Robust Bayesian Learning for Reliable Wireless AI: Framework and Applications

Matteo Zecchin, *Student Member, IEEE,* Sangwoo Park, *Member, IEEE,* Osvaldo Simeone, *Fellow, IEEE,* Marios Kountouris, *Fellow, IEEE*, David Gesbert, *Fellow, IEEE*

***Abstract*—This work takes a critical look at the application of conventional machine learning methods to wireless communication problems through the lens of reliability and robustness. Deep learning techniques adopt a frequentist framework, and are known to provide poorly calibrated decisions that do not reproduce the true uncertainty caused by limitations in the size of the training data. Bayesian learning, while in principle capable of addressing this shortcoming, is in practice impaired by model misspecification and by the presence of outliers. Both problems are pervasive in wireless communication settings, in which the capacity of machine learning models is subject to resource constraints and training data is affected by noise and interference. In this context, we explore the application of the framework of *robust* Bayesian learning. After a tutorial-style introduction to robust Bayesian learning, we showcase the merits of robust Bayesian learning on several important wireless communication problems in terms of accuracy, calibration, and robustness to outliers and misspecification.**

***Index Terms*—Bayesian learning, robustness, localization, modulation classification, channel modeling**

I. Introduction

Artificial intelligence (AI) is widely viewed as a key enabler of 6G wireless systems. Research on this topic has mostly focused on identifying use cases and on mapping techniques from the vast literature on machine learning to given problems [1]–[3]. At a more fundamental level, there have been efforts to integrate well-established communication modules, e.g., for channel encoding and decoding, with data-driven designs, notably via tools such as model unrolling [4], [5]. All these efforts have largely relied on *deep learning* libraries and tools. The present paper takes a critical look at the use of this conventional methodology through the lens of *reliability* and *robustness* . To this end, we explore the potential benefits of the alternative design framework of *robust Bayesian learning* by focusing on several key wireless communication applications, namely modulation classification, indoor and outdoor localization, and channel modeling and simulation.

A. Frequentist vs. Bayesian Learning

In *frequentist* learning, the output of the training process is a single model – typically, a single vector of weights for a neural network – obtained by minimizing the training loss. This approach is justified by the use of the training loss as an estimate of the population loss, whose computation would require averaging over the true, unknown distribution of the data. This estimate is only accurate in the presence of sufficiently large data sets. While abundant data is common in the benchmark tasks studied in the computers science literature, the reality of many engineering applications is that data are often scarce. In wireless communications, the problem is particularly pronounced at the physical layer, in which fading dynamics imply short stationary intervals for data collection and training [6]–[9].

The practical upshot of the reliance on frequentist learning is that, in the presence of limited data, decisions made by AI models (such as neural networks) tend to be *poorly calibrated*, providing confidence scores that do not match their true accuracy [10], [11]. As a result, an AI model may output a decision and a large estimate of its correctness, say 95%, while the accuracy of its prediction is significantly lower. This is an issue problem in many engineering applications, including emerging communication networks (e.g., 5G and beyond), in which a more or less confident decision should be treated differently by the end user [12].

The framework of *Bayesian learning* addresses the outlined shortcomings of frequentist learning [13], [14]. At its core, Bayesian learning optimizes over a *distribution* over the model parameter space. This way, if several models fit the data almost equally well, Bayesian learning does not merely select one of the models, disregarding uncertainty; rather it assigns similar distribution values to all such models [15]. As a result, decisions produced by AI models trained via Bayesian learning can account for the "opinions" of multiple models by averaging their predictions using the optimized distribution [16], [17]. Assuming that the model is well specified, the uncertainty quantification produced by Bayesian learning can hence be much more accurate than for frequentist learning. Bayesian learning has recently been applied in [11] by focusing on the problem of demodulation over fading channels; as well as in [18] for detection over multiple-antenna channels.

The work of M. Zecchin and D.Gesbert is funded by the Marie Curie action WINDMILL (grant No. 813999), while O. Simeone and S. Park have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (Grant Agreement No. 725731). M. Kountouris has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (Grant agreement No. 101003431).

Matteo Zecchin, Marios Kountouris and David Gesbert are with the Communication Systems Department, EURECOM, Sophia-Antipolis, France (e-mail: zecchin@eurecom.fr; kountour@eurecom.fr; gesbert@eurecom.fr).

Sangwoo Park and Osvaldo Simeone are with the King's Communications, Learning & Information Processing (KCLIP) lab, Department of Engineering, King's College London, London WC2R 2LS, U.K. (e-mail: sangwoo.park@kcl.ac.uk; osvaldo.simeone@kcl.ac.uk).

The second author has contributed to the problem definitions and to the experiments. The third author has had an active role in defining the problems, as well as in writing the text, while the last two authors have had a supervisory role.

B. Robust Bayesian Learning

Like frequentist learning, Bayesian learning assumes that the distribution underlying training data generation is the same as that producing test data. Furthermore, Bayesian learning implicitly assumes that the posited model – namely likelihood and prior distribution – is sufficiently close to the true, unknown data-generating distribution to justify the use of the posterior distribution as the optimized distribution in the model parameter space. As a result, the benefit of Bayesian learning is degraded when data is affected by outliers [19], [20] and/or when the model is misspecified [21]–[24].

Recent work has addressed both of these limitations, introducing a generalized framework that we will refer to as *robust Bayesian learning*. Robust Bayesian learning aims at providing well-calibrated, and hence reliable, decisions even in the presence of model misspecification and of discrepancies between training and testing conditions.

Model misspecification has been addressed in [22], [23]. These papers start from two observations. The first is that Bayesian learning can be formulated as the minimization of a *free energy* metric, which involves the average of the training loss, as well as an information-theoretic regularizing term dependent on a prior distribution. The conventional free energy metric can be formally derived as an upper bound on the population loss within the theoretical framework of *PAC Bayes theory* [25]–[27]. The second observation is that, in the presence of model misspecification, *model ensembling* can be useful in combining the decisions of different models that may be specialized to distinct parts of the problem space. Using these two observations, references [22], [23] introduced alternative free energy criteria that are based on a tighter bound of the population loss for ensemble predictors.

To address the problem of outliers (see e.g. [28]), different free energy criteria have been introduced, which are less sensitive to the presence of outliers. These metrics are based alternative scoring rules, such as the Brier score [29], and divergences, such as β-divergences [30], [31] and γ-divergence [32], [33], which generalize the Kullback-Leibler (KL) divergence underlying the standard free energy metric. Finally, a unified framework has been introduced in [34] that generalizes the free energy metrics introduced in [22], [23]. The approach is robust to misspecification, while also addressing the presence of outliers.

C. Main Contributions

In this paper, we explore the application of robust Bayesian learning to wireless communication systems. Our main purpose is twofold. On the one hand, we present a tutorial-style review of robust Bayesian learning in order to introduce the framework for an audience of communication engineers. On the other hand, we detail applications of robust Bayesian learning to communication systems, focusing on automated modulation classification (AMC), received signal strength indicator (RSSI)-based localization, as well as channel modeling and simulation. These applications have been selected in order to highlight the importance of considering uncertainty quantification, in addition to accuracy, while also emphasizing the problems of model misspecification and outliers in wireless communications [35]–[37].

Our specific contributions are as follows.

- We give a self-contained introduction to Bayesian and robust Bayesian learning by describing conceptual underpinnings and practical implications.
- We detail a series of applications of robust Bayesian learning to popular wireless communication problems, which are characterized by model misspecification and for which training must contend with data sets corrupted by outliers.
- As a first application, we focus on the AMC problem for intelligent spectrum sensing [38]. In this setting, the necessity of deploying lightweight models that satisfy the strict computational requirements of network edge devices can give rise to model misspecification. At the same time, the training data sets often contain non-informative outliers due to interfering transmissions from other devices. We demonstrate that robust Bayesian learning yields classifiers with good calibration performance despite model misspecification and the presence of outliers.
- As a second application, we study node localization based on crowdsourced RSSI data sets [39]. Such data sets typically contain inaccurately reported location measurements due to imprecise or malicious devices. Furthermore, owing to the complex relation between RSSI measurements and device locations, learning often happens using misspecified model classes. In this context, we demonstrate that robust Bayesian is able to properly estimate residual uncertainty about the transmitters' locations in spite of the presence of outliers and misspecified model classes.
- Finally, we apply robust Bayesian learning to the problem of channel modeling and simulation. We show via experiments that robust Bayesian learning produces accurate and well-calibrated generative models even in the presence of outlying data points.

D. Organization

This paper contains two main parts. In the first part, consisting of Sections II and III, we provide a tutorial-style review of robust Bayesian learning, along with the necessary background. The second part of the paper elaborates on the application discussed in the previous subsection.

We start the first part in Section II, where we define the learning setup and we provide a tutorial-style comparison between frequentist and Bayesian learning frameworks. In Section III-A, we introduce the concept of model misspecification and we review the m-free energy criterion [23] as a tool to mitigate the effect of misspecified model classes. In Section III-B, we define outliers and illustrate the role of robust losses to reduce the influence of outlying data samples. Finally, in Section III-C, we describe the robust Bayesian framework and review the robust m-free energy learning objective [34]. This approach simultaneously addresses model misspecification and outliers.

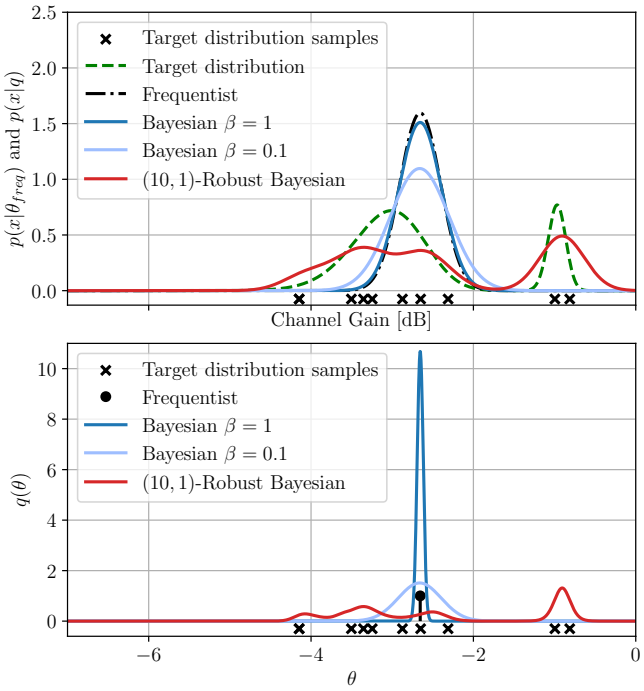


Fig. 1: Estimated distribution over a scalar channel gain (top panel) and corresponding posterior distribution $q(\theta)$ over the model parameter θ (bottom panel) for frequentist learning, Bayesian learning with $\beta \in \{1, 0.1\}$ and $(m, 1)$-robust Bayesian learning with $m = 10$. The training data set, represented as crosses, is sampled from the target distribution $\nu(x)$.

In the second part of this paper, we turn to a series of applications of robust Bayesian learning to important wireless communication problems. In Section IV, we consider the AMC task; Section V studies the problem of robust RSSI-based localization; and Section VI focuses on channel modeling and simulation. Finally, Section VII concludes the paper.

