Incertitude paramétrique des réseaux de neurones en tarification IARD

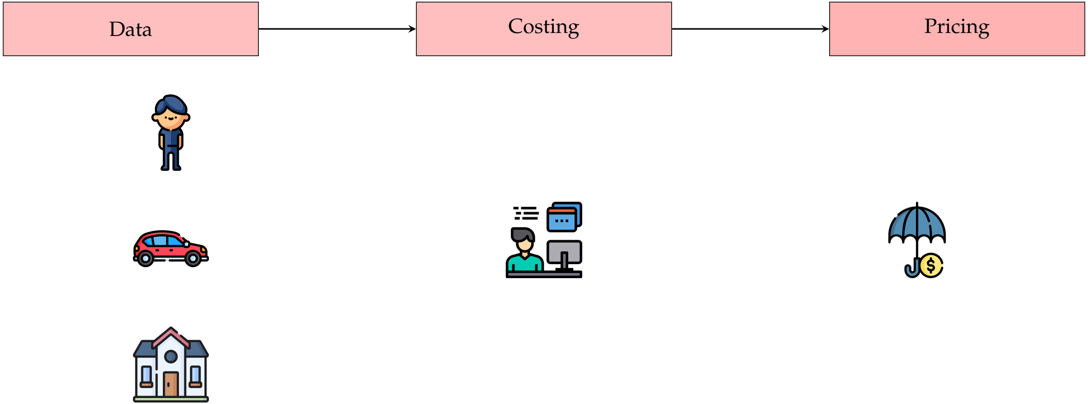
Christopher Blier-Wong

Joint work, Etienne Marceau, Hélène Cossette, Luc Lamontagne
Laboratoire ACT&RISK
École d'actuariat
Université Laval, Québec, Canada

christopher.blier-wong.1@ulaval.ca

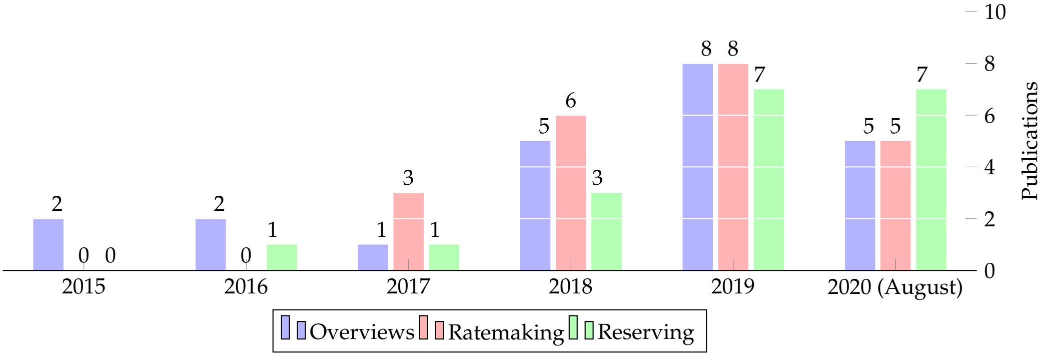
11 août 2021

Motivation



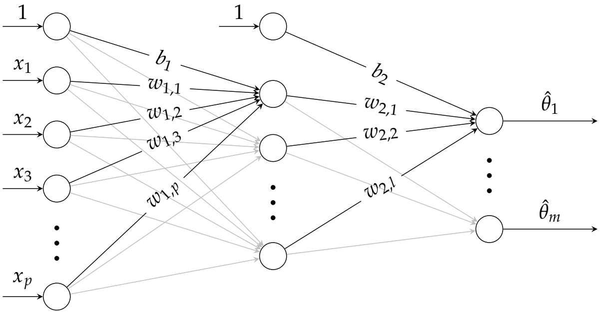
Motivation

Publications in P&C actuarial science using machine learning



Source : [Blier-Wong et al., 2021]

Neural networks



Neural networks for insurance pricing:

[Lowe and Pryor, 1996, Dugas et al., 2003, Noll et al., 2018, Ferrario et al., 2018, Schelldorfer and Wuthrich, 2019, Wuthrich, 2019b, Wuthrich, 2019a, Denuit et al., 2019, Richman and Wüthrich, 2020, Gao et al., 2021, Kuo and Richman, 2021, Shi and Shi, 2021, Delong et al., 2021]

Neural networks

- Let Y be the response variable, $\boldsymbol{x}$ be covariates
- Shape of a GLM:
$$E[Y] = g(\boldsymbol{x}\boldsymbol{\beta} + \beta_0),$$
where g is the inverse link function
- Shape of a neural network with one hidden layer:
$$E[Y] = g_2(\boldsymbol{h}\boldsymbol{W} + \boldsymbol{b}),$$
where
$$\boldsymbol{h} = g_1(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1)$$
- The vector $\boldsymbol{h}$ contains *intermediate representations* of the input variables $\boldsymbol{x}$

Motivation

Objective of the paper

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?

