Machine Learning Techniques using Key Performance Indicators for the Configuration Optimization of 4G Networks

Rui Miguel Maia Santos
Instituto Superior Técnico
Universidade de Lisboa
Email: rui.miguel.maia.santos@tecnico.ulisboa.pt

***Abstract*—This work focuses on the evaluation of the performance of a Long Term Evolution (LTE) network through the use of unsupervised automatic learning techniques. The main objective is to detect groups of cells that show similar performances and, consequently, identify the groups that perform below the desired level.**

Additionally, this work also aims to identify which cell configurations are associated with a better performance.

In order to fulfill the first objective, a methodology based on the application of clustering algorithms to features extracted from the original Key Performance Indicators (KPI) was developed. The following algorithms were tested: K-means, Expectation-Maximization (EM) using Gaussian Mixture Models (GMM), and Spectral Clustering.

Regarding the second objective, Fisher's exact test was used. This test evaluates the independence between the configuration values of the cells and the groups to which they belong.

Using this methodology it was verified that there is not a significant difference in the results obtained using the different algorithms. In the majority of the cases presented, only two groups of cells were identified: one group consisting essentially of the cells with the best performance and the other group containing the worst performing cells.

As far as the connection between configuration data and performance data is concerned, only one case, referring to a parameter associated with the subscription capacity of the cells, was detected.

I. Introduction

As the complexity of mobile networks increases due to the increase of performance requirements and number of subscribers, it becomes harder for the mobile network operators to not only maintain but also optimize the performance of those networks. As a result, mobile network operators are focusing more and more in creating tools and procedures that aim to not only assist radio engineers in the process of maintaining and optimizing the mobile networks, but also making the network itself more autonomous.

At the same time, Machine Learning (ML) and associated technologies are revolutionizing the way current systems work, by allowing machines to learn from available data and perform actions that would normally be taken by humans. Thus, mobile network operators can take advantage of this techniques to assist in the network management and automation process, increasing its efficiency and reducing costs.

The data used to evaluate the performance of a network is composed by KPIs. Some work related with the application of ML techniques to KPIs collected from mobile networks has already been developed, as in [1], where unsupervised learning techniques are used to automatically detect faults in a LTE network

The main objective of this work is to develop a model that is able to evaluate the performance of a LTE network, based on KPIs collected from the network. The system should apply unsupervised learning techniques in order to find cell groups that present similar performances. Furthermore, the system should classify those groups regarding the performance of the cells that constitute them, depending on levels of performance specified by the mobile network operator.

Additionally, this work also aims to analyze the relationship between the configuration parameters and the obtained groups of cells performance.

This work is organized as follows. Section II gives an overview of the Performance Management (PM) and Configuration Management (CM) data. Section III introduces the algorithms and validation metrics used. Section IV presents the methodology used. Section V summarizes the results regarding the network performance evaluation. Section VI presents the results regarding the correlation between performance and configuration. Lastly, conclusions are drawn in Section VII.

II. PM and CM Data

The data used in this work was collected from a real LTE network, deployed in a urban environment. It encompasses both PM and CM data.

Each site may support from one up to three of the following frequency bands: L800 (800 MHz), L1800 (1800 MHz) and L2600 (2600 MHz).

A. PM Data

The PM data is composed by several KPIs, which were collected periodically, every 15 minutes during 10 days, for each cell. Thus, the data collected for each cell is a multivariate time-series.

In order to achieve the goals of this work it is necessary to understand what each KPI represents and select the ones that are more relevant to analyze the network performance. Furthermore, each selected KPI is classified according to KPI class to which it belongs: Accessibility, Integrity or Availability. The description of each KPI is based on [2].

1) Accessibility:

- CB_RACH_fail% - indicates how often, in a Contention Based (CB) Random Access Channel (RACH) procedure, a transmitted RaMsg2 does not result in a successfully received RaMsg3;
- Added_E_RAB_Estab_fail% - the fail rate for end-user services that are carried by E-UTRAN Radio Access Bearer (E-RAB)s included in the E-RAB setup procedure;
- Init_E_RAB_Estab_fail% - the fail rate for end-user services that are carried by E-RABs included in the Initial User Equipment (UE) Context setup procedure;
- RRC_Estab_fail% - the fail rate for Radio Resource Control (RRC) connections establishment;
- S1_Estab_fail% - the fail rate for the establishment of signaling connections over the S1 interface.

2) Integrity:

- DL_Throughput_per_UE(Mbps) - the average throughput per user in the downlink, measured in Mbit/s;
- UL_Throughput_per_UE(Mbps) - average throughput per user in the uplink, measured in Mbit/s;
- DL_Pdcp_Cell_Tput(Mbps) - average cell throughput with respect to the Packet Data Convergence Protocol (PDCP) layer in the downlink, measured in Mbit/s;
- UL_Pdcp_Cell_Tput(Mbps) - average cell throughput with respect to the PDCP layer in the uplink, measured in Mbit/s;
- DL_MAC_Cell_Tput(Mbps) - average cell throughput with respect to the Medium Access Control (MAC) layer in the downlink, measured in Mbit/s;
- UL_MAC_Cell_Tput(Mbps) - average cell throughput with respect to the MAC layer in the uplink, measured in Mbit/s.

3) Availability:

- CellAvailMan_perc - the percentage of time that a given cell is available with respect to the time that has been disabled due to a reconfiguration request performed by the operator;
- CellAvailAuto_perc - the percentage of time that the cell is available with respect to the time that has been disabled due to a fault;
- CellAvail_perc - the overall percentage of time that the cell is available.

