Mathematical Modeling and Algorithm Application
ISSN: 3006-0842 | Vol. 8, No. 2, 2026

Prediction Model for Standing Long Jump Performance Constructed Using Lasso Regression

Guoshun He[#, 1, *], Bicheng Li[#, 2], Bojia Gao[#, 2]

[1]College of Automotive Engineering, Foshan Polytechnic, Foshan 528137, China

[2]College of Intelligent Manufacturing, Foshan Polytechnic, Foshan 528137, China

[*] **Corresponding author**: Guoshun He (Email: HeGuoShun0618@163.com)

[#]**These authors also contributed equally to this work.**

Abstract: Against the backdrop of promoting the National Student Physical Fitness Standards, this paper proposes an AI posture analysis and evaluation framework for the standing long jump. It identifies key events from keypoint sequences and explains performance variations. The movement process is divided into three phases: takeoff, flight, and landing. Posture features are constructed based on trajectories of 33 body keypoints. Collected data undergoes cleaning and smoothing, removing 438 anomalous samples with all zeros and performing mean imputation on 26 samples. Kinematic representations are extracted via least-squares quadratic trajectory fitting during the flight phase. Lasso regression establishes relationships between posture variables and jump distance to screen critical motion features. Taking Athlete 1 as an example, the model predicted a performance of 1.545m and provided a short-term training target of 1.676±0.068m. This framework can be applied to physical assessment and training focus determination, with further validation required on larger samples and diverse body types.

Keywords: Lasso; Least Squares Method; Quadratic Polynomial Regression; Spearman; Pearson.

1. Introduction

The implementation of the National Student Physical Fitness Standard has shifted school physical education from participation-oriented practice toward outcome-focused assessment. The standing long jump is one of the most widely used tests in both fitness examinations and routine classes. Yet, performance depends not only on lower-limb power but also on technical details such as arm swing, trunk lean, and hip–knee–ankle coordination. In practice, instructors often rely on visual observation and experience to provide corrective feedback; however, in large classes and fast-changing movements, subtle errors are easily missed, making evaluation difficult to keep objective, quantifiable, and trackable over time.

Recent studies on the standing long jump have shifted from distance-only scoring to technique-aware assessment, since arm swing and trunk control can affect performance [1]. To support large-scale school testing, markerless camera-based pipelines are increasingly being adopted; comparative studies and reviews clarify their feasibility and limitations [2,3]. Meanwhile, methods such as OpenPose, high-resolution pose networks, and on-device tracking enable reliable keypoint extraction and interpretable kinematic features [4–6]. However, under occlusion, illumination changes, and fast motion, keypoint jitter can propagate to feature instability and reduce the reliability of action-quality evaluation [3,7]. In the modeling stage, least-squares fitting with calibration/smoothing is commonly used [8,9], complemented by Adaptive LASSO and statistical inference frameworks for feature selection and uncertainty handling [10–13].

Existing studies have pursued objective standing long jump assessment using camera-based recording and human pose estimation. In these pipelines, keypoint trajectories and joint-angle descriptors are extracted from video and subsequently used for phase identification, action-quality evaluation, and performance prediction, thereby extending assessment beyond distance-only scoring toward a combined, technique-aware quantification. For school-based deployment, two key challenges remain: keypoint noise induced by real-world conditions can compromise feature stability, and model outputs must be sufficiently interpretable to support direct instructional use. Accordingly, this study estimates trajectory parameters from keypoint sequences via least squares, summarizes the flight-phase motion using quadratic polynomial regression, and employs a LASSO regression model to map pose-derived features to performance while enforcing sparsity for factor identification. Without additional hardware requirements, the proposed pipeline has the potential to support standardized large-scale testing and to provide interpretable and verifiable quantitative evidence for individualized training.

2. Model

2.1. Lasso Regression Model

(1) Lasso is a linear regression model with L1 regularization, which controls model complexity by adding the sum of absolute coefficients as a penalty term to the loss function.

(2) Lasso is primarily employed for feature selection and overfitting prevention. By leveraging the geometric properties of L1 penalty, it compresses the coefficients of unimportant features to zero, thereby achieving automatic feature screening.

(3) The key to establishing the Lasso model is to solve the optimization problem with L1 penalty. Since the absolute value term is non-differentiable, the coordinate descent method combined with a soft threshold function is used to iteratively solve the coefficients.

The Lasso Model is shown in Figure 1.

