Bayesian Neural Networks for Cellular Image Classification and Uncertainty Analysis

Giacomo Deodato[1,2] GIACOMO.DEODATO@GMAIL.COM
[1] *Novartis Institutes for Biomedical Research, Basel, Switzerland*
[2] *Eurecom, Sophia Antipolis, France*

Christopher Ball[1] CHRISTOPHER.BALL@NOVARTIS.COM
Xian Zhang[1] XIAN-1.ZHANG@NOVARTIS.COM

Abstract

Over the last decades, deep learning models have rapidly gained popularity for their ability to achieve state-of-the-art performances in different inference settings. Novel domains of application define a new set of requirements that transcend accurate predictions and depend on uncertainty measures. The aims of this study are to implement Bayesian neural networks and use the corresponding uncertainty estimates to improve predictions and perform dataset analysis. We identify two main advantages in modeling the predictive uncertainty of deep neural networks performing classification tasks. The first is the possibility to discard highly uncertain predictions to increase model accuracy. The second is the identification of unfamiliar patterns in the data that correspond to outliers in the model representation of the training data distribution. Such outliers can be further characterized as either corrupted observations or data belonging to different domains. Both advantages are well demonstrated on benchmark datasets. Furthermore we apply the Bayesian approach to a biomedical imaging dataset where cancer cells are treated with diverse drugs, and show how one can increase classification accuracy and identify noise in the ground truth labels with uncertainty analysis.

1. Introduction

Deep neural networks have seen a dramatic increase in popularity in recent years, due to their outstanding performances in complex prediction tasks (Krizhevsky et al., 2012; LeCun et al., 2015). The main drawback of neural networks lies in their lack of interpretability (they are often deemed as "black boxes" (Benítez et al., 1997; Shrikumar et al., 2017; Lundberg and Lee, 2017)) and their dependence on point estimates of their parameters. Despite their ability to outperform simpler models, a single prediction score (i.e. the accuracy of the prediction) is not sufficient for a variety of tasks and domain applications (Ghahramani, 2015). Modeling applications such as healthcare require an additional feature to the prediction score, that is a measure of confidence that reflects the uncertainty of the predictions. For example, a neural network performing diagnosis of brain tumors by analyzing magnetic resonance images needs a way to express the ambiguity of an image in the same way as a doctor may express uncertainty and ask for experts help. Moreover, predictive uncertainty provides further insights about the data because more certain predictions correspond to cleaner data both from a technical and a contextual point of view.

© G. Deodato, C. Ball & X. Zhang.

The output of neural networks is often treated as a probability distribution. However, despite there is a correlation between the accuracy of the prediction and this confidence score, that is the output value, this should not lead to think that it is an appropriate measure of uncertainty as this would show that the model makes mostly overconfident predictions (Gal and Ghahramani, 2016).

Instead of the described prediction score, we analyzed data by means of the predictive uncertainty, that can be decomposed into epistemic uncertainty, which stems from model's parameters as well as the specific architecture of the model, and aleatoric uncertainty, which depends on the noise of the observations (Der Kiureghian and Ditlevsen, 2009).

In the remaining sections of this paper we present our implementation of Bayesian neural networks using variational inference and our confidence measure formulation (2), we briefly analyze the most relevant alternative to our approach (3) and we show our results over multiple datasets belonging to different domains (4). Finally, we discuss the results and the advantages of our approach (5).

2. Methods

The idea behind Bayesian modeling is to make predictions by considering all possible values for the parameters of the model. In order to do so, the parameters are treated as random variables whose distribution is such that the most likely values are also the most probable ones. The posterior distribution of the model parameters, conditioned on the training data, is defined using Bayes theorem:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})} \tag{1}$$

where $\mathbf{w}$ is the set of model parameters and $\mathcal{D}$ is the training set. The estimation of the posterior requires the definition of a likelihood function $p(\mathcal{D}|\mathbf{w})$, a prior density $p(\mathbf{w})$, and the computation of the marginal likelihood $p(\mathcal{D})$ that is unfeasible for complex models such as neural networks.

2.1. Mean Field Variational Inference

Variational inference allows to avoid computing the marginal likelihood by directly approximating the posterior distribution with a simpler one (Jordan et al., 1999). In order to do so, it is necessary to minimize the Kullback–Leibler (KL) divergence between the proposed distribution and the posterior. The KL divergence is defined as follows:

$$KL\{q(\mathbf{w};\theta)||p(\mathbf{w}|\mathcal{D})\} = \int q(\mathbf{w};\theta)\log\frac{q(\mathbf{w};\theta)}{p(\mathbf{w}|\mathcal{D})}d\mathbf{w} \tag{2}$$

where θ is the set of variational parameters describing the proposed distribution q of the model's parameters $\mathbf{w}$. Since the posterior distribution is not known, we need to define a different objective to minimize the KL divergence. Such objective function is called Evidence Lower Bound (ELBO) and it is defined as follows:

$$ELBO = \mathbb{E}_{q(\mathbf{w};\theta)}\{\log\, p(\mathcal{D}|\mathbf{w})\} - KL\{q(\mathbf{w};\theta)||p(\mathbf{w})\} \tag{3}$$

Variational inference turns the marginal likelihood computation problem into an optimization one: maximizing the ELBO as a function of the variational parameters so that the proposed distribution fits the posterior (proof in Appendix A).

