Computational Statistics and Data Analysis 142 (2020) 106816

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Uncertainty quantification using Bayesian neural networks in classification: Application to biomedical image segmentation

Yongchan Kwon [a], Joong-Ho Won [a], Beom Joon Kim [b], Myunghee Cho Paik [a,*]

[a] Department of Statistics, Seoul National University, Seoul, 08826, South Korea
[b] Department of Neurology and Cerebrovascular Center, Seoul National University Bundang Hospital, Bundang, 13620, South Korea

ARTICLE INFO

Article history:
Received 8 October 2018
Received in revised form 16 July 2019
Accepted 17 July 2019
Available online 28 July 2019

Keywords:
Aleatoric and epistemic uncertainty
Bayesian neural network
Ischemic stroke lesion segmentation
Retinal blood vessel segmentation
Uncertainty quantification

ABSTRACT

Most recent research of deep neural networks in the field of computer vision has focused on improving performances of point predictions by developing network architectures or learning algorithms. Reliable uncertainty quantification accompanied by point estimation can lead to a more informed decision, and the quality of prediction can be improved. In this paper, we invoke a Bayesian neural network and propose a natural way of quantifying uncertainties in classification problems by decomposing the moment-based predictive uncertainty into two parts: aleatoric and epistemic uncertainty. The proposed method takes into account the discrete nature of the outcome, yielding the correct interpretation of each uncertainty. We demonstrate that the proposed uncertainty quantification method provides additional insights into the point prediction using two Ischemic Stroke Lesion Segmentation Challenge datasets and the Digital Retinal Images for Vessel Extraction dataset.

1. Introduction

While deep convolutional neural networks have shown outstanding performances in many different computer vision tasks and the field of biomedical imaging analysis (Esteva et al., 2017; Litjens et al., 2017), relatively less attention has been paid to assessing uncertainty in neural network outputs. Neural networks are prone to overfitting, and making decisions based on point prediction alone may provide incorrect classifications with spuriously high confidence (Guo et al., 2017; Pereyra et al., 2017). Reliable uncertainty quantification accompanied by point estimation can lead to a more informed decision, and the quality of prediction can be improved (Gal, 2016; Kendall and Gal, 2017). Improved decision support system bears significance in many applications, especially in healthcare.

Quantifying uncertainties have been major interests in the field of Bayesian analysis. Bayesian methods allow rich probabilistic interpretations for predicted outcomes via a posterior distribution. Although exact Bayesian inference with deep neural networks have been considered to be computationally intractable, Gal (2016) recently showed that a typical optimization of neural networks with dropout layers is equivalent to Bayesian variational inference with a specific variational distribution. Gal (2016) further showed that by randomly drawing Bernoulli random variables in dropout layers at inference time, one could readily obtain a variational predictive distribution. Teye et al. (2018) took a similar approach introducing random elements in batch normalization layers. Both methods provide practical solutions to training Bayesian neural network models and estimating a predictive uncertainty.

* Corresponding author.
E-mail addresses: ykwon0407@snu.ac.kr (Y. Kwon), wonj@stats.snu.ac.kr (J.-H. Won), kim.bj.stroke@gmail.com (B.J. Kim), myungheechopaik@snu.ac.kr (M.C. Paik).

https://doi.org/10.1016/j.csda.2019.106816
0167-9473/

In medical imaging literature, Leibig et al. (2017) used the moment based predictive uncertainties by Gal (2016) for the dataset from the Kaggle diabetic retinopathy detection competition (Kaggle, 2015). Using the same Kaggle dataset, Ayhan and Berens (2018) applied the method by Teye et al. (2018). Jungo et al. (2018) computed entropy based predictive uncertainties and illustrated how uncertainty measures can be used for improving prediction for a clinical brain tumor dataset.

In Bayesian modeling, predictive uncertainty can be decomposed into two different sources. Uncertainty can be expressed as aleatoric, capturing inherent randomness in the observations, and epistemic, accounting for model uncertainty (Der Kiureghian and Ditlevsen, 2009). Recently, Kendall and Gal (2017) proposed a moment-based uncertainty decomposition method by explicitly modeling the variability of last layer outputs of a neural network. Tanno et al. (2017) independently proposed the same method as Kendall and Gal (2017) and applied it to brain diffusion tensor images, although they neither emphasized the decomposition nor used the terminology of aleatoric or epistemic uncertainty.

In medical imaging, decomposition to aleatoric and epistemic uncertainties can provide extra information as follows. In the process of classifying images into normal or abnormal, an expert can identify distinctive features in an image and weigh each feature to judge the likelihood that the image is abnormal. If this likelihood or probability is near 0 or 1, the variability of binary indicators of declaring abnormal is low. For a tough image to evaluate, the probability can be near 0.5, the variability of declaring normal or abnormal would be high. This inherent variability can be captured by aleatoric uncertainty. On the other hand, experts may come up with different distinctive features and weigh differently for each evaluation leading to different likelihood to declare that the image is abnormal. This type of variability can be captured by epistemic uncertainty.

Existing studies on the uncertainty quantification for classification via deep neural network have utilized extra parameters for variances without reflecting the functional relationship between a mean and a variance of multinomial random variables. Furthermore, few studies considered decomposition of aleatoric and epistemic uncertainties in classification taking into account the relationship between mean and variance. In this paper, we suggest a natural way to decomposing uncertainties in classification settings. We show that the proposed method considers the functional relationship and correlation structures among classes.

Our main contributions of this paper are:

1. We propose a new method of quantifying uncertainties in classification based on Bayesian neural network models. Our method exploits the relationship between the variance and the mean of a multinomial random variable and avoids estimation of extra parameters for the variance.
2. We illustrate the proposed method using two Ischemic Stroke Lesion Segmentation Challenge (ISLES) datasets and the Digital Retinal Images for Vessel Extraction (DRIVE) dataset. Our results demonstrate interpretability of uncertainty maps and exhibit dependence between the epistemic uncertainty and the effective sample size. In addition, an implication of the number of realized posterior samples is examined.

2. Bayesian neural networks

In this section, we describe Bayesian neural networks and two main learning methods. Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ be a realization of independently and identically distributed random variables where $x_i \in \mathbb{R}^d$ and $y_i \in \{e_1, \ldots, e_K\}$ are the ith input and output, respectively, where e_k is the one-hot encoded vector with the only kth element being one and zero for the other elements. The number of different classes is denoted by K, the sample size, by N, and the dimension of input variables, by d.

We first formalize the model likelihood with a neural network model with L hidden layers. For $j \in \{1, \ldots, L+1\}$, let $h_{[j]} \in \mathbb{R}^{d_j}$ be the jth layer pre-activated output defined by $h_{[j]} := f_{[j]}^{\omega_{[j]}}(h_{[j-1]}) := \omega_{[j]}\eta(h_{[j-1]})$ where $\omega_{[j]} \in \mathbb{R}^{d_j \times d_{j-1}}$ is a parameter matrix representing random weights in a Bayesian neural network. The $\eta(\cdot)$ is an element-wise non-linearity operator. That is, $\eta(v) = (\eta(v_1), \ldots, \eta(v_{d_v}))$ for $v \in \mathbb{R}^{d_v}$. Note that $x = h_{[0]}$ and $d = d_0$, and we do not include bias terms for notational convenience. By stacking the layers, the last pre-activated output of a neural network can be defined hierarchically by

$$f^{\omega}(x) := f_{[L+1]}^{\omega_{[L+1]}}(f_{[L]}^{\omega_{[L]}}(\ldots f_{[1]}^{\omega_{[1]}}(x)\ldots)),$$

where $\omega = (\mathrm{vec}(\omega_{[1]}), \ldots, \mathrm{vec}(\omega_{[L+1]}))$ is a whole parameter used in the neural network. Here, $\mathrm{vec}(M)$ is a vectorization of a matrix M. With the last K-dimensional pre-activated output $f^{\omega}(x) = (f_1^{\omega}(x), \ldots, f_K^{\omega}(x))$, the model likelihood $p(y \mid x, \omega)$ is given by

$$p(y = e_k \mid x, \omega) = p\{y = e_k \mid f^{\omega}(x)\} = \frac{\exp\{f_k^{\omega}(x)\}}{\sum_{j=1}^K \exp\{f_j^{\omega}(x)\}}.$$

Assuming the Bayesian neural network model, we place a prior distribution $p(\omega)$ on a parameter vector $\omega \in \Omega$, weights and bias vectors in a neural network, resulting posterior distribution

$$p(\omega \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \omega)p(\omega)}{p(\mathcal{D})} = \frac{\prod_{i=1}^N p(y_i \mid x_i, \omega)p(\omega)}{p(\mathcal{D})},$$

and the predictive distribution

$$p(y^* \mid x^*, \mathcal{D}) = \int_{\Omega} p(y^* \mid x^*, \omega) p(\omega \mid \mathcal{D}) d\omega,$$

for a new input x^* and a new output y^*.

Learning this model is often intractable because calculating the posterior $p(\omega \mid \mathcal{D})$ requires integration with respect to the whole parameter space Ω for which a closed form often does not exist. MacKay (1992) proposed the Laplace method approximating the posterior by the Gaussian with maximum a posteriori estimate mean and the inverse of the Hessian of the log-likelihood variance. However, the performance of this method is often limited due to poor approximation.

Two methods have been widely used to train Bayesian neural network models, Markov Chain Monte Carlo (MCMC) algorithm and variational inference. Neal (1993) introduced the Hamiltonian Monte Carlo, an MCMC sampling approach using Hamiltonian dynamics. This yields a principled set of posterior samples without direct calculation of the posterior. This method is often computationally prohibitive because MCMC sampling runs on a whole dataset and the sampled parameters are needed to be stored for inferences. Recently, practical MCMC methods (Welling and Teh, 2011; Chen et al., 2014) based on the first- or second-order Langevin dynamics have been studied. They exploited stochastic optimization techniques to efficiently sample from desired distributions. These stochastic MCMC methods scale well with data size, but they still require storing all sampled parameters from a posterior at each training iteration step, which might be inefficient in terms of memory usage if one uses very deep neural networks.