The simulation code is publicly available at: https://github.com/MatteoEURECOM/mtBayes4Wireless.git

II. Frequentist vs. Bayesian Learning

Throughout this paper we consider a standard learning set-up in which the learner has access to a data set $\mathcal{D}$ of n data points $\{z_i\}_{i=1}^n$ sampled in an independent and identically distributed (i.i.d.) fashion from a *sampling distribution* $\nu_s(z)$. As we will see, owing to the presence of outliers, the sampling distribution may differ from the *target distribution* $\nu(z)$. The general goal of learning is that of optimizing models that perform well on average with respect to the target distribution $\nu(z)$. In this section, we assume that the sampling distribution $\nu_s(z)$ equals the target distribution $\nu(z)$, and we will address the problem of outliers – which arises when $\nu_s(z) \neq \nu(z)$ – in the next section.

We will consider both *supervised learning* problems and the *unsupervised learning* problem of density estimation with applications to wireless communications. In supervised learning, a data sample $z \in \mathcal{Z}$ corresponds to a pair $z = (x, y)$ that comprises a feature vector $x \in \mathcal{X}$ and a label $y \in \mathcal{Y}$. In contrast, for density estimation, each data point $z \in \mathcal{Z}$ corresponds to a feature vector $z = x \in \mathcal{X}$.

Supervised learning is formulated as an optimization over a family of discriminative models defined by a parameterized conditional distribution $p(y|x, \theta)$ of target y given input x. The conditional distribution, or model, $p(y|x, \theta)$ is parameterized by vector $\theta \in \Theta$ in some domain Θ. In contrast, density estimation amounts to an optimization over a model defined by parameterized densities $p(x|\theta)$. In both cases, optimization targets a real-valued *loss function*, which is used to score the model θ when tested on a data point z.

A. Frequentist Learning

The goal of frequentist learning consists in finding the model parameter vector θ that minimizes the *training loss* evaluated on the data set $\mathcal{D}$, i.e.,

$$\hat{\mathcal{L}}(\theta, \mathcal{D}) = \sum_{z \in \mathcal{D}} \ell(\theta, z), \tag{1}$$

where $\ell(\theta, z)$ is the loss of model θ evaluated at z. This optimization follows the *empirical risk minimization* (ERM) principle. Accordingly, the frequentist solution is a *single* model parameter $\theta^{\text{freq}} \in \Theta$ that minimizes the training loss, i.e.,

$$\theta^{\text{freq}} = \arg\min_{\theta \in \Theta} \hat{\mathcal{L}}(\theta, \mathcal{D}). \tag{2}$$

To simplify the discussion, we assume that the solution to the ERM problem is unique, although this does not affect the generality of the presentation.

ERM is motivated by the fact that the training loss (1) is a finite-sample approximation of the true, unknown, *population loss*

$$\mathcal{L}(\theta) = \mathbb{E}_{\nu(z)}[\ell(\theta, z)], \tag{3}$$

which averages the loss over the target, and here also sampling, distribution $\nu(z)$. The discrepancy between the population loss and its approximation given by the training loss introduces uncertainty regarding the optimal model parameter

$$\theta^* = \arg\min_{\theta \in \Theta} \mathcal{L}(\theta), \tag{4}$$

which is also assumed to be unique to simplify the discussion. The error between the optimal solution θ^* and the frequentist solution θ^{freq} is a form of *epistemic uncertainty*; namely, uncertainty about the optimal model's parameter due to the limited amount of data. The epistemic uncertainty can therefore be reduced by increasing the size of the data set $\mathcal{D}$ [40].

In practice, the short stationarity intervals of the data-generating distributions associated with wireless communications often limit the size of training data sets. In this scarce data regime, epistemic uncertainty may be significant. 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 (2). As a result, frequentist learning is known to lead to poorly calibrated decision [10], [11], resulting in

over- or under-confident predictions that may cause important reliability issues. More specifically, a model is said to be over-confident whenever its predictions are complemented by a correctness likelihood estimates that are larger than the ground truth values, and it is said to be under-confident when the opposite is true [41].

B. Bayesian Learning

Bayesian learning adopts a probabilistic reasoning framework by scoring all members in the model class by means of a distribution $q(\theta)$ over the model parameter space Θ. Bayesian learning encodes in the distribution $q(\theta)$ the information obtained from data $\mathcal{D}$, as well as prior knowledge about the problem, e.g., about the norm of the optimal model parameter vector θ^* or about sparsity patterns in θ^* [42].

Mathematically, given a prior distribution $p(\theta)$ on the model parameter space, Bayesian learning can be formulated as the minimization of the *free energy criterion*

$$\hat{\mathcal{J}}(q) = \mathbb{E}_{q(\theta)}[\hat{\mathcal{L}}(\theta, \mathcal{D})] + \frac{1}{\beta}\text{KL}(q(\theta)||p(\theta)), \tag{5}$$

where $\text{KL}(q(\theta)||p(\theta))$ denotes the KL divergence between the posterior distribution $q(\theta)$ and a *prior* distribution $p(\theta)$, i.e.

$$\text{KL}(q(\theta)||p(\theta)) = \mathbb{E}_{q(\theta)}\left[\log\left(\frac{q(\theta)}{p(\theta)}\right)\right], \tag{6}$$

while $\beta > 0$ is a constant, also known as inverse temperature. Accordingly, through problem

$$\underset{q}{\text{minimize}}\, \hat{\mathcal{J}}(q), \tag{7}$$

Bayesian learning minimizes a weighted sum of the average training loss and of the discrepancy with respect to the prior distribution $p(\theta)$.

The KL term in the free energy (5) plays an essential role in differentiating between Bayesian learning and frequentist learning for small data set sizes. In fact, the KL divergence term acts as a regularizer, whose influence on the solution of problem (7) is inversely proportional to the data set size n. When the regularizer is removed, i.e., when we set $\beta \to \infty$, the solution of the problem (7) reduces to the frequentist solution (2). More precisely, the distribution $q(\theta)$ that solves problem (7) reduces to a point distribution concentrated at θ^{freq}.

The optimization (7) of the free energy criterion (5) can be theoretically justified through the *PAC Bayes* generalization framework [27], [43]. In it, the KL term is proved to quantify an upper bound on the discrepancy between training loss and population loss on average with respect to the random draws of the model parameter vector $\theta \sim q(\theta)$. Mathematically, the free energy provides an upper bound on the average population loss (when neglecting constants that are inessential for optimization), i.e.,

$$\mathbb{E}_{q(\theta)}\left[\mathcal{L}(\theta)\right] \leq \hat{\mathcal{J}}(q) + \text{const.} \tag{8}$$

As we have discussed in the previous subsection, epistemic uncertainty is caused by the difference between training and population losses, and hence between the corresponding minimizers (4) and (2). By incorporating a bound on this error, the free energy criterion (5) unlike the frequentist training loss (1), provides a way to account for epistemic uncertainty. Specifically, the posterior distribution solving (7) scores multiple (possibly infinite) models that are compatible with the evidence provided by training data set $\mathcal{D}$.

Specializing the problem (7) to the *log-loss*

$$\ell(x, y, \theta) = -\log p(y|x, \theta) \tag{9}$$

for supervised learning, and

$$\ell(x, \theta) = -\log p(x|\theta) \tag{10}$$

for density estimation, the minimization of the free energy in (7) leads to the *β-tempered posterior distribution*

$$q^{Bayes}(\theta|\mathcal{D}) \propto \prod_{(x,y)\in\mathcal{D}} p(\theta)p(y|x,\theta)^{\beta} \tag{11}$$

for supervised learning. A similar expression applies to unsupervised learning for density estimation replacing in (11) the discriminative model $p(y|x, \theta)$ by the density model $p(x|\theta)$.The distribution (11) reduces to the standard posterior distribution when $\beta = 1$. In practice, computing the posterior distribution, or more generally solving problem (7), are computationally prohibitive tasks that requires to compute intractable integrals [44]. A common approach to address this issue is through *variational inference (VI)* [45]. VI limits the scope of the optimization over a tractable set of distributions $q(\theta)$, such as jointly Gaussian variables with free mean and covariance parameters.

Let us now assume that we have obtained a distribution $q(\theta)$ as a, generally approximate, solution of problem (7). We focus first on supervised learning. Given a test input x, the *ensemble predictor* obtained from distribution $q(\theta)$ is given by

$$p(y|x, q) = \mathbb{E}_{q(\theta)}[p(y|x, \theta)]. \tag{12}$$

The average in (12) is in practice approximated by drawing multiple, say m, samples $\theta \sim q(\theta)$ from distribution $q(\theta)$, obtaining the *m-sample predictor*

$$p(y|x, \theta_1, ..., \theta_m) = \frac{1}{m}\sum_{i=1}^{m} p(y|x, \theta_i), \tag{13}$$

where samples θ_i are generated i.i.d. from distribution $q(\theta)$ for $i = 1, ..., m$, which we write as $\theta_1, ..., \theta_m \sim q(\theta)^{\otimes m}$.

In the case of density estimation, the ensemble density $p(x|q)$ is similarly defined as

$$p(x|q) = \mathbb{E}_{q(\theta)}[p(x|\theta)], \tag{14}$$

which can be approximated as

$$p(x|\theta_1, \ldots, \theta_m) = \frac{1}{m}\sum_{i=1}^{m} p(x|\theta_i), \tag{15}$$

with $\theta_1, \ldots, \theta_m \sim q(\theta)^{\otimes m}$. Henceforth, when detailing expressions for supervised learning, it will be implied that the corresponding formulas for density estimation apply by replacing $p(y|x, \theta)$ with $p(x|\theta)$ as done above to define ensemble predictors.

Given a distribution $q(\theta)$, we define the *m-sample log-loss* as

$$\ell(x, y, \theta_1, ..., \theta_m) = -\log(p(y|x, \theta_1, ..., \theta_m)), \tag{16}$$

which measures the log-loss of the m-sample predictor (13).

