Dropout as Bayesian Approximation:
Representing Model Uncertainty in Deep Learning

Introduction

- Tools (like deep learning)for regression and classification do not capture model uncertainty (Root cause of the discussion)

- On contrary, Bayesian Model, is a mathematically grounded framework but with high computational cost.
  * A Bayesian model is a statistical model that uses probability to represent uncertainty in both the model's output and input. It's made up of a model for the data and a prior distribution for the model's parameters.

- This paper talks about casting dropout training in deep neural networks, as approximate Bayesian inference in deep Gaussian process.

- It mitigates the problem of representing uncertainty in deep learning without sacrificing computational complexity or test accuracy.

Paper’s view

- Neural network with arbitrary depth and non-linearities with dropout applied before every weight layer, is mathematically equivalent to an approximation to the probabilistic deep Gaussian process.

  *Deep Gaussian Processes are a sophisticated machine learning approach that combines the hierarchical structure of deep learning with the probabilistic modeling capabilities of Gaussian processes. offering a powerful tool for tasks requiring complex data modeling and uncertainty quantification

- The paper compare uncertainty obtained from different task.

- Shows that model uncertainty is indispensable for classification task.

When training a neural network

Important aspects

* Those marked in Red are where uncertainty of model is dealt

- Data Preprocessing
- Data Augmentation
- Network Architecture
- Regularization Techniques (Dropout)
- Loss Function and Optimization
- Learning Rate
- Batch size
- Evaluation Metrics
- Training Duration
- Hardware Considerations
- Monitoring Training
- Hyperparameter Tuning
- Reproducibility
- Validation strategies
- Error Analysis
- Model Deployment

What is Uncertainty

- Lack of certainty in the prediction by statistical model or algorithm, due to inherent limitation of model or training data.

- Some of the sources and type of Uncertainty

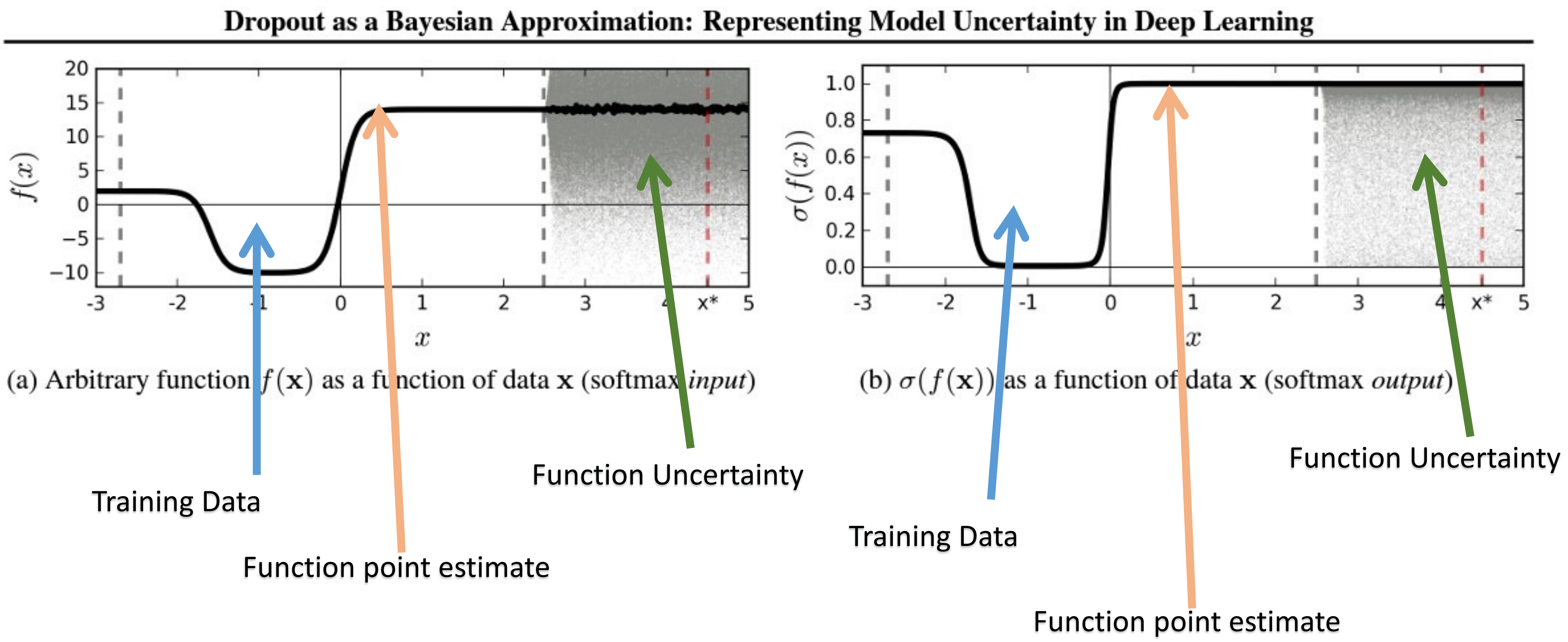
  Aleatoric Uncertainty
  - Data Uncertainty

