

# ANOMALY DETECTION IN LONG-TERM EVOLUTION NETWORK

Chang Brayden Jon Yon<sup>1</sup>, Ang Pau Huang Edwin<sup>2</sup>

<sup>1</sup>St. Joseph's Institution, 38 Malcolm Road, Singapore 308274

<sup>2</sup>Defence Science and Technology Agency, 1 Depot Road, Singapore 109679

---

## ABSTRACT

Long-Term Evolution (LTE) cellular networks are the central component to the success of one of the most essential and in-demand fields globally, telecommunications. However, these LTE networks simultaneously drain expensive and scarce radio resources, primarily electrical power and frequency spectrum. This project thus seeks to use existing fourth generation (4G) LTE telemetry data to develop two supervised classification Machine Learning (ML) models capable of being utilized in a dynamic Radio Resource Management (RRM) system for LTE networks. The ML models will be predicting if a cell exhibits (a) normal behaviour where no redistribution of radio resources is required or (b) anomalous behaviour that requires reconfiguration. Incorporating the use of Exploratory Data Analysis (EDA), feature engineering, and hyperparameter tuning, a Decision Tree (DT) model and an eXtreme Gradient Boosting (XGBoost) model was successfully trained. Model evaluation and explanation was conducted and further explored in a Graphical User Interface (GUI), where it was demonstrated that the DT and XGBoost models had F1-scores of 0.966 and 0.980 respectively, on top of an Area Under the Precision-Recall Curve (AUC) of 0.974 and 0.999 respectively, this indicates exceptional model performance and promising implementation potential predominantly in the latter model.

---

## TABLE OF CONTENTS

<b>ABSTRACT</b> .....	<b>1</b>
<b>TABLE OF CONTENTS</b> .....	<b>1</b>
<b>1. INTRODUCTION</b> .....	<b>2</b>
<b>2. MATERIALS AND METHODS</b> .....	<b>2</b>
2.1 DATASET .....	2
2.2 EXPLORATORY DATA ANALYSIS .....	3
2.3 EXPERIMENTAL SETUP .....	5
2.4 FEATURE ENGINEERING .....	5
2.5 MODEL TRAINING AND HYPERPARAMETER TUNING .....	6
<b>3. RESULTS AND DISCUSSION</b> .....	<b>7</b>
3.1 MODEL EVALUATION .....	7
3.2 MODEL EXPLANATION .....	9
<b>4. CONCLUSION</b> .....	<b>9</b>
4.1 REAL-WORLD IMPLEMENTATION .....	9
4.2 GRAPHICAL USER INTERFACE .....	10
<b>5. LIMITATIONS OF STUDY</b> .....	<b>10</b>
<b>6. ACKNOWLEDGEMENTS</b> .....	<b>10</b>
<b>REFERENCES</b> .....	<b>11</b>
<b>ANNEX</b> .....	<b>12</b>

## 1. INTRODUCTION

Telecommunications is now the most competitive and sought market globally, with existing literature estimating that Mobile Network Operators (MNOs) require the essential capacity to provide for over 100 billion connections in the global mobile communications network in the next 10 years [1]. The foundations for this exponentially growing market are Long-Term Evolution (LTE) cellular networks, capable of transmitting data with lower latency and at higher data rates than their third generation (3G) counterparts [2].

However, these LTE networks rely on the usage of expensive radio resources, particularly electrical power and frequency spectrum. The increased use of wireless devices and mobile appliances has resulted in the congestion of the frequency spectrum and a sharp rise in its price as a result [3]. Traditionally, MNOs resolved this issue of Radio Resource Management (RRM) by deploying more macro cells and overprovisioning cells to ensure smooth operation even during peak hours with high traffic load, a method which is no longer sustainable due to its large energy demands [4]. As such, newer generations of LTE networks including the currently employed fourth generation (4G) and fifth generation (5G) systems look upon Machine Learning (ML) applications as a means of dynamic RRM.

The project thus aims to statistically analyse past 4G LTE network telemetry data which will subsequently be used to train supervised classification ML models capable of classifying cell behaviour as (a) normal: where current activity corresponds to normal behaviour of any day and no reconfiguration is required or (b) anomalous: where current activity differs from the behaviour usually observed for that time of day due to external interruptions such as sports events or strikes which thus triggers a reconfiguration of the base station. A Graphical User Interface (GUI) will also be developed to allow MNOs to create predictions using the ML models and to facilitate the evaluation of the ML models using their testing telemetry data.

## 2. MATERIALS AND METHODS

### 2.1 Dataset

Telemetry data was obtained from a 4G LTE deployment spanning two weeks, where the dependent or outcome feature “Behaviour” along with 13 other predictor features listed in Table 1 were gathered from a set of 10 base stations, in total spanning 33 cells, every 15 minutes. Total sample size of the dataset is 36904. The telemetry dataset can be accessed under the “Annex A: Data Availability” section.

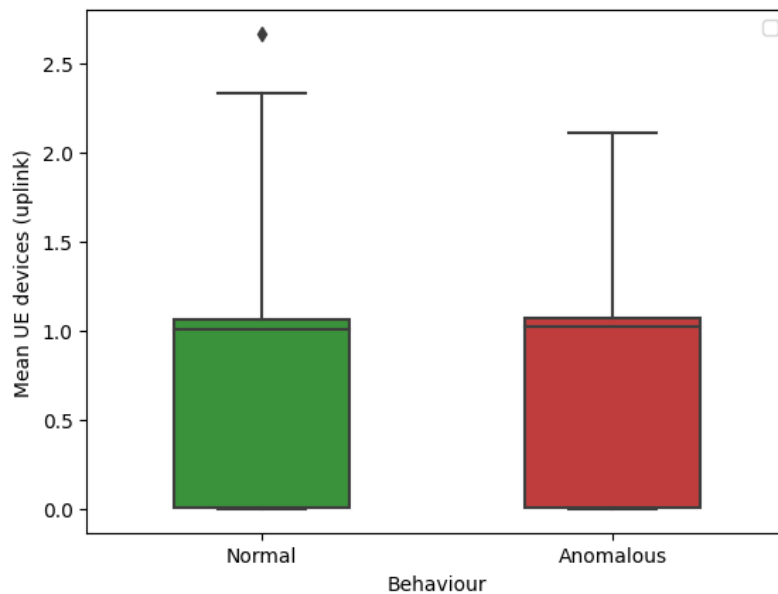
*Table 1. Table of Feature Names and Details*

Feature Type	Feature Name	Unit
Binary	Behaviour (Normal or Anomalous)	NA
Interval	Time (of day)	Hrs:Min
Metric	Percentage of Physical Radio Blocks (PRB) usage (uplink)	%
	Percentage of PRB usage (downlink)	
	Mean carried traffic (uplink)	Mbps
	Mean carried traffic (downlink)	
	Maximum carried traffic (uplink)	
	Maximum carried traffic (downlink)	

	Mean User Equipment (UE) devices (uplink)	NA
	Mean UE devices (downlink)	
	Maximum UE devices (uplink)	
	Maximum UE devices (downlink)	
	Maximum UE devices (uplink and downlink)	
Nominal	Cell ID	

## 2.2 Exploratory Data Analysis

Before approaching the ML model creation, Exploratory Data Analysis (EDA) allows for insights into the traits of the data and the fundamental relationships between features [5]. In this context, it incorporates assessing data quality, identifying distinct outliers, observing data distributions, correlations and relationships between features. Data quality was first assessed by checking for missing values within the rows of the dataset, of which there were none. However, the dataset was observed to be imbalanced with the large majority of cell behaviour being normal (72.4%) rather than anomalous (27.6%).

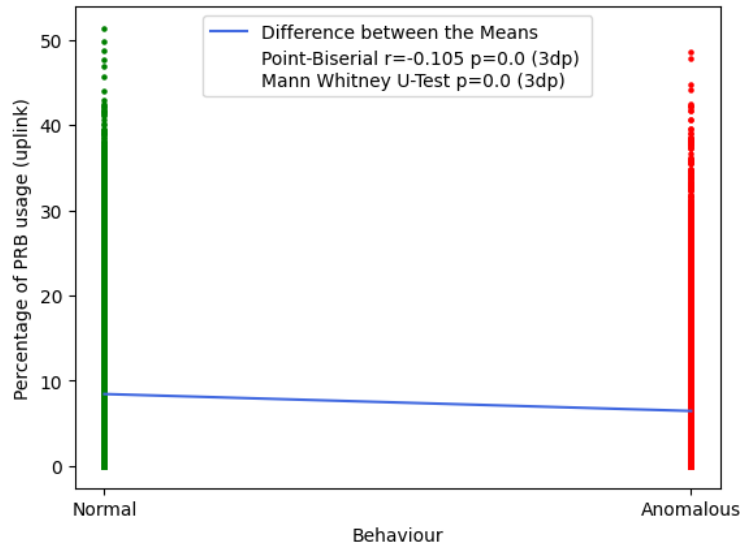


*Figure 1. Normal and Anomalous box plots of mean UE devices (Uplink)*

Data visualisation was subsequently done by plotting boxplots for normal and anomalous behaviour for all 12 predictor features (except “Time”, the feature “Time” cannot be analysed until it is converted into alternate forms and is thus negated for this section). Any values determined to be outliers by Tukey’s method [6] was plotted as a diamond dot. Utilizing this method, the only distinct global outlier was identified in the mean active UE devices (uplink) feature, shown in Figure 1, with a value of 2.668 and was subsequently removed for all following analysis and model training.

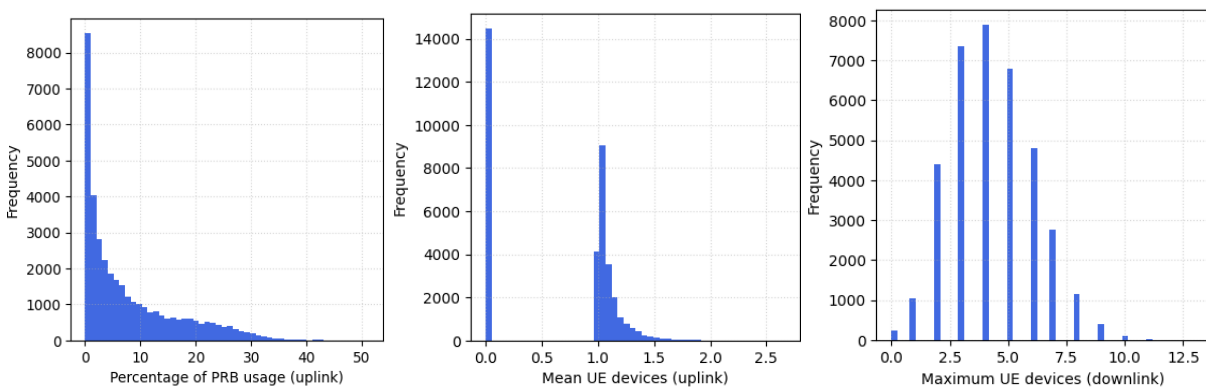
Afterwards, feature importance was examined to determine which features would benefit the performance of the models and should be utilised. This was done through hypothesis tests that would determine whether a feature has a statistically significant effect on the outcome feature “Behaviour”. To ascertain the type of statistical analysis to perform, a Shapiro-Wilk (SW) test

for normality [7] was conducted and it revealed that all features were not normal ( $p > 0.05$ ), as such, non-parametric hypothesis tests, the Chi Square test for homogeneity [8], Point-Biserial correlation (PB) [9] and Mann Whitney U-test (MW) [10] were selected.



**Figure 2.** Scatter plot of percentage of PRB usage (uplink) with hypothesis tests

As depicted in Figure 2, an example out of the 12 predictor features, we observe in the scatter plot visualisation a difference between the means that is supported by a Point-Biserial correlation coefficient,  $r_{pb}$ , of -0.105 that is statistically significant ( $p < 0.001$ ), indicating a weak negative correlation where cells with lower percentages of PRB usage (uplink) tend to have anomalous behaviour. This is reinforced by a statistically significant difference in the means shown by the Mann Whitney U-Test ( $p < 0.001$ ). As such, the feature is deemed to have a statistically significant impact on the outcome feature and is utilised for model training. All metric features underwent the above analysis, however, a separate hypothesis test, the Chi Square test for homogeneity (CS) was conducted for the nominal feature “Cell ID”.