Example 1: To illustrate the difference between the frequentist and Bayesian learning paradigms, in Figure 1, we consider the problem of estimating the probability distribution of the channel gain of a scalar wireless channel. This is an example of unsupervised learning for density estimation. Let us assume that the channel gain density follows a true, unknown, target distribution given by the mixture of two Gaussians $\nu(x) = 0.7\mathcal{N}(x|0.5, 0.05) + 0.3\mathcal{N}(x|0.8, 0.02)$. This is shown in the top part of Figure 1 as a dashed green line. The two components may correspond to line-of-sight (LOS) and non-line-of-sight (NLOS) propagation conditions [46]. We fix a Gaussian model class $p(x|\theta) = \mathcal{N}(x|\theta, 0.25)$ and a prior distribution $p(\theta) = \mathcal{N}(\theta| - 5, 5)$. Therefore the model class is misspecified since there does not exists a value θ such that $p(x|\theta) = \nu(x)$; on the other hand, a Gaussian mixture model would be well specified in this scenario. Given the 5 data points represented as crosses in the top part of Figure 1, the estimated distribution obtained by frequentist learning is reported as a dash-dotted black curve in the top panel. In contrast, Bayesian learning returns the posterior distribution (11), which in turn yields the ensemble density (12). The distributions are shown in the top and bottom parts of the Figure 1, respectively for inverse temperature parameters $\beta = \{1, 0.1\}$. The Bayesian predictive distribution is still unimodal but it has a larger variance, which results from the combination of multiple Gaussian models according to the Bayesian posterior that does not reduce to a point distribution in virtue of the KL regularization term whose influence is controlled by β. ■

III. Robust Bayesian Learning

As we have seen in the previous section, Bayesian learning optimizes the free energy by tackling problem (7). By (8), the free energy provides a bound on the population loss as a function of the training loss when averaging over the distribution $q(\theta)$ in the model parameter space [43]. This approach has two important limitations:

- *Model misspecification*: The bound (8) provided by the free energy is known to be loose in the presence of model misspecification. Model misspecification occurs when the assumed probabilistic model $p(y|x, \theta)$ cannot express the conditional target distribution $\nu(y|x) = \nu(x, y)/\nu(x)$, where $\nu(x) = \int \nu(x, y)dy$. This causes the β-tempered posterior distribution to be generally suboptimal when the model is misspecified [28]. There exist several techniques to mitigate the effect of misspecification, for example by using tighter approximations of the ensemble risk [22], [23], using pseudo-likelihoods [47] or modeling aleatoric uncertainty [24].
- *Discrepancy between sampling and target distributions*: The sampling distribution $\nu_s(z)$ that underlies the generation of the training data set $\mathcal{D}$ may not match the target distribution $\nu(x)$ used to test the trained model. A common model for this mismatch assumes the presence of outliers in the training data [48]. This discrepancy is not accounted for in the derivation of the free energy criterion, causing Bayesian learning to be suboptimal in the presence of outliers [28]. For this reason, alternative scoring rules, such as the Brier score [29], and divergences, such as β-divergences [30], [31] and γ-divergence [32], [33] have been considered to mitigate the presence of anomalous training data points.

We observe that the two causes of suboptimality outlined in the previous paragraph are distinct. In fact, model misspecification may reflect the ignorance of the learner concerning the data generation process, or it may be caused by constraint on the computational resources of the device implementing the model. In contrast, the presence of outliers amounts to an inherent source of distortion in the data, which cannot be removed even if the learner acquired more information about the data generation process or more computing power. In this section, we review robust Bayesian learning solutions that address these two issues.

We emphasize that robustness is a multifaceted property that may refer to aspects other than model misspecification and outliers, such as covariate shift [20], adversarial attacks at inference time [49], [50] or poisoning attacks in distributed learning settings [51]–[53]. In this paper, we focus solely on model misspecification and outliers, given the importance of these aspects for applications in communication engineering. Therefore, when referring to robust Bayesian learning we will implicitly consider only robustness with respect to these two impairments.

A. $(m, 1)$-Robust Bayesian Learning Against Model Misspecification

In this subsection, we describe a recently proposed method that makes Bayesian learning robust against model misspecification [23]. We start by providing a formal definition of misspecification. Recall that we are focusing on supervised learning, but the presentation also applies to density estimation by replacing the discriminative model $p(x|y, \theta)$ with the density model $p(x|\theta)$.

Definition 1 (Misspecification)**.** *A model class $\mathcal{F} = \{p(y|x, \theta) : \theta \in \Theta\}$ is said to be misspecified with respect to the target distribution $\nu(x, y)$ whenever there is no model parameter vector $\theta \in \Theta$ such that $\nu(y|x) = p(y|x, \theta)$, where $\nu(y|x)$ is the conditional target distribution obtained from the joint target distribution $\nu(x, y)$.*

Under model misspecification, the free energy criterion has been shown to yield a loose bound (8) on the population loss obtained by the ensemble predictor (12) [23].

To address this problem, the m-sample free energy criterion was introduced in [23], whose minimization yields $(m, 1)$-robust learning. The reason for the notation "$(m, 1)$" will be made clear in the next two subsections. The key observation underlying this approach is that the training loss $\hat{\mathcal{L}}(\theta, \mathcal{D})$ in the standard free energy (5) does not properly account for the

performance of ensemble predictors. In fact, the log-loss of an m-sample ensemble predictor is given by $\ell(x, y, \theta_1, \ldots, \theta_m)$ in (16), and not by the log-loss $\ell(x, y, \theta)$ in (9). This is not an issue when the model is well specified, since, in this case, the minimization of the free energy (5) yields the posterior distribution (11), which is the optimal solution to the learning problem when one trusts the model to match the data generation distribution [54]. In contrast, in the presence of model misspecification, the optimal solution is generally not given by the posterior distribution (11), and better predictive performance can be obtained by directly minimizing the prediction loss $\ell(x, y, \theta_1, \ldots, \theta_m)$ in (16) accrued by the ensemble predictor (13). To account for this performance metric in the formulation of Bayesian learning, reference [23] introduced the m-sample free energy, in which the training loss $\hat{\mathcal{L}}(\theta, \mathcal{D})$ in the free energy (5) is replaced by the *m-sample training loss*

$$\begin{aligned}\hat{\mathcal{L}}(\theta_1, \ldots, \theta_m, \mathcal{D}) &= \sum_{(x,y)\in\mathcal{D}} \ell(x, y, \theta_1, ..., \theta_m) \\ &= -\sum_{(x,y)\in\mathcal{D}} \log\left(\sum_{i=1}^{m} \frac{p(y|x, \theta_i)}{m}\right). \quad (17)\end{aligned}$$

Furthermore, the *m-sample free energy* is defined as

$$\hat{\mathcal{J}}^m(q) = \mathbb{E}_{q(\theta)^{\otimes m}}[\hat{\mathcal{L}}(\theta_1, \ldots, \theta_m, \mathcal{D})] + \frac{m}{\beta}\mathrm{KL}(q(\theta)||p(\theta)), \quad (18)$$

in which the m-sample training loss is averaged over the distribution of the m samples $\theta_1, ..., \theta_m \sim q(\theta)^{\otimes m}$ used in the ensemble predictor (13). We note that the m-sample free energy has an additional parameter m and it coincides with the standard free energy (5) for $m = 1$.

Finally, the *$(m, 1)$-robust Bayesian learning* problem is defined by the optimization

$$\underset{q}{\text{minimize}}\, \hat{\mathcal{J}}^m(q). \quad (19)$$

Intuitively, according to the discussion above, the solution of problem (19) yields an ensemble predictor (15) that is more robust to model misspecification since it directly accounts for the performance of the ensemble predictor. This way, the ensemble predictor can compensate for a model mismatch by averaging over the predictions of several models via the optimized distribution $q(\theta)$. This key point is illustrated with the next example.

Example 1 (continued): Let us return to Example 1 of Figure 1. The problem is characterized by model misspecification since the target distribution $\nu(x)$ is a mixture of two Gaussian components, while the model class comprises only unimodal Gaussian models $p(x|\theta)$. In contrast to standard Bayesian learning, the ensemble density (13) obtained with the distribution $q(\theta)$ returned by $(m, 1)$-robust Bayesian learning for $m = 10$ (red curve in the top panel) is able to take advantage of ensembling to approximate both the NLOS and LOS components of the target distribution. ■

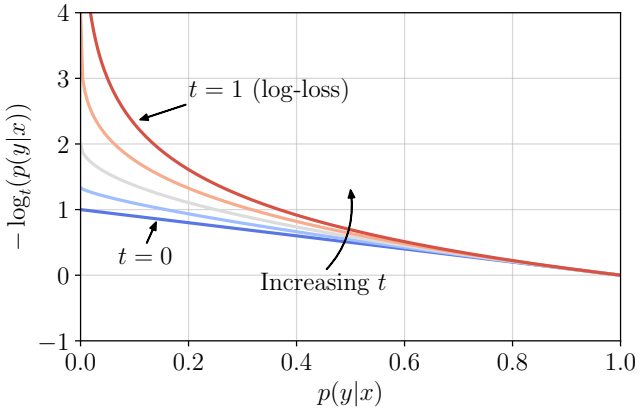


Fig. 2: t-log-loss $-\log_t(p(y|x)$ as a function of the predictive probability $p(y|x)$ for different values of t. For $t = 1$, the t-log-loss coincides with the conventional log-loss. A sample (x, y) with a low predictive probability $p(y|x) \to 0$ is assigned an unbounded log-loss value. In contrast, for $t < 1$, the t-log-loss is bounded by $(1-t)^{-1}$, limiting the influence of outliers.

B. $(1, t)$-Robust Bayesian Learning Against Outliers

We now turn to methods that make Bayesian learning robust against discrepancies between training and testing conditions. Specifically, we adopt the standard model introduced in the classical paper [48], which accounts for the mismatch between the distributions of the training data and of the test data via an additive error that can be interpreted in terms of the presence of outliers in the training set. Accordingly, we assume that the training data is generated from a sampling distribution $\nu_s(x, y)$ that is given by the weighted sum of the target, testing, distribution $\nu(x, y)$ and of an out-of-distribution (OOD) distribution $\xi(x, y)$. The OOD component models the said mismatch between training and testing distributions. A formal definition follows.

Assumption 1 (Outliers)**.** *The sampling distribution is given by*

$$\nu_s(x, y) = (1 - \epsilon)\nu(x, y) + \epsilon\xi(x, y) \quad (20)$$

where $\nu(x, y)$ is the target distribution; $\xi(x, y)$ is the OOD distribution; and $\epsilon \in [0, 1]$ denotes the mismatch error.

According to model (20), the training data set can be interpreted as containing, on average, a fraction ϵ of outliers, which are drawn from the distribution $\xi(x, y)$. In contrast, testing is done based on data from the target distribution $\nu(x, y)$. In this regard, in order for model (20) to be meaningful, one typically assumes that the OOD measure $\xi(x, y)$ is large for pairs of (x, y) at which the target measure $\nu(x, y)$ is small. This ensures that outlying data points $(x, y) \sim \xi(x, y)$ tend to be in part of the domain that is not covered by the target distribution. Therefore, the model (20) can be equivalently stated as assuming that the training data contains a fraction ϵ of training points that are sampled from low-probability regions of the testing distribution.

The performance of both frequentist and Bayesian learning is known to be sensitive to outliers when the log-loss is adopted to evaluate the training loss. This sensitivity is caused by the unbounded value of the log-loss (9) when evaluated on anomalous data points to which the model assigns low probabilities $p(y|x,\theta)$. This is illustrated in Figure 2 for a general conditional distribution $p(y|x)$. A number of papers have proposed to mitigate the effect of outliers by replacing the log-loss in favor of more robust losses [30]–[33], [55].

A well-explored solution is to adopt the t-log-loss. For for a model $p(y|x,\theta)$, the *t-log-loss* is defined as

$$-\log_t(p(y|x,\theta)) := -\frac{1}{1-t}\left(p(y|x,\theta)^{1-t}-1\right) \quad \text{for } p>0, \tag{21}$$

where $t \in [0,1)\cup(1,\infty)$; and

$$-\log_1(p(y|x,\theta)) := -\log(p(y|x,\theta)) \ \text{ for } p>0. \tag{22}$$

By (22) the standard log-loss is obtained with $t=1$, while for $t<1$ the associated loss function is bounded by $(1-t)^{-1}$, as shown in Figure 2.