The PM data goes through a preprocessing stage where any artifacts and null values that the data may contain are removed.

Since the different frequency bands serve may serve different purposes, it was decided to split the PM dataset per frequency band. Each resulting dataset is further divided per KPI class, so the performance of the network is evaluated for each frequency band and KPI class, individually.

B. CM Data

The CM data simply contains the configuration parameters for each cell.

The original CM dataset contained 24 distinct configuration features. However, since the performance of the network is evaluated per frequency band, it was verified that, for each frequency band, there were only 10 features that presented two or more distinct values. Given that one of the goals of this work is to analyze the link between the configuration parameters and the performance of the obtained clusters, only those 8 configuration parameters were considered. They are the following:

- **NOOFPUCCHCQIUSERS** - number of Channel Quality Indicator (CQI) resources available on the Physical Uplink Control Channel (PUCCH);
- **NOOFPUCCHSRUSERS** - number of Scheduling Request (SR) resources available on the PUCCH;
- **NOCONSECUTIVESUBFRAMES** - number of consecutive downlink sub-frames with positioning reference signals;
- **CELLSUBSCRIPTIONCAPACITY** - normalized subscription capacity of the cell. The value represents the total capacity of the cell used for traffic load balancing purposes;
- **PDCCHCFIMODE** - controls the Control Format Indicator (CFI) used for the control region;
- **LBTPNONQUALFRACTION** - fraction of non-qualified UEs at UE selection for throughput aware load balancing;
- **LBTPRANKTHRESHMIN** - minimum threshold for the relative gain at throughput aware load balancing;
- **TRANSMISSIONMODE** - defines the Transmission Mode that shall be used for the UEs that are connected to the cell;
- **INTERFERENCETHRESHOLDSINRCLPC** - Threshold value for measured noise plus interference level. If measured noise plus interference is higher than interferenceThresholdSinrClpc, then the Signal to Interference and Noise Ratio (SINR)-based UL Closed Loop Power Control (CLPC) can be considered;
- **RXSINRTARGETCLPC** - SINR target value for the Physical Uplink Shared Channel (PUSCH) SINR-based CLPC;

III. Clustering background

In order to find groups of cells with similar performances, unsupervised clustering algorithms were used. Since different clustering algorithms have distinct underlying principles and assumptions, it is unclear which one fits the available data the best. Consequently, different algorithms were considered: K-means, EM using GMMs and Spectral Clustering.

Typically, the optimal number of clusters is unknown. In such cases Clustering Validity Indexes (CVI) can be used.

A. Clustering Algorithms

K-means [3] is a clustering algorithm that finds the clusters by minimizing the sum of squared distances between each cluster centroid and the respective cluster points, using a greedy iterative approach. K-means is particularly suited for convex shaped clusters, namely spherical shaped clusters, or circular in two dimensions.

In EM clustering using GMMs [3], it is assumed that each cluster follows a multivariate normal distribution, characterized by a mean and a covariance matrix. The EM approach estimates the mean and covariance matrix associated with the multivariate normal distribution of each cluster by maximizing the probability of the observed points belonging, in fact, to the estimated multivariate normal distribution. Since each cluster is described by two parameters, mean and variance, this approach is more flexible when compared to K-means, being able to find elliptical shaped clusters.

The Spectral Clustering algorithm [4] is based on a similarity graph, which is created using the Gaussian Radial Basis Function (RBF) kernel as similarity measure [5]. A graph Laplacian is then computed from the similarity matrix. From the graph Laplacian, and assuming prior knowledge of the desired number of clusters, k, the first k eigenvectors are computed and used by the K-means to derive the final cluster partition. The Spectral Clustering algorithm is able to find non-convex clusters, as it is neither tied to any ideal cluster spherical shape, as K-means, nor to elliptical shapes, as the GMM approach.

B. Clustering Validation

Internal CVIs can be used to find the optimal input parameters when the ground truth of the data is not available. In this work, only internal CVIs that evaluate both the separation and compactness of the clusters were considered. The compactness is related to how closely related the objects in a cluster are, while the separation measures how well-separated the clusters are. The considered CVIs were the following [6]: Silhouette, Calinski-Harabasz (CH), Davies-Bouldin Index (DB), I index, Dunn's index and S_Dbw.

IV. Proposed Method

The proposed method consists in a feature-based approach, where new features are extracted through the comparison of each time-series against desired targets for each KPI. Even though this approach results in the loss of information regarding the cell behavior, in the time domain, it allows to evaluate each cell overall performance level and group the cells accordingly. The methodology flowchart is illustrated in Figure 1.

A. Feature Engineering

The feature engineering step corresponds to the process of extracting new features from the original data. The process of feature extraction consists in defining a set of target values $T = [T_1, T_2, ..., T_M]$, where each T_p corresponds to the target value for the p^{th} KPI, from the M considered KPIs. Then, for

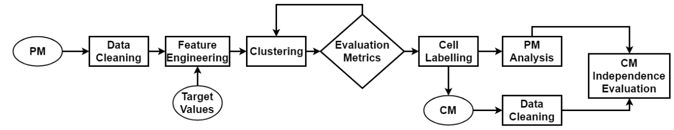


Figure 1: Methodology flowchart.

each cell, c, and KPI, p, each measure, x_{cp_t}, acquired in the instant t of the time-series of size N, is compared against the defined target, T_p, for that KPI.