An alternative to MCMC approaches is variational inference (Graves, 2011; Blundell et al., 2015) which approximates the posterior distribution by a tractable variational distribution $q_\theta(\omega)$ indexed by variational parameter $\theta \in \Theta$ for variational parameter space Θ. The optimal variational distribution is the closest distribution to the posterior among the pre-determined family $Q = \{q_\theta(\omega) : \theta \in \Theta\}$. The closeness is often measured by the Kullback–Leibler (KL) divergence between the variational distribution $q_\theta(\omega)$ and the posterior $p(\omega \mid \mathcal{D})$ defined by

$$KL\{q_\theta(\omega) \parallel p(\omega \mid \mathcal{D})\} := \int_{\Omega} q_\theta(\omega) \log \frac{q_\theta(\omega)}{p(\omega \mid \mathcal{D})} d\omega.$$

Minimizing the KL divergence is equivalent to minimizing the negative evidence lower bound given by

$$-\int_{\Omega} q_\theta(\omega) \log p(y \mid x, \omega) d\omega + KL\{q_\theta(\omega) \parallel p(\omega)\}. \tag{1}$$

Variational inference converts standard Bayesian learning from integration to optimization problem. Due to this conversion, Bayesian inference can be conducted in mini-batch mode, whereas MCMC-based inference is suited for batch computation.

Since variational distribution $q_\theta(\omega)$ approximates to the posterior, the quality of approximation depends on the family of distributions Q. Restricted family Q would help scalability but hurt approximation. Graves (2011) and Blundell et al. (2015) specified $q_\theta(\omega)$ as product of normal distribution invoking a mean field approximation. The mean field approximation with normal distributions makes the problem scalable but normality assumption doubles the number of parameters due to mean and variance, which makes optimization more challenging. Recently, Gal (2016) proved that a gradient-based optimization procedure on the dropout neural network is equivalent to a specific variational approximation on a Bayesian neural network. Teye et al. (2018) also showed similar results for neural networks with batch normalization layers. In short, there are many practical learning methods for Bayesian neural networks. We leave the discussion on variety families Q in Section 6.

3. Uncertainty quantification in classification

In this section, we introduce existing uncertainty quantification methods in classification settings. We focus on the existing methods based on variational inference (Gal, 2016; Kendall and Gal, 2017).

We first define the variational predictive distribution, approximating the predictive distribution $p(y^* \mid x^*, \mathcal{D})$, given by

$$q_\theta(y^* \mid x^*) = \int_{\Omega} p(y^* \mid x^*, \omega) q_\theta(\omega) d\omega,$$

for all $\theta \in \Theta$. Since this quantity is not easy to compute due to the integration, we consider a Monte Carlo estimator given by

$$\hat{p}_\theta(y^* \mid x^*) = \frac{1}{T} \sum_{t=1}^{T} p(y^* \mid x^*, \omega_t),$$

where a set of realized vectors $\{\omega_t\}_{t=1}^T$ is randomly drawn from the optimized variational distribution $q_\theta(\omega)$ with the pre-defined sampling number T. By the weak law of large numbers, it converges in probability to $q_\theta(y^* \mid x^*)$ for all $\theta \in \Theta$.

Let $\hat{\theta}$ be the optimized variational parameter obtained by iteratively minimizing an empirical version of Eq. (1) (Kingma and Welling, 2013; Gal, 2016). The plugged-in estimator is given by

$$\hat{p}_{\hat{\theta}}(y^* \mid x^*) = \frac{1}{T}\sum_{t=1}^{T} p(y^* \mid x^*, \hat{\omega}_t),$$

where a set of realized vectors $\{\hat{\omega}_t\}_{t=1}^T$ is randomly drawn from the optimized variational distribution $q_{\hat{\theta}}(\omega)$. The following are several existing uncertainty quantification methods.

3.1. Entropy based predictive uncertainty

Gal (2016) introduced a Shannon-entropy-based predictive uncertainty for classification given by

$$\begin{aligned} &H\{p(y^* \mid x^*, \mathcal{D})\} \\ &:= -\sum_{k=1}^{K} p(y^* = e_k \mid x^*, \mathcal{D}) \log\{p(y^* = e_k \mid x^*, \mathcal{D})\}, \end{aligned}$$

and suggested an easy-to-compute estimator given by

$$-\sum_{k=1}^{K} \hat{p}_{\hat{\theta}}(y^* = e_k \mid x^*) \log\{\hat{p}_{\hat{\theta}}(y^* = e_k \mid x^*)\}. \tag{2}$$

Further, Gal (2016) showed this plugged-in estimator converges in probability to the variational entropy $H\{q_{\hat{\theta}}(y^* \mid x^*)\}$ when T goes to infinity.

3.2. Moment based predictive uncertainty

The moment based predictive uncertainty quantification approach evaluates the variance of an output y^* given x^* over the predictive distribution $p(y^* \mid x^*, \mathcal{D})$, defined by

$$\begin{aligned} &\mathrm{Var}_{p(y^* \mid x^*, \mathcal{D})}(y^*) \\ &:= \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}\{y^{*\otimes 2}\} - \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^*)^{\otimes 2} \\ &= \mathrm{diag}\{p(y^* \mid x^*, \mathcal{D})\} - p(y^* \mid x^*, \mathcal{D})^{\otimes 2}, \end{aligned}$$

where $v^{\otimes 2} = vv^T$, $\mathrm{diag}(v)$ is the diagonal matrix with elements of vector v, and $\mathbb{E}_{m(y^*)}\{g(y^*)\} = \int g(y^*)m(y^*)dy^*$ for any measurable function g and a probability density function $m(\cdot)$ for y^*. Then the corresponding plugged-in estimator is:

$$\mathrm{diag}\{\hat{p}_{\hat{\theta}}(y^* \mid x^*)\} - \hat{p}_{\hat{\theta}}(y^* \mid x^*)^{\otimes 2}. \tag{3}$$

It can be shown that this plugged-in estimator converges in probability to the variational variance $\mathrm{Var}_{q_{\hat{\theta}}(y^* \mid x^*)}(y^*)$ when T goes to infinity (Gal, 2016, Section 3.3.1).

3.3. Aleatoric uncertainty and epistemic uncertainty

The two quantification methods, Eqs. (2) and (3), capture an overall predictive uncertainty. The predictive uncertainty can be partitioned into aleatoric and epistemic uncertainties with respect to the source of uncertainty (Der Kiureghian and Ditlevsen, 2009).

Kendall and Gal (2017) developed a novel way to directly estimate these two types of uncertainties. They constructed a Bayesian neural network model with the last layer before activation (*i.e.* a softmax function) consisting of mean and variance of logits. We denote, by $f^{\omega}_{\mathrm{kendall}}(x^*) = (\mu^T, (\sigma^2)^T)^T$, the last $2K$-dimensional pre-activated linear output of the neural network, where $\mu \in \mathbb{R}^K$ and $\sigma^2 \in \mathbb{R}^K$ represent the mean and the variance of the K nodes. For the realized vectors $\{\hat{\omega}_t\}_{t=1}^T$ and the corresponding outputs $f^{\hat{\omega}_t}_{\mathrm{kendall}}(x^*) = (\hat{\mu}_t^T, (\hat{\sigma}_t^2)^T)^T$, they suggested an estimator of two types of uncertainty as

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} \mathrm{diag}(\hat{\sigma}_t^2)}_{\text{aleatoric}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T} (\hat{\mu}_t - \bar{\mu})^{\otimes 2}}_{\text{epistemic}}, \tag{4}$$

where $\bar{\mu} = \sum_{t=1}^{T} \hat{\mu}_t / T$.

Quantifying uncertainty by Eq. (4) has several limitations to be applied to classification problems. First, Eq. (4) models the variability of linear predictors, not the predictive probabilities. Second, it ignores the fact that the variance of a multinomial random variable is a function of the mean vector. For example, in binary classification, the predicted values

closer to 0 or 1 have a small variance. The third limitation is that diagonal modeling of the aleatoric uncertainty does not reflect negative correlations of class indicators due to mutual exclusiveness. In the next section, we propose a method to quantify and decompose the predictive uncertainty while avoiding the aforementioned problems.

4. Proposed method

The main idea of the proposed method is to decompose the variability of the predicted probability directly without introducing extra variability of the linear predictor of the last layer. Using the moment based uncertainty, we can decompose the predictive uncertainty $\mathrm{Var}_{p(y^*|x^*,\mathcal{D})}(y^*)$ into the two parts as follows

$$\begin{aligned}&\mathrm{Var}_{p(y^*|x^*,\mathcal{D})}(y^*)\\ &= \underbrace{\int_\Omega \mathrm{Var}_{p(y^*|x^*,\omega)}(y^*)p(\omega \mid \mathcal{D})d\omega}_{=:\text{ aleatoric}}\\ &+ \underbrace{\int_\Omega \left\{\mathbb{E}_{p(y^*|x^*,\omega)}(y^*) - \mathbb{E}_{p(y^*|x^*,\mathcal{D})}(y^*)\right\}^{\otimes 2} p(\omega \mid \mathcal{D})d\omega}_{=:\text{ epistemic}}.\end{aligned}$$

The detailed derivation is provided in Appendix A. The aleatoric uncertainty, $\mathbb{E}_{p(\omega|\mathcal{D})}[\mathrm{Var}_{p(y^*|x^*,\omega)}(y^*)]$, captures inherent randomness of an output y^*. The epistemic uncertainty, $\mathbb{E}_{p(\omega|\mathcal{D})}[\left\{\mathbb{E}_{p(y^*|x^*,\omega)}(y^*) - \mathbb{E}_{p(y^*|x^*,\mathcal{D})}(y^*)\right\}^{\otimes 2}]$, comes from the variability of ω given data. The randomness of ω causes forming different distinctive features and weighing each feature differently. This quantity can be reduced and the weighting scheme becomes less variable as the sample size increased (Lee, 2000; Wang and Blei, 2018).