We approximated the posterior with a multivariate Gaussian distribution and, in order to simplify the optimization process, we used the mean field approximation. This choice allowed us to avoid a quadratic increase of the parameters to optimize by factorizing the posterior approximation:

$$q(\mathbf{w};\theta) = \prod_i q(w_i;\theta_i) = \prod_i N(\mu_i, \sigma_i) \tag{4}$$

2.2. Bayesian Neural Network Training

In order to be able to apply the backpropagation algorithm to the variational parameters of a Bayesian neural network, we applied the local reparameterization trick (Kingma et al., 2015) which separates the deterministic and the stochastic components of the weights, which are random variables. Furthermore, we also reparameterized the standard deviations of the weights using the softplus function to keep them positive.

The loss function of the Bayesian neural network is the variational objective, i.e. the negative ELBO, where the likelihood can be divided in the sum of the contributions of all the individual data points in the dataset (Hoffman et al., 2013; Mandt et al., 2016, 2017) and it is possible to employ a minibatch approach (Graves, 2011; Blundell et al., 2015):

$$f(\mathcal{D}_i, \theta) = -\mathbb{E}_{q(\mathbf{w};\theta)}\{\log\, p(\mathcal{D}_i|\mathbf{w})\} + \beta KL\{q(\mathbf{w};\theta)||p(\mathbf{w})\} \tag{5}$$

where $\mathcal{D}_i$ represents the i-th mini-batch, $\beta = 1/M$ is the scaling factor of the KL divergence due to the minibatch approach, and M is the number of mini-batches. The prior distribution is a Gaussian distribution that is equivalent to L2 regularization (Blundell et al., 2015).

2.3. Predictive Uncertainty

The probability distribution over the parameters of the model yields a predictive distribution whose mean can be approximated using Monte Carlo samples from the posterior approximation:

$$p(t|x,\mathcal{D}) = \int p(t|x,\mathbf{w})p(\mathbf{w}|\mathcal{D})d\mathbf{w} \approx \frac{1}{S}\sum_{s=1}^{S} p(t|x,\mathbf{w}_{(s)}), \quad \mathbf{w}_{(s)} \sim q(\mathbf{w};\theta) \tag{6}$$

where S is the number of weights samples taken, therefore the number of output samples too. The corresponding predictive uncertainty can be computed starting from the variance of the predictive distribution, considering both the epistemic and aleatoric components of uncertainty (Kwon et al., 2018; Kendall and Gal, 2017):

$$\underbrace{\frac{1}{S}\sum_{s=1}^{S} p(t_i|x,\mathbf{w}_{(s)}) - p(t_i|x,\mathbf{w}_{(s)})^2}_{\text{aleatoric}} + \underbrace{\frac{1}{S}\sum_{s=1}^{S}\Big(p(t_i|x,\mathbf{w}_{(s)}) - p(t_i|x,\mathcal{D})\Big)^2}_{\text{epistemic}} \tag{7}$$

where i is the index of the predicted class. Finally, we used the function $f(x) = 1 - 2\sqrt{x}$ to transform such uncertainty measure into a more intuitive Bayesian confidence score which is also easier to compare to the standard neural networks confidence score.

3. Related work

There exist many solutions to find the posterior distribution of complex models such as Bayesian neural networks. Among them, it is worth citing the work regarding the Laplace approximation (Ritter et al., 2018), that approximates the posterior with a Gaussian distribution, and the Markov Chain Monte Carlo methods (Neal et al., 2011), that approximate the posterior by directly sampling from it. The most relevant of such techniques is dropout (Hinton et al., 2012), a regularization technique that, when active both at training and test time, has been proved to be equivalent to the approximation of a Bayesian neural network (Gal and Ghahramani, 2016).

Dropout has been extensively used to approximate Bayesian neural networks because of its ease of implementation and retrieval of predictive uncertainty estimates. Moreover, dropout has shorter training and prediction times when compared to the previously mentioned approaches. For these reasons, dropout based Bayesian uncertainty measures have also been used to perform biomedical image analysis and prediction (Dürr et al., 2018; Leibig et al., 2017). However recent work has exposed the main limitations of such approach, mainly related to the use of improper priors and the correctness of the variational objective (Hron et al., 2018).

4. Results

In this section, we compare the Bayesian approach and the standard one on the MNIST dataset (LeCun and Cortes, 1998), we analyze the predictive uncertainty to find out-of-distribution data from a closely related dataset (EMNIST by Cohen et al. (2017)), and we validate our method against cellular microscopy images. The architectures of the neural networks and the corresponding training hyperparameters are discussed in Appendix C.

4.1. Standard and Bayesian neural networks comparison

We first trained a Bayesian convolutional neural network (LeNet5) using the MNIST training set and tested the model on the corresponding test set with $S = 100$ output samples (see Appendix B for discussion of the number of predictive samples). As demonstrated by predictions for 200 example images (Figure 1(a)), some images have a consistent prediction, while others produce more than one classification result over the range of samples, which suggests low confidence. Overall, predictions performed using the output samples individually showed good classification accuracy with small fluctuations (Figure 1(b)). For each image, we computed the Bayesian prediction by taking the average of the output samples as explained in the Methods section and illustrated in Figure 1(c) and Figure 1(d).