If the KPI is from the Accessibility group then the target is said to be fulfilled if x_{cp_i} is lower than the target, since each Accessibility KPI in the PM dataset corresponds to a fail rate. The value of x_{cp_i} is then changed to 0 or 1 accordingly:

$$x_{cp\_t} = \begin{cases} 1, & \text{if } x_{cp\_t} \leq T_p \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Contrarily, for an Integrity or Availability KPI, it is desired that x_{cp_i} is greater than the target value. Thus, the new value for x_{cp_i} is given by:

$$x_{cp\_t} = \begin{cases} 1, & \text{if } x_{cp\_t} \geq T_p \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Thereupon, for each cell, c, and KPI, p, the ratio between the number of times that KPI satisfies its defined target and the total number of times the KPI was measured is computed:

$$feature_{cp} = \frac{1}{N} \sum_{t=1}^{N} x_{cp\_t} \tag{3}$$

A visual interpretation of the feature engineering process is shown in Figure 2. Considering one cell, the KPI RRC_Estab_fail% is plotted over the time that was measured, against the target value for that KPI. The new feature generated from this KPI, and all the remaining extracted features for that matter, can be interpreted as the time period over the total considered time that the defined target for the original KPI was satisfied. Since the KPI shown in Figure 2 corresponds to a fail percentage then satisfying the target means that the measured value for the KPI is below its target.

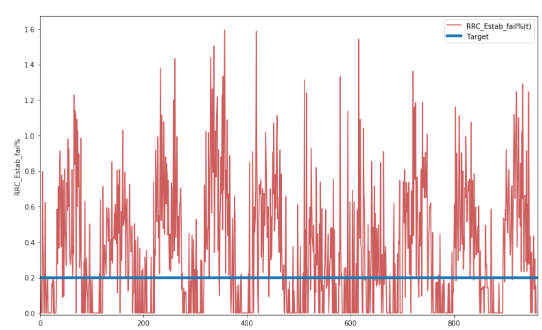


Figure 2: Feature engineering example.

_MECONTEXT	_VSDATAEUTRANCELLFDD	STARTDATE	Init_E_RAB_Estab_fail%
7.0	1	2018-05-07 00:00:00	0.205761
	1	2018-05-07 00:15:00	0.000000
	1	2018-05-07 00:30:00	0.529101
	...	...	...
	1	2018-05-16 23:15:00	0.180832
	1	2018-05-16 23:30:00	0.202840
	1	2018-05-16 23:45:00	0.373832

(a) PM data before the feature engineering process.

_MECONTEXT	_VSDATAEUTRANCELLFDD	Init_E_RAB_Estab_fail_target_compliance_ratio
7.0	1	0.56875

(b) PM data after the feature engineering process.

Figure 3: PM data before and after the feature engineering process.

Intuitively, every value obtained for the generated features is comprised between 0 and 1, with 1 representing a cell that was compliant with the target set for a KPI during the total period of time in which the time series were obtained.

As a result, for each pair, cell and KPI, the correspondent time series is converted into a single value, thus the resulting dataset is composed by only one row per cell. This is exemplified for one KPI in Figure 3.

B. Clustering Stage

The next step consists in applying a clustering algorithm to the dataset generated in the previous stage in order to find groups of cells that present similar behaviours. This is an iterative process:

1) Choose the clustering algorithm;
2) Run the clustering algorithm with different input parameters and evaluate the clustering result using the Kolmogorov-Smirnov test;
3) Select the set of input parameters that give the best clustering results using the metrics presented in Section III-B.

Regarding step 2 of the above process, each pair of clusters must have a distinct statistical behaviour for at least one feature in order to consider the clustering result relevant [1]. This is verified using the two sample Kolmogorov-Smirnov test [7], to test if the observed values for a feature of two different clusters are generated by the same distribution (null hypothesis). The null hypothesis is rejected when the resulting *p*-value is lower than the selected significance level. If there is at least one pair of clusters that present a similar statistical behaviour for every feature, meaning that the null hypothesis is never rejected, then the number of clusters that originated that partition is automatically discarded. The significance level used to test the null hypothesis was 0.01.

Each one of the clustering algorithms presented in Section III-A requires the number of clusters to be specified beforehand. In this work, it was chosen to run the chosen clustering algorithm for set of number of clusters that ranges from 2 to 8, to ensure that a wide enough range of possible partitions is analyzed by the system.

In step 3, the set of input parameters that give the best partitioning of the data is attained using an election mechanism. Each CVI presented in Section III-B contributes with one vote, and the set of input parameters that gets more votes is the one selected.

Once the set of input parameters that give the best partitioning result, according to the election mechanism, is identified, the algorithm is performed again with the optimal input in order to assign a label to each cell, identifying to which cluster that cell belongs. Thus, this is called the labelling stage.

C. PM Analysis

After each cell has been labelled according to the cluster to which it belongs, PM analysis is performed. This stage includes the following actions:

- **Data visualization** - t-Distributed Stochastic Neighbour Embedding (t-SNE) [8] is used for dimensionality reduction and the resulting points are plotted in a two-dimensional space for visual inspection;
- **Cluster scoring and classification** - a score for each cluster is computed. A higher score indicates a better performance. The clusters are then classified based on the obtained score;
- **Feature distribution analysis** - the distribution of each feature per cluster is plotted allowing the user to gain insight about the performance of each cluster regarding each KPI.

