Cell Phone Anomaly Detection

Hong Tang

https://www.linkedin.com/in/hong-tang/

Summary and Recommendation

- **Why:** detect anomaly using ML to motivate optimization in base station
- **Findings:**
  - Decrease in resources usage and decrease in active user number tends to have high correlation with occurrence of anomaly
  - Feature engineering improves prediction accuracy
  - KNN imputation outperforms other 6 methods
  - Final selected model has robust AUC 0.94 in both training and testing set
- **Lessons Learned and Best Practices**
  - Focus on feature engineering: creating two additional features from CellNum significantly improve AUC
  - ML Pipeline makes feature engineering, modeling easy to read and maintain
- **Plan Forward:**
- Improve feature engineering;
- Investigate unsupervised classification

ML Process

Cycle 1

Feature EDA

Existing numerical features

Model Selection

Simple Logistic regression

Cycle 2

Feature EDA

Time features

Numerical features

Categorical features

Generate new features

Modeling OVART

imputation methods

Log transform

Outlier treatment

Cycle 3

Further feature engineering

Finetune RFC

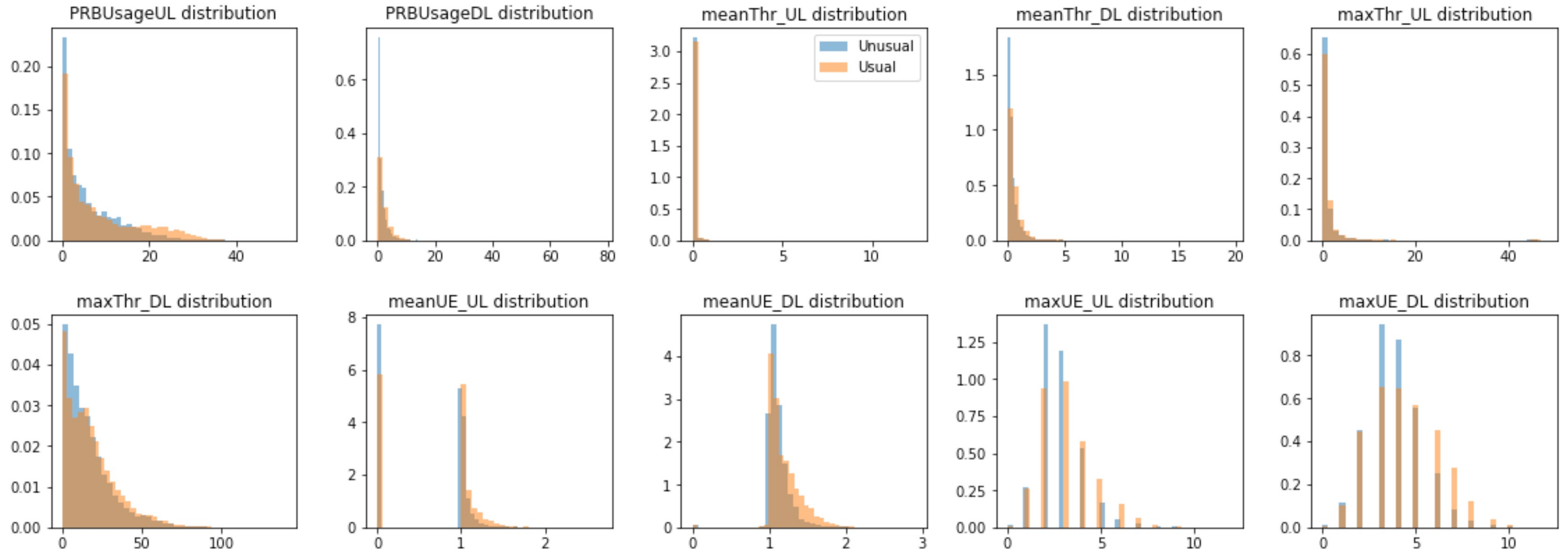
Model Deployment

Summary

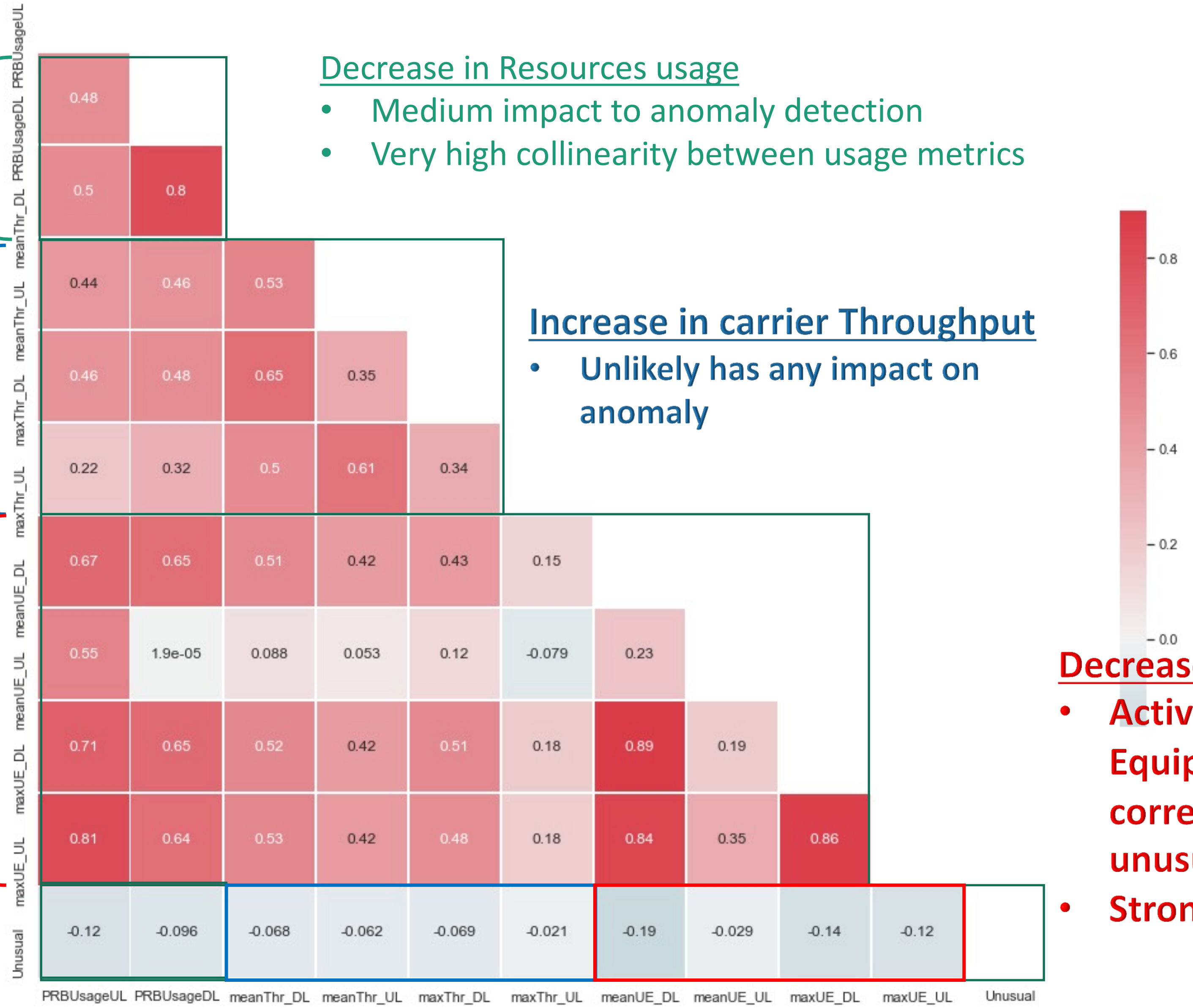
recommendation

Univariate Feature Analysis

- Skewed distribution for most features; log transformation is recommended to be applied prior to modeling
- In general, Unusual features(blue) tend to be more skewed compared to usual features(maroon)



Summary



Resources Usage

Carrier Throughput

Active User Equipment

Feature Correlation With Target

Decrease in Resources usage

- Medium impact to anomaly detection
- Very high collinearity between usage metrics

