Predicting Maximum Acceleration During Table Tennis Swings using Regression Techniques

Anton Risch

May 30, 2025

Abstract

This report presents a regression analysis to predict maximum acceleration (`a_max`) during table tennis swings using the TTSWING dataset (Chou et al. 2023). It focuses on three distinct regression techniques to predict `a_max`: Ordinary Least Squares (OLS), Ridge Regression, and a feedforward multilayer perceptron Neural Network (NN). Data preprocessing included exploratory data analysis, Principal Component Analysis (33 components explaining 90.1% variance), and 70%/15%/15% train/validation/test splitting. The NN achieved superior test performance using a symmetric ($160 \rightarrow 160 \rightarrow 1$) architecture with target scaling ($R^2 = 0.9906$, MSE = 541,457, MAE = 567.8). Statistical significance testing confirmed that the NN significantly outperformed linear models ($p < 0.0001$).

1 Introduction

Table tennis is a highly intensive sport requiring precise coordination in returning the ball. This includes controlling the amount of power and speed generated by a player during each individual swing. A swing's maximum acceleration is a **critical biomechanical parameter** that correlates with swing power, ball velocity, and the overall effectiveness of a player. Being able to have a faster maximum acceleration means that a player can return the ball at a quicker reaction time and with more power.

A modified version of the original TTSWING dataset provided by the COMP4702 Academic Team was utilised for this report. It comprises 97,355 swing measurements, each with comprehensive sensor-derived features extracted from 9-axis accelerometer/gyroscope waveforms. These features include statistical measures (mean, variance, RMS), frequency-domain characteristics (FFT, power spectral density), and higher-order statistical measurements (kurtosis, skewness, spectral entropy). In terms of reliability and bias, the original study used anonymised demographic information with a highly homogeneous spread of demographics across its elite player population.

The regression task was formally defined as predicting the continuous target variable `a_max` (maximum acceleration during swing) measured in LSB/G units and represents both swing power and intensity. When observing the dataset, the `a_max` exhibited large variability range:

$$0.0 - 48,890.8 \text{ LSB/G, mean: } 15,493.8 \pm 7,516.3 \text{ LSB/G} \tag{1}$$

indicating a diverse set of swing characteristics across the sample.

The analysis will focus on predicting `a_max` using three distinct regression approaches in order to meet the requirements of the task:

1. Ordinary Least Squares (OLS): a linear baseline model providing interpretable coefficients and establishing performance benchmarks for more complex approaches;
2. Ridge Regression: a linear model with L2 regularisation to control overfitting; and
3. Neural Network (NN): a non-linear model with a symmetric architecture and target scaling to improve both convergence and generalisation.

Further, Principal Component Analysis (PCA) was used for dimensionality reduction, achieving a 90.1% variance explanation with 33 components. We used a 90% cumulative explained variance threshold, which in practice required 33 components (explaining 90.1% of the variance). Similarly, standardised original features with target scaling (`StandardScaler` method from `sklearn`) was applied for the NN in order to address the large target variable range and thereby improve training efficiency for that specific model.

The performance of all three models will be evaluated using the following metrics:

Table 1: Evaluation Metrics

Metric	**Formula**
Mean Squared Error (MSE)	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
Mean Absolute Error (MAE)	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \hat{y}_i\rvert$
Coefficient of Determination (R^2)	$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$

Statistical significance testing via paired t-tests (with 95% confidence intervals) will be used to further assess performance differences between models.

2 Data Preprocessing

2.1 Initial Data Analysis

The target variable `a_max` demonstrated substantial variability, with a right-skewed distribution ranging from 0.0 to 48,890.8 LSB/G (mean: 15,493.8 ± 7,516.3 LSB/G) (Figure 1). This wide range reflects diverse swing characteristics across the elite player population, from controlled placement shots to powerful attacking serves.