  Epistemic Uncertainty
  - Parameter Uncertainty
  - Model Structure Uncertainty
  - Conceptual Uncertainty
  - Extrapolation Uncertainty

  - Aleatoric Uncertainty & Epistemic Uncertainty
    - Prediction Uncertainty

- Estimate and report the uncertainty of model Prediction provide information about reliability of the prediction.

Sample of Uncertainty



Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

(a) Arbitrary function $f(\mathbf{x})$ as a function of data $\mathbf{x}$ (softmax *input*)

(b) $\sigma(f(\mathbf{x}))$ as a function of data $\mathbf{x}$ (softmax *output*)

Regularization

Regularization refers to techniques used to prevent overfitting . It helps to improve the generalization of the model to new or unseen data.

Commonly used regularization techniques in neural network:

- Weight Regularization (L1 and L2 regularization) : Penalty to loss function based on the magnitude of weights.
- Dropout : Randomly selected neurons are ignored during the training.
- Early Stopping : Monitor model's performance on validation set during training and stop training once performance begin to degrade.
- Data Augmentation: Artificially increasing the size of training dataset, by creating modified version of data. (e.g. rotations, translations, zooming on images. Model generalize better when provided with larger or more diverse training set.
- Noise Addition : Adding noise to input or hidden layers during training prevents models from learning spurious (misleading) patterns in training data.
- Batch Normalization : Besides normalizing output of each layer's activations, batch normalization can have a regularizing effect as well.

  *Normalizing : adjusting input data or activation of neurons to keep them close to the particular distribution, often a standard distribution.
- Ensemble Methods : Techniques like bagging and boosting, where multiple models are trained, and their predictions are averaged. It improves generalization and performance.

Dropout computation

$$\mathcal{L}_{\text{dropout}} := \frac{1}{N}\sum_{i=1}^{N} E(\mathbf{y}_i, \widehat{\mathbf{y}}_i) + \lambda \sum_{i=1}^{L} \left(||\mathbf{W}_i||_2^2 + ||\mathbf{b}_i||_2^2 \right).$$

Average loss over all N samples

- N : Sample Count
- E() : loss term for the i-th sample
- λ : Hyperparameter
- L : Layer of the neural network
- $||wi||_2^2$: Sum of squares of the weights in the i-th layer
- $||bi||_2^2$: Sum of the squares of the biases in the i-th layer

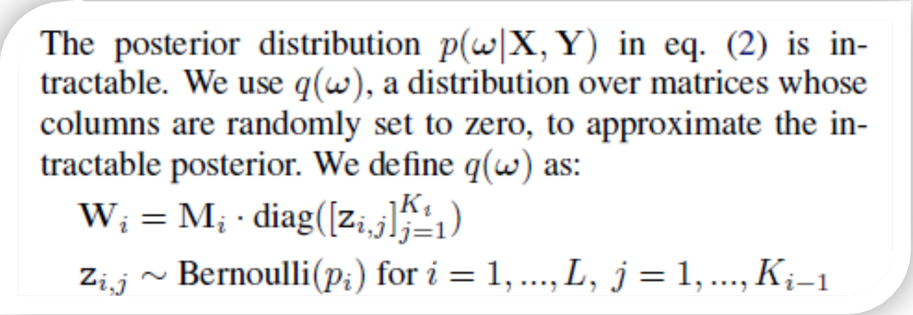
Predictive Probability of deep Gaussian process model

$$p(\mathbf{y}|\mathbf{x},\mathbf{X},\mathbf{Y}) = \int p(\mathbf{y}|\mathbf{x},\omega)p(\omega|\mathbf{X},\mathbf{Y})\mathrm{d}\omega \qquad (2)$$

