

# Investigating Plausibility of Biologically Inspired Bayesian Learning in ANNs

Ram J. Zaveri

rz0012@mix.wvu.edu

West Virginia University

## 1 Abstract

*Catastrophic forgetting has been the leading issue in the domain of lifelong learning in artificial systems. Current artificial systems are reasonably good at learning domains they have seen before; however, as soon as they encounter something new, they either go through a significant performance deterioration or if you try to teach them the new distribution of data, they forget what they have learned before. Additionally, they are also prone to being overly confident when performing inference on seen as well as unseen data, causing significant reliability issues when lives are at stake. Therefore, it is extremely important to dig into this problem and formulate an approach that will be continually adaptable as well as reliable. If we move away from the engineering domain of such systems and look into biological systems, we can realize that these very systems are very efficient at computing the reliance as well as the uncertainty of accurate predictions that further help them refine the inference in a life-long setting. These systems are not perfect; however, they do give us a solid understanding of the reasoning under uncertainty which takes us to the domain of Bayesian reasoning. We incorporate this Bayesian inference with thresholding mechanism as to mimic more biologically inspired models, but only at spatial level. Further, we reproduce a recent study on Bayesian Inference with Spiking Neural Networks for Continual Learning to compare against it as a suitable biologically inspired Bayesian framework. Overall, we investigate the plausibility of biologically inspired Bayesian Learning in artificial systems on a vision dataset, MNIST, and show relative performance improvement under the conditions when the model is forced to predict VS when the model is not.*

## 2 Introduction

In recent years, we have encountered many problems with Artificial Neural Networks (ANNs), for example, these very ANNs, even though, are pretty good approximators; they fall short when they meet information that does not belong to their prior distributions, and therefore; go through significant performance deterioration. Biological systems are well equipped to handle that. The population modeling of the neuronal dynamics suggests that each neuron possesses a temporal and spatial threshold to process the information flow to the next neuron. This essentially helps determine “if” the neuron should be activated by the prior input at this very time instance. If there is an accumulation of ions in the pre-synaptic cleft, the next neuron is more likely to get activated to process the signal [?]. I believe this is the time we go back to the biological drive to develop more efficient artificial systems. The reason is, that threshold-based mechanisms, at the very high level at least, help the neuron decide the functional co-dependence of the neuron’s activation and the signals it is processing.

In the case of ANNs, parameters such as weight and bias help it refine the distribution clusters in the hyperspace. The way the calculations are performed is through the accumulation of weights from previous neurons passed through an activation function (ReLU, Sigmoid, etc. ) to add the nonlinearity to the prior layer, and then process it to the next layer. Of course, activation layers are extremely important since they add non-linearity to the neural network and help with exploding weights from the previous layer. However, it is assumed that once the model has been trained, the learned parameters, weights, and biases, are optimal to the unseen data. This is not the case in general, the ANNs overfit to the seen data, and either do not perform well or if trained again on new data, they will experience catastrophic forgetting.

In essence, we need to add a means of certainty in the learning process. The other way to say this is to add a level of uncertainty to individual neurons. Scientists are trying to tackle this problem by defining means and variance of the weights; it is referred to as Bayesian Neural Networks (BNNs). The goal here is to learn the certainty of the accumulated weights based on the firing pattern [?].

As described in [?], the free energy hypothesis suggests measuring surprise in terms of variational free energy. If we put it in the context of the biological brain, it constantly updates an internal model of the world and tries to minimize the information-theoretic surprise, it is formerly known as the Bayesian brain hypothesis. In essence, synaptic plasticity can be formulated as a Bayesian learning mechanism where it keeps track of, not the weights, but the distribution of synaptic weights over time [?, ?]. Therefore, investigating the probability of an event occurring, rather than saying that this particular event has occurred, would give us a more generalized, and symmetrical point of view on the problem of continual adaptability. In essence, the goal here is to predict a continuous variable to determine certainty of an event would give a more biologically plausible answer.