**Figure 3.** Histogram of percentage of PRB usage (uplink) (left)

**Figure 4.** Histogram of mean active UE devices (uplink) (middle)

**Figure 5.** Histogram of maximum active UE devices (downlink) (right)

Data distribution was subsequently assessed through histograms and three types of distributions were observed among the 11 metric features: (a) exponential distribution as seen in Figure 2 with traits such as a peak at 0 and a long right tail. (b) Bimodal-Lognormal distribution as seen

in Figure 3, with a sudden peak at 0 and another peak in a log normal distribution around 1.0. (c) Normal distribution as seen in Figure 4, all features of this category had integer only values. All features were hence categorised into their different data distribution types. The table below illustrates the summary of the EDA findings, which will be used to influence future processes.

**Table 2.** Summary of EDA findings on twelve predictor features (excluding “Time”) (3dp)

Feature Name	Distribution	SW	CS	MW	PB		
		p-value			$r_{pb}$		
Percentage of PRB usage (uplink)	Exponential	.000	NA	.000	.000	-.105	
Percentage of PRB usage (downlink)		.000		.000	.000	-.077	
Mean carried traffic (uplink)		.000		.000	.000	-.030	
Mean carried traffic (downlink)		.000		.000	.000	-.054	
Maximum carried traffic (uplink)		.000		.000	.000	-.049	
Maximum carried traffic (downlink)		.000		.000	.003	-.015	
Mean UE devices (uplink)	Bimodal-	.000	NA	.000	.151	-.007	
Mean UE devices (downlink)	Lognormal	.000		.000	.000	-.083	
Maximum UE devices (uplink)	Normal	.000		.917	.068	-.009	
Maximum UE devices (downlink)		.000		.000	.000	-.037	
Maximum UE devices (uplink and downlink)		.000		.118	.000	-.023	
Cell ID	NA	.000		.999	NA		

### 2.3 Experimental Setup

All data manipulation, model training and creation of GUIs was done in Python 3.11.4, where all stochastic functions use a “random\_state” parameter of 53 (random seed = 53). The dataset was shuffled and split into training (80%) and testing (20%) datasets using scikit-learn’s [11] “train\_test\_split” function with the “shuffle” parameter set to “True”.

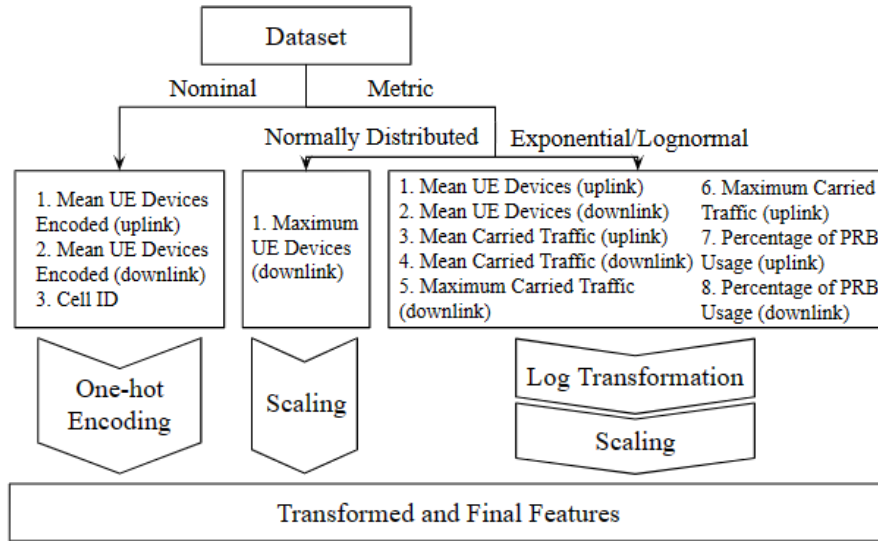
### 2.4 Feature Engineering

Through transforming or creating new suitable features, feature engineering aims to improve predictive performance by extracting the most important information from the data [12]. Three new features were created, the “Hour of the day” categorical feature from the “Time” feature using ordinal encoding, returning integers ranging from 0 to 24. Along with two features “Mean UE devices encoded (uplink)” and “Mean UE devices encoded (downlink)” which were created from the features “Mean UE devices (uplink)” and “Mean UE devices (downlink)” respectively using one hot encoding, returning 1 if the value of the original feature is 0 and 0 if the value is not, extracting information from the bimodal trait observed during the EDA component. These features were then statistically analysed to determine feature importance, as shown in Table 3.

**Table 3.** Summary of Chi-Square test results on three new created features (3dp)

Feature Name	p-value of CS Test for Homogeneity	p-value of CS Test for Independence
Hour of the day	.763	NA
Mean UE devices encoded (uplink)	NA	.047
Mean UE devices encoded (downlink)	NA	.001

Only predictor features identified during EDA that had a statistically significant effect on the outcome feature subsequently underwent pre-processing as illustrated in Figure 6, where these features were transformed into numeric forms with the most important information captured. The pre-processor was fit on the training data and used to transform all data.



**Figure 6.** Flowchart of pre-processor process

For nominal features, on top of the one-hot encoding already done during the creation of the “Mean UE devices encoded (uplink)” and “Mean UE devices encoded (downlink)” features, the “Cell ID” feature had also under went one hot encoding, resulting in the creation of 33 different features, one for each cell, each feature returning a value of 1 if the Cell ID matches with the assigned feature and returning a value of 0 if it does not.

The metric features were further categorised into normally distributed, and exponential or lognormal distributed features based on observations of the data distributions made during EDA. Exponential or Lognormal features would undergo a log transformation in order to normalise the data, improving model performance and training stability. These features would thereafter undergo scaling along with the normally distributed data, where the feature is scaled to have a mean of 0 and a standard deviation of 1. Scaling ensures that the magnitude of influence each feature has on the ML model is comparable and reduces excessive bias.

## 2.5 Model Training and Hyperparameter Tuning

The two selected ML models for training are the Decision Tree (DT) model [13], which will utilise “scikit-learn”’s “DecisionTreeClassifier()” function, and the eXtreme Gradient Boosting (XGBoost) model [14], which will utilise “xgboost”’s “XGBClassifier()” function, which is an ensemble learning technique that consists of multiple decision trees building upon the residuals of the previous tree. The chosen objective function for hyperparameter tuning is the F1-score:

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

where precision is the fraction of true positives over predicted positives and recall is the fraction of true positives over actual positives. The F1 score ranges from a perfect score of 1 to the worst

possible score of 0, it is the selected objective function due to its utilisation of both precision and recall in a manner which scores models with more balanced precision and recall values higher, it hence is more suitable for the imbalanced classification dataset being used.

Hyperparameter tuning of the models begins with the creation of a study in the Optuna library [15], the first trial then begins with stratified 10-fold validation [16] for both models, splitting the processed training dataset into a new validation (10%) and training dataset (90%) over 10 folds. Stratified k-fold validation is the selected validation method as it addresses the imbalanced dataset by ensuring each fold has the same proportion of anomalous to normal cell behaviour as the original training dataset. The function is performed using the “StratifiedKFold()” function in “scikit-learn” with the “shuffle” parameter set to “True”.

**Table 3.** Table of selected hyperparameters and ranges for DT and XGBoost Model

Decision Tree			XGBoost		
Hyperparameter	Range	Type	Hyperparameter	Range	Type
max_depth	5 - 500	Integer	n_estimators	50 - 1000	Integer
min_samples_split	2 - 100	Integer	max_depth	5 - 500	Integer
min_samples_leaf	2 - 100	Integer	learning_rate	0.00001 - 0.1	Float
max_leaf_nodes	2 - 2000	Integer	subsample	0 - 1	Float
ccp_alpha	0 - 0.01	Float	colsample_bytree	0 - 1	Float
NA			colsample_bylevel	0 - 1	Float

A random set of hyperparameters is then chosen from specified ranges shown in Table 3, where all non-specified hyperparameters use the default values described in their respective library documentation. For each fold, the models are subsequently fitted with the chosen hyperparameters on the new training data and tested on the validation data to obtain a F1 score, the mean F1 score of all the folds is then calculated, indicating the performance of the model with the chosen set of hyperparameters. In the following trial, the hyperparameters will be tuned using Optuna’s optimisation algorithm based on the mean F1 score of the previous trial, aiming for a “maximised” F1 score. The DT model ran 100 trials while the XGBoost model only ran 5 trials due to its increased computational intensity. After all trials are complete, the trial with the highest mean F1 score is selected as the model with the best performing hyperparameters. This set of hyperparameters is then used to fit the final model on the entirety of the training data.

### 3. RESULTS AND DISCUSSION

#### 3.1 Model Evaluation

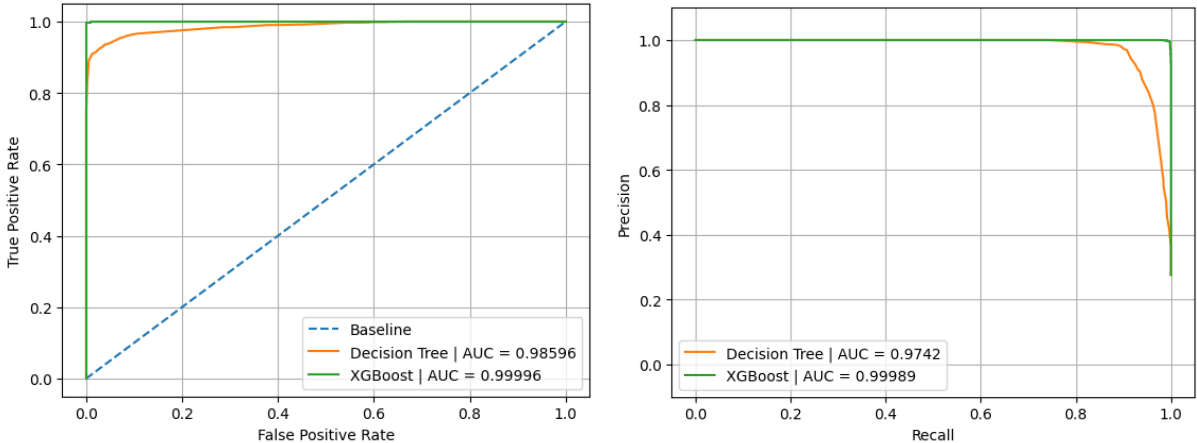
The two fitted ML models made predictions for the outcome feature “Behaviour” based on the testing dataset and was compared against the actual results as shown in the confusion matrix [17] in Table 4, showing a tabular summary of the number of correct and incorrect predictions.

**Table 4.** Table of Confusion Matrix for DT and XGBoost models

		Decision Tree		XGBoost	
		Predicted		Predicted	
		Normal	Anomalous	Normal	Anomalous
Actual	Normal	5296	50	5346	0
	Anomalous	204	1831	77	1958

It is observed from the confusion matrix that the XGBoost model particularly excels at minimising false positives, having a perfect precision for anomalous behaviour of 1.00 and creating 0 false positive predictions. However, it has a lower recall for anomalous behaviour of 0.962, indicating that the model prioritises accurate anomalous predictions over predicting all true anomalies. The DT model similarly prioritises accurate anomalous predictions but to a larger extent, with a greater precision for anomalous behaviour of 0.973 than its recall for anomalous behaviour of 0.900.

A high F1 score was also calculated for both models, with the XGBoost model being slightly higher at 0.980 compared to the DT model’s F1 score of 0.966. Hence, not only does the XGBoost model have a greater overall accuracy and score than the DT model, it also has a better balance between precision and recall values.



**Figure 7.** Receiver Operating Characteristic (ROC) curve of DT and XGBoost model  
**Figure 8.** Precision Recall Display (PRD) of DT and XGBoost model

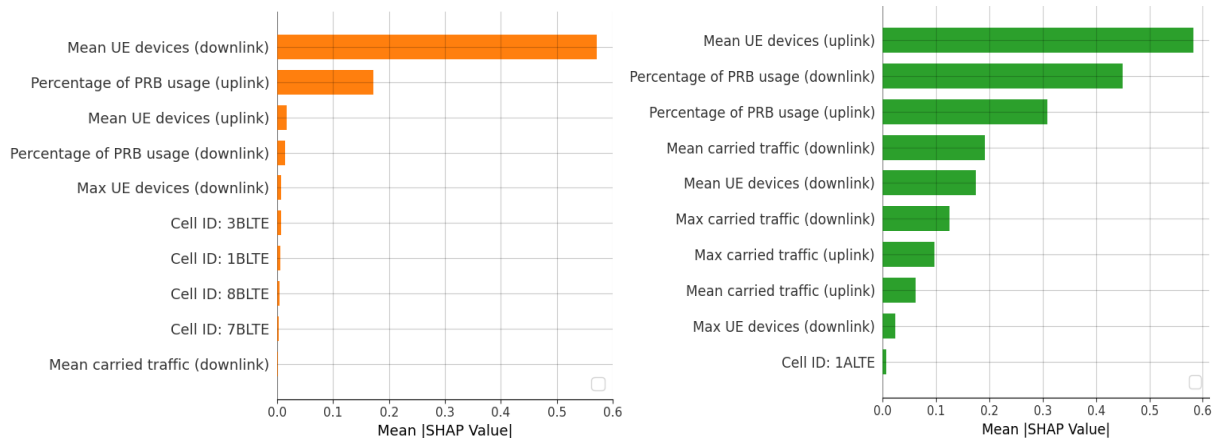
Both models were subsequently evaluated using a ROC curve [18], shown in Figure 7, which denotes the true positive rate and false positive rate of the models at different thresholds, where a higher true positive rate across all thresholds indicates better model performance. Both models are significantly higher than the “Baseline”, which acts as a random guess model that obtains no information from the predictor features.

