Estimating Pole Capacity from Radio Network Performance Statistics by Supervised Learning

C. Gijón, M. Toril, S. Luna-Ramírez, J. L. Bejarano-Luque M- L- Marí-Altozano
Department of Communications Engineering. University of Málaga, 29071, Málaga, Spain.

***Abstract*—Network dimensioning is a critical task for cellular operators to avoid degraded user experience and unnecessary upgrades of network resources with changing mobile traffic patterns. For this purpose, smart network planning tools require accurate cell and user capacity estimates. In these tools, throughput is often used as a capacity metric due to its close relationship with user satisfaction. In this work, a comprehensive analysis is carried out to compare different well-known Supervised Learning (SL) algorithms for estimating cell and user throughput in the DownLink in busy hours from radio measurements collected on a cell basis in the Operation Support System (OSS). The considered SL approaches include random forest, shallow multi-layer perceptron, support vector regression and k-nearest neighbors. Such algorithms are compared with classical multiple linear regression and deep learning approaches considered in previous works. All these algorithms are tested in two radio access technologies: High Speed DownLink Packet Access (HSDPA) and Long Term Evolution (LTE). To this end, two datasets with the most relevant performance indicators per technology are collected from live cellular networks. Results show that non-deep SL algorithms are the most appropriate option for applications with storage constraints, such as network planning tools, since they provide a higher accuracy with reduced datasets.**

***Index Terms*—Cellular network, supervised learning, pole capacity, user throughput, network measurements.**

I. Introduction

Network (re)dimensioning is a key process in current cellular networks due to the constantly changing spatiotemporal distribution of mobile traffic [1] [2]. As a consequence, operators have to revise and constantly update their capacity plans to detect capacity bottlenecks in advance. Then, different re-planning actions can be executed in the short-term (e.g., a more efficient voice coding scheme [3], new handover margin settings for traffic sharing between adjacent cells [4], etc.) or in the long-term (e.g., bandwidth extension [5] or deployment of new carriers/co-sited cells) before capacity problems occur and user experience is degraded. Such changes can have a strong impact on cell and user capacity. Thus, it is essential to predict cell/user capacity in the new network conditions to guarantee that the expected traffic will be carried without any capacity bottleneck or user experience degradation. To reduce the workload of these tasks, the most advanced capacity planning tools are developed in a Self-Organizing Networks (SON) framework, where these checks are performed automatically [6] [7].

In current mobile networks, cell capacity is often measured as cell throughput in the DownLink (DL) in high load conditions. Since throughput depends on many additional factors (e.g., traffic mix, terminal capabilities, etc.), it is extremely difficult to isolate the impact of the above re-planning actions on that metric. Alternatively, the impact of these actions on lower-level radio network performance indicators (e.g., bandwidth, channel quality indicator distribution, power, etc.) can more easily be predicted, and then such predictions can be used as inputs to a capacity estimation model.

In the literature, several works have proposed capacity estimation models from network performance and configuration data collected in the Operation Support System (OSS), so that the peculiarities of each cell can be taken into account [8] [9] [10]. Nonetheless, the constant change in network capabilities requires updating legacy models. With the increase of service diversity and the raise of user expectations, operators have been forced to change their network management procedures from a network-centric approach based on network performance to a user-centric approach focused on user satisfaction (Quality of Experience, QoE). Such a trend will continue in 5G systems, where services of very different nature (e.g., enhanced Mobile Broadband, eMBB, or Ultra Reliable Low Latency Communications, uRLCC) will coexist [2]. To this end, network performance indicators (e.g., cell capacity) have to be complemented by other indicators that better reflect the end user experience (e.g., user capacity). In this context, user throughput in the DL (hereafter, DL user throughput) is often regarded as a significant QoE metric for eMBB services [11] [12], and can therefore be considered in network management procedures such as network dimensioning or network slicing provisioning. It is expected that the higher DL user throughput experienced by users in a cell/slice in congestion scenarios, the higher QoE level can be guaranteed for eMBB services in such a cell/slice.

In this work, a comprehensive analysis is carried out to compare the performance of different supervised learning (SL) algorithms for cell and user throughput estimation in the DL in busy hours from network measurements collected in the OSS in two radio access technologies, namely HSDPA and LTE. For this purpose, two datasets with the most relevant performance indicators per technology are collected from a live HSDPA and a live LTE network, respectively.

The rest of the document is organized as follows. Section II presents related work and highlights the main contributions of this work. Section III describes the two considered datasets (one per technology). Section IV explains the method used to estimate capacity indicators. Section V presents the results. Finally, Section VI summarizes the main conclusions.

II. Related work and contributions

Cell capacity estimation has been extensively covered in the literature. Different metrics have been considered as capacity indicators. In [13], the authors present an admission control policy driven by an analytical model based on a multidimensional continuous-time Markov chain to estimate the varying capacity of cells in LTE caused by user mobility in terms of session blocking probability. In [14], the available bandwidth (i.e., channel spare capacity) is estimated from measurements taken in drive tests in MONROE 3G/4G testbed, and the relationship between available bandwidth and achievable throughput is analyzed. The MONROE platform is also used in [15] to characterize cell capacity offered by 11 operators in 4 different countries, measured as maximum throughput at the application layer. In [16], cell capacity for Voice over LTE (VoLTE) service is measured as the maximum number of simultaneous active users (i.e., users with data to transmit) that can be served by a cell. Then, an analytical model is proposed to estimate cell capacity in areas of the cell where users report different channel quality information. In [17], a model based on linear regression is proposed to measure the maximum allowed traffic in Erlangs in a multi-service HSDPA network for different transmit powers and quality of service requirements from network performance indicators.

A common approach is to measure cell capacity as DL cell throughput in high load conditions. Several analytical models have been developed to estimate cell throughput considering different Multiple-Input Multiple-Output (MIMO) antenna schemes [18], scheduling algorithms [19] and traffic classes [20]. However, the capacity of a live cell is highly dependent of multiple factors, such as service mix, terminal capabilities or propagation environment (e.g., indoor, outdoor...), which change with time and location. To deal with this diversity, in some studies cell throughput is estimated by means of simulations in GPRS [21] or HSDPA [22]. Nonetheless, it is virtually impossible to simulate all possible combinations of the above-mentioned factors. Alternatively, some works estimate cell throughput from network performance counters and configuration data collected in the OSS. In [8], the authors propose a model based on multiple linear regression to estimate DL cell throughput in the busy hour in HSDPA from code-related, quality-related and power-related indicators collected on a cell basis. In [9], a similar methodology is used to estimate the same metric in a multi-service live LTE network. In [10], delay in connection setup is also considered as an input to the linear model.

Regarding DL user throughput estimation, in [23], SL algorithms are applied over data collected in a crowdsourced speed test. Such tests collect terminal- and network-related data through a large number of over-the-air transmissions. As a consequence, they can overload the radio interface and drain user limited data plans. Alternatively, in [24], an analytical model is proposed to estimate DL user throughput using drive test data collected by a radio frequency scanner. However, drive tests are time-consuming and imply high operational costs, since they must be performed periodically to adapt to events in the area (e.g., new buildings) or in the network (e.g., new cells) affecting radio frequency measurements [25].

An efficient approach for operators is to estimate cell and user throughput from the same set of network measurements gathered in the OSS during normal network operation [26]. Unfortunately, these measurements are often aggregated per cell, and thus do not reflect the performance of individual connections. Moreover, unlike cell capacity metrics, user capacity may not be linearly related to cell-level indicators, causing that models based on linear regression do not perform well. To tackle this non-linearity issue, SL algorithms can be used, since they are able to capture non-linear dependencies among variables [27]. Closer to the work presented here, in [26], DL cell/user throughput in LTE is estimated with a Deep Neural Network (DNN) from a labeled dataset. The authors consider a set of 13 configuration parameters and performance indicators, collected in a live network hourly during 2 months to train their model. However, most network operators are not familiar with machine learning, and are thus reluctant to use complex models (e.g., deep learning models) with hundreds of hyperparameters and internal parameters, which are difficult to configure and interpret. Moreover, deep learning algorithms require very large training datasets (tens of thousands of samples) to perform well; otherwise, they are prone to overfitting. This is an issue for network operators, since collecting such an amount of data in the OSS implies deploying large databases and makes data pre-processing extremely time-consuming [28]. For these reasons, operators prefer simpler SL algorithms for their network planning tools.