The cluster score is a weighted average of the scores computed for each feature considered, and is given by:

$$score_{cluster} = \frac{1}{M}\sum_{p=1}^{M} \alpha_p score_{feature\_p} \tag{4}$$

where α_p corresponds to a weight given to $feature_p$ and $\sum_{p=0}^{M} \alpha_p = 1$ and $score_{feature\_p}$ is the score for the p^{th} feature of the considered cluster and is given by:

$$score_{feature} = P(feature \geq time_{threshold}) \tag{5}$$

where $time_{threshold}$ is a parameter that can take values between 0 and 1 and is specified by the user depending on the level of exigency desired. This score can be interpreted as the probability of a cell in the cluster being compliant with the target set for the feature being evaluated, for a period of time above the $time_{threshold}$. This is illustrated in Figure 4, where the $time_{threshold}$ was set at 0.8. The intersection of the vertical red line, representing the threshold, with the Empirical Cumulative Distribution Function (ECDF) of a cluster gives the value of $P(feature < time_{threshold})$. It is straightforward to see that the probability of a cell belonging to cluster 1 being compliant with the target defined for the RRC_Estab_fail% for more than 80% of the total period of time of the original time-series, is greater than for a cell belonging to cluster 0. This probability gives the $score_{feature}$ for each cluster.

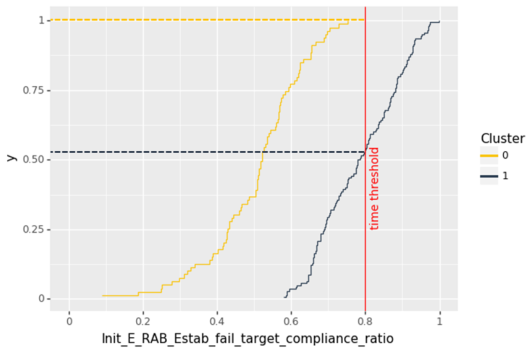


Figure 4: Visual interpretation of the cluster feature score.

Through the combination of the cluster scoring and the feature distribution analysis, one is able to immediately tell which clusters have better performance and what are the most distinct performance features.

D. CM Independence Evaluation

Regarding the CM features, the goal is to find the set of configuration parameters that are most distinct between clusters and evaluate if, for the class of KPIs being analyzed, there is a correlation between them and the performance of the cluster.

The process of identifying the most distinct configuration parameters is based on the Fisher's exact test of independence [9] while the process of evaluating if there is a correlation between the configuration parameters and the performance relies, at this stage, on the expertise of radio network engineers.

V. PM Results

This section presents the results obtained by applying the methodology described in Section IV. Given the high amount of results due to the division per frequency band a KPI class, only a few selected cases are discussed.

Additionally, the results for different clustering algorithms are summarized and compared. Only the Accessibility and Integrity classes were tested, as the Availability class, due to the reduced number of features and nature of the same, is quite straightforward to analyze.

The clustering and the t-SNE algorithms used are from the Scikit-learn library [10].

The scores obtained for each cluster using Equation 4 were computed with $time_{threshold} = 0.8$, unless explicitly stated otherwise.

A. Targets

A reasonable choice regarding the targets for the KPIs is, perhaps, the most influential step when applying the methodology presented in Section IV, as the dataset resulting from the feature engineering step depends on that choice.

The targets should be defined taking into account the knowledge of experts and the performance level that the operator wants to provide.

The targets used in this work for the Accessibility class of KPIs are presented in Table I.

Since all cells operating in the L800 frequency band have a 10 MHz bandwidth, while all cells operating in both L1800 and L2600 have a 20 MHz bandwidth, two sets of Integrity targets, shown in Table II, were defined. Due to the inherent difficulties in defining throughput targets, as throughputs depend on multiple factors and a low throughput is not necessarily bad, the approach taken for this case was to compute, for each Integrity KPI, the 25^{th} percentile with respect to the 10 MHz and 20 MHz bandwidths.

For each one of the Availability KPIs, the target was set at 99.7%.

KPI	**Target [%]**
CB_RACH_fail%	5
Added_E_RAB_Estab_fail%	0.01
Init_E_RAB_Estab_fail%	0.05
RRC_Estab_fail%	0.25
S1_Estab_fail%	0.25

Table I: Targets for Accessibility KPIs.

	Target [Mbps]	
KPI	**L800**	**L1800, L2600**
DL_Tput_per_UE(Mbps)	12.6	15.8
DL_Pdcp_Cell_Tput(Mbps)	7.2	8.6
DL_MAC_Cell_Tput(Mbps)	8.1	9.5
UL_Tput_per_UE(Mbps)	0.4	0.5
UL_Pdcp_Cell_Tput(Mbps)	0.59	0.68
UL_MAC_Cell_Tput(Mbps)	0.75	0.99

Table II: Targets for Integrity KPIs per frequency band.

B. Clustering using K-means

K-means was the first algorithm used to cluster the PM data due to its simplicity.

The performance evaluation results presented in this section include the Accessibility class for L800, and the Integrity and Availability classes for L1800.

L800

The PM dataset for the L800 frequency band is constituted by 219 cells.

Accessibility: Applying the K-means algorithm to the Accessibility data resulting from the feature engineering step with the Accessibility targets presented in Table I, the most voted number of optimal clusters was $k = 2$. The clusters can be observed in Figure 5.

There are 79 cells belonging to cluster 0 while cluster 1 contains 140 cells.

