Calories Burned Prediction: Machine Learning Regression Analysis

HPC, 2nd year: Matei Sonia
Ciocan Crina
Morar Cristina
Turcu Ciprian

Dataset Description

1

Dataset Summary

Total Samples: 973

Features: 14 (after encoding)

Data Split: 80% Training (778 samples), 20% Testing (195 samples)

2

Biometric Data

Age — Integer

Gender — String (encoded)

Weight (kg) — Float

Height (m) — Float

BMI — Float (derived)

3

Heart Rate & Workout Metrics

Max_BPM — Integer

Avg_BPM — Integer

Resting_BPM — Integer

Session_Duration (hours) — Float

Workout_Type — String (encoded)

Workout_Frequency (days/week) — Integer

4

Fitness & Target

Fat_Percentage — Float

Water_Intake (litres) — Float

Experience_Level — Integer (1–3)

Calories_Burned — Float (Target Variable)

Data Preprocessing Pipeline

A meticulous preprocessing pipeline was established to ensure data quality and model readiness.

1. Load Data

Dataset loaded from CSV, comprising 973 rows and 15 columns.

2. Handle Missing Values

Mean imputation for numeric, mode for categorical. No missing values detected in this dataset.

3. Handle Outliers

IQR method (1.5× threshold) used to cap outliers at boundaries, preserving sample size.

4. Categorical Encoding

Label Encoding applied to 'Gender' and 'Workout_Type' to convert strings to numeric values.

5. Column Normalization

Spaces replaced with underscores and special characters removed for consistency.

6. Train-Test Split

An 80/20 split (778 training, 195 testing samples) with `random_state=42` for reproducibility.

7. Feature Scaling

`StandardScaler` applied for zero mean and unit variance (`z = (x - μ) / σ`). Crucial for distance-based algorithms like KNN and Linear Regression. Gradient Boosting models do not require scaling.