In this work, these shortcomings associated to method complexity are addressed by solving DL cell/user throughput estimation via non-deep SL algorithms, which, as shown later, work accurately with reduced datasets. Moreover, feature selection is applied over the initial set of candidate features to select a minimal subset of metrics to be stored in the OSS for capacity estimation purposes. Likewise, we extend our study to consider not only LTE, but also HSDPA radio access technology. All SL techniques considered here are included in most data analytics packages and have already been used in several fields. Hence, the main novelty here is the assessment of well-established SL methods for cell/user throughput estimations in busy hours from measurements in different radio access networks. Specifically, the main contributions of this work are:

- Presenting the first comparison of non-deep SL schemes performance for DL cell/user throughput estimation in busy hours from network measurements in LTE. The considered approaches include random forest, multi-layer perceptron, support vector regression and k-nearest neighbors. These algorithms are compared with deep learning and linear regression techniques proposed in previous works [9] [26] [10].
- Extending the analysis to HSDPA, where the use of SL for cell/user throughput estimation from network measurements has not been covered yet. This is the first attempt to estimate user throughput from cell-level indicators in HSDPA, as previous works only covered cell throughput estimation with multiple linear regression [8].
- Identifying a minimal set of key network performance indicators to be stored in the OSS to estimate throughput

indicators in both technologies.

III. DATASETS

In this work, cell and user throughput estimations are evaluated in two different radio technologies. For this purpose, two datasets are collected from a live HSDPA and a live LTE network, respectively. This section describes such networks and how datasets are built.

A. Dataset A – HSDPA

Data is collected in a live 3G network serving an entire country (approximately 10,000 km^2), comprising 12,318 cells of very different sizes and environments. Two carrier frequencies are deployed per cell. A first carrier is used for Adaptive Multi-Rate Circuit-Switched (AMR CS) calls and non-HSDPA packet-switched traffic, while a second carrier is used for HSDPA traffic and AMR CS calls when the first carrier is full. The analysis is focused on the second carrier (i.e., HSDPA capacity).

Performance measurements are collected on a cell and hourly basis for a complete day in the OSS. The maximum cell/user capacity is defined as the average cell/user throughput in the DL when the Transmission Time Interval (TTI) utilization ratio, $\%TTI_{utilDL}$, is near the maximum threshold commonly set by the operator to avoid congestion problems (typically, $\%TTI_{utilDL,th}$=70%) [29]. Thus, only data from highly loaded cells is considered to obtain reliable estimates of the actual cell/user capacity. The selection of such cells is implemented through the observation of the cell busy hour, defined as the hour with the largest average number of active users (i.e., with data to transmit) over HSDPA. Analysis is restricted to those cells with $\%TTI_{utilDL} > 50\%$ during such busy hour. This filter results in a dataset comprising 1,095 samples with the following features:

1) Cell identifier.
2) Date (format DD/MM/YYYY HH:HH).
3) $\%TTI_{utilDL}$ in HSDPA, as a measure of cell load.
4) A set of 12 network performance indicators, shown in Table I, as candidate input features for capacity estimation. To allow the comparison with previous approaches, the considered input features are those already selected in [8], including code-related indicators (e.g., no. of codes used in HSDPA), traffic-related indicators (e.g., no. of active users), power-related indicators (e.g., avg. DL transmit power for HSDPA) quality-related indicators (e.g., median CQI). Table II presents the minimum, maximum and average value and the standard deviation of such parameters in the dataset.
5) The average cell throughput in the HSDPA DL, $ThCell_{HSDPA}$, defined as the total Packet Data Convergence Protocol (PDCP) data volume transmitted per second in active periods in the HSDPA DL of a cell, as a measure of cell capacity [30].
6) The average user throughput in the HSDPA DL, $ThUser_{HSDPA}$, defined as the average PDCP data volume transmitted per second in the DL to an active HSDPA user, excluding TTIs emptying the transmit buffer, as a measure of user capacity [30] [31].

TABLE I: Candidate features in HSDPA.

	Name	Description
Power	$Avg\_R99\_DL\_power$ [mW]	Avg. DL transmit power for Data Channel (DCH)
	$Avg\_HSDPA\_DL\_power$ [mW]	Avg. DL transmit power for HSDPA
Traffic	$Avg\_activeUE\_DL$	Avg. no. of HSDPA active users per TTI in the DL
Code	$Avg\_codes\_used\_HSDPA$	Avg. no. of codes used in HSDPA
	$Avg\_SF16\_codes\_HSDPA$	Avg. no. of codes with spreading factor 16 reserved for HSDPA
	$Avg\_codes\_HSDPA\_UE$	Avg. no. of codes used per HSDPA user
	$Code\_load$ [%]	Percentage of channelization codes used in both DCH and HSDPA
Quality	CQI_class_p50	Median Channel Quality Indicator (CQI) value
	CQI_class_p80	80th-tile of CQI distribution
	$16QAM\_usage$ [%]	Usage of 16QAM modulation (as opposed to QPSK)
	$RLC\_retx\_ratio\_DL$	Ratio of Radio Link Control (RLC) retransmissions in the DL
	$PDU656\_usage$ [%]	Percentage of Packet Data Units with size 656 B (as opposed to 310 B)

TABLE II: Statistics of dataset A (HSDPA scenario).

Indicator	Min.	Max.	Mean	Std. deviation
$Avg\_R99\_DL\_power$ [mW]	2014	15777	7512	1844
$Avg\_HSDPA\_DL\_power$ [mW]	1200	16060	5684	1368
$Avg\_activeUE\_DL$	0.85	62.68	20.43	10.85
$Avg\_codes\_used\_HSDPA$	1.10	9.20	4.30	1.10
$Avg\_SF16\_codes\_HSDPA$	5	14	9.48	1.93
$Avg\_codes\_HSDPA\_UE$	7.15	11.40	9.29	0.67
$Code\_load$ [%]	47.62	96.66	88.16	3.59
CQI_class_p50	6	22	15.06	1.90
CQI_class_p80	9	26	19.47	1.86
$16QAM\_usage$ [%]	0.10	89.20	22.09	14.41
$RLC\_retx\_ratio\_DL$	0.02	1.34	0.14	0.07
$PDU656\_usage$ [%]	0	135.80	32.82	26.02

B. Dataset B – LTE

The second dataset is collected in a live LTE network comprising 656 cells covering urban and residential areas. In this network, two carriers are deployed at 700 MHz and 2100 MHz with a system bandwidth of 10 MHz and 5 MHz, respectively. In this case, the analysis includes both carriers. To obtain the dataset, configuration and performance data is gathered on an hourly and cell basis for 6 days, resulting in $24 \cdot 6 \cdot 656 = 94,464$ samples (note that the lower size of the network allowed a longer data collection period compared to the HSDPA case). Again, to obtain reliable estimates, the analysis is restricted to those cells/hours where $\%TTI_{utilDL} > 50\%$. This filter results in a dataset with 2,141 samples including the following information:

1) Cell identifier.
2) Date (format DD/MM/YYYY HH:HH).
3) $\%TTI_{utilDL}$, as a measure of cell load.
4) A set of 10 network indicators, shown in Table III, as

TABLE III: Candidate features in LTE.