It was observed that the XGBoost model had performed better than the DT model at all possible thresholds, coming exceedingly close to being a perfect model of having a true positive rate of 1.0 when the false positive rate is 0.0. This is further reinforced by the Area Under the Curve (AUC) metric, where a greater AUC value indicates higher true positive rates across all thresholds. The XGBoost model had a AUC value of 0.99996 which was greater than the AUC value of 0.98596 for the DT model.

The PRD [18], shown in Figure 8, is capable of evaluating model performance in imbalanced datasets better than ROC curves as it plots the trade-off between precision and recall across different thresholds. Likewise, the XGBoost model outperforms the DT model, having either greater or equal precision and recall at all thresholds. The XGBoost model also has a high AUC value of 0.99989 compared to the DT model’s AUC value of 0.97420.



### 3.2 Model Explanation



**Figure 9.** Bar chart of top 10 SHapley Additive Explainer (SHAP) values for DT model (left)  
**Figure 10.** Bar chart of top 10 SHAP values for XGBoost model (right)

In order to identify the features that had the largest magnitude of influence on the models and the greatest feature importance, SHAP [18] values for each feature was found for all data points in the testing dataset, and the mean of the absolute SHAP values were found and plotted. SHAP values are based off cooperative game theory, determining how much the changing of a feature would affect the outcome of the predictive model, effectively identifying feature importance. This was done using the “tree.explainer()” and “summary\_plot” functions in the “shap” library.

As shown in Figure 9, the DT model is heavily influenced by mainly two features, “Mean UE devices (downlink)” with a mean absolute SHAP value around 0.56 and “Percentage of PRB Usage (uplink)” with a mean absolute SHAP value around 0.17. Similarly, the XGBoost model is most influenced by the same two features but with reversed communication directions, with a mean absolute SHAP value around 0.58 for “Mean UE devices (uplink)” and a mean SHAP value around 0.43 for “Percentage of PRB Usage (downlink)”.

These 4 features were generally observed to have some of the strongest point biserial correlation coefficients during the EDA component which would justify their high mean absolute SHAP values. The XGBoost model is also observed to have greater mean absolute SHAP values for other features besides the two most influential features unlike the DT model, which could hypothetically explain for the difference in predictive performance. Whereby the DT model is prone to overreliance on the two most influential features and is thus more likely to output different predictions due to minor fluctuations in the two aforementioned features. In contrast, the XGBoost model is more consistently influenced by a large set of features and is hence more reliable, leading to greater predictive performance.

## 4. CONCLUSION

### 4.1 Real-world Implementation

The DT model and XGBoost model were able to output highly accurate predictions and have

excellent model predictive performances, though the XGBoost model has exceedingly higher model performance across all evaluation metrics. The models are thus a consistently accurate way of predicting radio cell behaviour and could be a feasible method to implementing dynamic RRM.

However, one of the largest drawbacks of the model is its reliance on completeness of data quality, requiring values for all features in order to make predictions, which may not be feasible during real-life implementation and data collection. Although the usage of methods such as imputation could resolve the issue it may result in significant drawbacks in accuracy.

#### **4.2 Graphical User Interface**

A Graphical User Interface (GUI) was created utilising the “streamlit” library, creating an interface for MNOs and users to create predictions with raw testing data input or to evaluate the ML models with the aforementioned evaluation metrics. The website application can be accessed under the “Annex B: GUI Website Application” section.

### **5. LIMITATIONS OF STUDY**

This study is primarily focussed on the application of ML for cell behaviour prediction, but does not go into detail or development of the dynamic RRM system itself, placing heavier emphasis on the ML models. Future studies could delve into creating a dynamic RRM system capable of reconfiguring identified anomalies and evaluating effectiveness of the entire system.

Additionally, this study is limited with a dataset pertaining to just one fixed LTE deployment. The ML models may not perform up to its expected standard when used in new and foreign deployments due to the ML models usage of certain location specific features such as “Cell ID” and the fluctuation of telemetry data and patterns from region to region. A more comprehensive study could utilise data from multiple deployments and minimise location-specific data. An alternative fix for this in future applications would be the integration of Machine Learning Operations (MLOps) for the continued training and updating of the ML models to maintain model performance and for continued adaptation in different environments.

### **6. ACKNOWLEDGEMENTS**

The completion of this project could not have been possible without the mentorship and expertise of Mr Edwin Ang, I would like to express my immense gratitude for his crucial guidance and support throughout the entirety of the project. I also owe a great debt of gratitude to Ms Esther Chew for her skilful handling of all administrative matters and to LTE expert Mr Robin Ong Boon Ping for his support and assistance.

Last but not least, I would like to thank the director of InfoComm Infrastructure Programme Centre Mr Leong See Sum, as well as DSTA for providing me with this privileged opportunity to participate in Research@YDSP 2023.

## REFERENCES

- [1] Gandhi, J., Narmawala, Z. 2022. Capacity and Cost Analysis of 4G-LTE and 5G Networks. In: Singh, P.K., Weirzchon, S.T. Chhabra, J.K., Tanwar, S. (eds) Futuristic Trends in Networks and Computing Technologies, Lecture Notes in Electrical Engineering, vol 936
- [2] Ezhilarasan, E. and Dinakaran, M. 2017. A Review on Mobile Technologies: 3G, 4G and 5G. 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM) pp. 369-373.
- [3] Luong, N. C., Lu, X., Hoang, D. T., Nivato, D., and Kim, D. I. 2021. Radio Resource Management in Joint Radar and Communication: A Comprehensive Survey. IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 780-814, Secondquarter 2021
- [4] Arbi, A. 2017. Spectral and Energy Efficiency in Cellular Mobile Radio Access Networks. Department of Electronic and Electrical Engineering.
- [5] Jebb, A., Parrigon, S. and Woo, S. 2017. Exploratory data analysis as a foundation of inductive research. Human Resource Management Review, Volume 27, Issue 2, 2017, pp. 265-276, ISSN 1053-
- [6] Saleem, S., Aslam, M. and Rukh, S. 2021. A review and empirical comparison of univariate outlier detection methods. Pak. J. Statist. 2021 Vol. 37(4), 447-462.
- [7] Shapiro, S and Wilk, M. 1965. An analysis of variance test for normality (complete samples). Biometrika 52, 3 and 4, p. 591
- [8] David, C. n.d. Chi square test - analysis of contingency tables. Professor Emeritus, University of Vermont
- [9] Kornbrot, D. 2014. Point Biserial Correlation Encyclopedia of Statistics in Behavioral Science, © John Wiley & Sons, Ltd.
- [10] Emerson, R. 2023. Mann-Whitney U test and t-test. Journal of Visual Impairment & Blindness, 117(1), 99-100.
- [11] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, Fabian, Mueller, A., Grisel, O. and Ga"el V. 2013. API design for machine learning software: experiences from the scikit-learn project. In ECML PKDD Workshop: Languages for Data Mining and Machine Learning, pp. 108–122.
- [12] Fatemeh, N., Horst, S., Udayan, K., Elias, B., and Deepak, T. 2017. Learning Feature Engineering for Classification. Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17) 2529.
- [13] A. Navada, A. N. Ansari, S. Patil and B. A. Sonkamble. 2011. Overview of use of decision tree algorithms in machine learning. 2011 IEEE Control and System Graduate Research Colloquium, pp. 37-42.

[14] Ramraj, S., Nishant, U., Sunil, R. and Shatadeep, B. 2016. Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets. International Journal of Control Theory and Applications. ISSN : 0974–5572 © International Science Press Volume 9 Number 40.

[15] Akiba, T., Sano, S., Yanase, T., Ohta, T. and Koyama M. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. KDD '19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2623–2631.

[16] Berrar, D. 2018. Cross-Validation. Encyclopedia of Bioinformatics and Computational Biology, Volume 1, Elsevier, pp. 542-545.

[17] Fawcett, T. 2006. An Introduction to ROC Analysis. 27 (8): 861-874. Pattern Recognition Letters. 27 (8): 861–874.

[18] Scott, M. and Lee, S. 2017. A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems. 30: 4765-5774 arXiv:1705:07874.

## ANNEX

### Annex A: Data Availability

All raw telemetry data utilised in this project is openly available at the Kaggle database and can be accessed at <https://www.kaggle.com/competitions/anomaly-detection-in-4g-cellular-networks/overview> from Vidal, J. 2020. Anomaly detection in 4G cellular networks.

### Annex B: GUI Website Application

The GUI website application can be accessed at <https://anomalydetectionlte.streamlit.app>.

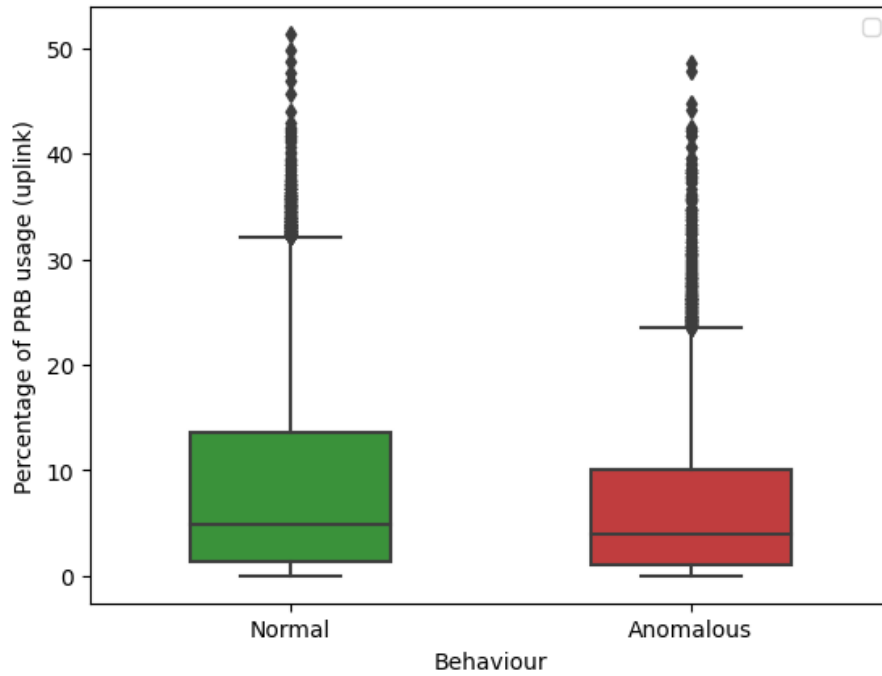
### Annex C: Final hyperparameters for ML models

All non-specified hyperparameters use the default values as specified in the respective libraries' documentation.