Widely understood mechanisms that describe spike-time plasticity, long-term potentiation (LTP), and long-term depression (LTD) follow similar concepts, where LTP occurs when the neurons spike in sequence to strengthen the synapse between two consecutive neurons, essentially experiencing an increase in weight; on the other hand, LTD does the opposite to weaken the synapses, essentially decreasing the weight. LTP contributes to long-term memory by consolidating synaptic connections, forming stability throughout the network. It occurs via either episodic replay of sequential pre-synaptic signals (repeating) or if the pre-synaptic signals were extremely strong (high-frequency) to induce an LTP. This essentially means that LTP can be caused by either temporal summation or spatial summation or both [?]. In biological networks, narrowly tuned  $\text{Ca}^{+2}$  release is known to be associated with different types of LTP, usually causing short-term potentiation (STP) for minutes to hours to cause late LTP [?, ?]. However, if the network has experienced a similar situation before, it will cause it to go through a synaptic change called metaplasticity [?]. Moreover, during an inactive state (sleep), these very connections become more refined by episodic replay [?]. We will leverage this information and design a paradigm for the artificial systems using learnable activation/thresholds as a means of spatial summation. Additionally, we will use terms activation/thresholds interchangeably throughout this paper.

### 3 Related Works

Bayesian Neural Networks are increasingly gaining popularity in deep learning community, where more recently, [?] combines the most recent research to prepare a tutorial specifically for deep learning practitioners. The advantage of BNNs is that they are not prone to overfitting as can be observed in traditional learning schemes. They are being used in many fields, i.e., computer vision [?], network traffic monitoring [?], medicine [?], active learning [?, ?], online learning [?], and so on. Additionally, as observed in [?], since the calculated posteriors can be reused as priors for the data the model has not seen before, it can inherently avoid the major problem of catastrophic forgetting. Other key advantages are they naturally learn to quantify uncertainty, they can even distinguish between uncertainty associated with the data it has seen and the data it has never seen, making it robust against anomaly [?]. Additionally, they are also different from, so called, black box algorithms with traditional deep learning models since, the prior that are used in BNNs are explicit [?]. [?, ?] further incorporates Bayesian Learning rule with natural gradient descent in a fashion it works, if not exactly, similarly, without destroying the core fundamentals of machine learning principles, with machine learning algorithms.

Continual learning refers to a setting where an algorithm is learning in a continuous fashion. Thus said, the algorithm does need to preserve the past information it learned; however, most approaches struggle in this particular setting and go through catastrophic forgetting [?]. As we mentioned earlier, BNNs are stochastic models and learn continuous distributions (one can also discretize this [?]), probabilistic approximations are more stable compared to their ANNs counterparts. Additionally, there are tremendous research works mentioning various mathematical and empirical tricks, i.e., meta learning [?], regularization [?], etc., to avoid catastrophic forgetting; however, none have made any significant breakthroughs. On the other hand, more on the biologically plausible models, which are less discrete,

[?, ?], have made progress in terms of conceptual significance of Spiking Neural Networks (SNNs), specifically leaky integrate-and-fire (LIF) model, on standard continual learning benchmarks. Here, SNNs are wonderful tools as they can be mimicked at both, software and hardware-level, which is to show the significance of memory and energy-efficiency, which we call neuromorphic tools. Here, our major focus [?], have proposed to use Bayesian Learning with SNNs in a continual fashion. They use variational inference with surrogate gradients to perform Bayesian learning in SNNs, and are achieving noticeable performance gain. In this paper, we are investigating a rather simplistic model with learnable activations to showcase the behavior of Bayesian framework on unforced certainty prediction and forced prediction. We further incorporate [?] framework to achieve extreme biological plausibility. We note that, we are thoroughly investigating Bayesian Learning in this paper to provide better conceptual understanding and not focusing on the basics of LIF spiking models since the class has covered major portions of that, the other mechanisms remain the same for the most part.

## 4 Methods

### 4.1 Overview

We investigate the plausibility of biologically inspired Bayesian Learning in ANNs by first investigating the core components of Bayesian Neural Networks (BNNs). We further describe thresholding as ANNs counterpart to biological activation of a neuron. We further discuss its applicability in a continual learning situation integrated with a more biologically plausible model, Spiking Neural Networks (SNNs) from [?].