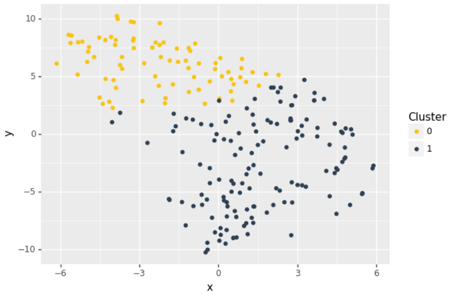
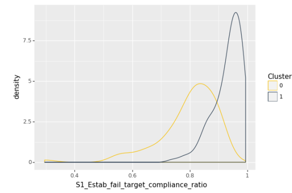
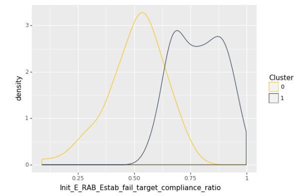


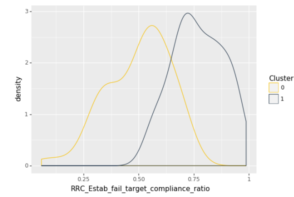
Figure 5: Clustering visualization for K-means in L800 (Accessibility).



(a) S1_Estab_fail_target_compliance_ratio.

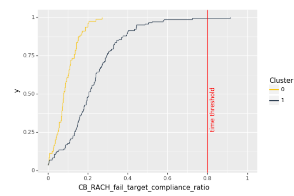


(b) Init_E_RAB_Estab_fail_target_compliance_ratio.

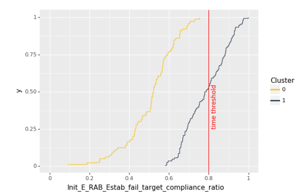


(c) RRC_Estab_fail_target_compliance_ratio.

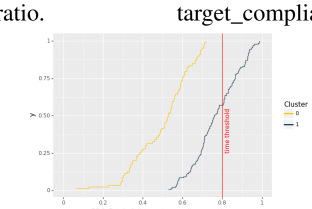
Figure 6: Histograms of Accessibility features for K-means in L800.



(a) CB_RACH_fail_target_compliance_ratio.



(b) Init_E_RAB_Estab_fail_target_compliance_ratio.



(c) RRC_Estab_fail_target_compliance_ratio

Figure 7: ECDFs of Accessibility features for K-means in L800.

The operator may choose to qualitatively classify the overall cluster performance based on the obtained score. In this work, to ease the result analysis, the following qualitative levels of performance were considered: *unsatisfactory*, for a cluster score below 0.25; *below average*, for a score between 0.25 and 0.5; *average*, for a cluster score between 0.5 and 0.75; and *above average* for a cluster score greater than 0.75. The quantitative and qualitative scores obtained for the clusters are presented in Table III.

Cluster	**Number of Cells**	**Score**	**Classification**
0	79	0.32	Below average
1	140	0.57	Average

Table III: Cluster classification for K-means in L800 (Accessibility).

Even though the main goal of the Kolmogorov-Smirnov test is to guarantee that the obtained clusters are statistically significant, it is also possible to gain knowledge regarding the features that contribute the most for the attained clustering partitions. The features that have a lower p-value for a pair of clusters are the ones that better explain the difference between those clusters. Furthermore, the visual comparison of the histograms of each feature gives a more intuitive understanding of the features that are more distinct between clusters. Figure 6 shows the histograms of each cluster for the features that presented the lowest p-values for the Kolmogorov-Smirnov test.

Even though Figure 6 presents the three most distinct features between the two clusters, these features are not necessarily the ones that better explain the obtained scores for the clusters. The visualization of the ECDFs of the features for each cluster allows one to understand which are the features that most contribute to lower the score of the clusters. In Figure 7 are presented the ECDFs of the features that better explain the scores presented in Table III.

It can be verified that the cells of both clusters have a poor performance regarding the CB_RACH_fail% KPI. For the other two features it can be seen that approximately half of the cells belonging to cluster 1 are compliant with the $time_{threshold}$ while none of the cells belonging to cluster 0 are compliant with that same value.

Since the behaviour that each cell presents is heavily influenced by its location, it can be interesting to observe if there are geographical areas with a high density of similarly performing cells. Figure 8, obtained with [11], shows where the cells are located, with each cell being identified with the color of the cluster to which it belongs.

It is possible to identify a few areas where the concentration of cells belonging to one of the clusters, is predominant. Moreover, there is a specific area, highlighted in the figure with a blue circle, which is mainly populated with cells from cluster 0, thus being a zone with accessibility issues for the 800 MHz frequency band.

Figure 8: Clusters geographical distribution for Accessibility KPIs (L800).

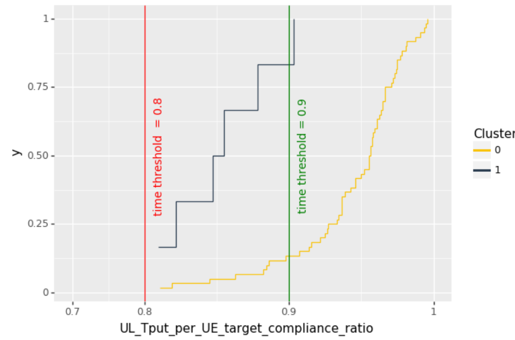


Figure 9: Score comparison for different $time_{threshold}$ values.

L1800

The PM dataset for the L1800 frequency band contains 69 cells.

Integrity: In this scenario, two clusters were obtained. Cluster 1 only contains 9 cells while cluster 0 contains 60 cells, thus being the most representative cluster of the dataset. Table IV shows the score for each cluster as well as its overall performance classification, regarding the Integrity class.

