-19 and Table 5. With a 99.09% test accuracy and 99.35% train accuracy, the model shows minimal overfitting and achieving almost perfect classification as visualised in Figure 19. This indicates proper regularisation and suggests the model will generalise well to new data. The precision indicates that when the model predicts a player as talented and it is correct 98.73% of the time. The strong recall infers the model successfully identifies 98.14% of all truly talented players. Figure 17 and the F1 Score of 98.22% further supports this as demonstrated with the model's robust performance on imbalanced data. Most notably, the 99.94% ROC AUC reveals the model has near flawless ability to distinguish between talented and non-talented swings. The model’s high accuracy is confirmed Figure 18 where 16,440 true negative with low 83 false positives from non-talented swings, and with low 95 false negatives out of 2,947 talented swings. These results suggest the basic DNN has effectively learned complex patterns in the data to produce highly accurate talent predictions, with particularly strong performance in discerning talent while maintaining good coverage of actual talent.

Optimised Model

The RandomisedSearchCV evaluated random parameter combinations using 3-fold CV to operate efficiently for DNN. Key hyperparameters tuned included the activation function, 16-64 batch size, 0.1-0.5 dropout rate, 50-150 epochs, and optimiser type (Adam, RMSprop, SGD). The selected parameters suggest the optimised model produces the same performance as the basic version.

The best parameters found:

- activation: elu
- batch size: 16
- dropout rate: 0.19171739493308l9
- epochs: 50
- optimiser: adam

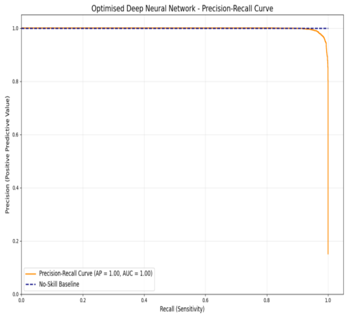


Figure 20: Optimised DNN Precision-Recall Curve

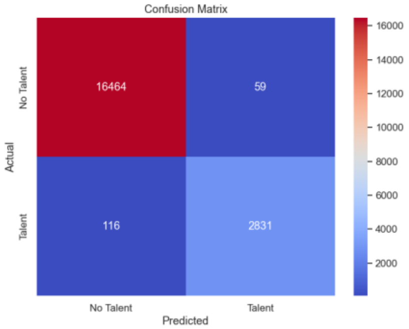


Figure 21: Optimal DNN Correlation Matrix

Table 6: Optimised DNN Metrixs	
Test Accuracy	99.10%
Train Accuracy	99.48%
Precision	98.63%
Recall	97.85%
F1 Score	98.24%
ROC AUC Score	99.94%

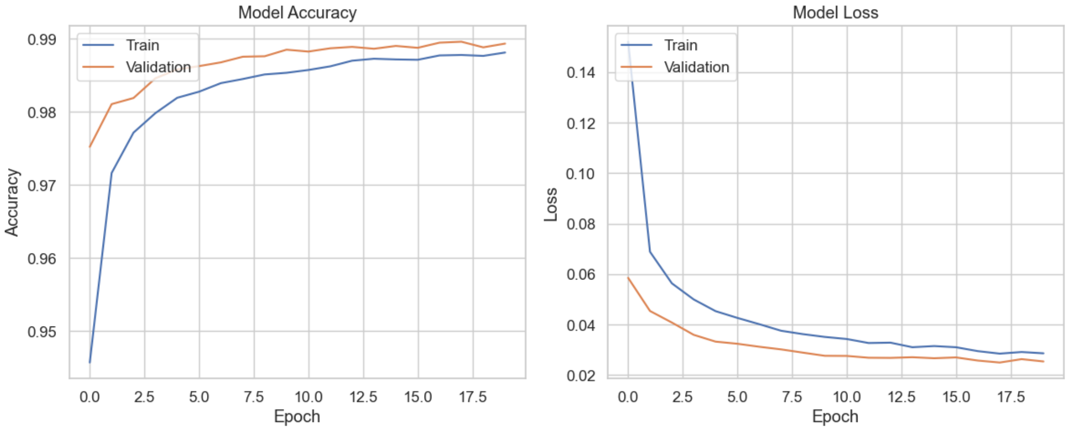


Figure 22:Optimised DNN Training and Validation Comparison

Performance and the Business Impact of the Model Used

The optimised DNN greatly outperforms the optimised LGBM and RFM in exchange with higher computation costs. The test accuracy, train accuracy and Figure 22 indicate near perfect prediction capability, excellent generalisation and minimised overfitting. This provides high confidence in the model's predictions. The 98.63% precision implies only 1.37% of players recommended are false positives, reducing wasted resources on unqualified athletes. The 97.85% recall shows the model misses 2.15% of true talents, a small margin that could impact competitive advantage if elite players are overlooked. The 98.24% F1 score, and Figure 20 confirms a well-balanced performance between precision and recall, essential for a reliable and valid selection process. The 99.94% ROC AUC supports the model's exceptional performance on unbalanced data. Figure 21 suggests the optimised DNN is slightly conservative in selections where it minimises false positive (thereby reducing costs) while accepting marginally increased missed talents. The hyperparameter choices make the model robust to the unbalanced data, ensuring consistent performance across different training environments.

By automating talent identification with such high reliability and validity, the optimsed DNN delivers business value by providing a competitive advantage, reducing subjective bias, optimising scouting resources and costs.

