Exploiting Spatial Correlation for Improved Prediction in 5G Cellular Networks

Michele Polese*, Rittwik Jana†, Velin Kounev†, Ke Zhang†, Supratim Deb†, Michele Zorzi*
*Department of Information Engineering (DEI), University of Padova, Italy. Email:{polesemi,zorzi}@dei.unipd.it.
†AT&T Labs, Bedminster, NJ 07921 USA. Email: rjana@research.att.com, {vk0366, kz3722}@att.com, supratim.deb@gmail.com.

***Abstract*—The complexity of the fifth generation of cellular systems (5G) will call for the deployment of advanced Machine Learning (ML) techniques to control and optimize the network. In this paper, we present an application of machine learning to predict the number of users in each base station of a cellular network, based on a dataset with real data collected from hundreds of base stations of a major U.S. national operator. We show that by exploiting the spatial correlation introduced by the mobility of the users it is possible to improve the prediction accuracy with respect to that of completely distributed methods, based on local information only, with a reduction of the prediction error by up to 53%. Finally, we describe a use case in which these predictions are used by a higher-layer application to route vehicular traffic according to network Key Performance Indicators (KPIs).**

I. Introduction

The future generations of cellular networks will be designed to satisfy the communication needs of the connected society for 2020 and beyond. According to [2], 5G networks will support data rates and spectral efficiencies much higher than 4G, with multi-gigabit-per-second peak data rates, ultra-low latency (i.e., 1 ms round-trip in the radio access), highly mobile communications, up to 10^6 connections per km^2 and a 100-fold increment in energy efficiency.

The solutions that next generation networks will introduce to satisfy these requirements (e.g., mmWave communications, ultra-dense networks, massive MIMO [3]) will increase the complexity of the network, which will consequently need to self-optimize and operate in an autonomous way. In this context, the integration of intelligence in the network through ML and Artificial Intelligence (AI) is seen as a promising enabler of self-organizing approaches [4], and also new applications, as we will show in this paper. The usage of ML and AI techniques to perform autonomous operations in cellular networks has been widely studied in recent years, with use cases that range from optimization of video flows [5] to energy-efficient networks [6] and resource allocation [7]. This direction is combined with the usage of big-data analytics, that leverages the massive amounts of data generated in mobile networks to yield deeper insights on the behavior of networks at scale [8]. However, despite the importance of this topic, the state of the art lacks considerations on the actual gains that ML approaches can introduce in real large-scale cellular network scenarios.

An extended version of this paper is presented in [1]. Supratim Deb is current employed by Facebook; work was done when the author was working at AT&T Labs.

In this paper we focus on the application of machine learning techniques to predict the number of active users in the base stations of a cellular network. We propose to exploit the spatial correlation that naturally exists in a cellular networks, and is introduced by the constrained mobility of the users in an urban environment, to improve the performance of the prediction algorithms with respect to the state of the art, with a reduction of the prediction error by up to 53%. We characterize the accuracy of the proposed prediction scheme by leveraging a dataset from a major U.S. operator, with hundreds of base stations in the San Francisco area. We test the proposed solution (based on Gaussian Process Regressors (GPRs)) against other techniques (Random Forest Regressors (RFRs), Bayesian Ridge Regressors (BRRs)), leveraging the architecture and clustering process described in [1]. Finally, we describe a possible application of the ML pipeline for the prediction of KPIs in a cellular network, where the users of the mobile networks drive through a city and choose among multiple available routes also according to the predicted quality of the connection on any path.

The remainder of the paper is organized as follows. In Sec. II we review the relevant state of the art, and in Sec. III we describe the dataset that will be used in this paper. Then, in Sec. IV we introduce the relevant ML algorithms, the prediction pipeline and the results we obtained, and illustrate the application of the ML infrastructure in Sec. V. Finally, we conclude the paper in Sec. VI.