We propose an analogous decomposition in variational inference setup. Let $q_\theta(y^* \mid x^*)$ be the variational distribution. The corresponding variational predictive uncertainty is

$$\begin{aligned}&\mathrm{Var}_{q_\theta(y^*|x^*)}(y^*)\\ &= \mathbb{E}_{q_\theta(y^*|x^*)}\{y^{*\otimes 2}\} - \mathbb{E}_{q_\theta(y^*|x^*)}(y^*)^{\otimes 2}\\ &= \underbrace{\int_\Omega \mathrm{Var}_{p(y^*|x^*,\omega)}(y^*)q_\theta(\omega)d\omega}_{=\text{ aleatoric}} \qquad (5)\\ &+ \underbrace{\int_\Omega \left\{\mathbb{E}_{p(y^*|x^*,\omega)}(y^*) - \mathbb{E}_{q(y^*|x^*)}(y^*)\right\}^{\otimes 2} q_\theta(\omega)d\omega}_{=\text{ epistemic}}, \qquad (6)\end{aligned}$$

for any $\theta \in \Theta$. In Eq. (5), the discrete nature of categorical variable y^* gives

$$\mathrm{Var}_{p(y^*|x^*,\omega)}(y^*) = \mathrm{diag}\{\mathbb{E}_{p(y^*|x^*,\omega)}(y^*)\} - \mathbb{E}_{p(y^*|x^*,\omega)}(y^*)^{\otimes 2}.$$

Once training is over and $\hat{\omega}$ is obtained, the proposed estimators of Eqs. (5) and (6) for given new image input x^* are

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T}\left[\mathrm{diag}\{p(y^* \mid x^*, \hat{\omega}_t)\} - p(y^* \mid x^*, \hat{\omega}_t)^{\otimes 2}\right]}_{\text{aleatoric}} \qquad (7)$$

$$+ \underbrace{\frac{1}{T}\sum_{t=1}^{T}\{p(y^* \mid x^*, \hat{\omega}_t) - \hat{p}_{\hat{\theta}}(y^* \mid x^*)\}^{\otimes 2}}_{\text{epistemic}}. \qquad (8)$$

Note that estimator, Eq. (7), incorporates the negative correlation structure among the classes. We can show that the estimators, Eqs. (7) and (8), converge in probability to Eqs. (5) and (6), respectively, as T increases.

Direct decomposition yields two main advantages, numerical stability and direct interpretation. Numerical instability of the method by Kendall and Gal (2017) mainly comes from divergence of σ^2. By eliminating σ^2, this problem can be avoided. Also, the dimension of the last layer of the proposed method is K, in contrast to $2K$ of the method by Kendall and Gal (2017), which leads to reduction of the amount of computation. While Kendall and Gal (2017) decomposed the variance of the linear predictor, the proposed method directly partition the variance of the predicted outcomes, which leads to meaningful interpretation of each term.

Table 1
Comparison of the SISS and SPES datasets. M, million; DWI, diffusion weighted imaging; FLAIR, fluid attenuation inversion recovery; T1, T1-weighted; T2, T2-weighted; TTP, time-to-peak; Tmax, time-to-max; CBV, cerebral blood volume; CBF, cerebral blood, flow; T1c, T1 contrast enhanced; Std, standard deviation.

	SISS	SPES
Number of subjects	28	30
Total number of voxels	227 M	22 M
Average dimension	$230 \times 230 \times 154$	$96 \times 110 \times 72$
MRI sequences	DWI, FLAIR T1, T2	TTP, Tmax, CBV CBF, T1c, T2, DWI
P(Stroke lesion) Mean (Std)	0.54% (0.73%)	2.51% (1.11%)

5. Experimental results

We use the two public medical imaging datasets, one is the Ischemic Stroke Lesion Segmentation (ISLES) Challenges datasets, and the other is the Digital Retinal Images for Vessel Extraction (DRIVE) dataset, to demonstrate the merits of the proposed method. Specifically, we compare the existing and the proposed uncertainty measures using the ISLES datasets, and examine the effect of sample size on the epistemic uncertainty using the ISLES and the DRIVE datasets. All the implementation details and the pointer to our Keras-based Python codes are provided in Appendix B.

5.1. Datasets

5.1.1. Ischemic stroke lesion segmentation (ISLES)

Ischemic stroke is a leading cause of death and is caused by an obstruction in the cerebral blood supply. The International Conference on the Medical Image Computing and Computer-Assisted Intervention has hosted the ISLES challenges annually since 2015 (Winzeck et al., 2018). We restrict our focus on the comparison of the two ISLES challenges, namely the ISLES 2015 acute stroke perfusion estimation (SPES) and the ISLES 2015 sub-acute ischemic stroke lesion segmentation (SISS). The main task for both competitions was constructing an automatic ischemic stroke lesion segmentation model. For both competitions, various 3-dimensional magnetic resonance imagings (MRIs) were provided such as FLAIR, DWI, T1, T2 and others listed in Table 1 and ground truth locating stroke lesion. The two datasets are briefly summarized in Table 1 and detailed information can be found in Kamnitsas et al. (2017), Maier et al. (2017), and the ISLES website.[1]

5.1.2. Digital Retinal Images for Vessel Extraction (DRIVE)

Staal et al. (2004) provided the DRIVE dataset allowing comparative studies on segmentation of blood vessels in retinal images. Detecting blood vessels in eyes is of a clinically important problem because it can be utilized in an automatic screening of numerous retinal diseases (Niemeijer et al., 2004; Siva Sundhara Raja and Vasuki, 2015). The dataset consists of 20 pairs of training and test datasets, respectively, of colored retina images and corresponding blood vessel annotations. Detailed information can be found in Staal et al. (2004) and Niemeijer et al. (2004) and the dataset is available on the DRIVE website.[2]

In our experiments, we used the randomly cropped retina images, small 2-dimensional patches, due to the high-resolution of the original images. Each cropped patch has a size of 96×96 pixels and some examples are presented in Fig. 1.

5.2. Comparison of the proposed and existing uncertainty quantification methods

First, we applied the proposed method and that by Kendall and Gal (2017) to the ISLES datasets to compare the uncertainty estimates of the two methods. Fig. 2 presents a slice of original MRI image, estimates of aleatory and epistemic uncertainties, and their sum labeled by "predictive," along with the predicted lesion and the ground truth. We present the FLAIR and DWI images as the original MRI image from SISS and SPES dataset, respectively, since the ground truth segmentation was made from respective imaging modes. These images are known to represent the core area of stroke well (Maier et al., 2017). Bright color represents large values. The rows labeled by 'P' and 'K' indicate the proposed and the method by Kendall and Gal (2017), respectively.

Although both methods produced similar predicted values, there were substantial differences in the uncertainty maps. In our experiments, the aleatoric uncertainty terms, the extra parameter in Kendall and Gal (2017) converged to zero and the resulting uncertainty image maps provided little information. The proposed maps show aleatoric, epistemic and

[1] http://www.isles-challenge.org/ISLES2015/.
[2] https://www.isi.uu.nl/Research/Databases/DRIVE/.

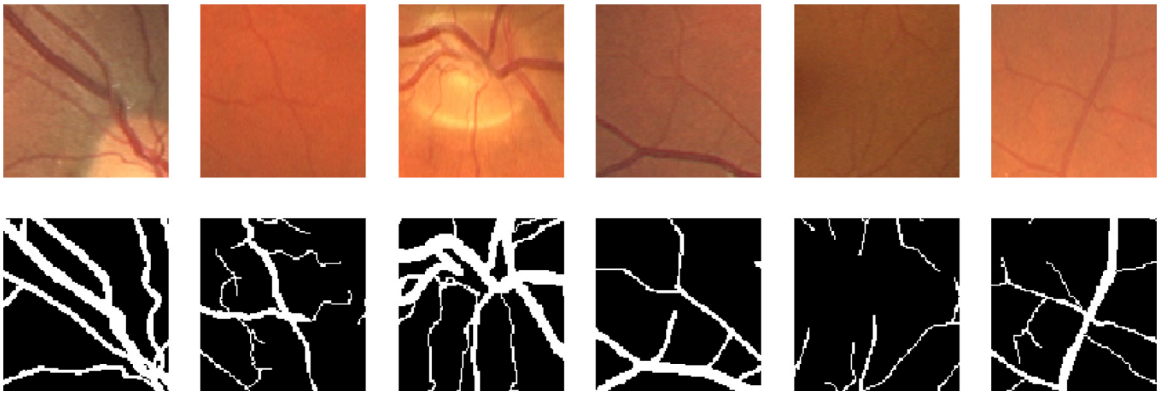

Fig. 1. Example of preprocessed images and labels. Images in the first row are retina images, used as input, and its annotations are, used as label, in the second row. Each column indicates different samples.