Limitations

- Despite high performance metrics such as the ROC AUC and F1 scores, the dataset remained imbalanced, which may have introduced a bias toward the majority class (non-talent swings) and affected the models’ ability to generalise to minority cases.
- Complex models like LGBM and DNN act as black boxes, making individual predictions difficult to interpret. This lack of transparency may hinder stakeholder trust and limit practical deployment where explainability is essential.
- DNN, while the most accurate model, required significant computational resources and training time, which may not be feasible for deployment in real-time or resource-constrained environments.
- Hyperparameter tuning was performed using RandomisedSearchCV with a limited number of iterations and constrained parameter ranges, possibly preventing discovery of more optimal configurations.
- The models were trained solely on features derived from sensor data. Any noise, measurement errors, or lack of contextual factors (i.e. psychological or environmental influences) could impact the prediction reliability and limited real-world applicability.

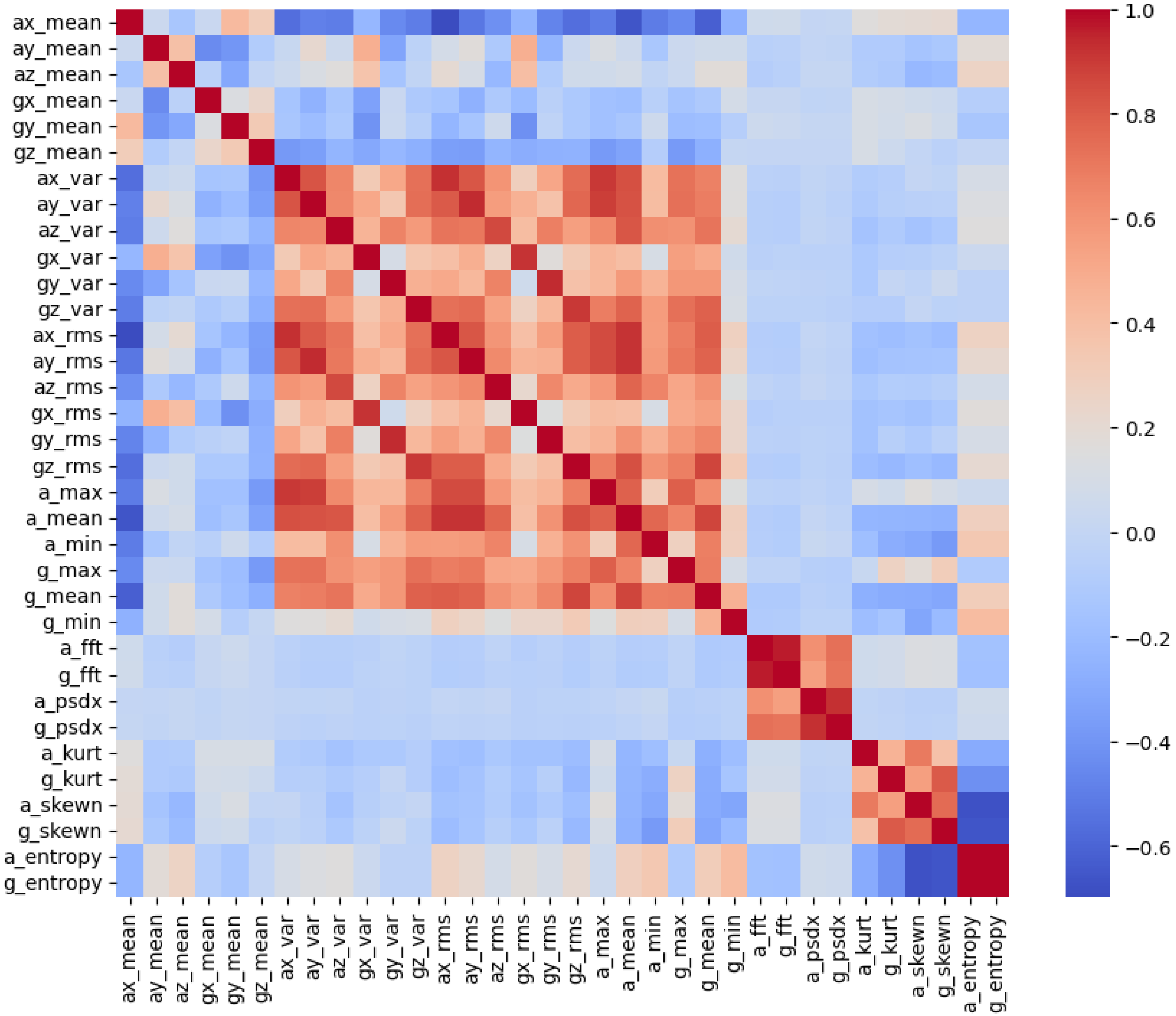
Future Work

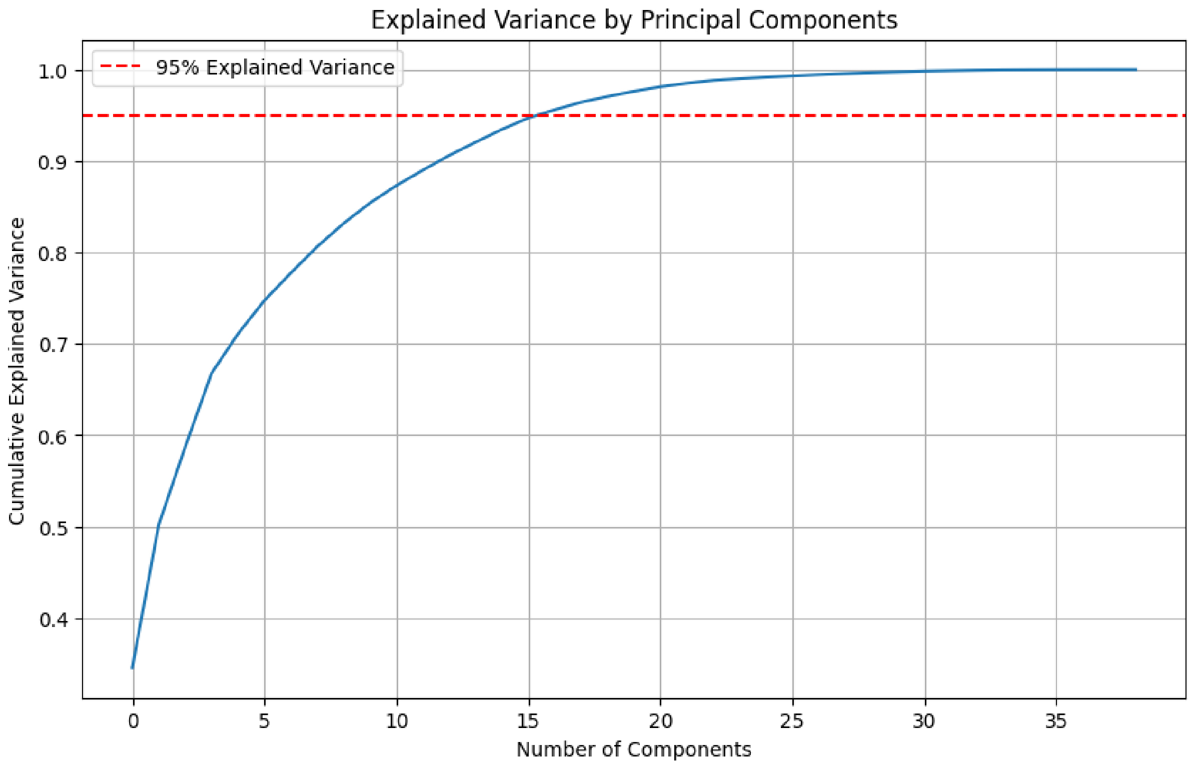
- Developing pipelines for real time data ingestion, model deployment, and monitoring would enable continuous adaptation and timely decision making in automating talent identification.
- Implementing explainability methods such as SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) could improve model transparency, allowing stakeholders to better understand and trust the rationale behind predictions.
- Combining strengths of multiple models through ensemble or hybrid architectures may further enhance the predictive performance and migrate the weaknesses of individual model.
- Replacing RandomisedSearchCV with more efficient tuning techniques such as Bayesian Optimisation could yield better performance with reduced computational cost.
- Expanding the dataset to include more diverse players, longer timeframes, and additional contextual data (physiological metrics, psychological assessments, and in-game scenarios) could improve model's robustness and predictive power.