II. State of The Art

The deployment of ML techniques in cellular networks is a theme that has drawn a lot of attention recently, due to the restored importance of ML and AI throughout all facets of the industry. The surveys in [9], [10] review recent results related to the application of regression and classification techniques to mobile and cellular scenarios, to optimize the performance of complex networks. Reference [11] gives an overview of how machine learning can play a role in next-generation 5G cellular networks, and lists relevant ML techniques and algorithms. The usage of big-data-driven analytics for 5G is considered in [12], [13], with a discussion on how data-driven approaches can empower self-organizing networks. However, none of these papers provides results based on real large-scale datasets of cellular operators that show the actual gains of data-driven and machine-learning-based approaches.

Furthermore, several papers report results on the prediction of mobility patterns of users in cellular networks. The authors of [14], [15] use network traces to study human mobility

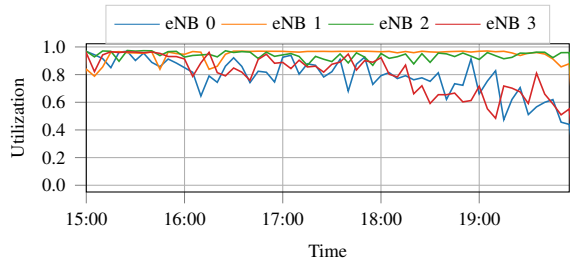
patterns, with the goal to infer large-scale patterns and understand city dynamics. Reference [16] proposes to use a leap graph to model the mobility pattern of single users. With respect to the state of the art, in this paper we focus on the prediction of the number of users associated to a base station, in order to provide innovative services to the users themselves, and propose a novel cluster-based approach to improve the prediction accuracy.

III. The Dataset

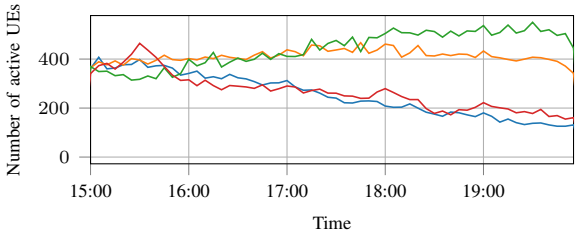
This section describes the dataset that will be used to train and test the machine learning algorithms discussed in this paper. The network traces we utilize are based on the monitoring logs of 472 base stations of a national U.S. operator in the San Francisco area, for more than 600000 User Equipments (UEs) per day, properly anonymized during the collection phase. The base stations in the dataset belongs to a 4G LTE-A deployment, which at the time of writing represents the most advanced cellular technology commercially deployed at a large scale. We argue that, even if 5G NR networks will have more advanced characteristics than Long Term Evolution (LTE), this dataset can represent a first NR deployment at sub-6 GHz frequencies in a dense urban scenario. The measurement campaign was run in February 2017 (01/31/2017 − 02/26/2017), with monitoring logs collected every day from 3 P.M. to 8 P.M.. Fig. 1 shows an example of time series for different metrics from 4 LTE evolved Node Bases (eNBs), with a time step of 5 minutes.

Given the sensitive nature of this data, we applied standard policies to make sure that individuals' privacy was not undermined with the data collection and processing. In this regard, the International Mobile Subscriber Identity (IMSI) (i.e., the identifier associated to a single user in the traces) of each UE was anonymized through hashing. Additionally, the analysis in this paper only uses aggregate metrics, which do not single out the behavior of any particular user. First, user data is grouped for each cell (i.e., mapped to a sector and carrier frequency) and, then, the data for the cells in the same base station (i.e., with the RF equipment in the same physical location) is aggregated again.

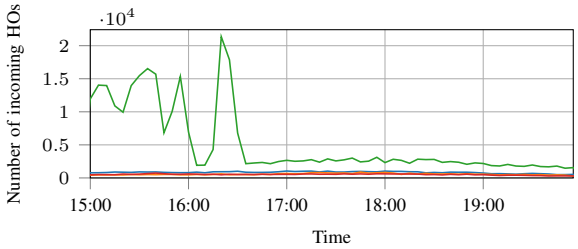
The traces used in this paper register a number of standardized events in LTE eNBs, mostly involving the mobility of users. The raw data is further processed to define time series of different quantities of interest in each eNB at different time scales (from minutes to weeks), such as (i) the eNB utilization, represented by the ratio of used and available Physical Resource Blocks (PRBs); (ii) the number of incoming and outgoing handovers; and (iii) the number of active UEs, i.e., connected and involved in a data exchange. Other metrics could also be extracted, for example related to the user latency, link statistics (e.g., error probability), or different estimates of the user and cell throughput, but the logs reporting these quantities are less frequent and regular than those we consider, and do not represent an accurate source for the estimation of the network performance.



