

# Gait Condition Classification Using Machine Learning on Kinematic Features

Luis

FER Technologies

Biomedical Engineer, Mechatronics MSc

luis@fertechnology.com

**Abstract**—This paper presents a comparative study of machine learning models for classifying gait conditions using three core kinematic features: range of motion (ROM), joint angle, and angular velocity. We evaluate Linear Regression, Random Forest, and K-Nearest Neighbors (KNN) to determine their effectiveness in distinguishing between normal gait, knee brace, and ankle brace conditions. Results show that Random Forest outperforms linear models, suggesting nonlinear relationships between features and gait condition. The modular codebase enables reproducible experimentation and future integration of additional classifiers.

## I. INTRODUCTION

Gait analysis is essential in clinical biomechanics and rehabilitation, often relying on expert interpretation of motion data. In this study, we explore supervised machine learning models to automate gait condition classification using three numerical features: ROM, joint angle, and angular velocity.

We implement and compare three models:

- **Linear Regression** — used as a baseline to assess linear separability.
- **Random Forest Classifier** — captures nonlinear relationships and feature interactions.
- **K-Nearest Neighbors (KNN)** — leverages local proximity in feature space.

Each model is trained to classify gait condition into one of three categories: *normal*, *knee brace*, or *ankle brace*. Initial results show that Random Forest achieves higher accuracy and F1-scores, indicating that nonlinear models are better suited for this task.

## II. METHODOLOGY

The dataset consists of motion capture samples labeled by gait condition. Each sample includes ROM, joint angle, and angular velocity. Categorical variables such as joint type, leg side, and replication index are encoded numerically but not used in the initial models.

Missing values in angular velocity are filled with zero. The target variable is encoded as an integer: 0 for normal, 1 for knee brace, and 2 for ankle brace. All models use an 80/20 train-test split with stratification.

### A. Model Training

- **Linear Regression** is trained to predict condition code as a continuous variable. Metrics include Mean Squared Error (MSE) and  $R^2$ .

- **Random Forest Classifier** is trained using default hyperparameters. Evaluation includes accuracy, precision, recall, and F1-score.
- **KNN Classifier** uses  $k = 5$  neighbors. Performance is evaluated on the test set.

## III. CODE STRUCTURE

The project is organized into modular components:

- `data_processing.py` — handles feature extraction and cleaning.
- `models/linear_regression.py` — trains and evaluates the baseline model.
- `models/random_forest.py` — implements Random Forest with logging.
- `models/knn.py` — contains the KNN classifier.
- `utils/logger.py` — centralized logging utility.
- `main.py` — orchestrates training and evaluation.

All models share a common feature set and evaluation pipeline, enabling fair comparison and reproducibility.

## IV. RESULTS

## V. CONCLUSION

Random Forest outperforms Linear Regression and KNN in classifying gait conditions using ROM, angle, and angular velocity. Future work includes feature engineering with joint and leg variables, testing additional classifiers, and deploying the model in wearable systems.