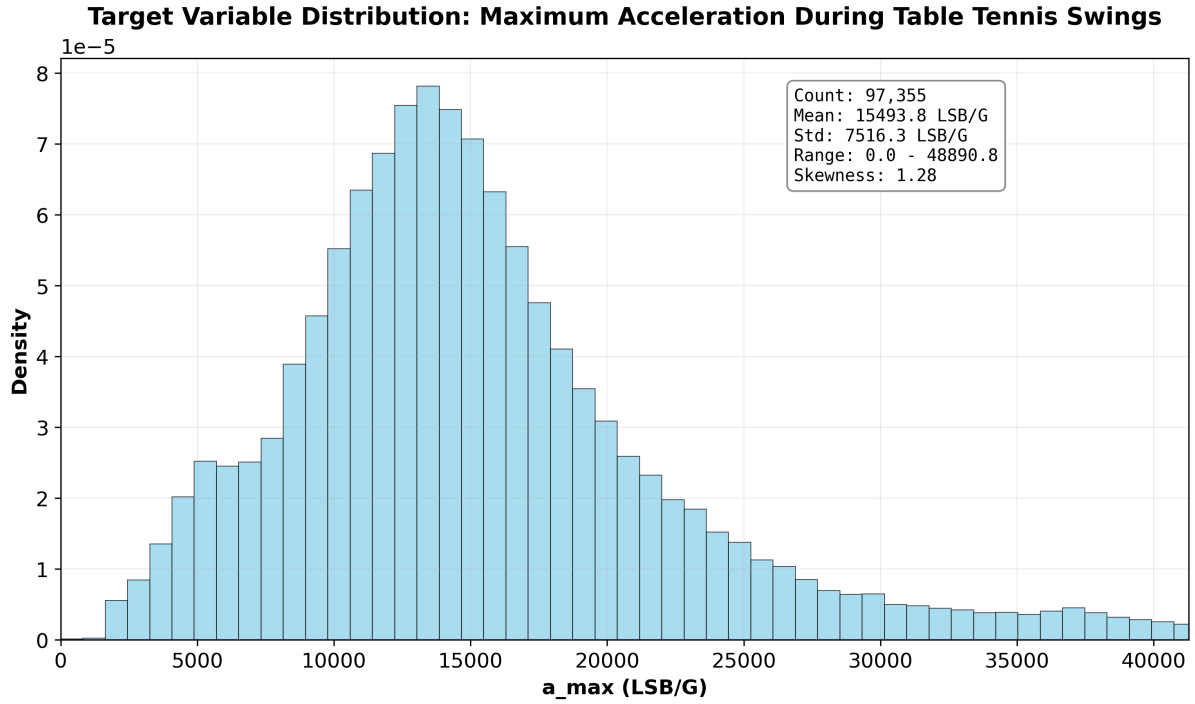

Figure 1: Target variable (`a_max`) distribution

Initial observations revealed heterogeneous scaling throughout the dataset. Accelerometer features (`ax`, `ay`, `as`) and gyroscope features (`gx`, `gy`, `gs`) exhibited different magnitude ranges, requiring careful normalisation strategies. Statistical features (mean, variance, RMS) showed varying levels of skewness, whereas frequency domain features (like FFT and power spectral density) showed more log-normal characteristics (Figure 2).

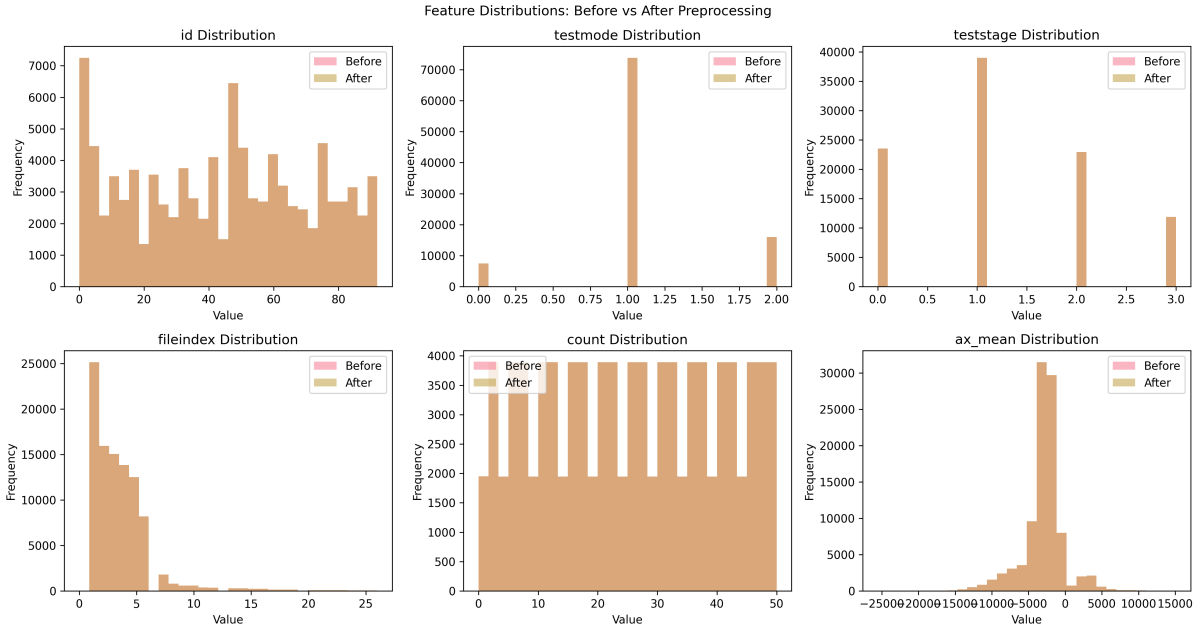

Figure 2: Feature distributions before and after preprocessing

The dataset contains no missing data, removing any potential bias from imputation methods.

Feature correlation analysis (Figure 3) identified 28 highly correlated feature pairs ($|r| \geq 0.8$), indicating high multicollinearity amongst features. Below are the key findings that were deduced from the correlation matrix:

- Accelerometer variance-RMS relationships: `ax_var` - `ax_rms` (r = 0.923), `ay_var` - `ay_rms` (r = 0.937)
- Cross-axis accelerometer correlations: `ax_var` - `ay_var` (r = 0.824), `ax_var` - `ay_rms` (r = 0.819)
- Target variable correlations: `ax_var` - `a_max` (r = 0.907), `ay_var` - `a_max` (r = 0.890)

These strong correlations justified the use of Principal Component Analysis (PCA) for dimensionality reduction and multicollinearity mitigation. The high correlation between variance and RMS features reflects their mathematical relationship, while cross-axis correlations indicate coordinated movement patterns during swing execution.

