Midsemester Project Report: Predictive Modelling of Signal Strength and Handover Events Using Machine Learning

Project Title: Predictive Modelling of Signal Strength and Handover Events Using ML Techniques

Author: Ashutosh Borthakur (Group 17)

Date: November 2025

Executive Summary

This report presents a comprehensive analysis of a machine learning project focused on predicting mobile network handover events using signal strength measurements and spatial location data. The project implements the Okumura-Hata propagation model to generate synthetic datasets and employs two classification algorithms—Random Forest and Logistic Regression—to predict handover occurrences. The analysis reveals that Random Forest achieves superior performance with 86.21% accuracy and an F1score of 0.60, compared to Logistic Regression's 81.03% accuracy and F1-score of 0.15. However, both models struggle with the minority class (handover events) due to significant class imbalance, highlighting the critical need for advanced techniques like SMOTE to improve recall and overall predictive capability.

1. Introduction and Background

1.1 Problem Statement and Motivation

Mobile network handovers represent critical transition events where active connections are transferred from one base station to another as users move through coverage areas. Poor handover decisions can lead to dropped calls, degraded data speeds, and excessive battery consumption. Traditional handover mechanisms rely on reactive, rule-based approaches that use simple signal strength thresholds, which often result in suboptimal decisions in complex and dynamic network environment. Machine learning offers predictive and adaptive solutions that can anticipate handovers before they occur, potentially reducing unnecessary transitions and improving overall network performance.

The motivation behind this project stems from the growing complexity of modern wireless networks, particularly with the expansion of 5G infrastructure and the proliferation of mobile devices globally. By leveraging historical data patterns and sophisticated learning algorithms, predictive handover models can enable proactive network management, optimize resource allocation, and enhance quality of service for end users. This approach represents a paradigm shift from reactive to proactive network management, where decisions are informed by learned patterns rather than static rule.

1.2 Research Objectives and Scope

The primary objectives of this project are multifaceted. First, the project aims to develop a realistic simulation environment using the Okumura-Hata propagation model to generate synthetic signal strength data from multiple base stations as mobile users traverse a defined geographic area. Second, it seeks to train and evaluate machine learning classification models—specifically Random Forest and Logistic Regression—to predict handover events based on Received Signal Strength Indicator (RSSI) values and user location coordinates. Third, the project addresses the challenge of class imbalance inherent in handover prediction, where the majority of data points represent non-handover scenarios, by exploring techniques such as SMOTE and class weighting.

The scope of the work encompasses dataset generation through MATLAB-based simulation, feature engineering from spatial and signal measurements, model training and evaluation using standard machine learning metrics, and comparative performance analysis between different algorithms. The project also acknowledges the potential for future extension through real-world data collection using Android-based signal tracking applications, though the current implementation focuses primarily on synthetic data generation.

1.3 Literature Context and Related Work

This project builds upon recent advances in machine learning-based handover optimization for wireless networks. Masri et al. (2021) proposed machine-learning-based predictive handover strategies specifically designed for 5G networks, demonstrating the viability of data-driven approaches in next-generation mobile systems. Zhang et al. (2023) developed a handover failure prediction framework using XGBoost classifiers, achieving significant improvements in handover success rates for 5G consumer communications. Additionally, Khan et al. (2021) demonstrated that machine learning schemes for handover prediction in Wi-Fi networks could reduce unnecessary handovers by up to 60% compared to traditional received signal strength methods.

The adaptation of propagation models for specific environments also plays a crucial role in signal strength prediction. Osei et al. (2019) adapted the Okumura-Hata model to urban environments in Accra, providing valuable insights into the customization of empirical path loss models for different geographical and morphological settings. These studies collectively establish a foundation for understanding how machine learning can enhance handover decision-making through improved prediction accuracy and reduced computational overhead.

