GUMBLE: Uncertainty-Aware Conditional Mobile Data Generation using Bayesian Learning

GUMBLE: Uncertainty-aware Conditional Mobile Data Generation using Bayesian Learning

Marco Skocaj, Lorenzo Mario Amorosa, *Graduate Student Member, IEEE,*
Michele Lombardi, and Roberto Verdone, *Senior Member, IEEE*

Abstract—In the context of mobile and Internet of Things (IoT) networks, data naturally originates at the edge, making crowdsourcing a convenient and inherent approach to data collection. However, crowdsourcing presents challenges related to privacy, sampling bias, statistical sufficiency, and the need for time-consuming post-processing. To this end, generating synthetic data using deep learning techniques emerges as a promising solution to overcome such limitations. In this study, we propose an innovative framework that transcends applications and data types, enabling the conditional generation of crowdsourced datasets with location information in mobile and IoT networks. A crucial aspect of our methodology lies in the ability to assess uncertainty in newly generated samples and produce calibrated predictions through approximate Bayesian methods. Without loss of generality, we ascertain the validity of our method on the task of minimization of drive test (MDT) data generation, presenting for the first time a comparison of synthetically generated data with an original large-scale MDT set collected from a mobile network operator's network infrastructure. By offering a versatile solution to data generation, our framework contributes to overcoming challenges associated with crowdsourced data, opening up possibilities for advanced analytics and experimentation in mobile and IoT networks.

Index Terms—Generative Artificial Intelligence, Bayesian Learning, Minimization of Drive Test Data, Crowdsourcing, Mobile Networks, Internet of Things.

1 Introduction

The integration of intelligence and autonomous adaptivity has emerged as the driving force behind the development of future-generation mobile and internet of things (IoT) networks. This trend reflects the growing recognition of the importance of intelligent capabilities and self-adaptive behaviors in network infrastructures, paving the way for advanced network management functionalities. In accordance with this stance, the 3rd generation partnership project (3GPP) has introduced dedicated study items in Release 18, focusing on the utilization of artificial intelligence (AI)/machine learning (ML) for the management and design of network procedures within the radio access network (RAN) and service and system aspects (SA) technical specification groups (TSGs). ML-based methods have the flexibility to adapt to dynamic and evolving environments, as they can continuously learn from new data and update their models accordingly. On the other hand, they rely on data quality and availability, which, when insufficient or inaccurate, can lead to biased or unreliable results. Acquiring high-quality and representative data can pose challenges, particularly within domains characterized by limited data availability, expensive or time-consuming data collection processes, as well as privacy and security considerations.
To this end, generating synthetic data emerges as a promising method to overcome such limitations. Deep generative models have consequently emerged as one of the most exciting and prominent sub-fields of deep learning (DL), given their remarkable ability to synthesize input data of arbitrary form by learning the distribution such that novel samples can be drawn [1]. While generative ML witnessed its biggest success in fields such as computer vision [2] and natural language processing [3], extensive research interest recently sparked in the synthetic generation of data for mobile and sensor networks [1], [4], [5]. Within this context, data are naturally originated at the edge, and crowdsourcing emerges as a convenient and inherent approach to data collection. This approach leverages connected devices' widespread connectivity and sensing capabilities to create a collaborative framework for data collection. A non-comprehensive list of illustrative data types commonly obtained via crowdsourcing is presented in Table 1. On the other hand, crowdsourcing is affected by concerns related to privacy, statistical significance, sampling bias, and time-consuming data collection and post-processing.

In our study, we introduce GUMBLE (Generation of Uncertainty-aware Mobile data using Bayesian Learning), an innovative, comprehensive framework that is independent of applications and data types, allowing for the conditional generation of crowdsourced datasets with location information in mobile and IoT networks. A key element of our proposed methodology is the ability to assess uncertainty within newly generated samples. As elaborated upon in Section 4, this involves the decomposition of predictive uncertainty into aleatoric and epistemic components, which is achieved by leveraging approximate Bayesian methods. In order to ascertain the validity of our method, we present numerical results on the illustrative task of minimization of drive test (MDT) data generation. The MDT mechanism, introduced in 3GPP Release 10 and employing user equipments (UEs) to collect field measurements encompassing radio features and location data [6], [7], enables the gathering of statistical channel state information, thereby providing a realistic network state description and facilitating proactive forecasting, troubleshooting, and network optimization [8]. For the first time, we present an in-depth comparison of synthetically generated data with an original large-scale MDT set collected from a mobile network operator (MNO)'s network infrastructure. Further, we provide numerical results on downstream tasks using our generated dataset, demonstrating that comparable outcomes to those achieved

M. Skocaj, L.M. Amorosa, R. Verdone are with the Department of Electrical, Electronic and Information Engineering (DEI), "Guglielmo Marconi", University of Bologna & WiLab - National Wireless Communication Laboratory (CNIT), Italy. M. Lombardi is with the Department of Informatics, Science and Engineering (DISI), University of Bologna, Italy.
E-mail: {marco.skocaj, lorenzomario.amorosa, michele.lombardi2, roberto.verdone}@unibo.it

Source of information	Data	Location Information
Smart cities	Environmental data [9], [10], Urban planning [11]	✓
Industry 4.0	Industrial Sensor Data [12], [13]	✓
Smart homes	Sensor Data, Energy Management [14], [15]	✓ / X
Cellular Networks	Minimization of Drive Test data [16]–[18], User analytics [19]	✓ / X
Vehicular Networks (VANETs)	Traffic, Transportation and Environmental data [20], [21]	✓
Manual drive tests	key performance indicators (KPIs)	✓
Ad-hoc measurement campaigns	key performance indicators (KPIs) (e.g., LTE PDCCH decoding [22])	✓
Wearable devices	Health and Biometric Data [23], [24]	X

TABLE 1: A non-comprehensive list of common data types collected via crowdsourcing in mobile and IoT Networks.

with a large-scale dataset of original MDT measurements can be attained.

2 Contributions & Novelty

2.1 State of the art

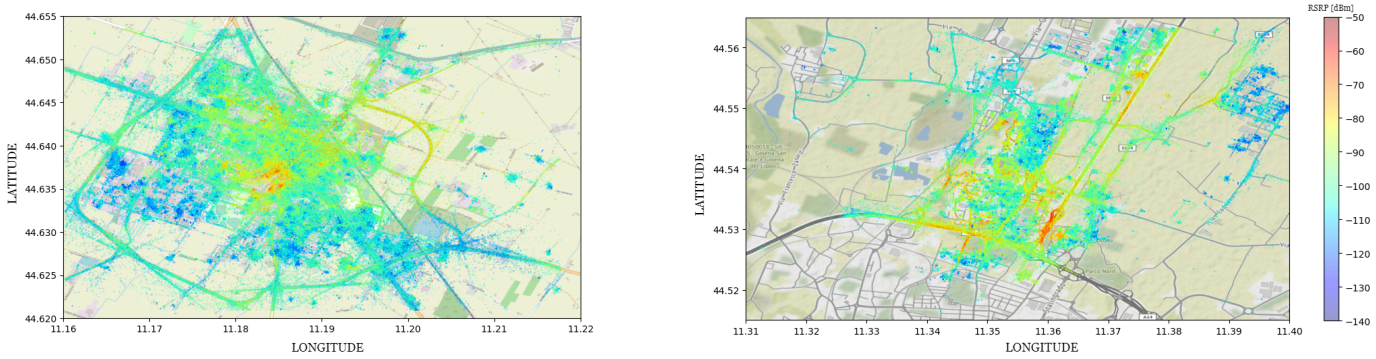
The application of generative artificial intelligence (Gen-AI) to communications and networking is an active and timely research topic that has found extensive practical applications. In this context, many works focus on the use of generative adversarial networks (GANs) [1], [4], [5] for the generation of synthetic datasets. Relevant works in this field feature [25], where the use of GANs is considered for augmenting a dataset comprising call data records (CDRs), a tabular data format containing information about the average start hour and duration of phone calls in a mobile network. The objective is to enhance the predictive accuracy of an autoregressive task by leveraging the expanded dataset generated through the GAN framework. The use of CDRs data is further exploited in [26], where Di Paolo et al. introduce a comprehensive framework for constructing an extensive dataset tailored for network planning purposes. The framework encompasses the modeling of distributions derived from a diverse collection of data, including CDRs, demographic information, and network deployment details obtained from MNOs. Within the IoT domain, Razghandi et al. [14], [15] focused on the synthetic generation of electrical load and solar panel energy production data in a smart grid network using a variational autoencoder-generative adversarial network (VAE-GAN). Although the aforementioned studies are prominent contributions to the field of Gen-AI in mobile and IoT systems, they address distinct problems and employ disparate data formats (e.g., CDRs, which lack UEs' positional information) compared to the focus of our current research.

On the contrary, closely associated with our work are the investigations conducted in [27], [28]. Sun et al. [27] present a deep generative framework able to produce synthetic time-series data associated to unseen trajectories during training time. To achieve the desired generalization capability, the presented framework leverages abstraction from network and environmental contextual information. This includes factors such as *cell site location, estimated transmit power, and cell orientation*, as well as surrounding environment information like *terrain, obstacles, clutter*, etc. Our work distinguishes from [27] in several ways. Firstly, our generated data format is distinct. While [27] focuses on generating time-series data based on a fixed trajectory input, our focus is set on jointly generating user samples in the time-spatial domain and performing conditional regression of their associated KPIs. Secondly, while [27] proposes a valuable solution for generalizing to unseen trajectories, its practicality is hindered by the requirement of costly (or even unavailable, e.g., private indoor buildings) contextual information, which needs to be collected and tailored for each specific application and scenario. In contrast, our work aims to provide an application-agnostic framework that does not rely on explicit location-dependent contextual information. Although we focus our numerical results on the task of MDT data generation, our proposed solution can be seamlessly applied to any type of geolocated data that includes direct location information, i.e., latitude and longitude, of measurements through a crowdsourcing mechanism (e.g., Table 1). Furthermore, our objective is not to generalize to unseen areas, but rather to produce uncertainty-aware predictions that can effectively support strategic planning for new measurements. Likewise, [28] proposes a solution for predicting signal quality metrics in long-term evolution (LTE) networks at unobserved locations. This approach utilizes raw GPS measurements, network contextual information (e.g., distance to transmitters), and satellite images. However, the reliance on geo-located contextual information still poses limitations on its applicability beyond the specific test region, and the inclusion of satellite images prevents the framework from being applied to indoor scenarios. Additionally, the mentioned work employs a point estimate of the considered metrics, and although it suggests the use of approximate Bayesian methods for uncertainty estimation, a comprehensive assessment of calibration capabilities and model misspecification analysis is not provided. These aspects [29]–[32] are of crucial importance and are thoroughly analyzed both analytically and experimentally in our study. Furthermore, our work addresses the conditional generation of metrics such as reference signal received quality (RSRQ), which is known to be dependent on network load. This aspect, which was not explicitly addressed in [27] and [28], is a key focus of our investigation. As a final remark, it is worth noting that both of the aforementioned works primarily focus on small-scale scenarios, and neither of them provides a comprehensive comparison with large-scale original crowdsourcing (e.g., MDT) datasets.

2.2 Motivation and Contributions

Crowdsourced datasets face various issues that drive the design objectives of our proposed framework:

- *Device Heterogeneity*: crowdsourced data quality varies due to device diversity, collection methods, and user behaviors, generally necessitating time-consuming post-processing and cleaning algorithms. This can result in data scarcity. To this end, augmenting post-processed data with high fidelity provides a notable advantage.



(a) Town of San Giovanni in Persiceto, Italy. Coverage of the area is provided by two tri-sectorial sites (6 eNBs).

(b) A peripheral area north of Bologna, Italy, where six tri-sectorial sites (18 eNBs) are deployed.

Fig. 1: Reference scenario - Map of geolocated reference datasets for distinct urban environments

- *Privacy and Security*: these are prominent concerns in crowdsourcing, as the data collection process is performed by individual users. To address this issue, the generation of synthetic data offers a viable solution.
- *Statistical insufficiency*: Incentivizing and motivating users in crowdsourcing measurements presents challenges, potentially leading to geographical areas characterized by statistical insufficiency. Hence, evaluating the interpolation capabilities of data generation methods and assessing the uncertainty in the generation process for low data density areas is crucial.
- *Sampling bias*: Crowdsourced datasets are prone to biases from conditioning characteristics in the environment. For instance, as discussed in Section 3, RSRQ is a radio KPI influenced by the network load. However, certain network load conditions are infrequent, and generating synthetic data as a function of the latter can eliminate the need for time-consuming measurement campaigns.

In order to address the challenges expressed above, our work features the following contributions:

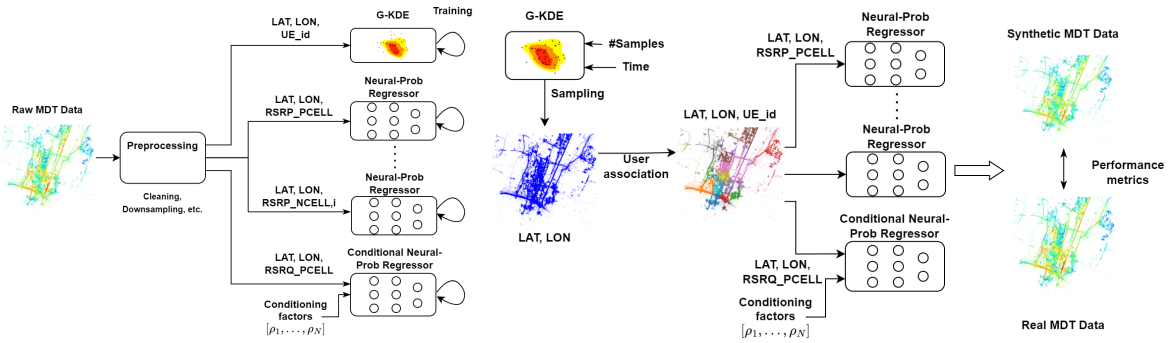
- We present a conditional generative framework designed to accurately produce synthetic crowdsourced datasets containing location information within mobile and IoT networks. The framework possesses two prominent features. Firstly, it enables the evaluation of uncertainty during the generation of new samples, which can be decomposed into epistemic and aleatoric confidence intervals. This is achieved by leveraging approximate Bayesian methods. Secondly, the framework allows for the generation of new samples based on conditioning factors pertaining to *non-location-specific* conditioning features of the environment.
- We ascertain the validity of our method on the illustrative task of MDT data generation. Leveraging on a variety of metrics, for the first time, we present an in-depth comparison of our generated data with an original large-scale set of MDT data collected from a MNO's network infrastructure. In particular, we focus on the algorithm's calibration, interpolation capability, and uncertainty assessment in the region of data extrapolation. While our experimental analysis is tailored to the use case of MDT data generation under tunable network and traffic conditions, the proposed methodology remains of general validity and well-suited for the generation of different kinds of crowdsourcing datasets with location information.
- To further demonstrate the validity of our framework, we provide numerical results on downstream tasks using our generated dataset, demonstrating that comparable outcomes to those achieved with a large-scale dataset of original MDT measurements can be attained. In particular, we focus our attention on the task of fingerprinting-based localization.

The remainder of the paper is organized as follows: in Section 3 we introduce our system model. In Section 4 we provide a technical overview of approximate Bayesian methods, which serves as a theoretical foundation for the rest of the paper. In Section 5 we delve into technical details about our proposed algorithms and the performance metrics we use to validate our framework. In Section 6 we present our numerical results, encompassing interpolation, extrapolation, conditional generation, and downstream task. Finally, we draw our conclusion in Section 7.

3 System Model

As mentioned in the previous section, we introduce our framework through the specific example of MDT data generation. To this end, we rely on MDT data collected from a MNO's network infrastructure, representing various scenarios of different scales. Specifically, we analyze data collected from distinct urban environments, as depicted in Fig. 1 above.
MDT data comprise a rich set of radio features. Here, we focus our attention on the generation of the following representative radio indicators:

- **RSRP of the serving cell**: The reference signal received power (RSRP) is defined as the narrow-band power measured by correlation of the LTE's channel reference signal (CRS). Therefore, it is not influenced by the co-channel interfering cells and average network load ρ, as the UE is able to perform interference cancellation and retrieve the useful signal. The RSRP can be formally expressed as the sum of the power carried by individual resource elements (REs) divided by the number N of subcarriers carrying CRS over the entire system bandwidth [33](1):



(a) Training architecture (b) Inference architecture

Fig. 2: System Model - Training vs Inference architecture. On the left side (a), the training phase involves the parallel training of distinct ML models on original MDT data. Each kind of model receives as an input relevant features, including user positions, measured KPIs, as well as network conditioning factors such as cell load ρ. On the right side (b), the generation of new synthetic samples involves a sequential pipeline that includes: (i) sampling of users in the time-space domain, (ii) user association to serving base stations, (iii) probabilistic (conditional) regression of radio features.

$$\text{RSRP} = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{14} P_{\text{RE},ik} \ , \tag{1}$$

where the sum in k takes into account the number of orthogonal frequency division multiplexing (OFDM) symbols for every time transmission interval (TTI) and $P_{\text{RE},ik}$ is defined as (2):

$$P_{\text{RE},ik} = \begin{cases} P_{\text{RE},i} & \text{if the } k-th \text{ OFDM symbol} \\ & \text{brings CRS} \\ 0 & \text{otherwise .} \end{cases} \tag{2}$$

The RSRP holds significant importance as a fundamental indicator utilized in various radio resource management (RRM) procedures, including mobility management and power control. Being a linear average of independent power samples, the RSRP is an estimator of the median power component perceived by the UE. For an analytical derivation of the lower error bound on the estimation error, we direct interested readers to consult Appendix A.

- **RSRP of neighbor cells**: Equivalently, MDT measurements collect information regarding the RSRP perceived from a set of neighboring cells within visibility range. Such measurements yield substantial information regarding interference patterns of the cellular network, aiding in network optimization, resource allocation, mobility management, and overall quality of service improvements.
- **RSRQ**: the RSRQ is inversely proportional to the received signal strength indication (RSSI), which is a wide-band measure of co-channel serving and non-serving cells (3):

$$\text{RSSI} = \sum_{i=1}^{M} \sum_{j=1}^{12} \sum_{k=1}^{14} P_{\text{RE},ijk} \ . \tag{3}$$

In (3), M indicates the number of physical resource blocks (PRBs) over the whole system bandwidth and the sum in j spans over all OFDM subcarriers in a PRB. The RSRQ can be formally expressed as the product between the number M of PRBs over the system bandwidth and the RSRP, divided by the RSSI (4).

$$\text{RSRQ} = \frac{M \cdot \text{RSRP}}{\text{RSSI}} \ . \tag{4}$$

The RSRQ yields important information and statistics related to the average usage of the network. As a result, the average load ρ is inversely correlated to the RSRQ.

- **User association**: Each MDT measurement report is associated to a serving base station. This association holds significant importance as it allows for the observation of traffic distribution and facilitates considerations for mobility load balancing operations. To this end, it is of utmost importance to associate each newly generated sample with a serving anchor/base station in order to ensure accurate analysis and decision-making.

It is noteworthy that our approach straightforwardly extends to any different set of features. Our selection constitutes a representative subset of KPIs that demonstrate either dependence (e.g., RSRQ) or independence (e.g., RSRP) on external factors (e.g., network load ρ) that are not tied to the sample's location and might be subject to sampling bias in the original dataset.

In our system model, we address the mobile data generation process through sub-problem decomposition: we approach the generation and clustering of space-time-dependent traffic samples independently from the probabilistic regression of their radio features. This allows us to implicitly capture inherent dependencies on geo-related conditioning factors, including radio environment characteristics, line-of-sight (LoS)/non-line-of-sight (NLoS) conditions, clutter, etc., during the training of the regression models. Fig. 2 at the top of the page illustrates our system architecture, highlighting the differences between the training and inference stages.

3.1 Training

The generation of space-time-dependent user samples and subsequent user association is formulated as a tri-variate density estimation problem on time, latitude, and longitude

components. It is important to note that user association holds significance within the scope of synthetic MDT data generation. However, this step may not be necessary for other types of datasets or for applications where the preservation of the serving anchor/base station is not a primary concern. Formally, we can express the density estimation as a maximum likelihood (negative log likelihood) problem:

$$\arg\min_{\theta} \quad \mathbb{E}_{x\sim P(x)} \left[-\log f(x|\theta)\right], \tag{5}$$

where the goal is minimizing the expected negative log-likelihood of sampled data, $x = (t, \text{lat}, \text{lon})$, $P(x)$ is the data distribution, and θ represents the model parametrization. As elaborated upon in Section 5, we aim to solve problem (5) via empirical risk minimization (ERM), so that the goal becomes that of minimizing the loss function:

$$\arg\min_{\theta} \quad \mathcal{L}(\theta) = \sum_{i=1}^{m} -\log f(x_i \mid \theta) \tag{6}$$

where $\mathcal{D} = \{x_i\}_{i=1}^{m}$ is the training sample. In practice, we also use a held-out validation set for calibrating certain model parameters.

As for the probabilistic regression of radio features, instead, we are interested in evaluating the uncertainty associated with the predictions. To achieve this, we adopt a Bayesian approach for the probabilistic regression of radio features. Specifically, we train our models via stochastic variational inference (SVI) by minimizing the variational free energy cost function (7):

$$\arg\min_{\lambda} \quad \text{KL}[Q_\lambda(\theta)\|P(\theta)] + \mathbb{E}_{\theta\sim Q_\lambda(\theta)} \left[\mathcal{L}(\theta)\right], \tag{7}$$

where $\mathcal{L}(\theta)$ indicates any loss function suitable to a regression problem computed on the training data, and the expectation is handled via a Monte-Carlo estimate. Further details on approximate Bayesian methods and on the specificity of the algorithms employed are elaborated in Section 4 and 5.

As depicted in Fig. 2, we employ independent learners to address each of the aforementioned problems. Upon preprocessing (e.g., cleaning and downsampling), an original set of MDT data is fed as input to the blocks depicted in Fig. 2. Additionally, external conditioning factors such as network load ρ are fed as residual input to a set of regressors whose scope is to perform conditional regression (e.g, RSRQ), as elaborated in Section 5.3.

3.2 Inference

After training, the inference phase foresees a sequential pipeline involving: (i) sampling of generated users in the time-space domain, (ii) user association, and (iii), probabilistic (conditional) regression of radio features. A synthetic MDT sample is therefore composed of the following artificial features:

- time_window_reference
- sample_latitude
- sample_longitude
- sample_serving_cell_ID
- sample_primary_RSRP
- sample_neighbour_RSRP_1,. . . ,N
- sample_neighbour_RSRQ.

4 Approximate Bayesian Methods

Conventional frequentist learning aims at identifying an optimal point estimate $\hat{\theta}$ of the parameters of a statistical model. The estimation of $\hat{\theta}$ is pursued under the assumption of ERM that the loss $\mathcal{L}(\theta)$, computed on the available training set, is representative of the true population loss. The estimation of $\hat{\theta}$ can be achieved via minimum mean squared error (MMSE), maximum likelihood estimation (MLE), or, if regularization is introduced, maximum a posteriori (MAP) criterion; in all cases, stochastic gradient descent (SGD) is typically used as the optimization approach. Nevertheless, the discrepancy between the population loss and the training loss is a function of the data set size and it introduces uncertainty regarding the optimal parametrization. This intrinsic model uncertainty is formally referred to as the *epistemic uncertainty* ε_{ep} and it can be expressed as:

$$\varepsilon_{ep} = |\mathcal{L}(\theta^\star) - \mathcal{L}(\hat{\theta})|, \tag{8}$$

where $\mathcal{L}(\theta^\star)$ refers to the loss corresponding to the optimal parametrization $\theta^\star$ and $\mathcal{L}(\hat{\theta})$ refers to the loss corresponding to the parametrization $\hat{\theta}$ pursued under the ERM assumption. Typically, ε_{ep} denotes a reducible form of estimation error, which can be mitigated by increasing the size of the training set. This is in contrast to a second form of uncertainty known as aleatoric uncertainty, namely ε_{al}, which is inherent in the data and cannot be reduced through additional training samples. By selecting a single model, frequentist learning neglects epistemic uncertainty as it discards information about other plausible models that fit training data almost as well as the ERM solution [30]. In turn, this translates to a lack of explainability, poor calibration capability, overfitting, and over-confident predictions in the extrapolation regime. Conversely, the Bayesian approach takes into account the explanations offered by a distribution $\theta \sim P(\theta)$ over the model parameters. In Bayesian neural networks (BNNs), this enables considering a probability distribution on the weights of a neural network in place of scalar values [34]. Each weight is assigned a distinct probability according to data dependent (i.e. posterior) distribution $P(\theta \mid \mathcal{D})$, where $\mathcal{D} = \{x_i, y_i\}_{i=1}^{m}$ refers to the training set. In a Bayesian model, the predictive posterior distribution $P(y \mid x, \mathcal{D})$, i.e. the distribution of the predictions that can be made based on the available data, is computed by marginalizing over θ, i.e. by computing an expectation over θ. Formally, this can be expressed as per (9):

$$P(y \mid x, \mathcal{D}) = \int_{\theta} P(y \mid x, \theta) \cdot P(\theta \mid \mathcal{D})\, d\theta, \tag{9}$$

where $P(y|x, \theta)$ expresses the likelihood given the parameterization θ, which can be computed using θ as weight vector for a model. A closed-form derivation for the expectation is usually intractable, therefore (9) is typically approximated at inference time by means of Monte-Carlo sampling:

$$P(y \mid x, \mathcal{D}) \approx \frac{1}{T} \sum_{i=1}^{T} P(y \mid x, \theta_i), \tag{10}$$

where T is the number of samples used for the approximation and θ_i is the i-th sampled weight vector. In contrast to Equation (9), relying on sampling eliminates the need for explicit weighting. By sampling from the posterior distribution, BNNs generate multiple plausible models, each producing slightly different predictions.

According to (9) and (10), the training procedure of BNNs revolves around the estimation of the true posterior $P(\theta \mid \mathcal{D})$. However, this calculation frequently proves to be computationally infeasible. To this end, various approximate methods (e.g., based on Monte-Carlo dropout [35] or variational inference (VI) [34]) have been proposed to find a tractable approximation to the true posterior distribution. VI involves approximating the true distribution $P(\theta \mid \mathcal{D})$ via a variational distribution $Q_\lambda(\theta)$ with parameter vector λ, then optimizing it to match the true posterior distribution, i.e. minimizing the Kullbach-Leibler distance $\mathrm{KL}[Q_\lambda(\theta) \,\|\, P(\theta \mid \mathcal{D})]$. By algebraic manipulation [36], this leads to the *variational free energy* cost function (7), which embodies a tradeoff between minimizing a given loss function $\mathcal{L}(\theta)$ (e.g., negative log-likelihood over the training data) and minimizing model complexity with respect to a prior $P(\theta)$. Blundell et al. [34] introduce an approximation of the closed-form variational free energy (7) by means of Monte-Carlo sampling during training time, in order to obtain a tractable objective function (11):

$$\begin{aligned}\arg\min_{\lambda}\ & KL[Q_\lambda(\theta) \,\|\, P(\theta)] + \mathbb{E}_{\theta\sim Q_\lambda(\theta)}[\mathcal{L}(\theta)] = \\ &= \int_\theta Q_\lambda(\theta) \log \frac{Q_\lambda(\theta)}{P(\theta)}\, d\theta + \int_\theta Q_\lambda(\theta)\mathcal{L}(\theta)\, d\theta \\ &\approx \sum_{i=1}^{T} \log Q_\lambda(\theta_i) - \log P(\theta_i) + \mathcal{L}(\theta_i)\ .\end{aligned} \tag{11}$$

Equation (11) presents a computationally feasible optimization problem that enables us to effectively approach the evidence lower bound (ELBO), a measure of the quality of approximation for estimating the true distribution. By minimizing the objective in Equation (11), we achieve a balance between minimizing the expected training loss $\mathcal{L}(\theta)$ under the variational distribution and minimizing the discrepancy between the variational distribution and a prior distribution, which acts as a regularization term. This approach establishes a principled framework to accurately quantify and integrate epistemic uncertainty into the output of an ensemble of predictors, playing a pivotal role in evaluating the prediction reliability of newly generated samples. As further elucidated within the next sections, leveraging uncertainty in predictions can drive the strategic planning of further measurements for augmenting crowdsourcing datasets in an effective manner.

5 Algorithms

This section is devoted to the description of the theoretical details, design principles, and training process of the data generation framework building blocks.

5.1 Samples Generation and User Association via Gaussian kernel density estimation (G-KDE)

Data generation for tabular datasets is addressable through a variety of approaches, and many of them are implemented as ready-to-use libraries [37]. State of the art methods comprise GANs, normalizing flows (NFs) [38], variational autoencoders (VAEs) [39], or Gaussian mixture models (GMMs). However, the MDT data spatial distribution poses concrete challenges to the models previously listed, since it is significantly complex, irregular, and it strongly depends on the topography of the area under investigation. To this end, we aim to tackle the generation of user samples through a density estimation approach, as previously anticipated in Section 3.1. Specifically, we leverage a non-parametric density estimation technique known as kernel density estimation (KDE). Our choice of KDE stems from its simplicity, interpretability, and suitability to the specific scenario of modeling UEs distribution. In Appendix B, we further justify our proposed approach by comparing numerical results obtained through the multivariate Kolmogorov-Smirnov (KS) test with respect to a set of parametric baselines, including GANs, NFs, and GMMs. Given a sequence of samples $\mathcal{D} = \{x_i\}_{i=1}^m$, with $x_i = \{\mathrm{lat}_i, \mathrm{lon}_i\}$, from a distribution $P(x)$, KDE aims to estimate the true density function by treating each training sample as the center for a density kernel $K(x, h)$. The training process involves an affine transformation of each kernel $K(x - x_i, h)$ around each data point. This yields an interpretable approach, resulting in a PDF estimate, noted as $f(x, \mathcal{D}, h) : \mathbb{R}^{2\times n} \to \mathbb{R}^+$, that stems directly from the observed data, taking advantage of all sample points' locations and convincingly suggesting multimodality [40]. Formally, we can express $f(x, \mathcal{D}, h)$ as

$$f(x, \mathcal{D}, h) = \frac{1}{n}\sum_{i=1}^{n} K(x - \hat{x}_i, h)\ , \tag{12}$$

where

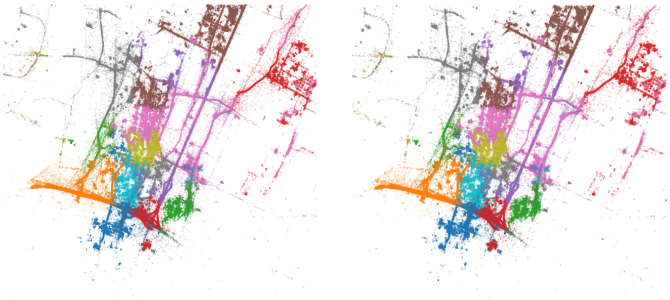
$$K(x - x_i, h) = \frac{1}{h\sqrt{2\pi}} \exp\left(-\frac{(x - x_i)^2}{h^2}\right) \tag{13}$$

is a Gaussian kernel function, x is the point in which the probability density function (pdf) is estimated, and the model parameters coincide with the training samples $\mathcal{D}$. The value h is a smoothing hyper-parameter named *bandwidth*. Our motivation behind the choice of a Gaussian kernel is grounded in domain knowledge. The location information in MDT data is captured through the use of GPS information available at the UE side. The estimation of the true position of the UE via GPS is commonly assumed to adhere to a normal distribution. Hence, considering Gaussian kernels with tunable bandwidth facilitates the incorporation of a normal a priori distribution regarding the uncertainty over the true user position. Intuitively, KDE is based on the assumption that the greater the density of samples in a specific area, the greater the probability that other samples will be located around that point. KDE is an unsupervised learning approach, and the hyper parameter h is tuned via ERM on a validation set, i.e. by minimizing the negative log-likelihood (5) on a downsampled set $\mathcal{D}^{ds} = \{x_i^{ds}\}_{i=1}^{m_{ds}}$, with $m_{ds} < m$ as per (14):

$$\arg\min_{h} -\sum_{i=1}^{m_{ds}} \log f(x_i^{val}, \mathcal{D}, h)\ , \tag{14}$$

where $\mathcal{D}^{val} = \{x_i^{val}\}_{i=1}^{m-m_{ds}} = \mathcal{D} \setminus \mathcal{D}^{ds}$.

Lastly, the user distribution over a given area shows dependency on the time domain. Within sufficiently small time intervals, $f(x, \mathcal{D}, h)$ can be assumed to be time-invariant. We refer to N as the necessary number of time windows to observe independent, time-invariant user probability distributions. As a consequence, the user distribution is modeled as a joint distribution, whose spatial components are continuous, and the time component is discrete over N values. We denote the time intervals as $j \in \{1, \ldots, N\}$, and we further distinguish among samples in the space domain according to their associated primary cell (PCELL) among the set



(a) Ground-Truth: each MDT sample is associated to a serving PCELL, which is represented with a different colour.

(b) Synthetic MDT samples after sampling from G-KDE and user association procedure described in Algorithm 1.

Fig. 3: Illustrative results of MDT samples generation and user association.

$\mathcal{O} = \{C_1, \ldots, C_K\}$. The selection of a time interval j and a primary cell C_i collectively defines the set $\mathcal{D}_{i,j}$ used for the training of a specific instance of a KDE model. In our framework, $K \times N$ distinct KDEs are thus trained distinctively according to their relative training samples in the time-space domain. Additionally, one KDE model is trained using the samples from the union of all cells $\mathcal{O}$. Accordingly, the resulting probability density function for each KDE model is denoted as $f_j^{C_i}$ ($f_j^{\mathcal{O}}$). This is crucial to ensure the correct association of newly generated samples $(x'_{\text{LAT}},\ x'_{\text{LON}})$ to a serving PCELL. The user association procedure consists in associating each sample $(x'_{\text{LAT}},\ x'_{\text{LON}}) \sim f_j^{\mathcal{O}}$, to the PCELL′ relative to the distribution $f_j^{C_i}$ holding maximum likelihood (15):

$$\text{PCELL}' = \arg\max_i \left\{ \log f_j^{C_i}((x'_{\text{LAT}},\ x'_{\text{LON}}) \sim f_j^{\mathcal{O}}(\hat{\mathbf{x}}_\mathbf{i}, h) \right\}, \tag{15}$$

where $\hat{\mathbf{x}}_\mathbf{i}$ is the training set of original MDT samples belonging to C_i. The complete generation process of new samples $(x'_{\text{LAT}},\ x'_{\text{LON}},\ \text{PCELL}')$ is described in Algorithm 1. Furthermore, an illustrative result of the proposed method 1 is portrayed in Fig. 3, shown above.

Algorithm 1 G-KDE samples generation

Require:
$j \in \{1, .., N\}$, the selected time window
$\mathcal{S}$ total set of samples to be generated
for $s \in \mathcal{S}$ **do**
  Initialize $(x'_{\text{LAT}},\ x'_{\text{LON}})_s \sim f_j^{\mathcal{O}}$
  for each $i \in \{1, ..., K\}$ **do**
    $\mathcal{L}(i) = \log\left(f_j^{C_i}((x'_{\text{LAT}},\ x'_{\text{LON}})_s | \hat{\mathbf{x}}_\mathbf{ij}, h\right)$
  end for
  $\text{PCELL}'_s = \arg\max_i\{\mathcal{L}(i)\}$
  $(x'_{\text{LAT}},\ x'_{\text{LON}},\ \text{PCELL}')_s \leftarrow \text{PCELL}'_s$
end for
return $(\mathbf{x'_{LAT}},\ \mathbf{x'_{LON}},\ \mathbf{PCELL'})$

5.2 RSRP: Bayesian Neural-Probabilistic Regression

To tackle feature regression, we employ a Bayesian neural-probabilistic regressor. The latter can be defined as a computational model that combines neural networks with VI (section 4) to perform a regression task. The final layer of the neural network is implemented as a parametrized probability distribution $P(y \mid x, \theta)$, and (as already discussed) the model is trained by minimization of the variational free energy cost function:

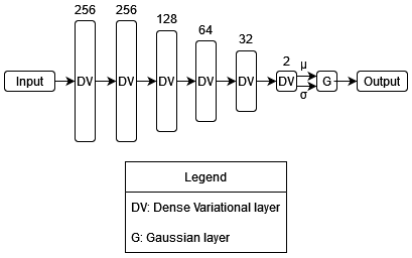


Fig. 4: Bayesian neural-probabilistic model architecture. Dense variational layers involve modeling distributions over model weights, with the final layer implementing a Gaussian distribution parameterized by the preceding layer.

$$KL[Q_\lambda(\theta) \,||\, P(\theta)] - \mathbb{E}_{\theta \sim Q_\lambda, y \sim P(y|x)}[\log P(y \mid x, \theta)]\ , \tag{16}$$

where the right-hand term refers to the negative log-likelihood cost and the expectation over y is addressed via ERM. Dealing with the expectation over θ and with the first loss term entails establishing a prior distribution for the model weights and specifying a parametric assumption for the final distribution. In our experiments, we adopt the common practice of utilizing an isotropic Gaussian prior with a symmetric covariance matrix $\Sigma_{\bar{\theta}} = \sigma^2 I$, where each component on the main diagonal has zero mean and unitary variance. Conversely, careful consideration should be placed upon selecting the parametric assumption for the final distribution. If this assumption does not align with the true distribution of the target value, it may lead to model misspecification, thereby potentially undermining the model's calibration capability. In the context of RSRP regression, we model the last layer as a Gaussian distribution. For an analytical derivation of this result, we direct the interested reader to consult Appendix A. Additionally, in Section 6, this assumption is confirmed by numerical results showing that the proposed model is calibrated out-of-the-box and thus does not suffer from model misspecification.
A compact representation of our neural-probabilistic model, depicted in Fig. 4 above, can be formulated as per (17):

$$\hat{y} \sim \mathcal{N}\left(\mu(x, Q_\lambda(\theta)), \sigma(x, Q_\lambda(\theta))\right) \tag{17}$$

In (17) a predicted sample $\hat{y}$ is drawn from a Normal distribution, characterized by the parameters μ and σ, which, in turn, depend on the variational distribution $Q_\lambda(\theta)$ and the input x. This formulation enables the generation of probabilistic outputs that simultaneously account for both aleatoric and epistemic uncertainties. However, it may be convenient for various reasons to individually decompose the two uncertainty components. When confronted with an ensemble of probabilistic regression models, namely $\{P(y \mid x, \theta_i \sim Q_\lambda(\theta)\}_{i=1}^M$, a feasible and convenient strategy to accomplish uncertainty decomposition involves employing the law of total variance [41]:

$$\underbrace{\mathbb{V}_{P(y|x,\mathcal{D})}(y)}_{\text{total uncertainty } \varepsilon} = \underbrace{\mathbb{V}_{P(\theta|\mathcal{D})}(\mathbb{E}_{P(y|x,\theta)}[y])}_{\text{epistemic uncertainty } \varepsilon_{ep}} + \underbrace{\mathbb{E}_{P(\theta|\mathcal{D})}[\mathbb{V}_{P(y|x,\theta)}(y)]}_{\text{aleatoric uncertainty } \varepsilon_{al}}\ . \tag{18}$$

Considering the Gaussian parametrization for the neural probabilistic regressor (17), equation (18) can be approximated by Monte Carlo sampling:

$$\mathbb{V}_{P(y|x,\mathcal{D})}(y) \approx \frac{1}{M}\sum_{i=1}^{M}\left[\mu_M - \mu_i\right]^2 + \frac{1}{M}\sum_{i=1}^{M}\sigma_i^2 \,. \quad (19)$$

In (19), μ_i and σ_i refer to the mean and variance of the i-th probabilistic regressor sampled from the Bayesian ensemble, and $\mu_M := \frac{1}{M}\sum_{i=1}^{M}\mu_i$. Equation (19) offers a convenient means to disentangle the epistemic uncertainty from the intrinsic noise in the data and it can be used to assess the trustfulness of model prediction in the extrapolation regime, as shown in section 6.

A further important design choice of our proposed model concerns the feature space selection. The chosen input features consist of the device position – latitude (LAT), longitude (LON). Despite the apparent simplicity of this selection, it is imperative to pursue implicit modeling of geolocation-dependent conditions that exert a specific influence on RSRP. As a result, the probabilistic model effectively captures the RSRP's reliance on factors such as the propagation environment, LoS/NLoS conditions, clutter, surrounding building materials, and other position-related dependencies, without the need for collecting and processing additional contextual information of the environment. In contrast, RSRQ is also affected by conditioning factors that are not strictly dependent on geolocation, such as the average cell load ρ. These factors need to be explicitly provided as inputs for conditional regression, as discussed in the subsequent subsection.

5.3 RSRQ: Conditional Bayesian Neural-Probabilistic Regression

In ML, a recurrent problem pertains to dataset imbalance, which leads to discrimination against underrepresented classes [42]. When undertaking probabilistic regression of KPI features in a crowdsourcing environment, this crucial aspect is exacerbated by sampling bias and must be properly accounted for. In particular, this challenge becomes especially pronounced for KPI features dependent on non-location-specific conditioning factors, such as RSRQ. Since very high or low traffic conditions are noticeably rare, RSRQ samples collected under these conditions are underrepresented. To address this issue, we refer to inverse probability weighting (IPW). IPW, when correctly applied, can potentially improve the efficiency and reduce the bias of unweighted estimators. Technically, IPW introduces a per-sample weighted cost function, where the weight α_i is proportional to the inverse of the probability of observing that sample within the training set. Considering as an example the log-likelihood cost function, the modified cost function using IPW becomes:

$$\mathcal{L}(\theta) = \log\prod_{i=1}^{|\mathcal{D}|}\alpha_i P(y_i|x_i,\theta) = \sum_{i=1}^{|\mathcal{D}|}\log\alpha_i P(y_i|x_i,\theta)\,, \quad (20)$$

where:

$$\alpha_i \propto \frac{1}{P(x_i \mid \mathcal{D})}\,. \quad (21)$$

where $P(x_i \mid \mathcal{D})$ is the probability of value x_i, estimated based on the training data $\mathcal{D}$. We approximate this term via a histogram-like appraoch, by first discretizing the x values, and then computing categorical probabilities. The discretization of the load space into bins ρ_j is a design choice. Thanks to the use of IPW, the objective function is biased toward less frequent samples, improving fairness and solving the problem of skewed data distribution.

5.4 Performance Metrics

Here, we introduce the metrics employed for evaluating the performance of the proposed generative algorithmic framework. Our specific focus lies on the task of probabilistic regression, as well as on the downstream task of fingerprinting-based localization, which is carried out using our synthetic dataset.

5.4.1 Probabilistic regression

- **MAE** and **RMSE**: Straightforward metrics to measure the effectiveness of the neural-probabilistic regressor are the root mean squared error (RMSE) (22) or the mean absolute error (MAE) (23):

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{D}_{\text{test}}|}\sum_{i\in\mathcal{D}_{\text{test}}}(y_i-\hat{y}_i)^2}\,, \quad (22)$$

$$\text{MAE} = \frac{1}{|\mathcal{D}_{\text{test}}|}\sum_{i\in\mathcal{D}_{\text{test}}}|y_i-\hat{y}_i|\,, \quad (23)$$

  where $\hat{y}_i$ is the estimated value for the i-th sample. However, both metrics alone are not sufficient as they fail to capture the probabilistic modeling aspect.
- **Calibration**: Inspired from [29], we assess the quality of probabilistic regression in terms of calibration plots and calibration error. The former displays the true frequency of points in each confidence interval relative to the predicted fraction of points in that interval, and is computed as per:

$$\tilde{P}_j = \frac{|\{y_i|F_i(y_i)\le P_j,\ i\in\mathcal{D}_{test}\}|}{|\mathcal{D}_{test}|}\,, \quad (24)$$

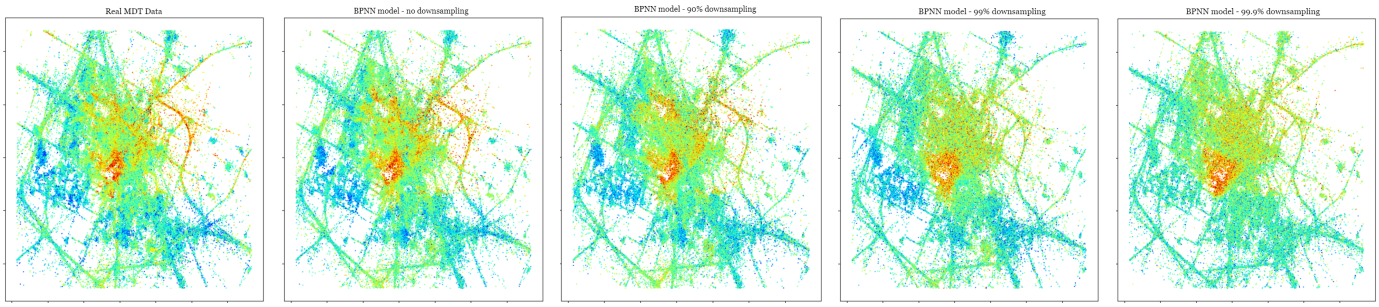
  where P_j refers to the true frequency of points for any given quantile $j \in \{0,\dots,1\}$, $\tilde{P}_j$ is the empirical frequency for that same quantile, and $F_i(y_i)$ is the cumulative distribution relative to the probabilistic output, given input x_i. In the case of a Gaussian parametrized output, as per (17), the latter can be computed as:

$$\begin{aligned} F_i(y_i) &= P\left(\mathcal{N}\left(\mu(x_i,Q_\lambda(\theta)),\sigma(x_i,Q_\lambda(\theta))\right)\le y_i\right) = \\ &= \frac{1}{\sigma_i\sqrt{2\pi}}\int_{-\infty}^{y_i}\exp\frac{(x-\mu_i)^2}{\sigma_i^2}\,dx. \end{aligned} \quad (25)$$

  Finally, from the knowledge of $\tilde{P}_j$ and P_j, we can compute a numerical score, denoted as calibration error (CE), describing the model calibration capability:

$$\text{CE} = \sum_j w_j|\tilde{P}_j - P_j|\,, \quad (26)$$

  where $w_j \propto |\{y_i|F_i(y_i)\le P_j,\ i\in\mathcal{D}_{test}\}|$, in order to reduce the importance of quantiles counting fewer examples.
- **Sharpness**: Consistent with the methodology suggested in [29] we assess the model sharpness, i.e., its ability to produce probabilistic outputs with tight



(a) Ground-Truth (b) Original training set (c) 10% training set size (d) 1% training set size (e) 0.1% training set size

Fig. 5: Visual comparison of ground-truth values and RSRP predictions obtained from a Bayesian neural-probabilistic model trained on independent instances of the training set, demonstrating the impact of increasing downsampling.

Metric	Full training set	Downsampling 90%	Downsampling 99%	Downsampling 99.9%
MAE [dB]	5.42	5.92	5.93	6.64
CE	2.09e-2	2.27e-2	2.67e-2	4.98e-2
S ($\mathbb{E}[S]$ [dB], $\sigma[S]$ [dB])	(7.12, 1.55)	(7.39, 1.43)	(8.23, 1.25)	(8.5, 0.95)

TABLE 2: Numerical comparison of ground-truth values and RSRP predictions obtained from a Bayesian neural-probabilistic model trained on independent instances of the training set, demonstrating the impact of increasing downsampling.

bounds, by means of the predictive standard deviation $\sigma(x_i, Q_\lambda(\theta))$, which accounts for both ε_{ep} and ε_{al}:

$$S = \sigma(x^{\text{test}}, Q_\lambda(\theta)) \,, \tag{27}$$

where x^{test} represents the vector of x values for the test data, and S is a vector of dimension $\mathbb{R}^{1\times m^{\text{test}}}$, where $m^{\text{test}} = |x^{test}|$ is the cardinality of the test set. In order to derive compact scalar metrics from (27), we can refer to the average sharpness over the test set, $\mathbb{E}[S]$, and the sharpness standard devation $\sigma[S]$:

$$\mathbb{E}[S] = \frac{1}{m^{\text{test}}} \sum_{i=1}^{m^{\text{test}}} \sigma(x_i, Q_\lambda(\theta)) \,, \tag{28}$$

$$\sigma[S] = \sqrt{\frac{1}{m^{\text{test}}} \sum_{i=1}^{m^{\text{test}}} (\sigma(x_i, Q_\lambda(\theta)) - \mathbb{E}\left[\sigma(x_i, Q_\lambda(\theta))\right])^2} \,. \tag{29}$$

Under equivalent calibration conditions, a model demonstrating narrower sharpness yields more informative predictions.

- **Average epistemic uncertainty** ε_{ep}: As a last metric, we focus on the epistemic uncertainty, which can be extracted from (17), following (18). In fact, a perfectly calibrated model cannot achieve narrower confidence bounds compared to a model with $\varepsilon_{ep} = 0$, as ε_{al} represents an irreducible source of uncertainty. We can thus evaluate the average epistemic uncertainty over a set of L positions $X_A = \{x_1, \ldots, x_L\}$, over a given area A:

$$\begin{aligned} \mathbb{E}_A[\varepsilon_{ep}] &= \mathbb{E}_A\left[\mathbb{V}_{P(\theta|\mathcal{D})}\left(\mathbb{E}_{p(y|x,\theta)}[y]\right)\right] = \\ &= \frac{1}{L\,M} \sum_{j=1}^{L} \sum_{i=1}^{M} \left[\mu_{M,j} - \mu_{i,j}\right]^2 \,, \end{aligned} \tag{30}$$

where we refer to $\mu(x_{i,j}, Q_\lambda(\theta))$ as $\mu_{i,j}$ for simplicity, $\mu_{M,j} := \frac{1}{M}\sum_{i=1}^{M} \mu_{i,j}$, and M refers to the number of Monte Carlo experiments performed for every x_j.

This analysis, as shown in the next section, further proves insightful in assessing the model's capacity to express uncertainty in the extrapolation regime and to establish a threshold for distinguishing between "reliable" and "unreliable" predictions.

5.4.2 *MDT-based fingerprinting*

In addition to evaluating probabilistic regression, we further assess the quality of synthetic generation by conducting a downstream task on the synthetic dataset. To gauge the quality of the results, we compare the outcomes of the same task performed by the identical models trained on the original dataset. The task at hand yields a point estimate of the ground truth variable y, which corresponds to the ground-truth position (lat, lon). Accordingly, we can assess performance by comparing the RMSE (22) obtained on the original dataset vs the synthetic one.

6 Empirical Results

This section provides a comprehensive quantitative assessment derived from the experimental evaluations conducted in this study. It elucidates the results pertaining to the aforementioned key metrics and presents a detailed comparison between our generated data and a large-scale original dataset of MDT data acquired from a MNO network infrastructure. To ensure a comprehensive evaluation of each model characteristic, the section is further divided into the following subsections: interpolation, extrapolation, conditional generation, and downstream tasks.

6.1 Interpolation

In this subsection, we aim to evaluate the interpolation performance of the proposed Bayesian neural probabilistic regressor. To this end, we conduct an evaluation using a 65/35 train-test split of our original dataset on the reference urban scenario in Fig. 1a. The considered performance metrics (MAE, Calibration, and Sharpness, Tab. 2) are examined as a function of the decreasing training set

size. We perform multiple independent training instances of the same neural-probabilistic architecture with varying degrees of downsampling applied to the training set while testing the performance on a held-out test set comprising the original locations and RSRP of the samples. The outcomes of these evaluations are presented in Fig. 5 and Table 2, shown above. Fig. 5 depicts a noticeable decline in predictive performance as a function of the increasing downsampling rate of the initial training set (consisting of 1 million examples). Nevertheless, Table 2 reveals that, despite the anticipated performance degradation, the model exhibits remarkable interpolation capability. Specifically, when employing a downsampling rate of 99.9%, the MAE decreases only by 1.22 [dB] compared to the original training set.
Below, Fig. 6 portrays a calibration analysis as a function of the downsampling ratio. As is evident from the figure and

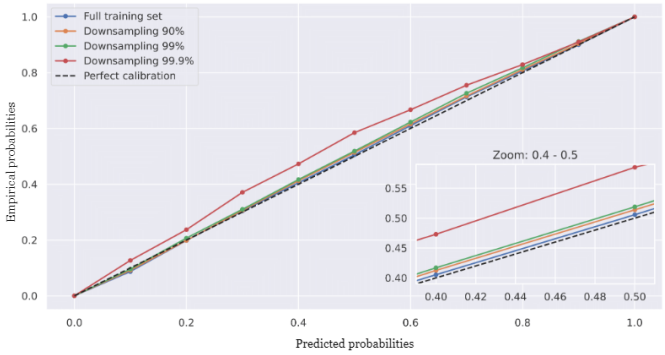


Fig. 6: Calibration plot - BPNNs are calibrated out-of-the-box, which means Gaussian assumption over RSRP distribution is empirically confirmed (no model misspecification).

supported by the CE and sharpness indicators presented in Table 2, the model demonstrates a tendency towards underconfidence with increasing downsampling rates. This outcome aligns with the desired objective of adopting a cautious approach when uncertain, - i.e., in the presence of fewer data samples. Both findings indicate that the proposed model possesses two key characteristics. Firstly, it demonstrates resilience to substantial downsampling, making it effective even when dealing with sparse datasets. Secondly, it exhibits a cautious approach when confronted with limited data samples, thereby showcasing its inherent ability to provide reliable uncertainty-aware predictions. Finally, Fig. 6 reveals another significant finding: the neural-probabilistic model exhibits out-of-the-box calibration, thus empirically validating our analytical assumption of a Gaussian parametrization (refer to Appendix A) over the RSRP distribution.

6.2 Extrapolation

Besides showing a conservative predictive approach when confronted with increasingly sparse data points, an essential characteristic of a probabilistic model is its proficiency in conveying model uncertainty in extrapolation regions. To evaluate this capability, we employ a neural probabilistic model trained on the urban scenario in Fig. 1a and conduct inference on two distinct areas. The first area corresponds to a region abundant in training data, while the second one represents a sparsely sampled region, as illustrated in Fig. 7. Results show an evident tendency of the model to produce high epistemic confidence intervals when inference is performed on extrapolation areas. Specifically, the average epistemic uncertainty measured over the non-extrapolation and extrapolation areas, namely $\mathbb{E}_{A1}[\varepsilon_{ep}]$ and $\mathbb{E}_{A2}[\varepsilon_{ep}]$, measure $0,87$ [dB] and $21,9$ [dB], respectively. This distinction is also evident from the distribution of epistemic uncertainty observed in non-extrapolation areas (depicted by the orange histogram plot in Figure 7) compared to that of extrapolation areas (represented by the blue histogram plot in Figure 7). This finding holds significant implications: by establishing a threshold on epistemic uncertainty, we can categorize each new data point as either "reliable" or "unreliable." Such categorization enables the strategic planning of new measurements in crowdsourced settings and provides a measure of trustworthiness for the proposed data-driven generative approach. This framework aligns with the principles of active learning [43], where the labeling of new data points is associated with a cost, such as the expense of conducting a drive test or the communication cost in a crowdsourcing environment.

6.3 Conditional Generation

A promising feature of the proposed generative framework, as elaborated upon in Section 5.3, is the ability to generate probabilistic outputs that rely on conditioning factors independent of sample locations and occur with varying probabilities. This characteristic aims to address the issue of sampling bias by compensating for its effects. Here, we assess the capability of the proposed neural probabilistic regressor, conditioned on the average cell load, to generate varying RSRQ values. Our analysis concentrates on assessing the MAE within the non-extrapolation regime. Below, Fig. 9 presents the registered MAE as a function of the quantiles of the average cell load within the training set, which may vary between individual cells. The utilization of IPW is evident in enabling the neural probabilistic regressor to achieve fairness and provide compensation for the lowest populated quantiles, specifically the 10% and 90% quantiles.

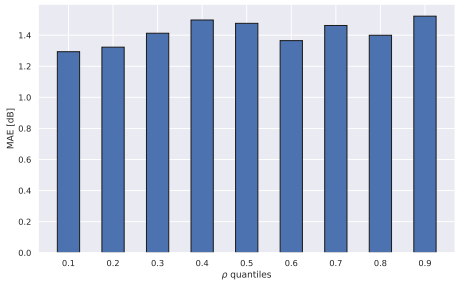


Fig. 9: MAE on RSRQ for non-extrapolation regime.

In addition, Fig. 8 provides a visual means of evaluating the effect of conditioning by ρ on the predicted RSRQ, which shows a tendency to lower predicted values when conditioned by higher loads.

6.4 Downstream Task

To further demonstrate the quality of the generated data, we present numerical results comparing the performance on the downstream task of fingerprinting-based localization, conducted on both the synthetic and the original data.

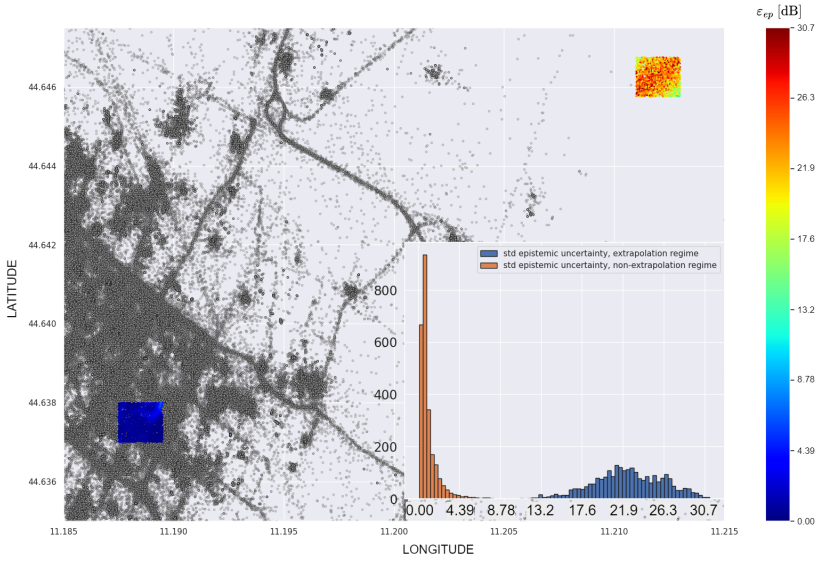
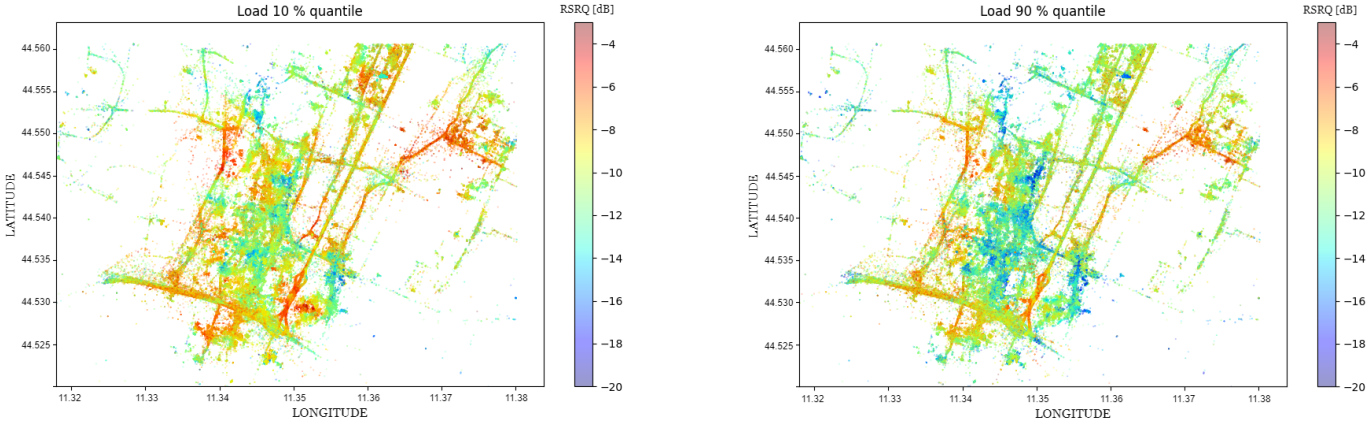


Fig. 7: Evaluation of epistemic uncertainty distribution in extrapolation vs non-extrapolation regime



(a) Low ρ (10 % quantile)

(b) High ρ (90 % quantile)

Fig. 8: Conditional probabilistic regression of RSRQ according to different configurations of ρ.

6.4.1 *Fingerprinting-based localization*

ML-based fingerprinting [44] is a technique that utilizes ML algorithms to determine the location of an UE by analyzing radio frequency (RF) signals. It involves two phases: during the *offline* phase a ML algorithm is trained based on a database of RF fingerprints, which are unique representations of signal characteristics observed at various known locations. During the *online* phase, inference is performed based on newly observed fingerprints. Here, we consider the task of ML-based fingerprinting localization using datasets comprising original MDT fingerprints and synthetically generated ones. The generation of synthetic fingerprints involves the generation of new samples in the space-time domain via G-KDE and the subsequent probabilistic regression of their features ($\text{RSRP}_{1,\dots,N}$), as depicted in Fig. 2b. For our experiments, we consider as a third scenario a dense-urban area of the city center of Bologna, Italy (Fig. 10), comprising a total of 11K samples, divided into training and test set using an 80/20 partitioning. We consider RF fingerprints comprising the RSRP measured from the three e-NodeBs (eNBs) represented in Fig. 10 (31).

$$\text{RF}_i = \{\text{RSRP}_{\text{i, A}}, \text{RSRP}_{\text{i, B}}, \text{RSRP}_{\text{i, C}}; \{\text{LAT}_i, \text{LON}_i\}\} \quad (31)$$

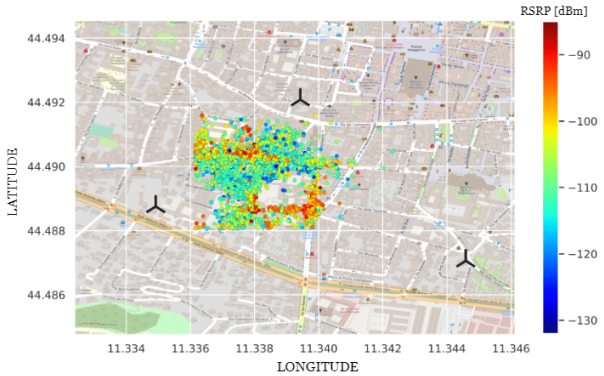


Fig. 10: Reference scenario for fingerprinting-based localization experiment. MDT data yielding RSRP samples from the three depicted cells. Data collected from the city center of Bologna, Italy.

As an illustrative ML algorithm, we employ a vanilla random forest (RF) regressor and evaluate the performance

obtained in the following cases: (i) RF regressor trained on the original user positions and original rsrp samples, (ii) RF regressor trained on the original user positions and generated rsrp samples, (iii) RF regressor trained on the generated user positions and generated rsrp samples. This allows us to determine the performance degradation brought by the individual components of our generative process. In Tab. 3 we present our numerical findings. The RMSE attained by

	RF (i)	RF (ii)	RF (iii)
RMSE	72.56 [m]	76.47 [m]	77.54 [m]

TABLE 3: Fingerprinting performance for an RF regressor trained on the original dataset (i), original positions, synthetic RSRP (ii), synthetic positions, synthetic RSRP (iii).

the RF regressor, trained on both the original and synthetic MDT fingerprints, resulted in values of 72.56 and 77.54 meters [m], respectively. These results provide additional evidence supporting the effectiveness of the proposed generative framework, as fingerprinting localization based on synthetic samples demonstrates comparable performance to the algorithm trained on original data. Notably, the integration of synthetic user positions generated through KDE results in a performance decrease of approximately 1 meter compared to case (ii). This highlights the probabilistic Bayesian regressor component as the main factor responsible for the (fair) performance degradation compared to case (i). In addition, we investigate the scenario where fingerprints are constructed based on simulated received power samples in place of real RSRP samples. Received power samples p_i are generated according to equation (32), with a fixed transmit power of $p_0 = 10$ [dB], an exponent of $\beta = [2, 4]$, and varying levels of shadowing standard deviation, denoted as σ_S [dB].

$$p_{i,A} = p_0 - \left(\frac{4\pi d_A}{\lambda}\right)^{\beta} + n \sim \mathcal{N}(0, \sigma_S^2) \ . \tag{32}$$

Accordingly, fingerprints are composed as:

$$\text{RF}_i = \{p_{i,A}, p_{i,B}, p_{i,C}; \{\text{LAT}_i, \text{LON}_i\}\} \ . \tag{33}$$

The primary aim of this evaluation is to assess our generative framework's performance under diverse levels of variability (σ_S) in the target variable. In Fig. 11, shown above, the error curves are reported as a function of σ. Notably, the RF regressor trained on the synthetic dataset exhibits very similar performance (on average, < 1 [m] of degradation) to the RF regressor trained on the original, simulated fingerprints. The marginally larger difference (< 5 [m]) observed using the real RSRP samples could potentially be attributed to location-dependent inherent noise σ present in the data, stemming from the crowdsourcing collection mechanism, as opposed to the constant σ_S used in the simulated scenario. This observation reinforces the validity and versatility of our proposed approach, as it demonstrates its effectiveness with diverse types of geo-located datasets such as real MDT and simulated power samples.

7 Conclusion

In this work, we propose an innovative and comprehensive framework that transcends specific applications and data types, enabling the conditional generation of crowdsourced datasets with location information in mobile and IoT networks. We conducted extensive numerical experiments on the illustrative task of generating MDT data, comparing it

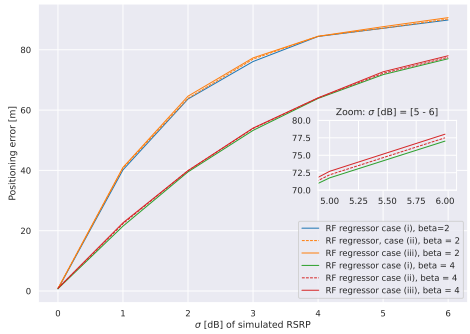


Fig. 11: Fingerprinting results: RF regressor trained on original vs synthetic fingerprints as a function of σ_S and β.

in-depth with a large-scale original dataset collected from an MNO's network infrastructure. The results demonstrate that our proposed framework is well-suited for the synthetic generation/augmentation of real-world crowdsourcing datasets. The framework exhibits remarkable interpolation capabilities, with minimal degradation in predictive performance (only 1.22 [dB]) when trained on a downsampled dataset (1M to 1K samples). Additionally, the model shows increased epistemic uncertainty values in areas of extrapolation, which enhances its trustworthiness and suitability for strategic planning of new measurement campaigns. Moreover, the model excels in performing conditional regression and accurately reproduces rare network conditions, such as predicting RSRQ at very high or very low average loads. Furthermore, we demonstrate the model's robustness to misspecification through analytical and empirical means. Finally, the generated synthetic samples faithfully retain comparable performance on downstream tasks, such as fingerprinting-based localization. In the future, our generative framework could be applied in an online active learning setting, where data is collected from a group of distributed agents. In this context, our framework's inherent ability to express uncertainty in newly generated samples could play a crucial role in balancing the tradeoff between generating synthetic data (which is cost-free) and collecting new data, which incurs in communication costs.

Acknowledgments

This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART").

References

[1] H. Navidan *et al.*, "Generative adversarial networks (gans) in networking: A comprehensive survey & evaluation," *Computer Networks*, vol. 194, p. 108149, 2021.

[2] Z. Wang *et al.*, "Generative adversarial networks in computer vision: A survey and taxonomy," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–38, 2021.

[3] A. Radford *et al.*, "Improving language understanding by generative pre-training," 2018.

[4] E. Ayanoglu *et al.*, "Machine learning in nextg networks via generative adversarial networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 2, pp. 480–501, 2022.

[5] C. Zou *et al.*, "Generative adversarial network for wireless communication: Principle, application, and trends," *IEEE Communications Magazine*, pp. 1–7, 2023.

[6] 3rd Generation Partnership Project (3GPP), *TS 37.320 - Universal Terrestrial Radio Access (UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRA); Radio measurement collection for Minimization of Drive Tests (MDT); Overall description; Stage 2*, 2021, release 16.6.0.

[7] W.A. Hapsari *et al.*, "Minimization of drive tests solution in 3gpp," *IEEE Communications Magazine*, vol. 50, no. 6, pp. 28–36, 2012.

[8] M. Skocaj *et al.*, "Cellular network capacity and coverage enhancement with mdt data and deep reinforcement learning," *Computer Communications*, vol. 195, pp. 403–415, 2022.

[9] C. Muller *et al.*, "Crowdsourcing for climate and atmospheric sciences: current status and future potential," *International Journal of Climatology*, vol. 35, no. 11, pp. 3185–3203, 2015.

[10] M.N. Kamel Boulos *et al.*, "Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, ogc standards and application examples," *International journal of health geographics*, vol. 10, no. 1, pp. 1–29, 2011.

[11] X. Kong *et al.*, "Mobile crowdsourcing in smart cities: Technologies, applications, and future challenges," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8095–8113, 2019.

[12] V. Pilloni, "How data will transform industrial processes: Crowdsensing, crowdsourcing and big data as pillars of industry 4.0," *Future Internet*, vol. 10, no. 3, 2018.

[13] L. Shu *et al.*, "When mobile crowd sensing meets traditional industry," *IEEE Access*, vol. 5, pp. 15 300–15 307, 2017.

[14] M. Razghandi *et al.*, "Variational autoencoder generative adversarial network for synthetic data generation in smart home," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 4781–4786.

[15] ——, "Smart home energy management: Vae-gan synthetic dataset generator and q-learning," *arXiv preprint arXiv:2305.08885*, 2023.

[16] J. Johansson *et al.*, "Minimization of drive tests in 3gpp release 11," *IEEE Communications Magazine*, vol. 50, no. 11, pp. 36–43, 2012.

[17] C. Mizzi *et al.*, "Unraveling pedestrian mobility on a road network using icts data during great tourist events," *EPJ Data Science*, vol. 7, 12 2018.

[18] A. Scaloni *et al.*, "Multipath and doppler characterization of an electromagnetic environment by massive mdt measurements from 3g and 4g mobile terminals," *IEEE Access*, vol. 7, pp. 13 024–13 034, 2019.

[19] A.J. Garcia *et al.*, "Big data analytics for automated qoe management in mobile networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 91–97, 2019.

[20] O. Urra *et al.*, "Spatial crowdsourcing with mobile agents in vehicular networks," *Vehicular Communications*, vol. 17, pp. 10–34, 2019.

[21] M.A. Lebre *et al.*, "Efficient vehicular crowdsourcing models in vanet for disaster management," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–5.

[22] H.D. Trinh *et al.*, "Mobile traffic prediction from raw data using lstm networks," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2018, pp. 1827–1832.

[23] L. Klus *et al.*, "Crowdsourcing solutions for data gathering from wearables," in *Proceedings of XXXV Finnish URSI Convention on Radio Science*. URSI, 2019.

[24] M. da Silva *et al.*, "Identifying privacy functional requirements for crowdsourcing applications in smart cities," in *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2018, pp. 106–111.

[25] B. Hughes *et al.*, "Generative adversarial learning for machine learning empowered self organizing 5g networks," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, 2019, pp. 282–286.

[26] P.D. Francesco *et al.*, "Assembling and using a cellular dataset for mobile network analysis and planning," *IEEE Transactions on Big Data*, vol. 4, no. 4, pp. 614–620, 2018.

[27] C. Sun *et al.*, "Gendt: Mobile network drive testing made efficient with generative modeling," ser. CoNEXT '22. New York, NY, USA: Association for Computing Machinery, 2022.

[28] J. Thrane *et al.*, "Drive test minimization using deep learning with bayesian approximation," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5.

[29] V. Kuleshov *et al.*, "Accurate uncertainties for deep learning using calibrated regression," in *International conference on machine learning*. PMLR, 2018, pp. 2796–2804.

[30] M. Zecchin *et al.*, "Robust bayesian learning for reliable wireless ai: Framework and applications," 2022.

[31] N. Di Cicco *et al.*, "Calibrated probabilistic qot regression for unestablished lightpaths in optical networks," in *2022 International Balkan Conference on Communications and Networking (BalkanCom)*, 2022, pp. 21–25.

[32] L. Zhu *et al.*, "Deep and confident prediction for time series at uber," in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 103–110.

[33] 3rd Generation Partnership Project (3GPP), *TS 36.214 - Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer; Measurements*, 3GPP, 2022, release 17.0.0.

[34] C. Blundell *et al.*, "Weight uncertainty in neural network," in *International conference on machine learning*. PMLR, 2015, pp. 1613–1622.

[35] Y. Gal *et al.*, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[36] O. Dürr *et al.*, *Probabilistic Deep Learning: With Python, Keras and TensorFlow Probability*. Manning Publications, 2020.

[37] N. Patki *et al.*, "The synthetic data vault," in *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2016, pp. 399–410.

[38] D. Rezende *et al.*, "Variational inference with normalizing flows," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach *et al.*, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 1530–1538.

[39] L. Xu *et al.*, *Modeling Tabular Data Using Conditional GAN*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[40] S. Wkeglarczyk, "Kernel density estimation and its application," in *ITM Web of Conferences*, vol. 23. EDP Sciences, 2018, p. 00037.

[41] A. Malinin *et al.*, "Uncertainty in gradient boosting via ensembles," in *International Conference on Learning Representations*, 2021.

[42] H. Kaur *et al.*, "A systematic review on imbalanced data challenges in machine learning: Applications and solutions," *ACM Comput. Surv.*, vol. 52, no. 4, aug 2019.

[43] P. Ren *et al.*, "A survey of deep active learning," *ACM computing surveys (CSUR)*, vol. 54, no. 9, pp. 1–40, 2021.

[44] D. Burghal *et al.*, "A comprehensive survey of machine learning based localization with wireless signals," *arXiv preprint arXiv:2012.11171*, 2020.

Marco Skocaj (Member, IEEE) is currently working toward the Ph.D. degree with the Department of Electronic, Information and Electrical Engineering at the University of Bologna and WiLab, CNIT. Since 2021, he is chair of the HA1 (Datasets) working group in COST action 20120 INTERACT. His research interests include Radio Resource Management, Machine Learning, Distributed Learning, Autonomous Networks and Optimization.

Lorenzo Mario Amorosa (Graduate Student Member, IEEE) is currently working toward the Ph.D. degree with the Department of Electronic, Information and Electrical Engineering at the University of Bologna. He is Research Associate at the National Laboratory of Wireless Communications (WiLab) of CNIT (the National, Inter-University Consortium for Telecommunications). His research interests include Decentralized Artificial Intelligence, Cooperative Multi-Agent Systems, Machine Learning for Industrial IoT.

Michele Lombardi is an associate professor at DISI, University of Bologna since October 2021, within the Artificial Intelligence group. His research activity is related to the integration of Optimization and Artificial Intelligence techniques, with an emphasis on Machine Learning, Constraint Programming, Mathematical Programming, and SAT Modulo Theories.

Roberto Verdone (Senior Member, IEEE) is full professor at the University of Bologna, since 2001. He is Director of WiLab, the Italian Laboratory of Wireless Communications of CNIT. He is also co-Director of the Joint Innovation Center on "Intelligent IoT for 6G" with Huawei. His main research interests are on the evolution from 5G to 6G, and the Internet of Things. He published 200 scientific papers and few books on various aspects of wireless communications.