Overview

Model uncertainty

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

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

- Most machine learning models: process variance only
  - Implicit assumption: parameters are certain

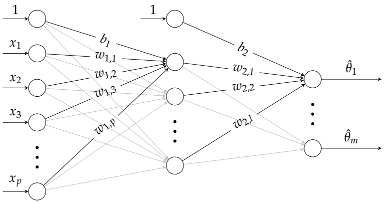
$$Var(Y) = E_{\boldsymbol{\Omega}}[Var(Y|\boldsymbol{\Omega})]$$

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

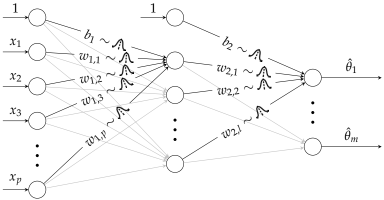
Neural networks

Adding model uncertainty

Neural network



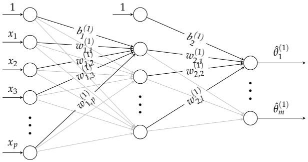
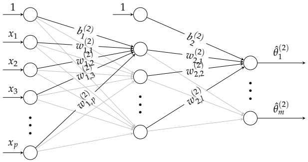
Bayesian neural network



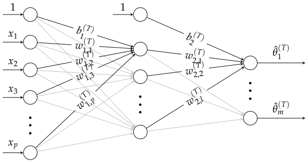
- Bayesian GLM: [Frees et al., 2016], section 14.3
- Bayesian neural network: [Graves, 2011, Blundell et al., 2015, Gal, 2016]
- Similar to chapter 2 of [Denuit et al., 2019]
- Applications in recommender systems, computer vision, medical imaging, natural language processing, time series forecasting [Wang and Yeung, 2020, Abdar et al., 2021]

Bayesian neural networks

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

$\cdots$



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 $\{x_i, y_i; i = 1, \ldots, n\}$
- $\boldsymbol{\omega} \in \boldsymbol{\Omega}$: 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 identify
- Integrating over conditionally independent weights & products of rvs
- Alternative: compute a *variational function* that approximates $\pi(\boldsymbol{\omega}|\mathcal{D})$
- Example variational function: independent weights

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

- 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

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

- Minibatch S of size M approximation:

$$\hat{\mathcal{L}}_{VI}(\psi) = \frac{n}{M}\sum_{i\in S} E_{q_\psi(\boldsymbol{\omega})}\left[\log f(\boldsymbol{y}_i|\boldsymbol{x}_i,\boldsymbol{\omega})\right] - KL(q_\psi(\boldsymbol{\omega})||\pi(\boldsymbol{\omega}))$$

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.;

Implications for prior pricing

Prediction

- Consider an arbitrary function $h(Y, \Omega)$ such that

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

- Estimation unconditional to Ω

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

Examples with the Poisson distribution

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

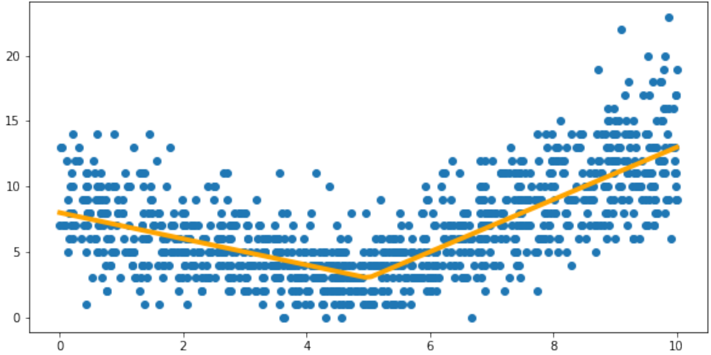
- Let variables for new customer be $\boldsymbol{x}^*$, objective: predict response variable Y^*
- Let $\tilde{\lambda}_t$ be a random sample from the variational distribution $\Lambda|\mathcal{D}$

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

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

Simulation data

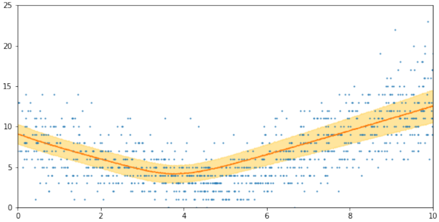
- See Example 2.1 from [Charpentier et al., 2021]
- Consider a single covariate $X \sim Unif(0, 10)$
- Let $Y \sim Pois(\mu(x))$, with $\mu(x) = 8 - x + 3\max(x - 5, 0)$

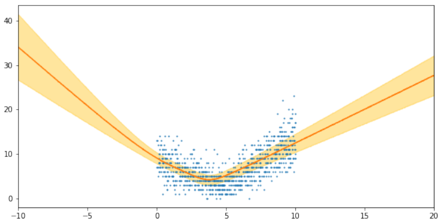


Simulation data

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: $\widehat{\mu(x)} \pm \sqrt{Var(\widehat{\mu(x)})}$