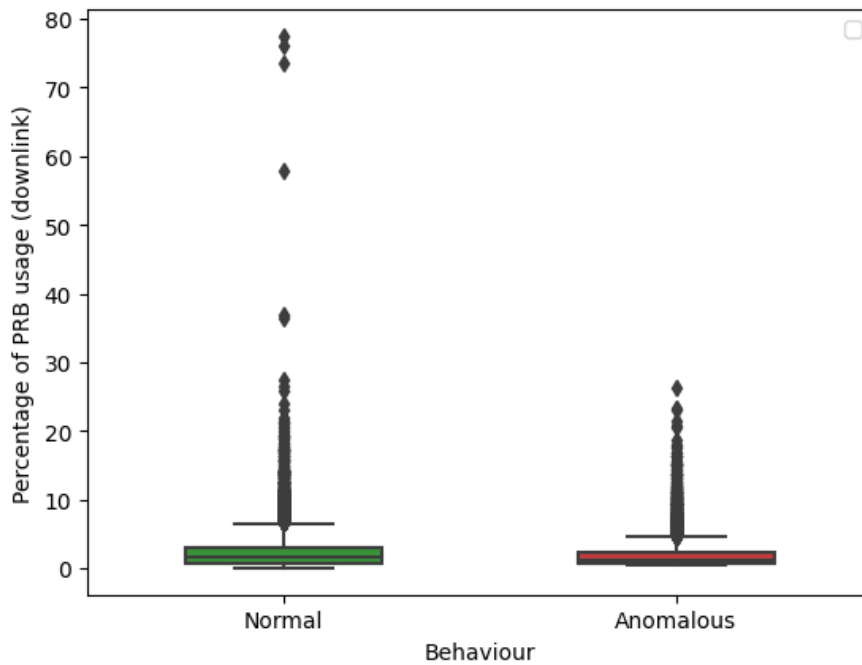
*Table 5. Table of final hyperparameters for DT and XGBoost Model (10 dp for float types)*

Decision Tree		XGBoost	
Hyperparameter	Value	Hyperparameter	Value
max_depth	443	n_estimators	447
min_samples_split	47	max_depth	178
min_samples_leaf	4	learning_rate	0.0440782828
max_leaf_nodes	678	subsample	0.7540475616
ccp_alpha	0.0001867337	colsample_bytree	0.0264677405
NA		colsample_bylevel	0.3361845730

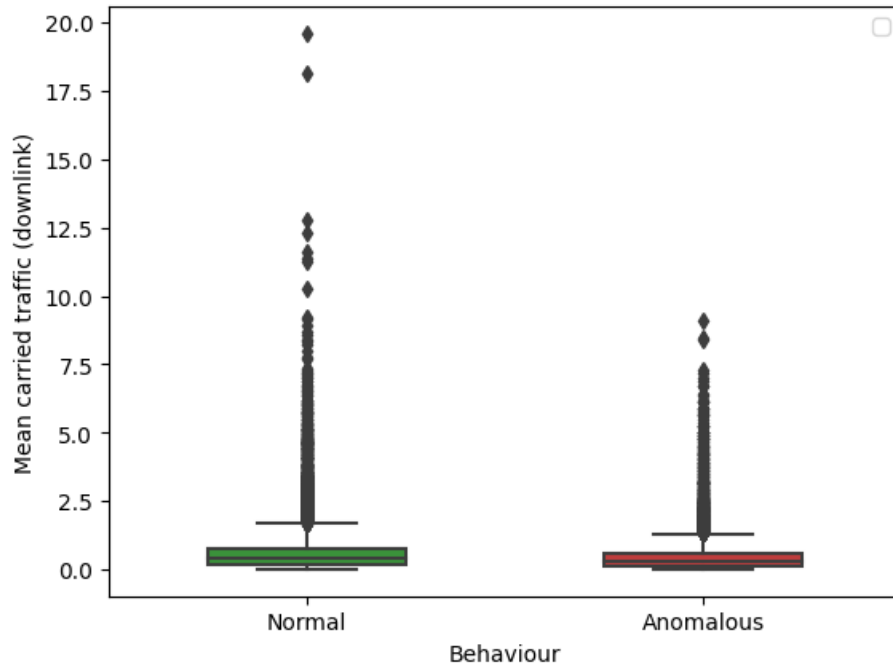
### Annex D: Exploratory Data Analysis Plots



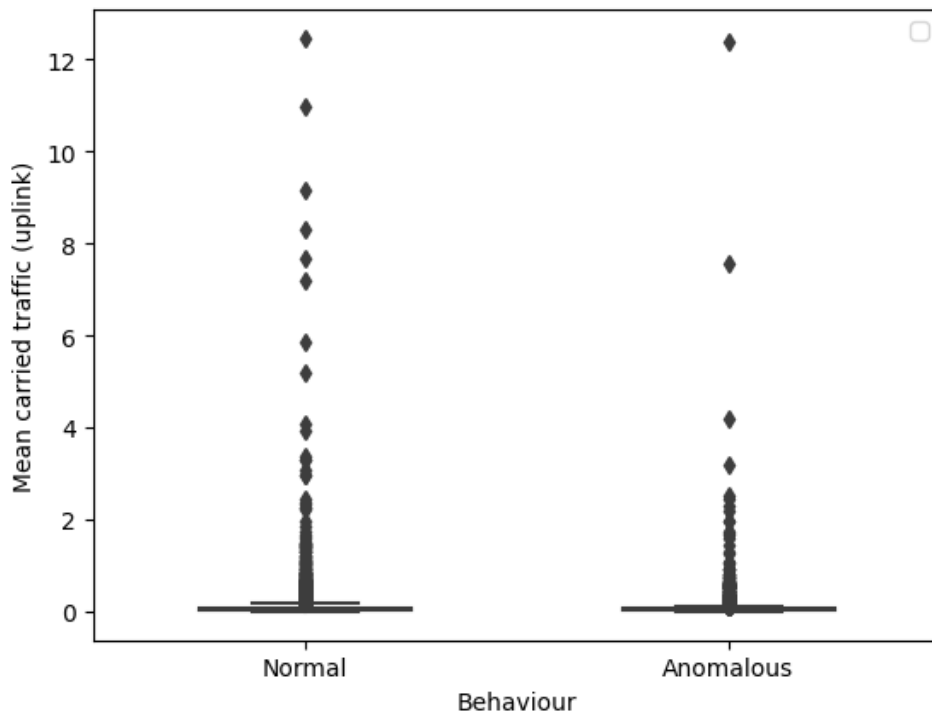
*Figure 11. Normal and Anomalous box plots of percentage of PRB usage (uplink)*



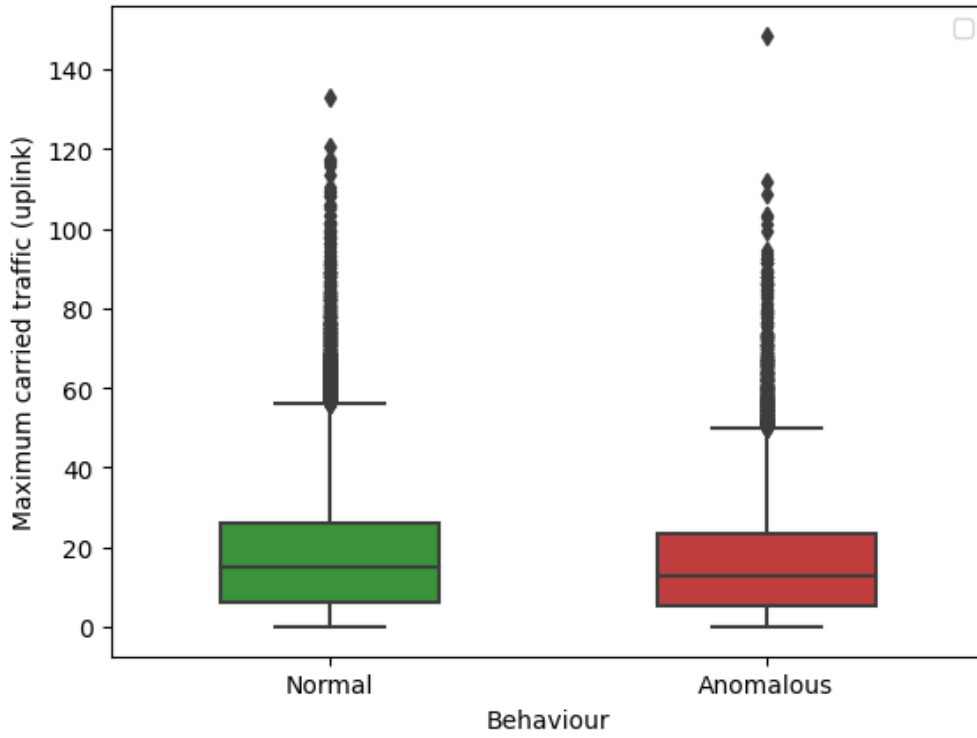
*Figure 12. Normal and Anomalous box plots of percentage of PRB usage (downlink)*



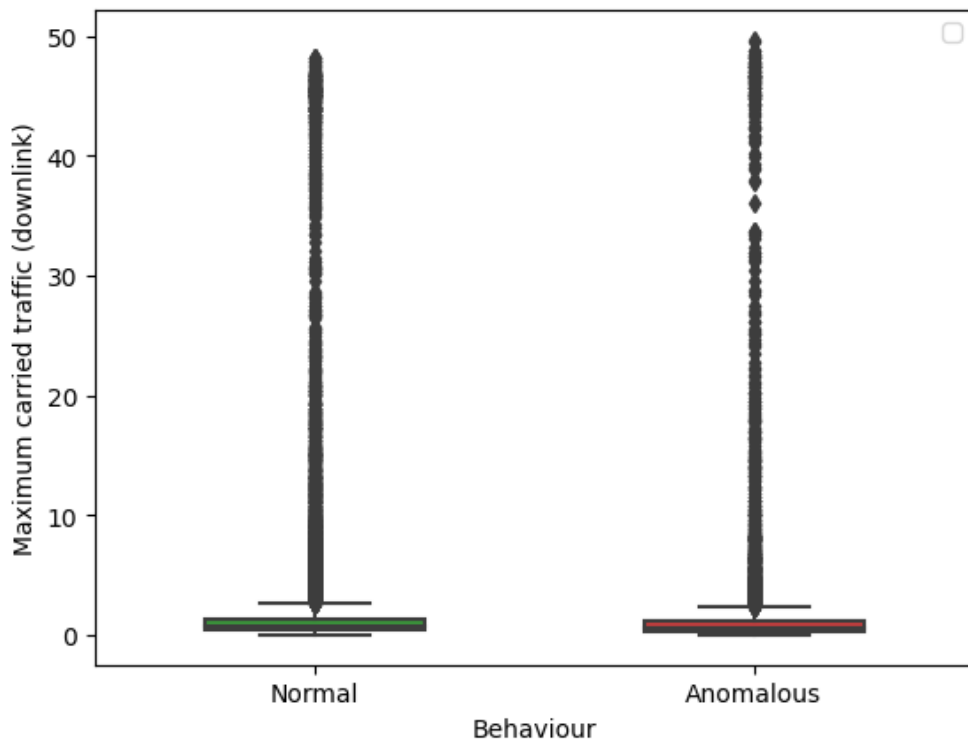
**Figure 13.** Normal and Anomalous box plots of mean carried traffic (downlink)



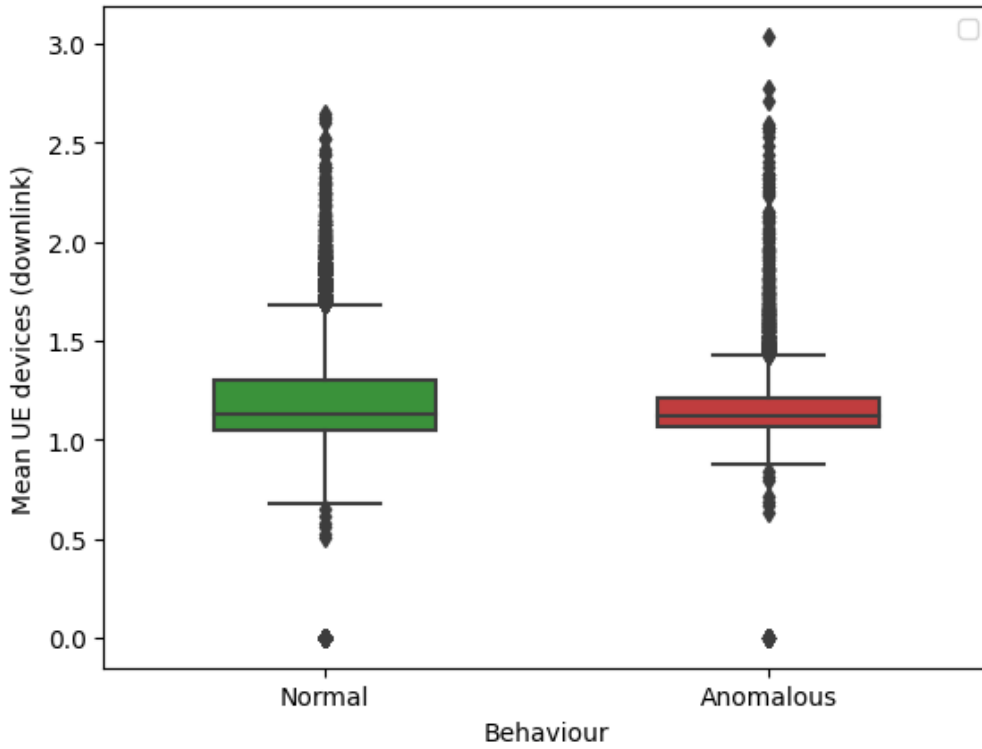
**Figure 14.** Normal and Anomalous box plots of mean carried traffic (uplink)