$$p(\mathbf{y}|\mathbf{x},\omega) = \mathcal{N}\left(\mathbf{y}; \hat{\mathbf{y}}(\mathbf{x},\omega), \tau^{-1}\mathbf{I}_D\right)$$

$$\hat{\mathbf{y}}\left(\mathbf{x},\omega=\{\mathbf{W}_1,...,\mathbf{W}_L\}\right) = \sqrt{\frac{1}{K_L}}\mathbf{W}_L\sigma\left(...\sqrt{\frac{1}{K_1}}\mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{x}+\mathbf{m}_1)...\right)$$

- P(y|x,X, Y) : Predictive distribution of a new output y, given a new input x. Observed data X (input) and Y(outputs)
- P(y|x,w) : Likelihood of y given the x and a specific set of weights w.
- λ : Hyperparameter
- L : Layer of the neural network
- $||wi||_2^2$: Sum of squares of the weights in the i-th layer
- $||bi||_2^2$: Sum of the squares of the biases in the i-th layer

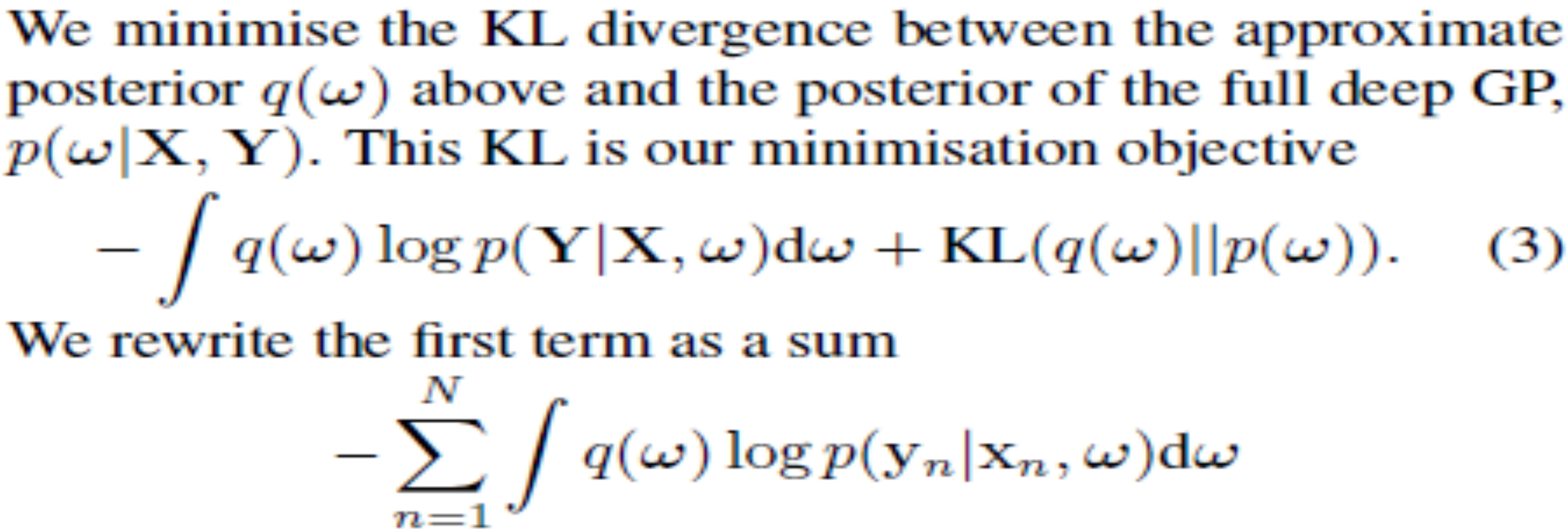


The posterior distribution $p(\omega|\mathbf{X},\mathbf{Y})$ in eq. (2) is intractable. We use $q(\omega)$, a distribution over matrices whose columns are randomly set to zero, to approximate the intractable posterior. We define $q(\omega)$ as:

$$\mathbf{W}_i = \mathbf{M}_i \cdot \mathrm{diag}([\mathbf{z}_{i,j}]_{j=1}^{K_i})$$

$$\mathbf{z}_{i,j} \sim \mathrm{Bernoulli}(p_i) \text{ for } i=1,...,L,\ j=1,...,K_{i-1}$$