2. Methodology and Technical Implementation

2.1 Okumura-Hata Propagation Model

The Okumura-Hata model serves as the cornerstone of the signal strength simulation in this project. This empirical propagation model, originally developed from extensive measurements in Tokyo, Japan, provides a computationally efficient method for estimating path loss in urban and suburban mobile communication environments. The model is particularly well-suited for frequencies between 150 MHz and 1500 MHz, base station antenna heights of 30-200 meters, mobile antenna heights of 1-10 meters, and link distances ranging from 1 to 20 kilometers.

The basic path loss formula implemented in the MATLAB simulation code follows the urban environment formulation:

$$L_{path} = 69.55 + 26.16\log_{10}(f) - 13.82\log_{10}(h_b) - a(h_m) + [44.9 - 6.55\log_{10}(h_b)]\log_{10}(d)$$

where f represents the carrier frequency in MHz, h_b is the base station antenna height in meters, h_m denotes the mobile antenna height in meters, d is the distance between transmitter and receiver in kilometers, and $a(h_m)$ is the mobile antenna height correction factor.

For the mobile antenna height correction factor in medium to small cities, the model use:

$$a(h_m) = (1.1\log_{10}(f) - 0.7)h_m - (1.56\log_{10}(f) - 0.8)$$

The simulation parameters were configured as follows: carrier frequency of 900 MHz, base station antenna height of 30 meters, mobile antenna height of 1.5 meters, and a transmit power of 43 dBm (equivalent to approximately 20 watts). These parameters align with typical GSM network specifications and fall within the valid operational ranges of the Okumura-Hata model.

The received signal strength indicator (RSSI) at each mobile position is calculated by subtracting the path loss from the transmit pow:

$$RSSI(dBm) = P_{tx}(dBm) - L_{path}(dB)$$

This formulation provides realistic signal strength values that typically range from 0 dBm (very strong signal near the transmitter) to -100 dBm or lower (weak signal at distant locations). In telecommunications practice, RSSI values of -70 dBm or higher indicate excellent signal strength, while values below -100 dBm suggest poor connectivity with potential connection loss.

2.2 Synthetic Dataset Generation

The dataset generation process implements a systematic grid-based simulation approach using MATLAB. A two-dimensional coverage area of 2 km × 2 km was discretized into a 14×14 grid, yielding 196 distinct user positions representing potential mobile device locations throughout the coverage are. Four base stations were randomly deployed within this area, with their positions fixed using a random seed for reproducibility.

For each of the 196 user positions, the simulation computes the Euclidean distance to all four base stations and applies the Okumura-Hata path loss formula to calculate the corresponding RSSI values. A minimum distance threshold of 0.01 km (10 meters) prevents logarithmic singularities when users are extremely close to base stations. The mobile device is assumed to connect to the base station providing the strongest RSSI at each position, implementing a simple yet realistic "strongest signal first" handover policy commonly used in mobile network.

Handover events are identified through temporal analysis of base station connectivity as the user progresses through sequential grid position. When the connected base station identifier changes between consecutive positions, a handover event is flagged with a binary label of 1; otherwise, the label remains 0. This sequential processing generates a realistic pattern of handover occurrences concentrated near cell boundaries where signal strengths from multiple towers are comparable.

The resulting dataset comprises 196 observations with eight features: X-coordinate (X_km), Y-coordinate (Y_km), connected base station ID (ConnectedBS), handover indicator (Handover), and four RSSI measurements from each base station (RSSI_BS1 through RSSI_BS. This feature set captures both spatial information and multi-tower signal measurements, providing comprehensive input for machine learning models to learn complex handover patterns.

2.3 Feature Engineering and Data Preprocessing

The machine learning pipeline extracts six input features from the generated dataset for model training. These features consist of the user's spatial coordinates (X_km and Y_km) and the four RSSI measurements from all base stations (RSSI_BS1, RSSI_BS2, RSSI_BS3, RSSI_BS4). The inclusion of all RSSI values, not just the strongest signal, enables the model to capture relative signal strength differences and detect potential handover candidates based on signal quality comparisons across multiple towers.

The dataset is partitioned into training and testing subsets using a 70-30 stratified split, implemented through MATLAB's cvpartition function. This configuration allocates 138 observations (70.4%) for model training and 58 observations (29.6%) for independent performance evaluation. Stratified partitioning ensures that the class distribution in both subsets reflects the original dataset proportions, maintaining consistent representation of the minority handover class.

