Micro:bit Accelerometer Data Classification

1. Data Recording and Preprocessing

We recorded accelerometer data from the BBC micro:bit following the sequence:

- 5 s steady
- 5 s shake
- 5 s steady
- 5 s shake

This produced **158 samples** with 11 recorded columns (t, ax, ay, az, vx, vy, vz, x, y, z, mag). The average sampling rate was **~105 ms/row**, slightly slower than the intended 100 ms/row but close enough for analysis.

To simplify analysis, we derived the **magnitude of acceleration (mag)** as a key feature for distinguishing motion states.

2. Dataset Summary

- **Shape:** 158 rows × 11 columns
- **Magnitude (mag) stats:**
  - Mean: 7.89
  - Std: 3.08
  - Min: 1.52

- Max: 10.15

Observation: During shaking periods, variance in mag increases noticeably compared to steady periods.

3. Classification Model

We trained a **Logistic Regression classifier** using features [t, ax, ay, az].

The dataset was split into training (112 rows) and test (46 rows).

Results

- **Accuracy:** 36.96%
- **Confusion Matrix:**
- [[0 29]
- [0 17]]
- Interpretation: The model predicted all samples as "shake," leading to 100% recall for class 1 but complete failure on class 0 ("steady").

4. Discussion

- **Model performance is poor.** The logistic regression model struggled to separate the two classes because:
  1. **Class imbalance:** More shake than steady samples in the training set.

2. **Feature selection:** Using raw accelerometer axes (ax, ay, az) and time is insufficient.
3. **Small dataset:** Only ~160 rows makes the model prone to bias.

- **Coefficient analysis:** Time (t) dominated the decision boundary (coef = 2.47), showing the model may be overfitting to sequence order rather than motion.

5. Recommendations for Improvement

1. **Balanced labeling:** Ensure equal samples for "steady" and "shake."
2. **Feature engineering:**
    - Use **standard deviation or variance of mag** in a rolling window (1 s).
    - Add **RMS, peak-to-peak range, and FFT energy** as features.
3. **Algorithm choice:** Try tree-based models (Decision Tree, Random Forest) or even a simple **rule-based threshold** (e.g., if std(mag) > 0.5 → "shake").
4. **More data:** Repeat the recording multiple times for robustness.
5. **Smoothing:** Apply a low-pass filter to reduce noise before feature extraction.

6. Conclusion

The initial logistic regression model achieved **only 36.9% accuracy**, misclassifying all "steady" samples.

This experiment shows the importance of feature engineering and dataset balancing. With additional features like rolling standard deviation of acceleration magnitude, the classification can likely reach **>90% accuracy**, as shaking produces a much higher variance signal than holding steady.

Time Comparison:

Experiment 1: 16.37 seconds

Experiment 2: 23 seconds