(a) Utilization (averaged over a 15-minute interval).



(b) Number of active UEs (summed over a 15-minute interval).



(c) Number of incoming handovers (summed over a 15-minute interval).

Fig. 1: Example of time series from the traces collected for 4 eNBs in the San Francisco dataset over 7 days.

IV. Predicting the Number of Users in Base Stations

In the following paragraphs, we will present procedures and results for the prediction of the number of active users in the base stations of a cellular network. This information can be exploited to predict other relevant KPIs, e.g., the network load, or the user throughput. We will compare the accuracy of the prediction according to two different methods. The first uses only *local information* (i.e., available in each single base station) to perform the training and the prediction. This strategy can be used in networks where there is no or limited coordination among base stations, which are complete and self-contained pieces of equipment, as in 4G LTE networks. The second strategy, instead, relies on the availability of *shared information* from a set of neighboring base stations, given that it aims at jointly predicting the number of users in each one, based on the common history of the cluster of base stations. Therefore, it requires a higher level of coordination and information exchange between the base stations and a network controller. For example, as described in [1], this approach can be implemented in a 5G NR network deployed following the xRAN/Open RAN paradigms, where groups of Next Generation Node Bases (gNBs) are associated with edge-based network controllers that handle their control plane.

Considering the San Francisco dataset described in Sec. III, we will first focus on prediction results for a specific set of 22 base stations, clustered with the procedure described in [1], and then extend the analysis to all the 472 available nodes, showing how a cluster-based approach reduces the prediction error with respect to a local-based approach.

A. Data Preprocessing

The number of users in the base stations of the San Francisco dataset has been sampled with a time step of $T_s = 5$ minutes. The time series were then split into a training set (used for k-fold cross validation) and a test set. In particular, the training set ranges from January 31st to February 20th, and the test set is in the interval between February 21st and February 26th.

Consider $\mathcal{B}$ as the set of all the base stations in San Francisco. The goal is to obtain a multi-step ahead prediction of the number of users $N_u^i(t+L)$ at times $t+1,\dots,t+L$, where $i \in \mathcal{B}$ is the identifier of the base station, and $L \geq 1$ is the look-ahead step of the forecast. The prediction is based on the data collected before time t, which is characterized by three different features. The first two are a boolean $b(t)$ that specifies if the considered day is a weekday, and an integer $h(t) \in \{0,\dots,4\}$ that indicates the hour of the day (from 3 P.M. to 8 P.M.). The third feature is given by the past W samples of the number of users, with W the window of the history used for the forecast, i.e., $N_u^i(t+\tau), \tau \in [-W+1, 0]$. We also analyzed other possible features, such as the number of handovers and the cell utilization, but they exhibited a small correlation with the prediction target. Given the daily discontinuities of the dataset, we discard the first W samples of each day, therefore the size of the training (N_{tr}) and test (N_{te}) sets depends on the value of W.

For the local-based prediction, in which each base station learns the future number of users using only its own data, the training and test set are represented by the feature matrix $\mathbf{X} \in \mathbb{R}^{N_k, 3W}, k \in \{tr, te\}$, with a vector

$$[N_u^i(t-W+1), h(t-W+1), b(t-W+1)\dots, N_u^i(t), h(t), b(t)] \quad (1)$$

in each row, and by the target vector $\mathbf{y} \in \mathbb{R}^{N_k,1}, k \in \{tr, te\}$. With the cluster-based method, instead, the target of the prediction is the vector of the numbers of users for all the base stations in the cluster. Thus, for the set $\mathcal{C}_d = \{i_d,\dots,j_d\} \subset \mathcal{B}$ with the N_b^d base stations of cluster d, each row of the target matrix $\mathbf{Y} \in \mathbb{R}^{N_k, N_b^d}, k \in \{tr, te\}$ is composed by a vector

$$[N_u^{i_d}(t+L),\dots,N_u^{j_d}(t+L)]. \quad (2)$$

Each row of the feature matrix $\mathbf{X} \in \mathbb{R}^{N_k, W(N_b^d+2)}, k \in \{tr, te\}$ is instead a vector

$$[N_u^{i_d}(t-W+1),\dots, N_u^{j_d}(t-W+1), h(t-W+1), b(t-W+1),\dots, N_u^{i_d}(t),\dots,N_u^{j_d}(t), h(t), b(t)]. \quad (3)$$