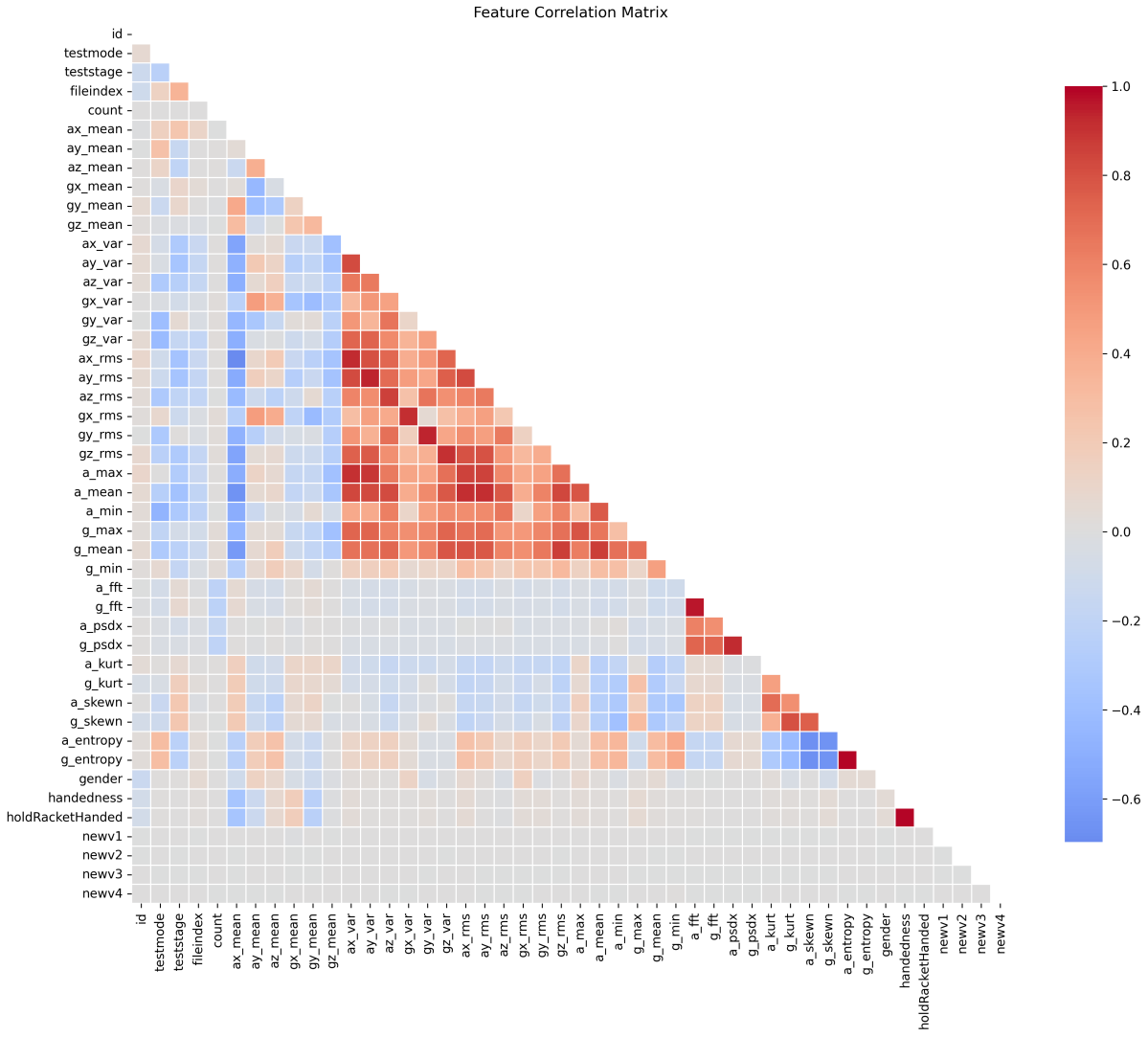

Figure 3: Feature correlation matrix heatmap

2.2 Features

PCA was used to address multicollinearity and reduce dimensionality while preserving essential variance. Mean centring of features (subtract mean method) was applied prior to PCA in order to satisfy its required assumptions and to ensure proper component extraction. Using a 90% cumulative explained variance threshold, 33 principal components were selected, explaining 90.1% of the total variance in the feature space. The 90% threshold follows standard practice for dimensionality reduction. This value balances information retention with computational efficiency while also avoiding overfitting from excessive dimensionality.

Table 2: PCA Ablation Study Results

Components	Explained Variance (%)	Validation R^2
5	38.8	0.7345
10	54.5	0.8633
15	64.7	0.8847
20	72.9	0.8866
25	80.4	0.8955
30	86.9	0.8972
33	90.1	0.9028

The scree plot (Figure 4) displays the individual and cumulative explained variance across principal components, showing the elbow point that guided the 33-component selection.