No explicit feature scaling or normalization was applied in the documented implementation. While Logistic Regression typically benefits from feature standardization due to its reliance on gradient-based optimization, Random Forest algorithms are inherently invariant to monotonic feature transformations and can effectively handle features with different scales. The RSSI measurements naturally share similar units (dBm) and magnitude ranges, reducing the urgency for normalization, though standardization could potentially improve Logistic Regression convergence speed and numerical stability.

2.4 Machine Learning Algorithms

2.4.1 Logistic Regression

Logistic Regression was implemented as a baseline binary classification model using MATLAB's fitglm function with binomial distribution and logit link function. This generalized linear model estimates the probability of a handover event as a logistic function of the input feature:

$$P(Handover = 1|X) = \frac{1}{1 + e^{-(\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_6 X_6)}}$$

where $\beta_0, \beta_1, \ldots, \beta_6$ represent model coefficients learned through maximum likelihood estimation. Classification decisions are made by applying a threshold of 0.5 to the predicted probabilities, with values exceeding this threshold classified as handover events.

Logistic Regression offers several advantages as a baseline model. It provides interpretability through coefficient weights that indicate feature importance and direction of influence. The model trains quickly with minimal computational overhead, making it suitable for rapid prototyping and deployment in resource-constrained environments. Additionally, it outputs calibrated probability scores that can be adjusted using different decision thresholds to balance precision and recall according to operational requirements.

However, Logistic Regression assumes linear relationships between features and log-odds of the target variable, limiting its capacity to capture complex non-linear patterns and feature interactions inherent in spatial signal propagation. This linear constraint may explain the model's relatively poor performance on handover prediction, particularly its extremely low recall of 0.09, indicating failure to detect most true handover events.

2.4.2 Random Forest Classifier

Random Forest was implemented using MATLAB's TreeBagger function configured with 100 decision trees. This ensemble learning method constructs multiple decision trees during training, with each tree built on a bootstrapped sample of the training data. For each node split within a tree, the algorithm randomly selects a subset of features (in this case, all features due to the 'NumPredictorsToSample' parameter set to 'all') and determines the optimal split criterion that maximizes information gain.

The final classification decision is made through majority voting across all trees in the forest. Out-of-bag (OOB) prediction was enabled to provide an internal estimate of generalization error without requiring a separate validation set. Feature importance was computed using permutation-based importance measures, which assess the increase in prediction error when each feature's values are randomly shuffl.

Random Forest offers substantial advantages for handover prediction. It naturally handles non-linear relationships and complex feature interactions without requiring manual feature engineering or transformation. The ensemble averaging mechanism reduces overfitting risk compared to single decision trees, providing robust predictions across diverse data distribution. Additionally, the algorithm is relatively insensitive to feature scaling and can accommodate both numerical and categorical variables seamlessly.

The bootstrapping and feature randomization introduce diversity among individual trees, enabling the ensemble to capture multiple perspectives on the decision boundary. This diversity is particularly valuable when dealing with class imbalance, as different trees may focus on different aspects of the minority class patterns. The model achieved significantly better performance than Logistic Regression across all evaluation metrics, demonstrating Random Forest's superior capacity for learning complex handover patterns from spatial and signal strength features.

2.5 Class Imbalance Handling

Class imbalance represents a critical challenge in this handover prediction task. Analysis of the test set reveals that 81.0% of observations belong to the "No Handover" class (47 samples), while only 19.0% represent actual handover events (11 samples). This imbalance ratio of approximately 4.27:1 reflects realistic mobile network behavior, where users spend most of their time within stable coverage zones and only occasionally cross cell boundaries that trigger handovers.

Imbalanced class distributions pose significant problems for machine learning algorithms. Standard classification models trained on such data tend to develop a bias toward the majority class, optimizing overall accuracy by correctly predicting the prevalent class while largely ignoring the minority class. This bias is particularly problematic in handover prediction, where correctly identifying actual handover events (the minority class) is operationally more important than correctly classifying nonhandover positions.