Bayesian Ridge Regressor [18], [19]	
α	$\{10^{-6}, 10^{-3}, 1, 10, 100\}$
λ	$\{10^{-6}, 10^{-3}, 1, 10, 100\}$
Random Forest Regressor [20], [21]	
Number of trees N_{rf}	$\{1000, 5000, 10000\}$
Gaussian Process Regressor [22]	
α	$\{10^{-6}, 10^{-4}, 10^{-2}, 0.1\}$
σ_k	$\{0.001, 0.01\}$

TABLE I: Values of the hyperparameters of the different regressors for the k-fold cross-validation.

The values of the numbers of users in the training and test sets are transformed with the function $\log(1+x)$ and scaled between 0 and 1. The scaling is fitted on the training set, and then applied also to the test set. The metric we consider for the performance evaluation of the different methods and prediction algorithms is the Root Mean Squared Error (RMSE), defined for a single base station i as

$$\sigma_i = \sqrt{1/N_{te} \sum_{t=1}^{N_{te}} (y_i(t) - \hat{y}_i(t))^2}, \quad (4)$$

with y_i the time series of the actual values for the number of users for base station i, and $\hat{y}_i$ the predicted one.

B. Algorithm Comparison

We consider and compare various machine learning algorithms for prediction: the BRR for the local-based prediction, and the RFR and the GPR for both the local- and the cluster-based forecasts.[1] We used the implementations from the well-know open-source library scikit-learn [17]. For each algorithm, we compared different values of the past history window $W \in \{1,\dots,10\}$ and computed the prediction at values of the future step $L \in \{1,\dots,9\}$, i.e., over a maximum time span of 45 minutes. We performed 3-fold cross-validation to select the best hyperparameters for each regressor and values of L and W. The range of the hyperparameters we tested is summarized in Table I. Each fold is split using the `TimeSeriesSplit` class of scikit-learn, i.e., without shuffling the training set, and with indices monotonically increasing in each split, to keep the consecutive temporal samples ordered in time.

The BRR, which was used for traffic prediction in an urban scenario in [19], integrates a Bayesian probabilistic approach and the ridge L_2 regularization [18]. The Bayesian approach automatically fits the available data, and only needs the selection of the α and λ parameters of the Gamma priors. However, it does not account for multi-output prediction, thus it can be applied only to the local-based scenario.

The RFR, which for example was used in [21] for population forecast, is a classic ensemble algorithm that (i) trains N_{rf} regression trees from samples bootstrapped from the training set and (ii) averages their output for the prediction [20]. Only

[1]We also considered a strategy based on neural networks (i.e., Long Short Term Memory (LSTM)), however, given the reduced size of the training set, it underperformed with respect to the other regression methods.

Fig. 2: Deployment of a sample cluster of San Francisco base stations involved in the joint prediction described in Sec. IV-C, obtained using the method described in [1].

two hyperparameters need to be tuned, i.e., the number of trees N_{rf} and the number of random features to sample. For N_{rf}, a higher value yields improved generalization capabilities, at the cost of a longer training time. We set the number of random features to sample when splitting the nodes to build additional tree branches equal to the number of features for regression problems, i.e., the number of columns in the training/testing matrix $\mathbf{X}$. RFRs support the prediction of both scalars and vectors, therefore they can be applied in the local- and the cluster-based scenarios.