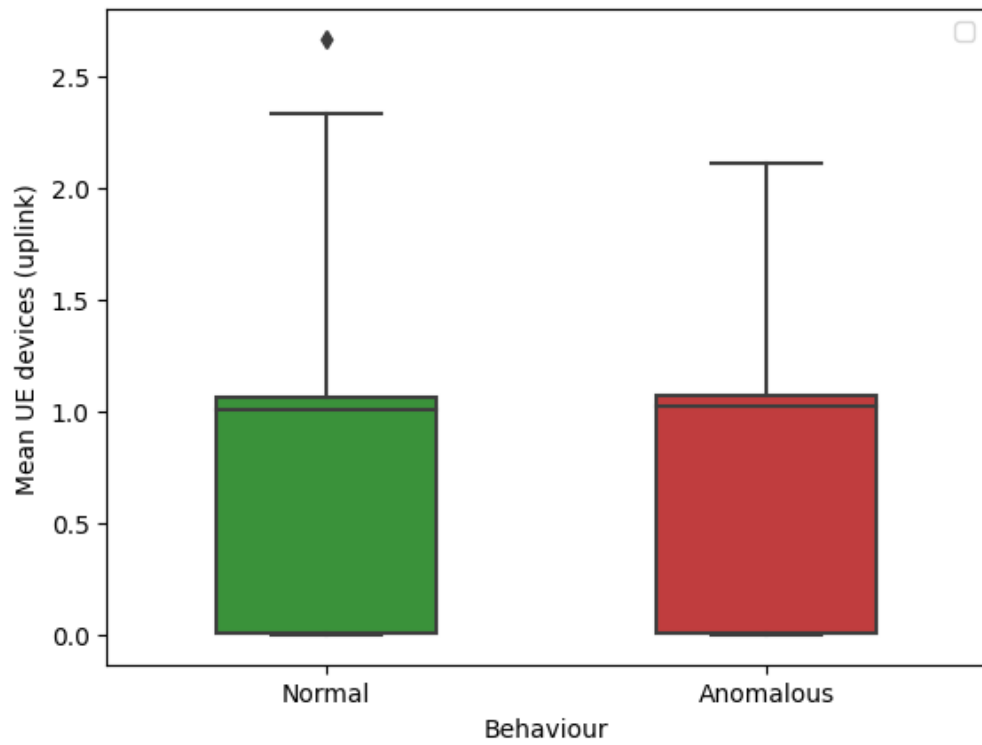
**Figure 15.** Normal and Anomalous box plots of maximum carried traffic (uplink)



**Figure 16.** Normal and Anomalous box plots of maximum carried traffic (downlink)

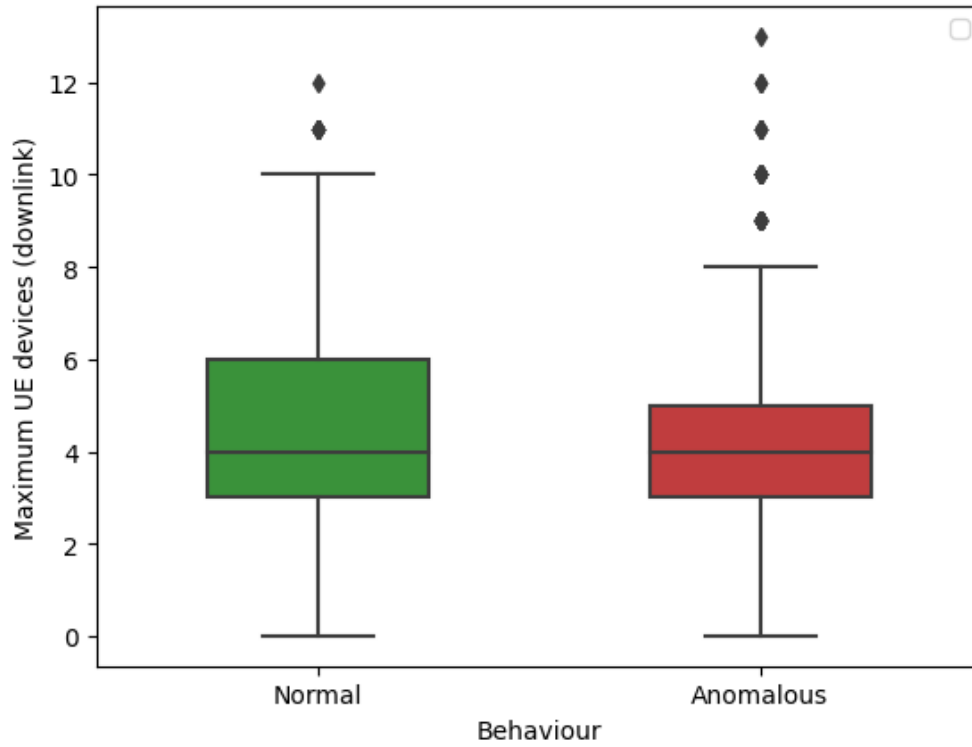


**Figure 17.** Normal and Anomalous box plots of mean UE devices (downlink)

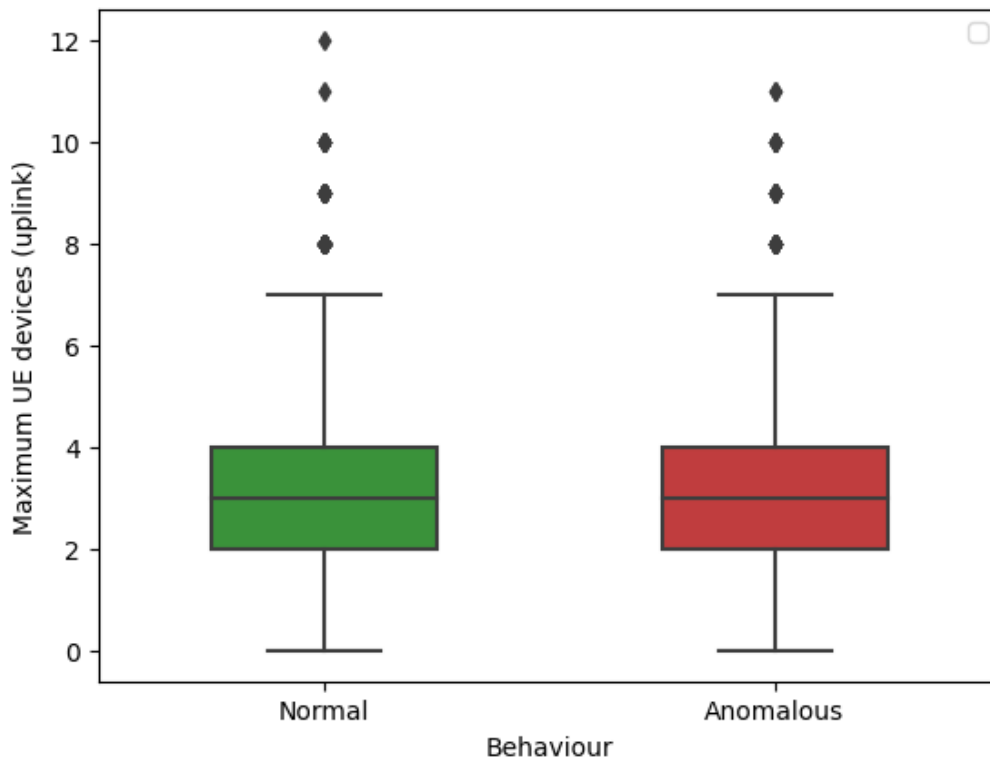


**Figure 1.** Normal and Anomalous box plots of mean UE devices (Uplink)

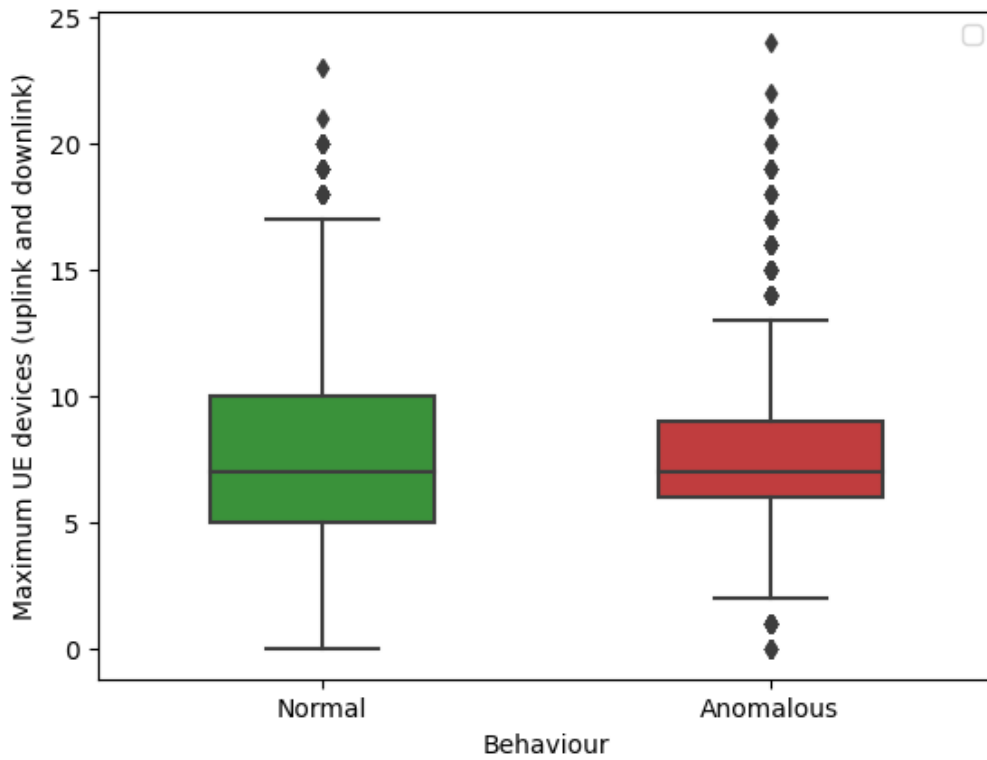




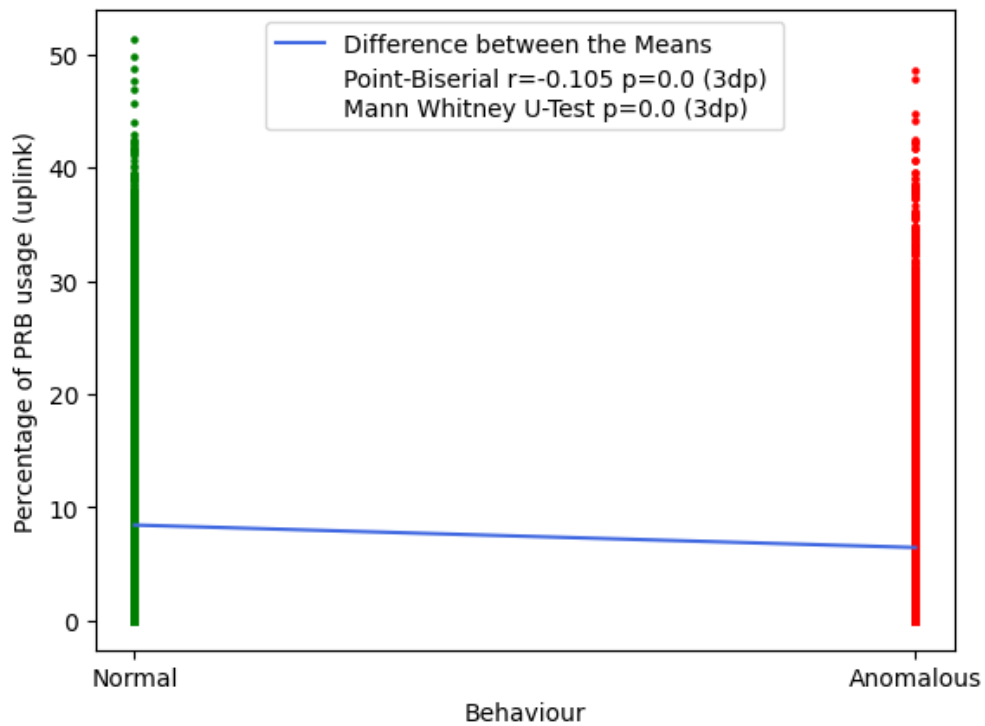
**Figure 18.** Normal and Anomalous box plots of maximum UE devices (downlink)