The project documentation mentions plans to employ the Synthetic Minority Over-sampling Technique (SMOTE) and class weighting to address this imbalance. SMOTE generates synthetic minority class samples by interpolating between existing minority instances and their nearest neighbors in the feature space. This approach increases minority class representation without simple duplication, potentially helping models learn more generalizable decision boundaries.

SMOTE operates by selecting a minority class instance, identifying its k nearest neighbors, and generating synthetic samples along the line segments connecting the instance to its neighbors. The oversampling ratio controls the number of synthetic samples created, typically aiming to balance class distributions or achieve a target minority-to-majority ratio. However, research indicates that SMOTE's effectiveness can be limited in high-dimensional datasets or scenarios with low signal-to-noise ratio. For the current six-dimensional feature space, SMOTE should theoretically provide benefits, though its impact on the final model performance has not yet been empirically validated in the documented result.

Class weighting offers an alternative approach where the learning algorithm assigns higher misclassification penalties to minority class errors. This encourages the model to pay more attention to minority samples during training without modifying the dataset itself. Many algorithms, including Logistic Regression and Random Forest, support class weighting parameters that can be tuned to balance the trade-off between precision and recall according to application requirements.

3. Results and Performance Analysis

3.1 Model Performance Metrics

The models were evaluated using standard binary classification metrics: accuracy, precision, recall, and F1-score. These metrics provide complementary perspectives on model performance, particularly important when dealing with imbalanced datasets where accuracy alone can be misleading.

Accuracy measures the proportion of correct predictions (both true positives and true negatives) out of all predictions.
While intuitive, accuracy can be deceptive in imbalanced scenarios, as a naive classifier that always predicts the majority class would achieve 81% accuracy on this dataset simply by never predicting handovers.

Precision quantifies the proportion of predicted handover events that are actually true handovers:

$$Precision = \frac{TP}{TP + FP}$$

where TP represents true positives and FP denotes false positives. High precision indicates that when the model predicts a handover, it is usually correct, minimizing unnecessary handover preparations.

Recall (also called sensitivity or true positive rate) measures the proportion of actual handover events that the model successfully identifies:

$$Recall = \frac{TP}{TP + FN}$$

where FN represents false negatives. High recall is critical for handover prediction, as missing actual handover events (false negatives) can lead to service disruptions and degraded user experience.

F1-score combines precision and recall into a single metric using their harmonic mean:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

The harmonic mean ensures that both precision and recall must be reasonably high for a good F1-score, making it particularly suitable for imbalanced classification evaluation.

3.2 Logistic Regression Performance

Logistic Regression achieved an accuracy of 81.03% on the test set. The confusion matrix reveals the following classification outcomes:

- True Negatives (TN): 46 - correctly predicted non-handover cases
- False Positives (FP): 1 - incorrectly predicted handover when none occurred
- False Negatives (FN): 10 - missed actual handover events
- True Positives (TP): 1 - correctly predicted handover cases

From these values, the performance metrics are calculated as]:

- **Precision**: 0.50 - only half of the predicted handovers were correct
- **Recall**: 0.09 - the model detected only 9% of actual handover events
- **F1-Score**: 0.15 - poor overall balance between precision and recall

The extremely low recall of 0.09 represents the most critical weakness of the Logistic Regression model. Out of 11 actual handover events in the test set, the model correctly identified only 1, missing 10 handover occurrences. This severe underdetection of the minority class likely stems from the model's linear decision boundary and bias toward the majority class due to class imbalance.

The model's high number of true negatives (46 out of 47 non-handover cases correctly classified) contributes to the relatively acceptable overall accuracy of 81.03%. However, this accuracy is misleading, as a trivial baseline that always predicts "no handover" would achieve 81.0% accuracy simply due to class distribution. The Logistic Regression model provides minimal improvement over this baseline, demonstrating limited practical utility for handover prediction in its current form.

