The International Journal of Engineering and Science (IJES)
|| Volume || 14 || Issue 7 || July || Pages || PP 51-59 || 2025 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

Automatic Modulation Classification with Bayesian Neural Network

Tsung-Cheng Wu
Department of Intelligent Network Technology, I-Shou University, Taiwan

----------ABSTRACT----------

This study applies Bayesian learning techniques, specifically Variational Inference (VI) and Monte Carlo Dropout (MC Dropout) to Automatic Modulation Classification (AMC). Both methods are built upon a Long Short-Term Memory (LSTM) framework and are capable of detecting certain out-of-domain (OOD) or novel modulations through threshold-based decision making. This paper illustrates the framework's new observation to quantify uncertainty and assess whether predictions belong to novel classes. The results indicate that the proposed relative frequency of confidence scores assigned to novel modulations are influenced by their proximity to the trained non-novel classes. Specifically, novel signals that closely resemble known modulation types tend to be assigned high confidence scores by the classifier. It is essential to train only on the necessary modulation classes. Training on an excessive number of classes can degrade performance of novel modulation detection and increase computational burden.

Keywords: *Automatic Modulation Classification, LSTM, Variational Inference, Monte Carlo Dropout*

Date of Submission: 11-07-2025 Date of acceptance: 24-07-2025

I. Introduction

Automatic modulation classification (AMC) is a technology used in communication systems to manage and monitor whether a signal source is subject to external interference or malicious attacks, including spectrum awareness, adaptive transmission switching to a more effective modulation method when frequency is congested, spectrum allocation and optimization efficiency based on different signal characteristics, and interference avoidance. The methods of automatic modulation classification can be summarized into two points: likelihood-based and feature-based. Likelihood-based has traditional likelihood optimization, which regards modulation classification as a multiple hypothesis testing problem and requires the best solution, but it will lead to high computational complexity. The second is feature-based methods, which have been proposed in the past, including higher order moments, higher order cumulants, cyclostationary features, etc. In the first stage, feature-based methods extract some feature signals from the received data, and in the second stage, use classical classifiers to determine the modulation category [2].

With the emergence of deep learning, data-driven machine learning models have demonstrated superior performance over conventional methods in AMC tasks [1]. For example, KNN (K-nearest neighbor, KNN) and SVM (support vector machine, SVM) replace the second-stage classifier, and various other machine learning and deep learning methods are used to solve such multi-classification problems. These methods are called classical or frequentist learning, which belongs to classical statistics. These frequentist learning approaches, grounded in classical statistics, treat model parameters as fixed but unknown quantities.

Recently, Bayesian deep learning has gained attention for its ability to estimate posterior distributions over network weights, enabling the quantification of model uncertainty [7]. Treating the model output as a random variable is called probabilistic learning, and treating the neural network parameters as random variables is called Bayesian learning. Bayesian learning has been proven to capture epistemic uncertainty (model-related) and aleatoric uncertainty(data-related).

Aleatoric uncertainty refers to the intrinsic uncertainty present in data, often caused by unavoidable noise, environmental interference, or adversarial attacks. This type of uncertainty stems from the inherent randomness of the system, meaning that even identical inputs may yield different outputs. To address this, model outputs can be treated as random variables governed by a probability distribution, known as the predictive probability $p(y|x, \theta)$.

In contrast, epistemic uncertainty arises from the model itself, such as uncertainty in model parameters or insufficient training data. Training with a limited data set will result in an incomplete understanding of the real situation. This type of uncertainty reflects the model's lack of knowledge and can be captured by the posterior

distribution $p(\theta|D)$.

Bayesian learning is believed to be able to capture system and data uncertainties, and can face the uncertainty of out-of-domain modulation classes, or so-called novel classes. Compared with classical statistical (i.e. frequentist) learning methods, the weights of its neural network obtained through training are scalar values, and when predicting out-of-domain modulations, they will definitely be mistakenly inferred as in-domain modulations.