	Name	Description
Quality	Avg_CQI	Avg. DL CQI
	σ_{CQI}	Standard deviation of CQI distribution
	CQI_class_p5	$5th$-tile of CQI distribution
	CQI_class_p10	$10th$-tile of CQI distribution
	$HARQ\_fail\_ratio\_DL$	Hybrid Automatic Repeat request failure ratio in the DL
	$RLC\_retx\_ratio\_DL$	Ratio of RLC retransmissions in the DL
	DL_assign_Ack	Ratio of correct resource assignments in the DL control channel
Traffic	$Avg\_activeUE\_DL$	Avg. no. of active users per TTI in the DL
CMs	BW [MHz]	LTE system bandwidth
	$PUCCH\_SR\_users$	Max. no. of users allowed to send Scheduling Request in UL

TABLE IV: Statistics of dataset B (LTE scenario).

Indicator	Min.	Max.	Mean	Std. deviation
Avg_CQI	5.50	12.21	7.81	0.95
σ_{CQI}	0.29	3.13	0.85	0.19
CQI_class_p5	1.31	7.12	3.48	0.68
CQI_class_p10	1.69	8.29	4.17	0.74
$HARQ\_fail\_ratio\_DL$	0.05	0.11	0.07	$7.3{\cdot}10^{-3}$
$RLC\_retx\_ratio\_DL$	$1.2{\cdot}10^{-5}$	0.05	$7.2{\cdot}10^{-4}$	$1.6{\cdot}10^{-3}$
DL_assign_Ack	0.26	0.99	0.96	0.07
$Avg\_activeUE\_DL$	0.30	16.97	1.69	1.06
BW [MHz]	5	10	9.44	1.57
$PUCCH\_SR\_users$	560	730	646.35	34.06

candidate input features. These include network settings (e.g., system bandwidth), quality-related statistics (e.g., average CQI) and traffic-related statistics (e.g., no. of active users) provided by most vendors and used in previous studies for capacity estimation in LTE [9]. Table IV presents the minimum, maximum and average value and the standard deviation of such parameters in the dataset.

5) The average cell throughput in the DL, $ThCell_{LTE}$, as a measure of cell capacity.
6) The average user throughput in the DL, $ThUser_{LTE}$, as a measure of user capacity.

From the comparison of Tables I and III, it is observed that some of the considered indicators provide similar information in both technologies (e.g., no. of active users, RLC retransmissions, CQI, etc.), and, thus, both analysis rely on similar initial information. Nonetheless, other indicators are distinctive of the technology (e.g., code-related indicators in HSDPA), so that specific technology information is also taken into account. For a deeper analysis, Fig. 1 shows the Cumulative Distribution Function (CDF) of $Avg\_activeUE\_DL$ and CQI_class_p50 indicators in dataset A (solid lines) and $Avg\_activeUE\_DL$ and Avg_CQI indicators in dataset B (dashed lines). Note that, although only high-load cells are considered in both technologies, $Avg\_activeUE\_DL$ in LTE is lower than in HSDPA, revealing that users in LTE demand more data-hungry services. Likewise, the highest value of CQI

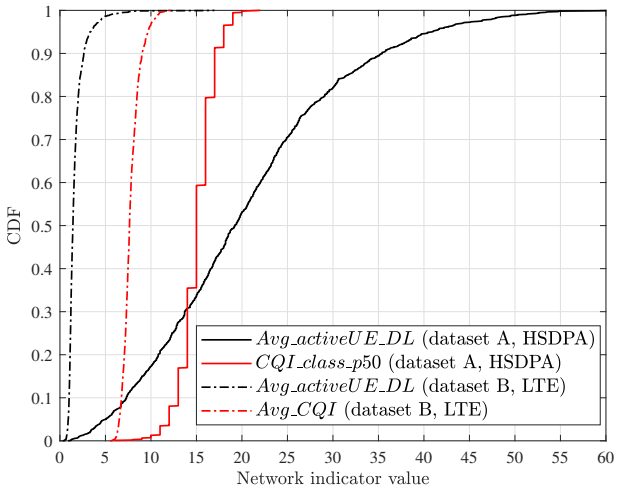


Fig. 1: Cumulative distribution function of network indicators.

measured per cell (i.e., 22 in HSDPA scenario and 12 in LTE scenario) are below the maximum CQI value defined in each technology (i.e., 30 in HSDPA and 15 in LTE [32]).

The two datasets used in this work (i.e., datasets A and B) combine a large geographical area (hundreds of cells) with an adequate time resolution (hour), similarly to those used by operators for capacity estimation purposes. This fact guarantees the reliability and significance of results. Note that, because of the busy hour and TTI utilization ratio filtering, these datasets have a reduced number of samples. This property increases the interest of assessing the performance of non-deep SL algorithms for cell/user capacity estimation, since these algorithms are less prone to overfitting than deep learning approaches when trained with reduced datasets.

IV. Capacity Estimation Method

Fig. 2 illustrates the procedure followed to carry out the data analysis. Such a process is identical for HSDPA and LTE technologies. First, network parameters and measurements are collected from the live network to build the dataset as described in Section III. Then, data is pre-processed to normalize the values of each variable and create training and test datasets. Next, performance models are created separately to estimate each of the considered output features (i.e., $ThCell_{HSDPA/LTE}$ and $ThUser_{HSDPA/LTE}$). For this purpose, hyperparameters of the different SL algorithms must first be configured. Then, for each algorithm, three different models are trained. A first model, referred as to full model (FULL), considers all the collected indicators as input features, whereas the remaining two models, referred to as Feature Selection (FS) models, select a representative subset of indicators as input features based on two different criteria. Finally, model performance is assessed on the corresponding test dataset. A more detailed explanation of each step is given next.

A. Data pre-processing

Different input features show very different ranges (e.g., Avg_CQI in Table IV varies from 5.50 up to 12.21, while

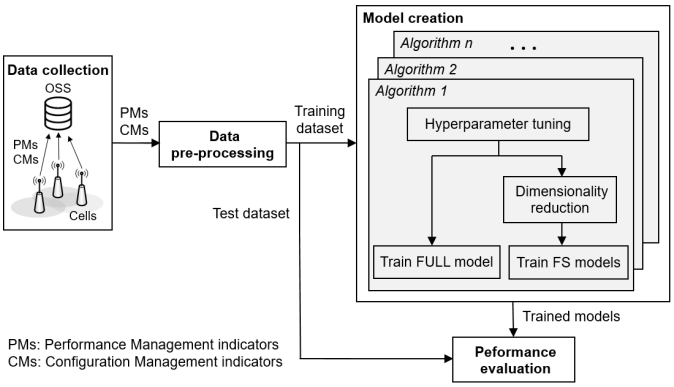


Fig. 2: Flow diagram for cell/user capacity estimation.

$HARQ\_fail\_ratio\_DL$ varies between 0.05 and 0.11). To guarantee a good accuracy and faster convergence of SL algorithms, it is convenient to normalize data, so that all variables are comparable. In this work, a Z-score normalization method is used [33]. The normalized value of a certain indicator, denoted as i_{norm}, is computed as

$$i_{\text{norm}} = \frac{i - \mu}{\sigma}, \tag{1}$$

where i is the original value of the indicator and μ and σ are the average and standard deviation, respectively, considering all values available in the dataset for indicator i.

After data normalization, the N_s samples in the dataset are split into training and test subsets by creating a random partition, so that the training set comprises 80% of samples and the test set includes the remaining 20%.