Linear Regression

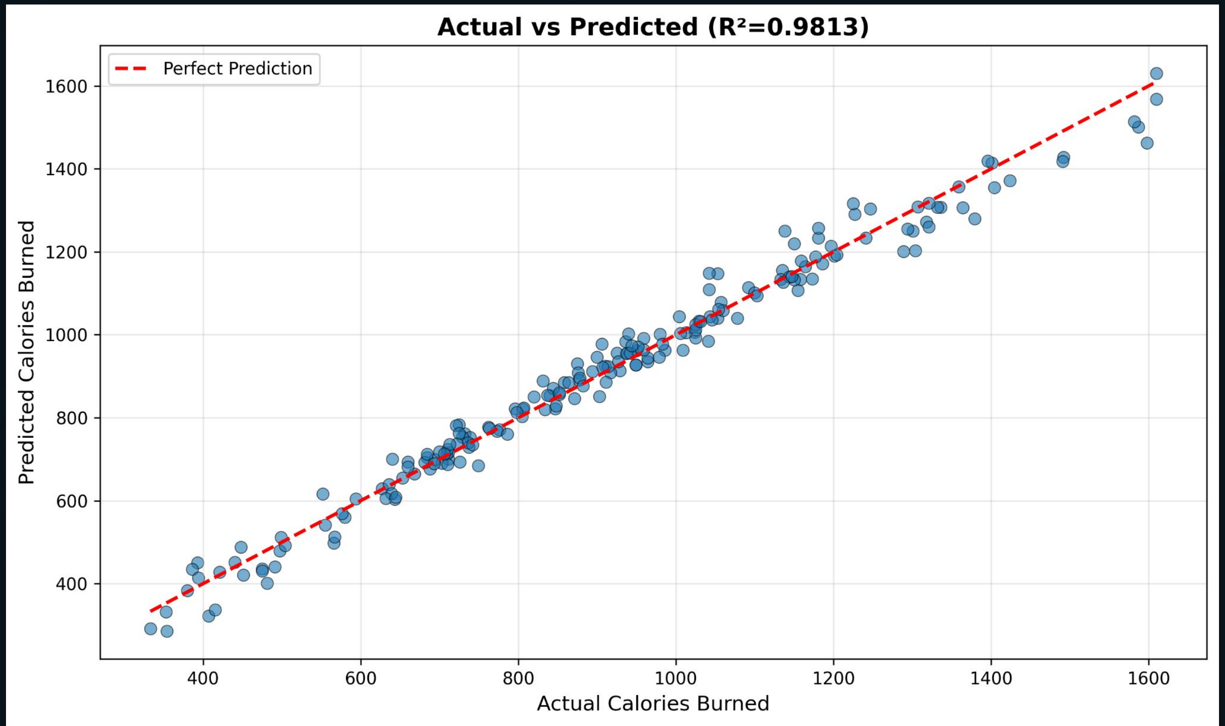
Methodology

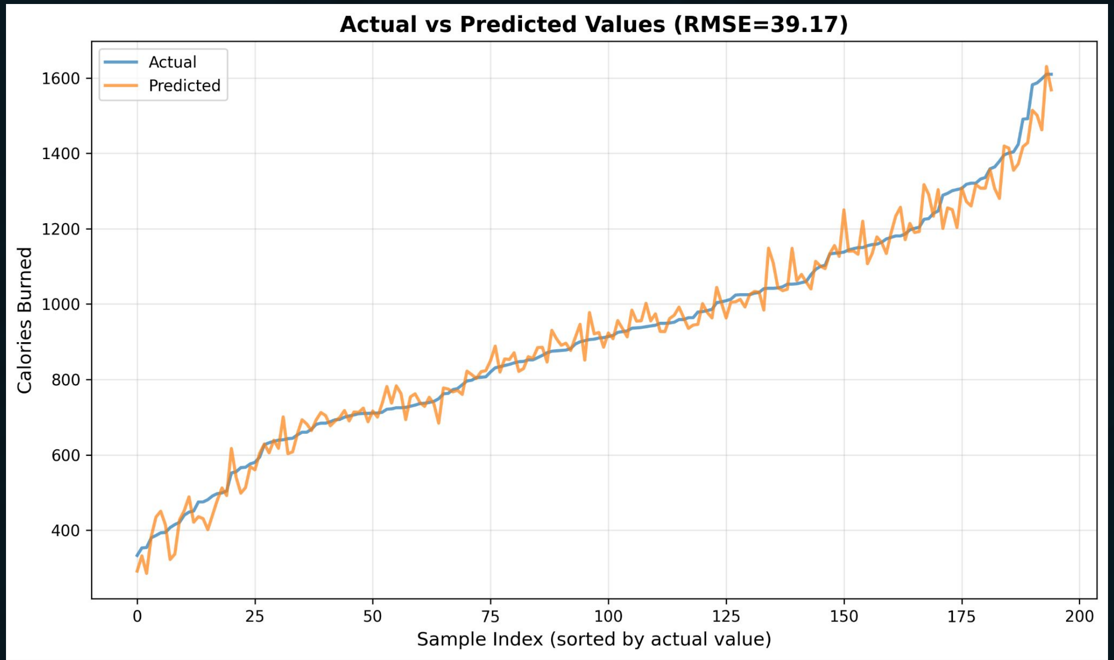
Normal Equation Method:

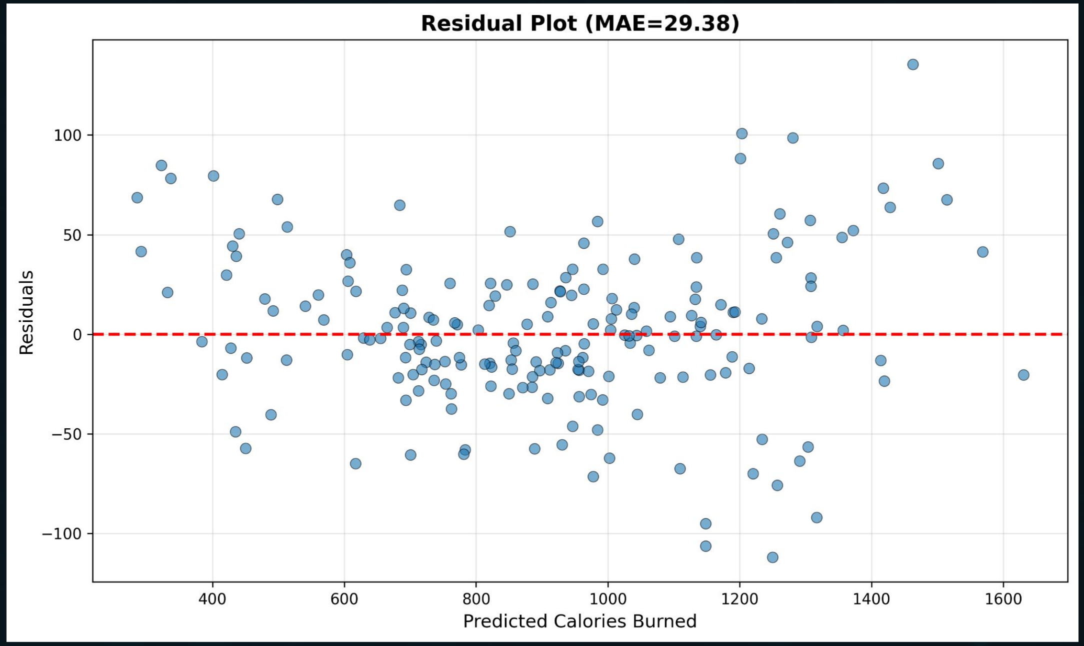
- Formula: $\theta = (X^T X)^{-1} X^T y$
- Where θ = model parameters, X = features, y = target values