### 4.2 Bayesian Neural Networks

As opposed to the frequentist approach, Bayesian Inference considers the probability as a measure of belief in the occurrences of events rather than the limit on the frequency, meaning the hypothesis testing is done on the probability distribution of events occurring, which again brings us to the theory of variational free energy that describes the probabilistic model to predict observations from the hypothesized causes [?, ?, ?]. Therefore, in Bayesian Inference, the prior beliefs influence the posterior beliefs. See equation ??.

$$P(H|D) = \frac{P(D|H)P(H)}{P(D)} = \frac{P(D, H)}{\int_H P(D, H')dH'} \quad (1)$$

Where  $P(D|H)P(H)$  is the likelihood,  $P(H)$  is the prior,  $P(D) = \int_H P(D, H')dH'$  is the evidence from the data  $D$ ,  $P(H|D)$  is the posterior, and  $P(D, H)$  is the joint probability.

As mentioned in [?], BNNs are referred to as stochastic artificial neural networks using Bayesian Inference. Here, stochastic models utilize probability distributions rather than a point estimate of the value unlike the traditional neural networks. This gives a better understanding of precision, or uncertainty, of the weight vector that is associated with every neuron. This is typically done by incorporating a stochastic activation or weight as observed in BNNs. Therefore, they have capacity to simulate various possible solutions, inherently forming an ensemble of networks. For example, assume  $\theta = (\mathbf{W}, \mathbf{b})$ , where  $\theta$  is the model parameters,  $\mathbf{W}$  and  $\mathbf{b}$  are weights and biases, respectively. The traditional networks will learn through a point estimate equation,  $s(\mathbf{W}\mathbf{l} + \mathbf{b})$ , where  $s$  is the activation function. In stochastic models, either  $s$  or  $\theta = (\mathbf{W}, \mathbf{b})$  are associated with probability distributions. And since the computation is inherently dependent on the probability distributions of either  $s$  or  $\theta$ , stochastic models are forming ensembles of distributions, thus, can be used as a special case of ensemble learning [?].

To train the BNNs, we will first have to establish the Bayesian Posterior:

$$p(\theta|D) = \frac{p(D_y|D_x, \theta)p(\theta)}{\int_{\theta} p(D_y|D_x, \theta')p(\theta')d\theta'} \propto p(D_y|D_x, \theta)p(\theta) \quad (2)$$

During prediction for a given  $p(\theta|D)$ , we can compute the prediction,  $p(y|x, D)$  (sampled from  $y = \Phi_{\theta}(x) + \epsilon$ , where  $\Phi$  is an approximation and  $\epsilon$  is the noise) as follows:

$$p(y|x, D) = \int_{\theta} p(y|x, \theta')p(\theta'|D)d\theta' \quad (3)$$

Here, computing  $\int_{\theta} p(D_y|D_x, \theta')p(\theta')d\theta'$ , is extremely difficult and time-consuming, since we are trying to perform integral over all the possible parameters, extremely in-efficient. Therefore, one of the two major approaches is used in this work, called Variational Inference. The other one is Markov Chain Monte Carlo; which falls short with increasing scalability as opposed to variational inference [?].

### Variational Inference.

Variational Inference [?] is not exact; however, provides very good approximation with increasing scale. Here, instead of sampling from an exact posterior mentioned in Equation ??, we compute a distribution  $q_{\phi}(H)$ , namely variational distribution, parameterized by parameters  $\phi$ . Overall, we want  $q_{\phi}(H)$  to be as close to the exact posterior  $P(H|D)$  as possible. We use Kullback-Leiber (KL) divergence [?] function to measure that based on Shannon's information theory [?]. The overall KL-Divergence loss is as follows:

$$D_{KL}(q_{\phi}||P) = \int_H q_{\phi}(H') \log\left(\frac{q_{\phi}(H')}{P(H'|D)}\right) dH' \quad (4)$$

However, KL-Divergence loss still takes  $P(H|D)$  into account, and will still need to calculate it; rather, if we simplify the formula, we get the following:

$$\log(P(D)) - D_{KL}(q_{\phi}||P) = \int_H q_{\phi}(H') \log\left(\frac{q_{\phi}(H')}{P(H')}\right) dH' \quad (5)$$

Here, during gradient calculation,  $\log(P(H'|D))$  becomes  $\log(P(H', D)) - \log(P(D))$ , where  $\log(P(D))$  is a constant and becomes 0. Therefore, we can safely eliminate this term and reduce it to the equation ??, which is also referred to as evidence lower bound ELBO loss and the way to optimize this loss is called stochastic variational inference [?].