B. Model creation

1) Overview of estimation algorithms: Since DL cell/user throughput can be computed offline from information collected in the OSS, creating a labeled dataset is not an issue for network operators. For this reason, this work considers SL approach. Six well-known SL regression algorithms are compared: Multiple Linear Regression (MLR), Support Vector Regression (SVR), Random Forest (RF), k-Nearest Neighbors (KNN) and shallow/deep artificial Neural Networks (ANN). A high-level description of these techniques is provided next [27]:

- MLR captures the relationship between a response variable and multiple explanatory variables by fitting a line through a multi-dimensional space of samples. The optimal solution is reached by linear least squares fitting, which minimizes the sum of squares of residuals between estimates and ground-truth data. This approach is considered in [8] [9] [10] for cell capacity estimation, and here it is used as a benchmark.
- SVR maps a set of inputs into a higher dimensional feature space to find the regression hyperplane that best fits every sample in the dataset. For this purpose, a linear or non-linear (a.k.a., *kernel*) mapping function is used. Unlike MLR, SVR neglects all deviations below an error sensitivity parameter, ϵ. Moreover, the regularization parameter, C, restricts the absolute value of regression coefficients. Both parameters control the trade-off between regression accuracy and model complexity (i.e., the smaller ϵ and larger C, the better the model fits the training data, but overfitting is more likely).
- RF is an ensemble learning method where several decision trees are created by considering different subsets of the training data (a.k.a. bootstrap aggregating or bagging). To avoid model overfitting, trees are pruned. To reduce the correlation among trees, a different random subset of input attributes is selected at each candidate split in the learning process (feature bagging) [34]. Then, estimations obtained from different trees are averaged to perform a robust regression.
- KNN is a distance-based algorithm that relies on the fact that observations with similar characteristics tend to have similar outcomes. To estimate the response variable of a new observation, KNN identifies the k nearest neighbors of the sample in the training dataset according to some previously defined distance metric (e.g., Euclidean distance) and then computes the outcome based on the averaging (sometimes weighted) of such neighbors.
- ANN is a statistical learning method inspired by the structure of a human brain. In ANN, entities called nodes act as neurons, performing non-linear computations by means of activation functions [35]. Two feed-forward ANNs (i.e., without memory) are considered in this work [36]. Both are Multi-Layer Perceptrons (MLPs), differing in the number of hidden layers. The first one, denoted as MLP–SNN, has a single hidden layer (i.e., is a shallow neural network). The second one, denoted as MLP–DNN, is a deep neural network based on that tested in [26] to estimate cell and user throughput in LTE. The architectures of such networks (number of layers, number of neurons per layer, activation functions, etc.) are detailed later.

2) Hyperparameter tuning: Hyperparameters are internal model parameters controlling the learning process in machine learning algorithms. SL algorithms often have dozens (or even hundreds) of hyperparameters, some of which have a strong impact on model performance. The best setting strongly depends on the problem. Hence, in this work, hyperparameters are tuned separately for each output feature (i.e., $ThCell_{HSDPA}$, $ThUser_{HSDPA}$, $ThCell_{LTE}$ and $ThUser_{LTE}$).

For simplicity, the less influential parameters for each algorithm are fixed, and only the most influential parameters are tuned. Based on previous works [37] and on our own experience, the following parameters are tuned: a) the kernel function, ϵ and C in SVR, b) the number of trees, the distance metric and the pruning criterion in RF, c) the distance metric and the number of neighbors in KNN and d) the activation function, the number of neurons, the optimization algorithm, the number of iterations and the number of layers in MLP–SNN and MLP–DNN (the latter parameter is only adjusted in MLP–DNN). Such parameters are fixed by a grid

search in the parameter space [38]. For each algorithm, the best hyperparameter value (or tuple) is that minimizing the Mean Absolute Percentage Error ($MAPE$) in the training dataset, computed as

$$MAPE = \frac{1}{N_s} \sum_{s=1}^{N_s} \left| 100 \cdot \frac{\widehat{y}(s) - y(s)}{y(s)} \right| , \quad (2)$$

where N_s is the number of samples in the dataset, and $y(s)$ and $\widehat{y}(s)$ are the measured and estimated values of the output feature in sample s, respectively.

3) Feature selection: For most algorithms, it is expected that the larger the number of input features (i.e., network indicators), the better estimation of the output feature (i.e., cell/user throughput). However, when it comes to SON tools for cellular networks, additional factors must be considered. Note that cell/user capacity estimation is performed by a centralized network planning tool running in the OSS. Collecting measurements in the OSS poses a storage problem for operators, since large databases must be deployed to gather a continuous stream of information coming from thousands of cells along the network [28]. Likewise, sending information from base stations to the OSS might cause congestion problems in the backhaul and node processor. Moreover, data pre-processing becomes extremely time-consuming when the number of indicators grows, requiring expensive processing platforms with a large computational power. As a consequence, operators are reluctant to collect more than 5 or 6 indicators in their network planning tools. Hence, dimensionality reduction a key aspect in this work.

Dimensionality reduction can be tackled through Feature Selection (FS) or Feature Extraction (FE) [39]. FS consists on identifying the subset of features that are more relevant according to a certain criterion. In contrast, in FE, a new (and reduced) set of features is built by combining the features from the original set. As explained above, the main motivation to perform dimensionality reduction in this work is to reduce the number of network key indicators collected in the OSS. Thus, FS is performed, since it allows to disable the monitoring of non-relevant indicators (which may not be possible with FE).

FS comprises filtering, wrapper and ensemble methods. In *filtering* methods, features are selected according to their correlation with the outcome variable. These methods are very efficient and can be used as a pre-processing step, since they are independent of the SL algorithm. However, they might fail to find the optimal subset of features. *Wrapper* methods select subsets of variables according to their usefulness for a given SL algorithm. Although they are computationally expensive, and make the model more prone to overfitting, they provide the best subset of features. Finally, *ensemble* methods implement a combination of filtering and wrapper methods [40].

In this work, two different FS methods are considered. A first method, denoted as FS-COR, is a simple filtering method that considers as relevant those features whose linear correlation, ρ, with the response variable is high, i.e., $|\rho| > 0.5$. A second method, denoted as FS–SFS (Sequential Forward Selection), is a wrapper method. It starts with an empty model, where the most relevant features are sequentially added until adding an additional feature does not lead to a significant improvement in a predefined loss function. In this work, the $MAPE$ is selected as loss function, and the stop condition is when the decrease in $MAPE$ after adding a new feature is lower than 1% (provided that a target $MAPE$ threshold is fulfilled). To prevent overfitting in FS–SFS, a 5-fold cross validation is performed over the dataset [27].

Note that the set of candidate features differs in HSDPA and LTE. Moreover, for a given technology, some features can be relevant for estimating cell capacity, but negligible for user capacity (or vice versa). Thus, FS-COR must be performed per technology and output feature, and FS–SFS must be performed per technology, output feature and SL algorithm.

C. Performance evaluation

The main figure of merit used to assess models is the $MAPE$, defined in (2). Additionally, the execution time is also considered as a measure of computational load.

V. Performance assessment

In this section, the performance of the different capacity estimation algorithms is compared over the two datasets presented in Section III. The assessment methodology is described first and results are presented later. For clarity, results are broken down per radio technology.

A. Analysis set-up

The six regression algorithms described in Section IV are implemented with *scikit-learn* and *Keras*, two machine learning libraries for Python extensively used in several fields [41] [42]. Table V describes the hyperparameter settings when estimating cell and user throughput in HSDPA and LTE. Only the parameters configured through the grid search process are included in the tables, together with the parameter space tested. The reader is referred to [41] [42] for further information about the (fixed) configuration of the other hyperparameters.

As explained in Section IV, three models are derived with each regression algorithm: a full model with all predictors (FULL), a simplified model with predictors selected by a filtering method (FS-COR) and a simplified model with predictors selected by a wrapper method (FS–SFS). Thus, 18 regression models are tested.

A model (i.e., combination of SL regression algorithm and FS scheme) is considered acceptable to estimate cell/user throughput if $MAPE$<10%. This value has been considered as an acceptable error in previous works [10], providing a trade-off between model complexity and required accuracy. A too restrictive $MAPE$ threshold (e.g., 5%) can only be achieved by complex models, which require large datasets to be trained. In network planning tools, such an increase in complexity does not pay off, since operators have to take the same re-planning actions whether capacity problems are detected with a $MAPE$ of 5% or 10%. In contrast, a too relaxed threshold (e.g., 15%) can lead to significant errors in capacity estimations. Note that underestimating cell/user

TABLE V: Hyperparameter settings.