Key Advantages:

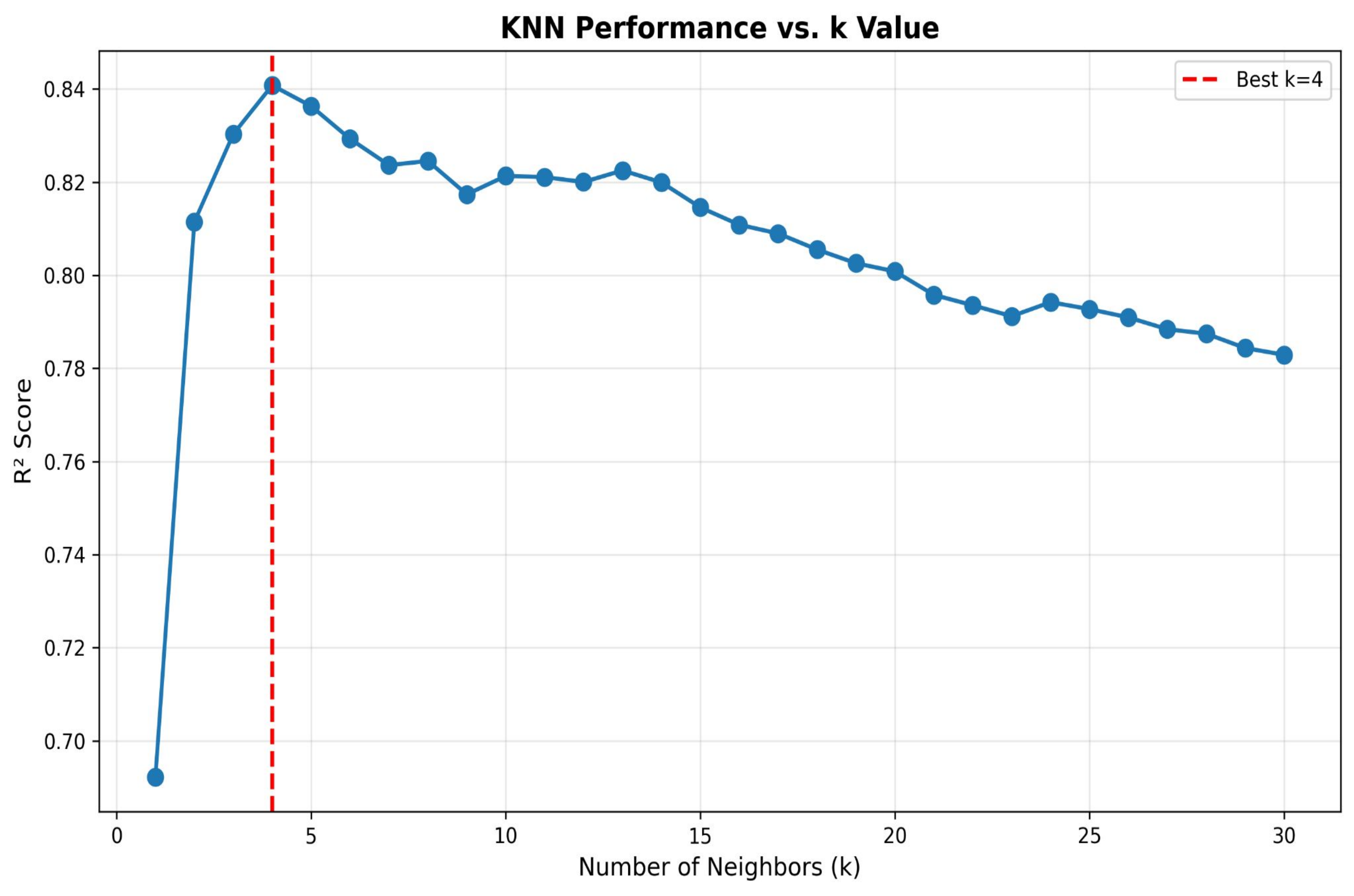
- Closed-form solution (no iterations needed)
- Computes optimal weights directly
- No hyperparameters to tune
- Guaranteed global minimum convergence
- Very fast for moderate-sized datasets