We then compared the Bayesian prediction results with a standard neural network trained on the same MNIST data. As shown in Figure 1(e), confidence scores from the standard neural network tend to be close to 1 and there is no differentiation between correct and wrong predictions. In contrast, Bayesian confidence scores are high for correct

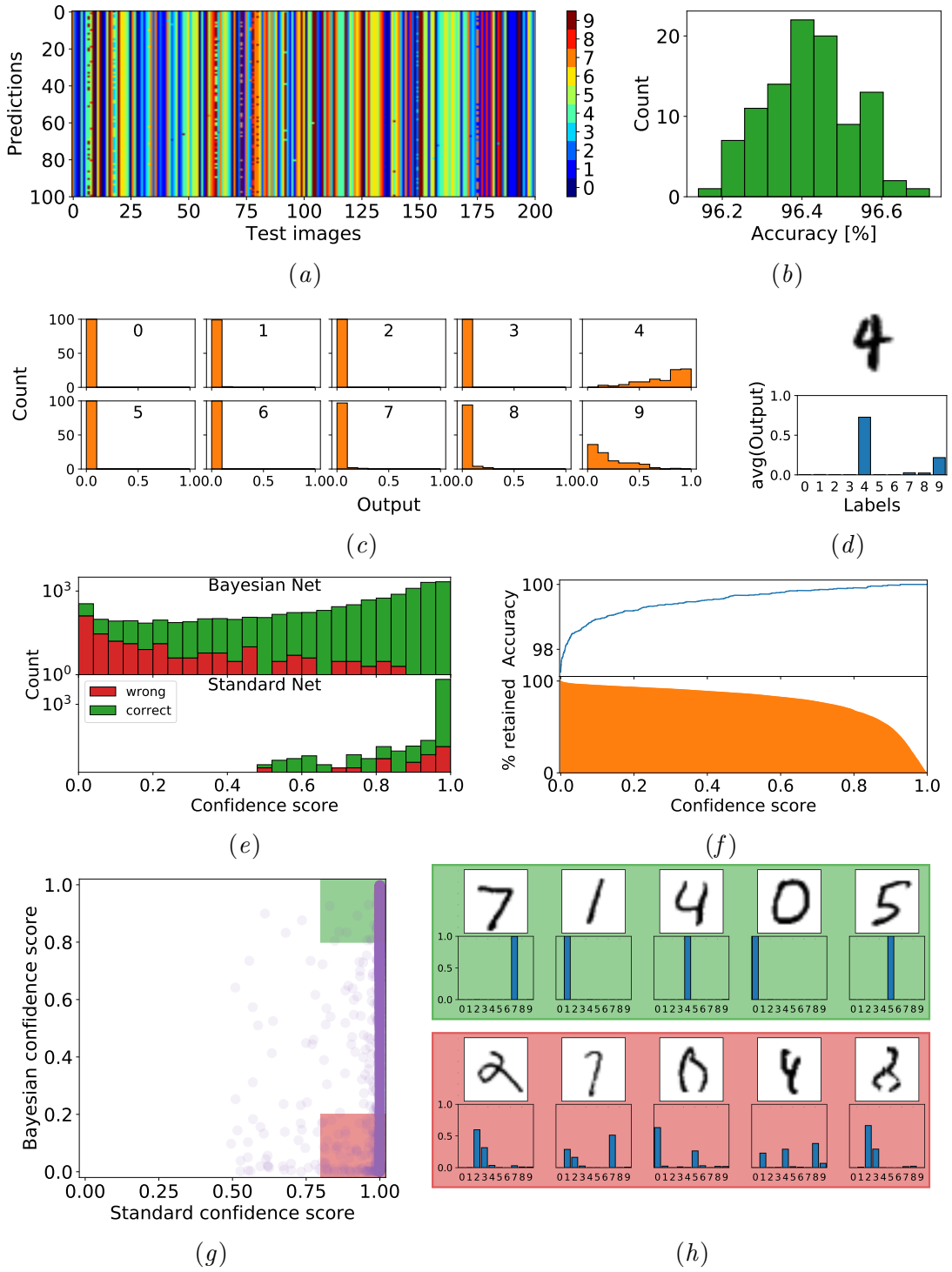
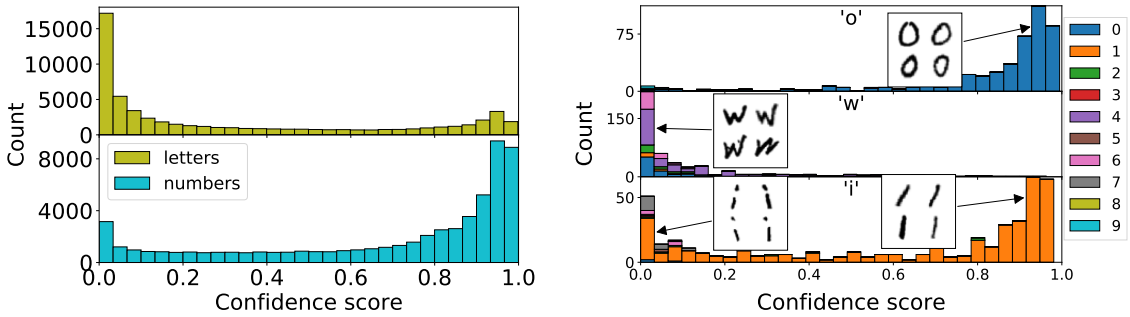


Figure 1: (a) Predictions from 100 output samples of 200 test images; (b) Corresponding histogram of accuracy; (c) Per class histograms of 100 output samples; (d) Corresponding final output for a test image of an ambiguous 4; (e) Confidence score distribution over the MNIST test set; (f) Increasing Bayesian confidence score cutoff increases accuracy while decreases the percentage of retained images; (g) Comparison of Bayesian and standard confidence scores over the MNIST test set; (h) Image samples, with related outputs, predicted with the corresponding colored areas confidence scores.

predictions and low for incorrect ones. Therefore it is possible to improve the overall accuracy by increasing the confidence threshold and retaining only high-confidence predictions as illustrated in Figure 1(f)). We also plotted the confidence scores from the Bayesian neural network and the standard neural network against each other for each test image (Figure 1(g)). By examining individual images in Figure 1(h), we see that images with high Bayesian confidence are the canonical digits while images with low Bayesian confidence correspond to corrupted or ambiguous observations. The standard neural network however is not able to distinguish them.