	Hyperparameter name	Value for $ThCell_{HSDPA}$	Value for $ThUser_{HSDPA}$	Value for $ThCell_{LTE}$	Value for $ThUser_{LTE}$	Parameter Space
SVR	Sensitivity, ϵ	0.1	0.1	0.1	0.1	[0.05,1]
	Regularization, C	12	40	80	100	[1,150]
	Kernel function	Linear	Radial Basis Function (RBF)	RBF	RBF	{Lineal, sigmoid, polynomial, RBF}
RF	No. trees	30	40	30	40	[5,50]
	Maximum depth	20	20	20	20	[5,50]
	Distance metric	MAE	MAE	MAE	MAE	{MAE, MSE}
KNN	No. neighbors	5	5	5	5	[4,20]
	Distance metric	Euclidean	Euclidean	Euclidean	Euclidean	{Euclidean, Manhattan, Chebyshev}
MLP-SNN	No. layers	3	3	3	3	Fixed
	No. neurons in hidden layer	5	6	13	19	[2,100]
	Activation function inn hidden layer	Rectified Linear Unit (ReLU)	ReLU	Hyperbolic tangent	ReLU	{Identity, logistic, ReLU, hyperbolic tangent}
	Optimization algorithm	LBFGS	LBFGS	LBFGS	LBFGS	{SGD, Adam, LBFGS}
	No. iterations	1000	1000	1000	1000	[100,10000]
MLP-DNN	No. layers	5	5	6	6	[4,10]
	No. neurons in hidden layers	N_{feat}	N_{feat}	N_{feat}	N_{feat}	Fixed, based on [26]
	Activation function inn hidden layers	ReLU	ReLU	Hyperbolic tangent	ReLU	{Identity, logistic, ReLU, hyperbolic tangent}
	Optimization algorithm	Adam	Adam	Adam	Adam	{SGD, Adam, FTRL, RMSprop}
	No. iterations	1000	1000	750	750	[100,10000]

capacity can lead to unnecessary investments (e.g., bandwidth extension licenses), whereas overestimating such metrics can cause capacity bottlenecks (e.g., underprovision of radio resources), leading to user experience degradation. The best model for each output feature is selected as a trade-off between an acceptable $MAPE$ (i.e., high accuracy) and a reduced number of input features, N_{feat} (i.e., low storage capacity requirements in the OSS).

B. Results – HSDPA

Table VI breaks down the results obtained for the considered regression algorithms when estimating $ThCell_{HSDPA}$ and $ThUser_{HSDPA}$ with the FULL, FS-COR and FS–SFS models. $ThCell_{HSDPA}$ results are analyzed first and $ThUser_{HSDPA}$ is considered later.

Results from FULL models show that, as stated in [8], classical MLR achieves a high accuracy (i.e., $MAPE$=7.37%) when estimating $ThCell_{HSDPA}$. This result gives clear evidence that some input features have a strong linear relationship with the output variable. RF, MLP–SNN and MLP–DNN improve MLR accuracy, with a $MAPE$ of 6.69%, 5.60% and 5.81%, respectively. KNN shows the worst results, although its $MAPE$ (=10.03%) is still acceptable. An analysis of the Pearson correlation coefficients (not shown here) reveals that CQI_class_p50, CQI_class_p80, $Avg\_codes\_used\_HSDPA$ and $16QAM\_usage$ features are linearly correlated with $ThCell_{HSDPA}$ (i.e., $|\rho| > 0.5$). This is reinforced by the fact that FS-COR models, which use only those 4 indicators as input features, achieve a $MAPE$ below 10% for all algorithms. Fig. 3 shows the evolution of the $MAPE$ obtained across FS–SFS process. As expected, in general, the larger number of features, the higher accuracy. However, KNN performance degrades progressively when the number of features grows above N_{feat}=5. This unexpected behavior is due to the fact that KNN is an algorithm based on distance. As the number of input features grows, distances among data points become all approximately equal, which can degrade model performance. MLP–SNN and RF perform similarly, providing the best results with a low number of input features ($N_{feat} \geq 3$). Table VI includes, in FS–SFS column, the $MAPE$ value obtained for each algorithm with the N_{feat} value selected with the pre-defined convergence criterion. $MAPE$ values show that FS–SFS models reduce the required storage capacity compared to the FULL models at the expense of a negligible degradation in $MAPE$ ($\lessapprox$ 1% in absolute terms for all algorithms). In KNN, FS–SFS model is more accurate than the FULL model (i.e., $MAPE$ = 10.03% with FULL model, and 8.27% for FS–SFS model) for the above explained reasons. Overall, the best model is MLP–SNN with FS–SFS, since it achieves a $MAPE$ very close to the best model (6.80%) with only 4 input features (CQI_class_p50, $Avg\_SF16\_codes\_HSDPA$, $Avg\_codes\_used\_HSDPA$ and $16QAM\_usage$). Nonetheless, Fig. 3 shows that an acceptable $MAPE$ (i.e., < 10%) can be achieved with all non-deep SL algorithms by selecting a subset of only 2 features (specifically, $Avg\_codes\_used\_HSDPA$ and $16QAM\_usage$). Hence, it can be concluded that MLR is competitive with more sophisticated SL algorithms when estimating busy-hour cell throughput in HSDPA.

It is remarkable that the subset of features in the best option

TABLE VI: $MAPE$ performance in HSDPA.

	$ThCell_{HSDPA}$			$ThUser_{HSDPA}$		
Model	FULL	FS–COR	FS–SFS	FULL	FS–COR	FS–SFS
N_{feat}	12	4	—	12	4	—
MLR	7.37	9.39	8.13 (N_{feat} = 4)	43.11	49.15	41.24 (N_{feat} = 4)
SVR	7.36	9.34	8.17 (N_{feat} = 4)	13.31	21.75	11.03 (N_{feat} = 4)
RF	6.69	9.06	7.04 (N_{feat} = 4)	13.42	22.44	12.11 (N_{feat} = 4)
KNN	10.03	8.46	8.27 (N_{feat} = 3)	17.09	22.52	12.33 (N_{feat} = 4)
MLP–SNN	5.60	9.33	6.80 (N_{feat} = 4)	10.64	21.68	9.23 (N_{feat} = 5)
MLP–DNN	5.81	8.99	7.13 (N_{feat} = 4)	8.79	20.95	10.26 (N_{feat} = 10)

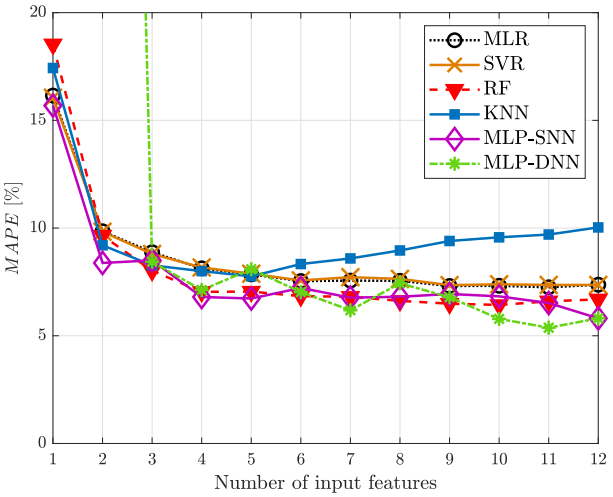


Fig. 3: $MAPE$ evolution – $ThCell_{HSDPA}$ estimation.

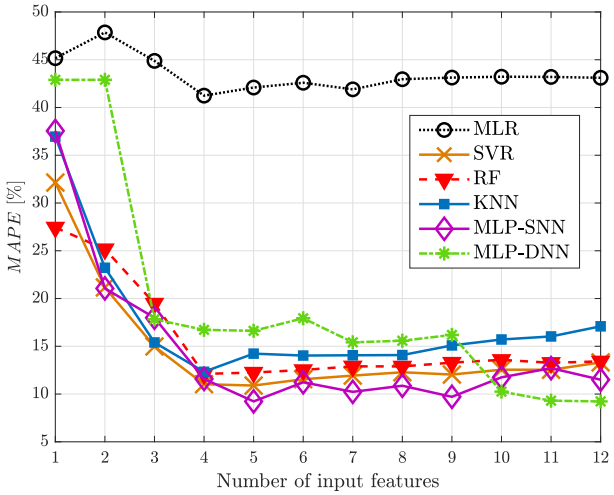


Fig. 4: $MAPE$ evolution – $ThUser_{HSDPA}$ estimation.