3.3 Random Forest Performance

Random Forest demonstrated substantially better performance across all metrics. The confusion matrix shows:

- True Negatives (TN): 44 - correctly predicted non-handover cases
- False Positives (FP): 3 - incorrectly predicted handover when none occurred
- False Negatives (FN): 5 - missed actual handover events
- True Positives (TP): 6 - correctly predicted handover cases

The resulting performance metrics are:

- **Accuracy**: 86.21% - correctly classified 50 out of 58 test samples
- **Precision**: 0.67 - two-thirds of predicted handovers were correct
- **Recall**: 0.55 - the model detected 55% of actual handover events
- **F1-Score**: 0.60 - better balance between precision and recall

Random Forest's recall of 0.55 represents a dramatic improvement over Logistic Regression's 0.09, detecting 6 out of 11 actual handover events compared to just 1 for Logistic Regression. This sixfold increase in true positive detections demonstrates Random Forest's superior ability to learn the complex patterns distinguishing handover from non-handover scenarios.

The precision of 0.67 indicates that while the model generates some false alarms (3 false positives), the majority of its handover predictions are accurat. In operational contexts, false positives may be more tolerable than false negatives, as unnecessarily preparing for a handover incurs lower costs than missing an actual handover that could disrupt service. The F1-score of 0.60 reflects a reasonable balance between precision and recall, though significant room for improvement remains.

Comparing Random Forest to Logistic Regression reveals consistent superiority across all metrics. Random Forest achieves 5.18 percentage points higher accuracy (86.21% vs 81.03%), 17 percentage points higher precision (0.67 vs 0.50), 46 percentage points higher recall (0.55 vs 0.09), and 45 percentage points higher F1-score (0.60 vs 0.15). These substantial improvements validate the choice of Random Forest for handover prediction and align with broader literature findings that ensemble methods often outperform linear models on complex classification tasks.

3.4 Confusion Matrix Analysis

The confusion matrices provide detailed insights into model behavior and error patterns. For Logistic Regression, the extreme imbalance between true negatives (46) and true positives (1) reveals severe bias toward the majority class. The model correctly identifies nearly all non-handover cases but fails catastrophically on handover detection, with false negatives (10) outnumbering true positives by a factor of 10.

Random Forest's confusion matrix shows better-balanced performance. While true negatives (44) still outnumber true positives (6), reflecting the inherent class imbalance, the model achieves a more reasonable true positive count. The five false negatives represent missed handover opportunities, but this is considerably better than Logistic Regression's ten false negatives. The three false positives indicate occasional over-prediction of handovers, which may be acceptable depending on the operational cost of unnecessary handover preparation.

Error analysis suggests that both models struggle most with handover events occurring in ambiguous regions where signal strengths from multiple base stations are similar. These boundary zones present inherently difficult classification challenges, as small variations in position or signal quality can determine whether a handover occurs. The models may benefit from additional features such as signal quality metrics (RSRQ, SINR), historical user trajectory information, or tower load statistics to better disambiguate these challenging cases.

3.5 Comparative Analysis with Literature

The observed performance aligns with patterns reported in the machine learning literature comparing Random Forest and Logistic Regression for binary classification. A large-scale benchmarking study by Couronné et al. (2018) found that Random Forest outperformed Logistic Regression in approximately 69% of 243 real datasets, with mean improvements of 0.029 in accuracy and 0.041 in AUC. Our results showing 5.18 percentage point accuracy improvement fall within this expected range.

Research on handover prediction specifically demonstrates that machine learning approaches can achieve substantial performance gains over traditional methods[5] [6] [7]. Khan et al. (2021) reported that their ML-based handover prediction scheme reduced unnecessary handovers by 60% compared to RSSI-based methods[6]. While our current models have not yet achieved such dramatic improvements, the results represent early-stage baseline performance with considerable room for optimization through hyperparameter tuning, feature engineering, and class imbalance mitigation.