The contributions of this paper are: 1). it demonstrates that Bayesian LSTM-based frameworks using VI and MC Dropout can achieve satisfactory accuracy on raw measurement data, providing a foundation for practical edge computing applications; and 2). it illustrates the framework's new observation to quantify uncertainty and assess whether predictions belong to novel classes.

II. Bayesian Learning

Bayesian learning has demonstrated the capability to quantify both aleatoric and epistemic uncertainties, enabling not only uncertainty-aware predictions but also the estimation of prediction confidence for novel modulation inputs. Traditional deep learning, as a form of frequentist learning, typically forces a prediction among pre-defined classes, even for inputs that do not belong to any trained category, thus failing to express uncertainty effectively. Applications of Bayesian methods in AMC show promise recently. For example, [4] treats raw input data as a time series and employs an LSTM architecture with variational inference and Laplace approximation to detect a novel OQPSK modulation class. This paper adopts the method of directly treating the original data I/Q channel data as time-series -like representations, and applies a LSTM-based framework to perform Variational Inference (VI) and Monte Carlo (MC) Dropout for inference and novel-class uncertainty detection.

The VI is a method of approximating the posterior probability of a connection weight using a group of simple probability distributions. The parameters of the probability distribution are called variational parameters. It is hoped that the variational distribution $p(\theta|D)$ is as close as possible to the posterior distribution $p(\theta|D)$ or to minimize the KL divergence between the two distributions. The KL divergence is defined as

$$KL[q_\lambda(\theta)||p(\theta|D)] = log[p(D)] - \int q_\lambda(\theta) \cdot log\left[\frac{p(\theta,\ D)}{q_\lambda(\theta)}\right] d\theta$$

By adjusting the variational parameters to find the appropriate variational distribution $q_\lambda(\theta)$, variational inference becomes an optimization problem with KL divergence as the loss function.

According to [5], the loss function $KL[q_\lambda(\theta)||p(\theta|D)]$ can be simplified to the following formula, that is,

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}}\{KL[q_\lambda(\theta)||p(\theta)] - E_{\theta\sim q_\lambda}[log(p(D|\theta))]\}$$

where

$$E_{\theta\sim q_\lambda}[log(p(D|\theta))] = \int q_\lambda(\theta) \cdot log[p(D|\theta)]\, d\theta$$

In this way, the optimal variational parameter λ^* is found to minimize the loss function. $q_\lambda(\theta)$ is selected to be a Gaussian distribution, and the Gaussian variational distribution is defined by two parameters: mean and standard deviation, i.e., $\lambda = (\mu, \sigma)$, which is the variational parameter. VI adjusts the variational parameter $\lambda = (\mu, \sigma)$ so that $q_\lambda(\theta)$ is as close as possible to the true posterior probability $p(\theta|D)$. It is also necessary to define the prior distribution. A common choice is to use the standard normal $N(0, 1)$ as the prior probability. Therefore, this type of neural network does not learn a single weight value w, but learns two parameters of the weight distribution, i.e., the mean and standard deviation of the Gaussian distribution.

MC Dropout treats each neural network weight as a discrete random variable following a Bernoulli distribution. At each forward pass, a weight is sampled: a value of 0 indicates the connection is dropped (i.e., deactivated), while a value of 1 retains the connection. MC Dropout is also used in the training and prediction stages.

The modulation class is predicted by the trained model. Let $\boldsymbol{Y} = \{c_1, c_2, \ldots, c_K\}$ denote the finite set of class labels, and consider a labeled test dataset $\boldsymbol{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \boldsymbol{X}$ represents the input sample vector and $y_i \in \boldsymbol{Y}$ denotes the associated label.

The predicted modulation class is

$$\hat{y} = \operatorname{argmax}_{c_k \in \boldsymbol{Y}} \hat{P}(Y = c_k|x)$$

where *y* denotes class label, $\hat{P}(Y = c_k|x)$ denotes predicted probability, and $\hat{y}$ is the predicted class label.