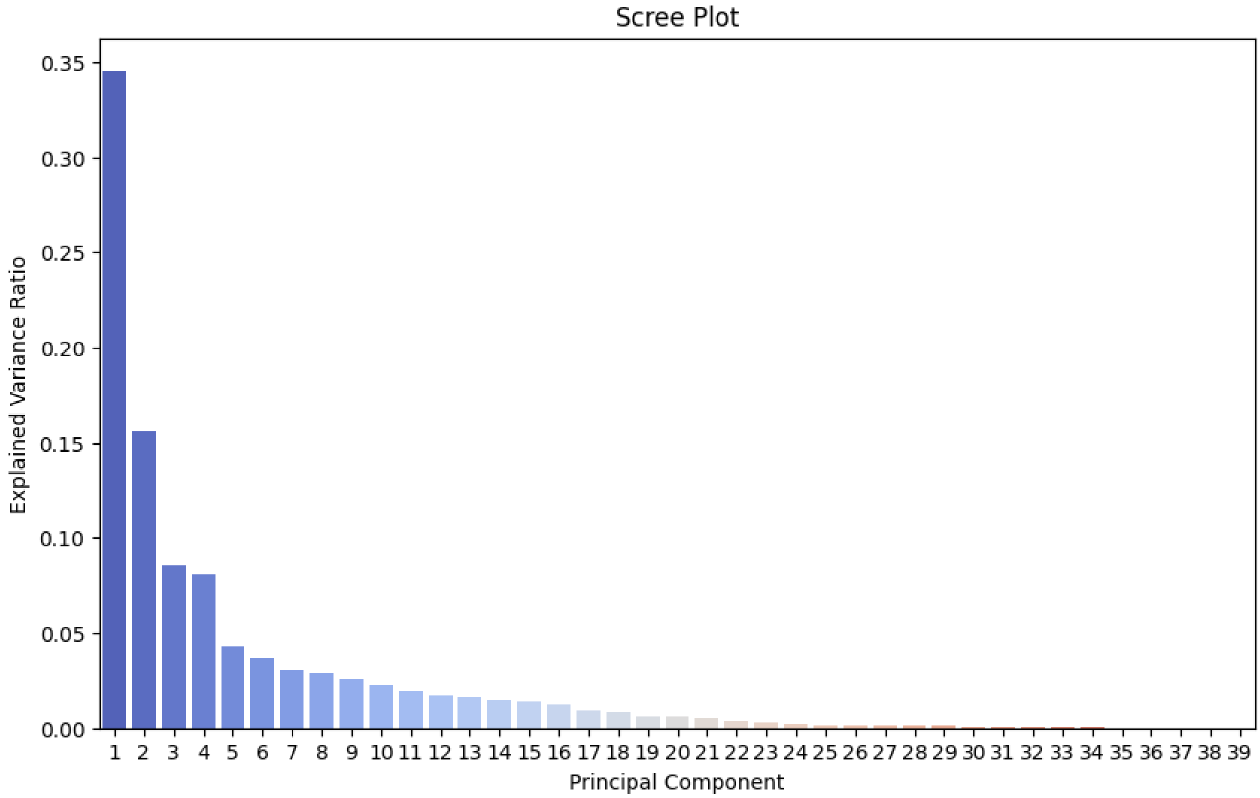
Conclusion

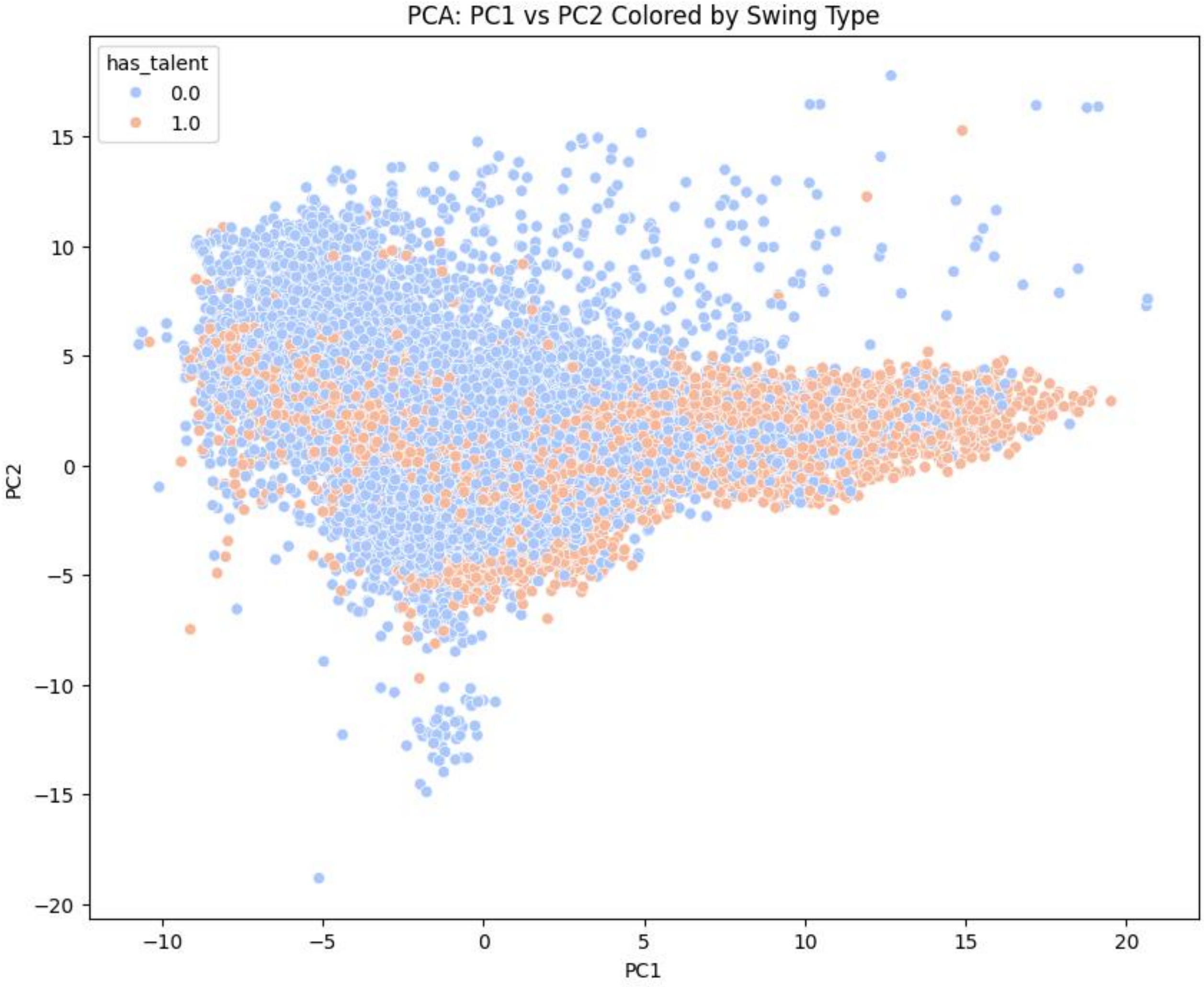
This report demonstrated the effective application of multiple machine learning models to identify talented swings. Among the models tested, the optimised LGBM emerged as the most practical for deployment, balancing strong predictive performance with efficient computation cost. While DNN achieved the highest overall accuracies, its computational demands may limit its suitability in real time or resource constrained scenarios. The RFM offered greater interpretability, but comparatively lower predictive power. Despite their individual trade-offs, all models demonstrated a strong ability to identify talent with minimal misclassification. These findings highlight the potential of machine learning to revolutionise talent identification through enabling data driven decisions, reducing bias, and enhancing scouting efficiency. By integrating sensor data with sophisticated algorithms, organisations can gain a competitive edge through more accurate and cost-effective recruitment. Overall, this report highlights the transformative potential of machine learning to modernise talent scouting and elevate data driven decision making in professional table tennis.