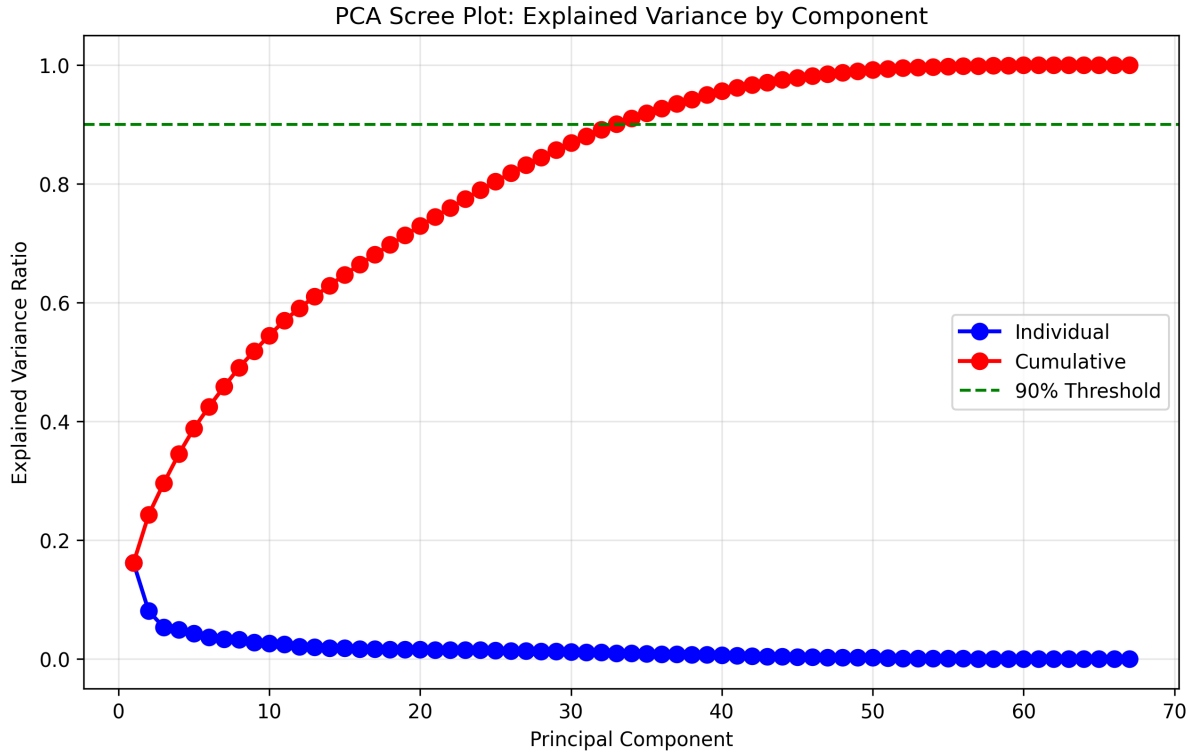

Figure 4: PCA scree plot illustrating individual and cumulative explained variance

2.3 Scaling & Normalisation

Linear Models (OLS and Ridge): Min-max scaling applied to normalise numerical features to [0, 1] using:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{2}$$

This approach ensures comparability across features without assuming Gaussian distribution and is appropriate for the bounded sensor data. It's also useful because it preserves the relative relationships between features while also standardising their ranges for optimal linear model performance, decreasing the risk of overfitting.

Neural Network: standardisation (z-score normalisation) applied using:

$$x' = \frac{x - \mu}{\sigma} \tag{3}$$

where μ is the feature mean and σ is the standard deviation. Standardisation is preferred for NNs as it aids gradient flow during training and helps with convergence stability. Additionally, target scaling using `StandardScaler` from `sklearn` was applied to the target variable, transforming the range from 0-48,890.8 LSB/G to approximately -2.064 to 4.454, significantly improving NN convergence.

2.4 Dataset Splitting

The dataset was partitioned using **stochastic** sampling to ensure representative distribution across all subsets. This 70%/15%/15% ratio represents standard practice in machine learning, as it maximises training data availability while providing sufficient validation and test sets for reliable performance estimation and hyperparameter tuning. The large dataset size (97,355) ensures adequate statistical power across all subsets:

- **Training Set:** 68,147 samples (70%) - used for model parameter estimation;
- **Validation Set:** 14,604 samples (15%) - used for hyperparameter tuning and model selection; and
- **Test Set:** 14,604 samples (15%) - used for the final model evaluation and performance measuring.

All three models used the same split and a fixed `math.random` seed to ensure fair and unbiased model comparisons. I used this approach as it eliminates variability inherently introduced by using different data partitions, thereby increasing the statistical validity of performance comparisons and ensuring that observed differences reflect true model capabilities *rather* than artefacts that could be caused by inconsistent sampling.

3 Regression Models

3.1 Ordinary Least Squares (OLS)

Ordinary Least Squares regression served as the baseline model using the 33 principal components derived from PCA preprocessing. The model provides a simple linear relationship between the transformed features and the target variable `a_max`, serving as a benchmark for more sophisticated approaches.

The OLS model achieved solid baseline performance on the training and validation sets:

Table 3: OLS Model Performance

Metric	Training Set	Validation Set
MSE	5,433,105	5,480,291
MAE	1,715.9	1,725.1
R^2	0.9034	0.9028

The close alignment between training and validation metrics (R^2 difference: 0.0006) indicates minimal overfitting, confirming the appropriateness of the 33-component PCA transformation for linear modelling. The model explains approximately 90.3% of the variance in swing acceleration, establishing a strong baseline for comparison.