The third algorithm is GPR, i.e., a regressor that fits a Gaussian Process over the input data [22]. It uses a prior with zero mean, and the covariance matrix determined by a kernel. For this problem, we chose a kernel with the following form:

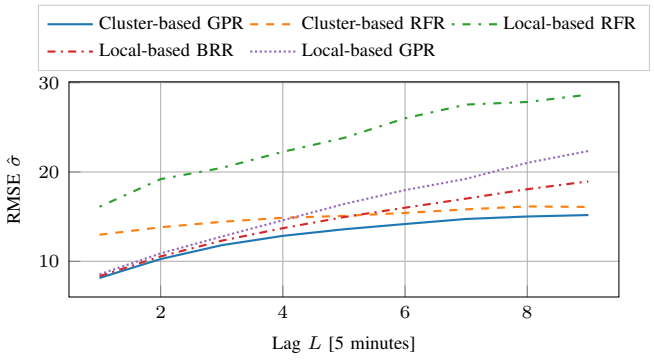
$$k(x_i, x_j) = \sigma_k^2 + x_i \cdot x_j + \left(1 + \frac{d(x_i, x_j)^2}{2\alpha l^2}\right)^{-\alpha} + \delta_{x_i x_j}. \quad (5)$$

It is given by the sum of a dot product kernel, that can fit non-stationary behaviors, a rational quadratic kernel with $l = 1$, and a white kernel, that models the noisy part of the input. We used the GPR for both single-output and multi-output predictions.

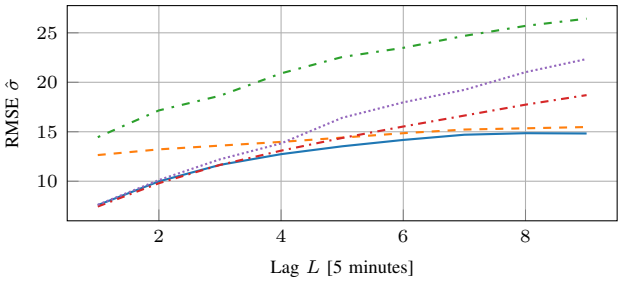
C. Performance analysis

Before applying the aforementioned regressors to the whole available dataset, we consider a sample cluster of $N_d^0 = 22$ base station (obtained with the clustering process described in [1]), whose relative positions are shown in Fig. 2.

Fig. 3 shows the average RMSE $\hat{\sigma} = \mathbb{E}_{i \in \mathcal{C}_0}[\sigma_i]$ of the base stations in the set $\mathcal{C}_0$ associated to the sample cluster, for different methods using either the local information only, or the cluster-based approach, and a fixed value of the past window $W = 1$. Among the local-based algorithms, the BRR shows the best performance for all the values of the look-ahead step L, with a reduction of the RMSE of up to 18% and 55% with respect to the GPR and RFR for $L = 9$. The GPR, instead, yields better results than the RFR for the cluster-based techniques, with an improvement up to 50% (for $L = 1$). As expected, by increasing the look-ahead step L the prediction accuracy decreases. However, when comparing the cluster- and the local-based methods, the former perform better, especially as the look-ahead step increases: the RMSE for the cluster-based GPR saturates around $\hat{\sigma} = 14.8$, while that for both the BRR and the local-based GPR keeps increasing. For small



(a) $W = 1$



(b) A different window W is selected for each method and look-ahead step L to minimize the RMSE $\hat{\sigma}$. The values of W are reported in Table II.

Fig. 3: RMSE $\hat{\sigma}$ for different local- and cluster-based prediction methods, as a function of the look-ahead step L, and for different windows W.

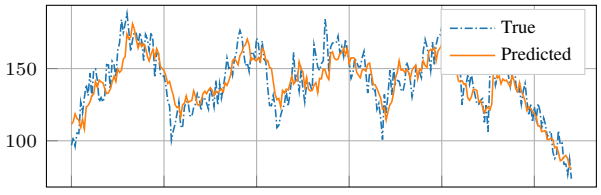
Look-ahead step L	1	2	3	4	5	6	7	8	9
BRR	6	6	4	4	3	3	3	2	2
cluster-GPR	3	2	2	2	2	1	6	5	4

TABLE II: Values of W for the plot in Fig. 3b for the BRR and the cluster-based GPR

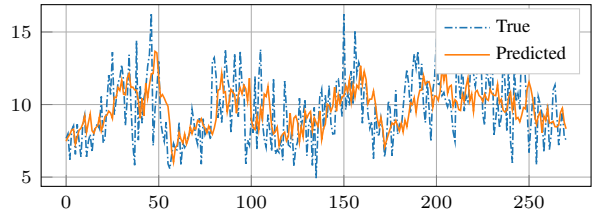
values of L, instead, the accuracy of local- and cluster-based methods is similar.