4.2. Out of distribution data

The EMNIST dataset is designed with the same image format as MNIST, but it expands to include hand-written letters. In order to validate the capability of the model to identify out-of-distribution samples — that is, images that cannot be labeled with any of the possible classes — we performed predictions over the EMNIST dataset with the Bayesian neural network previously trained on the MNIST dataset. As shown in Figure 2(a), the model predicts most numbers with high confidence, while it predicts letters with low confidence as they do not belong to the domain of the MNIST training data. We examined the confidence score distribution of each letter and illustrated in Figure 2(b) three representative examples. As expected, the letter "o" is predicted as 0 with high confidence while the letter "w", that does not resemble any of the digits, is predicted with low confidence. Interestingly, the confidence scores of letter "i", show a bimodal distribution. After manually checking individual images, we realized that some "i"s are very similar to the number 1 and are predicted with high confidence, while other "i"s are written with a dot on top, therefore considered as out-of-distribution samples with low confidence predictions.



(a) (b)

Figure 2: (a) Confidence score distribution of letters and numbers in EMNIST; (b) Three example letters: "o", "w" and "i".

4.3. Cellular microscopy images

As part of the Broad Bioimage Benchmark Collection (BBBC), the BBBC021 dataset is made of microscopy images of human MCF-7 breast cancer cells treated with 113 compounds

over eight concentrations and labeled with fluorescent markers for DNA, F-actin, and β-tubulin (Ljosa et al., 2012; Caie et al., 2010). A subset of BBBC021 with 38 compounds is annotated with a known mechanism of action (MoA) and it has been widely used to benchmark diverse analysis methods (Ljosa et al., 2013; Kandaswamy et al., 2016; Godinez et al., 2017; Ando et al., 2017). The MoA annotations come both from visual inspection by experts as well as scientific literature, using a course-grained set of 12 MoA, with each MoA containing multiple compounds and concentrations.

We applied the Bayesian approach, as described above, to a simplified version of the Multi-Scale Convolutional Neural Network (MSCNN) previous designed by Godinez et al. (2017). For validation, we adopted the rigorous leave-one-compound-out process, where in each session, all except one compound are used to train the model and the hold-out compound is used for validation. Figure 3(a) illustrates the predictive confidence for all hold-out compounds. As expected, the wrong predictions distribution has a peak on the low confidence side while the correct predictions distribution has a peak on the high confidence side. Thus with increasing threshold, one can improve overall classification accuracy as shown in Figure 6(a) in Appendix D.

The effects of compound treatment are complex due to how compounds interact with one or multiple protein targets and how these interactions are reflected on cell morphology labeled with fluorescent markers. For this reason, we further examined the Bayesian confidence scores for each of the 12 MoA and displayed two of the most relevant ones in Figure 3(b).

Observations belonging to the "protein synthesis" MoA (96 images and three compounds) are mainly predicted correctly. As expected, the confidence score of the four images predicted incorrectly is 0.04 ± 0.05 (median $\pm$ median absolute deviation), substantially lower than the correctly predicted images at 0.92 ± 0.10. Moreover, such images are considerably different that the rest, in fact they are mostly black, probably due to noise during the acquisition and annotation processes.

Similarly, incorrect predictions of "microtubule destabilizers", show a different morphology than the expected one. These anomalous images correspond to cells treated with colchicine, one of the 4 compounds associated with this MoA (see Figure 6(b) in Appendix D). This observation is consistent with the results of unsupervised approaches on the same dataset (Godinez et al., 2018) and indicates an error in the ground truth annotations. Furthermore, most of the 168 images of microtubule destabilizers are predicted correctly but some of them have low confidence scores. This is due to the corrupted observations from colchicine, that are used to train the model in order to validate the other compounds.

5. Discussion

The performance of deep neural networks in computer vision tasks has been explored for biological research and drug development. Compared with natural images, biomedical images have various challenges, such as noisy labels and out-of-distribution samples. To address these challenges, we have implemented a Bayesian neural network with variational inference and exploited the confidence score derived from the predictive variance of the model. Such uncertainty measure proved to be more precise than the standard neural networks confidence score on simple, well known, benchmark datasets, as well as complex biomedical