In Bayesian learning, the predictive distribution is obtained by marginalizing over the posterior distribution of the model parameters. Perform Monte Carlo sampling over the same input. By drawing Monte Carlo samples from the conditional probability distribution *T* times regarding to the same input *x*, a set of $\{\hat{P}^{(t)}(Y|x)\}_{t=1}^{T}$ is obtained. Prediction probability with respect to class c_k is given by:

$$\hat{P}(Y = c_k|x) \approx \frac{1}{T}\sum_{t=1}^{T} \hat{P}^{(t)}(Y = c_k|x)$$

where $\hat{P}(Y = c_k|x)$ denotes the *t*-th sampling prediction probability.

There are two approaches to implementing VI method within the LSTM architecture: one involves applying VI to the weights of the LSTM neural network itself, and the other applies VI to the fully connected (dense) layers following the LSTM. In this paper, we adopt the latter approach, utilizing the LSTM as a frontend for feature extraction, while the subsequent dense layers with VI are employed for classification.

III. Prediction Results

In this paper, the actual measured waveform dataset RadioML2018.01a [6] is employed to train and test. The dataset comprises 26 Signal-to-Noise Ratio (SNR) levels ranging from - 20 dB to 30 dB in 2 dB increments, encompassing 24 modulation types. Each SNR level contains 1024 waveform samples represented as I/Q channel pairs. This work selects eight modulation types from the RadioML2018.01a dataset, namely 4ASK, BPSK, QPSK, 16PSK, 16QAM, FM, AM-DSB-WC, and 32APSK. The input waveform samples are interpreted as 1024 × 2 matrices, where each element corresponds to a sampled point from the I and Q channels, respectively. Due to the varying range and polarity of the values in the two channels, all data are normalized to the range [0, 1] as a preprocessing step. Waveform samples with SNRs between 24 and 30 dB are selected for training and evaluation. A stratified split is applied allocating 80% of the data of 131072 samples for training and 20% for testing.

The VI method adopts LSTM architecture followed by Dense Layer with VI approach. Using the probabilistic tensorflow package, each layer in the Dense layer is parameterized as a probability distribution. The KL divergence function is also provided by the package as shown Table 1.

In this paper, Bayesian learning evaluates models through Monte Carlo Sampling and tests data sets.

Through various configurations, we observed that setting a high number of units in the LSTM tends to cause the loss to escalate dramatically, often resulting in NaN values. The input format to the LSTM is structured as (number of samples, number of time steps, number of features), with the time step set to 1024 and the number of features fixed at 2.

The model LSTM adding Dense with VI is trained over 100 epochs, achieving a training accuracy of 89.9% and a testing accuracy of 89.8%. The training data accuracy only represents the averaged results across all categories. The confusion matrix concentrates on the accuracy of each individual modulation type of trained model for multi-class classification problems.

As shown in Figure 1, the confusion matrix highlights that QPSK performs the worst comparison to the other modulation types and 16QAM the next. Confusion matrix of LSTM with VI is shown in Figure 1.

Table 1：LSTM VI Model Structure

Layer	Parameters
LSTM	100, return_sequences=True
LSTM	100, return_sequences=True
Flatten()	
DenseFlipout	500, kernel_divergence_fn=lambda q, p, _: tfd.kl_divergence(q, p) / tf.cast(batch_size, tf.float32)
Dropout	Drop rate = 0.3
DenseFlipout	256, kernel_divergence_fn=lambda q, p, _: tfd.kl_divergence(q, p) / tf.cast(batch_size, tf.float32)
Dropout	Drop rate = 0.3
DenseFlipout	100, kernel_divergence_fn=lambda q, p, _: tfd.kl_divergence(q, p) / tf.cast(batch_size, tf.float32)
DenseFlipout	units = 8, activation = 'softmax'