Using the t-log-loss in lieu of the standard log-loss in the training loss (1) we obtain the t-training loss

$$\hat{\mathcal{L}}_t(\theta,\mathcal{D}) = -\sum_{(x,y)\in\mathcal{D}} \log_t\left(p(y|x,\theta)\right), \tag{23}$$

which leads to the corresponding *t-free energy*

$$\hat{\mathcal{J}}_t(q) = \mathbb{E}_{q(\theta)}[\hat{\mathcal{L}}_t(\theta,\mathcal{D})] + \frac{1}{\beta}\mathrm{KL}(q(\theta)||p(\theta)). \tag{24}$$

Accordingly, *$(1,t)$-robust Bayesian learning* is defined by the minimization [23]

$$\underset{q}{\text{minimize}}\ \hat{\mathcal{J}}_t(q). \tag{25}$$

The solution of problem (25) becomes more robust to outliers as t is reduced below 1 towards 0. In fact, with a t value close to 1, if a few, outlying, training points are assigned an incorrect probability by the model, the overall training loss tends to be large (see Figure 2), causing conventional Bayesian learning to be sensitive to outliers. In contrast, as t decreases, the t-log-loss does not penalize as much models that do not properly "cover" outlying training points, enhancing robustness to outliers.

Example 2: To highlight the effect of outliers, in Figure 3, we consider the same channel gain estimation problem described in Example 1, but we now assume that the original training data set (black crosses) is contaminated by an outlying data point (red cross). The $(m,1)$-robust Bayesian learning solution (red curve with $m=10$) is based on the standard log-loss and is observed to be significantly affected by the presence of the outliers. As a result, the estimated distribution for the $(m,1)$-robust Bayesian learning concentrates a relevant fraction of its mass around the outlier. In contrast, the $(1,t)$-robust Bayesian solution (gray curve) with $t=0.4$ is less influenced by the outlying data point. However, like Bayesian learning, it is not able to take advantage of ensembling and to approximate both LOS and NLOS components. This observation justifies the (m,t)-robust Bayesian learning approach described next. ■

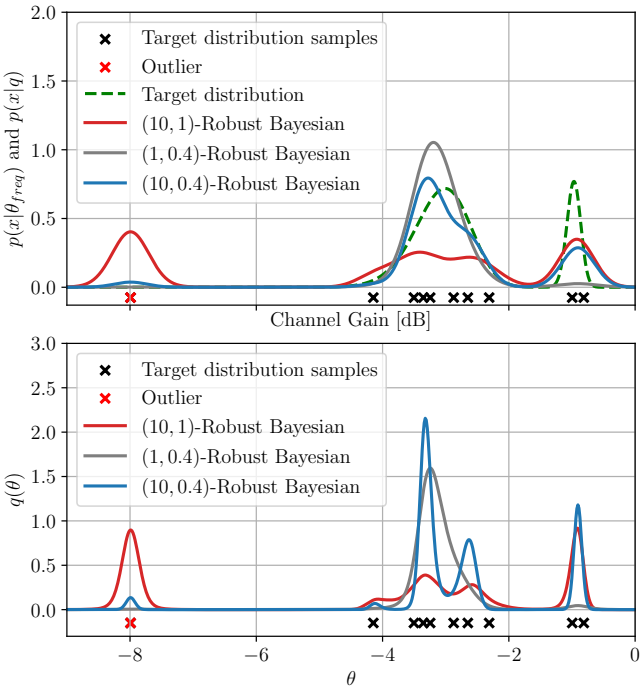


Fig. 3: Estimated distribution over channel gains (top panel) and posterior distribution over the model parameter θ (bottom panel) of a density model trained following $(m,1)$-robust Bayesian learning, the $(1,t)$-robust Bayesian learning and the (m,t)-robust Bayesian learning. The training data set, represented as crosses, comprises samples from the sampling distribution $\nu(x)$ (black) and an outlier (red).

C. (m,t)-Robust Bayesian Learning Against Model Misspecification and Outliers

To concurrently address model misspecification and the presence of outliers, reference [34] formally introduced (m,t)-robust Bayesian learning, which minimizes a free energy metric integrating both m-sample predictors and the t-log-loss. To describe it, let us first define the *(m,t)-training loss*

$$\hat{\mathcal{L}}_t(\theta_1,\ldots,\theta_m,\mathcal{D}) = -\sum_{(x,y)\in\mathcal{D}} \log_t\left(\sum_{i=1}^{m}\frac{p(y|x,\theta_i)}{m}\right), \tag{26}$$

which is obtained from the m-sample training loss (17) by replacing the log-loss with the t-log-loss. The *(m,t)-free energy* is accordingly defined as

$$\hat{\mathcal{J}}_t^m(q) = \mathbb{E}_{q(\theta)^{\otimes m}}[\hat{\mathcal{L}}_t(\theta_1,\ldots,\theta_m,\mathcal{D})] + \frac{m}{\beta}\mathrm{KL}(q(\theta)||p(\theta)), \tag{27}$$

and *(m,t)-robust Bayesian learning* amounts to the minimization

$$\underset{q}{\text{minimize}}\ \hat{\mathcal{J}}_t^m(q). \tag{28}$$

Note that (m,t)-robust Bayesian learning recovers standard Bayesian learning by setting $t=1$ and $m=1$, as well as $(m,1)$-robust Bayesian learning with $t=1$ and the $(1,t)$-robust Bayesian learning for $m=1$.

Example 2 (continued): Returning to Example 2, we now consider the performance of (m,t)-robust Bayesian learning for $m = 10$ and $t = 0.4$. The resulting distribution (blue line) with $m = 10$ and $t = 0.4$ seems to be able to better to approximate the target distribution by reducing the effect of the outliers, while also taking advantage of ensembling to combat misspecification.

IV. Robust and Calibrated Automatic Modulation Classification

As a first application of robust Bayesian learning we consider the AMC problem. This is the task of classifying received baseband signals in terms of the modulation scheme underlying their generation. The relation between the received signal and the chosen modulation scheme is often mediated by complex propagation phenomena, as well as hardware non-idealities at both the receiver and the transmitter side. As a result, model-based AMC methods often turn out to be inaccurate because of the overly simplistic nature of the assumed models [56]. In contrast, machine learning based AMC has been shown to be extremely effective in correctly classifying received signals based on signal features autonomously extracted from data [57]. We refer to [58] and references therein for a comprehensive overview.

All prior works on learning-based AMC, reviewed in [58], have adopted frequentist learning. In this section, we consider the practical setting in which AMC must be implemented on resource-constrained devices, entailing the use of small, and hence mismatched, models; and the training data sets are characterized by the presence of outliers due to interference.

A. Problem Definition and Performance Metrics

The AMC problem can be framed as an instance of supervised classification, with the training data set $\mathcal{D}$ comprising pairs (x, y) of discrete-time received baseband signal x and modulation label y, with $\mathcal{Y}$ being the set of possible modulation schemes. Each training data point $(x, y) \in \mathcal{D}$ is obtained by transmitting a signal with a known modulation $y \in \mathcal{Y}$ over the wireless channel, and then recording the received discrete-time vector x at the receiver end. The outlined procedure determines the unknown sampling distribution $\nu_s(x, y)$.

We evaluate the performance of AMC on a testing data set $\mathcal{D}_{te}$ in terms of accuracy and *calibration*. To describe calibration performance metrics, let us consider a predictive distribution $p(y|x)$, which may be the frequentist distribution $p(y|x, \theta^{\text{freq}})$, or the ensemble distribution (13) in the cases of Bayesian learning and robust Bayesian learning. A hard prediction $\hat{y}$ is obtained as the maximum-probability solution

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} p(y|x). \tag{29}$$

The corresponding *confidence score* assigned by the predictor $p(y|x)$ is the probability $p(\hat{y}|x) \in [0, 1]$. The calibration of a classifier measures the degree to which the confidence score $p(\hat{y}|x) \in [0, 1]$ reflects the true probability of correct classification $P[\hat{y} = y|x]$ conditioned on the input x.

We adopt the standard reliability diagrams [59] and the expected calibration error as diagnostic tools for the calibration performance [10]. Both metrics require binning the output of the classifier confidence score $p(\hat{y}|x)$ into M intervals of equal size, and then grouping the testing data points $(x, y) \in \mathcal{D}_{te}$ based on the index of the bin for the confidence score $p(\hat{y}|x)$. For each bin $\mathcal{B}_m$, the *within-bin accuracy* is defined as

$$\text{Acc}(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|} \sum_{(x,y) \in \mathcal{B}_m} \mathbb{1}\{\hat{y} = y\}, \tag{30}$$

which measures the fraction of test samples within the bin that are correctly classified; and the *within-bin confidence* as

$$\text{Conf}(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|} \sum_{(x,y) \in \mathcal{B}_m} p(\hat{y}|x), \tag{31}$$

which is the average confidence level for the test samples within the bin.

The *reliability diagram* plots within-bin accuracy and within-bin confidence as a function of the bin index m. As a result, a reliability diagram visualizes the relation between confidence and accuracy of a predictor, establishing whether a classifier is over-confident $(\text{Conf}(\mathcal{B}_m) > \text{Acc}(\mathcal{B}_m))$, under-confident $(\text{Conf}(\mathcal{B}_m) < \text{Acc}(\mathcal{B}_m))$ or well-calibrated $(\text{Conf}(\mathcal{B}_m) \approx \text{Acc}(\mathcal{B}_m))$.

The *expected calibration error (ECE)* summarizes the calibration performance of a classifier as a single number obtained as the weighted sum of the absolute difference between within-bin accuracy and within-bin confidence, namely

$$\text{ECE} = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{\sum_{m=1}^{M} |\mathcal{B}_m|} \left|\text{Conf}(\mathcal{B}_m) - \text{Acc}(\mathcal{B}_m)\right|. \tag{32}$$

By this definition, one can generally conclude that a lower ECE indicates a better calibrated predictor.

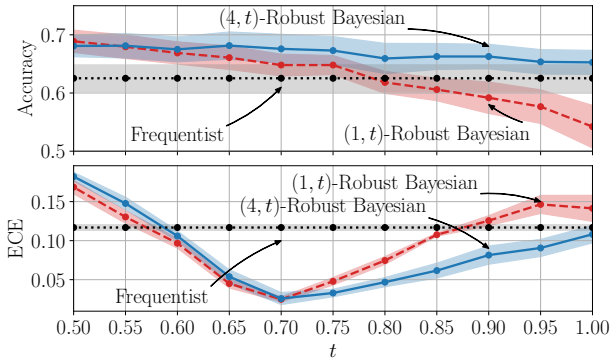


Fig. 4: Average test accuracy and ECE for AMC over the DeepSIG: RadioML 2016.10A data set [57] for frequentist and (m, t)-robust Bayesian learning as a function of the parameter t. The test set is free from interference, while the training set is subject to interference $(\epsilon = 0.5)$.