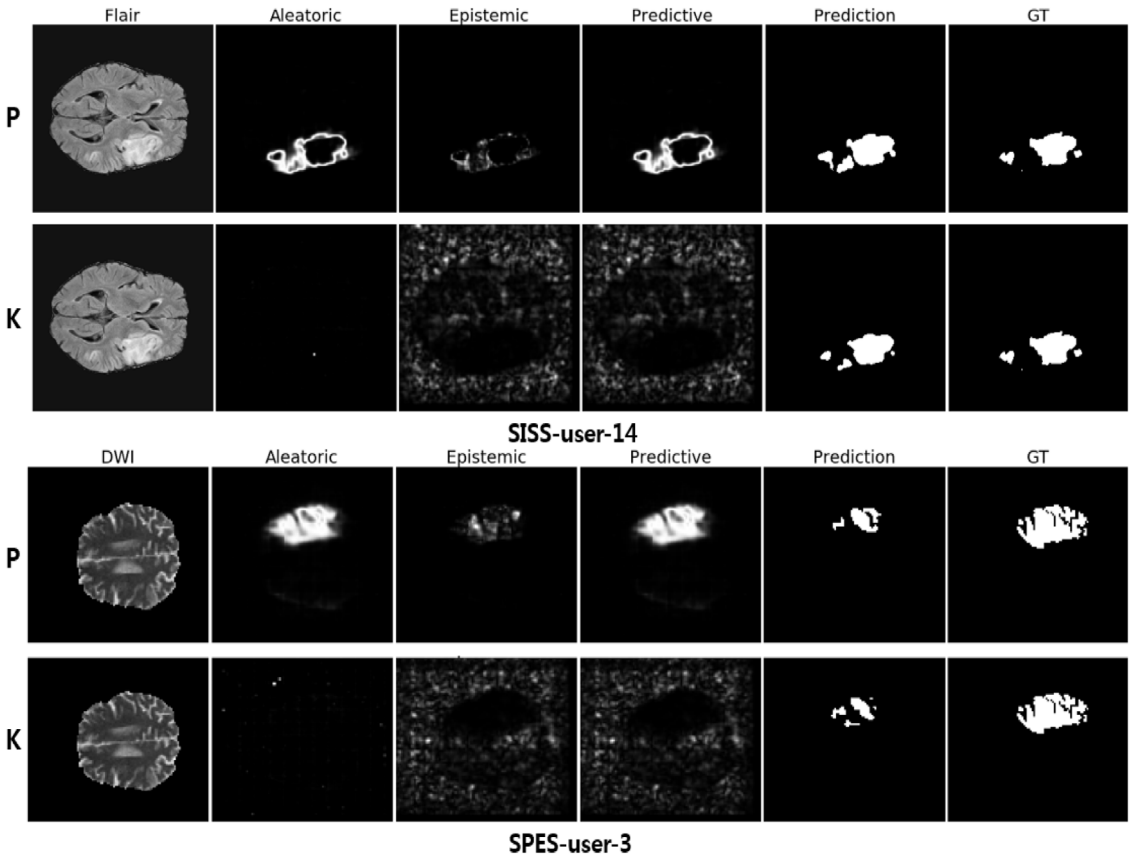

Fig. 2. Visual comparisons of uncertainty quantification methods using the user '14' of the SISS dataset (row 1 and 2) and the user '3' of the SPES dataset (row 3 and 4). Rows 'P' and 'K' indicate the proposed method and the uncertainty quantification method by Kendall and Gal (2017), respectively.

combined uncertainties. In the images shown in Fig. 2, the uncertainty maps provided extra information in addition to the prediction map. We can see that the main source of uncertainty came from the aleatoric part. Also, the boundaries were shown to be more uncertain than the interior or exterior regions. When the prediction map disagrees with the ground truth, the uncertainty maps identified the stroke region missed by prediction map. In SISS-user-14, the proposed prediction map incorrectly identified the non-stroke region as a stroke, and the uncertainty map reflected a lack of confidence around the incorrectly identified region. The prediction map from the SPES-user-3 partially failed to identify

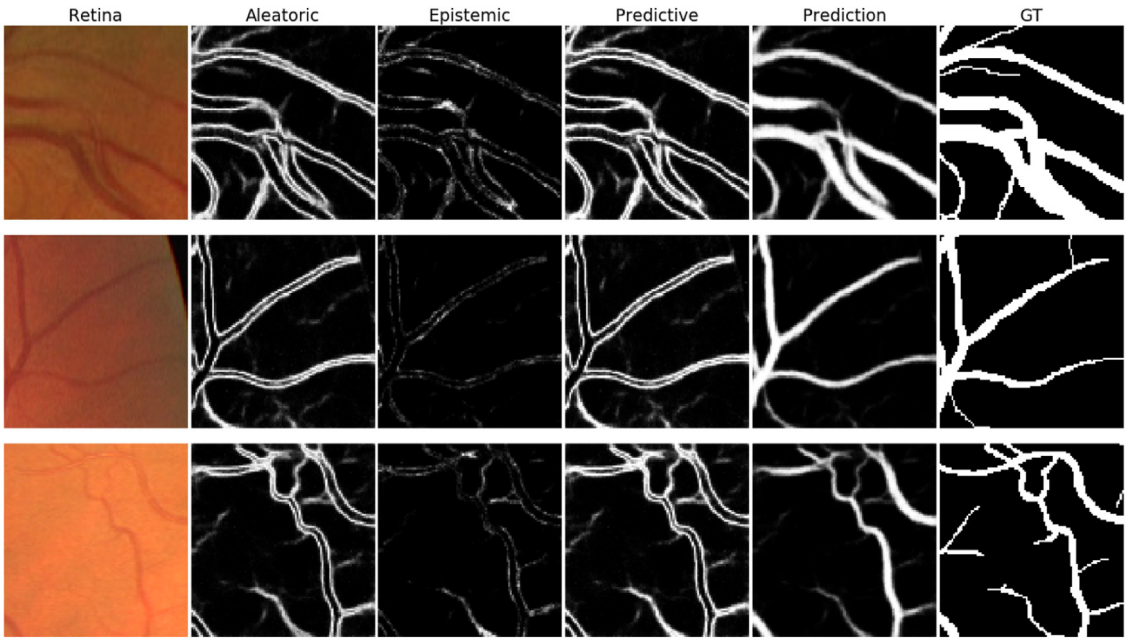

Fig. 3. Uncertainty quantification results using the DRIVE dataset. Each row indicates a different patch. First column shows the original retina image and the last column shows the corresponding blood vessel annotation.

some lesion locations but the uncertainty maps recovered the missed region close to the ground truth. Appendix C shows visual display of other images.

We conducted similar experiments using the DRIVE dataset, but the Kendall and Gal (2017) method failed to be trained due to a numerical divergence on the variance part. We show aleatoric and epistemic uncertainties using the proposed method in Fig. 3. More visual results are available in Appendix D.

5.3. Implication of difference in resolution or patch numbers

The main goal of this subsection is to evaluate the effects of sample size on epistemic uncertainties. In segmentation tasks, a unit of analysis is a voxel (or a pixel in 2-dimensional images) and the outcomes constitute high-dimensional multivariate binary random variables. The total number of voxels in medical imaging datasets can be determined by the number of subjects, the number of patches, and the resolution of each image. In many practical cases, one cannot control the number of subjects. In this subsection, we demonstrate two scenarios where the number of voxels is different through the quality of resolution using ISLES dataset, and through a different number of patches using DRIVE dataset.

Since the voxels are not independent, the number of voxels does not directly translate to the sample size. On the other hand, the voxels are not perfectly correlated, and the effective sample size is between the number of voxels and the number of subjects (Fleiss et al., 2003).

5.3.1. ISLES dataset

We compared aleatoric and epistemic uncertainties for SPES and SISS of the ISLES datasets. Although the number of subjects is comparable between SPES and SISS, SISS has a larger number of voxels due to higher quality resolutions of the images than SPES. Recall that aleatoric uncertainty is irreducible but the epistemic uncertainty decreases with the sample size. Therefore we anticipate the epistemic uncertainty to be smaller for SPES than SISS.

To verify this, we examined bivariate density plots of the aleatoric and epistemic uncertainties among the top 1% data points with respect to the largest aleatoric uncertainty. All the results are evaluated in five-folds cross-validation for a fair comparison between the two ISLES datasets. Fig. 4 shows bivariate density plots of SISS on the left and SPES on the right. Quantified uncertainty tends to be smaller for SISS than SPES. The SISS dataset also exhibits smaller epistemic uncertainty than SPES as anticipated. Similar trends were observed when using the top 5% data.

Table 2 compares the average of the aleatoric and epistemic uncertainties of SPES and SISS. The aleatoric uncertainty of SPES is about five times as large as that of SISS. The epistemic uncertainty of SISS is smaller than that of SPES as anticipated since the epistemic uncertainty decreases with the effective sample size.

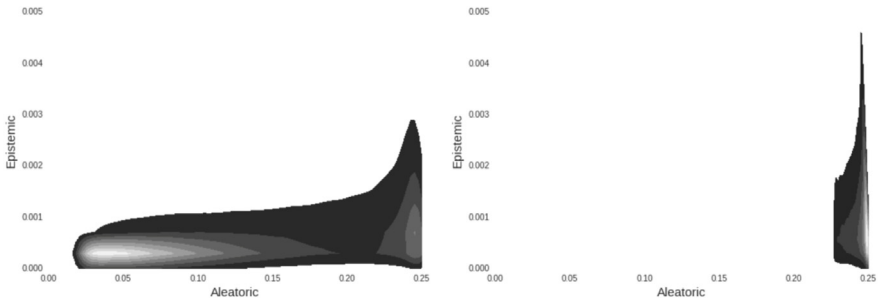

Fig. 4. Bivariate density plots for aleatoric and epistemic uncertainties among the top 1% data points in terms of the aleatoric uncertainty. The number of realized vectors T equals 5. Left, SISS, and right, SPES datasets.

Table 2
Comparison of the averages of aleatoric and epistemic uncertainties of the SISS and SPES datasets. The number of realized vectors T equals 5.

	SISS	SPES
Aleatoric	7.6×10^{-4}	3.6×10^{-3}
Epistemic	2.2×10^{-5}	1.4×10^{-4}

Table 3
Comparison of the conditional expectation of epistemic uncertainty given an aleatoric intervals for the SISS and SPES datasets. The number of realized vectors T equals 5. Epi, Epistemic; Ale, Aleatoric.

	SISS	SPES
$E(\text{Epi} \mid 0.05 < \text{Ale} < 0.1)$	4.4×10^{-4}	4.6×10^{-4}
$E(\text{Epi} \mid 0.1 < \text{Ale} < 0.15)$	8.9×10^{-4}	9.4×10^{-4}
$E(\text{Epi} \mid 0.15 < \text{Ale} < 0.2)$	13.8×10^{-4}	14.5×10^{-4}
$E(\text{Epi} \mid 0.2 < \text{Ale} < 0.25)$	17.6×10^{-4}	19.4×10^{-4}

Table 4
Comparison of the averages of aleatoric and epistemic uncertainties of the DRIVE-test set. The number of realized vectors T equals 5.

