Human Activity Recognition using Hidden Markov Models

1. Background and Motivation

Human activity recognition (HAR) has become increasingly important in various domains, including health monitoring, fitness tracking, and smart home applications. Our use case focuses on developing a reliable system to automatically detect and classify common human activities using smartphone sensor data. The ability to accurately recognize activities such as standing, walking, jumping, and remaining still has practical applications in fitness applications, fall detection systems for elderly care, and context-aware mobile computing. By leveraging the ubiquitous nature of smartphones equipped with accelerometers and gyroscopes, we can create an accessible and cost-effective solution for continuous activity monitoring without requiring specialized hardware.

2. Data Collection and Preprocessing

2.1 Data Collection

We collected sensor data using smartphone accelerometer and gyroscope sensors at a sampling frequency of 100 Hz. Four distinct activities were recorded:

- Standing: Phone held steady at waist level (5-10 seconds per session)
- Walking: Consistent pace movement (5-10 seconds per session)
- Jumping: Continuous jumping motions (5-10 seconds per session)
- Still: Phone placed on a flat surface (5-10 seconds per session)

A total of 102 sessions were collected across all activities, with a minimum of 1 minute 30 seconds of data per activity type. The data was organized into labeled folders and stored in CSV format with the following structure:

seconds_elapsed, ax, ay, az, gx, gy, gz

2.2 Preprocessing Steps

Windowing: The continuous sensor data was segmented using a sliding window approach:

- Window size: 1.0 second (100 samples at 100 Hz)
- Step size: 0.5 seconds (50% overlap)
- This resulted in 1,270 total windows across all sessions

Feature Extraction: For each window, we extracted comprehensive time-domain and frequency-domain features:

Time-domain features (per axis):

- Mean, standard deviation, variance
- Root Mean Square (RMS)
- Peak-to-peak amplitude
- Signal Magnitude Area (SMA) for accelerometer
- Correlation coefficients between accelerometer axes

Frequency-domain features (per axis):

- Dominant frequency (via FFT)
- Spectral energy

This resulted in 48 features per window before dimensionality reduction.

Normalization and Dimensionality Reduction:

- Z-score normalization was applied across the entire dataset
- Principal Component Analysis (PCA) reduced features to 10 principal components
- This preserved over 95% of variance while reducing computational complexity

3. HMM Setup and Implementation

3.1 Model Architecture

We implemented a Gaussian Hidden Markov Model with the following specifications:

Model Parameters:

- Number of hidden states: 4 (corresponding to our 4 activities)
- Observation space: 10-dimensional (post-PCA features)
- Covariance type: Full covariance matrices
- Maximum EM iterations: 200
- Random state: 42 (for reproducibility)

Key Components:

- Hidden States (Z): [Standing, Walking, Jumping, Still]
- Observations (X): 10-dimensional feature vectors
- Transition Matrix (A): P(state_t | state_{t-1})
- Emission Model (B): Gaussian distributions with full covariance
- Initial Probabilities (π): Starting state distribution

3.2 Training Algorithm

The model was trained using the Baum-Welch (Expectation-Maximization) algorithm:

- Iteratively refined transition and emission probabilities
- Converged when log-likelihood change < ε
- Utilized the forward-backward algorithm internally
- All 102 sessions (1,270 windows) were used for training

3.3 Decoding Algorithm

We implemented a custom Viterbi algorithm for optimal state sequence decoding:

- Used log-space computations for numerical stability
- Dynamic programming approach with $O(T \cdot N^2)$ complexity
- Found most likely state sequence given observations

3.4 State Mapping

Since HMM states are arbitrary indices (0-3), we mapped them to activity labels using majority voting:

```
python
state_to_activity = {
    0: 'stand',
    1: 'jump',
    2: 'jump',
    3: 'still'
}
```

The model achieved an overall accuracy of **78%** on the training data, with the following classification report:

```
          precision    recall  f1-score   support
   jump      0.88       0.99     0.94       385
  stand      0.55       1.00     0.71       275
  still      1.00       1.00     1.00       336

walking      0.00       0.00     0.00       274
```

4.2 Activity-Specific Performance

Still Activity (Best Performance):

- Sensitivity: 100%
- Specificity: 100%
- Overall Accuracy: 100%

The perfect classification of the "still" state is expected, as this activity has the most distinctive sensor signature with minimal motion across all axes.

Jumping Activity (Strong Performance):

- Sensitivity: 99.5%
- Specificity: 94.2%
- Overall Accuracy: 95.8%

The high accuracy for jumping is due to its unique motion pattern characterized by significant vertical acceleration and distinct frequency components.