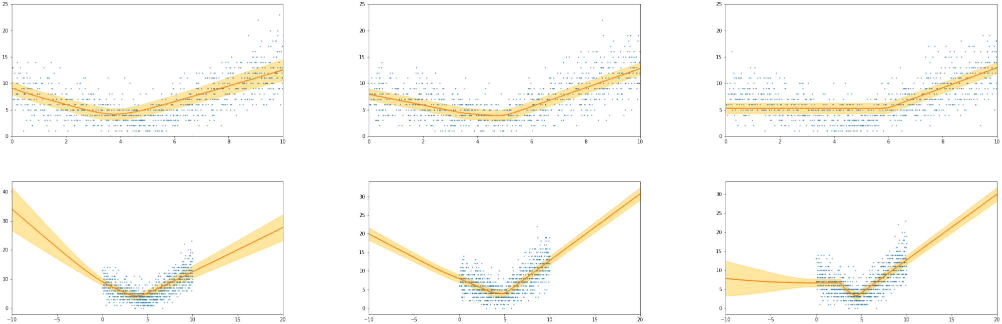




Simulation data

Confidence intervals for the rate parameter

2×20 $\quad$ 2×3 $\quad$ 2×3



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 $v(\boldsymbol{\omega}) = Var(Y^* | \boldsymbol{x}^*, \mathcal{D}, \boldsymbol{\Omega} = \boldsymbol{\omega})$

Also,

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

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:

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

Poisson case: Approximate with

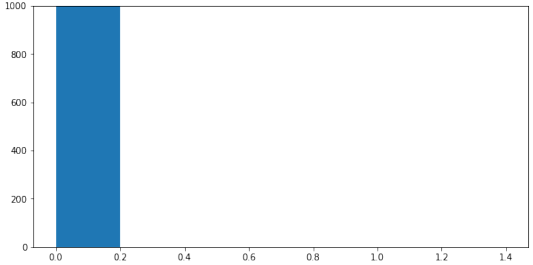
$$k \approx \frac{\sum_{t=1}^{T} \tilde{\lambda}_t}{\frac{1}{T-1}\sum_{t=1}^{T}(\tilde{\lambda}_t - \overline{\lambda})^2},$$

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

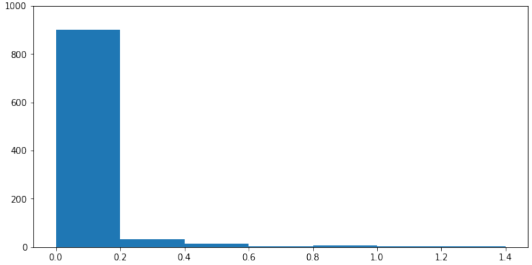
Implications for posterior pricing

- 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



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 predictions
- Empirical finding: well calibrated neural networks do not introduce much parameter uncertainty

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.
- Blier-Wong, C., Cossette, H., Lamontagne, L., and Marceau, E. (2021).
  Machine learning in P&C insurance: A review for pricing and reserving.
  Risks, 9(1):4.

References II

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

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.

References III

- Charpentier, A., Denuit, M., and Trufin, J. (2021).
  Autocalibration and Tweedie-dominance for Insurance Pricing with Machine Learning.
  arXiv:2103.03635 [cs, econ, stat].
- Delong, L., Lindholm, M., and Wuthrich, M. V. (2021).
  Gamma Mixture Density Networks and their application to modelling insurance claim amounts.
  page 38.
- Denuit, M., Hainaut, D., and Trufin, J. (2019).
  Effective Statistical Learning Methods for Actuaries III: Neural Networks and Extensions.
  Springer Actuarial. Springer International Publishing, Cham.

References IV

Dugas, C., Bengio, Y., Chapados, N., Vincent, P., Denoncourt, G., and Fournier, C. (2003).
Statistical Learning Algorithms Applied to Automobile Insurance Ratemaking, pages 137–197.
WORLD SCIENTIFIC.

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

Ferrario, A., Noll, A., and Wuthrich, M. V. (2018).
Insights from Inside Neural Networks.
SSRN Electronic Journal.

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.

Gao, G., Wang, H., and Wüthrich, M. V. (2021).
Boosting Poisson regression models with telematics car driving data.
Machine Learning.

References VI

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

Kuo, K. and Richman, R. (2021).
Embeddings and Attention in Predictive Modeling.
arXiv:2104.03545 [q-fin, stat].

Lowe, J. and Pryor, L. (1996).
Neural Networks *v*. GLMs in pricing general insurance.
In *1996 General Insurance Convention*, page 22.

Noll, A., Salzmann, R., and Wuthrich, M. V. (2018).
Case Study: French Motor Third-Party Liability Claims.
SSRN Electronic Journal.

Richman, R. and Wüthrich, M. V. (2020).
Nagging Predictors.
Risks, 8(3):83.

Schelldorfer, J. and Wuthrich, M. V. (2019).
Nesting Classical Actuarial Models into Neural Networks.
SSRN Electronic Journal.

Shi, P. and Shi, K. (2021).
Nonlife Insurance Risk Classification Using Categorical Embedding.
page 30.

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

References VIII

Wuthrich, M. V. (2019a).
Bias Regularization in Neural Network Models for General Insurance Pricing.
page 23.

Wuthrich, M. V. (2019b).
From Generalized Linear Models to Neural Networks, and Back.
SSRN Electronic Journal.