Residual analysis for the OLS model reveals systematic patterns indicating potential non-linear relationships not captured by the linear formulation. The residual vs. fitted plot (Figure 5) shows heteroscedasticity and curved patterns, suggesting that more sophisticated modelling approaches may capture additional variance in the data.

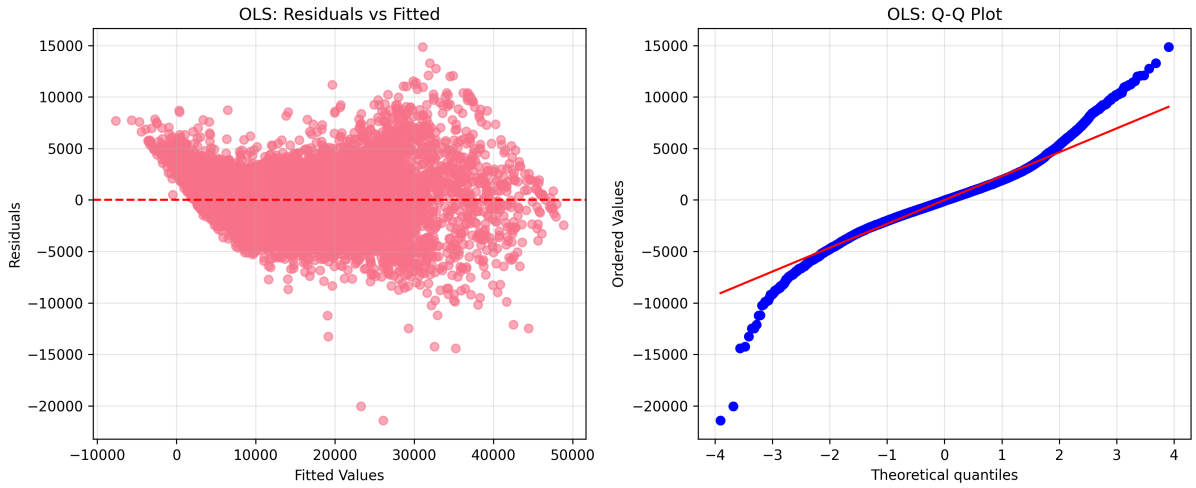

Figure 5: OLS residual vs. fitted and Q-Q plots

3.2 Ridge Regression

Ridge Regression extends the OLS approach by incorporating L_2 regularisation in order to control overfitting and improve generalisation. See the regularised cost function below:

$$J'(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \boldsymbol{\theta}^T \mathbf{x}_i)^2 + \lambda |\boldsymbol{\theta}|_2^2 \tag{4}$$

where λ represents the penalty strength controlling the trade-off between model fit and coefficient magnitude.

Comprehensive grid search was conducted over penalty strengths $\Lambda = \{10^{-3+0.1n} : 0 \leq n \leq 60\}$, spanning 61 values from 0.001 to 1000. The optimal penalty strength was $\alpha = 0.001$, producing a validation $R^2 = 0.9028$. This indicates that minimal regularisation was required for Ridge in order to achieve good performance.

The Ridge Regression model with optimal penalty strength achieved nearly identical performance to OLS:

Table 4: Ridge Regression Model Performance

Metric	Training Set	Validation Set	Difference from OLS
MSE	5,433,105	5,480,291	$< 0.001\%$
MAE	1,715.9	1,725.1	$< 0.001\%$
R^2	0.9034	0.9028	$< 0.001\%$

The minimal difference between OLS and Ridge performance suggests that PCA preprocessing effectively addressed multicollinearity concerns, reducing the need for further regularisation. The optimal penalty strength $\alpha = 0.001$ represents minimal regularisation, confirming that the 33-component feature space was well-suited for linear modelling.

The regularisation path demonstrated that performance remained stable throughout a wide range of penalty values until it started to degrade at higher regularisation strengths. This stability indicates robust model behaviour and suggests that the feature engineering process successfully created a well-conditioned learning problem.