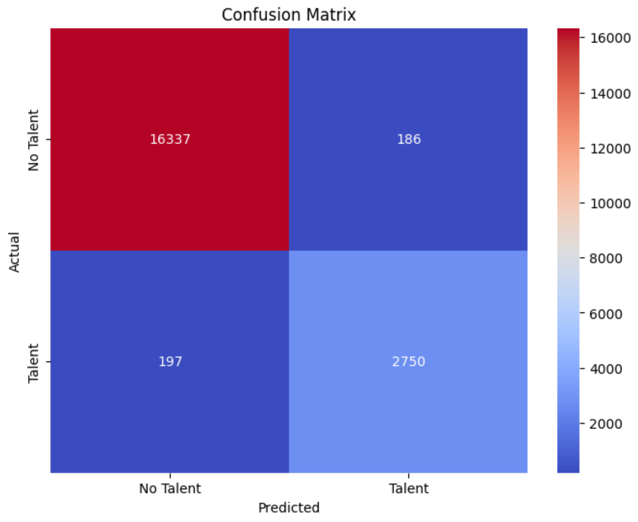
Appendix

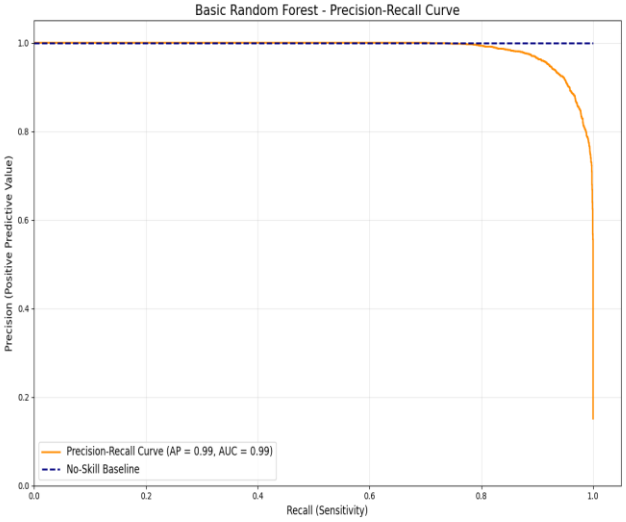

ax_mean
ay_mean
az_mean
gx_mean
gy_mean
gz_mean
ax_var
ay_var
az_var
gx_var
gy_var
gz_var
ax_rms
ay_rms
az_rms
gx_rms
gy_rms
gz_rms
a_max
a_mean
a_min
g_max
g_mean
g_min
a_fft
g_fft
a_psdx
g_psdx
a_kurt
g_kurt
a_skewn
g_skewn
a_entropy
g_entropy
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6

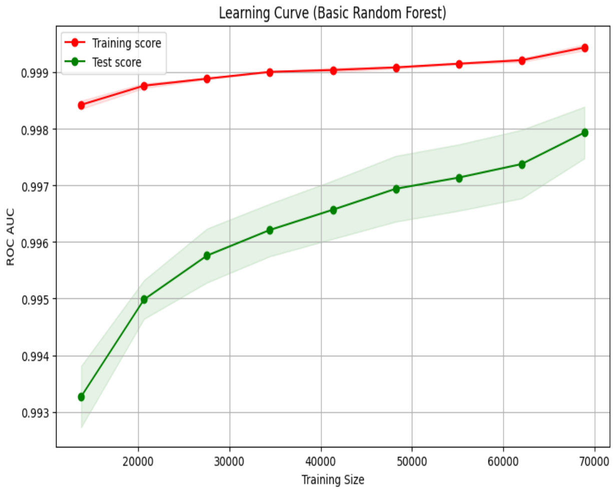

Explained Variance by Principal Components
95% Explained Variance
Cumulative Explained Variance
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0
5
10
15
20
25
30
35
Number of Components

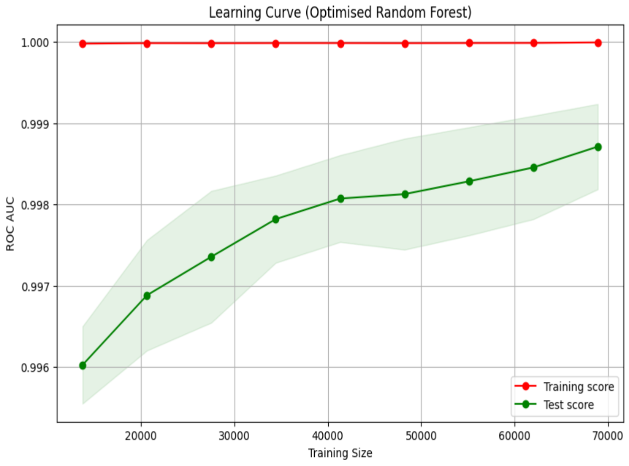

Scree Plot
Explained Variance Ratio
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39
Principal Component

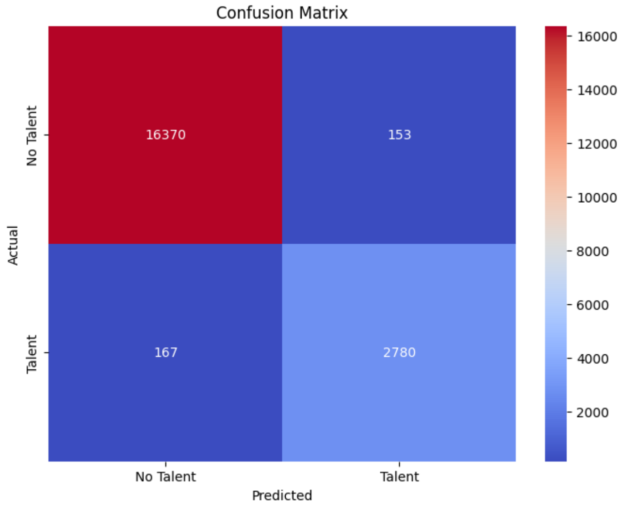

PCA: PC1 vs PC2 Colored by Swing Type
has_talent
0.0
1.0
15
10
5
0
−5
−10
−15
−20
PC2
−10
−5
0
5
10
15
20
PC1

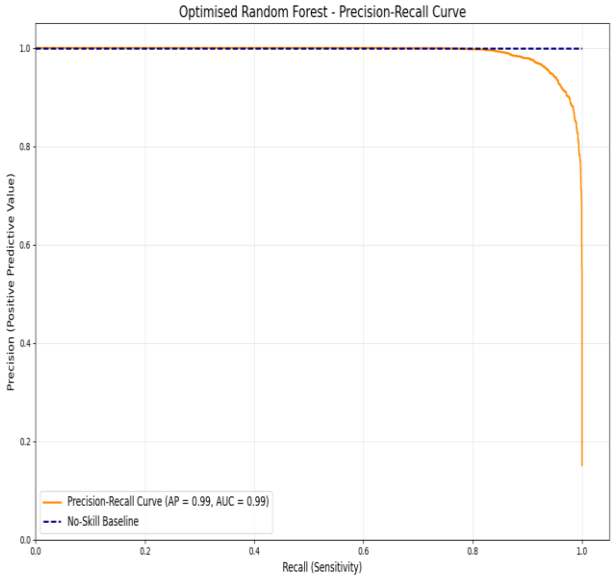

Confusion Matrix
No Talent
Talent
Actual
16337
186
197
2750
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000