Fig. 3b instead reports the RMSE varying the value of W, which is selected in order to minimize the RMSE $\hat{\sigma}$ for each prediction method and value of L. Table II reports the values W for the two most accurate methods, the local-based BRR and the cluster-based GPR. It can be seen that, with respect to Fig. 3a, where W is fixed, the difference is limited for the GPR and BRR (i.e., below 5%), while it is more considerable for the local-based RFR.

Furthermore, the spatial dimension (i.e., the usage of a joint prediction based on the cluster) is more impactful on the RMSE than the temporal one (i.e., the past history used as a feature). Indeed, while the RMSE for the GPR and BRR improves by up to 5% by varying W, it decreases by up to 50% when comparing the local- and the cluster-based methods. The target of the prediction is indeed the number of users at a cell level (contrary to prior work which focused on single-user mobility prediction [16]), thus the geography of the scenario in which the base stations are deployed actually limits the possible movements of users across neighboring cells. These constraints on the mobility flow translate into a



(a) High number of users



(b) Low number of users

Fig. 4: Example of predicted vs true time series, for $L = 3$ (i.e., 15 minutes ahead), $W = 3$ and the cluster-based GPR on two base stations for cluster 0.

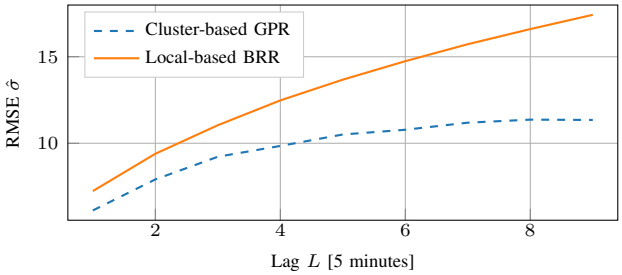


Fig. 5: Average cluster-based GPR vs local-based BRR for all the San Francisco base stations.

spatial correlation among the number of users in neighboring base stations at time t and at time $t + L$.

However, there are still some limitations to the accuracy of the prediction. Fig. 4 reports an example of the true and the predicted (for $L = 3$, i.e., 15 minutes) time series for two different base stations, with a low and high number of users. It can be seen that the true time series exhibit daily patterns, but also a high level of noise. As a consequence, the predicted values manage to track the main trend of the true time series, but do not represent the exact value of the number of users in all cases. This is more noticeable with a low number of UEs, as in Fig. 4b, which also shows more limited daily variations.

Given the promising results of the cluster-based approach on the sample cluster, we selected the best performing local- and cluster-based methods, i.e., respectively, the BRR and the GPR, and performed the prediction on all the base stations of the San Francisco area, once again clustered according to the approach in [1]. The average RMSE over all the base stations is reported in Fig. 5. The cluster-based method consistently outperforms the local-based one. The reduction in the average RMSE over all the clusters $\mathbb{E}_{clusters}[\hat{\sigma}]$ is 18.3% for $L = 1$ (from $\mathbb{E}_{clusters}[\hat{\sigma}] = 7.24$ to $\mathbb{E}_{clusters}[\hat{\sigma}] = 6.11$) and is as large as 53% for $L = 9$ (from $\mathbb{E}_{clusters}[\hat{\sigma}] = 17.42$ to $\mathbb{E}_{clusters}[\hat{\sigma}] = 11.34$).

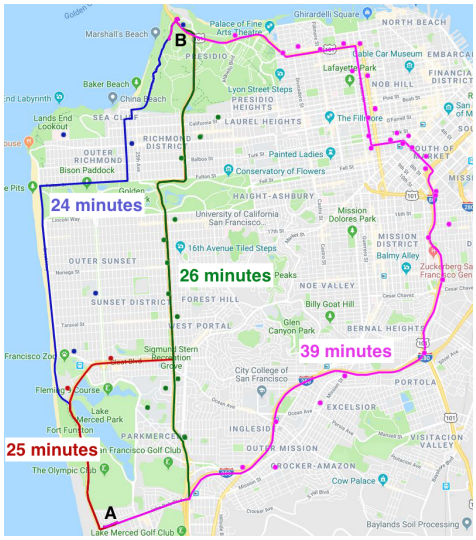


Fig. 6: Map of the routes. The dots represent the visited base stations. Notice that, for route 2 (the red one), several base stations are shared with either the blue or the green routes.