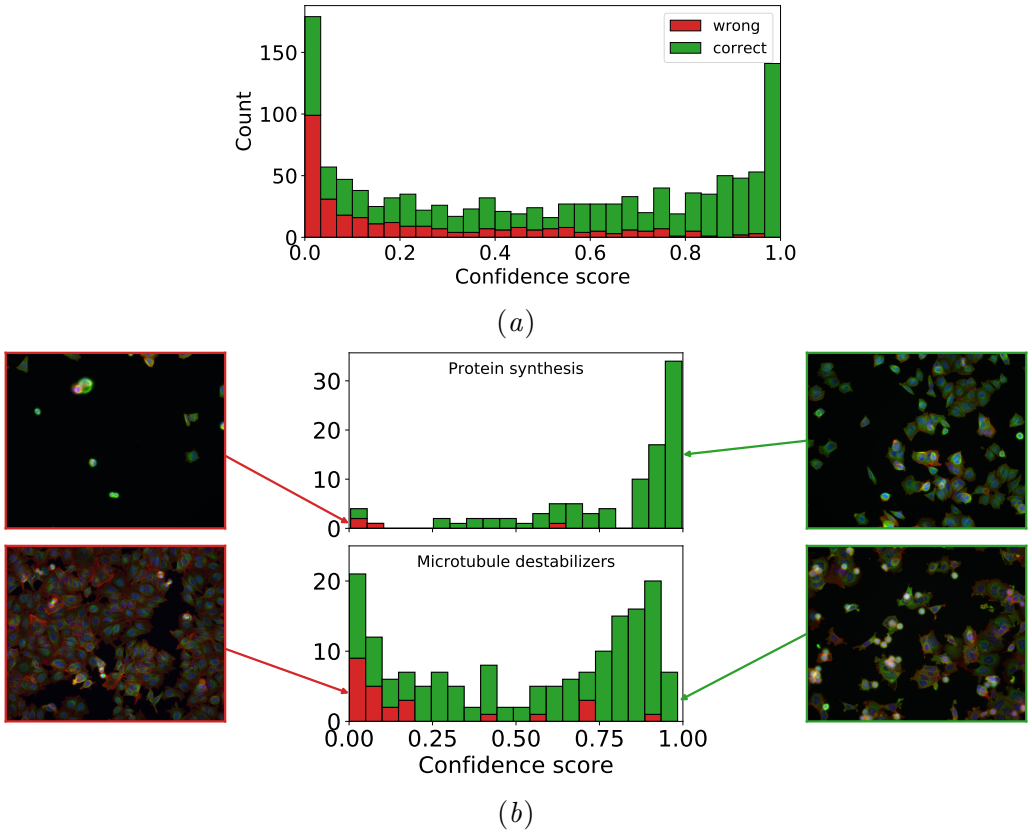


Figure 3: (a) Confidence score distribution of all hold-out compounds from BBBC021; (b) Two representative classes: protein synthesis and microtubule destabilizers, and the corresponding example images.

ones. Moreover, we were able to identify multiple anomalies in cellular images and the corresponding annotations, therefore we believe this Bayesian neural network approach has added value to the field of biomedical image classification.

Ground-truth labels for biomedical images are often impossible or prohibitively expensive to generate, which is why the field has moved towards unsupervised approaches. We envision future applications of the Bayesian framework to clustering and unlabeled images retrieval.

Acknowledgments

GD would like to acknowledge Prof. Maurizio Filippone (Eurecom, Department of Data Science) for the supervision of GD's Master thesis[1] from which this work derives.

1. https://webthesis.biblio.polito.it/10920/1/tesi.pdf

References

D. Michael Ando, Cory Y. McLean, and Marc Berndl. Improving phenotypic measurements in high-content imaging screens. *bioRxiv*, 2017. doi: 10.1101/161422. URL https://www.biorxiv.org/content/early/2017/07/10/161422.

José Manuel Benítez, Juan Luis Castro, and Ignacio Requena. Are artificial neural networks black boxes? *IEEE Transactions on neural networks*, 8(5):1156–1164, 1997.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

Peter D. Caie, Rebecca E. Walls, Alexandra Ingleston-Orme, Sandeep Daya, Tom Houslay, Rob Eagle, Mark E. Roberts, and Neil O. Carragher. High-content phenotypic profiling of drug response signatures across distinct cancer cells. *Molecular Cancer Therapeutics*, 9(6):1913–1926, jun 2010. ISSN 15357163. doi: 10.1158/1535-7163.MCT-09-1148.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009.

Oliver Dürr, Elvis Murina, Daniel Siegismund, Vasily Tolkachev, Stephan Steigele, and Beate Sick. Know when you don't know: A robust deep learning approach in the presence of unknown phenotypes. *Assay and drug development technologies*, 16(6):343–349, 2018.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016.

Zoubin Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521 (7553):452, 2015.

William J Godinez, Imtiaz Hossain, Stanley E Lazic, John W Davies, and Xian Zhang. A multi-scale convolutional neural network for phenotyping high-content cellular images. *Bioinformatics*, 33(13):2010–2019, 2017.

William J. Godinez, Imtiaz Hossain, and Xian Zhang. Unsupervised phenotypic analysis of cellular images with multi-scale convolutional neural networks. *bioRxiv*, 2018. doi: 10.1101/361410. URL https://www.biorxiv.org/content/early/2018/07/03/361410.

Alex Graves. Practical variational inference for neural networks. In *Advances in neural information processing systems*, pages 2348–2356, 2011.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Jiri Hron, Alexander G de G Matthews, and Zoubin Ghahramani. Variational bayesian dropout: pitfalls and fixes. *arXiv preprint arXiv:1807.01969*, 2018.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Chetak Kandaswamy, Luís M Silva, Luís A Alexandre, and Jorge M Santos. High-Content Analysis of Breast Cancer Using Single-Cell Deep Transfer Learning. *Journal of biomolecular screening*, 21(3):252–9, 2016. ISSN 1552-454X. doi: 10.1177/1087057115623451. URL http://jbx.sagepub.com/content/21/3/252.abstract.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Durk P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. In *Advances in Neural Information Processing Systems*, pages 2575–2583, 2015.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Yongchan Kwon, Joong-Ho Won, Beom Joon Kim, and Myunghee Paik. Uncertainty quantification using bayesian neural networks in classification: Application to ischemic stroke lesion segmentation. In *Medical Imaging with Deep Learning*, 04 2018.