Now, to perform backpropagation over a neural network, we need to calculate gradients over equation ?? according to the following equation:

$$\frac{\partial}{\partial \phi} \int_{\phi} q_{\phi}(\theta') f(\theta', \phi) d\theta' = \int_{\epsilon} q(\epsilon) \left( \frac{\partial f(\theta, \phi)}{\partial \theta} \frac{\partial \theta}{\partial \phi} + \frac{\partial f(\theta, \phi)}{\partial \phi} \right) \quad (6)$$

where,  $q_{\phi}(\theta) \partial \theta = q(\epsilon) \partial \epsilon$ .

Further, to estimate the priors during inference, we first need to establish that equation ?? is KL-divergence of  $q(\phi)$  subtracted from the log-likelihood of the data,  $\log(P(D)) - D_{KL}(q_{\phi}||P)$ , which means equation ?? is a function of both the variational parameters,  $\phi$ , to estimate the posterior, and (addition of ) a parameterized prior distribution,  $p_{\xi}(H)$ , given the parameters,  $\xi$ . Therefore, the loss function now becomes,

$$L = \log(q_{\phi}(\theta)) - \log(p_{\xi}(D_y|D_x, \theta)p_{\xi}(\theta)) \quad (7)$$

Here, we perform gradient decent with respect to both  $\phi$  and  $\xi$  using the equation ?? (replace  $\phi$  with  $\xi$ ).

## 4.3 Thresholding

As our baseline comparison, we follow the guidelines provided in section ??. We further describe the stochastic models that make use of the probabilistic parametric functions that complies with the concept of adding a variational threshold for activation for introducing nonlinearity in a stochastic fashion. We diverge from this idea, and test a simple threshold-based mechanism, which takes learnable threshold combined with ReLU as the activation function. Here, the basic idea is to consider spatial information as a means of collective-ness, in-short, threshold. This works as masking of the neurons based on the training data. Consider  $s(\mathbf{W}\mathbf{l} + \mathbf{b})$  terms where  $\theta = (\mathbf{W}, \mathbf{b})$  are learened parameterized distributions and  $s$  is learnable activation/threshold. Here, we should note that, this thresholding is not a probability distribution just yet, it is a means to provide weight of the neuron given the experience during training, results are shown in section ??. So far, we can call this scheme as learnable activations. Traditional neural networks have shown some performance gain with this scheme, but not significant [?]. However, with Bayesian Learning, it turns out, when the model was forced to predict, the overall accuracy decreased, but surprisingly, when the model was not forced, rather, skipped some inputs when it was "uncertain", the accuracy jump was relatively high, suggesting with learnable activations, the model might be plastic to change, but is relatively stable towards the things it understands.

Table 1: Preliminary Testing when the model is forced to predict VS when the model is not.

	Forced Pred.		Not Forced Pred.	
	Skipped	Accuracy	Skipped	Accuracy
Generic BNNs	0/10000	0.87	1415/10000	0.95
BNNs with Learnable Activations	0/10000	0.79	3735/10000	0.96

## 4.4 Continual Learning

To test the BNNs with baseline thresholding in a continual learning situation, we choose MNIST [?] dataset that consists of 10 digits, and following [?] method, and split them in 5 groups ([0,1], [2,3], [4,5], [6,7], [8,9]) sequentially. We train the model with each group once and in sequence and calculate the performance in terms of accuracy for this particular task. The results are shown in section ?? . This test was designed to test our baseline thresholding. However, if we incorporate more biologically inspired models, such as leaky integrate-and-fire (LIF), we need to go into the realm of Spiking Neural Networks (SNNs) [?]. But that is not the only motivation, learnable thresholding with the temporal component associated with it would result in a more biologically plausible Bayesian Learning framework. This is an inherent property of the SNNs. Therefore, it is natural to investigate these spiking models for continual learning, specifically as a Bayesian Learning framework. [?] tackles this and achieves remarkable performance. We incorporate their framework and test it against previously described mechanisms, and show the results in section ??.