(MLP–SNN with N_{feat}=4) differs in number of features and in some of the selected features from the subset in [8], where $ThCell_{HSDPA}$ is estimated via MLR and feature selection is performed based on p-values. In that work, the authors propose a model with 5 input features: CQI_class_p50, $Avg\_codes\_used\_HSDPA$, $Avg\_HSDPA\_DL\_power$, $16QAM\_usage$ and $PDU656\_usage$. Thus, it can be concluded that, when estimating cell capacity, not only how many features, but also which features must be stored in the OSS, depend on the model and feature selection approach selected.

When it comes to user throughput estimation, MLR does not perform well. Specifically, a $MAPE$ of 43.41% is achieved by MLR with the FULL model. This poor performance suggests that there is a non-linear relationship between the input features and the output variable, $ThUser_{HSDPA}$ (e.g., $ThUser_{HSDPA}$ is not linearly dependent on the number of simultaneous active users, $Avg\_activeUE\_DL$). All other algorithms outperform MLR, with $MAPE$ values below 18%, but only MLP–DNN achieves a $MAPE$ below the 10% threshold (8.79%). FS-COR models strongly degrade their accuracy for all regression algorithms. For instance, in MLP–SNN, $MAPE$ grows from 10.64% to 21.68% when comparing FULL and FS-COR models (i.e., an increase of 103% in relative terms). These numbers are consistent with the above statement about the non-linear relationship among input and output features, since FS-COR is a FS process based on linearity. Fig. 4 shows the evolution of $MAPE$ across FS–SFS process. In this case, even for algorithms not based on distance, such as MLP–SNN, the larger number of features does not necessarily lead to a higher accuracy, revealing that some of the considered input features are irrelevant for estimating $ThUser_{HSDPA}$ or provide redundant information. MLP–SNN achieves the best results when N_{feat} is between 4 and 10, whereas MLR clearly shows the worst performance at every point. Again, the best point is $N_{feat}\leq 5$ for all algorithms but MLP–DNN, whose best performance is with N_{feat}=10. $MAPE$ obtained over the test dataset at those points are shown in Table VI. Results show that, unexpectedly, most FS–SFS models outperform FULL models (e.g., in KNN, $MAPE$ decreases from 17.09% to 12.33%). Overall, the best results are obtained with the combination MLP–SNN+FS–SFS, being the only one achieving a $MAPE$ lower than 10% (9.23%) with a reduced subset of input features. The features selected in that model are $16QAM\_usage$, $Avg\_codes\_used\_HSDPA$, $Avg\_SF16\_codes\_HSDPA$, $Avg\_R99\_DL\_power$ and $Avg\_activeUE\_DL$.

C. Results – LTE

Table VII summarizes the results obtained for the considered algorithms when estimating $ThCell_{LTE}$ and $ThUser_{LTE}$

TABLE VII: $MAPE$ performance in LTE.

	$ThCell_{LTE}$			$ThUser_{LTE}$		
Model	FULL	FS–COR	FS–SFS	FULL	FS–COR	FS–SFS
N_{feat}	10	1	—	10	2	—
MLR	9.09	14.12	9.84 (N_{feat} = 5)	17.79	23.09	18.03 (N_{feat} = 5)
SVR	7.36	13.72	8.32 (N_{feat} = 5)	12.36	17.33	12.14 (N_{feat} = 5)
RF	7.25	16.29	7.36.21 (N_{feat} = 5)	10.04	16.72	10.04 (N_{feat} = 6)
KNN	7.64	15.58	8.86 (N_{feat} = 4)	10.13	16.59	9.62 (N_{feat} = 5)
MLP–SNN	6.96	13.93	8.98 (N_{feat} = 8)	7.95	15.17	8.86 (N_{feat} = 9)
MLP–DNN	6.86	14.11	8.34 (N_{feat} = 5)	8.86	17.73	8.73 (N_{feat} = 9)

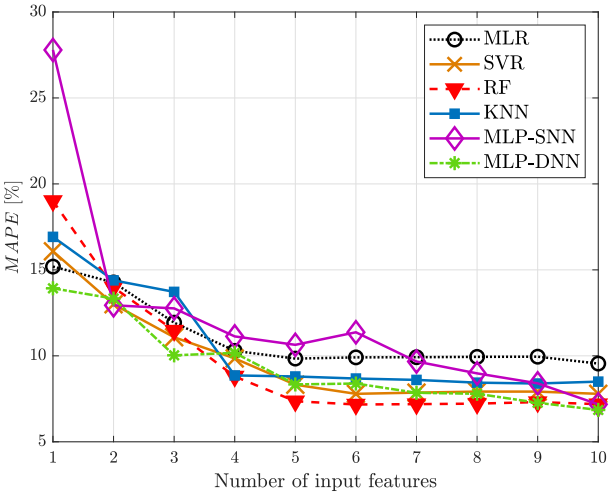


Fig. 5: $MAPE$ evolution – $ThCell_{LTE}$ estimation.

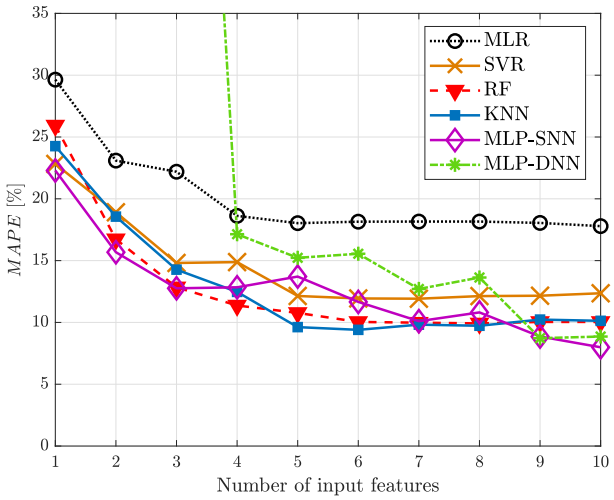


Fig. 6: $MAPE$ evolution – $ThUser_{LTE}$ estimation.

with the FULL, FS-COR and FS–SFS models in the LTE dataset. Again, MLR provides acceptable accuracy with FULL model when estimating $ThCell_{LTE}$ ($MAPE$=9.09%), but not when estimating $ThUser_{LTE}$ ($MAPE$=17.79%). All other algorithms outperform MLR in both cell and user throughput estimations. When estimating $ThCell_{LTE}$ with FULL model, SL algorithms perform very similar ($MAPE$ ≈7%). However, when estimating $ThUser_{LTE}$, only MLP–SNN and MLP–DNN fulfill the 10% threshold ($MAPE$ ≈ 8% and 9%, respectively). FS-COR models degrade accuracy significantly, showing that the most relevant features do not have a strong linear relation to the output variables. In fact, an analysis of Pearson correlation coefficients (not shown here) reveals that only $HARQ\_fail\_ratio\_DL$ has a significant linear correlation with $ThCell_{LTE}$, and only $HARQ\_fail\_ratio\_DL$ and $Avg\_activeUE\_DL$ have a significant linear correlation with $ThUser_{LTE}$.

Fig. 5 and 6 show the $MAPE$ evolution across FS–SFS process when estimating $ThCell_{LTE}$ and $ThUser_{LTE}$, respectively. In general, the larger number of features, the higher accuracy. The best option for estimating each indicator depends on storage constraints (e.g., when estimating $ThUser_{LTE}$, $MAPE$≈9% for MLP–SNN and MLP–DNN with N_{feat}=9, but $MAPE$=9.62% for KNN with N_{feat}=5). Table VII includes, in FS–SFS columns, the $MAPE$ of each method with the selected value of N_{feat}. When considering the trade-off between $MAPE$ and N_{feat} values, KNN is the best option for estimating both $ThCell_{LTE}$ ($MAPE$=8.86% when N_{feat}=4) and $ThUser_{LTE}$ ($MAPE$=9.62% when N_{feat}=5). The most relevant input features for estimating cell throughput are Avg_CQI, DL_assign_ACK, BW and $HARQ\_fail\_ratio\_DL$. Unlike [9], DL_assign_ACK is selected instead of $Avg\_activeUE\_DL$. Likewise, most relevant input features for estimating user throughput are CQI_class_p10, Avg_CQI, $Avg\_activeUE\_DL$, $PUCCH\_SR\_users$ and DL_assign_ACK.