	DRIVE-2000	DRIVE-200
Aleatoric	2.8×10^{-2}	2.8×10^{-2}
Epistemic	2.3×10^{-3}	3.7×10^{-3}

Since the variability of a binary random variable depends on the mean, the difference in the aleatoric uncertainty can confound the comparison of the epistemic uncertainty. To circumvent this problem, we compared the estimated conditional expectations of the epistemic uncertainty given ranges of the aleatoric uncertainty. The results are given in Table 3. Although the differences are small, there is a coherent trend that the epistemic uncertainty is smaller for the SISS dataset than for the SPES dataset by 5%–10%.

5.3.2. DRIVE dataset

To evaluate the effects of sample size on epistemic uncertainties, we constructed two training datasets by varying the numbers of extracted patches, either 2000 or 200, from the training set. We call these datasets DRIVE-2000 and DRIVE-200, respectively. Effective sample size is larger for DRIVE-2000 than for DRIVE-200. For a test dataset, we applied the same preprocessing steps, extracting 1000 patches from the original test set. We call this preprocessed dataset DRIVE-test.

We conducted the parallel experiment using DRIVE-2000 and DRIVE-200. We first trained separate Bayesian neural networks for DRIVE-2000 and for DRIVE-200. Using these trained Bayesian neural networks, we evaluated uncertainties of DRIVE-test set. We show the bivariate density plots for the aleatoric and epistemic uncertainties in Fig. 5. The results exhibit a similar trend as in the ISLES in that the epistemic uncertainties are smaller for DRIVE-2000 than for DRIVE-200.

Table 4 compares the averages of aleatoric and epistemic uncertainties of the DRIVE-test set. While the aleatoric uncertainties are comparable, the average of epistemic uncertainties of the DRIVE-200 dataset is about 1.8 times as large as that of the DRIVE-2000.

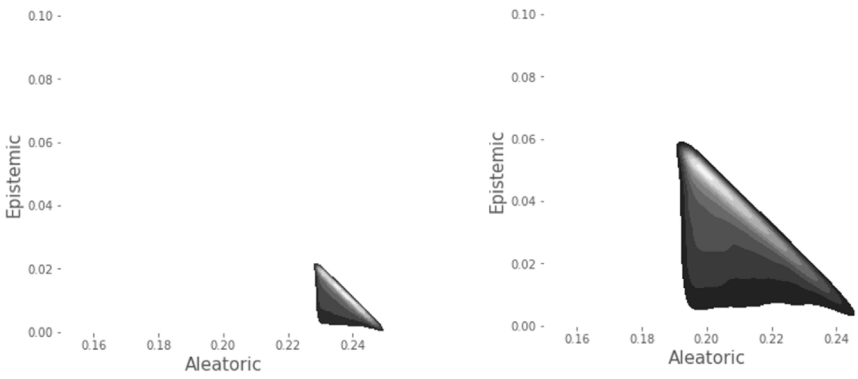

Fig. 5. Bivariate density plots for aleatoric and epistemic uncertainties among the top 1% data points in terms of the aleatoric uncertainty. The number of realized vectors T equals 5. Left, DRIVE-2000, and right, DRIVE-200 datasets.

Table 5
Comparison of the conditional expectation of epistemic uncertainty given an aleatoric intervals for the DRIVE-test dataset. The number of realized vectors T equals 5. Epi, Epistemic; Ale, Aleatoric.

	DRIVE-2000	DRIVE-200
$E(\text{Epi} \mid 0.05 < \text{Ale} < 0.1)$	6.7×10^{-3}	1.1×10^{-2}
$E(\text{Epi} \mid 0.1 < \text{Ale} < 0.15)$	1.4×10^{-2}	3.8×10^{-2}
$E(\text{Epi} \mid 0.15 < \text{Ale} < 0.2)$	2.0×10^{-2}	4.4×10^{-2}
$E(\text{Epi} \mid 0.2 < \text{Ale} < 0.25)$	1.6×10^{-2}	2.6×10^{-2}

Table 5 shows the comparison of the conditional expectation of epistemic uncertainties using the DRIVE datasets to adjust for the confounding effect of increasing uncertainty with the mean. The epistemic uncertainties conditioning on the aleatoric uncertainties of DRIVE-2000 are about half those of the DRIVE-200. In our experiment, the effect of sample size on the number of patch sizes is more outstanding than that through the quality of resolution.

5.4. Computational time

Quantifying uncertainty based on Bayesian inference can be time-consuming due to multiple samplings and feed-forward computations. For example, evaluating Eqs. (7) and (8) requires multiple samples of $\{\hat{\omega}_t\}_{t=1}^T$ from the variational distribution $q_{\hat{\theta}}(\omega)$ and the feed-forward computation of $\{p(y^* \mid x^*, \hat{\omega}_t)\}_{t=1}^T$. In addition, since the total amount of calculations is proportional to T, the computational time will increase linearly as T increases. In this subsection, we evaluated the computational time and uncertainty by varying the number of realized samples T. We used the same implementation setting as described in Section 5.3.2, except T. The training time for θ took 72 and 207 s when the dataset is DRIVE-200 and DRIVE-2000, respectively.

Fig. 6 shows the average of elapsed time per subject, aleatoric uncertainty, and epistemic uncertainty when $T \in \{3, 5, 10, 15, 20, 25, 30\}$. As expected, the computational times linearly increased as T increased. Elapsed time per each subject is around 0.30 s when $T = 30$, which suggests that the sampling procedure is feasible with a reasonable number of T.

6. Conclusion and discussion

In this paper, we presented a natural way of quantifying the uncertainty of predicted outcomes for classification via Bayesian neural networks by decomposing a predictive uncertainty measure into two types. The proposed method has advantages over the existing one in that the inherent variability is expressed in terms of the underlying distribution of the outcome and that it is numerically stable. Application of our method to the DRIVE and the two ISLES datasets provided additional insights into the corresponding medical imaging analysis not possible with point estimation alone.

We acknowledge that variational predictive uncertainty may not be a good approximation of the predictive uncertainty. However, the epistemic uncertainty can be useful in comparing the variability due to the effective sample size. Our comparison studies between the two datasets with different effective sample sizes support the utility of the epistemic uncertainty.

The proposed method of uncertainty quantification can be extended to different functional forms of variational densities (Miller et al., 2017; Louizos and Welling, 2017). Also it can be extended to variational inference settings other

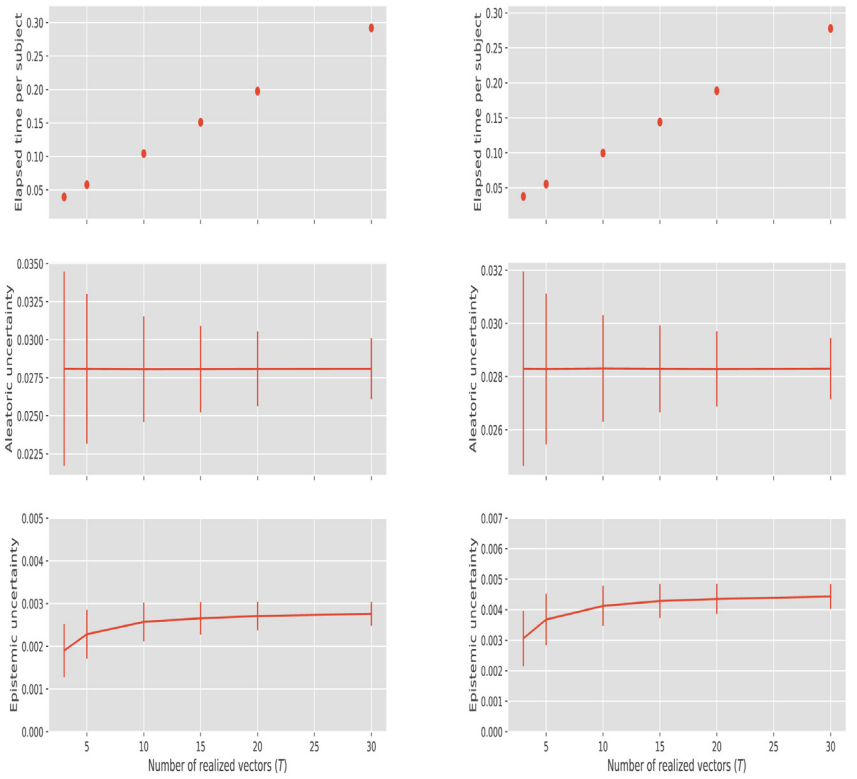

Fig. 6. Changes in (top) computational time, (middle) aleatoric uncertainty and (bottom) epistemic uncertainty with respect to the number of realized vectors T. Error bars indicate the standard errors based on the repetition T. Left, DRIVE-2000, and right, DRIVE-200 datasets.

than Kullback–Leibler divergence (Li and Turner, 2016; Dieng et al., 2017) or to general Bayesian learning other than variational inference, such as MCMC (Welling and Teh, 2011; Chen et al., 2014).

Acknowledgments

The authors are grateful to Walter de Back for helpful comments. The authors were supported by the National Research Foundation of Korea under grant NRF-2017R1A2B4008956.

