Biomechanical Regression: Predicting Peak Force from Running Kinematics

Biomechanics Data Regression Report

January 29, 2026

Abstract

This report summarizes a supervised learning workflow for predicting peak ground reaction force during running foot plant using biomechanical features (e.g., stride characteristics and joint angles). Two targets are considered: left peak force (`lpeakforce`) and right peak force (`rpeakforce`). The analysis includes data cleaning and median imputation, exploratory visualization, feature ranking via univariate F-tests (`SelectKBest` with `f_regression`), and model benchmarking using `LazyPredict` followed by focused evaluation of linear regression and random forest baselines.

1 Background and Objective

Biomechanics studies the mechanical principles governing human movement, including forces and motions during walking and running. Predicting peak force in a runner's stride can support performance optimization and injury-risk reduction by linking force outcomes to measurable kinematic variables.

Objective. Predict peak force (`lpeakforce` and `rpeakforce`) using the measured biomechanical features in the dataset.

2 Dataset and Features

The workflow assumes a tabular dataset loaded from `alldata-track.csv`. Non-numeric identifiers (e.g., `ID`, `username`, `email`) are removed if present.

Inputs

Example features include left ground contact time (`lgt`), stride rate (`sr`), stride length (`sl`), left knee swing angle (`lkneeswing`), left hip flexion/extension (`lhipflex`, `lhipext`), and center-of-mass and limb displacement metrics.

Outputs

- `lpeakforce`: left leg peak force
- `rpeakforce`: right leg peak force

3 Methods

3.1 Preprocessing

The provided Python analysis performs the following preprocessing steps:

1. Load the dataset with `pandas`.
2. Drop non-numeric identifier columns when present.
3. Report missing values; if any missing values exist, impute numeric columns with the median.
4. Restrict features to numeric columns prior to model fitting.

3.2 Exploratory Visualizations

Exploratory plots include:

- Correlation matrix over numeric columns.
- Target distributions for `lpeakforce` and `rpeakforce`.
- Scatter plots of each target versus plantar velocity (`lplantarvel`).