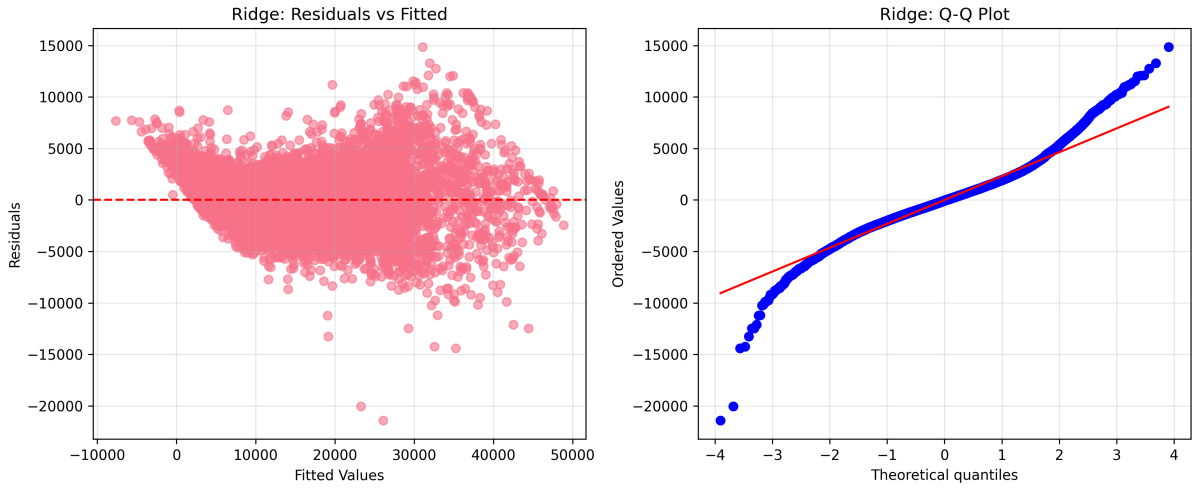

Figure 6: Ridge residual vs. fitted and Q-Q plots

3.3 Neural Network

A feedforward multilayer perceptron neural network was used as the third and final model to explore possible non-linear relationships arising from the TTSWING dataset. This approach uses multiple layers of connected nodes to learn complex patterns that linear models cannot detect. Additionally, neural networks are well-suited for this type of sensor data as they can identify interactions between different types of measurements without making assumptions about how these features should be combined. I further made two choices to improve the NN's performance:

- using the original 67 features instead of PCA transformed features in order to preserve complex relationships; and
- scaling the target variable to help the model learn on the dataset more effectively.

Feedforward Multilayer Perceptron Architecture:

Each neuron in a feedforward multilayer perceptron neural network applies a weighted linear combination of its inputs followed by a non-linear activation function:

$$z_j^{(l)} = \sum_i w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \tag{5}$$

$$a_j^{(l)} = f(z_j^{(l)}) \tag{6}$$

where $z_j^{(l)}$ is the pre-activation value, $a_j^{(l)}$ is the activation output, $w_{ij}^{(l)}$ are the weights, $b_j^{(l)}$ are the biases, and $f(\cdot)$ is the activation function (ReLU in our case).

The network learns through backpropagation, computing gradients of the loss function with respect to each parameter and updating weights using gradient descent. The chain rule enables efficient gradient computation through the network layers:

$$\frac{\partial L}{\partial w_{ij}^{(l)}} = \frac{\partial L}{\partial a_j^{(l)}} \cdot \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} \cdot \frac{\partial z_j^{(l)}}{\partial w_{ij}^{(l)}} \tag{7}$$

Below is the finalised architecture used for the NN, with the number of nodes in each layer being 160:

- Input layer (67 units);

- Hidden Layer 1 (160 units, ReLU + BatchNorm);
- Hidden Layer 2 (160 units, ReLU + BatchNorm);
- Output layer (1 unit, linear activation);
- The symmetric (160 → 160 → 1) design provides sufficient capacity for complex pattern learning while maintaining computational efficiency with a total of 37,441 trainable parameters.

Formulae:

$$\mathbf{q}^{(1)} = \text{ReLU}(\text{BN}(\mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)})) \tag{8}$$

$$\mathbf{q}^{(2)} = \text{ReLU}(\text{BN}(\mathbf{W}^{(2)}\mathbf{q}^{(1)} + \mathbf{b}^{(2)})) \tag{9}$$

$$\hat{y} = \mathbf{W}^{(3)}\mathbf{q}^{(2)} + \mathbf{b}^{(3)} \tag{10}$$

where $\mathbf{q}^{(1)}$ and $\mathbf{q}^{(2)}$ are the outputs of the first and second hidden layers, respectively, and $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, and $\mathbf{W}^{(3)}$ are the weight matrices for the first, second, and third layers, respectively.

The NN was trained using MSE loss with Adam optimiser (learning rate 0.0001, batch size 64). Regularisation included dropout (p=0.2), L2 weight decay ($\lambda = 1 \times 10^{-4}$), and early stopping (patience=100 epochs). A critical innovation was applying `StandardScaler` from `sklearn` to the target variable. This transformed the range from 0-48,890.8 LSB/G to approximately -2.064 to 4.454, dramatically improving convergence behaviour.

In summary, I found that the NN observed strong positive convergence characteristics (Figure 7), with the best validation loss achieved at epoch 52. However, training continued until early stopping was triggered at approximately epoch 152 due to the 100-epoch patience setting. Training loss reduced from 0.457 to 0.025 (94.6% reduction) while validation loss decreased from 0.060 to 0.009 (84.9% reduction). This indicates the model achieved **stable learning without overfitting**.