It should be pointed out that, among the considered regression algorithms, MLP approaches have the largest number of hyperparameters. The optimal value of these hyperparameters may vary at each step of the sequential feature selection (SFS) process. In this work, for efficiency, only the most relevant parameters have been tuned (as network operators do). At the same time, MLP models also have a larger number of internal parameters (e.g., up to 700 for FULL MLP–DNN model), which makes them prone to overfitting with reduced datasets (e.g., with few input features). These are probably the reasons for the unstable behavior of MLP approaches across SFS, translated into peaks in MAPE (e.g., MLP–SNN with N_{feat}=6 in Fig. 5 and N_{feat}=5 in Fig. 6) and severe performance degradation below a certain number of features (e.g., MLP–DNN with N_{feat}≤2 in Fig. 3 and Fig. 4, or N_{feat}≤3 in Fig. 6).

D. Computational complexity

The time to perform throughput estimation comprises the time of building the datasets, pre-processing data, training and testing the models. The most time-consuming task is model training (note that, for FS–SFS models, the model has to be trained N_{feat} times).

For MLR, training time grows linearly with the number of input features, N_{feat}, and the number of samples in the dataset, N_s. Thus, the worst-case time complexity is $O(N_s \times N_{feat})$. The back propagation algorithm used to train a MLP with N_{feat} inputs, 1 output and 3 layers, has a worst-case time complexity of $O(N_s \times N_{feat} \times N_l \times N_i)$, where N_l is the size of the hidden layer and N_i is the number of iterations. Time complexity of sequential minimal optimization used to train SVR is quadratic with the training set size, $O(N_s^2)$. Likewise, the worst-case time complexity of RF is given by the time of building a complete decision tree, $O(N_{feat} \times N_s \times \log N_s)$. Finally, for KNN, the worst-case complexity is given by $O(N_{feat} \times N_s \times k)$., where k is the number of neighbors.

Table VIII summarizes the time taken to train the models in a centralized server with Intel Xenon octa-core processor, clock frequency of 2.4 GHz and 64 GB of RAM. For clarity, only the FULL models are tested. Results show that model training in HSDPA is faster than in LTE, maybe due to the highest number of data points in dataset B. For a given technology and regression algorithm, training is faster when estimating cell throughput than when estimating user throughput. MLP–DNN takes the highest execution time for every technology and output feature, whereas MLR and KNN take the lowest execution times. Nonetheless, even in the worst case (i.e., training a MLP–DNN model to estimate $ThUser_{DL}$, the obtained execution time is less than one minute (specifically, 33 s). Such time is negligible in network planning tools, where new cell/user capacity models must be created only when significant changes in terminal and base station capabilities are introduced (e.g., change of release). For other applications with real-time constraints, specialized hardware (e.g., field-programmable gate arrays or application-specific integrated circuits) can be used to reduce execution times.

TABLE VIII: Execution times (seconds).

	HSDPA		LTE	
	$ThCell$	$ThUser$	$ThCell$	$ThUser$
MLR	<0.01	<0.01	0.02	<0.01
SVR	0.05	0.09	0.84	0.17
RF	1.14	2.53	3.02	3.89
KNN	<0.01	<0.01	<0.01	<0.01
MLP–SNN	0.23	0.71	0.66	0.95
MLP–DNN	22.92	23.72	32.47	33.28

VI. Conclusions

Accurate cell and user capacity estimation is crucial for network dimensioning in current cellular networks. Moreover, the diversity of services and terminals expected in upcoming 5G networks will boost the importance of capacity estimates for the dynamic provisioning and monitoring of network slices [43]. Throughput is considered an adequate metric for cell/user capacity, since it is closely related to user satisfaction. In this work, a comparative study has been carried out to assess the performance of different supervised learning algorithms to estimate cell and user throughput in the DL in busy hours from network measurements collected in the OSS. Analysis has been carried out with two datasets taken from live HSDPA and LTE networks. Four no-deep supervised learning methods have been compared with classical multiple linear regression and deep learning approaches considered in previous works.

Results show that classical MLR performs well when estimating cell throughput in both HSDPA and LTE ($MAPE$=7.37% and 9.09%, respectively), but not when estimating user throughput ($MAPE$=41.14% and 17.79%, respectively), probably due to the non-linear relationship between cell-level indicators and user-level metrics. Nonetheless, the other approaches outperform MLR in terms of accuracy in both cell and user throughput estimation problems. Deep learning approach achieves adequate accuracy (i.e., $MAPE$<10%) in all cases when a full set of network indicators is available. However, its performance strongly degrades when the number of features decreases, and is thus not suitable for applications with storage constraints, such as network planning tools Alternatively, with non-deep supervised learning, it is possible to estimate cell/user throughput with similar accuracy by means of reduced datasets (less than 2,000 samples and collection of 5 or 6 indicators in the OSS). To achieve this goal, a feature selection process must be performed by wrapper methods.

When considering the trade-off between accuracy and storage capacity, MLP–SNN has shown the best results in HSDPA, with $MAPE$=6.80% with 4 input features when estimating cell throughput, and $MAPE$=9.23% with 5 input features for user throughput. In contrast, in LTE, KNN algorithm has shown the best trade-off between accuracy and storage capacity, with $MAPE$=8.86% with 4 features for cell throughput, and $MAPE$=9.62% with 5 features for user throughput. Future work will focus on 5G system by extending the analysis to the up link, which will have a strong influence on enhanced mobile broadband services (e.g., live video upload) and massive machine-type communications (e.g., sensor networks), and other capacity indicators more significant for delay-sensitive services (e.g., mission critical services).

Acknowledgment

This work has been funded by the Spanish Ministry of Science, Innovation and Universities (RTI2018-099148-B-I00), Junta de Andalucía (UMA18-FEDERJA-256) and the Spanish Ministry of Education, Culture and Sports (FPU17/04286).

References

[1] Ericsson, "Ericsson Mobility Report," Nov. 2018.
[2] NGMN, "5G White paper," *White paper*, 2015.
[3] M. Toril, R. Ferrer, S. Pedraza, V. Wille, and J. J. Escobar, "Optimization of half-rate codec assignment in GERAN," *Wireless Personal Communications*, vol. 34, no. 3, pp. 321–331, 2005.

[4] C. Gijon, M. Toril, S. Luna-Ramirez, and M. L. Mari-Altozano, "A data-driven traffic steering algorithm for optimizing user experience in multi-tier lte networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 9414–9424, 2019.

[5] E. Onur, H. Deliç, C. Ersoy, and M. U. Çağlayan, "Measurement-based replanning of cell capacities in GSM networks," *Computer Networks*, vol. 39, no. 6, pp. 749–767, 2002.

[6] J. Moysen, L. Giupponi, and J. Mangues-Bafalluy, "A mobile network planning tool based on data analytics," *Mobile Information Systems*, vol. 2017, 2017.

[7] Net2Plan, "The open source network planner." Available in: http://www.net2plan.com/index.php. Online. Accessed: Sep 12, 2019.

[8] V. Wille, M. Toril, and S. Luna-Ramirez, "Estimating pole capacity in a live HSDPA network," *IEEE Communications Letters*, vol. 17, no. 6, pp. 1260–1263, 2013.

[9] J. A. Fernández-Segovia, S. Luna-Ramírez, M. Toril, and J. J. Sánchez-Sánchez, "Estimating cell capacity from network measurements in a multi-service LTE system," *IEEE Communications Letters*, vol. 19, no. 3, pp. 431–434, 2015.