# 5 Results

## 5.1 Implementation Details

All the code was written in Python and the models are implemented in PyTorch [?]. All the models in this work are using 784 as input dimension (MNIST [?] dimensions are  $28 \times 28 \in R^2 \rightarrow 784 \in R^1$ ), 400 as hidden dimensions, and 10 as output dimensions given the number of digits in MNIST [?] are 10. All the nodes are fully connected nodes. Models without spikes are merely traditional MLPs with additional functionality of learnable thresholding or activation. For direct test on all 10 digits, we train the model for 5 epochs, and during continual learning, we only train it for one epoch every task following the implementation of [?], i.e., [0,1] category, then [2,3] category, so on and so forth.

## 5.2 Baseline Testing

To investigate biological plausibility of Bayesian Inference, we incorporated learnable threshold/activation to the generic BNNs. Table ?? illustrates that under forced conditions generic BNNs achieve significantly better accuracy compared to their counterpart with learnable activations. However, as we incorporate their uncertainty into the prediction and discard values with lower precision than 0.8 for any given class, we see that BNNs with learnable thresholds achieves comparatively good accuracy. We should note that, this is only a preliminary test and is to investigate if the BNNs are compatible with learnable thresholds. This shows that they are indeed compatible, and our speculation is that lower certainty describes their ability to be plastic towards fluctuations and stable only when absolutely certain. However, it still needs more investigation to solidify the study since the approach with learnable activation does skip a large number of samples, which might not be optimal.

## 5.3 Testing on Continually changing tasks

Table ?? illustrates empirical study on continually changing tasks, specifically on generic BNNs, BNNs with learnable thresholds/activation in conditions where model is forced to predict and when the model is not (skipping predictions are allowed based on the probability), and BNNs with SNNs. Here, this continually changing tasks, as described in section ??, are five sequential tasks split evenly from the MNIST [?] dataset. As observed from the results, Generic BNNs are performing wonderfully well compared to BNNs with learnable activations in conditions where the model is forced to predict. However, as we can notice, BNNs with learnable activations had a significant improvement when the model was not forced to

fig/qual\_baseline.pdf

Figure 1: **Qualitative results on BNNs with Learnable Thresholding.** First two rows are results from the learned classes, the following two rows are from alphabets, (meaningful but) not learned domain, and the last two rows are results with random noise.

Table 2: Comparative study on Generic BNNs, BNNs with learnable thresholds/activations (with and without skipping allowed), and BNNs with SNNs (LIF model).

Training	Tasks														
	Task 1	Task 2		Task 3			Task 4				Task 5				
	Task 1	Task 1	Task 2	Task 1	Task 2	Task 3	Task 1	Task 2	Task 3	Task 4	Task 1	Task 2	Task 3	Task 4	Task 5
Generic BNNs	0.998	0.950	0.973	0.987	0.845	0.813	0.967	0.880	0.790	0.790	0.949	0.850	0.899	0.704	0.543
Generic BNNs (skipping allowed)	0.999	0.978	0.968	0.997	0.901	0.902	0.965	0.883	0.840	0.930	0.976	0.896	0.895	0.842	0.526
BNNs with Learnable Activations	0.998	0.917	0.946	0.978	0.784	0.818	0.950	0.911	0.641	0.793	0.966	0.862	0.722	0.614	0.567
BNNs with Learnable Activations (skipping allowed)	0.998	0.963	0.970	0.999	0.912	0.903	0.997	0.967	0.850	0.929	0.998	0.981	0.951	0.882	0.849
BNNs with SNNs	0.999	0.995	0.800	0.956	0.952	0.605	0.976	0.878	0.882	0.914	0.959	0.867	0.782	0.841	0.892

predict (skipping allowed). We suspect, the reason for that would be the model being more plastic towards the change, whereas the stable connections are maintaining the model performance when the continually learning the tasks. Further, we test BNNs with SNNs (LIF model) based on [?] study to showcase the neuromorphic capacity of the BNNs, and the results show comparable improvement over the BNNs without SNNs. Additionally, to get an idea of uncertainty in BNNs qualitatively, please consider Figure ??.