B. Data Set

We adopt the *DeepSIG: RadioML 2016.10A* data set [57]. This is a synthetic data set that contains 220K vectors of I/Q samples of signals comprising 8 digital modulation schemes

(BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK) and 3 analog modulations (WB-FM, AM-SSB, AM-DSB). We focus on the problem of classifying the 8 digital modulation schemes using received signals recorded at different SNR levels ranging from 0 dB to 18 dB. To account for a mismatch between training and testing conditions under model (20), we focus on a scenario in which an additional transmitter is occasionally active during training data collection, and it is later removed or deactivated when testing takes place. This way, the training data set comprises informative training examples, not affected by interference, and uninformative training examples, for which it is impossible to assign an unambiguous label given the simultaneous presence of multiple signals.

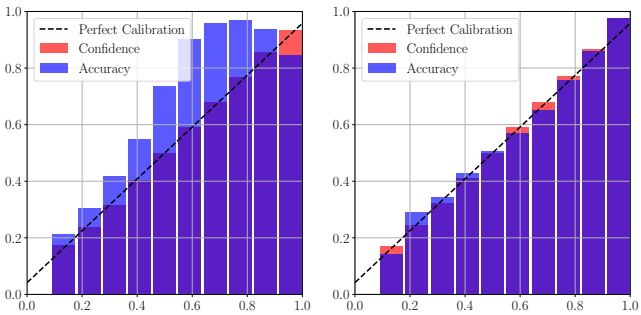
Formally, we model the presence of interference during training by generating an ϵ-"contaminated" version of the original data set. In it, with probability $\epsilon \in [0, 1)$, the original training sample x is summed to an interfering signal x' picked uniformly at random from the data set. Note that the interfering signal can be possibly generated from a different modulation scheme. According to the contamination model of Assumption 1, the samples affected by interference represent *outliers* distributed according to $\xi(y, x)$, since no interference is assumed during testing.

With regards to the adopted scenario, we emphasize that the model (20) is not limited to situations in which deviations between testing and training distributions are characterized by additional interference. In particular, it may be that interference affects the testing phase (as described by distribution $\nu(x, y)$) and not the training phase (as described by distribution $\nu_s(x, y)$), or that the training phase is occasionally affected by a different type of interference.

We consider 30% of the available samples for training; 20% of the samples for validation; and the remaining 50% for testing. The use of a small training data set is intentional, as we wish to focus on a regime characterized by data scarcity.

C. Implementation

We consider a probabilistic model $p(y|x, \theta)$ represented by a lightweight convolutional neural network (CNN) architecture comprising of two convolutional layers followed by two linear layers with 30 neurons each. The first convolutional layer has 16 filters of size 2×3, and the second layer has 4 filters of size 1×2. We adopt the Exponential Linear Unit (ELU) activation with parameter $\alpha = 1$. The lightweight nature of the architecture is motivated by the strict computational and memory requirements at network edge devices. As a result, the CNN model is generally *misspecified*, in the sense that, following Definition 1, the complex relation between received signal and chosen modulation scheme cannot be exactly represented using the model. In fact, the probabilistic model $p(y|x, \theta)$ is obtained here by composing the output of a CNN model $f_\theta(x)$ with the softmax function. When the CNN model $f_\theta(x)$ is capacity constrained, this class of predictive distribution may well not contain the ground-truth distribution $p(y|x)$ relating the received signal x and the modulation scheme y, as the latter may be more complex due to noise, transmitter and receiver circuit non-idealities, and propagation phenomena.



(a) Frequentist Learning (b) Robust Bayesian Learning

Fig. 5: Reliability diagrams for frequentist (left) and (m, t)-robust Bayesian learning for $m = 4$ and $t = 0.7$ (right) for AMC over the DeepSIG: RadioML 2016.10A data set [57].

In the training data set, half of the samples are affected by interference, i.e., $\epsilon = 0.5$. For Bayesian learning, we adopt a Gaussian variational distribution $q(\theta) = \mathcal{N}(\theta|\mu, \Sigma)$ over the CNN model parameter vector θ, which has dimension 16404 and it includes all the weights of the neural network. Accordingly, the mean μ and diagonal covariance matrix Σ are optimized, while we fix the prior $p(\theta) = \mathcal{N}(\theta|0, I)$. Optimization for both frequentist and Bayesian methods is carried out via Adam with a learning rate $\eta = 0.001$, and the reparametrization trick is implemented for Bayesian learning [60]. In our experiments we set $\beta = 0.01$. The number of samples used to evaluate the ensemble prediction (13) is $m = 10$. Note that this may differ from the value of m used to define the training criterion.

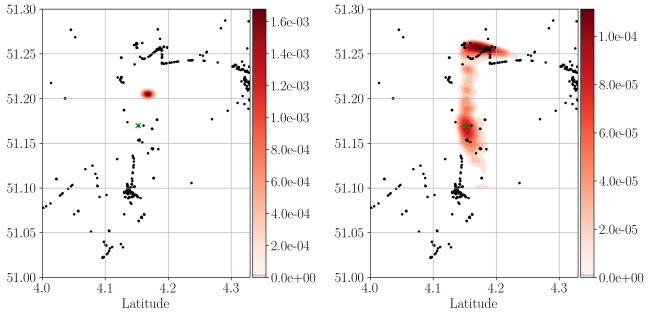
D. Results

In Figure 4 we report the average test accuracy and ECE for frequentist and (m, t)-robust Bayesian as a function of t for $m = 1$ and $m = 4$. These two values are chosen to highlight the difference between $(1, t)$-robust Bayesian learning $(m = 1)$ and (m, t)-robust Bayesian learning for $m > 1$. Furthermore, when estimating the m-sample training loss we set $m = 4$ as we do not appreciate performance gains for larger values. The main observation is that, with suitably chosen parameters (m, t), robust Bayesian learning can outperform standard frequentist learning both in terms of accuracy and calibration for $t < 1$. The smallest ECE is obtained by robust Bayesian learning for $t = 0.7$, and it is five times smaller compared to the one obtained using conventional Bayesian learning $(t = 1)$. Overall, (m, t)-robust Bayesian paradigm is able to improve the final accuracy by 5% and to reduce the ECE by five times via suitable choice of parameters (m, t).

To further elaborate on the calibration performance, in Figure 5 we compare the reliability diagrams obtained via frequentist and (m, t)-robust Bayesian learning for $m = 4$ and $t = 0.7$. While frequentist learning provides under-confident predictions, robust Bayesian learning offers well-calibrated predictions that consistently offer a small discrepancy between accuracy and confidence levels.

TABLE I: Test negative log-likelihood for RSSI localization (34) with $t = 1$ and no outliers ($\epsilon = 0$). The case $m = 1$ corresponds to conventional Bayesian learning.

	$m = 1$	$m = 2$	$m = 10$
SigfoxRural	1.70 ± 1.03	-0.43 ± 0.61	$\mathbf{-1.59 \pm 0.36}$
UTSIndoor	4.33 ± 2.32	2.25 ± 1.69	$\mathbf{2.17 \pm 1.76}$
UJIIndoor	4.86 ± 1.02	2.74 ± 0.46	$\mathbf{1.44 \pm 0.33}$



(a) Bayesian Learning (b) Robust Bayesian Learning

Fig. 6: Predictive distribution $p(y|x)$ as a function of the estimated position of the transmitter y, where x is the RSSI vector associated to the true location shown as a green cross. The black dots correspond to the locations recorded in the *SigfoxRural* data set. The left panel shows the predictive distribution for Bayesian learning, while the right panel depicts the predictive distribution for (m, t)-robust Bayesian learning with $m = 10$ and $t = 1$. No outliers are considered in the training set, i.e., $\epsilon = 0$.

V. Robust and Calibrated RSSI-Based Localization

In this section, we turn to the problem of localization. In outdoor environments, accurate localization information of a wireless device can be obtained leveraging the global navigation satellite system (GNSS). However, the performance of satellite-based positioning is severely degraded in indoor environments [61], and its power requirements are not compatible with Internet-of-Things (IoT) application characterized by ultra-low power consumption [62]. For this reason, alternative techniques have been investigated that rely on so-called *channel fingerprints*, i.e., feature extracted from the received wireless signals [63].

Among such methods, the use of *received signal strength indicators* (RSSI) measured at multiple wireless access points has been shown to provide an accessible, yet informative, vector of features. Owing to the complexity of defining explicit models relating the device location $y \in \mathcal{Y}$ with the RSSI-measurements vector $x \in \mathcal{X}$, data-driven RSSI-based localization techniques have been recently explored [64], [65]. The outlined prior work in this area has focused on machine learning models trained using the conventional frequentist approach.

In this section, we study a setting in which the training data set is collected using noisy, e.g., *crowd-sourced*, fingerprints. As such, the training set contains outliers. Furthermore, we aim at developing strategies, based on robust Bayesian learning, which can offer accurate localization, while also properly quantifying residual uncertainty.

A. Problem Definition and Performance Metrics

The RSSI-based localization problem is a supervised regression task. In it, a training sample (x, y) is obtained by measuring the RSSI fingerprint y corresponding to the transmission of a reference signal at a device located at a known position x. The general goal is to train a machine learning model $p(y|x)$ to predict the location y associated to a RSSI vector x so as to optimize accuracy and uncertainty quantification.

Given a test data set $\mathcal{D}_{te}$ and assuming that the predictive location is the mean of the predictive distribution, i.e. $\bar{y} = \mathbb{E}_{p(y|x)}[y]$, we adopt the *mean squared error (MSE)* metric

$$\text{MSE}(\mathcal{D}_{te}, p) = \frac{1}{|\mathcal{D}_{te}|} \sum_{(x,y)\in\mathcal{D}_{te}} \|y - \bar{y}\|_2 \tag{33}$$

as a measure of accuracy. Furthermore, in order to estimate the residual uncertainty about y predicted by the model, we adopt the *negative test log-likelihood* [66]

$$\text{NLL}(\mathcal{D}_{te}, p) = -\frac{1}{|\mathcal{D}_{te}|} \sum_{(x,y)\in\mathcal{D}_{te}} \log(p(y|x)). \tag{34}$$

Note that the negative log-likelihood is large if the model assigns a small probability density $p(y|x)$ to the correct output y.

B. Data Sets

We experiment on different publicly available RSSI fingerprint data sets, encompassing both outdoor and indoor conditions:

- The *SigfoxRural* data set [62] comprises $25,638$ Sigfox messages measured at 137 base stations and emitted from vehicles roaming around a large rural area (1068 km^2) between Antwerp and Gent.
- The *UTSIndoorLoc* data set [67] contains 9494 WiFi fingerprints sampled from 589 access points inside the FEIT Building at the University of Technology of Sydney, covering an area of $44,000$ m^2.
- The *UJIIndoorLoc* data set [68] contains $21,049$ WiFi fingerprints measured at 520 access points and collected from 3 building of the Jaume I University, spanning a total area of $108,703$ m^2.

To model the presence of *outliers*, we modify the training data sets described above, producing ϵ-contaminated data sets $\mathcal{D}$ as per Definition 2. This is done by replacing the target variable y for a fraction ϵ of the data points $(x, y) \in \mathcal{D}$ with a uniformly random location y within the deployment area.