Appendix A. Detailed derivation of the decomposition

Recall that the decomposition is as follows,

$$\begin{aligned}\mathrm{Var}_{p(y^*|x^*,\mathcal{D})}(y^*) &:= \mathbb{E}_{p(y^*|x^*,\mathcal{D})}\{y^{*\otimes 2}\} - \mathbb{E}_{p(y^*|x^*,\mathcal{D})}(y^*)^{\otimes 2} \\ &= \underbrace{\int_\Omega \mathrm{Var}_{p(y^*|x^*,\omega)}(y^*)p(\omega \mid \mathcal{D})d\omega}_{=:\text{ aleatoric uncertainty}} \\ &+ \underbrace{\int_\Omega \left\{\mathbb{E}_{p(y^*|x^*,\omega)}(y^*) - \mathbb{E}_{p(y^*|x^*,\mathcal{D})}(y^*)\right\}^{\otimes 2} p(\omega \mid \mathcal{D})d\omega}_{=:\text{ epistemic uncertainty}}.\end{aligned}$$

The first equation follows from the definition of variance and it is enough to show the second equation. From the definition of $\mathbb{E}_{p(y^*|x^*,\mathcal{D})}\{g(y^*)\}$ and by Fubini's Theorem, we have

$$\begin{aligned}\mathbb{E}_{p(y^*|x^*,\mathcal{D})}(y^*) &:= \int y^* p(y^* \mid x^*, \mathcal{D})dy^* \\ &= \int y^* \int_\Omega p(y^* \mid x^*, \omega)p(\omega \mid \mathcal{D})d\omega dy^*\end{aligned}$$

$$= \int_\Omega \int y^* p(y^* \mid x^*, \omega) dy^* p(\omega \mid \mathcal{D}) d\omega$$
$$= \int_\Omega \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*) p(\omega \mid \mathcal{D}) d\omega,$$

and similarly,

$$\begin{aligned}
\mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^{*\otimes 2}) &:= \int y^{*\otimes 2} p(y^* \mid x^*, \mathcal{D}) dy^* \\
&= \int y^{*\otimes 2} \int_\Omega p(y^* \mid x^*, \omega) p(\omega \mid \mathcal{D}) d\omega dy^* \\
&= \int_\Omega \int y^{*\otimes 2} p(y^* \mid x^*, \omega) dy^* p(\omega \mid \mathcal{D}) d\omega \\
&= \int_\Omega \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^{*\otimes 2}) p(\omega \mid \mathcal{D}) d\omega \\
&= \int_\Omega \{\mathrm{Var}_{p(y^* \mid x^*, \omega)}(y^*) + \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*)^{\otimes 2}\} p(\omega \mid \mathcal{D}) d\omega.
\end{aligned}$$

Thus,

$$\begin{aligned}
&\mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}\{y^{*\otimes 2}\} - \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^*)^{\otimes 2} \\
&= \int_\Omega \{\mathrm{Var}_{p(y^* \mid x^*, \omega)}(y^*) + \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*)^{\otimes 2}\} p(\omega \mid \mathcal{D}) d\omega - \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^*)^{\otimes 2} \\
&= \int_\Omega \mathrm{Var}_{p(y^* \mid x^*, \omega)}(y^*) p(\omega \mid \mathcal{D}) d\omega \\
&\quad + \int_\Omega \{\mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*)^{\otimes 2} - \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^*)^{\otimes 2}\} p(\omega \mid \mathcal{D}) d\omega \\
&= \int_\Omega \mathrm{Var}_{p(y^* \mid x^*, \omega)}(y^*) p(\omega \mid \mathcal{D}) d\omega \\
&\quad + \int_\Omega \left\{\mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*) - \mathbb{E}_{p(y^* \mid x^*, \mathcal{D})}(y^*)\right\}^{\otimes 2} p(\omega \mid \mathcal{D}) d\omega.
\end{aligned}$$

For categorical variable y^*, $\mathrm{Var}_{p(y^* \mid x^*, \omega)}(y^*) = \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^{*\otimes 2}) - \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*)^{\otimes 2}$ and it is $\mathbb{E}_{p(y^* \mid x^*, \omega)}\{\mathrm{diag}(y^*)\} - \mathbb{E}_{p(y^* \mid x^*, \omega)}(y^*)^{\otimes 2}$ because y^* is one-hot encoded.

Appendix B. Implementation details

In this section, we introduce implementation details for Section 5. For both datasets, the variational inference with the dropout variational distribution (Gal, 2016) was used to train Bayesian neural networks. For the sake of completeness, we specify the dropout variational distribution and the used variational inference algorithm.

To this end, let $j \in \{1, \ldots, L+1\}$ and $k \in \{1, \ldots, d_{j-1}\}$. Let $\omega_{[j],k} \in \mathbb{R}^{d_j}$ be the parameter representing the random weights between the kth neuron in the $(j-1)$-th layer and the neurons in the jth layer. Denote $\omega_{[j]} = (\omega_{[j],1} \mid \ldots \mid \omega_{[j],d_{j-1}}) \in \mathbb{R}^{d_j \times d_{j-1}}$ and $\omega = (\mathrm{vec}(\omega_{[1]}), \ldots, \mathrm{vec}(\omega_{[L+1]}))$. Similarly, let $\theta_{[j],k} \in \mathbb{R}^{d_j}$ be the variational parameter representing the actual weights between the kth neuron in the $(j-1)$-th layer and the neurons in the jth layer. Set $\theta_{[j]} = (\theta_{[j],1} \mid \ldots \mid \theta_{[j],d_{j-1}}) \in \mathbb{R}^{d_j \times d_{j-1}}$ and $\theta = (\mathrm{vec}(\theta_{[1]}), \ldots, \mathrm{vec}(\theta_{[L+1]}))$. Note that the dimension of the parameter $\omega_{[j],k}$ and that of the variational parameter $\theta_{[j],k}$ are equal. With the mean field assumption, the dropout variational distribution is defined by $q_\theta(\omega) := \prod_{j=1}^{L+1} \prod_{k=1}^{d_{j-1}} q_{\theta_{[j],k}}(\omega_{[j],k})$, where $q_{\theta_{[j],k}}(\omega_{[j],k}) = (1-p_j)^{\mathbb{1}(\omega_{[j],k} = \theta_{[j],k})} (p_j)^{\mathbb{1}(\omega_{[j],k} = \mathbf{0}_{d_j})}$ for the zero vector $\mathbf{0}_{d_j}$ in d_j dimensions. Note that $q_{\theta_{[j],k}}(\omega_{[j],k})$ is a probability mass function of a discrete random variable $\omega_{[j],k}$ defined by

$$\omega_{[j],k} = \begin{cases} \theta_{[j],k} & \text{with probability } 1-p_j, \\ \mathbf{0}_{d_j} & \text{with probability } p_j. \end{cases}$$

In our experiments, the drop rate p_j's are 0.5 and 0.25 for ischemic stroke lesion segmentation (ISLES) dataset and digital retinal images for vessel extraction (DRIVE) dataset, respectively. Note that this dropout variational distribution can be implemented by building a neural network with dropout layers (Gal, 2016).

Algorithm 1 Proposed uncertainty quantification with dropout variational inference

procedure LEARNING A BAYESIAN NEURAL NETWORK (GAL, 2016)
  Initialize all the weights θ in the neural network by He et al. (2015).
  while θ does not converge **do**
    # Randomly sample a mini-batch $\{(x_{(i)}, y_{(i)})\}_{i=1}^{B}$ from a training set $\mathcal{D}$.
    # Randomly sample a set of realized vector $\{\omega_t\}_{t=1}^{T_{\text{tr}}}$ from $q_\theta(\omega)$.
    # Under the KL condition,[3] we compute the gradient estimate of the negative ELBO.

$$\nabla\mathcal{L} = \frac{\partial}{\partial\theta}\left(-\frac{1}{T_{\text{tr}}B}\sum_{t=1}^{T_{\text{tr}}}\sum_{i=1}^{B}\log p(y_{(i)} \mid x_{(i)}, \omega_t) + \sum_{j=1}^{L+1}\lambda_j \left\|\theta_{[j]}\right\|_2^2\right).$$

    # Update the variational parameter with a learning rate ν.
    $\theta \leftarrow \theta - \nu\nabla\mathcal{L}$.
procedure QUANTIFYING UNCERTAINTIES
  # Let $\hat{\theta}$ be the optimized variational parameter.
  # Randomly sample a set of realized vector $\{\hat{\omega}_t\}_{t=1}^{T}$ from $q_{\hat{\theta}}(\omega)$.
  # Compute aleatoric and epistemic uncertainties for a new sample x^* by

$$\frac{1}{T}\sum_{t=1}^{T}\left[\text{diag}\{p(y^* \mid x^*, \hat{\omega}_t)\} - p(y^* \mid x^*, \hat{\omega}_t)^{\otimes 2}\right],$$

  and

$$\frac{1}{T}\sum_{t=1}^{T}\{p(y^* \mid x^*, \hat{\omega}_t) - \hat{p}_{\hat{\theta}}(y^* \mid x^*)\}^{\otimes 2},$$

  respectively, and $p(y^* \mid x^*, \hat{\omega}_t) = f^{\hat{\omega}_t}(x^*)$.

We demonstrate the used training phase and the proposed uncertainty quantification procedure in Algorithm 1. We set $T_{\text{tr}} = 1$, the number of realized samples $T = 5$, and $\lambda_j = 0$ for all $j \in \{1, \ldots, L+1\}$. A learning rate ν for each dataset is specified in the following subsections. All the implementation codes and Python notebooks are publicly available at http://github.com/ykwon0407/UQ_BNN.

B.1. Ischemic stroke lesion segmentation (ISLES)

Our learning process and implementation details mostly follow the method as proposed in Section 2.1 of Choi et al. (2016). We trained neural networks through patch learning and omitted the fine-tuning step which was turned out to be not effective in terms of the Dice coefficient.

Fig. B.7 shows our network architecture, the 3D multiscale residual U-Net (Choi et al., 2016) without residual blocks. The total number of parameters were around 293k. For the method by Kendall and Gal (2017), we removed the softmax activation layer and changed the number of channels in the last convolutional layer from 1 to 2. Then, we randomly drew a sample from normal distribution using the network outputs, and used it to compute numerically-stable stochastic loss, which is a numerical stable version of a negative log likelihood function (Kendall and Gal, 2017).