Cluster	Number of Cells	Score	Classification
0	60	0.99	Above average
1	9	0.72	Average

Table IV: Cluster classification for K-means in L1800 (Integrity) and $time_{threshold} = 0.8$.

It is interesting to see how the classification of a cluster changes by changing the value of $time_{threshold}$. Let us consider a more demanding cluster evaluation by setting $time_{threshold} = 0.9$ instead of $time_{threshold} = 0.8$, which was the value used thus far. The scores and classification obtained in this case are presented in Table V.

Cluster	Number of Cells	Score	Classification
0	60	0.83	Above average
1	9	0.25	Unsatisfactory

Table V: Cluster classification for K-means in L1800 (Integrity) and $time_{threshold} = 0.9$.

Comparing Tables IV and V, it can be verified that cluster 0 still has an above average performance. However, the score of cluster 1 drops drastically after changing the value of $time_{threshold}$, being classified as unsatisfactory.

Figure 9 exemplifies the difference in the score obtained for the feature UL_Tput_per_UE_target_compliance_ratio with $time_{threshold}$ set to 0.8 and 0.9.

It can be verified that the ECDF of cluster 1 is shifted to the right of the red line, representing $time_{threshold} = 0.8$, thus the score obtained with respect to that value is 1. Contrarily, for $time_{threshold} = 0.9$, identified with the green line, the score for cluster 0 is nearly 0, with only one cell being compliant with the target for a period of time above the $time_{threshold}$.

Availability: Regarding the Availability KPIs, it could be argued that only the Cell_Avail_perc could be used, as this KPI presents the overall availability of the cell, thus containing the information provided by the other two KPIs. However, this KPI also provides insight about the cell sleep mode. Thus, if one would analyze the availability of the cells, based solely on the Cell_Avail_perc KPI, it could be mislead into thinking that the cell was unavailable due to a fault or a reconfiguration request when in fact it was in sleeping mode.

The optimal number of clusters obtained for each CVI was $k = 2$.

The resulting clusters can be visualized in Figure 10. Cluster 0 contains 44 cells while cluster 1 contains 25 cells.

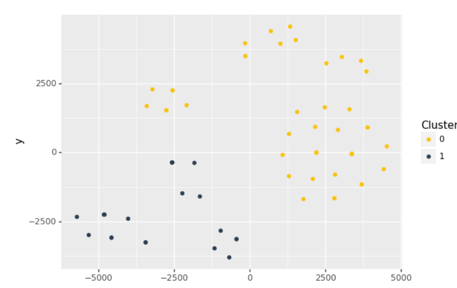
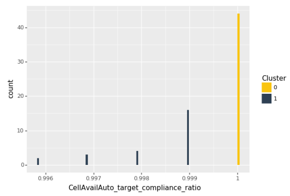


Figure 10: Clustering visualization for K-means in L1800 (Availability).

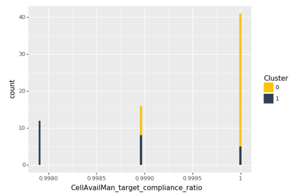
Through the visualization of the histograms for each feature, in Figure 11, the situation explained above is verified.

It can be observed that even though both clusters would present a score close to 1 for the features CellAvailAuto_target_compliance_ratio and CellAvailMan_target_compliance_ratio, the same would not be verified for cluster 0 in regard to CellAvail_target_compliance_ratio.

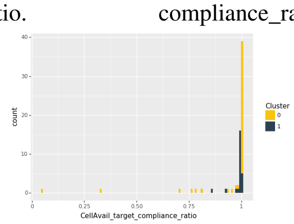
Thus, the overall score for the Availability should be computed taking into account only the features CellAvailAuto_target_compliance_ratio and CellAvailMan_target_compliance_ratio. In this case it is straightforward to see that both clusters



(a) CellAvailAuto_target_ compliance_ratio.



(b) CellAvailMan_target_ compliance_ratio.



(c) CellAvail_target_ compliance_ratio.

Figure 11: Histograms of Availability features for K-means in L1800.

would have a score near 1, even for a value of $time_{threshold}$ as high as 0.995.

Clustering Results Comparison

The results obtained with the three tested algorithms regarding the Accessibility and Integrity KPIs for each frequency band is summarized in Tables VI, VII and VIII.

KPI Class	Algorithm	Clusters	Number of cells	Score	Classification
Accessibility	K-means	2	79	0.32	Below average
			140	0.57	Average
	EM with GMM	2	15	0.2	Unsatisfactory
			204	0.5	Average
	Spectral	2	89	0.33	Below average
			130	0.58	Average
Integrity	K-means	2	89	0.11	Unsatisfactory
			130	0.67	Average
	EM with GMM	2	68	0.09	Unsatisfactory
			151	0.6	Average
	Spectral	2	90	0.11	Unsatisfactory
			129	0.68	Average

Table VI: Results comparison for L800.

KPI Class	Algorithm	Clusters	Number of cells	Score	Classification
Accessibility	K-means	3	4	0.7	Average
			26	0.8	Above average
			39	0.85	Above average
	EM with GMM	3	5	0.76	Above average
			31	0.86	Above average
			33	0.83	Above average
	Spectral	2	18	0.81	Above average
			51	0.85	Above average
Integrity	K-means	2	9	0.72	Average
			60	0.99	Above average
	EM with GMM	2	6	0.58	Average
			63	0.99	Above average
	Spectral	2	20	0.85	Above average
			49	1	Above average

Table VII: Results comparison for L1800.