Yann LeCun and Corinna Cortes. The MNIST database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

Christian Leibig, Vaneeda Allken, Murat Seçkin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific reports*, 7(1):17816, 2017.

Vebjorn Ljosa, Katherine L. Sokolnicki, and Anne E. Carpenter. Annotated high-throughput microscopy image sets for validation, jul 2012. ISSN 15487091.

Vebjorn Ljosa, Peter D Caie, Rob ter Horst, Katherine L Sokolnicki, Emma L Jenkins, Sandeep Daya, Mark E Roberts, Thouis R Jones, Shantanu Singh, Auguste Genovesio, Paul A Clemons, Neil O Carragher, and Anne E Carpenter. Comparison of Methods for Image-Based Profiling of Cellular Morphological Responses to Small-Molecule Treatment. *Journal of Biomolecular Screening*, 18(10):1321–1329, dec 2013. ISSN 1087-0571.

doi: 10.1177/1087057113503553. URL http://journals.sagepub.com/doi/10.1177/1087057113503553.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, pages 4765–4774, 2017.

Stephan Mandt, Matthew Hoffman, and David Blei. A variational analysis of stochastic gradient algorithms. In *International Conference on Machine Learning*, pages 354–363, 2016.

Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian inference. *The Journal of Machine Learning Research*, 18(1): 4873–4907, 2017.

Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2, 2011.

Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable laplace approximation for neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Skdvd2xAZ.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *arXiv preprint arXiv:1704.02685*, 2017.

Appendix A. Evidence Lower Bound derivation

Variational inference employs the exclusive version of the KL divergence. Since the posterior distribution is not known, a different objective needs to be defined starting from the logarithm of the marginal likelihood of the model:

$$\log\, p(\mathcal{D}) = \log \int p(\mathcal{D}, \mathbf{w}) d\mathbf{w}$$

The computation of the marginal likelihood is the core issue of Bayes theorem, in order to proceed we use an auxiliary function which corresponds to the proposal for the posterior approximation $q(\mathbf{w}|\theta)$ that we write as $q(\mathbf{w})$ to simplify the notation:

$$\log\, p(\mathcal{D}) = \log \int q(\mathbf{w}) \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} d\mathbf{w} = \log\, \mathbb{E}_{q(\mathbf{w})} \left\{ \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} \right\}$$

Then, it is needed to bring the logarithm inside the expectation which can be done by applying the Jensen's inequality:

$$\log\, \mathbb{E}_{q(\mathbf{w})} \left\{ \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} \right\} \geq \mathbb{E}_{q(\mathbf{w})} \left\{ \log \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} \right\}$$

Since we are now using an inequality, its right term is a lower bound of the logarithm of the marginal likelihood, also called model evidence, hence the name Evidence Lower Bound (ELBO). The ELBO can now be reformulated in the following way:

$$\mathbb{E}_{q(\mathbf{w})} \left\{ \log \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} \right\} = \int q(\mathbf{w}) \log \frac{p(\mathcal{D}, \mathbf{w})}{q(\mathbf{w})} d\mathbf{w} = \int q(\mathbf{w}) \log \frac{p(\mathcal{D}|\mathbf{w}) p(\mathbf{w})}{q(\mathbf{w})} d\mathbf{w} =$$

$$= \int q(\mathbf{w}) \log\, p(\mathcal{D}|\mathbf{w}) d\mathbf{w} - \int q(\mathbf{w}) \log \frac{p(\mathbf{w})}{q(\mathbf{w})} d\mathbf{w} =$$
$$= \mathbb{E}_{q(\mathbf{w})}\{\log\, p(\mathcal{D}|\mathbf{w})\} - KL\{q(\mathbf{w})||p(\mathbf{w})\} = \mathcal{L}(\theta)$$

Maximizing this lower bound with respect to the variational parameters θ of $q(\mathbf{w};\theta)$ provides a value as close as possible to the logarithm of the marginal likelihood and it is equivalent to minimizing the initial KL divergence between $q(\mathbf{w})$ and $p(\mathbf{w}|\mathcal{D})$:

$$KL\{q(\mathbf{w})||p(\mathbf{w}|\mathcal{D})\} = \int q(\mathbf{w}) \log \frac{q(\mathbf{w})}{p(\mathbf{w}|\mathcal{D})} d\mathbf{w} = \int q(\mathbf{w}) \log \frac{q(\mathbf{w}) p(\mathcal{D})}{p(\mathcal{D}|\mathbf{w}) p(\mathbf{w})} d\mathbf{w} =$$
$$= -\int q(\mathbf{w}) \log\, p(\mathcal{D}|\mathbf{w}) d\mathbf{w} + \int q(\mathbf{w}) \log \frac{q(\mathbf{w})}{p(\mathbf{w})} d\mathbf{w} + \int q(\mathbf{w}) \log\, p(\mathcal{D}) d\mathbf{w} =$$
$$= -\mathbb{E}_{q(\mathbf{w})}\{\log\, p(\mathcal{D}|\mathbf{w})\} + KL\{q(\mathbf{w})||p(\mathbf{w})\} + \log\, p(\mathcal{D})$$
$$KL\{q(\mathbf{w})||p(\mathbf{w}|\mathcal{D})\} = -\mathcal{L}(\theta) + \log\, p(\mathcal{D})$$

The marginal likelihood is constant, therefore maximizing the ELBO is equivalent to minimizing the KL divergence between the posterior and its approximation.