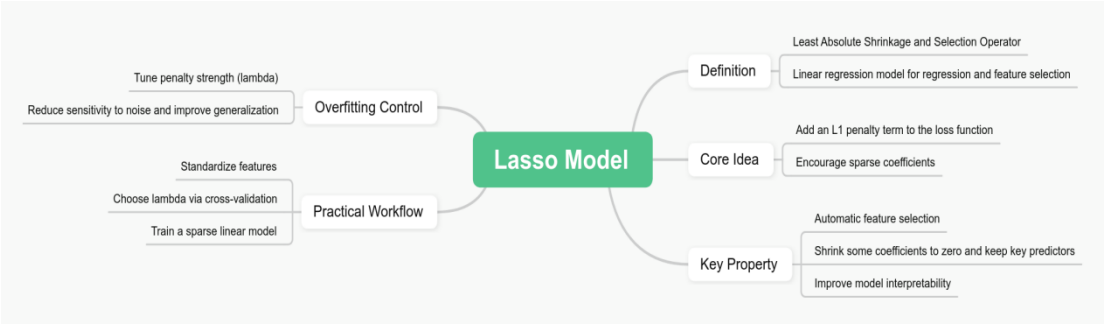


Figure 1 Schematic of the Lasso model

To incorporate both error fitting and sparsity constraints into a single objective, the Lasso model is formulated as the optimization problem in Eqs. (1)-(3), which shrinks the coefficients of less informative features toward zero and thus performs variable selection. Specifically, Eq. (1) gives the penalized least-squares objective, Eq. (2) shows the equivalent L1 constraint, and Eq.(3) provides the soft-thresholding update for iterative coefficient estimation.

$$\min_{\beta}\left\{\sum_{i=1}^{n}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}\left|\beta_j\right|\right\} \tag{1}$$

where n denotes the sample size, p the number of features, i the sample index, and j the feature index. Here, x_{ij} represents the j-th feature value of the i-th sample, y_i is the actual jump distance of the ii-th sample, β_j is the regression coefficient for the j-th feature, and λ is the regularization parameter that controls the penalty strength.

$$\sum_{j=1}^{p}\left|\beta_j\right|\le t \tag{2}$$

where $\beta=(\beta_1,\beta_{2,}.....\beta_\eta)$ is the coefficient vector, and t is a constraint boundary related to λ. A smaller value of t imposes a stronger constraint, resulting in a sparser model.

$$\beta_j^{new}=S\left(\sum_{i=1}^{n}x_{ij}\left(y_i-\hat{y}_i^{(j)}\right),\lambda\right) \tag{3}$$

where β_j^{new} stands for the updated coefficient of the j-th feature, and $S_\lambda(\cdot)$ denotes the soft-threshold function defined as $S_\lambda(z)=\mathrm{sign}(z)(|z|-\lambda)_+$. The term $\hat{y}_i^{(-j)}$ indicates the predicted value for the i-th sample using the current model without the j-th feature.

In Formula 1, the first term measures the model's goodness-of-fit, while the second term is the L1 penalty term where λ governs the penalty strength.

In Formula 2, t is a constraint parameter linked to λ; a smaller t leads to a stronger constraint and a sparser model.

In Formula 3, $S_\lambda(\cdot)$ is the soft-threshold function that compresses coefficients with absolute values smaller than λ to zero.

2.2. Quadratic Polynomial Regression

(1) Quadratic polynomial regression uses a parabolic curve to model the relationship between data points and variables.

(2) It is mainly used to describe and predict the relationship that has a "curve trend" or a "turning point".

(3) Quadratic polynomial regression models the relationship between the dependent variable y and the independent variable x. When a scatter plot of y versus x shows a single U-shaped (or inverted U-shaped) pattern, a quadratic form is a natural candidate. To estimate a nonlinear curve using linear least squares, the predictor is expanded by adding the squared term of x. In practice, we construct the design matrix with two regressors, x and fit.

The Quadratic Polynomial Regression is shown in Figure 2.