KPI Class	Algorithm	Clusters	Number of cells	Score	Classification
Accessibility	K-means	3	6	0.57	Average
			35	0.79	Above average
			80	0.89	Above average
	EM with GMM	2	14	0.71	Average
			107	0.86	Above average
	Spectral	3	8	0.63	Average
			32	0.8	Above average
			81	0.88	Above average
Integrity	K-means	2	20	0.56	Average
			101	0.99	Above average
	EM with GMM	2	15	0.53	Average
			106	0.97	Above average
	Spectral	2	21	0.57	Average
			100	0.99	Above average

Table VIII: Results comparison for L2600.

Through the comparison of the results obtained for each clustering algorithm, it can be verified that, using the proposed mechanism to find the values of the input parameters which give the optimal partitioning, there were no significant differences between the clustering algorithms. As such, the K-means algorithm was considered the best out of the three tested algorithms, since it has less input parameters to tune, thus making it easier to use.

In addition, the proposed election mechanism with multiple CVIs to acquire the optimal configuration parameters, and therefore the optimal number of clusters, predominantly selects $k = 2$ for the optimal partitioning. This selection allows to capture the overall performance of the network, by identifying a cluster mostly composed of the best performing cells and a cluster mostly composed of the worst performing cells. However, it fails to find clusters of cells with more specific behaviours.

Lastly, it can be interesting to compare the results for the different frequency bands. The L800 frequency band shows an overall worse performance, for the Integrity and Accessibility classes, with the obtained clusters being classified as unsatisfactory, below average or average. On the other hand, the L1800 and L2600 frequency bands show a better performance, with all clusters having either average or above average performance.

VI. CM Results

This section presents the process used to correlate the performance of the clusters with the configuration parameters of the cells that constitute those clusters and, consequently, finding the optimal cell configuration.

The CM independence evaluation is performed after the clustering and PM analysis stages, as seen in Figure 1. Thus, the cells are labelled with the cluster to which they belong and those clusters have already been classified with respect to their performance for the KPI class being evaluated.

As briefly explained in Section IV, firstly, it is applied an independence test, for each configuration feature, to test if there are distinct configurations regarding that feature for cells belonging to different clusters. In this regard, Fisher's exact test was used.

Then, in case a correlation between a CM feature and the clusters is found, *i.e.* the values for that feature are dependent of the cluster, an engineer should evaluate that result to conclude if that configuration parameter as, in fact, any impact on the performance of class of KPIs being evaluated. If yes, then it should be straightforward to understand which values of the configuration feature are associated with the clusters that exhibit better performances.

A. Fisher's Exact Test

Fisher's exact test of independence is used when one has two nominal variables and wants to know, with a level of certainty defined through the significance level, whether the proportions of one variable change depending on the values of the other variable. The null hypothesis then corresponds to the relative proportions of one variable being independent of the value of the other variable. This test can be used in the problem of connecting the configuration parameters with the performance of each cluster because both the labels that identify the cluster and the configuration features can be considered nominal variables, since the optimal number of clusters k is forced to be small (between 2 and 8) and the observations of each one of the available CM features can be classified into a small number of categories.

Furthermore, the null hypothesis in this context is that, for each configuration feature, its relative proportions are independent of the labels (*i.e.* clusters). When this null hypothesis is rejected it means that different clusters have different proportions regarding the values of the configuration parameter being evaluated which, in turn, might indicate that there is a direct correlation between that configuration parameter and the performance of each cluster. In such case, that should be further investigated by an expert to identify if there is, in fact, a direct influence of the parameter value on the behaviour of the cells regarding the KPI class being evaluated and also what is the configuration that results in a better performance.

B. Use Case: Cell Subscription Capacity

The use case presented in this section is related to the clustering results obtained using K-means, for both the 800 MHz and 2600 MHz frequency bands, when evaluating the performance of the respective cells regarding the Accessibility KPI class. The significance level used for the Fisher's exact test was 0.05. If the p-value obtained for a CM feature when testing the null hypothesis is lower than the significance level, the null hypothesis is rejected, otherwise is accepted. If the null hypothesis is rejected than it can be inferred that there is a correlation between the performance level of a cluster and the configuration of the cells.

Let us consider the 800 MHz frequency band first. For this frequency band and accessibility features, two clusters were obtained. As presented in Table III, cluster 1 has an average performance with a score of 0.57, while cluster 0 has below average performance with a score of 0.32.

Fisher's exact test was then applied for each CM feature with respect to the obtained clusters. The results are presented in Table IX.

CM Feature	p-value	Independent?
CELLSUBSCRIPTIONCAPACITY	0.000004	No
LBTPNONQUALFRACTION	0.058	Yes
LBTPRANKTHRESHMIN	0.058	Yes
RXSINRTARGETCLPC	0.14	Yes
INTERFERENCETHRESHOLDSINRCLPC	0.31	Yes
NOCONSECUTIVESUBFRAMES	0.41	Yes
NOOFPUCCHCQIUSERS	0.54	Yes
NOOFPUCCHSRUSERS	0.54	Yes
PDCCHCFIMODE	0.75	Yes

Table IX: Fisher's exact test results for L800 (Accessibility).

From IX it can be verified that the only feature that is dependent on the cluster is the CELLSUBSCRIPTIONCAPACITY. Figure 12 shows the percentage of cells, for each one of the clusters, with respect to the configuration they have for the CELLSUBSCRIPTIONCAPACITY feature.