Minimization Objective



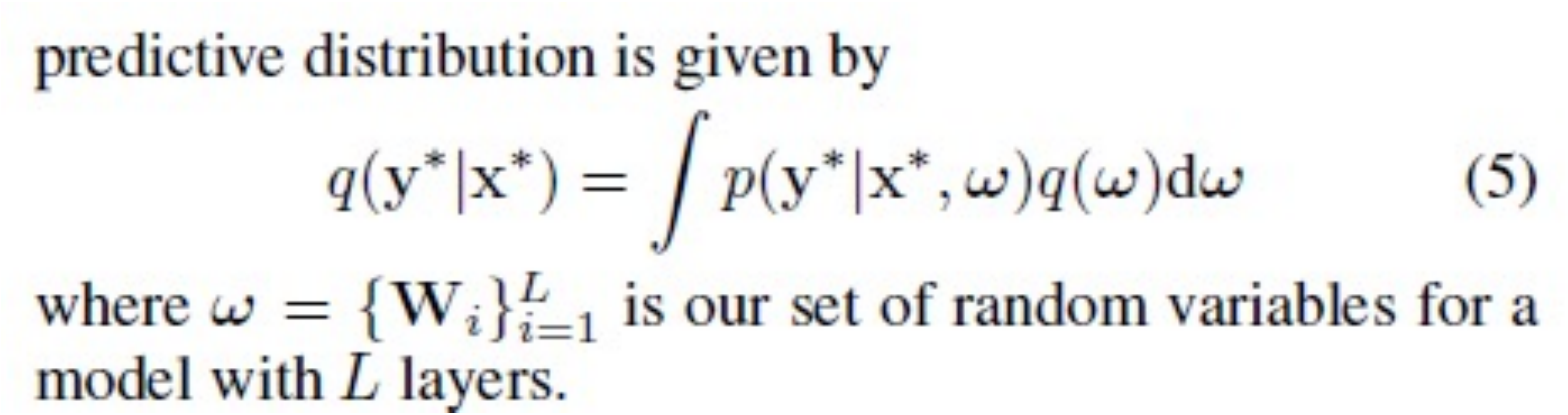
We minimise the KL divergence between the approximate posterior $q(\omega)$ above and the posterior of the full deep GP, $p(\omega|\mathbf{X}, \mathbf{Y})$. This KL is our minimisation objective

$$-\int q(\omega) \log p(\mathbf{Y}|\mathbf{X}, \omega) \mathrm{d}\omega + \mathrm{KL}(q(\omega)||p(\omega)). \qquad (3)$$

We rewrite the first term as a sum

$$-\sum_{n=1}^{N} \int q(\omega) \log p(\mathbf{y}_n|\mathbf{x}_n, \omega) \mathrm{d}\omega$$