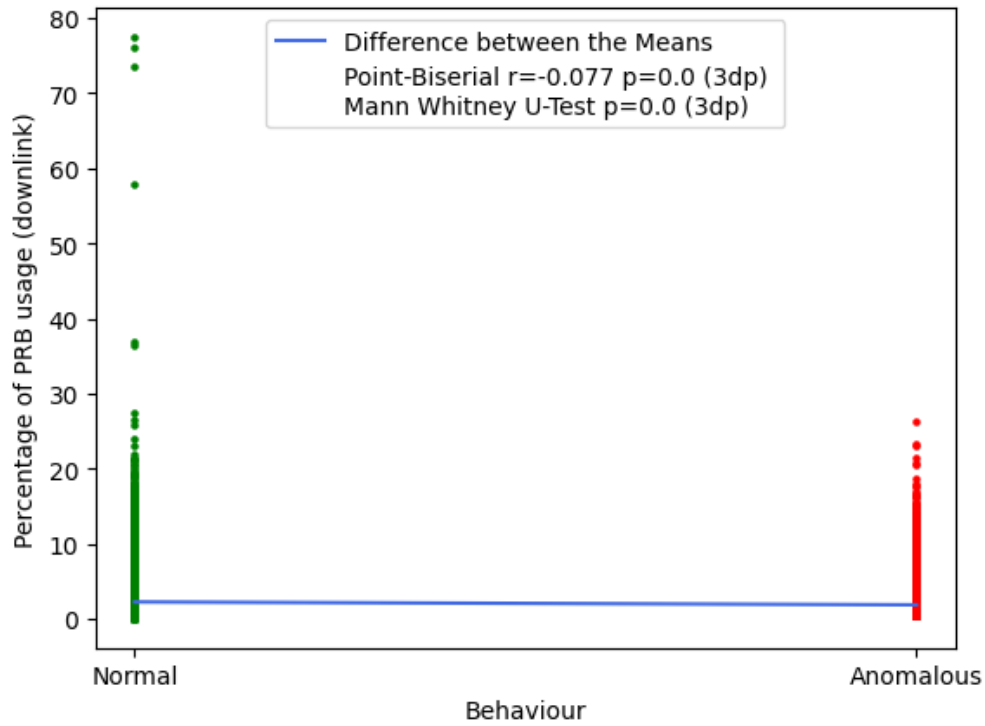
**Figure 19.** Normal and Anomalous box plots of maximum UE devices (uplink)



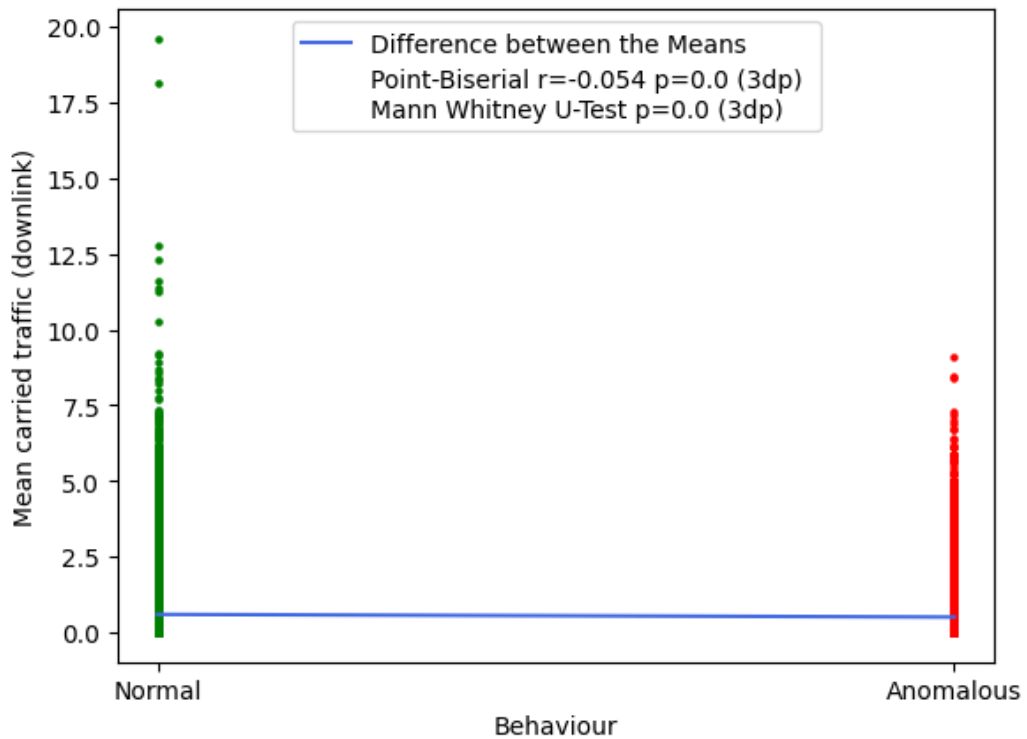
**Figure 20.** Normal and Anomalous box plots of maximum UE devices (uplink and downlink)



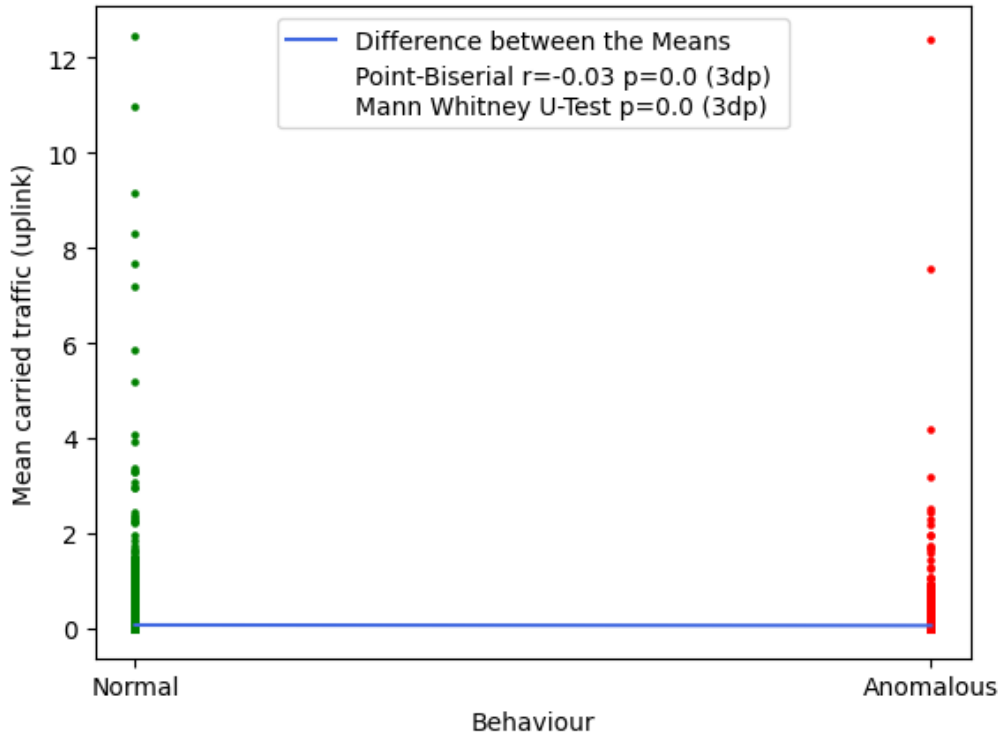
**Figure 2.** Scatter plot of percentage of PRB usage (uplink) with hypothesis tests



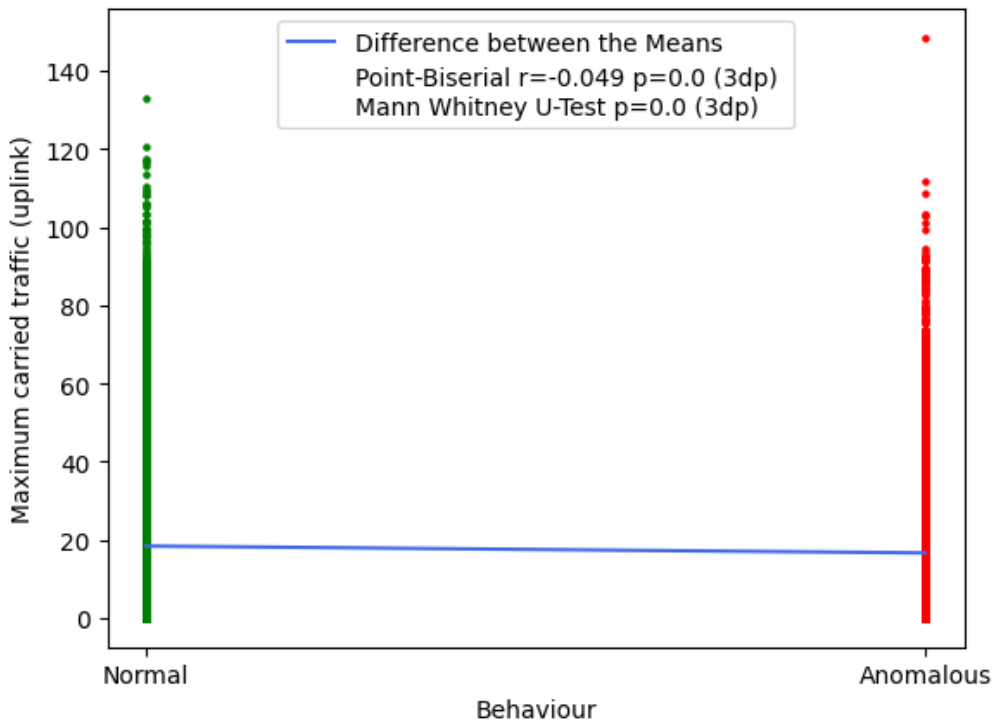
**Figure 21.** Scatter plot of percentage of PRB usage (downlink) with hypothesis tests



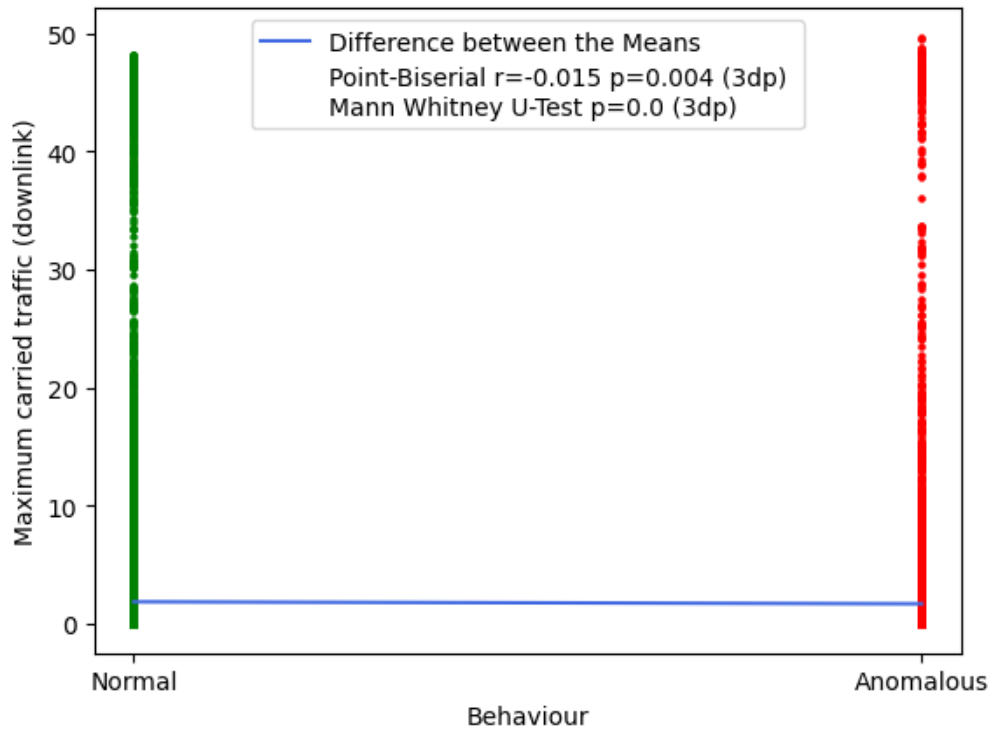
**Figure 22.** Scatter plot of mean carried traffic (downlink) with hypothesis tests