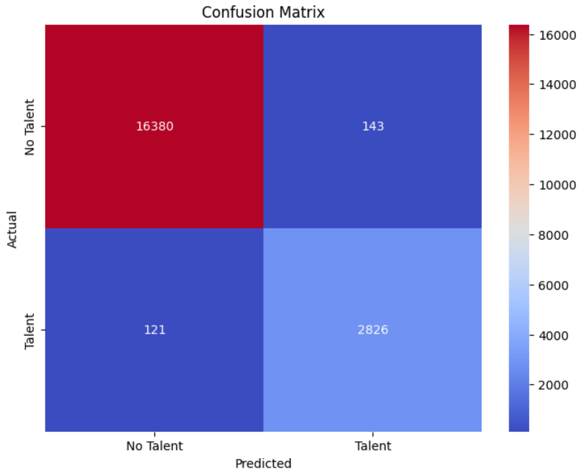

Basic Random Forest - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)
Precision-Recall Curve (AP = 0.99, AUC = 0.99)
No-Skill Baseline

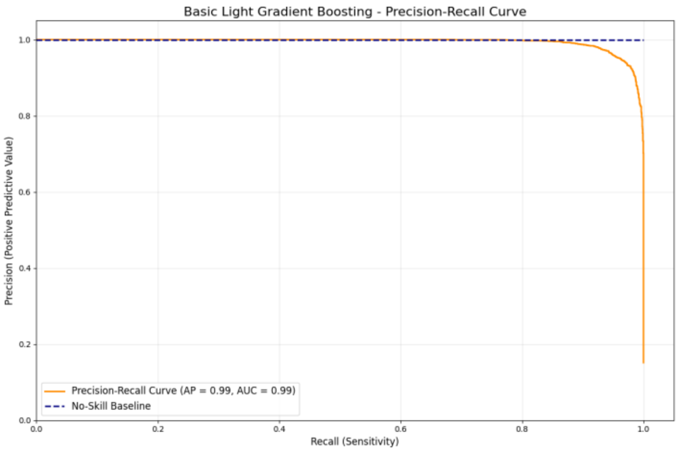

Learning Curve (Basic Random Forest)
Training score
Test score
ROC AUC
0.999
0.998
0.997
0.996
0.995
0.994
0.993
20000
30000
40000
50000
60000
70000
Training Size

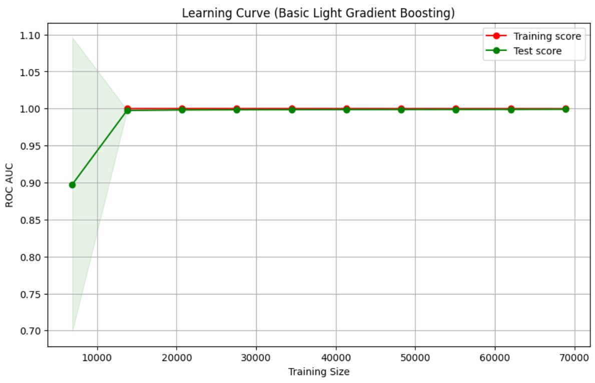

Learning Curve (Optimised Random Forest)
1.000
0.999
0.998
0.997
0.996
ROC AUC
Training score
Test score
20000
30000
40000
50000
60000
70000
Training Size

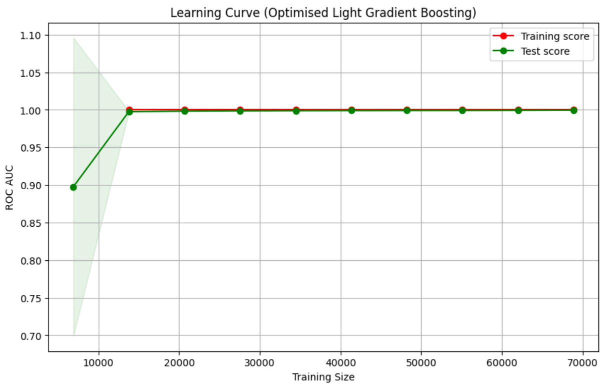

Confusion Matrix
No Talent
Talent
Actual
16370
153
167
2780
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000

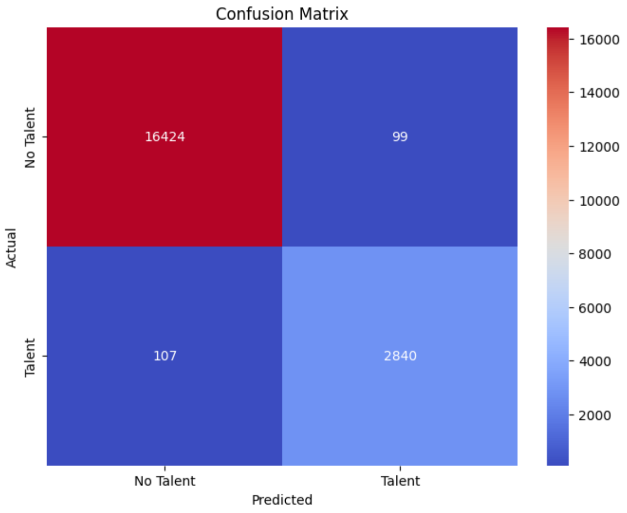

Optimised Random Forest - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
Precision-Recall Curve (AP = 0.99, AUC = 0.99)
No-Skill Baseline
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)