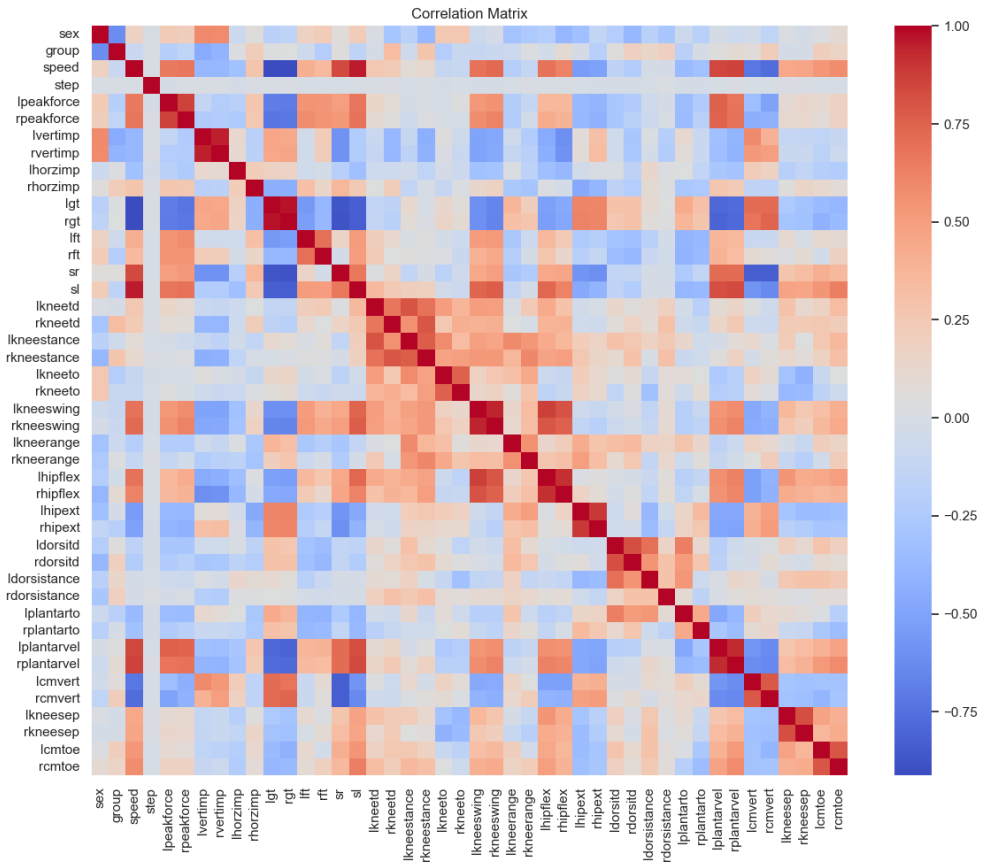
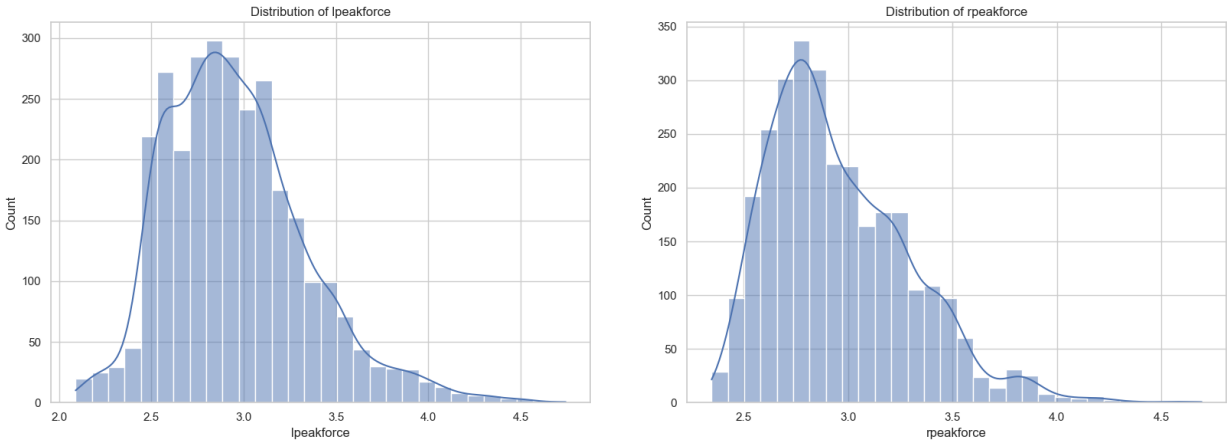
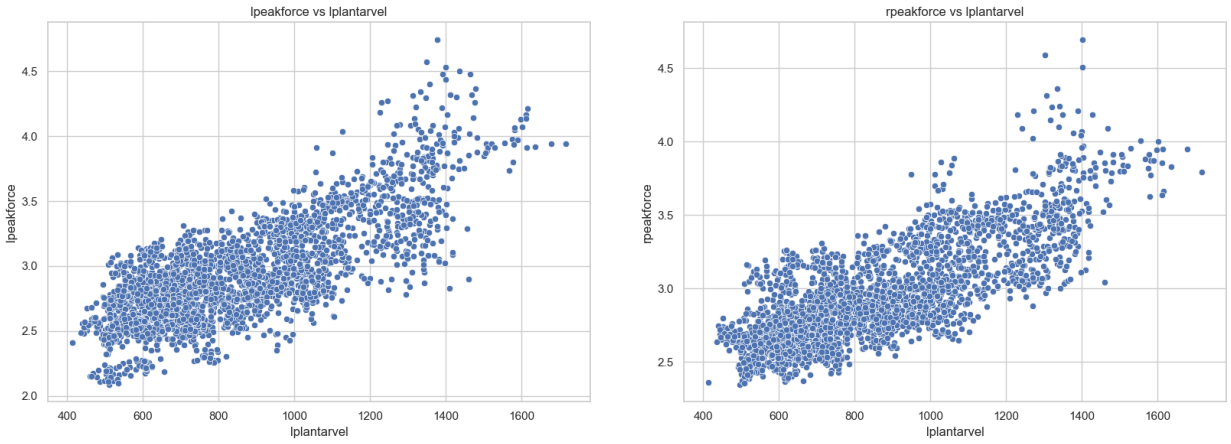


Figure 1: Correlation matrix over numeric features (Pearson correlation).



(a) `lpeakforce` (b) `rpeakforce`

Figure 2: Distributions of target variables.



(a) `lpeakforce` vs. `lplantarvel` (b) `rpeakforce` vs. `lplantarvel`

Figure 3: Scatter plots of peak force against plantar velocity (`lplantarvel`).

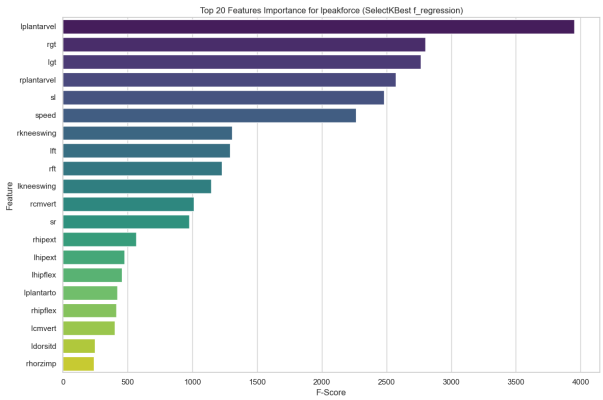
3.3 Feature Ranking

Feature importance is estimated using a univariate linear association test (`SelectKBest` with `f_regression`) applied to each target.