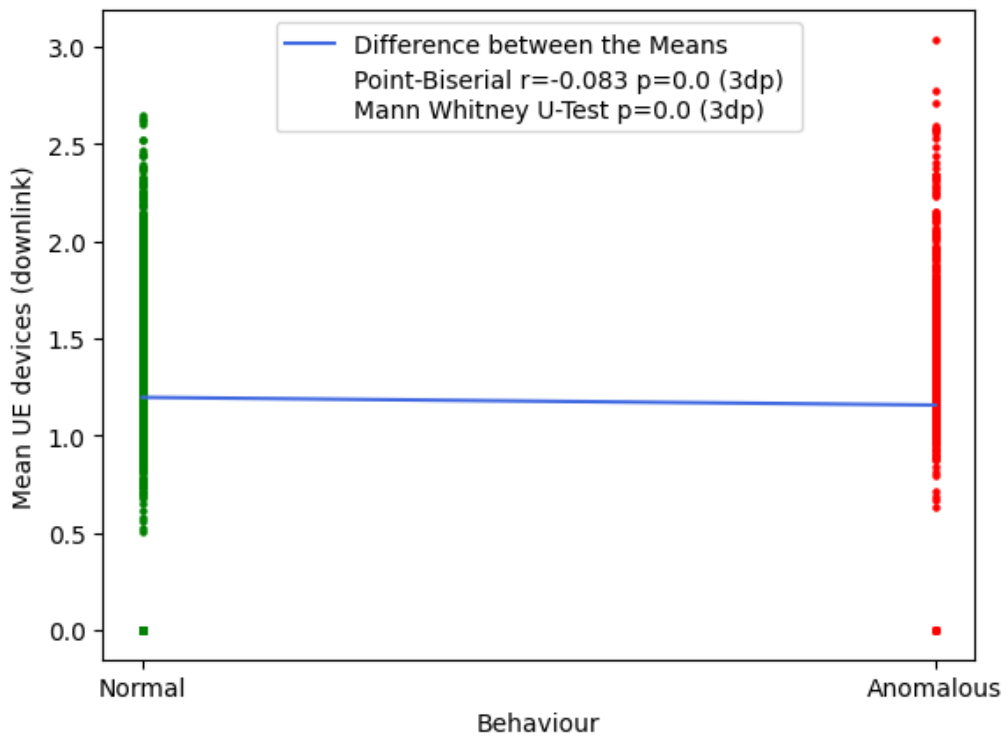
*Figure 23. Scatter plot of mean carried traffic (uplink) with hypothesis tests*



*Figure 24. Scatter plot of maximum carried traffic (uplink) with hypothesis tests*



**Figure 25.** Scatter plot of maximum carried traffic (downlink) with hypothesis tests



**Figure 26.** Scatter plot of mean UE devices (downlink) with hypothesis tests

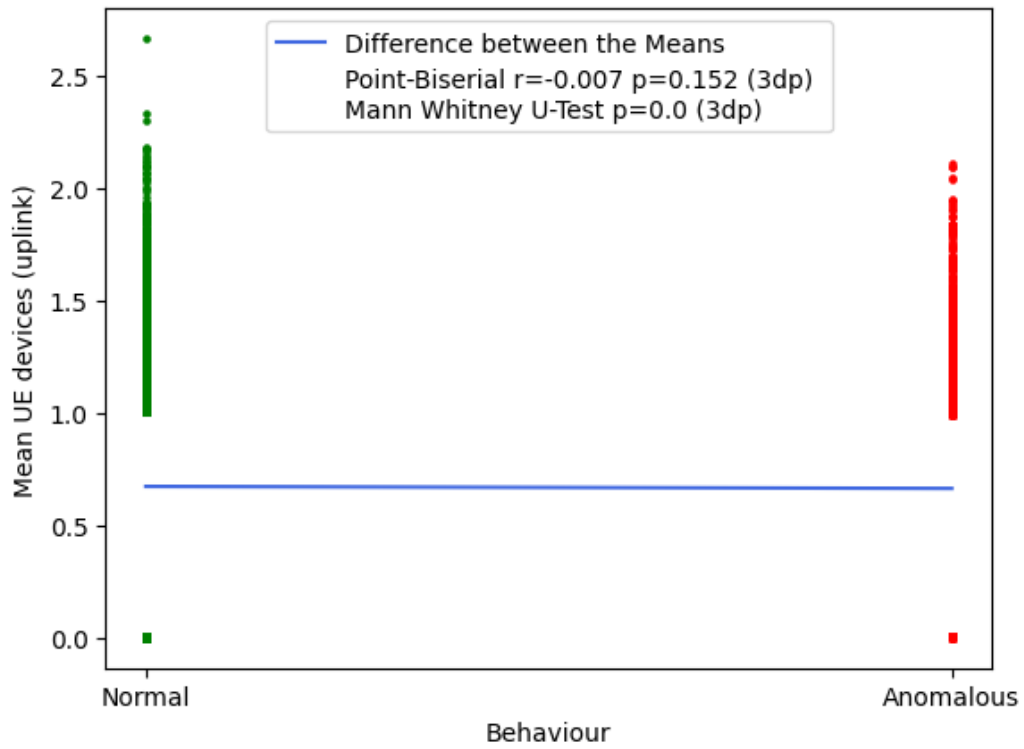


Figure 27. Scatter plot of mean UE devices (uplink) with hypothesis tests

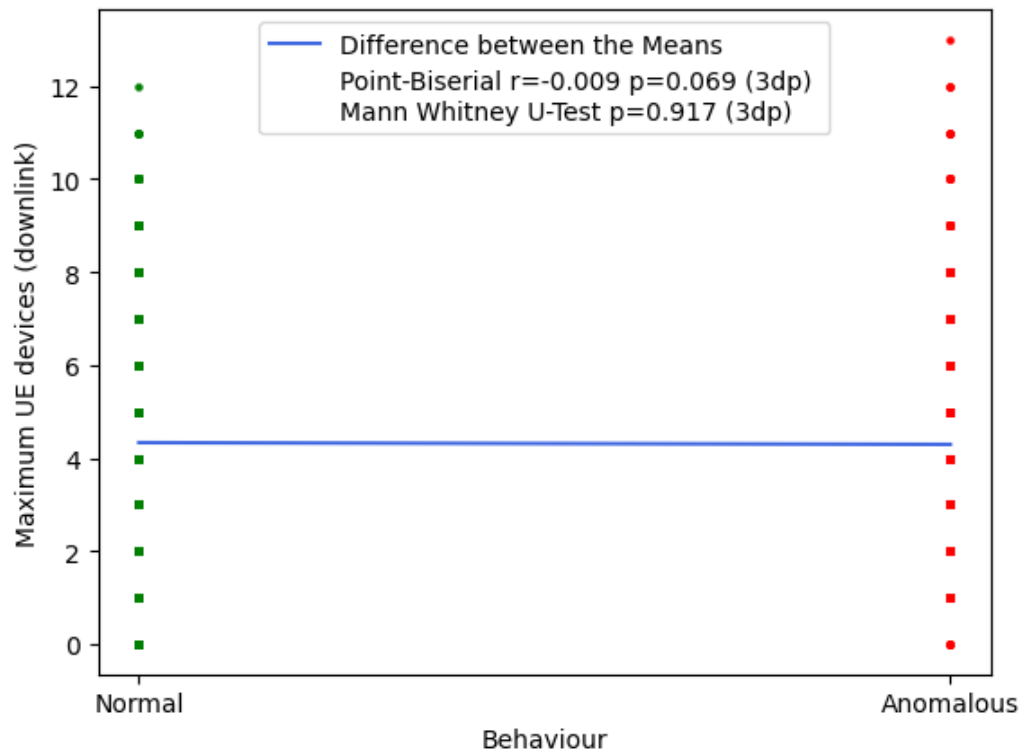
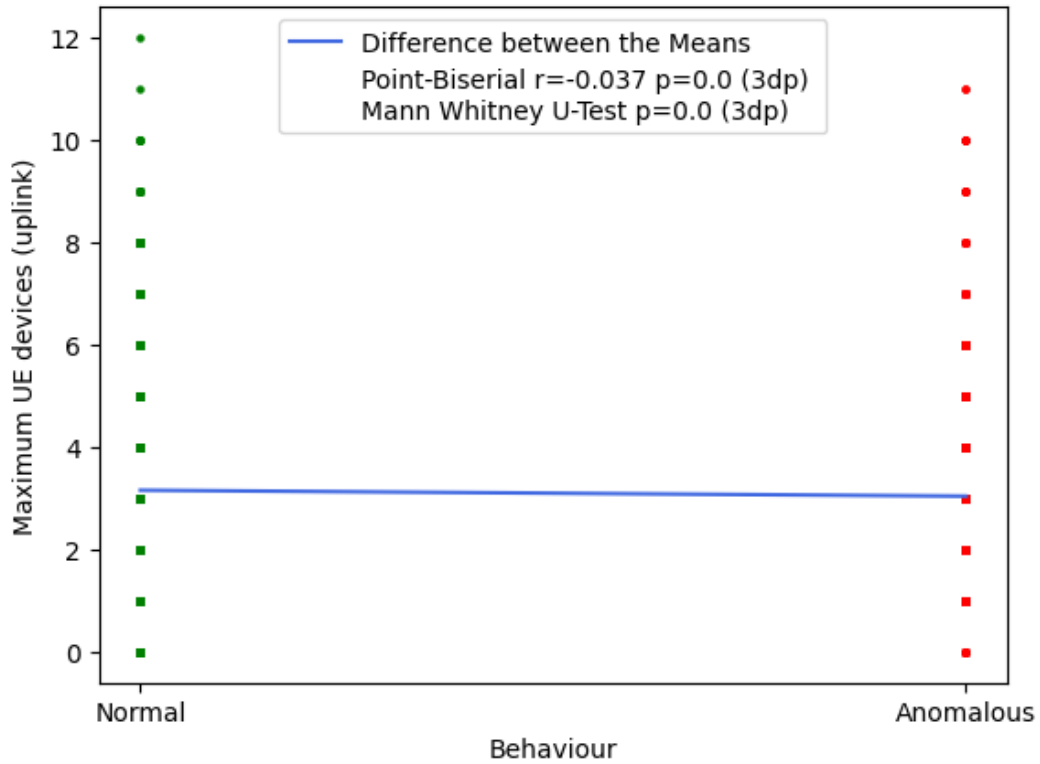
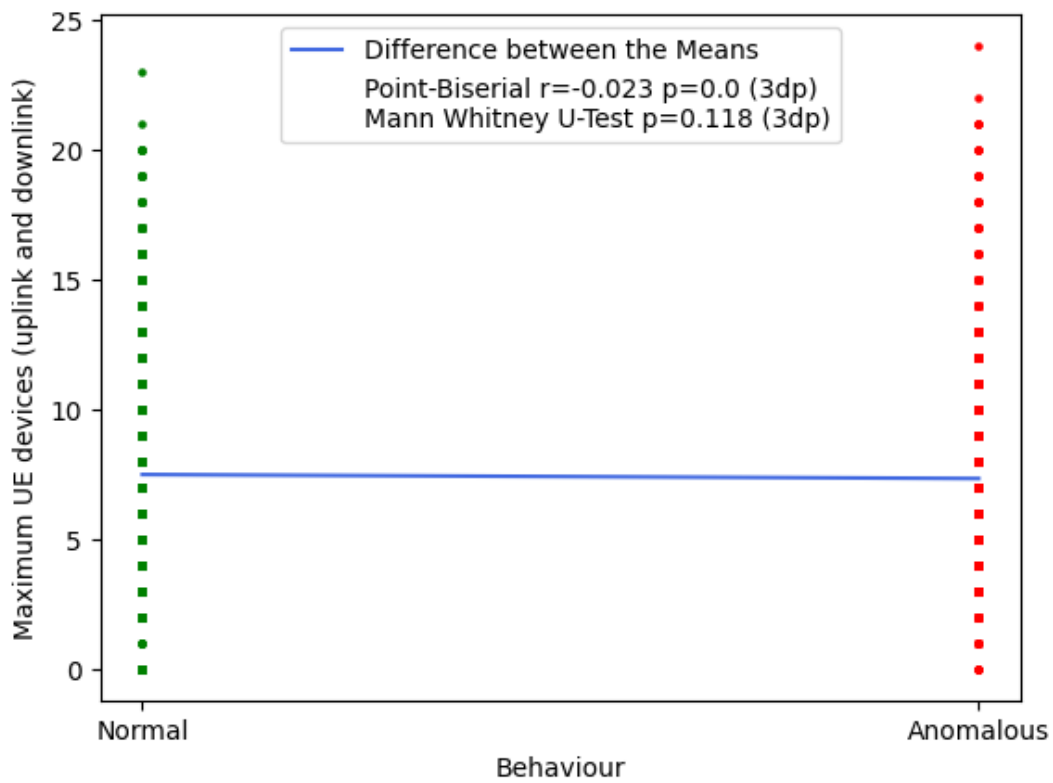