## 6 Discussion

Since the motivation is to derive a more biologically inspired Bayesian modeling of the network, we investigated, a simple learnable activation/thresholding as can be observed in biological networks in terms of tuning curves. This simple trick helped model learn relevant information in a continual learning situation, where all the data is not present at once, instead, it becomes available sequentially. Further we used study proposed in [?] and reproduce the results on BNNs with SNNs to show with additional complexity of spiking networks, we can leverage a more biologically inspired Bayesian learning scheme, and shows incredible

performance increase. Additionally, our initial motivation was to utilize learnable activation/thresholds in spiking networks; however, upon investigation, it was found that it is their inherent property, spiking rate as measure of significance/activation, and therefore, they are more biologically plausible networks. The results indeed support that. Further, one of the biggest limitations of this work is that the work is still in its preliminary stage, and requires a lot more investigation on simple learnable activation-based BNNs and spiking models. It is quite difficult to draw conclusions based on this study this early; however, that does inspire us to investigate more of their plausibility in the realm of spiking models in the future. Another limitation is the scalability of the network, we are unsure if this particular scheme can be scaled to bigger models and datasets given the complexity of the model is quite limited and also the data. We would like to investigate bigger and more complex models with larger datasets to test the applicability of this work in the real world. Another major limitation is allowing the model to skip certain predictions, the question would be, how many skips are considered okay, we need more investigation on this as well.

## 7 Conclusion

In this work, we proposed to use learnable thresholding/activation of the neurons as a measure of tuning curves, inspired by biological neurons, to investigate the plausibility of biologically inspired Bayesian learning as mentioned in [?], who are using spiking neural networks that work in the LIF fashion. The study still requires more investigation; however, this preliminary study shows that learnable threshold provide a measure of tuning curves and can maintain both stability and plasticity throughout their continual learning tasks when model is not forced to predict. This gives us a new direction of ideas to start thinking in terms of, do the artificial networks really need to make solid decisions, or they can refuse if they are uncertain. Ethically, this could potentially make a huge difference in the field of Artificial Intelligence, and is a wonderful question to ask.

## 8 Acknowledgements

We appreciate the work shared by [?] at <https://github.com/kclip/bayesian-snn>, and a basic tutorial: <https://github.com/paraschopra/bayesian-neural-network-mnist>. We further thank Dr. Nicholas Szczecinski for approving this work.

## References

- [1] L. Barrett, *Seven and a Half Lessons about the Brain*. Houghton Mifflin Harcourt Publishing Company, 2020.
- [2] E. T. Jaynes, *Probability Theory*, G. L. Bretthorst, Ed. Cambridge University Press, apr 2003.
- [3] N. Skatchkovsky, H. Jang, and O. Simeone, “Bayesian continual learning via spiking neural networks,” *Frontiers in Computational Neuroscience*, vol. 16, nov 2022.
- [4] L. Aitchison, J. Jegminat, J. A. Menendez, J.-P. Pfister, A. Pouget, and P. E. Latham, “Synaptic plasticity as bayesian inference,” *Nature Neuroscience*, vol. 24, no. 4, pp. 565–571, mar 2021.
- [5] K. Friston, “The history of the future of the bayesian brain,” *NeuroImage*, vol. 62, no. 2, pp. 1230–1233, aug 2012.
- [6] J. C. McEachern and C. A. Shaw, “The plasticity-pathology continuum: Defining a role for the LTP phenomenon,” *Journal of Neuroscience Research*, vol. 58, no. 1, pp. 42–61, sep 1999.
- [7] C. R. Raymond and S. J. Redman, “Different calcium sources are narrowly tuned to the induction of different forms of LTP,” *Journal of Neurophysiology*, vol. 88, no. 1, pp. 249–255, jul 2002.
- [8] G. Cao and K. M. Harris, “Developmental regulation of the late phase of long-term potentiation (l-LTP) and metaplasticity in hippocampal area CA1 of the rat,” *Journal of Neurophysiology*, vol. 107, no. 3, pp. 902–912, feb 2012.