Increase in carrier Throughput

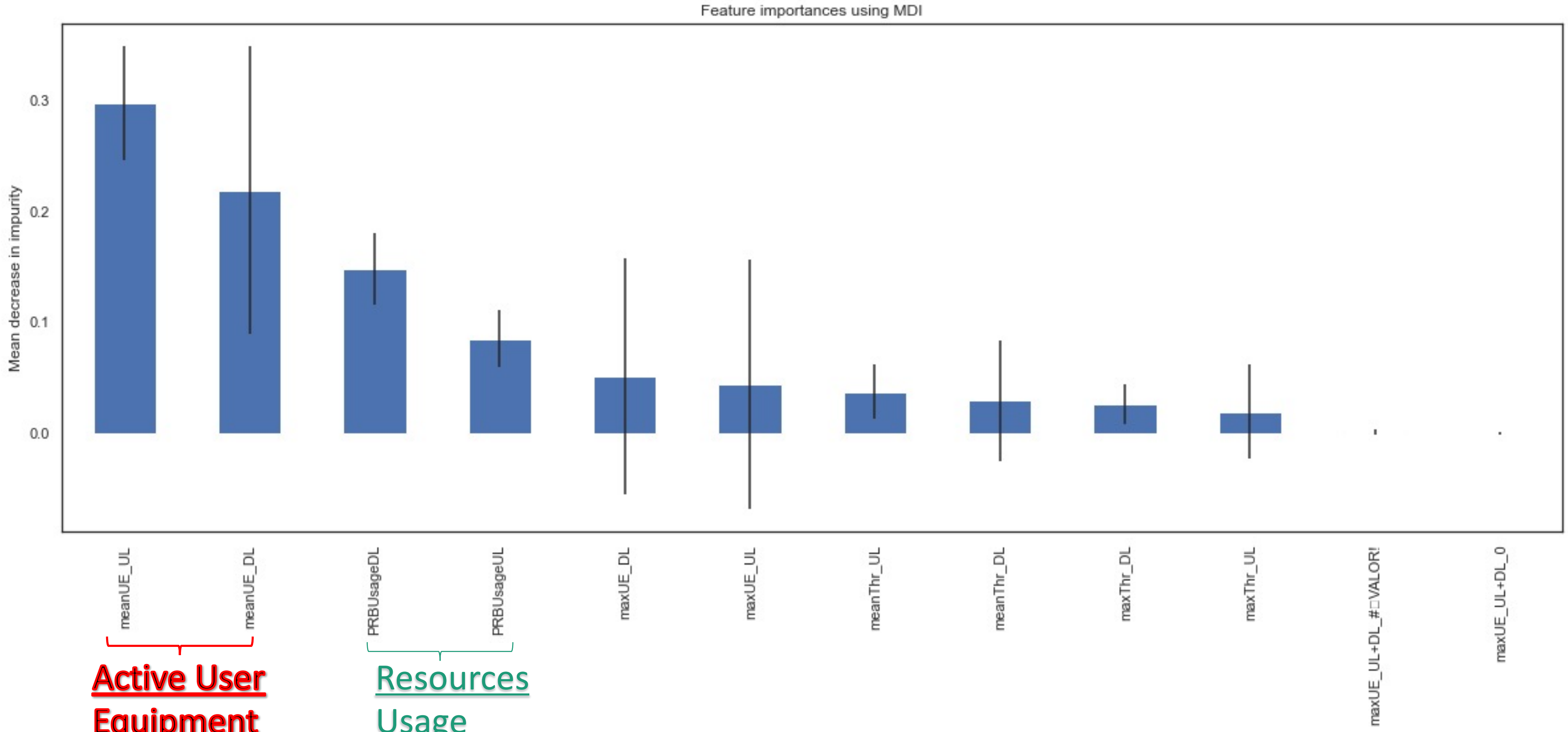
- Unlikely has any impact on anomaly

Decrease Active User Equipment

- Active Number ofUser Equipment=>Medium to strong correlation with anomaly or unusual behavior
- Strong colinearlity