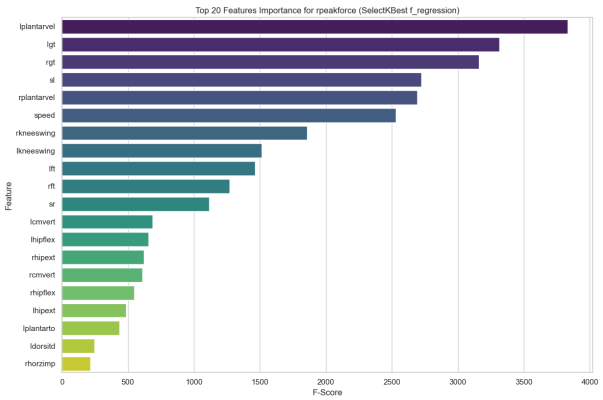
3.4 Model Training and Evaluation

Two evaluation stages are performed:

1. **Model screening:** Standardize features and targets, then benchmark many scikit-learn regressors via `LazyRegressor` (excluding a small set of models).
2. **Focused models (unscaled):** Train linear regression and random forest regressors on the original feature scale, evaluate on a held-out test set (80/20 split, `random_state=42`).



(a) Top features for `lpeakforce`



(b) Top features for `rpeakforce`

Figure 4: Univariate feature ranking using F-scores (higher indicates stronger linear association with the target).

The focused evaluation reports mean absolute error (MAE), root mean squared error (RMSE), and R^2.

Table 1: Top 5 regressors from LazyPredict screening (standardized target).

(a) `lpeakforce`

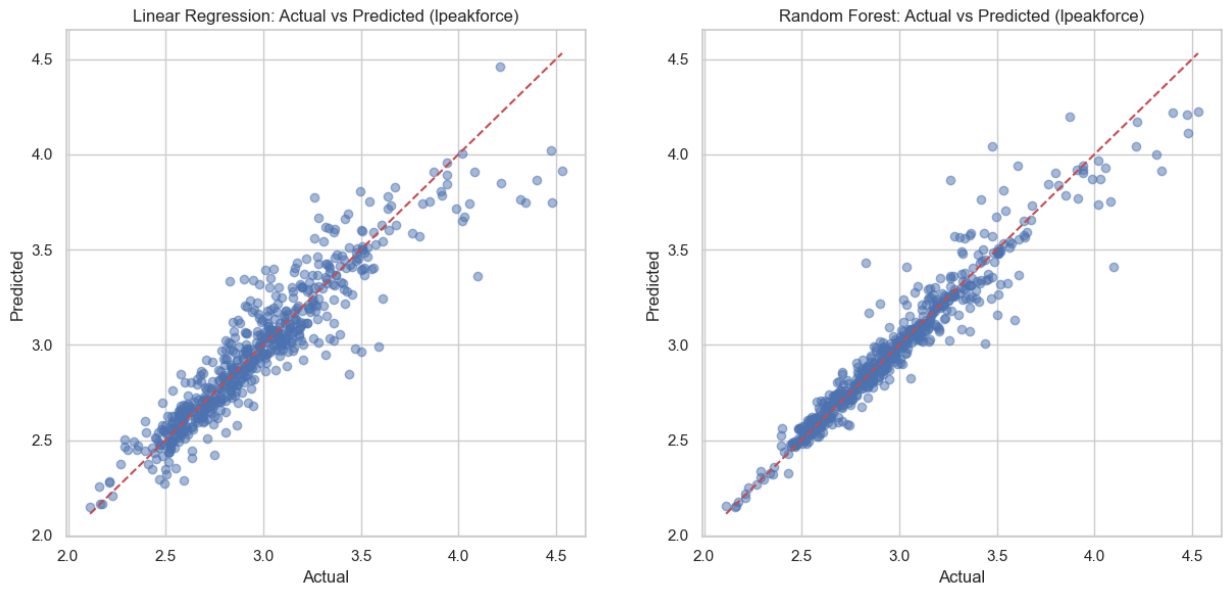
Model	Adj. R^2	R^2	RMSE
ExtraTreesRegressor	0.943	0.947	0.233
MLPRegressor	0.938	0.942	0.243
KNeighborsRegressor	0.937	0.941	0.245
NuSVR	0.936	0.941	0.247
SVR	0.936	0.940	0.247

(b) `rpeakforce`

Model	Adj. R^2	R^2	RMSE
ExtraTreesRegressor	0.931	0.936	0.251
NuSVR	0.931	0.936	0.251
SVR	0.930	0.935	0.253
HistGradientBoostingRegressor	0.928	0.933	0.256
KNeighborsRegressor	0.927	0.933	0.258

4 Discussion

Relationship with plantar velocity. The scatter plots in Figure 3 highlight how peak force varies with plantar velocity (`lplantarvel`) for both the left and right targets. Because `lpeakforce` and `rpeakforce` are both output labels (and are typically strongly correlated), plotting one against the other is not informative for feature understanding and can encourage target leakage; focusing on `lplantarvel` provides a more interpretable predictor–response view.