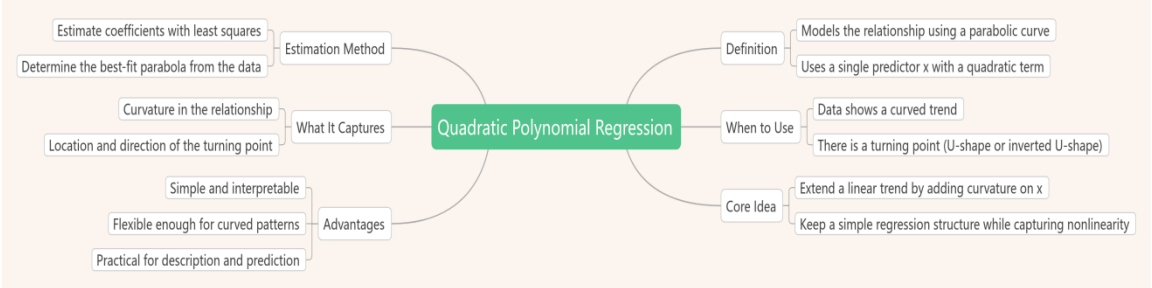


Figure 2 Schematic of quadratic polynomial regression

To capture the curved trend observed in the scatter plot while keeping the model simple and interpretable, the predictor is augmented with a quadratic term. Accordingly, the relationship between the dependent variable y and the independent variable x is modeled by a second-order polynomial, as shown in Eq. (4).

$$y=\beta_0+\beta_1 x+\beta_2 x^2 \tag{4}$$

where β_0 is nodal increment，β_1 represents the coefficient of a term, and β_2 means the quadratic coefficient.

Into a binary linear equation with x as the independent variable. Subsequently, we apply the established least squares method to estimate all coefficients in a single step, thereby determining the optimal fitted parabola. This model not only captures curve trends and turning points more flexibly but

also relies entirely on the robust framework of linear regression, combining intuitive understanding with practical utility.

2.3. Least Squares Method

(1) As a parameter estimation method, least squares determines the coefficients that minimize the sum of squared residuals between observations and model predictions.

(2) After fitting, the estimated coefficients support prediction and provide an interpretable summary of how the response changes with respect to the predictor (e.g., slope and intercept in a linear model).

(3) The methodology of least squares begins with a practical problem: extracting the fundamental mathematical relationships between variables from a dataset containing random fluctuations. This approach transforms the subjective concept of "optimal fitting" into a quantifiable optimization goal—minimizing the sum of squared prediction errors across all observation points. Its development process shifts from directly minimizing errors to pursuing the sum of squared errors, which not only avoids sign ambiguity but also assigns higher penalty weights to larger errors. For linear relationship assumptions, the residual squared values of each data point are summed to establish a loss function. Using calculus tools, the partial derivatives of this loss function are derived and set to zero, yielding a system of normal equations for parameters a and b. Ultimately, this leads to an analytically solvable formula that can be directly computed.

$$\hat{a} = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{5}$$

$$\hat{b} = \frac{\sum_{i=1}^{n} y_i - \hat{a}\sum_{i=1}^{n} x_i}{n} \tag{6}$$

where $\hat{a}$ is the slope estimate of the fitted line, representing the average change in the dependent variable y per unit change in the independent variable x, $\hat{b}$ represents Estimates the intercept of the fitted line, representing the predicted baseline value of y when x=0, n means sample size, x_i represents the independent variable value at the i-th observation point and y_i means actual observed value of the dependent variable at the i-th observation point.

3. Results

3.1. Preliminary Correlation Analysis

Before training the Lasso model, we examined the relationship between each posture feature and the jump distance using Pearson's and Spearman's correlation coefficients. Features showing significant correlations ($p<0.05$) with either measure were retained for subsequent modeling. This dual-coefficient approach ensured that both linear and monotonic nonlinear associations were considered, reducing the risk of omitting informative predictors due to distributional assumptions or outlier sensitivity.

Table 1 Comparison of Pearson's and Spearman's correlation measures used in feature screening

Aspect	Pearson's Correlation	Spearman's Correlation
Relationship type	Linear	Monotonic (linear or nonlinear)
Data assumption	Interval/ratio, normality, homoscedasticity	Ordinal/interval/ratio, no distributional assumption
Sensitivity to outliers	High	Low
Interpretation	"A unit change in X relates to a consistent change in Y"	"As X increases, Y tends to increase/decrease"

3.2. Model Specification and Training

A Lasso regression model was constructed to predict standing long jump performance based on standardized posture-derived features and anthropometric variables. Prior to model training, all features were standardized to zero mean and unit variance to ensure comparability and stabilize the regularization process.

The data set, sourced from the official competition portal (https://cumcm.cnki.net),was cleaned by removing 438 records with entirely missing keypoints and imputing 26 partially missing records using feature-wise means.