Appendix B. Number of predictive samples

The predictive distribution is approximated using Monte Carlo samples, therefore, depending on the number of outputs averaged to compute the final prediction, the corresponding accuracy can be more or less precise. Figure 4 shows 100 accuracies per numbers of samples used to compute the output, and the corresponding median and variance per number of samples. As the number of samples increases we get a closer estimate of the real accuracy value and its variance decreases. Figure 4 data correspond to a Bayesian neural network trained on the MNIST dataset.

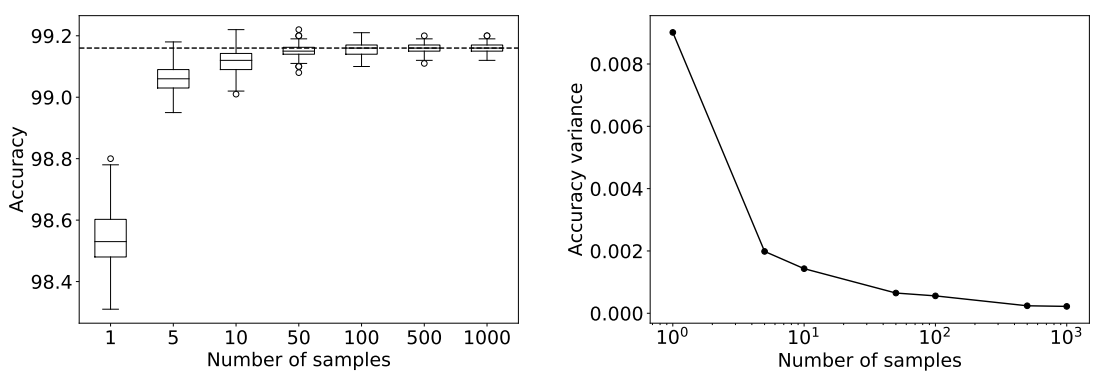


Figure 4: (left) Boxplot of 100 values of the accuracy of a Bayesian neural network per different number of samples and (right) the variance per number of samples. The dashed line corresponds to the median of the accuracies with 1000 samples.

Appendix C. Neural networks architectures and training hyperparameters

The architectures used for the experiments are illustrated in Figure 5. We used the same architecture (LeNet 5) for both the standard and Bayesian case in order to be consistent when performing comparison. We performed the optimization of the neural networks parameters using the Adam optimizer (Kingma and Ba, 2014). The training hyperparameters for all the architectures used are illustrated in Table 1.

In order to train the standard LeNet5 we used the cross entropy loss function and a learning rate schedule consisting of a single step at epoch 170, decreasing the learning rate by a factor of 10. Furthermore, we added weight decay with a factor of 5×10^{-3}. It is important to underline that this is not the best model to train on the MNIST dataset, this implementation choice has been taken because the model is still able to perform very well and, since the dataset is contextually simple, further improvements could lead to higher accuracies but complicate the comparison with the Bayesian counterpart.

Given the different nature of Bayesian LeNet5 with respect to the standard version, it has been trained with a lower batch size and for more epochs, with a single learning rate step at 200 epochs. The loss function is the variational objective, namely the ELBO loss. No weight decay has been applied because the regularization is already imposed by the shape and parameters of the prior.

In order to train the MSCNN, we adopted a weighted loss function because of the BBBC021 classes imbalance. The weights of this loss were computed as the inverse of the number of observations for each class, so that all the classes have the same influence on the updates of the model parameters. We finally trained the neural network using a learning rate schedule consisting of a single step at epoch 60.

Neural network architecture	**Dataset**	**Learning rate**	**Batch size**	**# Epochs**
Standard LeNet5	MNIST	10^{-3}	256	200
Bayesian LeNet5	MNIST	10^{-3}	32	300
Bayesian MSCNN	BBBC021	10^{-2}	20	80

Table 1: Training hyperparameters of the different architectures used in the experiments.