K-Nearest Neighbors Regression Model



K Optimization Process:

- Tested k = 1 to 30 systematically
- Peak performance at **k = 4** ($R^2 = 0.8408$)
- k < 4: Overfitting (too sensitive to noise)
- k > 4: Underfitting (loses local patterns)

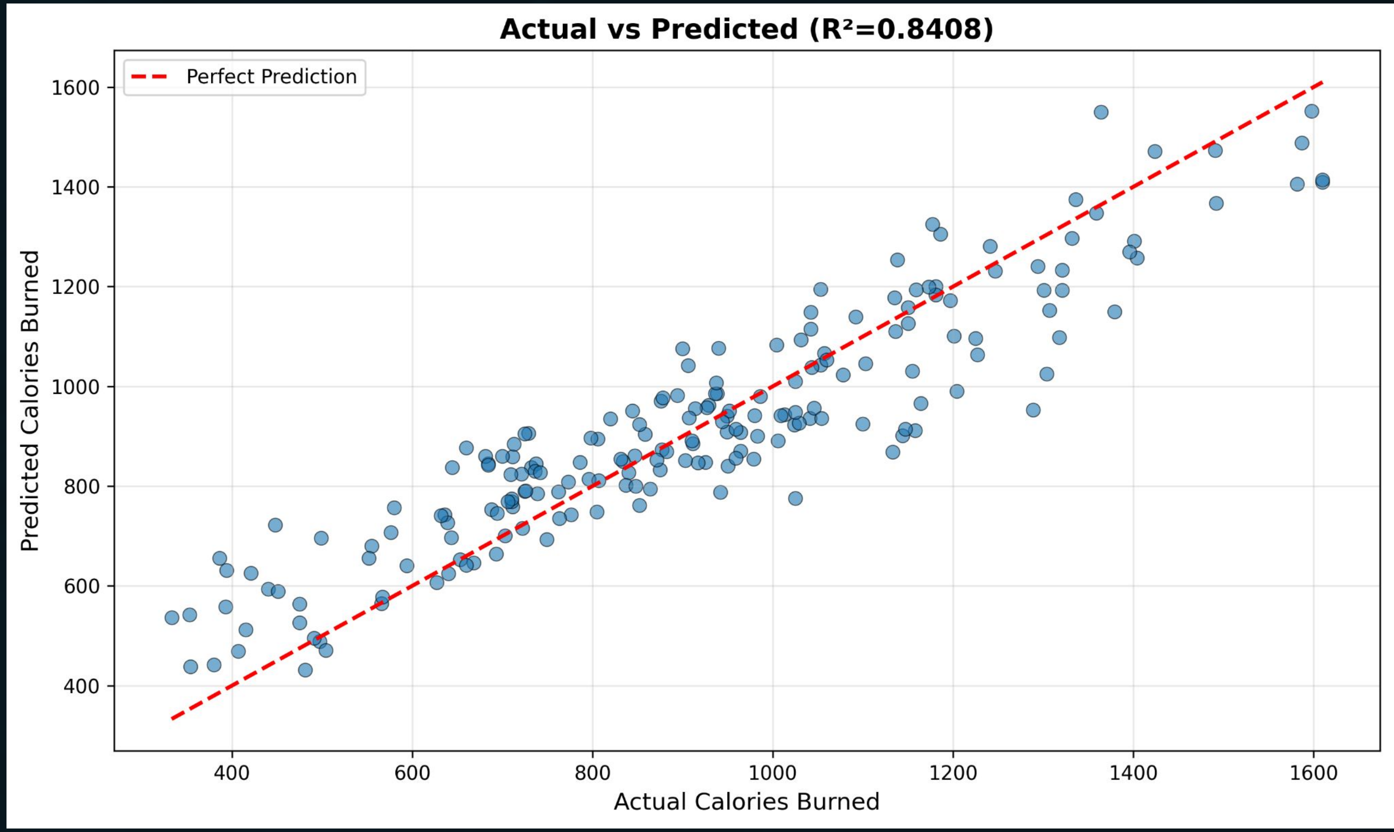
Imagine predicting calories for someone's profile:

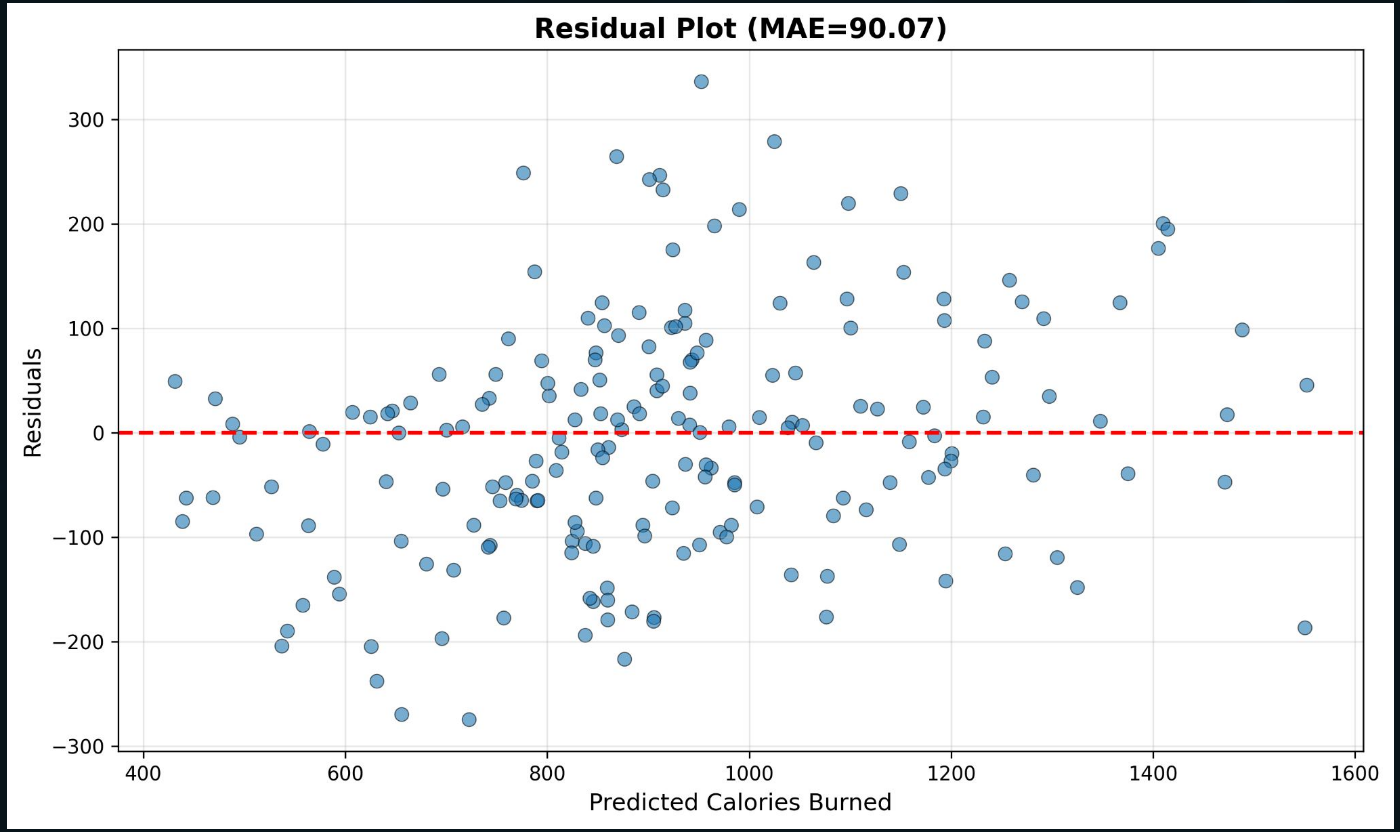
- Age: 25, Weight: 70kg, Workout: 45 min

k=1: "You're exactly like this one person who burned 850 calories"
→ Risky if that person is unusual!