The model was trained by minimizing a penalized least-squares objective with L1 regularization. The optimal regularization strength (λ) was determined via 5-fold cross-validation, with the mean squared error (MSE) on validation folds as the selection criterion. The following formulations summarize the core methodology.

A. These is the Feature standardization:

$$X_{ij}^{*} = \frac{x_{ij} - \mu_j}{\sigma_j}, j=1,2,\ldots,p. \tag{7}$$

where μ_j and σ_j representing the mean and standard deviation of the j-th feature.

B. The Lasso estimation problem is formulated as:

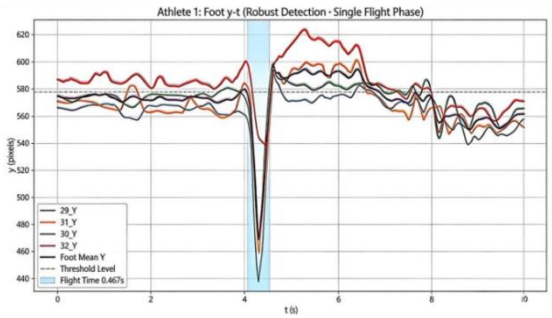
$$\hat{\beta} = \arg\min_{\beta} \left\{ \frac{1}{2n} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right\}. \tag{8}$$

$$\hat{y}_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}^{*}, \quad i=1,2,\ldots,n. \tag{9}$$

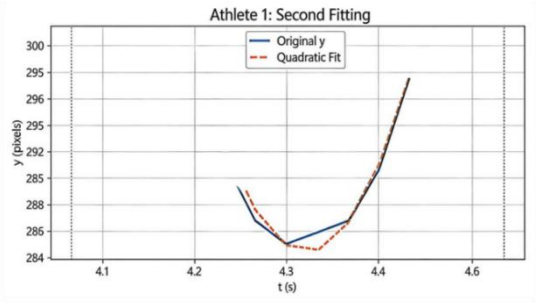
$$\text{CV-MSE}(\lambda) = \frac{1}{K}\sum_{k=1}^{K} \frac{1}{|D_k|} \sum_{i D_k} \left(y_i - \hat{y}_i^{(-k)}(\lambda)\right)^2. \tag{10}$$

where X is the standardized predictor matrix, y is the jump distance, β is the coefficient vector, λ is the L1 penalty, and CV-MSE is computed over K folds. With D_k representing the k-th validation fold and $\hat{y}_i^{(-k)}$ the prediction for sample ii when the k-th fold was held out during training.

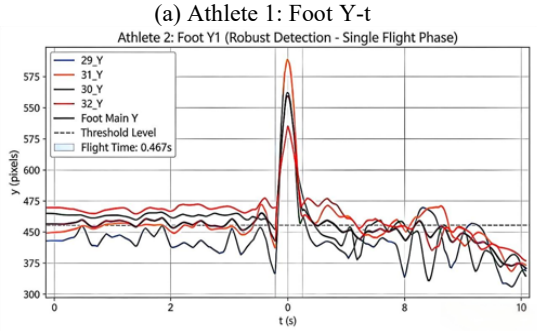
The Athlete 1 & 2: Foot y–t and Quadratic Fit on Flight Phase is shown Figure 3.



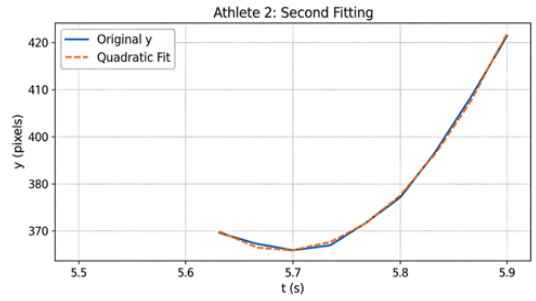
(a) Athlete 1: Foot Y-t



(b) Athlete 1: Second Fit



(a) Athlete 2: Foot Y-t



(b) Athlete 2: Second Fit

Figure 3 Athlete 1 & 2: Foot y–t and Quadratic Fit on Flight Phase

To connect the flight-interval detection step with the quadratic trajectory fitting for extracting flight-phase features, we show the time series of the vertical coordinate of the foot keypoint for Athletes 1 and 2 and apply a quadratic fit to the detected single-flight segment. The shaded region marks the extracted flight window, and the right subplots compare the fitted curve with the original trajectory to assess the extraction quality and to provide inputs for subsequent feature computation.