V. Route Optimization with Network KPIs

Figs. 3 and 5 show that, by exploiting the spatial correlation available in real network deployments, it is possible to accurately forecast the conditions of the network (e.g., in terms of number of users) and predict critical situations even on a medium-timescale horizon (e.g., 45 minutes). The knowledge of the future number of users in base stations can be exploited to improve the performance of the network in various ways. For example, it may allow network operators to perform predictive load-balancing, pre-configure bearers, scale the radio resources to meet the expected demands, and so on. We argue that a practical deployment of these prediction-based optimizations can benefit from the reduction in prediction error yielded by the cluster-based method we propose.

Moreover, the prediction can be used by network operators to also provide innovative services to their users. Consider for example a scenario in which a vehicle travels between two locations (e.g., points A and B in Fig. 6). During the journey, the passengers may use a cellular connection to surf the web, stream multimedia content or attend a conference call. Thus, if multiple routes with similar Estimated Times of Arrival (ETAs) are available, the driver may prefer a route to another according to the expected quality of the connection on the path. Therefore, using the procedure described in this paper, the cellular network operators can predict the number of active users in the cells to forecast relevant KPIs and inform the end users on which is the best route for their journey, thus offering them novel predictive services.

Fig. 6 reports an example of four different routes in the San Francisco area, three with a similar ETA and a longer one, as reported in Table III. Table III also provides different metrics (which depend on the predicted number of users) associated to the 4 routes and in different dates. Route 1 (in blue), i.e., the quickest one, does not always offer the best performance for the three departure times considered. For example, among

	Feb. 23rd, 19:00				Feb. 24th, 19:00				Feb. 24th, 19:20			
Route	R1	R2	R3	R4	R1	R2	R3	R4	R1	R2	R3	R4
$\hat{S}$ [Mbit/s]	1.93	2.51	2.36	2.74	1.72	2.00	2.28	2.89	2.05	2.49	1.98	2.86
$D_{o,\max}$ [s]	133.47	157.8	172.5	171.2	152.4	157	148.8	169.1	152.1	123.7	172.5	116.7

TABLE III: Average throughput $\hat{S}$ and maximum outage duration $D_{o,\max}$ on the four itineraries from Fig. 6, for different departure times in February 2017. For the three routes with a similar duration, the colored cells represent the best route for the metric of interest.

the journeys with comparable travel time, route 2 (in red) has the highest throughput on Feb. 23rd, 19:00, while route 3 (in green) is the best at the same time of the next day. Instead, the longest route, which connects points A and B but is 50% longer than the shortest, always offers the highest average throughput, but, in some cases, may be affected by the longest outage.

Therefore, according to the passengers' preferences, multiple routes can be identified, with varying throughput and outage performance. Furthermore, different departure times translate into different rankings for the routes. It is not thus possible to simply rely on previous-days average statistics to rank the routes, and this calls for the adoption of the medium-term prediction strategy described in this paper to get a reliable estimate for the actual time interval in which the user will drive, based on the network conditions for that time interval.

VI. Conclusions

In this paper we described a possible application of machine learning techniques in cellular networks, i.e., the prediction of the number of users which are connected and exchanging data with the base stations of the network. The results we obtained are based on a dataset from a major U.S. operator, comprising 472 base stations in the San Francisco area, with measurements collected for a month in 2017.

We showed for the first time that the prediction of the number of users is more accurate when the spatial correlation introduced by mobility patterns is accounted for, and that the spatial information yields a higher reduction in the prediction error than the temporal one. Finally, we outlined possible applications of the prediction, ranging from network control to novel user services, with an example based on the real network dataset from San Francisco. As extensions of this work, we will compare additional prediction techniques, and apply the regression to other metrics in the network, to understand the limits of predictability in a cellular network.