[10] D. Parracho, D. Duarte, I. Pinto, and P. Vieira, "An improved capacity model based on radio measurements for a 4G and beyond wireless network," in *21st International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 314–318, 2018.

[11] J. Navarro-Ortiz, J. M. Lopez-Soler, and G. Stea, "Quality of experience based resource sharing in IEEE 802.11 e HCCA," in *2010 European Wireless Conference (EW)*, pp. 454–461, 2010.

[12] L. R. Jiménez, M. Solera, and M. Toril, "A network-layer QoE model for YouTube live in wireless networks," *IEEE Access*, vol. 7, pp. 70237–70252, 2019.

[13] B. Sas, E. Bernal-Mor, K. Spaey, V. Pla, C. Blondia, and J. Martinez-Bauset, "Modelling the time-varying cell capacity in LTE networks," *Telecommunication Systems*, vol. 55, no. 2, pp. 299–313, 2014.

[14] G. Aceto, F. Palumbo, V. Persico, and A. Pescapé, "Available bandwidth vs. achievable throughput measurements in 4G mobile networks," in *14th International Conference on Network and Service Management (CNSM)*, pp. 125–133, 2018.

[15] A. S. Khatouni, M. Mellia, M. A. Marsan, S. Alfredsson, J. Karlsson, A. Brunstrom, O. Alay, A. Lutu, C. Midoglu, and V. Mancuso, "Speedtest-like measurements in 3G/4G networks: The MONROE experience," in *29th International Teletraffic Congress (ITC 29)*, vol. 1, pp. 169–177, 2017.

[16] R. Senapati and H. K. Pati, "VoLTE cell capacity estimation using AMR-WB codec," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI 2018)*, pp. 1885–1889, 2018.

[17] D. Parracho, D. Duarte, I. Pinto, and P. Vieira, "An enhanced capacity model based on network measurements for a multi-service 3G system," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, pp. 203–208, 2019.

[18] M. Assaad and D. Zeghlache, "On the capacity of HSDPA," in *IEEE Global Telecommunications Conference (GLOBECOM'03)*, vol. 1, pp. 60–64, 2003.

[19] O. Østerbø, "Scheduling and capacity estimation in LTE," in *23rd International Teletraffic Congress (ITC 2011)*, pp. 63–70, 2011.

[20] C. Dou and Y.-H. Chang, "Class-based downlink capacity estimation of a WCDMA network in a multiservice context," *Computer Communications*, vol. 28, no. 12, pp. 1443–1455, 2005.

[21] K. Ivanov, C. Ball, and F. Treml, "GPRS/EDGE performance on reserved and shared packet data channels," in *IEEE 58th Vehicular Technology Conference (VTC 2003-Fall)*, vol. 2, pp. 912–916, 2003.

[22] K. I. Pedersen, F. Frederiksen, T. E. Kolding, T. F. Lootsma, and P. E. Mogensen, "Performance of high-speed downlink packet access in coexistence with dedicated channels," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 3, pp. 1262–1271, 2007.

[23] K. Kousias, Ö. Alay, A. Argyriou, A. Lutu, and M. Riegler, "Estimating downlink throughput from end-user measurements in mobile broadband networks," in *IEEE 20th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*, pp. 1–10, 2019.

[24] K. Chang and R. P. Wicaksono, "Estimation of network load and downlink throughput using RF scanner data for LTE networks," in *2017 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, pp. 1–8, 2017.

[25] P. J. M. Johansson and Y.-S. Chen, "Location for minimization of drive test in lte systems," 2014. US Patent 8,903,420.

[26] T. ur Rehman, M. A. I. Baig, and A. Ahmad, "LTE downlink throughput modeling using neural networks," in *IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 265–270, 2017.

[27] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, *The elements of statistical learning: data mining, inference and prediction, Second edition*. Springer series in statistics, 2001.

[28] D. Palacios, S. Fortes, I. de-la Bandera, and R. Barco, "Self-healing framework for next-generation networks through dimensionality reduction," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 170–176, 2018.

[29] R. Kwan, R. Arnott, R. Paterson, R. Trivisonno, and M. Kubota, "On mobility load balancing for LTE systems," in *IEEE 72nd Vehicular Technology Conference Fall (VTC-2010-Fall)*, pp. 1–5, 2010.

[30] V. Buenestado, J. M. Ruiz-Aviles, M. Toril, S. Luna-Ramírez, and A. Mendo, "Analysis of throughput performance statistics for benchmarking lte networks," *IEEE Communications letters*, vol. 18, no. 9, pp. 1607–1610, 2014.

[31] 3rd Generation Parthnership Project, "Key Performance Indicators (KPI) for Evolved Universal Terrestrial Radio Access Network (E UTRAN): Definitions," in *TS 32.450, version 9.1.0*, 2018.

[32] S. Sesia, M. Baker, and I. Toufik, *LTE - the UMTS long term evolution: from theory to practice*. John Wiley & Sons, 2011.

[33] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*. Elsevier, 2011.

[34] B. Efron, "Bootstrap methods: another look at the jackknife," in *Breakthroughs in statistics*, pp. 569–593, Springer, 1992.

[35] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *13th International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

[36] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[37] J. Moysen, L. Giupponi, and J. Mangues-Bafalluy, "On the potential of ensemble regression techniques for future mobile network planning," in *IEEE Symposium on Computers and Communication (ISCC)*, pp. 477–483, 2016.

[38] M. Claesen and B. De Moor, "Hyperparameter search in machine learning," *arXiv preprint arXiv:1502.02127*, 2015.

[39] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 Science and Information Conference*, pp. 372–378, 2014.

[40] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3 (March), pp. 1157–1182, 2003.

[41] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

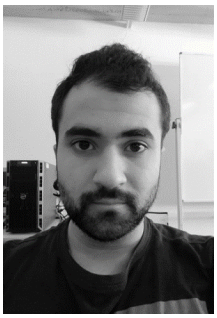
[42] F. Chollet *et al.*, "Keras: Deep learning library for theano and tensorflow." Available in: https://keras.io. Online. Accessed: Jun 12, 2020.

[43] 4G Americas, "Network slicing for 5G networks and services," 2016.

Carolina Gijón received her B.Sc. degree in Telecommunication Systems Engineering and her M.Sc. Degree in Telecommunication Engineering from the University of Málaga, Spain, in 2016 and 2018, respectively. Currently, she is working towards the Ph.D. degree. Her research interests include self-organizing networks, machine learning and radio resource management.

Matías Toril received his M.S in Telecommunication Engineering and the Ph.D degrees from the University of Málaga, Spain, in 1995 and 2007 respectively. Since 1997, he is Lecturer in the Communications Engineering Department, University of Málaga, where he is currently Full Professor. He has co-authored more than 130 publications in leading conferences and journals and 8 patents owned by Nokia or Ericsson. His current research interests include self-organizing networks, radio resource management and data analytics.

Salvador Luna-Ramírez received his M.S in Telecommunication Engineering and the Ph.D degrees from the University of Málaga, Spain, in 2000 and 2010, respectively.Since 2000, he has been with the department of Communications Engineering, University of Málaga, where he is currently Associate Professor. His research interests include self-optimization of mobile radio access networks and radio resource management.



Juan L. Bejarano-Luque received the B.S. degree in telecommunications engineering and the M.S. degree in acoustic engineering from the University of Málaga, Málaga, Spain, in 2015 and 2016, respectively. He is currently pursuing the Ph.D. degree in telecommunications engineering at the same university. His research interests include optimization of radio resource management for mobile networks, location-based services and management and data analytics.

María Luisa Marí-Altozano received her M.S. degree in Telecommunication Engineering from the University of Málaga, Spain, in 2012. From 2013 to 2016, she was with Ericsson in a collaborative project with the University of Málaga. Since 2017, she has been working toward the Ph.D with the Communication Engineering Department, University of Málaga. Her interests are focused on self-optimization of mobile radio access networks based on quality of experience.