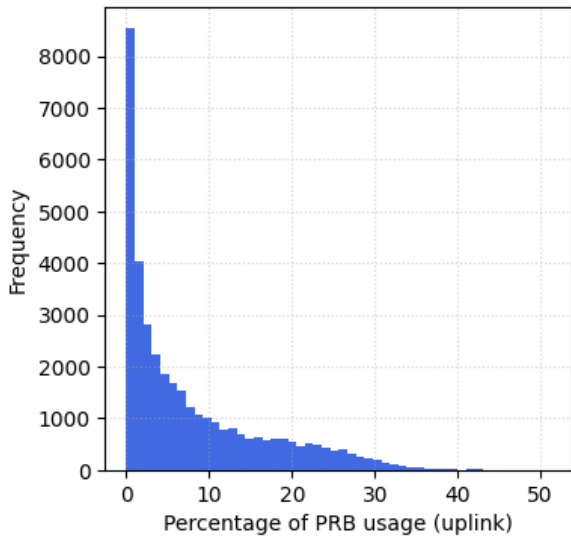
Figure 28. Scatter plot of maximum UE devices (downlink) with hypothesis tests



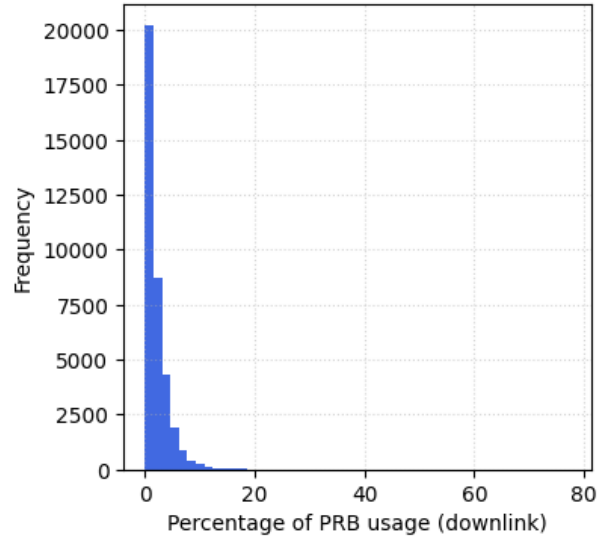
**Figure 29.** Scatter plot of maximum UE devices (uplink) with hypothesis tests



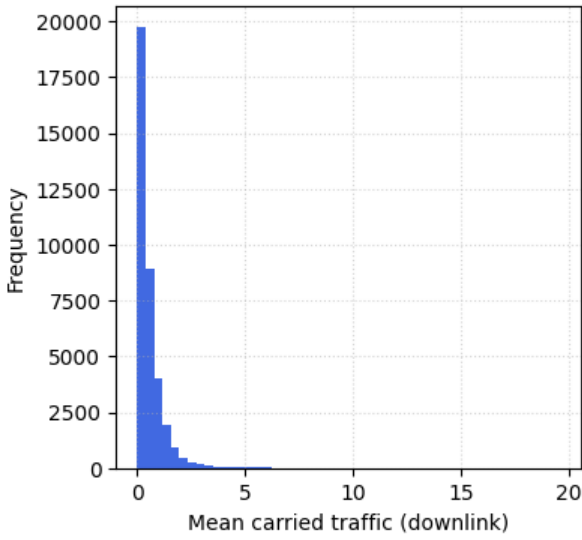
**Figure 30.** Scatter plot of maximum UE devices (uplink and downlink) with hypothesis tests



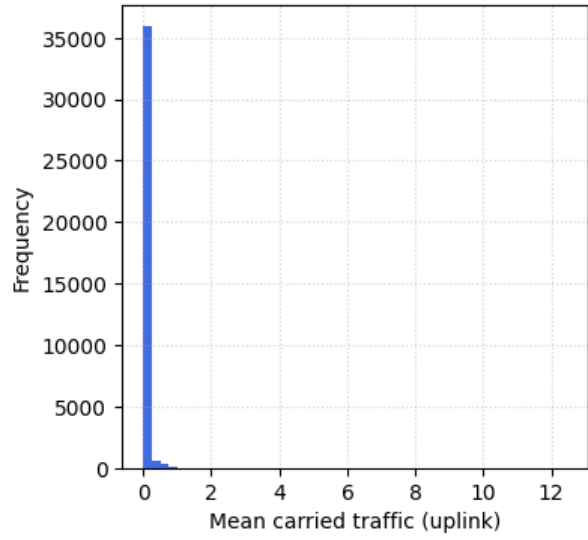
**Figure 3.** Histogram of percentage of PRB usage (uplink) (left)



**Figure 31.** Histogram of percentage of PRB usage (downlink) (right)

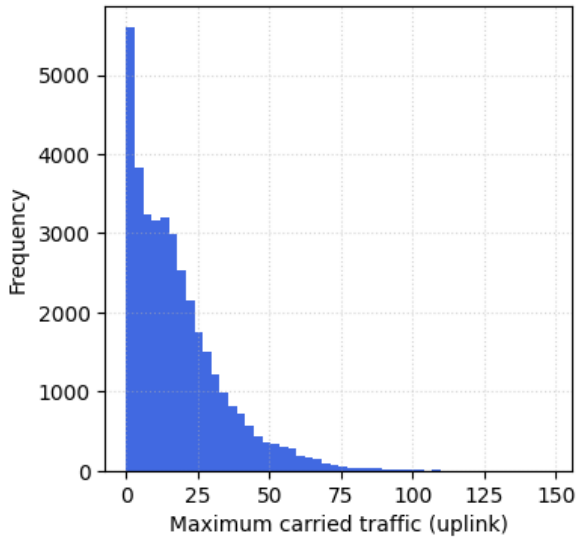


**Figure 32.** Histogram of mean carried traffic (downlink) (left)

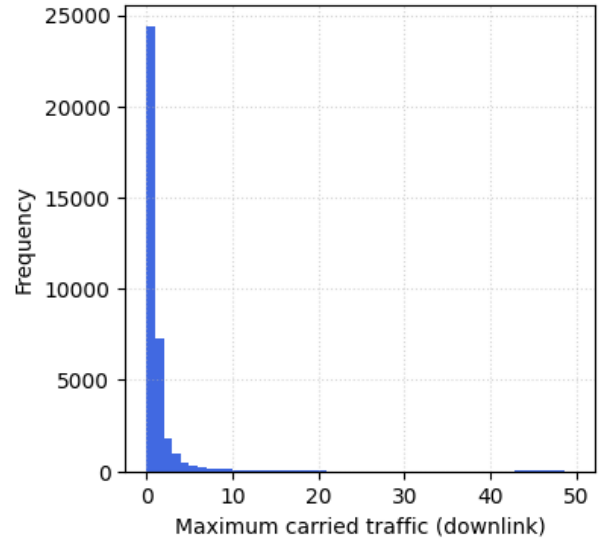


**Figure 33.** Histogram of mean carried traffic (uplink) (right)

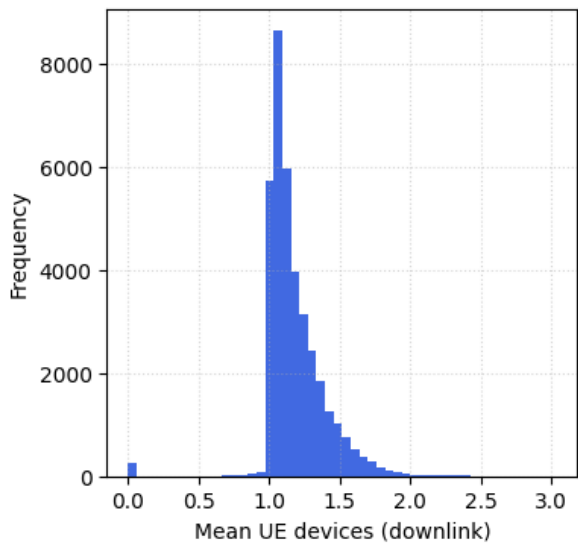




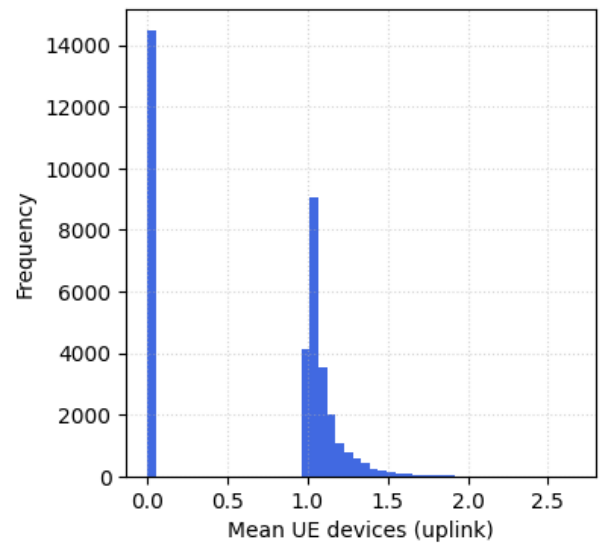
**Figure 34.** Histogram of maximum carried traffic (uplink) (left)



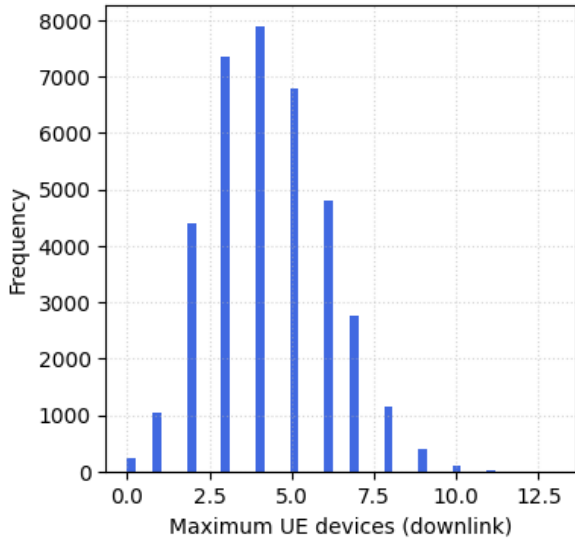
**Figure 35.** Histogram of maximum carried traffic (downlink) (right)



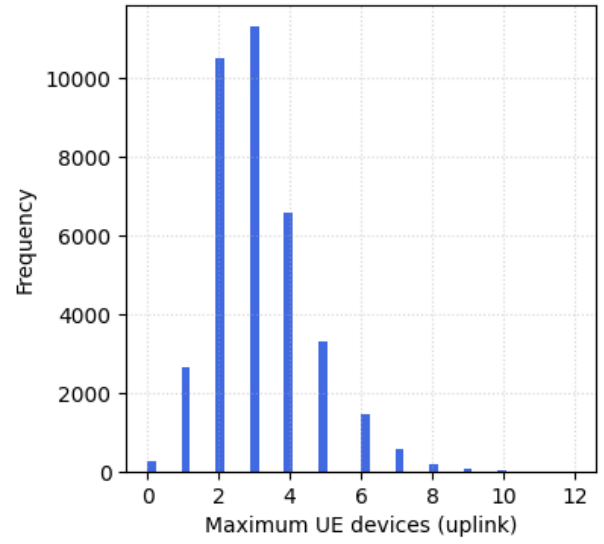
**Figure 36.** Histogram of mean UE devices (downlink) (left)



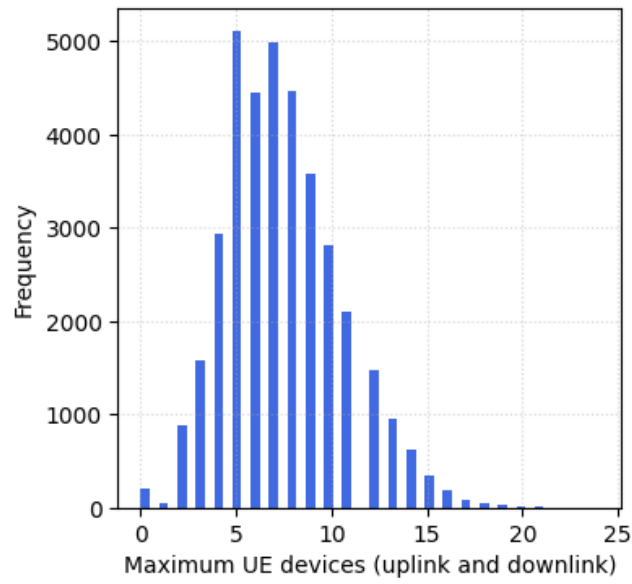
**Figure 4.** Histogram of mean UE devices (uplink) (right)



**Figure 5.** Histogram of maximum UE devices (downlink) (left)



**Figure 37.** Histogram of maximum UE devices (uplink) (right)



**Figure 38.** Histogram of maximum UE devices (uplink and downlink)