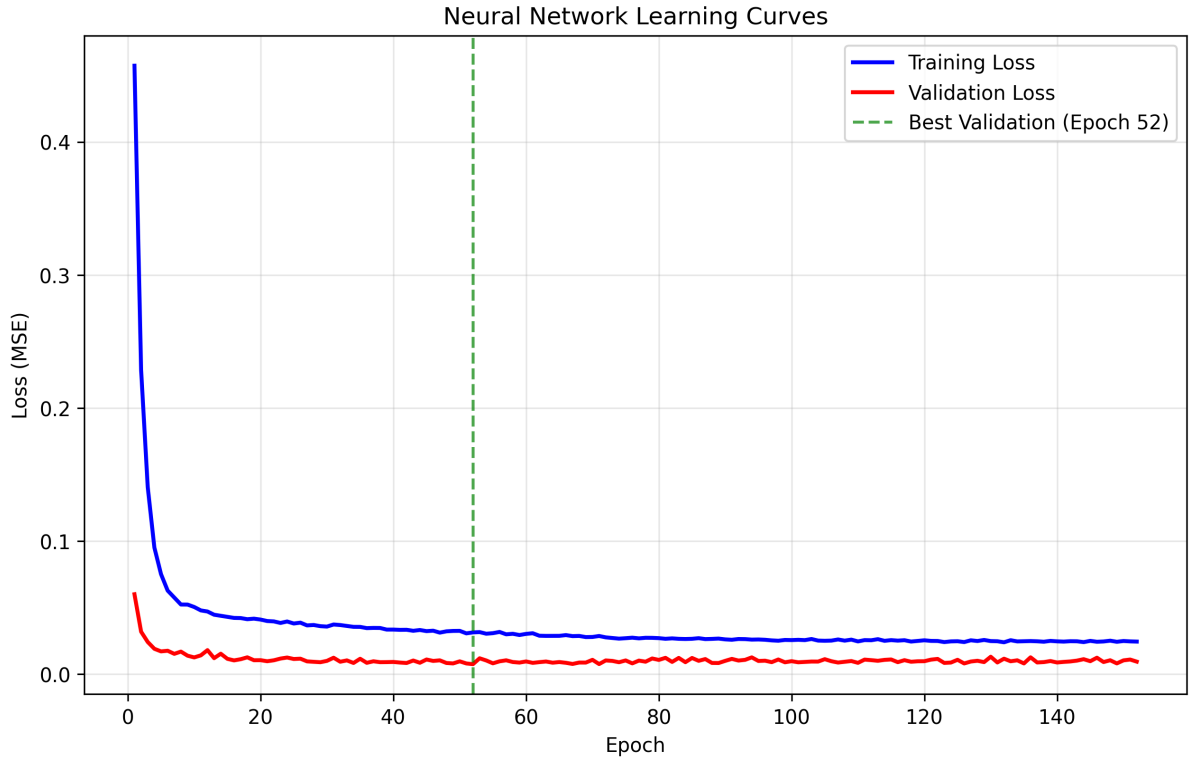

Figure 7: Neural Network training and validation learning curves

Table 5: Neural Network Model Performance

Metric	Training Set	Validation Set	Improvement over OLS
MSE	502,089	533,191	90.8% reduction
MAE	555.5	570.8	66.9% reduction
R^2	0.9911	0.9905	8.7% improvement

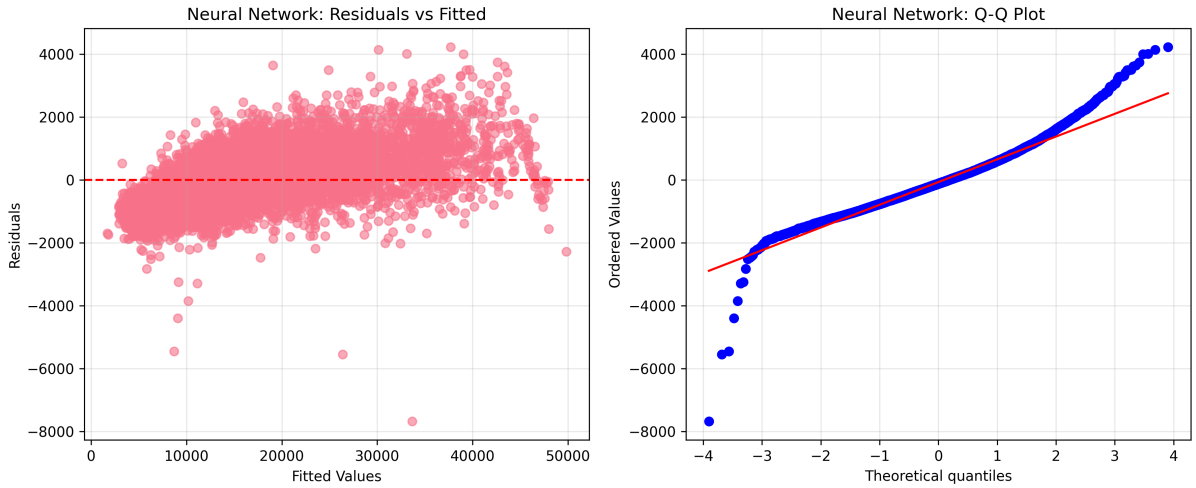

Figure 8: Neural Network residual vs. fitted and Q-Q plots

The NN's superior performance is evident in both the quantitative metrics (Table 4) and the residual analysis (Figure 8). The NN achieved substantially better performance than the linear models with minimal generalisation gap (training-validation R^2 difference: 0.0006). The target scaling approach proved essential for handling the large magnitude range of swing acceleration measurements, allowing the model to capture more complex biomechanical patterns that linear approaches inherently cannot represent.

4 Analysis

4.1 Test Set Evaluation

Final model evaluation was conducted on the held-out test set (14,604 samples, 15% of total data) to assess generalisation performance. The NN achieved exceptional test performance with $R^2 = 0.9906$, explaining 99.06% of variance in maximum swing acceleration (Table 6). OLS and Ridge achieved virtually identical performance (MSE difference $< 0.01\%$), confirming that minimal regularisation was optimal for this well-conditioned problem. The NN observed a 90.3% reduction in MSE compared to linear models, representing a large improvement in predictive accuracy.