Approximate predictive distribution

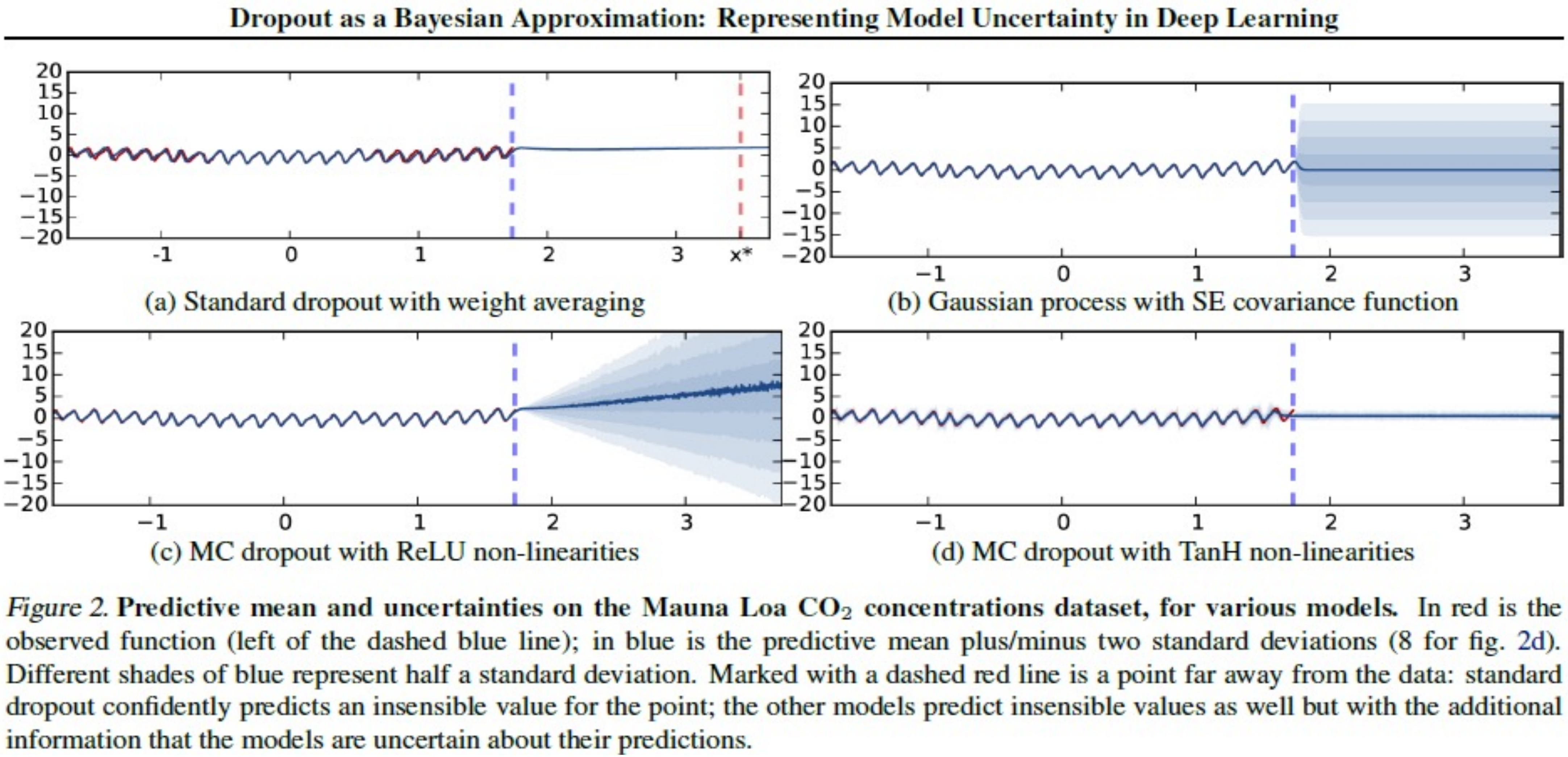


predictive distribution is given by

$$q(\mathrm{y}^*|\mathrm{x}^*) = \int p(\mathrm{y}^*|\mathrm{x}^*, \omega) q(\omega) \mathrm{d}\omega \qquad (5)$$

where $\omega = \{\mathrm{W}_i\}_{i=1}^{L}$ is our set of random variables for a model with L layers.