We used the following hyper-parameters. All the MRI sequences were resized to the same dimension: (width $\times$ height $\times$ depth) = ($100 \times 100 \times 72$). For pre-processing step, we extracted 600 patches with the size of (width $\times$ height $\times$ depth) = ($16 \times 16 \times 16$). All the weights were initialized as He et al. (2015). The Adam optimizer (Kingma and Ba, 2014) with the learning rate of 0.0002 and an early stopping rule with 10-epoch patience were applied; the batch size was 16. The negative log likelihood function was used as a loss function. Keras (Chollet, 2015) was used for implementation. The main Python scripts are available at https://github.com/ykwon0407/UQ_BNN/tree/master/ischemic.

B.2. Digital Retinal Images for Vessel Extraction (DRIVE)

We used a small variant of the original U-Net (Ronneberger et al., 2015). The specification of a network architecture is visually presented in Fig. B.8. Total number of parameters were around 1.1M. Most hyperparameter settings were the same as we used in the ISLES analysis. We randomly split a dataset into 20% for a validation set and 80% for a training set.

The optimizer was the Adam (Kingma and Ba, 2014) with the learning rate of 0.001 and the batch size was 32. Python scripts are available at https://github.com/ykwon0407/UQ_BNN/tree/master/retina.

[3] Derivative of KL can be approximated by L_2 norm of the weights (Gal, 2016, Section 3.2.3).

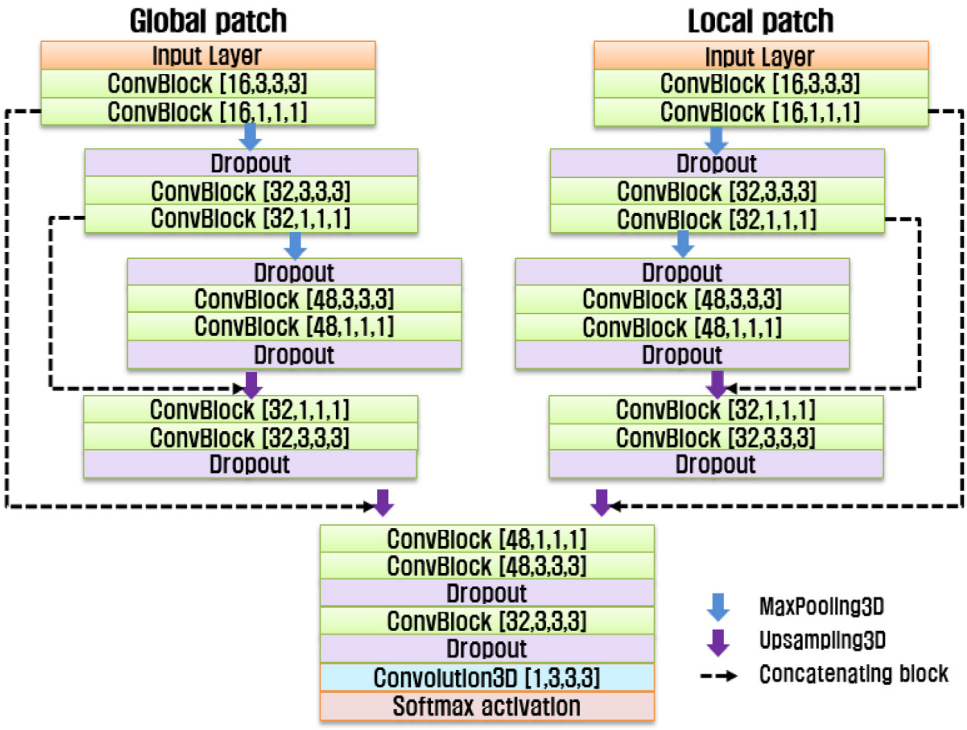

Fig. B.7. Visual description of the main network architecture for the ISLES datasets analysis. The 'ConvBlock' is made up of three dimensional convolutional layer, ELU activation (Clevert et al., 2015), and batch normalization layer (Ioffe and Szegedy, 2015). The numbers in parenthesis denote the number of channels and kernel size in three dimension.

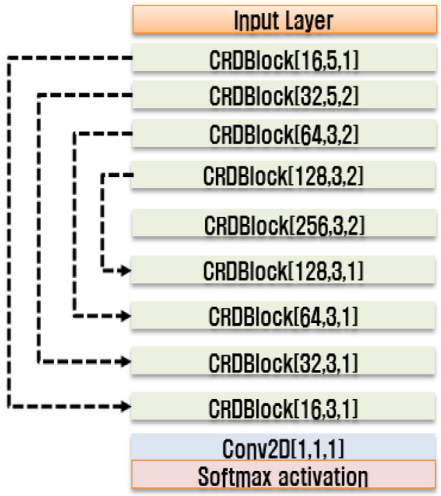

Fig. B.8. Visual description of the network architecture for the DRIVE datasets analysis. The 'CRDBlock' is made up of two 2-dimensional convolutional layers, ReLU activation, a batch normalization layer, and dropout layer with 0.25 drop rate. The numbers in parenthesis denote the number of channels, kernel size, and stride of the second convolutional layer. The first convolutional layer has the same number of channels but a kernel size and a stride are fixed as 1. Dotted line indicates concatenation as the original U-Net (Ronneberger et al., 2015).

Appendix C. Additional ISLES visualization results

In Fig. C.9, we present additional visualization results comparing proposed method with the method by Kendall and Gal (2017). Rows 'K' and 'P' indicate the uncertainty quantification method by Kendall and Gal (2017) and the proposed method, respectively. In multiple examples, the proposed method demonstrates informative uncertainty maps.

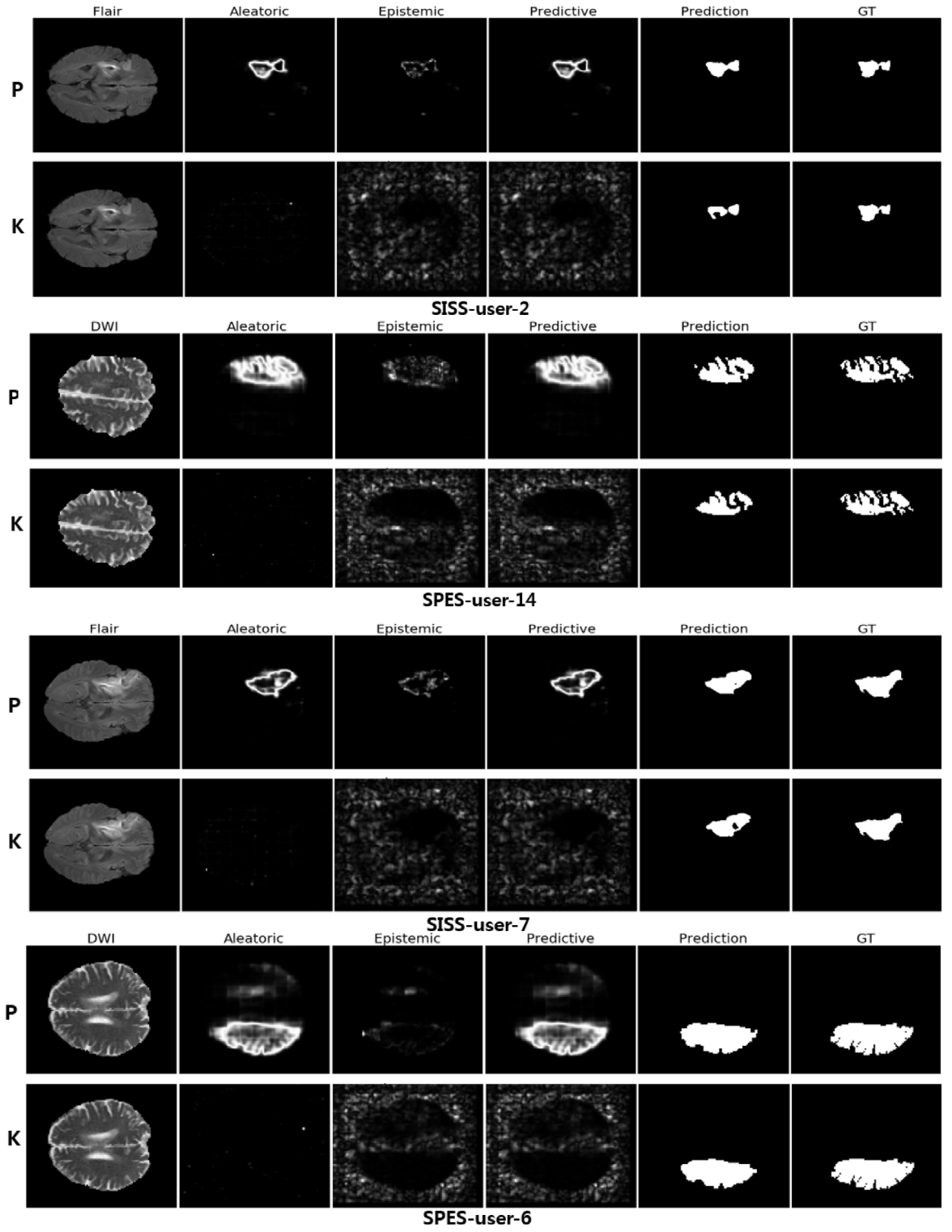

Fig. C.9. Additional visual comparison results using the SISS and the SPES datasets.