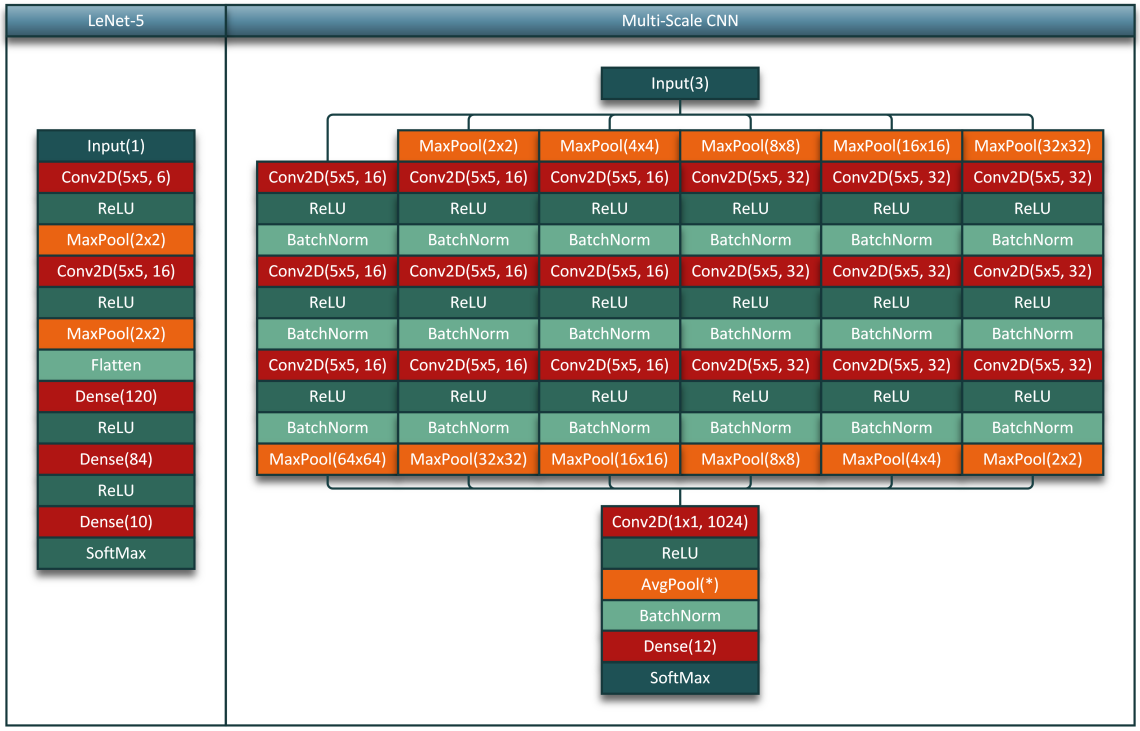
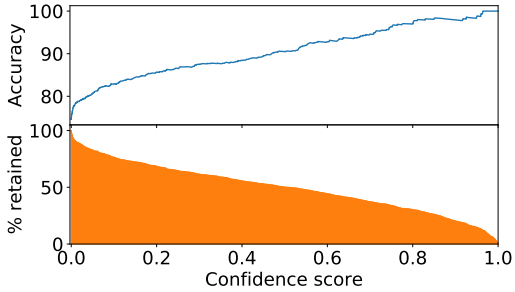
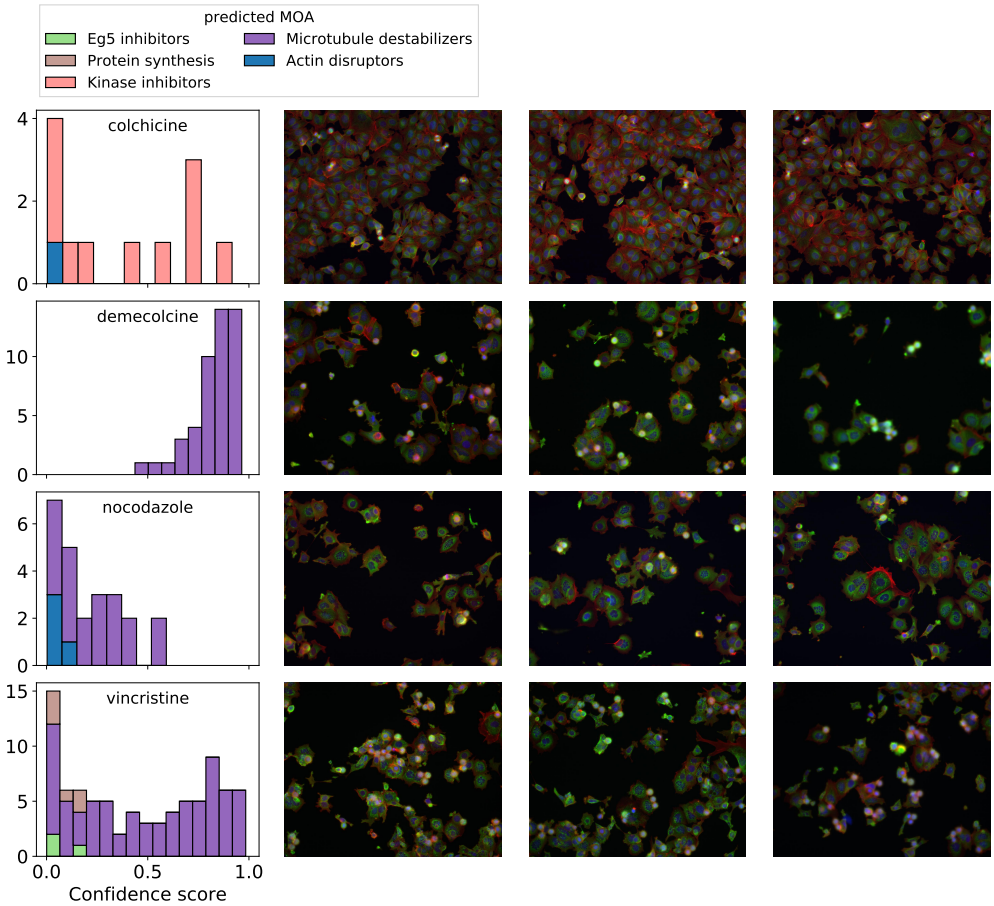


Figure 5: Neural networks architectures.

Appendix D. BBBC021 analysis. Supplementary figures



(*a*)



(*b*)

Figure 6: (a) Confidence score threshold analysis for BBBC021 predictions (same as Figure 1(*f*); (b) Confidence score distribution of each compound belonging to the microtubule destabilizers MoA and corresponding image samples.