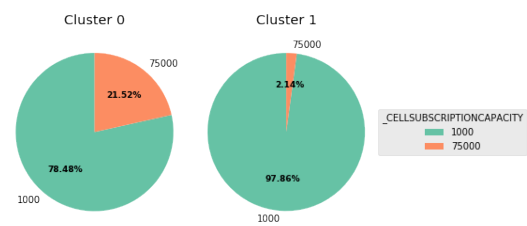


Figure 12: Proportions for CELLSUBSCRIPTIONCAPACITY per cluster in L800 (Accessibility).

It can be seen that cluster 0, that has a lower score, contains a higher percentage of cells with the CELLSUBSCRIPTION-CAPACITY feature set to 75000 when compared to cluster 1.

For the 2600 MHz frequency band and accessibility features, a similar situation is verified, where the clusters with worse accessibility performance have a higher proportion of cells with CELLSUBSCRIPTIONCAPACITY higher than the default value, which is 1000.

Taking these two cases into account, it seems that setting a higher value for CELLSUBSCRIPTIONCAPACITY results in a lower performance regarding the accessibility.

The CELLSUBSCRIPTIONCAPACITY is a feature that impacts the load balancing in a cell. Let $cellSubscriptionCapacity$ be the variable associated with the CELLSUBSCRIPTIONCAPACITY feature. The $cellSubscriptionCapacity$ is used to compute the $SubscriptionRatio$ as follows:

$$SubscriptionRatio = \frac{\sum qciSubscriptionQuanta}{cellSubscriptionCapacity} \quad (6)$$

where $qciSubscriptionQuanta$ is a weight given to an established E-RAB based on its CQI. Thus, the

$SubscriptionRatio$ can be viewed as the load in the cell. A higher value of $cellSubscriptionCapacity$ will cause the value of $SubscriptionRatio$ to drop, meaning that the cell will try to accommodate more users which may cause accessibility issues.

The CELLSUBSCRIPTIONCAPACITY feature is only taken into account when another CM feature, that controls the load balancing process, is active. Since there was no information available regarding the feature that activates the load balancing mechanism, it was assumed in this work that that feature was, in fact, activated.

VII. Conclusion

Since the dataset to which the clustering algorithms are applied depend on the set of targets defined, it is straightforward to understand that the definition of these targets is a key aspect of the methodology proposed. Thus, the targets for each KPI should be specified by the mobile network operator according to the desired level of performance for the network.

Moreover, it was also presented a possible qualitative classification for the clusters based on their score. Yet again, the score depends on a target, $time_{threshold}$, that should be set in agreement with the requirements of the network operator regarding the level of performance of the network. It was verified, for the 1800 MHz frequency band and Integrity KPIs, that a slight change in the value of $time_{threshold}$ results in a very distinct classification for one of the clusters obtained using K-means.

Regarding the clustering results using K-means, it was observed that the optimal number of clusters is given by the election mechanism with multiple CVIs is predominantly two, with one cluster mainly containing the best overall performing cells, for the KPI class and frequency band being evaluated, while the other is predominantly composed by the poorest performing cells.

Through the visualization of the attained clustering, using t-SNE, it is possible to infer about the separability of the data. For the Accessibility class of KPIs in L800 it was observed that it did not exist a clear separation between the two attained clusters.

Spectral Clustering and EM using GMM were also tested regarding the Accessibility and Integrity in the three frequency bands. It was verified that there were no significant differences in the results obtained with both EM with GMM and Spectral Clustering, when compared to the ones obtained with K-means. Therefore, given the simplicity in tuning the input parameters for the K-means algorithm, this was considered as the best out of the three.

Regarding the performance per frequency band, it was observed that the L800 exhibits a worse performance, for the Integrity and Accessibility classes, with the obtained clusters being classified as unsatisfactory, below average or average. The clusters obtained for both the L1800 and L2600 frequency bands are classified with either average or above average performance, thus it can be inferred that the cells operating in these frequency bands exhibit better overall performance when compared to the ones operating in L800.

The obtained results showed that the system is able to find different groups of cells regarding their performance and most importantly, is able to detect clusters of cells that show a performance level that is below the desired. However, it mostly captures the overall performance regarding the features being evaluated, having trouble to find clusters with more specific behaviours.

Regarding the configuration parameters, only a use case, related with the feature CELLSUBSCRIPTIONCAPACITY, was detected. For the L800 and L2600 frequency bands it was verified that the clusters with better performance, regarding the Accessibility features, are constituted by a higher percentage of cells with a lower configuration value for CELLSUBSCRIPTIONCAPACITY, while for the clusters with lower performance, the opposite happens.

References

[1] A. Gómez-Andrades, P. Muñoz, I. Serrano, and R. Barco, "Automatic root cause analysis for lte networks based on unsupervised techniques," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2369–2386, 2016.

[2] Ericsson, "Key Performance Indicators - User Guide," 2016.

[3] M. J. Zaki, W. Meira Jr, and W. Meira, *Data mining and analysis: fundamental concepts and algorithms*. Cambridge University Press, 2014.

[4] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[5] Shawe-Taylor, John and Cristianini, Nello, *Kernel Methods for Pattern Analysis*. Cambridge, UK: Cambridge University Press, 2004, pp. 40–42.

[6] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. IEEE, 2010, pp. 911–916.

[7] F. J. Massey Jr, "The kolmogorov-smirnov test for goodness of fit," *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[8] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[9] J. H. McDonald, *Handbook of Biological Statistics*. Sparky House Publishing, 2009, vol. 2.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[11] Google Inc., "Google Earth Pro," 2018. [Online]. Available: https://www.google.com/earth/download/gep/agree.html