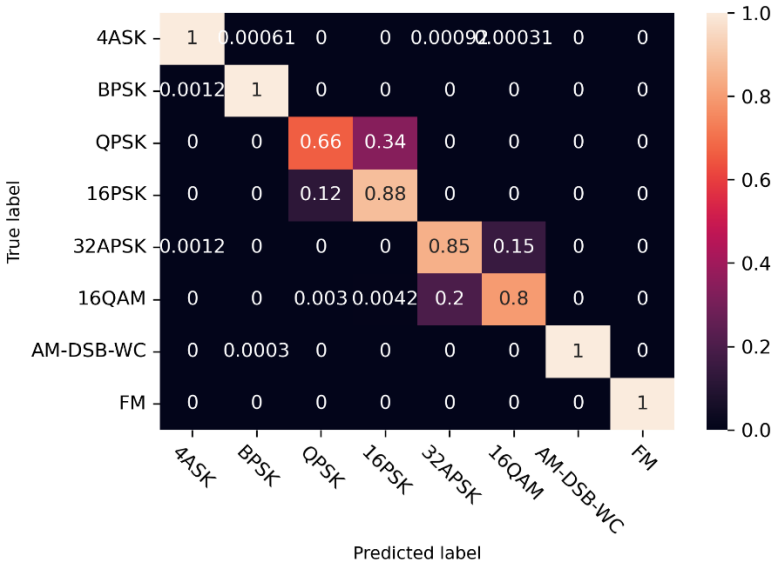

Figure 1: Confusion matrix of LSTM with VI.

For MC Dropout, the same LSTM and Dense (without VI) architecture are used, but all standard dropout layers are replaced with MC Dropout layers, each with a dropout rate of 0.4. This approach yields a training accuracy of 97.57% and a testing accuracy of 91.34% after 60 epochs. As shown in Figure 2, the confusion matrix highlights that 32APSK performs the worst comparison to the other modulation types and 16QAM the next. Confusion matrix of LSTM with VI is shown in Figure 1. Confusion matrix of LSTM with MC Dropout is shown in Figure 2.

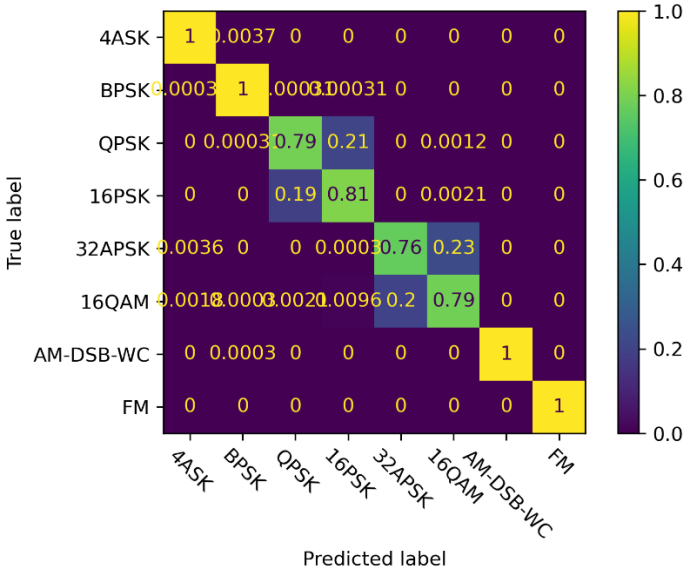

Figure 2: Confusion matrix of LSTM with MC Dropout.

To evaluate model robustness under noisy conditions, waveforms with SNR levels from –20 dB to 22 dB are selected for testing (not overlapping with training data). At low SNR levels as shown in Figure 3, the accuracy is indeed just over 12%. However, as the SNR exceeded 10 dB, the accuracy of all two models approached 90%, demonstrating their robustness at higher SNR levels.

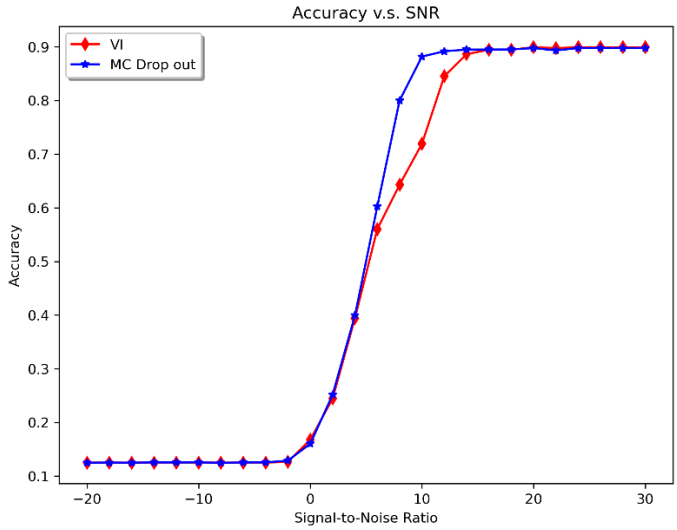

Figure 3: Accuracy of LSTM with VI and MC Dropout for SNR= -20dB to 30 dB.

IV. The Proposed Uncertainty Measurement

In prior studies, repeated predictions on the same waveform sample are used to compute the average probability across all classes, resulting in a class probability. In this work, we propose using relative frequency of confidence score as an alternative criterion for determining modulation class at a given SNR level. Specifically, we calculate the number of times each class is inferred and divide it by the total number of inference runs, resulting in a relative frequency. In this study, we set T=100 where T denotes the number of repeated estimations for a single input, and N=4096 where N denotes number of samples.

Prediction probability can quantify arbitrary uncertainty, that is, the output is random, and the same input value will produce different outputs. The maximum output of prediction probability in (1) is called confidence score denoted as Conf(x). We use confidence score to observe whether it can detect the uncertainty of novel modulation.

The empirical prediction frequency (i.e., relative frequency of predicted class assignments) is then expressed as:

$$\hat{P}(\hat{Y} = c_k) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\{\text{argmax}_{c_k \in Y}\hat{P}(Y = c_k|x_i)=c_k\}}$$

where $\hat{P}(\hat{Y} = c_k)$ represents the model's estimated output for class c_k, and $\mathbf{1}_{\{\}}$ is the indicator function, returning 1 when the condition holds and 0 otherwise. This formulation captures how often class c_k is predicted across a dataset of N samples.

In this paper, confidence score for uncertainty measurement are performed over T sampling and N test data.

The confidence score is given by

$$\text{Conf}(x) = \max_{c_k \in Y}\hat{P}(Y = c_k|x) \qquad (1)$$

Average confidence score is given by

$$\text{Conf}_{avg}(x) = \frac{1}{T}\sum_{t=1}^{T}\text{Conf}^{(t)}(x) \qquad (2)$$

where $\text{Conf}^{(t)}(x) = \max_{c_k \in Y}\hat{P}^{(t)}(Y = c_k|x)$.

The proposed relative frequency of decision-making through $\text{Conf}_{avg}(x)$ is given by

$$\frac{1}{N}\sum_{i=1}^{N}\mathbf{1}_{\{\text{Conf}_{avg}(x_i) > c_{th}\}} \qquad (3)$$

where C_{th} denotes threshold of decision-making novel class, and $\mathbf{1}_{\{\}}$ is the indicator function, returning 1 when the condition holds and 0 otherwise, and N is the total number of samples.

4.1 VI prediction

It is new point to use boxplot observing VI prediction confidence. A boxplot, also known as a box-and-whisker plot, is a statistical visualization that summarizes the distribution of a dataset. The box represents the range between the first quartile (Q1) and third quartile (Q3) or IQR. The line inside the box indicates the median (Q2) of the dataset. The whiskers extend from Q1 and Q3 to cover data points within 1.5 times the IQR.

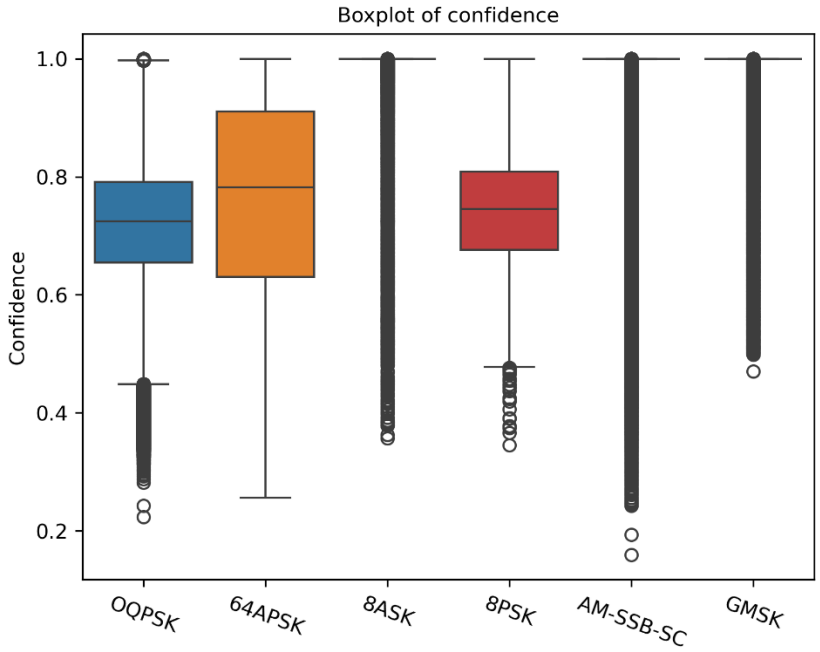

Figure 4: A Boxplot for confidence of novel classes prediction.

In Figure 4, VI prediction confidence for novel classes shows a box very close to 1.0, indicating high confidence. It can be seen that the median (Q2) of the novel classes is 50%, and the horizontal line in the box (representing the median (i.e. 50%)). It can be found that the OQPSK and 8PSK boxes (showing the IQ3-IQ1 range) is lower than 0.8, and the AM-SSB-WC, 8ASK, and GMSK have no boxes.

Figure 4 shows that some novel modulation predictions have high confidence in the traditional constellation form of ASK, PSK, and QAM. On the contrary, Figure 4 shows that 8PSK and OQPSK modulations have always maintained low confidence.

The descriptive statistics include: count stands for number of non-null entries, mean stands for average, std stands for standard deviation, min / max stands for minimum and maximum, percentiles stand for 25th, 50th (median), and 75th by default, respectively. Descriptive statistics of novel classes for VI is shown in Table 2.

Table 2: Descriptive statistics for novel classes: VI

statistics	64APSK	GMSK	AM-SSB-SC	8PSK	8ASK	OQPSK
count	409600	409600	409600	409600	409600	409600
mean	0.769	0.996	0.983	0.739	0.999	0.723
std	0.154	0.026	0.074	0.099	0.017	0.118
min	0.256	0.470	0.159	0.345	0.357	0.223
25%	0.630	0.999	1.0	0.676	1.0	0.654
50%	0.781	0.999	1.0	0.745	1.0	0.724
75%	0.910	1.0	1.0	0.808	1.0	0.791
max	1.0	1.0	1.0	0.999	1.0	1.0

A critical value is set to clearly distinguish the difference between novel and non-novel modulation. When the predicted output is higher than this critical value, it means that the predicted result can be trusted. If it exceeds the threshold value, it can be inferred to be a non-novel modulation. When the predicted confidence is lower than this critical value, it means that the prediction result is not confident enough and may be wrong, or the prediction result is novel classes. The prediction will output the highest confidence of the training class.

The Bayesian VI models exhibit a reliable ability to distinguish in-domain samples from novel modulations when combined with threshold-based confidence decision. The 64APSK, 8PSK, and OQPSK novel classes have

smaller average confidence than value of 0.8. Threshold can be used to make decision for novel class. Average confidence of prediction novel classes (2) is shown in Table 3.

Table 3: Average confidence: VI

64APSK	GMSK	AM-SSB-SC	8PSK	8ASK	OQPSK
0.769	0.996	0.983	0.739	0.999	0.723

Prediction novel classes can use relative frequency (3) of confidence exceeds C_{th} for VI is shown in Table 4. When the confidence threshold is set from 0.8 to 0.95, the relative frequency of prediction after calculating 4096 samples is shown in Table 4. The 64APSK means that the prediction of 64APSK input, and the frequency is 0.837 x 100%, which means that 1585 samples out of the 4096 input samples are predicted to be any of non-novel classes.

In Table 4, less than 1% of the 4096 samples of 8PSK and OQPSK will be predicted as novel classes. In Table 4, two novel classes of AM-SSB-SC and 8ASK result in 99% at confidence threshold=0.8. The two classes cannot be decided by threshold-decision as novel classes.

Table 4: Relative frequency of confidence exceed C_{th}: VI

	C_{th} =0.8	C_{th} =0.85	C_{th} =0.9	C_{th} =0.95
GMSK	0.999	0.999	0.996	0.985
AM-SSB-SC	0.972	0.960	0.938	0.903
OQPSK	0.065	0.046	0.025	0.008
64APSK	0.387	0.272	0.134	0.036
8ASK	0.999	0.999	0.998	0.995
8PSK	0.025	0.011	0.005	0.001

4.2 MC Dropout prediction

The descriptive statistics include: count stands for number of non-null entries, mean stands for average, std stands for standard deviation, min / max stands for minimum and maximum, percentiles stand for 25th, 50th (median), and 75th by default, respectively. Descriptive statistics of novel classes for MC Dropout is shown in Table 5. Observe together the boxplot of MC Dropout in Figure 5, and descriptive statistics in Table 5.

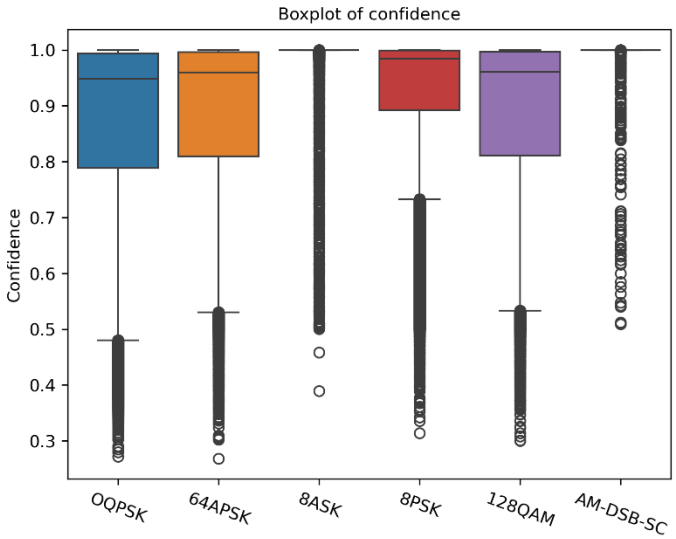

Figure 5: A Boxplot of confidence of novel classes prediction.

Median of confidence for novel classes shown in Table 5 all exceed 0.95. Compared to Table 2 for VI, median of confidence for novel classes 64APSK, 8PSK, and OQPSK do not exceed 0.8. The threshold-decision on novel class for VI model is possible.

Table 5 Descriptive statistics for novel classes: MC Dropout

statistics	64APSK	GMSK	AM-SSB-SC	AM-DSB-SC	128QAM	8PSK	8ASK	OQPSK
count	409600	409600	409600	409600	409600	409600	409600	409600
mean	0.886	0.986	0.996	0.999	0.886	0.918	0.999	0.873
std	0.143	0.059	0.033	0.003	0.143	0.126	0.008	0.152
min	0.268	0.347	0.342	0.508	0.30	0.314	0.389	0.271
25%	0.809	0.999	1.0	1.0	0.81	0.892	1.00	0.788
50%	0.959	1.0	1.0	1.0	0.960	0.984	1.00	0.948
75%	0.996	1.0	1.0	1.0	0.996	0.998	1.00	0.993
max	1.0	1.0	1.0	1.0	1.0	1.0	1.00	1.0

Average confidence of prediction novel classes (2) is shown in Table 6. This suggests that the proposed Bayesian models-MC Dropout exhibit unreliable ability of threshold-based confidence decision. Average confidence of MC Dropout is higher than 0.8, i.e., threshold-decision cannot distinguish between novel and non-novel classes, all with high confidence.

Table 6: Average confidence: MC Dropout

64APSK	GMSK	AM-SSB-SC	AM-DSB-SC	128QAM	8PSK	8ASK	OQPSK
0.886	0.986	0.996	0.999	0.886	0.918	0.999	0.873

Prediction novel classes can use relative frequency (3) of confidence exceeds C_{th} for MC Dropout is shown in Table 7.The values in Table 7 are the proportion of average confidence $Conf_{avg}(x)$ exceeding the threshold for each out of domain of 4096 samples when MC Dropout is set to threshold=0.8, among which 0.85×100% of 128QAM, 95% of 8PSK, 84% of 64APSK and 85% of OQPSK are decided to be non-novel classes.

Table 7: Relative frequency of confidence exceed C_{th} : MC Dropout

	C_{th} =0.8	C_{th} =0.85	C_{th} =0.9	C_{th} =0.95
GMSK	1.0	0.99	0.99	0.95
AM-SSB-SC	1.0	0.99	0.99	0.97
OQPSK	0.85	0.65	0.37	0.15
64APSK	0.84	0.67	0.47	0.26
GMSK	1.0	1.0	0.99	0.99
8PSK	0.95	0.84	0.63	0.38
128QAM	0.85	0.67	0.47	0.26
AM-DSB-SC	1.0	1.0	1.0	1.0

Conclusion

This paper presents the application of Bayesian learning techniques, variational inference (VI) and Monte Carlo (MC) Dropout, to Automatic Modulation Classification (AMC). Raw I/Q channel waveforms are directly input into LSTM-based VI and MC Dropout models. Both models achieve classification accuracies exceeding 90% within a 20--30 dB SNR range.This paper has proposed relative frequency of confidence score and illustrates the framework's new observation to quantify uncertainty. However, only the VI-based model demonstrates the ability to detect novel modulation types using a threshold-based novelty detection mechanism. In contrast, the LSTM-based MC Dropout model frequently misclassifies previously unseen modulation signals as known classes with high confidence.

Furthermore, we observe that the confidence assigned to novel modulation types is influenced by their proximity within the learned feature space to in-domain classes. This finding underscores the importance of limiting training to essential modulation classes. Expanding the set of in-domain classes may inadvertently increase the likelihood that novel signals are assigned high-confidence predictions to known classes, thereby complicating the detection of truly novel modulation schemes.

Reference

[1]. T. Huynh-The, Q. V. Pham, T. V. Nguyen, T. T. Nguyen, R. Ruby, M. Zeng, and D. S. Kim, "Automatic modulation classification: A deep architecture survey," *IEEE Access*, vol. 9, pp. 142950–142971, 2021.

[2]. B. Jdid, K. Hassan, I. Dayoub, W. H. Lim, and M. Mokayef, "Machine learning based automatic modulation recognition for wireless communications: A comprehensive survey," *IEEE Access*, vol. 9, pp. 57851–57873, 2021.

[3]. F. Zhang, C. Luo, J. Xu, Y. Luo, and F.-C. Zheng, "Deep learning based automatic modulation recognition: Models, datasets, and challenges," *Digit. Signal Process.*, vol. 129, Sep. 2022.

[4]. V.-C. Luu, J. Park, and J.-P. Hong, "Uncertainty-aware incremental automatic modulation classification with Bayesian neural network," *IEEE Internet Things J.*, vol. 13, no. 13, pp. 24300–24309, Jul. 2024.
[5]. O. Durr, B. Sick, and E. Murina, *Probabilistic Deep Learning*. New York, NY, USA: Manning Publ., 2020.
[6]. T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Top. Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.
[7]. L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on Bayesian neural networks—A tutorial for deep learning users," *IEEE Comput. Intell. Mag.*, vol. 17, no. 2, pp. 29–48, May 2022.