Table 6: Test Set Performance

Model	MSE	MAE	R^2
OLS	5,553,727	1,724.7	0.9037
Ridge	5,553,727	1,724.7	0.9037
Neural Network	541,457	567.8	0.9906

Table 7: Training vs. Test Performance Comparison

Model	Training R^2	Validation R^2	Test R^2	Generalisation Gap
OLS	0.9034	0.9028	0.9037	+0.0003
Ridge	0.9034	0.9028	0.9037	+0.0003
Neural Network	0.9911	0.9905	0.9906	+0.0001

All models showed strong generalisation with minimal performance degradation from training to test sets (Table 6). The NN showed high consistency throughout all data splits, observing only a 0.0001 R^2 difference between validation versus test performance.

4.2 Statistical Significance Testing

Comprehensive paired t-tests were conducted to assess statistical significance of performance differences between models using 95% confidence intervals ($\alpha = 0.05$) (Table 8). OLS vs. Ridge showed no statistically significant difference (t = 1.016, p = 0.310, Cohen's d $\approx$ 0), confirming that minimal regularisation was appropriate for this dataset. The NN substantially outperformed both OLS and Ridge (t = 48.52, p $<$ 0.0001, Cohen's d = 0.549), indicating that the NN's performance is significantly superior with large effect size and tight CI's.

Table 8: Statistical Significance Testing Results

Comparison	Significant	p-value
OLS vs. Ridge	No	0.310
OLS vs. Neural Network	Yes	< 0.0001
Ridge vs. Neural Network	Yes	< 0.0001

4.3 Summary

1. **OLS:** Simple and interpretable baseline model with fast training and stable performance across data splits. However, linear assumptions limit complex pattern capture and result in lower predictive accuracy compared to neural approaches. Appropriate for interpretable baseline analysis and situations requiring simple, explainable models.

2. **Ridge Regression:** Controls overfitting via L_2 regularisation while maintaining linear model interpretability. Observed minimal improvement over OLS, and is still constrained by linear model assumptions.

3. **Neural Network:** Maps complex non-linear biomechanical patterns with high predictive performance (R^2 = 0.9906) and high generalisation. Target scaling innovation enables convergence for large-magnitude targets. However, it uses a single optimised configuration rather than extensive hyperparameter search, less interpretable than linear models, and computationally intensive training. Therefore, it is optimal for high-accuracy prediction tasks where performance is prioritised over interpretability.

See Figure 9 for the performance comparison across all metrics, clearly indicating the NN's superior performance with 95% confidence intervals.

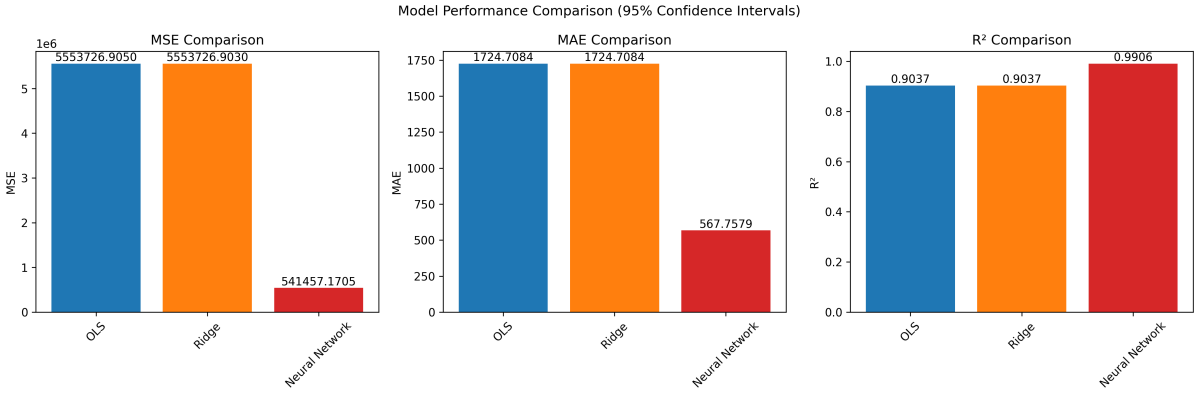

Figure 9: Model performance comparison (MSE, MAE, R^2 with 95% confidence intervals)

Residual analysis showed distinct patterns across the three modeling approaches (Figures 5, 6, and 8). OLS and Ridge display systematic residual patterns that indicate unmodelled non-linear relationships, with heteroscedasticity and curved trends suggesting that linear assumptions were violated for predicting `a_max`. In contrast, the NN achieved a more stochastic distribution around zero with reduced systematic patterns, indicating successful capture of non-linear relationships and improved model performance.

References

[1] Chou, C., Chen, S., Sheu, Y., Chen, H., Sun, M., & Wu, S. K. (2023). TTSWING: A novel dataset for table tennis swing analysis. *arXiv preprint arXiv:2306.17440v1.*

[2] COMP4702 Academic Team. (2025). COMP4702 2025 Course Notes. School of EECS, The University of Queensland.

[3] Jurafsky, D., & Martin, J. H. (2024). *Speech and Language Processing.* Draft of January 12, 2025.