Studies addressing class imbalance through SMOTE and other techniques report varying degrees of success depending on data dimensionality and signal-to-noise ratios. In low-dimensional settings similar to our six-feature space, SMOTE typically provides measurable improvements in minority class recall. Implementation of SMOTE in this project, as planned in the documentation, would likely boost recall for both models, potentially bringing Random Forest recall above 0.70 and Logistic
Regression recall into a more respectable range above 0.30.

4. Technical Strengths and Limitations

4.1 Strengths of the Approach

The project demonstrates several notable technical strengths. First, the implementation of the Okumura-Hata propagation model provides a physically realistic foundation for signal strength simulation. Unlike arbitrary synthetic data generators, this approach grounds the dataset in empirically validated radio propagation principles, ensuring that the simulated RSSI values reflect actual mobile network behavior.

Second, the choice of Random Forest and Logistic Regression as complementary algorithms enables meaningful performance comparison between a simple linear baseline and a sophisticated ensemble method. This comparative approach helps establish baseline expectations and quantify the benefits of algorithm complexity, providing actionable insights for model selection in production deployments.

Third, the comprehensive evaluation using multiple metrics beyond accuracy demonstrates methodological rigor. By reporting precision, recall, and F1-score alongside confusion matrices, the analysis provides a complete picture of model behavior, particularly important given the class imbalance present in the dataset.

Fourth, the MATLAB implementation with code documentation facilitates reproducibility and potential extension by other researchers. The systematic parameter specification, random seed usage for base station placement, and clear variable naming conventions enable others to replicate the simulation and build upon this work.

4.2 Limitations and Challenges

Despite these strengths, several limitations constrain the current implementation's effectiveness. The small dataset size of 196 samples represents a significant constraint. Modern machine learning systems typically require thousands to millions of training examples to achieve optimal performance, particularly for complex models like Random Forest. The limited sample size may prevent the models from learning generalizable patterns, instead capturing dataset-specific idiosyncrasies that do not transfer to real-world scenarios.

The severe class imbalance with a 4.27:1 ratio between non-handover and handover cases substantially impairs model learning.
While this imbalance reflects realistic network behavior, it biases algorithms toward the majority class and reduces sensitivity to minority class patterns. Although SMOTE implementation is planned, it has not yet been applied in the documented results, leaving this critical issue unaddressed.

The reliance on purely synthetic data generated from a single propagation model limits ecological validity. Real-world mobile networks experience complex phenomena not captured by the Okumura-Hata model, including multipath fading, shadowing from buildings and terrain, dynamic interference patterns, and time-varying load distributions across cells. Models trained exclusively on synthetic data may fail to generalize when deployed in production environments with these additional complexities.

The feature set, while capturing essential spatial and signal information, omits potentially valuable predictors. Signal quality metrics such as RSRQ (Reference Signal Received Quality) and SINR (Signal-to-Interference-plus-Noise Ratio) provide complementary information beyond raw RSSI values. User mobility features such as velocity and trajectory direction could help predict imminent handovers before they occur. Network load indicators and historical handover patterns might also improve prediction accuracy.

The absence of hyperparameter optimization represents another limitation. Random Forest performance depends critically on parameters such as the number of trees, maximum tree depth, minimum samples per leaf, and feature sampling strategy. Logistic Regression can benefit from regularization parameter tuning to prevent overfitting. Systematic hyperparameter search using cross-validation could yield substantial performance improvements beyond the default parameter settings used in the current implementation.

4.3 Methodological Considerations

The 70-30 train-test split provides reasonable separation between training and evaluation data, but the small overall sample size means the test set contains only 58 observations. This limited test set size increases uncertainty in performance estimates, as a few additional correct or incorrect predictions could swing metrics substantially. Cross-validation techniques such as k-fold cross-validation would provide more robust performance estimates by averaging results across multiple train-test splits.

The grid-based user position generation creates a regular sampling pattern that may not reflect realistic user distributions. In actual mobile networks, users concentrate in high-traffic areas such as urban centers, transportation corridors, and commercial zones.

A more sophisticated sampling strategy incorporating realistic user density patterns and mobility models could enhance dataset realism.