3.3. Results, Error Characteristics, and Phase Sensitivity

Applying the tuned model to Athlete11 produced a predicted standing long jump distance of 1.545 m. The model achieved CV-MSE=6986 px² under cross-validation, with the remaining error largely attributable to pixel-level keypoint noise and feature-to-score conversion. The pre-contact deceleration stage exhibited a 5.1% bias, indicating higher sensitivity near landing, where jitter and posture variability are amplified.

4. Conclusions

This study develops a pose-driven assessment and prediction pipeline for the standing long jump. The movement is partitioned into takeoff, flight, and landing, and key moments are extracted from the keypoint sequences to support phase-specific feature construction. The airborne segment is characterized using least-squares quadratic fitting, yielding compact kinematic descriptors for downstream modeling.

Data quality is addressed explicitly:438 all-zero records are removed, and 26 records are treated via mean imputation. Based on the cleaned dataset, an L1-regularized Lasso model is trained to associate posture-related indicators and physical profile variables with jump distance while retaining interpretability through sparsity. When applied to Athlete11, the model outputs a predicted distance of 1.545 m. Cross-validation yields CV-MSE=6986 px², and a 5.1% bias is observed in the pre-contact deceleration stage, indicating that landing is the most sensitive phase under the keypoint jitter.

Limitations include the current sample size and potential sensitivity to camera configuration and body-type diversity. Future work will expand the dataset, strengthen cross-subject validation, and incorporate uncertainty quantification and biomechanics-informed constraints to improve robustness and deployment readiness.

Acknowledgements

The study team thanks all participants for providing standing long jump recordings and granting permission to use pose data for research purposes. Appreciation is also extended to colleagues and mentors for feedback on the experimental design, as well as for assistance in verifying preprocessing rules and evaluation settings. Computational resources and support from the project group are gratefully acknowledged.

References

[1] ASHBY B M, HEEGAARD J H. Role of arm motion in the standing long jump[J]. Journal of Biomechanics, 2002, 35(12): 1631-1637.

[2] SONG K, HULLFISH T J, SILVA R S, et al. Markerless motion capture estimates of lower extremity kinematics and kinetics are comparable to marker-based across 8 movements[J]. Journal of Biomechanics, 2023, 157: 111751.

[3] DESMARAIS Y, MOTTET D, SLANGEN P, et al. A review of 3D human pose estimation algorithms for markerless motion capture[J]. Computer Vision and Image Understanding, 2021, 212: 103275.

[4] CAO Z, HIDALGO G, SIMON T, et al. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(1): 172-186.

[5] SUN K, XIAO B, LIU D, et al. Deep high-resolution representation learning for human pose estimation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019: 5686-5696.

[6] BAZAREVSKY V, GRISHCHENKO I, RAVEENDRAN K, et al. BlazePose: On-device real-time body pose tracking[EB/OL]. arXiv:2006.10204, 2020-06-17[2026-01-16].

[7] LIU J, WANG H, STAWARZ K, et al. Vision-based human action quality assessment: A systematic review[J]. Expert Systems with Applications, 2025, 263: 125642.

[8] ZONG Junji, LI Hongliang, AI Bo, et al. Design and analysis of a nonlinear coefficient calibration algorithm based on the least squares method[J]. Equipment for Electronic Products Manufacturing, 2025, 54(04): 33-36+77.

[9] MA Zhihua, CHEN Guanghui. Model calibration sampling estimation based on local polynomial regression[J]. Journal of Applied Statistics and Management, 2016, 35(01): 47-56.

[10] WANG Xiaoning, SUN Min, ZOU Mengwen. Integrating non-probability and probability samples via calibration assisted by the Adaptive LASSO model[J]. The World of Survey and Research, 2025(09): 84-96.

[11] ZHANG Tiande, YE Hong. Probability Theory and Mathematical Statistics[M]. Beijing: Posts & Telecommunications Press, 2024: 226.

[12] JIN Yongjin, HAO Yiwei. Model-based inference for non-probability samples[J]. Mathematics in Practice and Theory, 2019, 49(05): 246-255.

[13] WU Yaxuan, LIU Xiaoyu. Small area estimation based on multilevel models: Ratio estimation considering sampling error and measurement error[J]. Statistics and Decision, 2024, 40(07): 52-56.