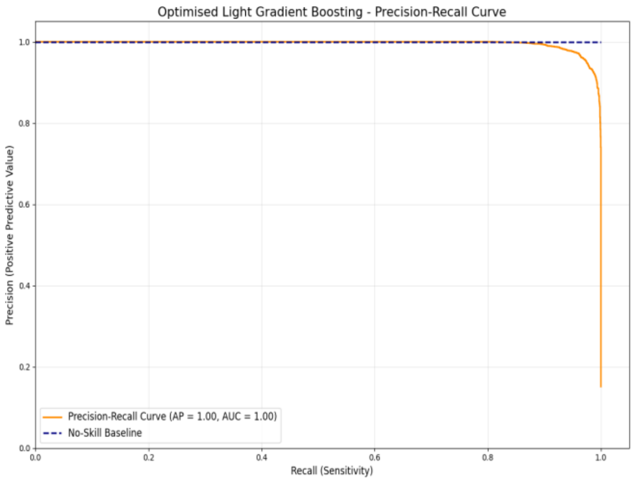

Confusion Matrix
No Talent
Talent
Actual
16380
143
121
2826
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000

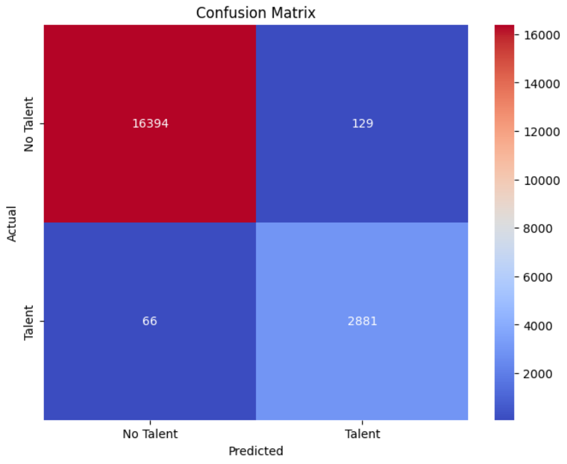

Basic Light Gradient Boosting - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
Precision-Recall Curve (AP = 0.99, AUC = 0.99)
No-Skill Baseline
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)

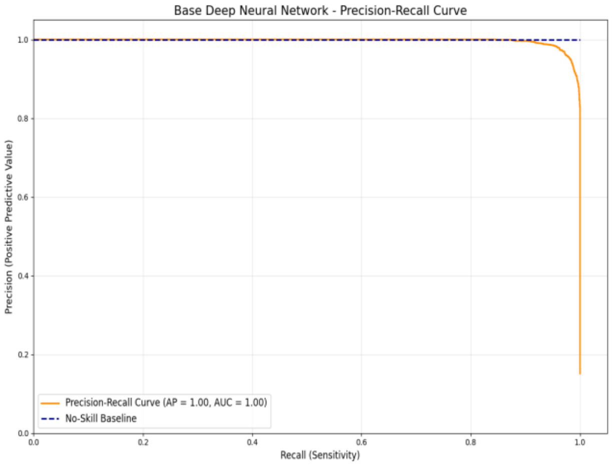

Learning Curve (Basic Light Gradient Boosting)
Training score
Test score
ROC AUC
1.10
1.05
1.00
0.95
0.90
0.85
0.80
0.75
0.70
10000
20000
30000
40000
50000
60000
70000
Training Size

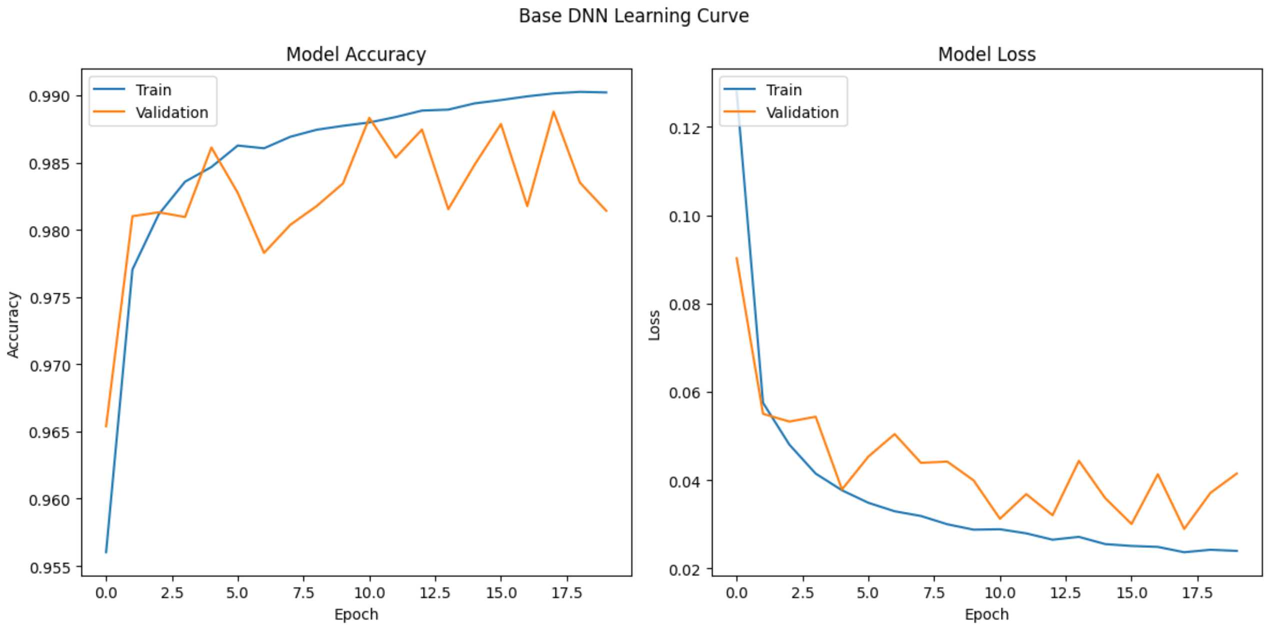

Learning Curve (Optimised Light Gradient Boosting)
Training score
Test score
ROC AUC
1.10
1.05
1.00
0.95
0.90
0.85
0.80
0.75
0.70
10000
20000
30000
40000
50000
60000
70000
Training Size