C. Implementation

We consider a model class specified by a Gaussian likelihood $p(y|x,\theta) = \mathcal{N}(y|f_\theta(x), 0.01)$, where the mean $f_\theta(x)$ is the output of a fully connected neural network with two hidden layers, each with 50 neurons with ELU activations and a total of 19004 parameters. Despite the expressive power of the neural network model, each model $p(y|x,\theta)$ in this class can only account for unimodal, Gaussian distributed, residual uncertainties around the estimated position $f_\theta(x)$. Therefore, whenever the residual uncertainty about the receiver location is multimodal, the model class is *misspecified* by Definition 1. We emphasize that this situation is distinct from that studied in the previous section, in which model misspecification was caused by model capacity limitations. In fact, here, misspecification holds irrespective of the capacity of the neural network model $f_\theta(x)$. As we will see, given the complex relation between RSSI vector and location, particularly when the number of RSSI measurements is sufficiently small, residual uncertainty tends to be multimodal, making this an important problem. Training for frequentist and Bayesian learning is carried out as described in the previous section, and ensembling uses $m = 50$ samples during testing time.

D. Results

We start by considering the case in which there are no outliers, i.e., $\epsilon = 0$, thus focusing solely on the problem of misspecification. In Figure 6, we plot the predictive distribution obtained via Bayesian learning ($m = 1$, left panel) and robust Bayesian learning with $m = 10$ and $t = 1$ (right panel) for a testing sample x corresponding to the position shown as a green cross. The black dots correspond to the positions covered by the training set in the *SigfoxRural* data set. The resulting predictive distribution for conventional Bayesian learning provides a poor estimation of the true device position, and is unable to properly quantify uncertainty. In contrast, robust Bayesian learning is able to counteract model misspecification, producing a more informative predictive distribution. The distribution correctly suggests that the receiver can be in two possible areas, one of which indeed containing the true node location.

To further elaborate on the capacity of robust Bayesian learning for uncertainty quantification, in Table I we report the negative log-likelihood (34) attained by Bayesian learning ($m = 1$), as well as by robust Bayesian learning with $t = 1$ and $m = 2$ or $m = 10$ on the three data sets. Increasing the value of m is seen to yield lower negative log-likelihood scores, confirming that robust Bayesian learning provides a more precise quantification of uncertainty.

We now introduce outliers by carrying out training on contaminated data sets with different levels of contamination ϵ. Recall that the trained models are tested on a uncorrupted ($\epsilon = 0$) test data set $\mathcal{D}_{te}$. In Figure 7, we plot the test MSE (33) of frequentist learning, robust frequentist learning based on the minimization of the t-log-loss (21) with $t = 0.99$ and (m,t)-robust Bayesian learning with $m = 10$ and $t \in \{1, 0.96\}$ as a function of ϵ. For the robust learners, the parameter t is tuned by choosing the best performing value in the range $(1, 0.95]$. The MSE of frequentist learning and $(10,1)$-robust Bayesian learning are seen to degrade significantly for increasing values of ϵ. The performance loss is particularly severe for $(m,1)$-robust Bayesian learning. This is due to the mass-covering behavior entailed by the use of m-sample training loss, which in this case becomes detrimental due to the presence of outliers. In contrast, both robust frequentist with $t = 0.99$ and robust Bayesian learning with $t = 0.96$ are able to counteract the effect of outliers, retaining good predictive performance even in case of largely corrupted data sets.

VI. Robust and Calibrated Channel Simulation

The design of communication systems has traditionally relied on analytical channel models obtained via measurements campaigns. Due to the complexity of multipath propagation scenarios, in recent years generative machine learning models have introduced as an alternative to analytical models. Generative models can be trained to produce samples that mimic hard-to-model channel conditions. Applications of deep generative models in the form of variational autoencoders (VAEs) [60] and generative adversarial networks (GANs) [69] were specifically reported in the context of end-to-end simulation of wireless systems in [70], [71] and for channel modeling in [72]–[75] for earlier applications to satellite communications.

The outlined prior work has focused on frequentist methods and has assumed the availability of uncorrupted data sets that are free from outliers. In this section, we explore the use of robust Bayesian learning to account for both outliers and model misspecification.

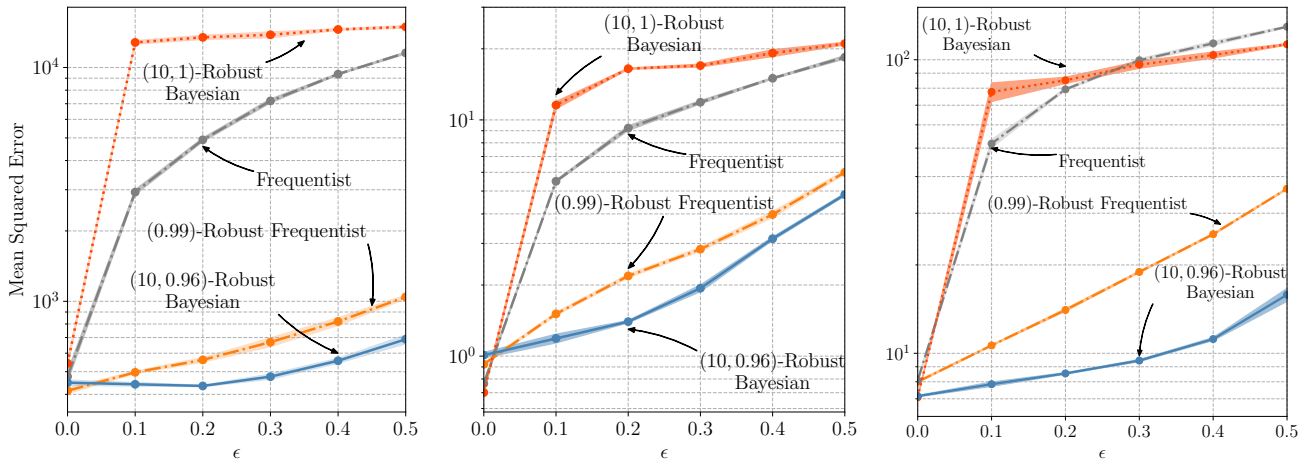
A. Problem Definition and Performance Metrics

Generative models are trained in an unsupervised manner by assuming the availability of a training set $\mathcal{D}$ of examples x corresponding to channel impulse responses. We focus on VAEs, i.e., on generative models with latent variables. VAEs comprise a parameterized *encoder* $q(h|x,\theta_e)$, mapping an input $x \in \mathcal{X}$ into a lower-dimensional latent vector $h \in \mathcal{H}$; as well as a parameterized *decoder* $p(x|h,\theta_d)$ that reconstructs the input sample $x \in \mathcal{X}$ from the latent representation $h \in \mathcal{H}$. Note that the vector of model parameters encompasses both encoding and decoding parameters as $\theta = (\theta_e, \theta_d)$.

Let us define as $p(h)$ a fixed *prior* distribution on the latent variables h. Once training is complete, samples x of channel responses can be generated from the model as follows. For frequentist learning, given the trained model θ^{freq}, one generates a sample $h \sim p(h)$ for the latent vector, and then produces a channel sample $x \sim p(x|h,\theta^{\text{freq}})$. For Bayesian learning, given the optimized distribution $q(\theta)$, we produce a random sample $\theta \sim q(\theta)$ and then generate channel sample $x \sim p(x|h,\theta_d)$. The role of the encoder $q(h|x,\theta_e)$ will be made clear in Section VI-C when discussing the training method.

According to the discussion in the previous paragraph, the channel distribution implemented by the model is given by

$$p(x) = \mathbb{E}_{p(h)}[p(x|h,\theta_d^{\text{freq}})] \tag{35}$$



(a) *SigfoxRural* (b) *UTSIndoor* (c) *UJIIndoor*

Fig. 7: Test mean squared error (33) as a function of the corruption level ϵ of the standard frequentist solution based on the log-loss, the robust frequentist solution based on the t-log-loss with $t = 0.99$, and the (m, t)-robust Bayesian learning with $m = 10$ and $t = \{1, 0.96\}$. As ϵ increases, the training data sets are increasingly affected by outliers.

for frequentist learning; and by

$$p(x) = \mathbb{E}_{p(h)q(\theta_d)}[p(x|h, \theta_d)] \tag{36}$$

for Bayesian learning. Note that the average is taken only over the latent vector $h \sim p(h)$ for frequentist learning; while in Bayesian learning the expectation is also taken over the optimized distribution $q(\theta_d)$ for the decoder's parameters θ_d.

To evaluate the performance of the generative model, we consider two different metrics accounting for accuracy and uncertainty quantification. Accuracy is measured by the "distance" between the target distribution $\nu(x)$ and the distribution $p(x)$ produced by the model. We measure the "distance" between $\nu(x)$ and $p(x)$ via the *maximum-mean discrepancy* (MMD) [76], which is defined as

$$\begin{aligned}\text{MMD}(p, \nu) =& \mathbb{E}_{x,x' \sim p(x)}[k(x, x')] + \mathbb{E}_{x,x' \sim \nu(x)}[k(x, x')] \\ & - 2\mathbb{E}_{x \sim \nu(x), x' \sim p(x)}[k(x, x')]\end{aligned} \tag{37}$$

where $k(x, x')$ is a positive definite kernel function. In the experiments reported below, we have approximated the MMD based on empirical averages. These are evaluated using samples from distribution $p(x)$, which are generated as explained above, as well as samples from the sampling distribution $\nu(x)$, i.e., examples from the training set $\mathcal{D}$. Moreover, we use the Gaussian kernel $k(x, x') = \mathcal{N}(\|x - x'\| \, |0, 1)$.

To evaluate the performance in terms of uncertainty quantification, we focus on the problem of *out-of-distribution (OOD) detection* (see, e.g., [77]). A well-calibrated model $p(x)$, when fed with an input x, should return a small value if x is an OOD sample, that is, if it has a low target distribution $\nu(x)$. To obtain a quantitative measure, we consider the task of distinguishing between samples drawn from the target distribution $\nu(x)$ and from the OOD distribution $\xi(x)$. Specifically, we adopt the model probability distribution $p(x)$ as the test statistic, classifying x as in-distribution (ID) if $p(x)$ is larger than some threshold γ and as OOD otherwise. As in ([78]), we take the area under the receiver operating characteristic curve (AUROC) score for this test as a measure of how distinguishable the two samples are. The AUROC metric is obtained by integrating the ROC traced by probability of detection versus probability of false alarm as the threshold γ is varied. A larger AUROC indicates that the model provides a better quantification of uncertainty, as reflected in its capacity to detect OOD samples against ID samples.

B. Data Set

We consider the simulation of the magnitudes of a frequency-selective channel response $x \in \mathbb{R}^{128}$ that mimics the target distribution $\nu(x)$ defined by the 3GPP TDL-A channel model distribution [79] with a delay spread of $\tau = 100$ ns. In particular we generate 1000 TDL-A channel responses using Matlab's 5G Toolbox [80]. *Outliers* are accounted for by constructing an ϵ-contaminated training set $\mathcal{D}$ that contains a fraction $\epsilon = 0.2$ of samples distributed according to the same channel model but with a larger delay spread $\tau = 300$ ns (see the top row in Fig. 8).

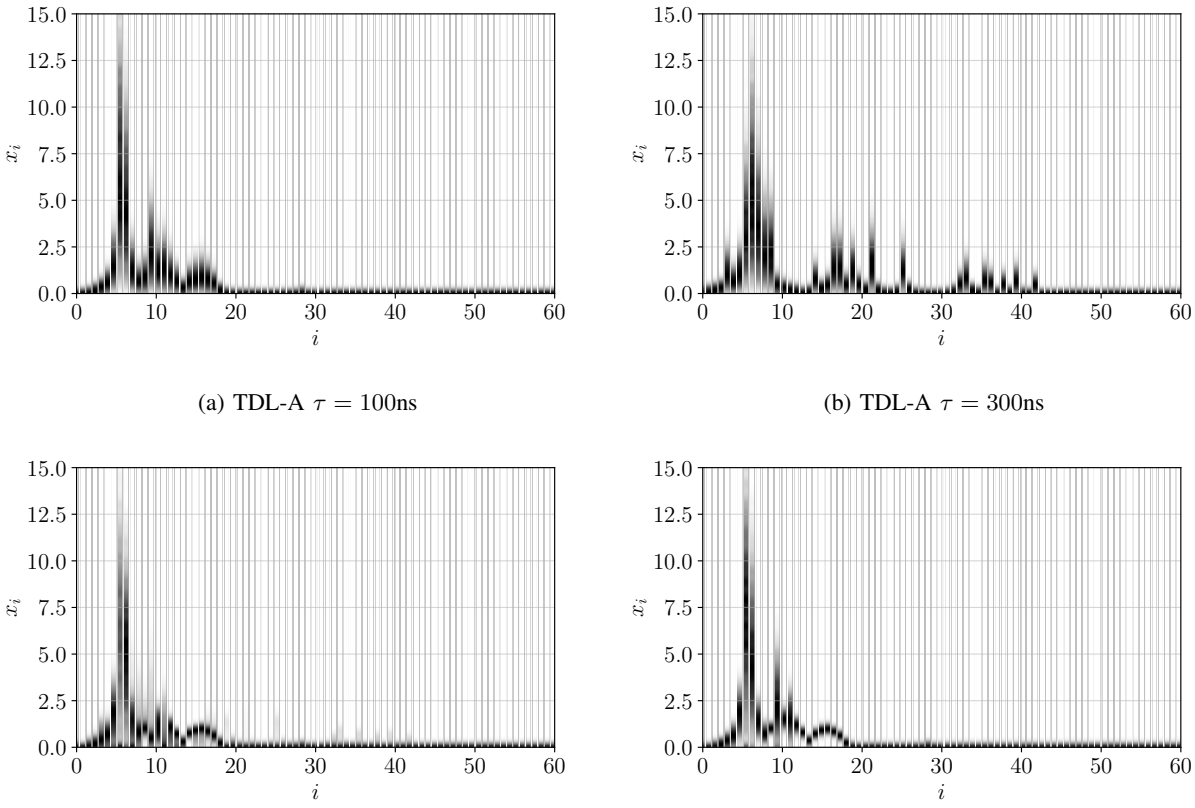
C. Implementation

For models with latent variables, the direct adoption of the log-loss generally yields intractable optimization problems (see, e.g., [17]). To address this problem, training of VAEs replaces the training loss (1) with the *variational lower bound*

$$\begin{aligned}\hat{\mathcal{L}}^{VAE}(\theta, \mathcal{D}) = & \sum_{x \in \mathcal{D}} \mathbb{E}_{p(h|x,\theta_e)}[\log(p(x|h, \theta_d)] \\ & - \sum_{x \in \mathcal{D}} \text{KL}(p(h|x, \theta_d)||p(h)),\end{aligned} \tag{38}$$

which involves the use of the encoder model $p(h|x, \theta_e)$. Accordingly, the frequentist training objective is modified as

$$\underset{\theta}{\text{minimize}} \, \hat{\mathcal{L}}_{VAE}(\theta, \mathcal{D}), \tag{39}$$



(a) TDL-A $\tau = 100$ns

(b) TDL-A $\tau = 300$ns

(c) Frequentist Learning

(d) $(4, 0.7)$-Robust Bayesian Learning

Fig. 8: The top row shows a sample of the magnitude for the TDL-A channel response given a delay spread $\tau = 100$ns in panel (a), while an outlier sample corresponding to the larger delay spread $\tau = 300$ ns is depicted in panel (b). The bottom row reports a sample from the trained model for frequentist learning in panel (c) and for $(4, 0.7)$-robust Bayesian learning in panel (d).

while Bayesian learning addresses the problem

$$\underset{q(\theta)}{\text{minimize}}\, \mathbb{E}_{q(\theta)} \left[\hat{\mathcal{L}}_{VAE}(\theta_e, \theta_d, \mathcal{D})\right] + \frac{1}{\beta}\text{KL}(q(\theta)||p(\theta)). \tag{40}$$

The robust free energy metrics are obtained in a similar manner, yielding the following formulation for (m, t)-robust Bayesian learning

$$\hat{\mathcal{L}}_t^{VAE}(\theta_1, \ldots, \theta_m, \mathcal{D}) = \sum_{x\in\mathcal{D}} \mathbb{E}_{p(h|x,\theta_e)} \log_t \left(\sum_{i=1}^{m} \frac{p(x|h, \theta_{d,i})}{m} \right) - \sum_{x\in\mathcal{D}} \text{KL}(p(h|x, \theta_d)||p(h)). \tag{41}$$

The prior latent variable distribution is $p(h) = \mathcal{N}(h|0, \mathbb{I}_5)$. We implement both the encoder and the decoder by using fully connected neural networks with a single hidden layer with 10 units, for a total of 6438 parameters. Specifically, the encoder distribution $p(h|x, \theta_e) = \mathcal{N}(h|\mu_{\theta_e}(x), \Sigma_{\theta_e}(x))$ has mean vector $\mu_{\theta_e}(x) \in \mathbb{R}^5$ and diagonal covariance matrix $\Sigma_{\theta_e}(x) \in \mathbb{R}^{5\times 5}$ obtained from the output of the neural network. The decoder $p(x|h, \theta_d) = \mathcal{N}(\hat{x}|\mu_{\theta_d}(h), \sigma\mathbb{I}_{128})$ has mean vector $\mu_{\theta_d}(h)$ obtained as the output of the neural network with a fixed variance value $\sigma = 0.1$. For Bayesian learning, we optimize distribution $q(\theta_d)$ as in the previous sections, while we consider a distribution $q(\theta_e)$ concentrated at a single vector θ_e. Ensembling during testing time is carried out with $m = 50$ samples.

D. Results

To start, in Figure 8 we illustrate a sample of the magnitude for the TDL-A channel response given a delay spread $\tau = 100$ ns in panel (a), while an outlier sample corresponding to the larger delay spread $\tau = 300$ ns is depicted in panel (b). The bottom row of Figure 8 reports a sample from the trained model for frequentist learning in panel (c) and for $(4, 0.7)$-robust Bayesian learning in panel (d). Visual inspection of the last two panels confirms that (m, t)-robust Bayesian learning can mitigate the effect of outliers as it reduces the spurious multipath components associated with larger delays.

For a numerical comparison, Figure 9 compares frequentist and $(4, t)$-robust Bayesian learning in terms of both accuracy – as measured by the MMD – and uncertainty quantification – as evaluated via the AUROC. For $t < 0.85$ robust Bayesian learning is confirmed to have the capacity to mitigate the effect of the outlying component, almost halving the MMD obtained

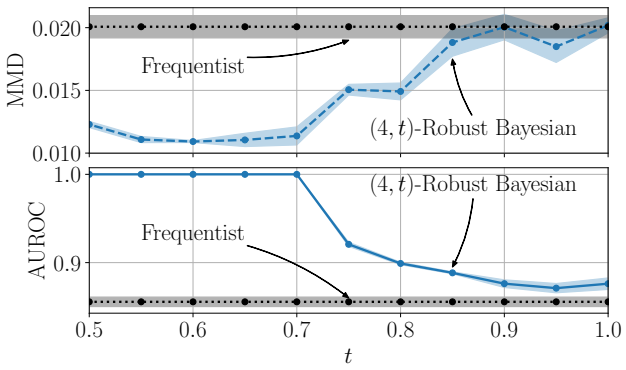


Fig. 9: Maximum mean discrepancy (MMD) and area under receiving operating curve (AUROC) for frequentist learning and $(4,t)$-robust Bayesian learning. Both models are trained on a corrupted data set with $(\epsilon = 0.2)$.

by frequentist learning. Furthermore, robust Bayesian learning has a superior uncertainty quantification performance, with gain increasing for decreasing values of t.

VII. Conclusion

This work has focused on the problem of ensuring that AI models trained for wireless communications satisfy reliability and robustness requirements. We have specifically addressed two important problems: model misspecification, arising from limitations on the available knowledge about the problem and on the complexity of the AI models that can be implemented on network devices; and outliers, which cause a mismatch between training and testing conditions. We have argued that standard frequentist learning, as well as Bayesian learning, are not designed to address these requirements, and we have explored the application of *robust Bayesian learning* to achieve robustness to model misspecification and to the presence of outliers in the training data set. Robust Bayesian learning has been shown to consistently provide better accuracy and uncertainty estimation capabilities in a range of important wireless communication problems. These results motivate a range of extension of robust Bayesian learning and applications. For instance, the integration of robust Bayesian learning to the meta-learning framework, in order to enable robust and sample effective learning, or the application of robust Bayesian learning to higher layers of the protocol stack as a tool to empower semantic communication.

References

[1] O. Simeone, "A very brief introduction to machine learning with applications to communication systems," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 648–664, 2018.

[2] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3072–3108, 2019.

[3] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[4] Y. Jiang, H. Kim, H. Asnani, S. Kannan, S. Oh, and P. Viswanath, "Learn codes: Inventing low-latency codes via recurrent neural networks," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 207–216, 2020.

[5] H. Kim, Y. Jiang, S. Kannan, S. Oh, and P. Viswanath, "Deepcode: Feedback codes via deep learning," *Advances in neural information processing systems*, vol. 31, 2018.

[6] S. Park, O. Simeone, and J. Kang, "Meta-learning to communicate: Fast end-to-end training for fading channels," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5075–5079, IEEE, 2020.

[7] S. Park, H. Jang, O. Simeone, and J. Kang, "Learning to demodulate from few pilots via offline and online meta-learning," *IEEE Transactions on Signal Processing*, vol. 69, pp. 226–239, 2020.

[8] O. Simeone, S. Park, and J. Kang, "From learning to meta-learning: Reduced training overhead and complexity for communication systems," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*, pp. 1–5, IEEE, 2020.

[9] Y. Yuan, G. Zheng, K.-K. Wong, B. Ottersten, and Z.-Q. Luo, "Transfer learning and meta learning-based fast downlink beamforming adaptation," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1742–1755, 2020.

[10] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International Conference on Machine Learning*, pp. 1321–1330, PMLR, 2017.

[11] K. M. Cohen, S. Park, O. Simeone, and S. Shamai, "Learning to learn to demodulate with uncertainty quantification via Bayesian meta-learning," *arXiv preprint arXiv:2108.00785*, 2021.

[12] P. H. Masur and J. H. Reed, "Artificial intelligence in Open Radio Access Network," *arXiv preprint arXiv:2104.09445*, 2021.

[13] D. J. C. MacKay, *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

[14] K. Osawa, S. Swaroop, M. E. E. Khan, A. Jain, R. Eschenhagen, R. E. Turner, and R. Yokota, "Practical deep learning with Bayesian principles," *Advances in neural information processing systems*, vol. 32, 2019.

[15] I. Nikoloska and O. Simeone, "BAMLD: Bayesian Active Meta-Learning by Disagreement," *arXiv preprint arXiv:2110.09943*, 2021.

[16] D. Madigan, A. E. Raftery, C. Volinsky, and J. Hoeting, "Bayesian model averaging," in *Proceedings of the AAAI Workshop on Integrating Multiple Learned Models, Portland, OR*, pp. 77–83, 1996.

[17] O. Simeone, *Machine Learning for Engineers*. Cambridge university press, 2022.

[18] N. Zilberstein, C. Dick, R. Doost-Mohammady, A. Sabharwal, and S. Segarra, "Annealed Langevin dynamics for massive mimo detection," *arXiv preprint arXiv:2205.05776*, 2022.

[19] R. Martinez-Cantin, K. Tee, and M. McCourt, "Practical Bayesian optimization in the presence of outliers," in *International Conference on Artificial Intelligence and Statistics*, pp. 1722–1731, PMLR, 2018.

[20] P. Izmailov, P. Nicholson, S. Lotfi, and A. G. Wilson, "Dangers of bayesian model averaging under covariate shift," *Advances in Neural Information Processing Systems*, vol. 34, pp. 3309–3322, 2021.

[21] P. Domingos, "Bayesian averaging of classifiers and the overfitting problem," in *ICML*, vol. 747, pp. 223–230, 2000.

[22] A. Masegosa, "Learning under model misspecification: Applications to variational and ensemble methods," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5479–5491, 2020.

[23] W. R. Morningstar, A. A. Alemi, and J. V. Dillon, "PAC^m-Bayes: narrowing the empirical risk gap in the misspecified Bayesian regime," *arXiv preprint arXiv:2010.09629*, 2020.

[24] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?," *Advances in neural information processing systems*, vol. 30, 2017.

[25] S. T. Jose and O. Simeone, "Free energy minimization: A unified framework for modeling, inference, learning, and optimization [lecture notes]," *IEEE Signal Processing Magazine*, vol. 38, no. 2, pp. 120–125, 2021.

[26] O. Catoni, "A PAC-Bayesian approach to adaptive classification," *preprint*, vol. 840, 2003.

[27] P. Alquier, "User-friendly introduction to PAC-Bayes bounds," *arXiv preprint arXiv:2110.11216*, 2021.

[28] J. Knoblauch, J. Jewson, and T. Damoulas, "Generalized variational inference: Three arguments for deriving new posteriors," *arXiv preprint arXiv:1904.02063*, 2019.

[29] T. Gneiting and A. E. Raftery, "Strictly proper scoring rules, prediction, and estimation," *Journal of the American statistical Association*, vol. 102, no. 477, pp. 359–378, 2007.

[30] A. Basu, I. R. Harris, N. L. Hjort, and M. Jones, "Robust and efficient estimation by minimising a density power divergence," *Biometrika*, vol. 85, no. 3, pp. 549–559, 1998.

[31] A. Ghosh and A. Basu, "Robust Bayes estimation using the density power divergence," *Annals of the Institute of Statistical Mathematics*, vol. 68, no. 2, pp. 413–437, 2016.

[32] H. Fujisawa and S. Eguchi, "Robust parameter estimation with a small bias against heavy contamination," *Journal of Multivariate Analysis*, vol. 99, no. 9, pp. 2053–2081, 2008.

[33] T. Nakagawa and S. Hashimoto, "Robust Bayesian inference via γ-divergence," *Communications in Statistics-Theory and Methods*, vol. 49, no. 2, pp. 343–360, 2020.

[34] M. Zecchin, S. Park, O. Simeone, M. Kountouris, and D. Gesbert, "Robust PAC^m: Training ensemble models under model misspecification and outliers," *arXiv preprint arXiv:2203.01859*, 2022.

[35] A. Fawzy, H. M. Mokhtar, and O. Hegazy, "Outliers detection and classification in wireless sensor networks," *Egyptian Informatics Journal*, vol. 14, no. 2, pp. 157–164, 2013.

[36] R. Jin, Z. Che, H. Xu, Z. Wang, and L. Wang, "An RSSI-based localization algorithm for outliers suppression in wireless sensor networks," *Wireless Networks*, vol. 21, no. 8, pp. 2561–2569, 2015.

[37] S. Kalyani and K. Giridhar, "OFDM channel estimation in the presence of NBI and the effect of misspecified NBI model," in *2007 IEEE 8th Workshop on Signal Processing Advances in Wireless Communications*, pp. 1–5, IEEE, 2007.

[38] Y.-C. Liang, K.-C. Chen, G. Y. Li, and P. Mahonen, "Cognitive radio networking and communications: An overview," *IEEE transactions on vehicular technology*, vol. 60, no. 7, pp. 3386–3407, 2011.

[39] E. S. Lohan, J. Torres-Sospedra, H. Leppäkoski, P. Richter, Z. Peng, and J. Huerta, "Wi-Fi crowdsourced fingerprinting dataset for indoor positioning," *Data*, vol. 2, no. 4, p. 32, 2017.

[40] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods," *Machine Learning*, vol. 110, no. 3, pp. 457–506, 2021.

[41] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, and D. Tran, "Measuring calibration in deep learning.," in *CVPR Workshops*, vol. 2, 2019.

[42] S. Theodoridis, *Machine learning: a Bayesian and optimization perspective*. Academic Press, 2015.

[43] O. Catoni, "PAC-bayesian supervised classification: the thermodynamics of statistical learning," *arXiv preprint arXiv:0712.0248*, 2007.

[44] S. Nakajima, K. Watanabe, and M. Sugiyama, *Variational Bayesian learning theory*. Cambridge University Press, 2019.

[45] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 859–877, 2017.

[46] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom, and J. Frolik, "Identification and mitigation of non-line-of-sight conditions using received signal strength," in *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 667–674, IEEE, 2013.

[47] B.-E. Chérief-Abdellatif and P. Alquier, "Mmd-bayes: Robust bayesian estimation via maximum mean discrepancy," in *Symposium on Advances in Approximate Bayesian Inference*, pp. 1–21, PMLR, 2020.

[48] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[49] G. Carbone, M. Wicker, L. Laurenti, A. Patane, L. Bortolussi, and G. Sanguinetti, "Robustness of bayesian neural networks to gradient-based attacks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15602–15613, 2020.

[50] M. Wicker, L. Laurenti, A. Patane, Z. Chen, Z. Zhang, and M. Kwiatkowska, "Bayesian inference with certifiable adversarial robustness," in *International Conference on Artificial Intelligence and Statistics*, pp. 2431–2439, PMLR, 2021.

[51] D. Alistarh, Z. Allen-Zhu, and J. Li, "Byzantine stochastic gradient descent," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[52] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[53] S. Minsker, S. Srivastava, L. Lin, and D. Dunson, "Scalable and robust bayesian inference via the median posterior," in *International conference on machine learning*, pp. 1656–1664, PMLR, 2014.

[54] E. T. Jaynes, *Probability theory: The logic of science*. Cambridge university press, 2003.

[55] J. Jewson, J. Q. Smith, and C. Holmes, "Principles of Bayesian inference using general divergence criteria," *Entropy*, vol. 20, no. 6, p. 442, 2018.

[56] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[57] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*, pp. 213–226, Springer, 2016.

[58] R. Zhou, F. Liu, and C. W. Gravelle, "Deep learning for modulation recognition: A survey with a demonstration," *IEEE Access*, vol. 8, pp. 67366–67376, 2020.

[59] M. H. DeGroot and S. E. Fienberg, "The comparison and evaluation of forecasters," *Journal of the Royal Statistical Society: Series D (The Statistician)*, vol. 32, no. 1-2, pp. 12–22, 1983.

[60] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[61] R. Mautz, "Overview of current indoor positioning systems," *Geodezija ir kartografija*, vol. 35, no. 1, pp. 18–22, 2009.

[62] M. Aernouts, R. Berkvens, K. Van Vlaenderen, and M. Weyn, "Sigfox and LoRaWAN datasets for fingerprint localization in large urban and rural areas," *Data*, vol. 3, no. 2, p. 13, 2018.

[63] G. Pecoraro, S. Di Domenico, E. Cianca, and M. De Sanctis, "CSI-based fingerprinting for indoor localization using LTE signals," *EURASIP Journal on Advances in Signal Processing*, vol. 2018, no. 1, pp. 1–18, 2018.

[64] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent neural networks for accurate RSSI indoor localization," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10639–10651, 2019.

[65] R. S. Sinha and S.-H. Hwang, "Comparison of CNN applications for RSSI-based fingerprint indoor localization," *Electronics*, vol. 8, no. 9, p. 989, 2019.

[66] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[67] X. Song, X. Fan, C. Xiang, Q. Ye, L. Liu, Z. Wang, X. He, N. Yang, and G. Fang, "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.

[68] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, and J. Huerta, "Ujiindoorloc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *2014 international conference on indoor positioning and indoor navigation (IPIN)*, pp. 261–270, IEEE, 2014.

[69] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[70] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pp. 298–303, IEEE, 2018.

[71] H. Ye, L. Liang, G. Y. Li, and B.-H. Juang, "Deep learning-based end-to-end wireless communication systems with conditional GANs as unknown channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3133–3143, 2020.

[72] T. J. O'Shea, T. Roy, and N. West, "Approximating the void: Learning stochastic channel models from observation with variational generative adversarial networks," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, pp. 681–686, IEEE, 2019.

[73] T. Orekondy, A. Behboodi, and J. B. Soriaga, "MIMO-GAN: Generative MIMO channel modeling," *arXiv preprint arXiv:2203.08588*, 2022.

[74] Y. Yang, Y. Li, W. Zhang, F. Qin, P. Zhu, and C.-X. Wang, "Generative-adversarial-network-based wireless channel modeling: Challenges and opportunities," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 22–27, 2019.

[75] M. Ibnkahla, "Applications of neural networks to digital communications–a survey," *Signal processing*, vol. 80, no. 7, pp. 1185–1215, 2000.

[76] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola, "A kernel method for the two-sample-problem," *Advances in neural information processing systems*, vol. 19, 2006.

[77] E. Daxberger and J. M. Hernández-Lobato, "Bayesian variational autoencoders for unsupervised out-of-distribution detection," *arXiv preprint arXiv:1912.05651*, 2019.

[78] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[79] D. 3rd Generation Partnership Project (3GPP), "Study on channel model for frequencies from 0.5 to 100 GHz," *3GPP TR 38.901*, 2020.

[80] Matlab, *5G Toolbox*. Natick, Massachusetts: The MathWorks Inc., 2021.