The binary handover classification formulation simplifies the prediction task but may not capture the full complexity of handover decision-making. In practice, handover decisions involve multi-criteria optimization considering not only signal strength but also available capacity, network load, user service requirements, and mobility patterns. Extending the model to predict handover timing or target cell selection would provide greater operational utility.

5. Future Directions and Recommendations

5.1 Dataset Enhancement

Several strategies could substantially improve dataset quality and model performance. First, expanding the dataset size through denser grid sampling or larger coverage areas would provide more training examples, particularly for the minority handover class. Generating 1,000 to 10,000 samples would better support complex model training and enable more confident performance evaluation.

Second, implementing SMOTE or other oversampling techniques could directly address class imbalance. The MATLAB implementation should experiment with different oversampling ratios, potentially targeting a 1:1 balanced distribution or intermediate ratios such as 2:1. Comparing model performance with and without SMOTE would quantify its effectiveness for this specific application.

Third, collecting real-world data using Android applications such as Network Cell Info Lite or G-NetTrack would provide ground truth validation. While challenging to implement, augmenting synthetic data with real measurements from actual mobile devices moving through cellular coverage areas would enhance model generalizability. The documentation acknowledges this challenge, particularly regarding establishing causal relationships between handovers and battery consumption.

Fourth, simulating more complex propagation scenarios incorporating terrain effects, building shadowing, and multipath fading would increase dataset realism. The Okumura-Hata model's simplicity, while computationally efficient, omits many real-world signal propagation phenomena. Extending the simulation with more sophisticated propagation models or ray-tracing techniques would capture these effects.

5.2 Feature Engineering and Model Enhancement

Expanding the feature set beyond spatial coordinates and RSSI values could significantly improve predictive capability.

Derived features such as RSSI differences between the serving cell and neighboring cells would directly capture handover decision criteria used in traditional algorithms. Rate of RSSI change over time could indicate whether signal quality is improving or degrading, providing predictive value for imminent handovers.

Signal quality metrics including RSRQ, SINR, and bit error rate would complement raw signal strength measurements.
These metrics capture interference and noise conditions that affect connection quality independently of received power. User mobility features such as estimated velocity and direction could help predict trajectory and future cell associations.

Historical handover patterns and temporal features might also prove valuable. Time-of-day indicators, day-of-week patterns, and recent handover history could capture periodic traffic patterns and user behavior regularities. Cell load and capacity information would enable the model to consider network resource availability in handover predictions.

5.3 Advanced Modeling Techniques

Exploring additional machine learning algorithms beyond Random Forest and Logistic Regression could yield performance improvements. Gradient boosting methods such as XGBoost or LightGBM often achieve state-of-the-art results on structured tabular data. Neural network architectures, particularly recurrent neural networks for sequence modeling, could capture temporal dependencies in user trajectories.

Deep learning approaches including convolutional neural networks and long short-term memory networks have shown promise in handover prediction research. These architectures can automatically learn complex feature representations from raw data without manual feature engineering. However, deep learning typically requires substantially larger datasets than the current 196 samples, necessitating dataset expansion before implementation.

Ensemble methods combining multiple algorithms through stacking or blending could leverage the complementary strengths of different models. For example, combining Random Forest's robustness with gradient boosting's optimization efficiency might yield better overall performance than either algorithm alone.

5.4 Hyperparameter Optimization

Systematic hyperparameter tuning using grid search, random search, or Bayesian optimization could substantially improve model performance. For Random Forest, key parameters to optimize include the number of trees (typically 100-1000), maximum tree depth, minimum samples per split, and the number of features sampled at each split. For Logistic Regression, regularization strength and penalty type (L1, L2, or elastic net) require tuning.

Cross-validation should guide hyperparameter selection to prevent overfitting to the training set. K-fold cross-validation with k=5 or k=10 provides robust performance estimates across multiple data splits, particularly important given the small dataset size. Performance metrics from cross-validation would offer more reliable estimates than the single train-test split currently used.