Importance of Influence from RandomForrest Modeling

Active User Equipment and Resources Usage are big hitter for anomaly detection

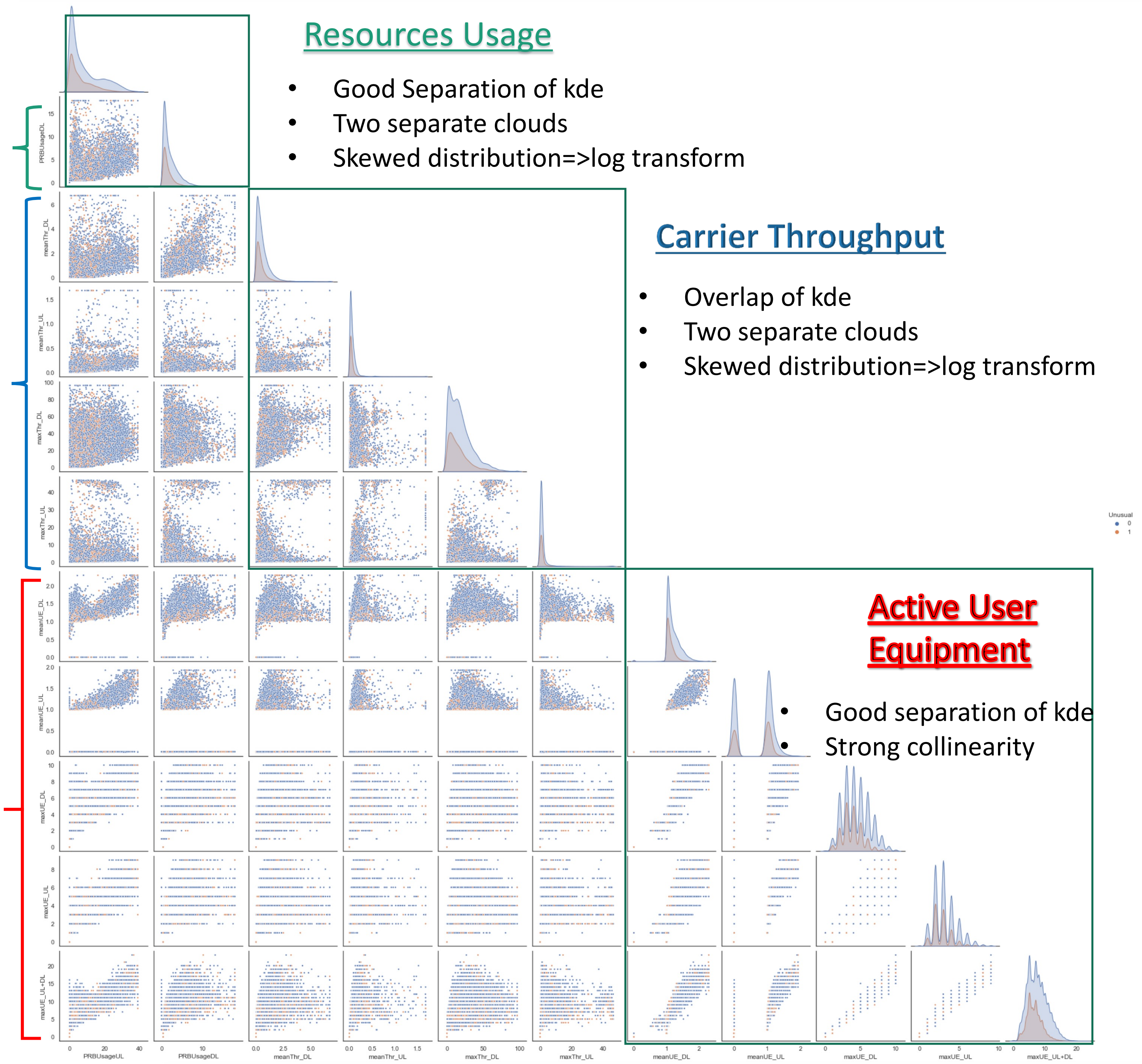


Active User Equipment

Resources Usage

Data Colineality with Unusal labels

- In general, KDE distribution with significant difference or cross plots with clear separation could be informative for classification
- Strong colineality between MaxUE_DL-MaxUE_UL-MeanUE_DL-MeanUE_UL, should be addressed in modeling stage
- Bimodal distribution on maxThr_UL, maxThr_DL(two clear cloud groups)



Resources Usage

- Good Separation of kde
- Two separate clouds
- Skewed distribution=>log transform

Carrier Throughput

- Overlap of kde
- Two separate clouds
- Skewed distribution=>log transform

Active User Equipment

- Good separation of kde
- Strong collinearity

Imputation Methods Sensitivity Study

- Per customer request, six imputation methods are tested;
- relating AUC indicates that there is no significant improvement of final AUC.
- The tests were done by holding other modeling parameters are the same, only vary imputation methods.