- [9] A. P. Crestani, J. N. Krueger, E. V. Barragan, Y. Nakazawa, S. E. Nemes, J. A. Quillfeldt, J. A. Gray, and B. J. Wiltgen, “Metaplasticity contributes to memory formation in the hippocampus,” *Neuropsychopharmacology*, vol. 44, no. 2, pp. 408–414, may 2018.
- [10] N. Soures, P. Helfer, A. Daram, T. Pandit, and D. Kudithipudi, “Tacos: task agnostic continual learning in spiking neural networks,” in *Theory and Foundation of Continual Learning Workshop at ICML’2021*, 2021.
- [11] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, “Hands-on bayesian neural networks—a tutorial for deep learning users,” *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29–48, 2022.
- [12] 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.
- [13] T. Auld, A. W. Moore, and S. F. Gull, “Bayesian neural networks for internet traffic classification,” *IEEE Transactions on neural networks*, vol. 18, no. 1, pp. 223–239, 2007.
- [14] W. Beker, A. Wołos, S. Szymkuć, and B. A. Grzybowski, “Minimal-uncertainty prediction of general drug-likeness based on bayesian neural networks,” *Nature Machine Intelligence*, vol. 2, no. 8, pp. 457–465, 2020.
- [15] Y. Gal, R. Islam, and Z. Ghahramani, “Deep bayesian active learning with image data,” in *International conference on machine learning*. PMLR, 2017, pp. 1183–1192.
- [16] T. Tran, T.-T. Do, I. Reid, and G. Carneiro, “Bayesian generative active deep learning,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 6295–6304.
- [17] M. Opper and O. Winther, “A bayesian approach to on-line learning,” 1999.
- [18] H. Ritter, A. Botev, and D. Barber, “Online structured laplace approximations for overcoming catastrophic forgetting,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [19] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft, “Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning,” in *International Conference on Machine Learning*. PMLR, 2018, pp. 1184–1193.
- [20] M. Khan and W. Lin, “Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models,” in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 878–887.
- [21] M. E. Khan and H. Rue, “The bayesian learning rule,” *arXiv preprint arXiv:2107.04562*, 2021.
- [22] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, “Meta-learning in neural networks: A survey,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 5149–5169, 2021.
- [23] H. Ahn, S. Cha, D. Lee, and T. Moon, “Uncertainty-based continual learning with adaptive regularization,” *Advances in neural information processing systems*, vol. 32, 2019.
- [24] A. Laborieux, M. Ernoult, T. Hirtzlin, and D. Querlioz, “Synaptic metaplasticity in binarized neural networks,” *Nature communications*, vol. 12, no. 1, p. 2549, 2021.
- [25] J. Bruineberg, J. Kiverstein, and E. Rietveld, “The anticipating brain is not a scientist: the free-energy principle from an ecological-enactive perspective,” *Synthese*, vol. 195, no. 6, pp. 2417–2444, 2018.
- [26] A. Etz, Q. F. Gronau, F. Dablander, P. A. Edelsbrunner, and B. Baribault, “How to become a bayesian in eight easy steps: An annotated reading list,” *Psychonomic bulletin & review*, vol. 25, no. 1, pp. 219–234, 2018.
- [27] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. CRC press, 2012.
- [28] 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.



- [29] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [30] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [31] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, 2013.
- [32] M. Goyal, R. Goyal, and B. Lall, "Learning activation functions: A new paradigm of understanding neural networks. arxiv 2019," *arXiv preprint arXiv:1906.09529*, 1906.
- [33] Y. LeCun, "The mnist database of handwritten digits," <http://yann.lecun.com/exdb/mnist/>, 1998.
- [34] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.