(a) Linear regression (`lpeakforce`)

(b) Random forest (`lpeakforce`)

Figure 5: Actual vs. predicted peak force for `lpeakforce`. The dashed diagonal indicates perfect predictions.

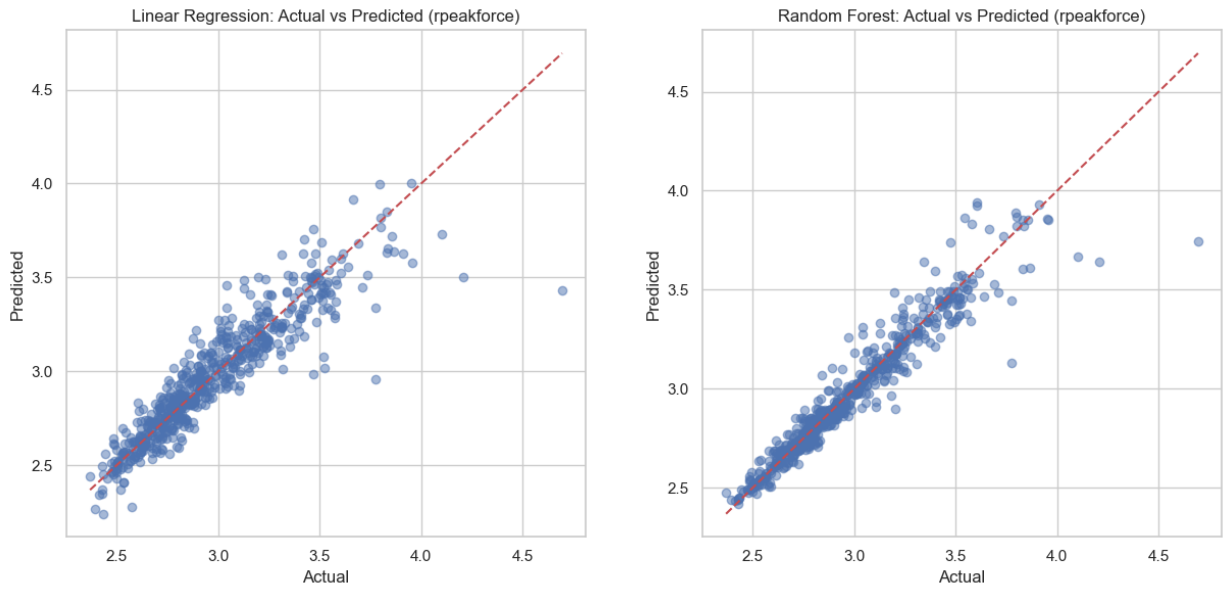
Common high-performing models. The top-5 benchmarked regressors share several model families across both targets (Table 1), notably `ExtraTreesRegressor`, support vector regression variants (`SVR`/`NuSVR`), and `KNeighborsRegressor`. This consistency suggests that the underlying mapping from kinematics to peak force is moderately nonlinear and benefits from flexible function classes, while remaining stable across left/right outcomes.

Recommendation. For a strong default choice, use `ExtraTreesRegressor`: it is the highest-ranked model for both `lpeakforce` and `rpeakforce` in the screening results and typically provides excellent accuracy with minimal feature engineering. If interpretability and a simpler model are priorities, consider `KNeighborsRegressor` (competitive performance and straightforward behavior) or a linear baseline; if smooth nonlinear fits are desired and computation is acceptable, `SVR`/`NuSVR` are also consistently strong.

Interpretation. The correlation and feature-ranking plots help identify which kinematic variables are most strongly associated with peak force under a linear association assumption. Model comparisons highlight whether nonlinear models (e.g., random forests) improve predictive accuracy over linear baselines.

Potential improvements.

- Use cross-validation (e.g., 5-fold) and report mean ± standard deviation of metrics.
- Consider leakage explicitly when predicting left vs. right targets (e.g., whether using the contralateral peak force as an input is permissible).
- Add robust outlier handling and domain-driven validity checks (e.g., plausible angle ranges, contact-time ranges).



(a) Linear regression (`rpeakforce`)

(b) Random forest (`rpeakforce`)

Figure 6: Actual vs. predicted peak force for `rpeakforce`. The dashed diagonal indicates perfect predictions.

- Report model interpretability (permutation importance or SHAP) for the best-performing model.