Methods	Training AUC	Test AUC
KNN Impute	0.917	0.903
Iterative Impute	0.915	0.903
Simple Impute	0.917	0.902
Median Impute	0.915	0.901
RandomSample Impute	0.914	0.899
Median Imputewith endpoints	0.914	0.898

Feature Engieering Steps Sensitivity Study

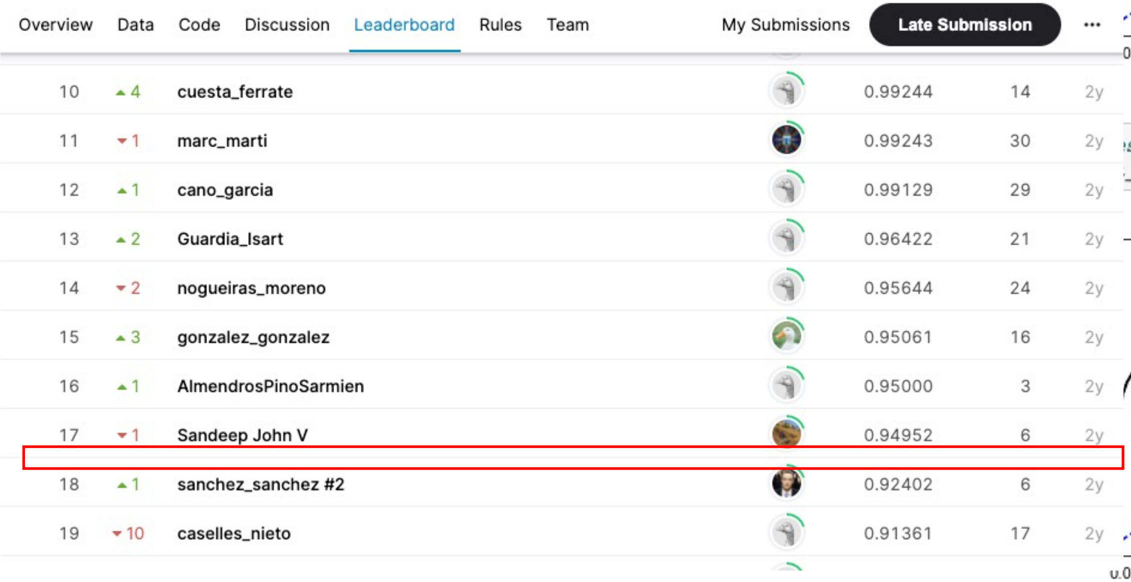
- IQR outlier treatment and Log Transformation were tested on the data set.
- The impact on final AUC is low; however, these process are still recommended since they allows input data to better follow statistics assumptions (normality)
- These feature engineering steps are implemented into data processing pipeline for streamlined deployment

Methods	Training AUC	Test AUC
Outlier Treatment	0.917	0.901
No Outlier Treatment	0.917	0.900
Log Transformation	0.917	0.902
No Log Tranformation	0.915	0.901

Model improvement from additional feature creation
AUC improves from 90-94%

- Feature engineering
  - XaLTE: X is the base station
  - a is the cell within base station

Benchmark Kaggle ranking 17 place



Overview | Data | Code | Discussion | Leaderboard | Rules | Team | My Submissions | Late Submission

10	▲4	cuesta_ferrate	0.99244	14	2y
11	▼1	marc_marti	0.99243	30	2y
12	▲1	cano_garcia	0.99129	29	2y
13	▲2	Guardia_Isart	0.96422	21	2y
14	▼2	nogueiras_moreno	0.95644	24	2y
15	▲3	gonzalez_gonzalez	0.95061	16	2y
16	▲1	AlmendrosPinoSarmien	0.95000	3	2y
17	▼1	Sandeep John V	0.94952	6	2y
18	▲1	sanchez_sanchez #2	0.92402	6	2y
19	▼10	caselles_nieto	0.91361	17	2y