Experiments – Regression Task



Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

(a) Standard dropout with weight averaging

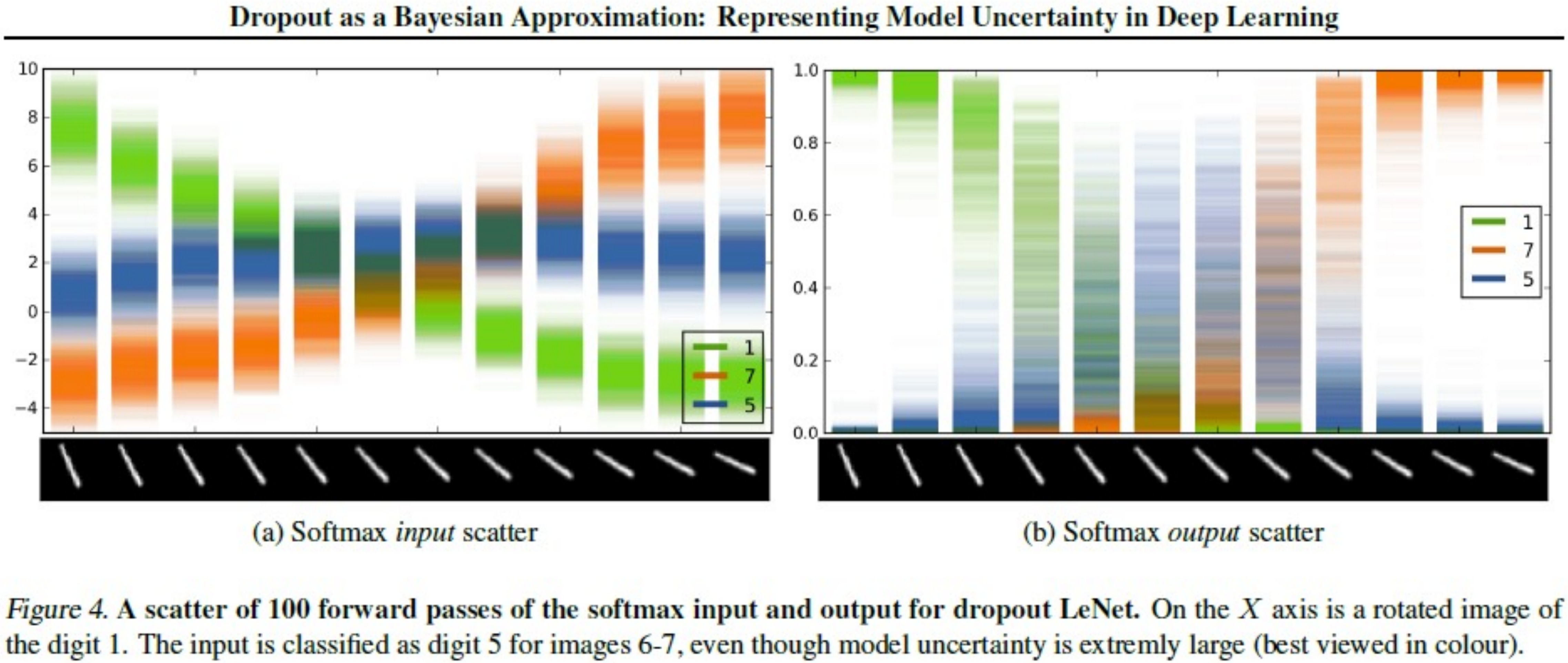
(b) Gaussian process with SE covariance function

(c) MC dropout with ReLU non-linearities

(d) MC dropout with TanH non-linearities

Figure 2. **Predictive mean and uncertainties on the Mauna Loa CO_2 concentrations dataset, for various models.** In red is the observed function (left of the dashed blue line); in blue is the predictive mean plus/minus two standard deviations (8 for fig. 2d). Different shades of blue represent half a standard deviation. Marked with a dashed red line is a point far away from the data: standard dropout confidently predicts an insensible value for the point; the other models predict insensible values as well but with the additional information that the models are uncertain about their predictions.

Experiments – Classification



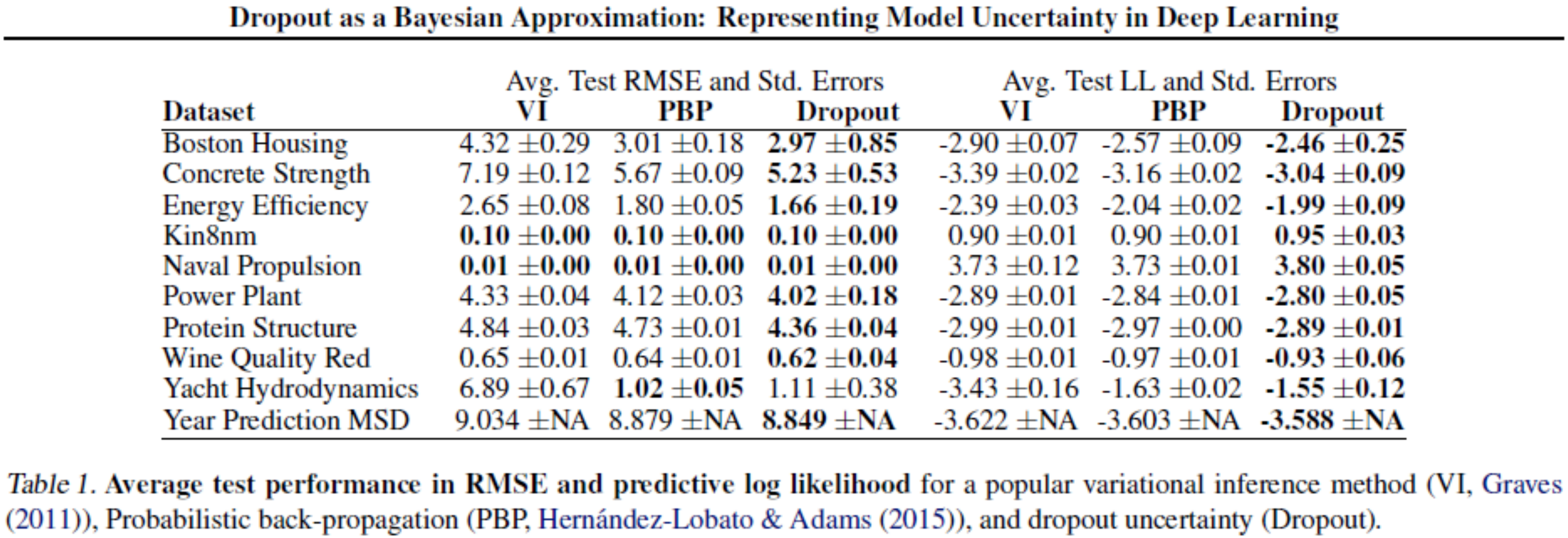
Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

(a) Softmax *input* scatter

(b) Softmax *output* scatter

Figure 4. **A scatter of 100 forward passes of the softmax input and output for dropout LeNet.** On the X axis is a rotated image of the digit 1. The input is classified as digit 5 for images 6-7, even though model uncertainty is extremly large (best viewed in colour).

Experiments

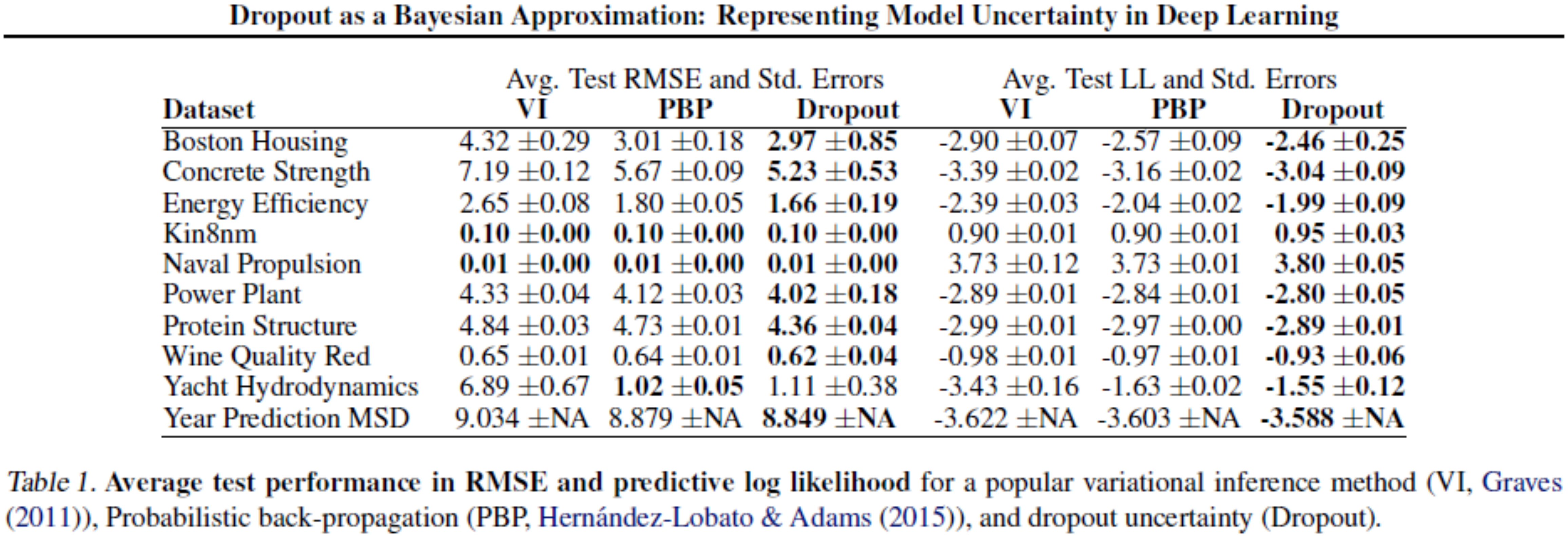


Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

	Avg. Test RMSE and Std. Errors			Avg. Test LL and Std. Errors		
Dataset	**VI**	**PBP**	**Dropout**	**VI**	**PBP**	**Dropout**
Boston Housing	4.32 ±0.29	3.01 ±0.18	**2.97 ±0.85**	-2.90 ±0.07	-2.57 ±0.09	**-2.46 ±0.25**
Concrete Strength	7.19 ±0.12	5.67 ±0.09	**5.23 ±0.53**	-3.39 ±0.02	-3.16 ±0.02	**-3.04 ±0.09**
Energy Efficiency	2.65 ±0.08	1.80 ±0.05	**1.66 ±0.19**	-2.39 ±0.03	-2.04 ±0.02	**-1.99 ±0.09**
Kin8nm	**0.10 ±0.00**	**0.10 ±0.00**	**0.10 ±0.00**	0.90 ±0.01	0.90 ±0.01	**0.95 ±0.03**
Naval Propulsion	**0.01 ±0.00**	**0.01 ±0.00**	**0.01 ±0.00**	3.73 ±0.12	3.73 ±0.01	**3.80 ±0.05**
Power Plant	4.33 ±0.04	4.12 ±0.03	**4.02 ±0.18**	-2.89 ±0.01	-2.84 ±0.01	**-2.80 ±0.05**
Protein Structure	4.84 ±0.03	4.73 ±0.01	**4.36 ±0.04**	-2.99 ±0.01	-2.97 ±0.00	**-2.89 ±0.01**
Wine Quality Red	0.65 ±0.01	0.64 ±0.01	**0.62 ±0.04**	-0.98 ±0.01	-0.97 ±0.01	**-0.93 ±0.06**
Yacht Hydrodynamics	6.89 ±0.67	**1.02 ±0.05**	1.11 ±0.38	-3.43 ±0.16	-1.63 ±0.02	**-1.55 ±0.12**
Year Prediction MSD	9.034 ±NA	8.879 ±NA	**8.849 ±NA**	-3.622 ±NA	-3.603 ±NA	**-3.588 ±NA**

Table 1. **Average test performance in RMSE and predictive log likelihood** for a popular variational inference method (VI, Graves (2011)), Probabilistic back-propagation (PBP, Hernández-Lobato & Adams (2015)), and dropout uncertainty (Dropout).

Experiments



Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

	Avg. Test RMSE and Std. Errors			Avg. Test LL and Std. Errors		
Dataset	**VI**	**PBP**	**Dropout**	**VI**	**PBP**	**Dropout**
Boston Housing	4.32 ±0.29	3.01 ±0.18	**2.97 ±0.85**	-2.90 ±0.07	-2.57 ±0.09	**-2.46 ±0.25**
Concrete Strength	7.19 ±0.12	5.67 ±0.09	**5.23 ±0.53**	-3.39 ±0.02	-3.16 ±0.02	**-3.04 ±0.09**
Energy Efficiency	2.65 ±0.08	1.80 ±0.05	**1.66 ±0.19**	-2.39 ±0.03	-2.04 ±0.02	**-1.99 ±0.09**
Kin8nm	**0.10 ±0.00**	**0.10 ±0.00**	**0.10 ±0.00**	0.90 ±0.01	0.90 ±0.01	**0.95 ±0.03**
Naval Propulsion	**0.01 ±0.00**	**0.01 ±0.00**	**0.01 ±0.00**	3.73 ±0.12	3.73 ±0.01	**3.80 ±0.05**
Power Plant	4.33 ±0.04	4.12 ±0.03	**4.02 ±0.18**	-2.89 ±0.01	-2.84 ±0.01	**-2.80 ±0.05**
Protein Structure	4.84 ±0.03	4.73 ±0.01	**4.36 ±0.04**	-2.99 ±0.01	-2.97 ±0.00	**-2.89 ±0.01**
Wine Quality Red	0.65 ±0.01	0.64 ±0.01	**0.62 ±0.04**	-0.98 ±0.01	-0.97 ±0.01	**-0.93 ±0.06**
Yacht Hydrodynamics	6.89 ±0.67	**1.02 ±0.05**	1.11 ±0.38	-3.43 ±0.16	-1.63 ±0.02	**-1.55 ±0.12**
Year Prediction MSD	9.034 ±NA	8.879 ±NA	**8.849 ±NA**	-3.622 ±NA	-3.603 ±NA	**-3.588 ±NA**

Table 1. **Average test performance in RMSE and predictive log likelihood** for a popular variational inference method (VI, Graves (2011)), Probabilistic back-propagation (PBP, Hernández-Lobato & Adams (2015)), and dropout uncertainty (Dropout).

END