k=4: "You're similar to these 4 people who burned 820, 865, 840, 830 calories. Average = 839"
→ More stable, less affected by outliers!

k=20: "You're kinda like these 20 people ranging from 600-1100 calories. Average = 850"
→ Too generic, loses individual patterns!





Overall Performance: R^2 = 0.8408

Residual Analysis: MAE = 90.07

Points **on the line** = perfect predictions
Points **above the line** = model overestimates calories burned
Points **below the line** = model underestimates calories burned

Strong Linear Relationship: The points cluster nicely around the diagonal line across the entire range (400-1600 calories), showing your model captures the general trend well.

Consistent Performance: The spread of points is fairly uniform across all calorie ranges - the model doesn't perform significantly worse at low vs. high values.

Random Scatter Pattern: The residuals are randomly distributed around zero with no discernible pattern, indicating the model has captured all systematic relationships in the data.

Practical Interpretation: With MAE = 90.07 calories, predictions typically fall within ±90 calories of actual values. For example, if actual burn is 1000 calories, the model predicts 910-1090 calories.

Evaluation Metrics Explained

RMSE (Root Mean Squared Error)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum(y - \hat{y})^2}$$

Penalises larger errors more heavily. Interpretable in original units (calories). Lower values indicate better model performance.

MAE (Mean Absolute Error)

$$\text{MAE} = \frac{1}{n}\sum |y - \hat{y}|$$

Represents the average absolute prediction error. More robust to outliers than RMSE. Lower values indicate better model performance.

R² (Coefficient of Determination)

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Proportion of variance in the dependent variable predictable from the independent variables. Ranges from 0 to 1, with 1 indicating a perfect fit. Higher values are better.

MAPE (Mean Absolute Percentage Error)

$$\text{MAPE} = \frac{100}{n}\sum\left|\frac{y - \hat{y}}{y}\right|$$

Provides a percentage error for business interpretation. Scale-independent, making it useful for comparisons across different datasets. Lower values indicate better model performance.

Adjusted R²

$$\text{Adjusted } R^2 = 1 - \left[\frac{(1 - R^2)(n - 1)}{(n - p - 1)}\right]$$

Adjusts R² for the number of features (p) in the model, preventing overfitting caused by too many predictors. Higher values are better.

Model Performance Results

Model	R^2	RMSE	MAE	MAPE	Adjusted R^2
Linear Regression	0.9813	39.17	29.38	3.58	0.9798
KNN Regression (k=4)	0.8407	114.26	90.06	11.43	0.8284
Gradient Boosting	0.9948	20.47	15.36	1.80	0.9944

Training vs. Test Performance (Overfitting Check)

Linear Regression	0.9804	0.9813	0.0009
KNN Regression	0.8853	0.8407	0.0446
Gradient Boosting	0.9973	0.9949	0.0024

Conclusions

1. **Gradient Boosting is the Premier Performer**

   For this calories prediction task, Gradient Boosting achieved a near-perfect R^2 of 0.9949, demonstrating its superior capability in handling complex datasets.

2. **All Approaches are Viable**

   Each of the three implemented models yielded an R^2 greater than 0.96, confirming that calories burned can be accurately predicted from gym exercise tracking data.

3. **Session Duration is Key**

   Session duration is likely the most critical feature, exhibiting a direct correlation with energy expenditure, thus serving as a strong predictor.

4. **From-Scratch Implementation is Invaluable**

   Building algorithms from their fundamental principles provides a profound understanding of their operational mechanics and underlying mathematics.

5. **Proper Preprocessing is Essential**

   Effective data preprocessing, including scaling for distance-based algorithms, encoding for categorical variables, and robust outlier handling, is crucial for accurate and reliable predictions.