5.5 Operational Deployment Considerations

Transitioning from research prototype to operational deployment requires addressing several practical considerations. Model latency and computational requirements must be evaluated, as handover decisions often require real-time predictions within milliseconds. Random Forest with 100 trees may be too slow for low-latency applications, potentially necessitating model compression or simpler algorithms.

Integration with existing network infrastructure and handover protocols presents technical challenges. The machine learning model must interface with base station controllers and mobility management entities to receive input data and influence handover decisions. Standardized interfaces and protocols would facilitate deployment across heterogeneous network equipment from different vendors.

Model updating and retraining strategies must be established to handle evolving network conditions and user behavior patterns. Online learning or periodic batch retraining ensures that models adapt to changing environments without requiring manual intervention. Monitoring systems should track model performance metrics in production to detect degradation and trigger retraining when necessary.

Safety mechanisms and fallback strategies protect against model failures or unexpected inputs. If the machine learning model produces unreliable predictions or encounters data outside its training distribution, the system should gracefully fall back to traditional handover logic. Confidence thresholds and prediction uncertainty estimates help identify cases where model predictions may be unreliable.

5.6 Visualization and Interpretability

Developing advanced visualizations would enhance understanding of model behavior and facilitate debugging. Real-time plots of user movement across the coverage area with overlaid signal strength contours and handover predictions would provide intuitive insight into model decisions. Feature importance visualizations showing which input variables most strongly influence predictions could guide feature engineering efforts.

The presentation materials mention plans for GUI development in MATLAB to provide interactive visualization capabilities. Such interfaces would enable researchers and network engineers to explore different scenarios, adjust simulation parameters, and observe resulting handover patterns. Interactive tools facilitate hypothesis testing and rapid iteration during model development.

Decision boundary visualizations projected into two-dimensional feature subspaces would reveal how models separate handover from non-handover regions. Comparing these boundaries between Random Forest and Logistic Regression would illustrate the fundamental difference between linear and non-linear decision surfaces.

6. Conclusion

This project successfully demonstrates the feasibility and potential of machine learning approaches for predicting mobile network handover events based on signal strength and spatial location data. The implementation of the Okumura-Hata propagation model provides a physically realistic foundation for synthetic dataset generation, while the comparative evaluation of Random Forest and Logistic Regression illuminates the trade-offs between model complexity and performance.

Random Forest achieved substantially superior performance across all evaluation metrics, with 86.21% accuracy, 0.67 precision, 0.55 recall, and 0.60 F1-score. These results demonstrate the value of ensemble learning methods for capturing the complex, non-linear relationships between spatial position, signal measurements, and handover events. In contrast, Logistic Regression's linear decision boundary proved insufficient for this task, achieving only 81.03% accuracy and critically low 0.09 recall.

The severe class imbalance with handover events representing only 19% of the dataset poses a fundamental challenge that requires dedicated mitigation strategies. The planned implementation of SMOTE and class weighting techniques represents appropriate next steps toward improving minority class detection. Current recall values, while better for Random Forest than Logistic Regression, remain suboptimal for operational deployment, where failing to detect actual handover events could lead to service disruptions.

The project establishes a solid foundation for future research in predictive handover modeling. The documented MATLAB code, systematic evaluation methodology, and comprehensive analysis provide a reproducible baseline that other researchers can build upon. Recommended future directions include dataset expansion, real-world data collection, advanced feature engineering, hyperparameter optimization, and exploration of deep learning architectures.

Ultimately, this work contributes to the growing body of research demonstrating that machine learning can enhance mobile network performance through predictive analytics. By anticipating handover events before they occur, networks can proactively manage resources, reduce unnecessary transitions, and improve quality of service for end users. As mobile networks continue evolving toward 5G and beyond, intelligent, data-driven approaches to mobility management will become increasingly essential for meeting demanding performance requirements.

The practical implications extend beyond handover prediction to broader network optimization challenges including traffic forecasting, resource allocation, anomaly detection, and adaptive quality of service management. The methodologies and insights developed in this project provide a template applicable to these related problems, contributing to the larger goal of cognitive, selfoptimizing mobile networks that leverage artificial intelligence to deliver superior user experiences with minimal human intervention.