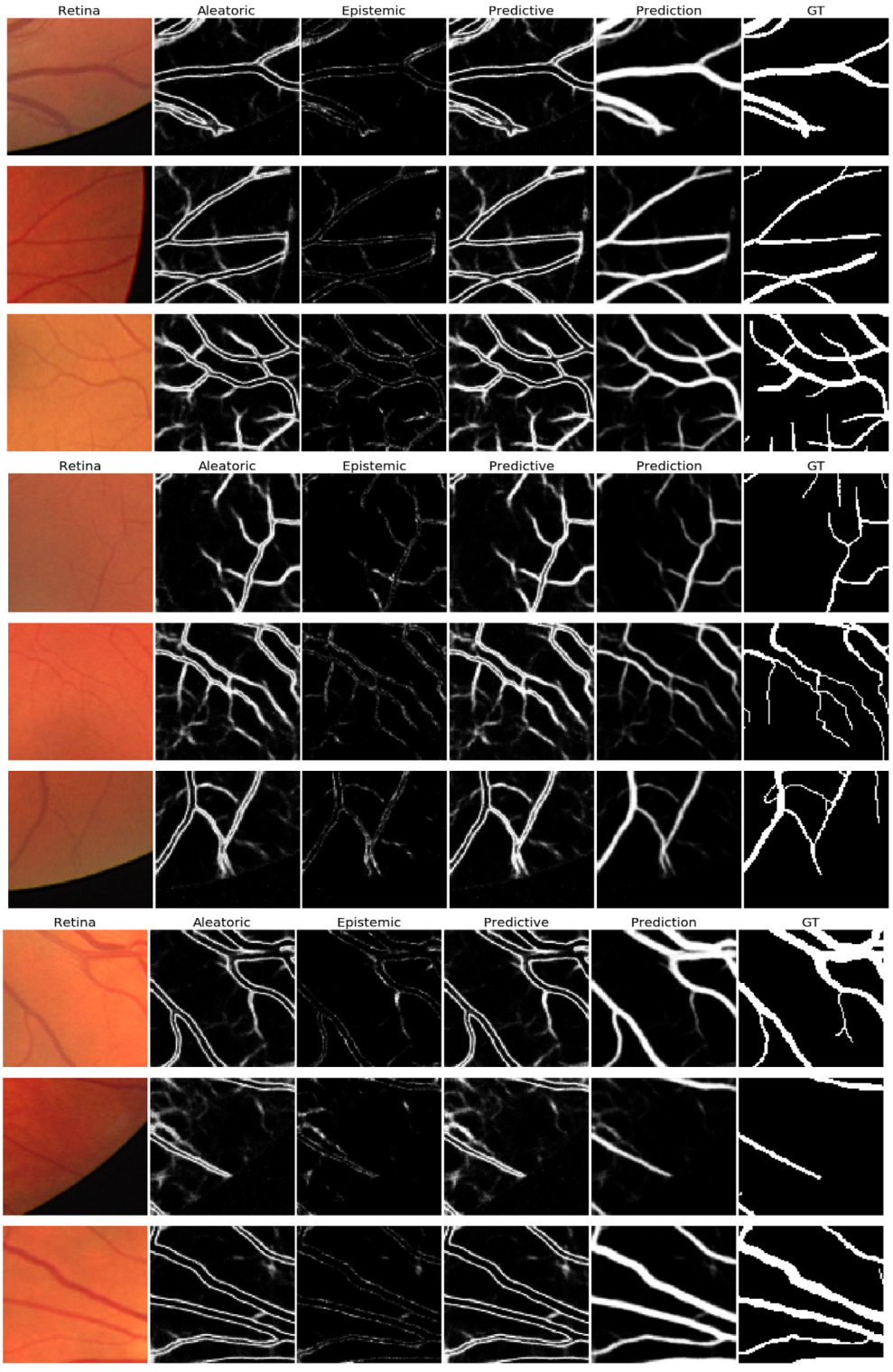

Fig. D.10. Uncertainty quantification results using the DRIVE-test dataset. Bayesian neural networks are trained with the DRIVE-2000 dataset as described in Appendix B.

Appendix D. Additional DRIVE visualization results

In Fig. D.10, we present additional visualization results demonstrating merits of proposed method using DRIVE dataset.

References

Ayhan, M.S., Berens, P., 2018. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In: International Conference on Medical Imaging with Deep Learning.

Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D., 2015. Weight uncertainty in neural networks. In: International Conference on Machine Learning. pp. 1613–1622.

Chen, T., Fox, E., Guestrin, C., 2014. Stochastic gradient hamiltonian monte carlo. In: International Conference on Machine Learning. pp. 1683–1691.

Choi, Y., Kwon, Y., Lee, H., Kim, B.J., Paik, M.C., Won, J.-H., 2016. Ensemble of deep convolutional neural networks for prognosis of ischemic stroke. In: International Workshop on Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Springer, pp. 231–243.

Chollet, F., 2015. keras, https://github.com/fchollet/keras.

Clevert, D.-A., Unterthiner, T., Hochreiter, S., 2015. Fast and accurate deep network learning by exponential linear units (elus), arXiv preprint arXiv:1511.07289.

Der Kiureghian, A., Ditlevsen, O., 2009. Aleatory or epistemic? does it matter?. Struct. Saf. 31 (2), 105–112.

Dieng, A.B., Tran, D., Ranganath, R., Paisley, J., Blei, D., 2017. Variational inference via χ upper bound minimization. In: Advances in Neural Information Processing Systems. pp. 2729–2738.

Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S., 2017. Dermatologist-level classification of skin cancer with deep neural networks. Nature 542 (7639), 115.

Fleiss, J., Levin, B., Paik, M., 2003. Statistical Methods for Rates and Proportions. In: Wiley Series in Probability and Statistics, Wiley.

Gal, Y., 2016. Uncertainty in Deep Learning (Ph.D. thesis). University of Cambridge.

Graves, A., 2011. Practical variational inference for neural networks. In: Advances in Neural Information Processing Systems. pp. 2348–2356.

Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q., 2017. On Calibration of modern neural networks. In: International Conference on Machine Learning, pp. 1321–1330.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1026–1034.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. pp. 448–456.

Jungo, A., Meier, R., Ermis, E., Herrmann, E., Reyes, M., 2018. Uncertainty-driven sanity check: Application to postoperative brain tumor Cavity segmentation. In: International Conference on Medical Imaging with Deep Learning.

2015. Kaggle, Kaggle diabetic retinopathy detection competition. https://www.kaggle.com/c/diabetic-retinopathy-detection [Online; accessed 31.07.18].

Kamnitsas, K., Ledig, C., Newcombe, V.F., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B., 2017. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. Med. Image Anal. 36, 61–78.

Kendall, A., Gal, Y., 2017. What uncertainties do we need in bayesian deep learning for computer vision?. In: Advances in Neural Information Processing Systems. pp. 5580–5590.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.

Kingma, D.P., Welling, M., 2013. Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114.

Lee, H.K., 2000. Consistency of posterior distributions for neural networks. Neural Netw. 13 (6), 629–642.

Leibig, C., Allken, V., Ayhan, M.S., Berens, P., Wahl, S., 2017. Leveraging uncertainty information from deep neural networks for disease detection. Sci. Rep. 7 (1), 17816.

Li, Y., Turner, R.E., 2016. Rényi Divergence variational inference. In: Advances in Neural Information Processing Systems. pp. 1073–1081.

Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B., Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Med. Image Anal. 42, 60–88.

Louizos, C., Welling, M., 2017. Multiplicative normalizing flows for variational Bayesian neural networks. In: International Conference on Machine Learning. pp. 2218–2227.

MacKay, D.J., 1992. A practical Bayesian framework for backpropagation networks. Neural Comput. 4 (3), 448–472.

Maier, O., Menze, B.H., von der Gablentz, J., Häni, L., Heinrich, M.P., Liebrand, M., Winzeck, S., Basit, A., Bentley, P., Chen, L., et al., 2017. Isles 2015-a public evaluation benchmark for ischemic stroke lesion segmentation from multispectral MRI. Med. Image Anal. 35, 250–269.

Miller, A.C., Foti, N.J., Adams, R.P., 2017. Variational boosting: Iteratively refining posterior approximations. In: International Conference on Machine Learning. pp. 2420–2429.

Neal, R.M., 1993. Bayesian Learning via stochastic dynamics. In: Advances in Neural Information Processing Systems. pp. 475–482.

Niemeijer, M., Staal, J., van Ginneken, B., Loog, M., Abramoff, M., 2004. Comparative study of retinal vessel segmentation methods on a new publicly available database. In: Fitzpatrick, J.M., Sonka, M. (Eds.), SPIE Medical Imaging, Vol. 5370. SPIE, pp. 648–656.

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, Ł., Hinton, G., 2017. Regularizing neural networks by penalizing confident output distributions, arXiv preprint arXiv:1701.06548.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 234–241.

Siva Sundhara Raja, D., Vasuki, S., 2015. Automatic detection of blood vessels in retinal images for diabetic retinopathy diagnosis. Comput. Math. Methods Med. 2015.

Staal, J., Abràmoff, M.D., Niemeijer, M., Viergever, M.A., Van Ginneken, B., 2004. Ridge-based vessel segmentation in color images of the retina. IEEE Trans. Med. Imaging 23 (4), 501–509.

Tanno, R., Worrall, D.E., Ghosh, A., Kaden, E., Sotiropoulos, S.N., Criminisi, A., Alexander, D.C., 2017. Bayesian image quality transfer with cnns: Exploring uncertainty in dmri super-resolution. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 611–619.

Teye, M., Azizpour, H., Smith, K., 2018. Bayesian Uncertainty Estimation for Batch Normalized Deep Networks, arXiv preprint arXiv:1802.06455.

Wang, Y., Blei, D.M., 2018. Frequentist consistency of variational Bayes. J. Amer. Statist. Assoc. (just-accepted), 1–85.

Welling, M., Teh, Y.W., 2011. Bayesian Learning via stochastic gradient langevin dynamics. In: International Conference on Machine Learning, pp. 681–688.

Winzeck, S., Hakim, A., McKinley, R., Pinto, J.A., Alves, V., Silva, C., Pisov, M., Krivov, E., Belyaev, M., Monteiro, M., et al., 2018. ISLES 2016 and 2017-benchmarking ischemic stroke lesion outcome prediction based on multispectral MRI. Front. Neurol. 9.