Standing Activity (Moderate Performance):

- Sensitivity: 99.6%
- Specificity: 77.3%
- Overall Accuracy: 82.1%

Standing shows high sensitivity but lower specificity, indicating confusion with other activities, particularly with "still" when the phone is stationary.

Walking Activity (Poor Performance):

- Sensitivity: 0%
- Specificity: 100%
- Overall Accuracy: 78.4%

The complete failure to detect walking is the most significant limitation. This occurred because the HMM states were not properly discriminating walking from other activities during training.

4.3 Confusion Matrix Analysis

The confusion matrix reveals specific misclassification patterns:

- 50 walking instances were misclassified as jumping
- 224 walking instances were misclassified as standing
- Minimal confusion between other activity pairs

This suggests that walking motion patterns share characteristics with both standing (during slower phases) and jumping (during more dynamic phases).

4.4 Transition Matrix

The learned transition probabilities show reasonable activity sequences with high self-transition probabilities, indicating stable states. The model learned realistic patterns of how activities typically flow into each other.

4.5 Model Generalization

Training Set Evaluation: The results above are based on decoding the same data used for training, which is a significant limitation.

Generalization Assessment: To properly evaluate generalization:

- Approach: We would need to collect separate test sessions not used during training
- Participants: Data from different individuals to test person-independence
- Environments: Different locations and conditions
- Expected Performance: We anticipate degradation in accuracy due to:

- Individual variation in movement patterns
- Different phone placements and orientations
- Environmental factors (terrain, temperature)

Recommended Evaluation Protocol:

1. Reserve 20-30% of sessions for testing (held-out validation)
2. Implement k-fold cross-validation across sessions
3. Test with data from new participants
4. Evaluate robustness to different phone positions

Without a proper train/test split, the reported 78% accuracy represents an optimistic upper bound on performance.

5. Discussion and Conclusion

5.1 Strengths

1. Robust Feature Engineering: The combination of time and frequency domain features effectively captures motion characteristics
2. Excellent Performance on Distinct Activities: The model performs exceptionally well on activities with unique signatures (still, jumping)
3. Computational Efficiency: PCA dimensionality reduction enables real-time processing
4. Probabilistic Framework: HMM naturally models temporal dependencies between activities

5.2 Limitations

1. Walking Recognition Failure: The model completely failed to recognize walking, likely due to:
   - Insufficient state separation during training
   - Possible confusion between walking and transition periods
   - Need for more discriminative features or better state initialization
2. Training/Testing Methodology: Using the same data for training and testing provides inflated accuracy estimates
3. Limited Generalization Assessment: No evaluation on truly unseen data from different sources:
   - Participants (person-dependent model)
   - Phone positions (assumes consistent placement)
   - Environmental conditions
4. Class Imbalance: Uneven distribution of activities may bias the model

5.3 Future Improvements

Data Collection:

- Increase sample size to 200+ recordings
- Include multiple participants for diversity
- Explicitly record transition periods between activities

- Collect data in various environments

Feature Engineering:

- Add frequency band energy ratios
- Include entropy measures
- Implement wavelet transform features
- Explore cross-sensor correlations (accelerometer-gyroscope)

Model Enhancements:

- Initialize HMM states using k-means clustering for better separation
- Experiment with different covariance structures
- Implement left-right HMM topology constraints
- Consider a hierarchical HMM for sub-activities

Advanced Approaches:

- Compare with deep learning models (LSTM/GRU)
- Implement semi-supervised learning with unlabeled data
- Develop online learning for adaptive models
- Explore multi-modal sensor fusion

5.4 Conclusion

This project successfully demonstrated the application of Hidden Markov Models for human activity recognition using smartphone sensors. The model achieved strong performance on distinctive activities (still: 100%, jumping: 95.8%) but struggled with walking recognition (0%). The 78% overall accuracy, while promising, should be interpreted cautiously due to the lack of proper train/test separation.

The key insight is that HMMs can effectively model temporal patterns in sensor data when activities have sufficiently distinct signatures. However, the walking recognition failure highlights the importance of proper feature engineering, state initialization, and the need for larger, more diverse training datasets.

For practical deployment, we recommend:

1. Collecting significantly more data with proper train/test split
2. Implementing better state initialization strategies
3. Adding more discriminative features for walking detection
4. Validating on truly unseen data from new participants
5. Considering hybrid approaches that combine HMMs with deep learning

Despite current limitations, this work provides a solid foundation for developing robust activity recognition systems and demonstrates both the potential and challenges of applying probabilistic models to real-world sensor data.