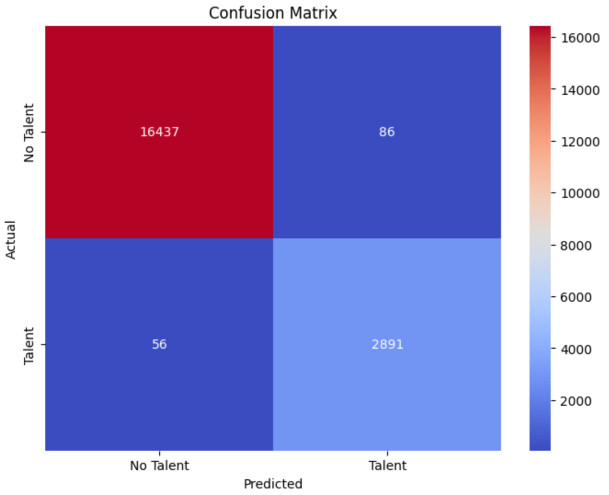

Confusion Matrix
Actual
No Talent
Talent
16424
99
107
2840
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000

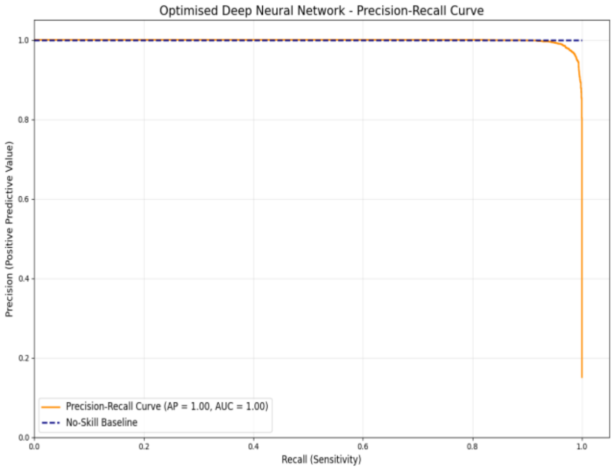

Optimised Light Gradient Boosting - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)
Precision-Recall Curve (AP = 1.00, AUC = 1.00)
No-Skill Baseline

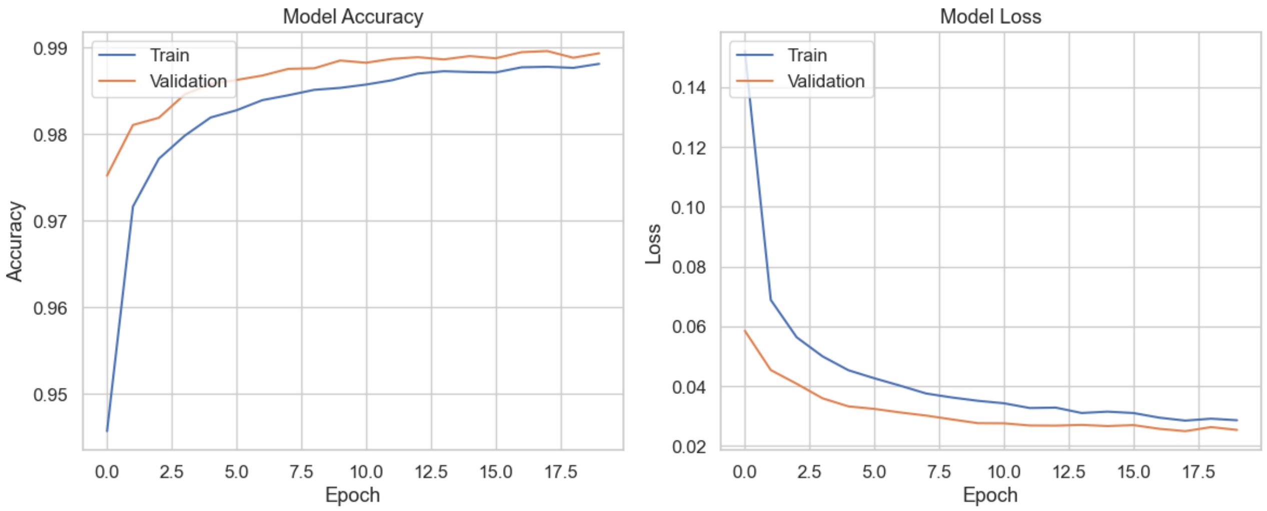

Confusion Matrix
Actual
No Talent
Talent
16394
129
66
2881
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000


Base Deep Neural Network - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)
Precision-Recall Curve (AP = 1.00, AUC = 1.00)
No-Skill Baseline


Base DNN Learning Curve
Model Accuracy
Train
Validation
Accuracy
0.990
0.985
0.980
0.975
0.970
0.965
0.960
0.955
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch
Model Loss
Train
Validation
Loss
0.12
0.10
0.08
0.06
0.04
0.02
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch


Confusion Matrix
Actual
No Talent
Talent
16437
86
56
2891
No Talent
Talent
Predicted
16000
14000
12000
10000
8000
6000
4000
2000


Optimised Deep Neural Network - Precision-Recall Curve
Precision (Positive Predictive Value)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Sensitivity)
Precision-Recall Curve (AP = 1.00, AUC = 1.00)
No-Skill Baseline


Model Accuracy
Train
Validation
Accuracy
0.99
0.98
0.97
0.96
0.95
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch
Model Loss
Train
Validation
Loss
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch