Insurance pricing with Bayesian neural networks

Christopher Blier-Wong

École d'actuariat
Université Laval, Québec, Canada

christopher.blier-wong.1@ulaval.ca

May 30 2023

Context and motivation

- For actuarial fairness in insurance pricing, one requires a model with low bias
- To reduce bias (improve accuracy) of models, one can
  1. Increase quantity of data
  2. Increase model flexibility
- Data = expensive
- Recent years: increase popularity of machine learning models (gradient boosting, neural networks)
- For risk management, one requires a model that quantifies uncertainty

Motivation

Objective of the presentation

Explore the effects of parameter uncertainty within neural networks for insurance pricing

- How does one model parameter uncertainty within neural networks?
- How does this affect prior insurance pricing?
- How does this affect posterior insurance pricing?
- How well do estimation methods work for frequency modelling?

Overview

Model uncertainty

- Let $\mathbf{\Omega}$ be the model parameters
- Let the likelihood be of the shape $f_Y(y) = E_{\mathbf{\Omega}}[f_Y(y|\mathbf{\Omega})]$
- Law of total variance

$$Var(Y) = E_{\mathbf{\Omega}}[Var(Y|\mathbf{\Omega})] + Var_{\mathbf{\Omega}}(E[Y|\mathbf{\Omega}])$$

- Most machine learning models: process uncertainty only (randomness inherent to the data-generating process)
  - Implicit assumption: parameters are certain

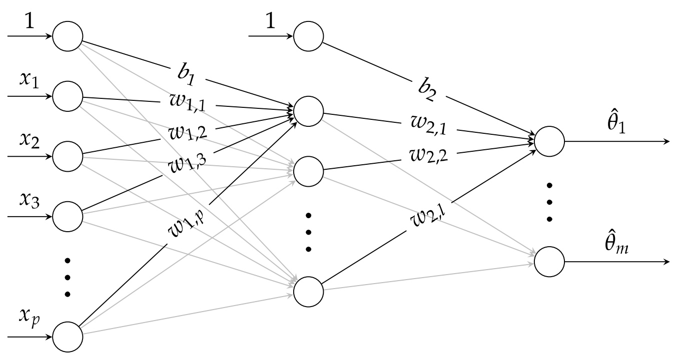
$$Var(Y) = E_{\mathbf{\Omega}}[Var(Y|\mathbf{\Omega})]$$

- Explored in mortality modeling [Cairns, 2000, Cairns et al., 2006]
- Is this assumption viable for insurance pricing ?

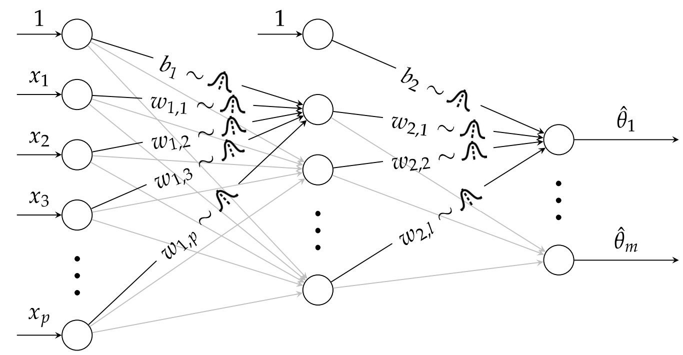
Neural networks

Adding model uncertainty

Neural network



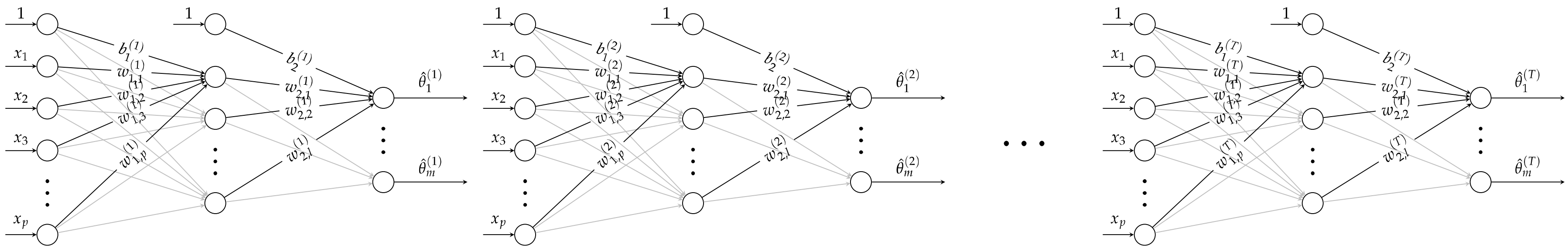
Bayesian neural network



- Bayesian GLM: [Frees et al., 2016], section 14.3
- Bayesian neural network: [Graves, 2011, Blundell et al., 2015, Gal, 2016]
- Applications in recommender systems, computer vision, medical imaging, natural language processing, time series forecasting [Wang and Yeung, 2020, Abdar et al., 2021]

Bayesian neural networks

Sample $\boldsymbol{\omega}^{(1)}$ Sample $\boldsymbol{\omega}^{(2)}$ Sample $\boldsymbol{\omega}^{(T)}$



Predictions from Bayesian neural networks:

$$\hat{\theta}_j = \frac{1}{T}\sum_{t=1}^{T}\hat{\theta}_j^{(t)}, \quad j = 1, \ldots, m$$

Consider the following notation

- $\boldsymbol{y}$: Response vector
- $\boldsymbol{X}$: Design matrix
- $\mathcal{D}$: Training dataset $\{\boldsymbol{x}_i, y_i; i = 1, \ldots, n\}$
- $\boldsymbol{\omega} \in \boldsymbol{\Omega}$: Set of all parameters
- $f(\boldsymbol{y}|\boldsymbol{\omega}, \boldsymbol{X})$: likelihood
- $\pi(\boldsymbol{\omega})$: posterior distribution of parameters

- Instead of gradient descent, one trains the model by finding the *posterior distribution of parameters*

$$
\begin{aligned}
\pi(\boldsymbol{\omega}|\mathcal{D}) &= \frac{f(\boldsymbol{Y}|\boldsymbol{\omega}, \boldsymbol{X})\pi(\boldsymbol{\omega})}{f(\boldsymbol{Y}|\boldsymbol{X})} \\
&= \frac{\prod_{i=1}^{n} f(\boldsymbol{y}_i|\boldsymbol{\omega}, \boldsymbol{x}_i)\pi(\boldsymbol{\omega})}{\int \cdots \int_{\Omega} \left[\prod_{i=1}^{n} f(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\omega})\pi(\boldsymbol{\omega})\right] d\boldsymbol{\omega}}
\end{aligned}
$$

- Denominator: intractable

Variational function

- Distribution $\pi(\boldsymbol{\omega}|\mathcal{D})$ is too complex to compute
- Alternative: MCMC (may be unsuitable for high dimensions)
- Alternative: compute a *variational function* that approximates $\pi(\boldsymbol{\omega}|\mathcal{D})$
- Example variational function: independent Gaussian weights (mean-field)

$$q_\psi(\boldsymbol{\omega}) = \prod_{i=\text{layers}} \prod_{j=\text{neurons}} \left(q_{\mu_{ij0},\sigma^2_{ij0}}(b_{ij}) \times \prod_{k=\substack{\text{next}\\ \text{neurons}}} q_{\mu_{ijk},\sigma^2_{ijk}}(w_{ijk}) \right)$$

- Variational inference: train parameters of $q_\psi(\boldsymbol{\omega})$ such that it approximates $\pi(\boldsymbol{\omega}|\mathcal{D})$
- [Graves, 2011, Blei et al., 2017]

Evidence lower bound

- Loss function for variational inference: evidence lower bound

$$\mathcal{L}_{VI}(\psi) = E_{q_\psi(\boldsymbol{\omega})}\left[\log f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\omega})\right] - KL(q_\psi(\boldsymbol{\omega})||\pi(\boldsymbol{\omega}))$$

Implications for prior pricing

Prediction

- Let variables for new customer be $\boldsymbol{x}^*$, objective: predict response variable Y^*
- Prediction given by

$$E[h(Y^*)|\boldsymbol{x}^*, \mathcal{D}] = E_{\boldsymbol{\Omega}}[E[h(Y^*)|\boldsymbol{x}^*, \mathcal{D}, \Omega]] \approx E_{\Psi}[E[h(Y^*)|\boldsymbol{x}^*, \mathcal{D}, \Psi]]$$
$$\underset{T\to\infty}{\longrightarrow} \frac{1}{T}\sum_{t=1}^{T} E[h(Y^*)|\boldsymbol{x}^*, \Psi = \tilde{\psi}_t, \mathcal{D}]$$

where

$$E[h(Y^*)|\boldsymbol{x}^*, \Omega, \mathcal{D}] = \int h(y) f(y|\boldsymbol{x}^*, \Omega) dy$$

Examples with the Poisson distribution

Example with a Poisson response variable, i.e. $Y|\Omega \sim Poisson$

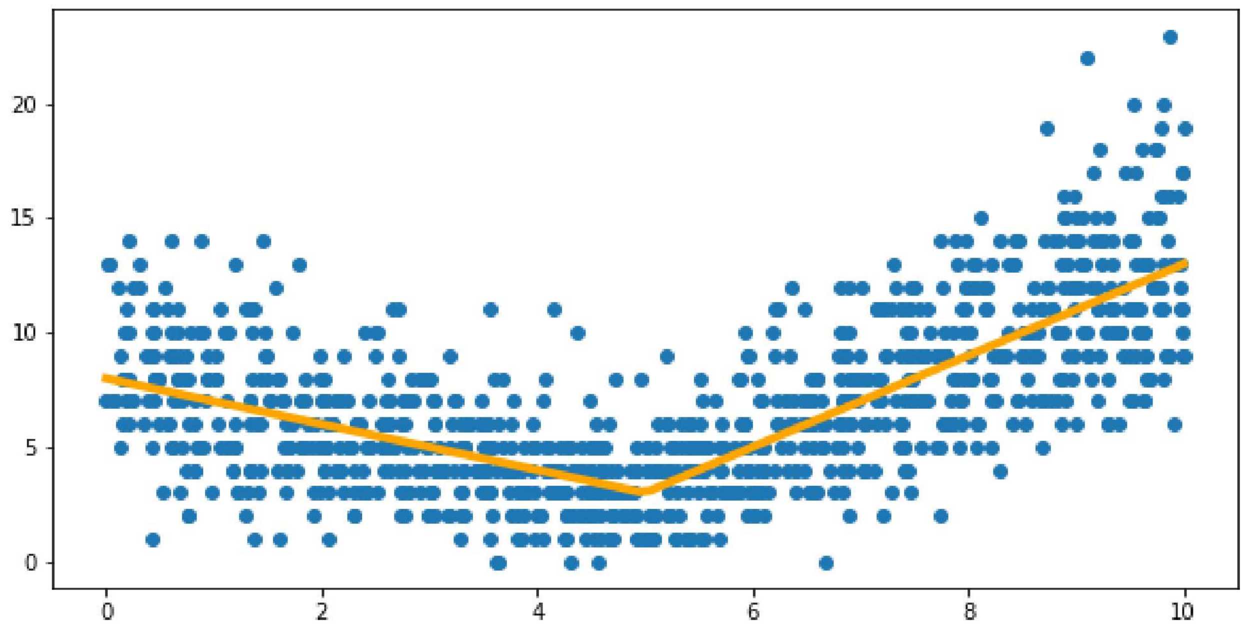
- Let $\tilde{\psi}_t$ be a random sample from the variational distribution $\Psi|\mathcal{D}$

$$E[Y^*|\mathcal{D}] = E[E[Y^*|\Omega, \mathcal{D}]] \approx E[E[Y^*|\Psi, \mathcal{D}]] \approx \frac{1}{T}\sum_{t=1}^{T} E[Y^*|\Psi = \tilde{\psi}_t] = \frac{1}{T}\sum_{t=1}^{T} \tilde{\lambda}_t = \overline{\lambda}.$$

$$\begin{aligned} Var(Y^*|\mathcal{D}) &= E[Var(Y^*|\Omega, \mathcal{D})] + Var(E[Y^*|\Omega, \mathcal{D}]) \\ &\approx E[Var(Y^*|\Psi, \mathcal{D})] + Var(E[Y^*|\Psi, \mathcal{D}]) \\ &= \overline{\lambda} + \frac{1}{T-1}\sum_{t=1}^{T} \left(\tilde{\lambda}_t - \overline{\lambda}\right)^2. \end{aligned}$$

Simulation data

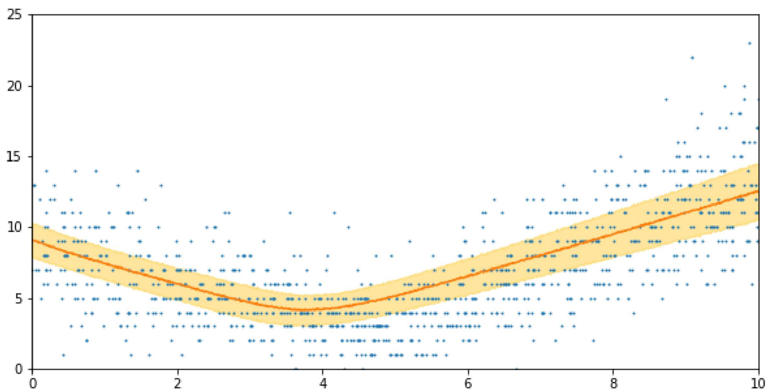
- See Example 2.1 from [Denuit et al., 2021]
- Consider a single covariate $X \sim Unif(0, 10)$
- Let $Y \sim Pois(\mu(x))$, with $\mu(x) = 8 - x + 3\max(x - 5, 0)$

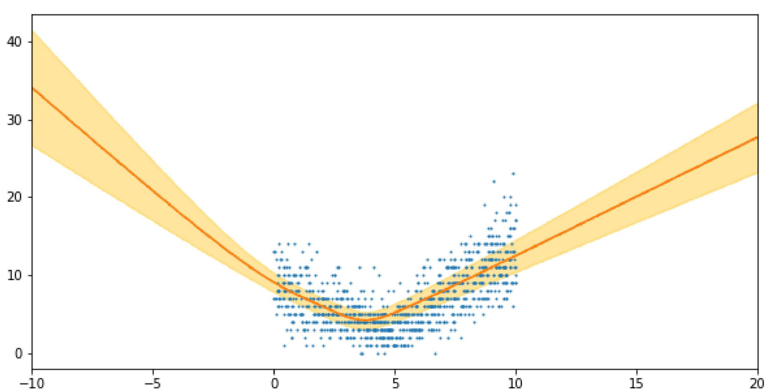


Simulation data

Confidence intervals for the rate parameter

- Train BNN with two hidden layers
- 20 hidden neurons in each hidden layer
- ReLU activations
- Confidence interval of the rate parameter: $\widehat{\mu(x)} \pm \sqrt{\widehat{Var(\mu(x))}}$





Implications for posterior pricing

Bayesian neural networks and posterior pricing

Given a model prediction μ and n periods of experience $\overline{Y}$, how should one combine the two quantities within a BLUP to minimize mean squared error ?

Define

- Hypothetical mean $\mu(\boldsymbol{\omega}) = E[Y^*|\boldsymbol{x}^*, \mathcal{D}, \boldsymbol{\Omega} = \boldsymbol{\omega}]$
- Process variance $\nu(\boldsymbol{\omega}) = Var(Y^*|\boldsymbol{x}^*, \mathcal{D}, \boldsymbol{\Omega} = \boldsymbol{\omega})$

Also,

$$\mu = E[\mu(\boldsymbol{\Omega})], \quad v = E[\nu(\boldsymbol{\Omega})], \quad a = Var(\mu(\boldsymbol{\Omega}))$$

Implications for posterior pricing

One has

$$E[Y^*] = \mu, \quad Var(Y^*) = v + a, \quad Cov(Y_1^*, Y_2^*) = a$$

Best weighted average of model prediction and n periods of actual experience:

$$Z\overline{Y} + (1 - Z)\mu, \quad Z = \frac{n}{n+k}, \quad k = \frac{E[Var(Y|\mathbf{\Omega})]}{Var(E[Y|\mathbf{\Omega}])}$$

Poisson case: Approximate with

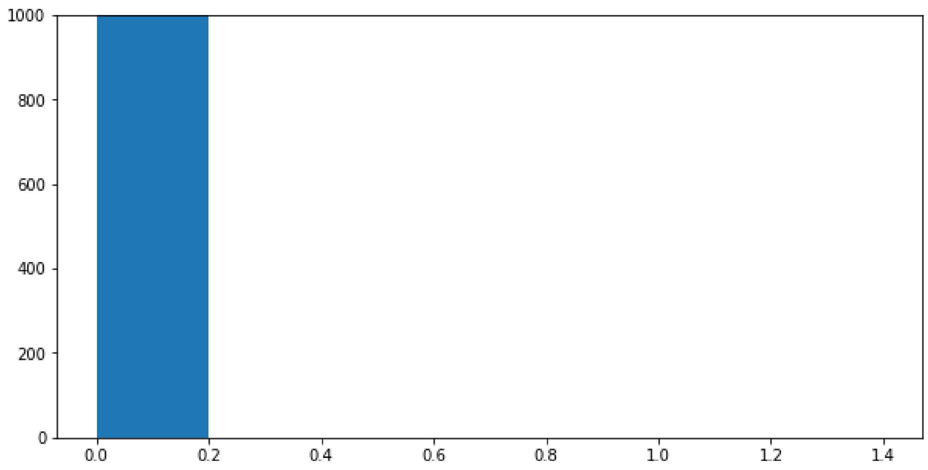
$$k \approx \frac{\frac{1}{T}\sum_{t=1}^{T}\tilde{\lambda}_t}{\frac{1}{T-1}\sum_{t=1}^{T}(\tilde{\lambda}_t - \overline{\lambda})^2},$$

where $\tilde{\lambda}_t, t = 1, \ldots T$ are samples from the variational distribution $\Lambda|\mathcal{D}$

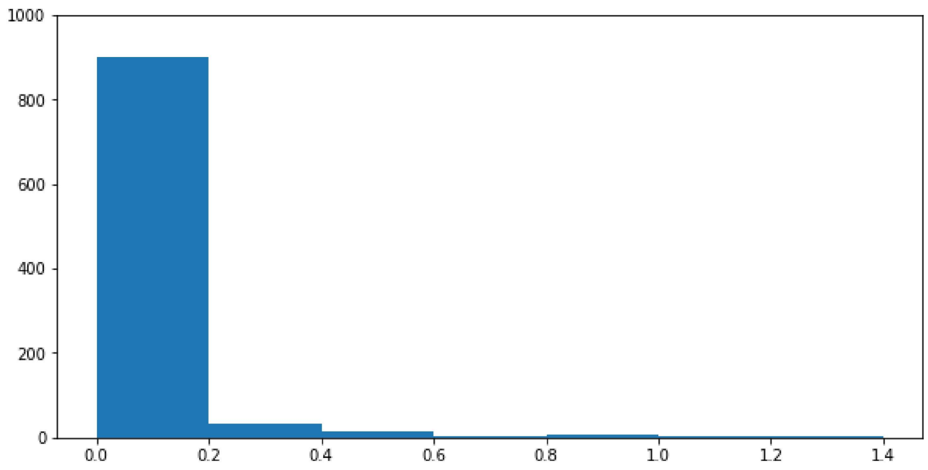
Implications for posterior pricing

- MTPL frequency data [Dutang and Charpentier, 2020]
- Poisson neural network
- Histograms of $1/k = \frac{Var(E[Y|\mathbf{\Omega}])}{E[Var(Y|\mathbf{\Omega})]}$

2×10



2×50

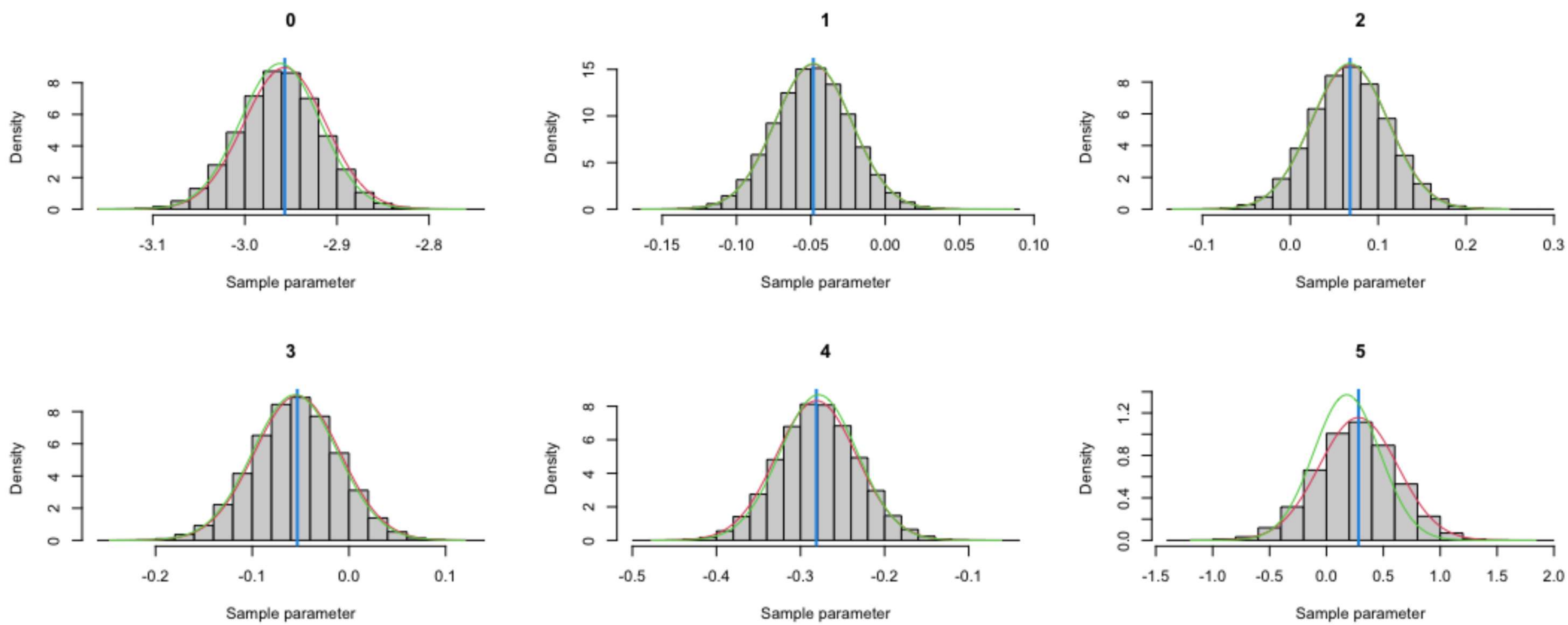


But how good is the approximation?

- To quantify the accuracy of the approximation, one must compare with a method where true uncertainty is available
- Return to the GLM framework where the loss function is convex
- In that case, the lowest possible predictive variance is given by the Cramer-Rao lower bound
- One obtains the parameter variance through the inverse Fisher information matrix
- One obtains the predictive variance through the Delta method

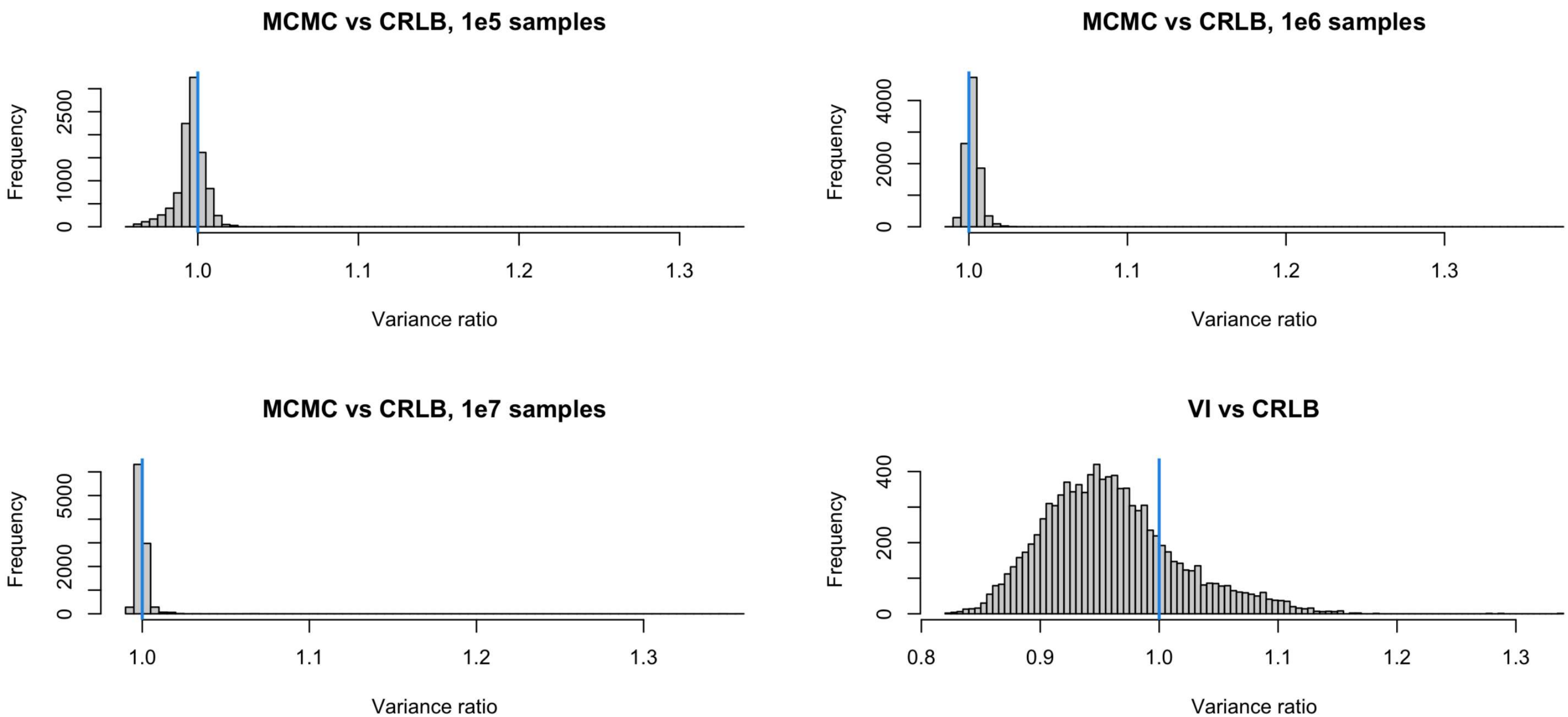
But how good is the approximation?

Comparing CRLB with MCMC estimation and mean-field VI in a GLM framework



But how good is the approximation?

Comparing CRLB with MCMC estimation and mean-field VI



Conclusions

- Bayesian neural networks capture parameter uncertainty
- Can identify where models are confident, where models fail
- Implications for prior pricing: increased predictive variance
- Implications for posterior pricing: update rules for BLUP
- Empirical finding: well calibrated neural networks with mean-field VI do not introduce much parameter uncertainty
- Mean-field VI typically underestimates predictive variance for frequency prediction

Conclusion

- Thanks for your attention!
- `christopher.blier-wong.1@ulaval.ca`

References I

Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U. R., Makarenkov, V., and Nahavandi, S. (2021).
A review of uncertainty quantification in deep learning: Techniques, applications and challenges.
Information Fusion, 76:243–297.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017).
Variational Inference: A Review for Statisticians.
Journal of the American Statistical Association, 112(518):859–877.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015).
Weight Uncertainty in Neural Networks.
arXiv:1505.05424 [cs, stat].

References II

Cairns, A. J. G. (2000).
A discussion of parameter and model uncertainty in insurance.
Insurance: Mathematics and Economics, 27(3):313–330.

Cairns, A. J. G., Blake, D., and Dowd, K. (2006).
A Two-Factor Model for Stochastic Mortality with Parameter Uncertainty: Theory and Calibration.
Journal of Risk & Insurance, 73(4):687–718.

Denuit, M., Charpentier, A., and Trufin, J. (2021).
Autocalibration and tweedie-dominance for insurance pricing with machine learning.
Insurance: Mathematics and Economics, 101:485–497.

References III

Dutang, C. and Charpentier, A. (2020).
CASdatasets: Insurance datasets.
R package version 1.0–11.

Frees, E. W., Derrig, R. A., and Meyers, G. (2016).
Predictive Modeling Applications in Actuarial Science, Volume I: Predictive Modeling Techniques.
International Series on Actuarial Science. Cambridge University Press, New York, NY.

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

References IV

Graves, A. (2011).
Practical Variational Inference for Neural Networks.
page 9.

Wang, H. and Yeung, D.-Y. (2020).
A Survey on Bayesian Deep Learning.
ACM Computing Surveys, 53(5):108:1–108:37.

Maximizing the ELBO

Update rule for all variational distribution parameters (mean and variance)

Input: Dataset $\boldsymbol{X}, \boldsymbol{Y}$, learning rate η;

Initialize ψ parameter randomly;

repeat

Sample a minibatch S of size M $\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{\omega}}$;

Compute the stochastic derivative with respect to ψ:

$$\widehat{\Delta\psi} \leftarrow \frac{N}{M} \sum_{i \in S} \frac{\partial}{\partial \psi} \log f(\boldsymbol{y}_i | \boldsymbol{x}_i, \hat{\boldsymbol{\omega}}) + \frac{\partial}{\partial \psi} KL(q_{\boldsymbol{\psi}}(\hat{\boldsymbol{\omega}}) || \pi(\hat{\boldsymbol{\omega}}))$$

Update ψ:

$$\psi \leftarrow \psi - \eta \widehat{\Delta\psi}$$

until *convergence of* ψ.;