References

[1] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, and M. Zorzi, "Machine learning at the edge: A data-driven architecture with applications to 5G cellular networks," *submitted to IEEE Journal on Selected Areas of Communications*, 2018. [Online]. Available: https://arxiv.org/abs/1808.07647

[2] ITU-R, "IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond," Recommendation ITU-R M.2083, September 2015.

[3] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, February 2014.

[4] M. Zorzi, A. Zanella, A. Testolin, M. D. F. D. Grazia, and M. Zorzi, "Cognition-based networks: A new perspective on network optimization using learning and distributed intelligence," *IEEE Access*, vol. 3, pp. 1512–1530, 2015.

[5] M. Gadaleta, F. Chiariotti, M. Rossi, and A. Zanella, "D-DASH: A Deep Q-Learning Framework for DASH Video Streaming," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 703–718, Dec 2017.

[6] R. Li, Z. Zhao, X. Zhou, G. Ding, Y. Chen, Z. Wang, and H. Zhang, "Intelligent 5G: When Cellular Networks Meet Artificial Intelligence," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 175–183, October 2017.

[7] S. Chinchali, P. Hu, T. Chu, M. Sharma, M. Bansal, R. Misra, M. Pavone, and K. Sachin, "Cellular network traffic scheduling with deep reinforcement learning," in *National Conference on Artificial Intelligence (AAAI)*, 2018.

[8] Y. He, F. R. Yu, N. Zhao, H. Yin, H. Yao, and R. C. Qiu, "Big data analytics in mobile cellular networks," *IEEE Access*, vol. 4, pp. 1985–1996, 2016.

[9] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, I. Malanchini, and J. Widmer, "A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1790–1821, thirdquarter 2017.

[10] V. Pejovic and M. Musolesi, "Anticipatory mobile computing: A survey of the state of the art and research challenges," *ACM Comput. Surv.*, vol. 47, no. 3, pp. 47:1–47:29, Apr. 2015.

[11] C. Jiang, H. Zhang, Y. Ren, Z. Han, K. C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 98–105, April 2017.

[12] Y. He, F. R. Yu, N. Zhao, H. Yin, H. Yao, and R. C. Qiu, "Big data analytics in mobile cellular networks," *IEEE Access*, vol. 4, pp. 1985–1996, March 2016.

[13] A. Imran, A. Zoha, and A. Abu-Dayya, "Challenges in 5G: how to empower SON with big data for enabling 5G," *IEEE Network*, vol. 28, no. 6, pp. 27–33, Nov 2014.

[14] R. Becker, R. Cáceres, K. Hanson, S. Isaacman, J. M. Loh, M. Martonosi, J. Rowland, S. Urbanek, A. Varshavsky, and C. Volinsky, "Human mobility characterization from cellular network data," *Communications of the ACM*, vol. 56, no. 1, pp. 74–82, Jan 2013.

[15] R. A. Becker, R. Caceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky, "A tale of one city: Using cellular network data for urban planning," *IEEE Pervasive Computing*, vol. 10, no. 4, pp. 18–26, April 2011.

[16] W. Dong, N. Duffield, Z. Ge, S. Lee, and J. Pang, "Modeling cellular user mobility using a leap graph," in *International Conference on Passive and Active Network Measurement*. Springer, 2013, pp. 53–62.

[17] 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, October 2011.

[18] D. J. MacKay, "Bayesian interpolation," *Neural computation*, vol. 4, no. 3, pp. 415–447, May 1992.

[19] Q. Shi, M. Abdel-Aty, and J. Lee, "A Bayesian ridge regression analysis of congestion's impact on urban expressway safety," *Accident Analysis & Prevention*, vol. 88, pp. 124–137, Mar. 2016.

[20] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, October 2001.

[21] R. W. Douglass, D. A. Meyer, M. Ram, D. Rideout, and D. Song, "High resolution population estimates from telecommunications data," *EPJ Data Science*, vol. 4, no. 1, p. 4, Dec. 2015.

[22] C. E. Rasmussen, "Gaussian processes in machine learning," in *Advanced lectures on machine learning*. Springer, 2004, pp. 63–71.