Statistical Modeling: From Generalized Linear Models to Neural Networks

Mario V. Wüthrich
RiskLab, ETH Zurich

March 24, 2021

Polskie Stowarzyszenie Aktuariuszy

Actuarial Data Science (ADS) initiative of the Swiss Association of Actuaries SAA

- The Swiss Association of Actuaries has a working party that
  - ⋆ produces tutorials on data science topics,
  - ⋆ provides data and code to explore these tools,
  - ⋆ provides education in terms of workshops.

- The tutorials can be downloaded from
  https://www.actuarialdatascience.org/ADS-Tutorials/

- The next workshop will take place on October 14-15, 2021:
  https://www.actuaries.ch/de/kurs/block_course_deep_learning_with_actuarial_applications_in_r/ereig!5860/

The modeling cycle

(1) data collection, data cleaning and data pre-processing ($\geq 80\%$ of total time)

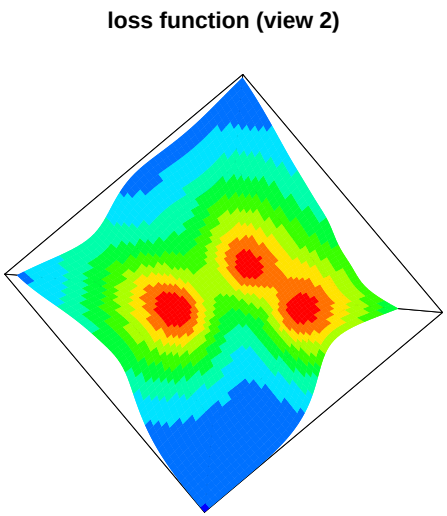
(2) selection of model class (data or algorithmic modeling culture, Breiman 2001)

(3) choice of objective function

(4) 'solving' a (non-convex) optimization problem

(5) model validation

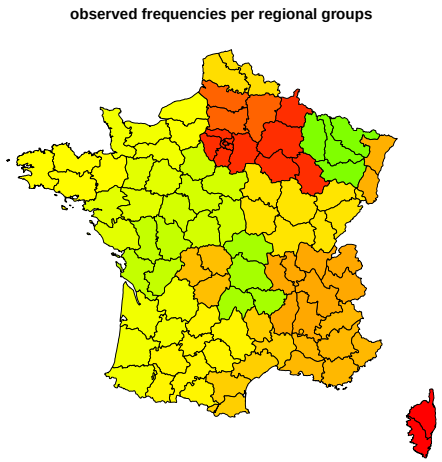
(6) possibly go back to (1)

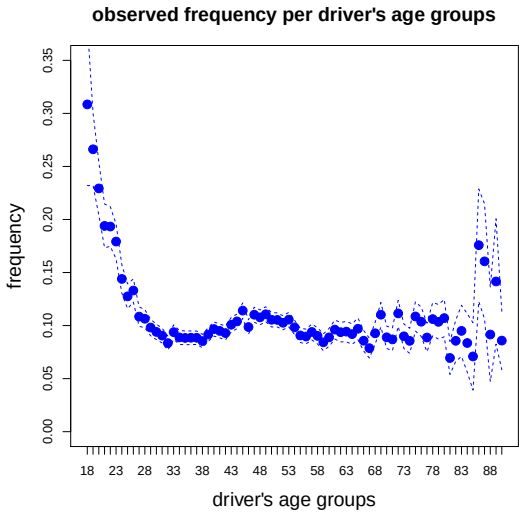


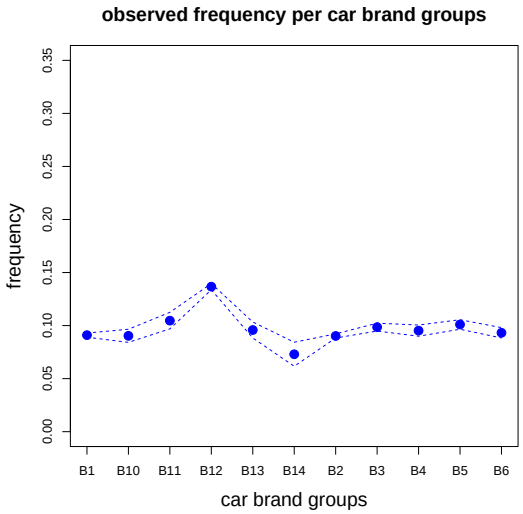
- ▷ 'solving' involves:
  - ⋆ choice of algorithm
  - ⋆ choice of stopping criterion, step size, etc.
  - ⋆ choice of seed (starting value)

Car insurance frequency example: tabular data

```
> str(freMTPL2freq)      #source R package CASdatasets
'data.frame':   678013 obs. of  12 variables:
 $ IDpol      : num  1 3 5 10 11 13 15 17 18 21 ...
 $ ClaimNb    : num 1 1 1 1 1 1 1 1 1 1 ...
 $ Exposure   : num  0.1 0.77 0.75 0.09 0.84 0.52 0.45 0.27 0.71 0.15 ...
 $ Area       : Factor w/ 6 levels "A","B","C","D",..: 4 4 2 2 2 5 5 3 3 2 ...
 $ VehPower   : int  5 5 6 7 7 6 6 7 7 7 ...
 $ VehAge     : int  0 0 2 0 0 2 2 0 0 0 ...
 $ DrivAge    : int  55 55 52 46 46 38 38 33 33 41 ...
 $ BonusMalus: int  50 50 50 50 50 50 50 68 68 50 ...
 $ VehBrand   : Factor w/ 11 levels "B1","B10","B11",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ VehGas     : Factor w/ 2 levels "Diesel","Regular": 2 2 1 1 1 2 2 1 1 1 ...
 $ Density    : int  1217 1217 54 76 76 3003 3003 137 137 60 ...
 $ Region     : Factor w/ 22 levels "R11","R21","R22",..: 18 18 3 15 15 8 8 20 20 12 ...
```







Generalized linear models (GLMs)

- Determine from data $\mathcal{D} = \{(Y_1, \boldsymbol{x}_1), \ldots, (Y_n, \boldsymbol{x}_n)\}$ an unknown regression function

$$\boldsymbol{x} \;\mapsto\; \mu(\boldsymbol{x}) = \mathbb{E}[Y].$$

- Selection of model class: Poisson GLM with canonical (log-)link:

$$\boldsymbol{x} \;\mapsto\; \mu_{\boldsymbol{\beta}}^{\mathrm{GLM}}(\boldsymbol{x}) \;=\; \exp\langle \boldsymbol{\beta}, \boldsymbol{x} \rangle \;=\; \exp\left\{ \beta_0 + \sum_j \beta_j x_j \right\}.$$

- Estimate regression parameter $\boldsymbol{\beta}$ with maximum likelihood $\widehat{\boldsymbol{\beta}}^{\mathrm{MLE}}$ by minimizing the corresponding deviance loss (objective function)

$$\boldsymbol{\beta} \;\mapsto\; \mathcal{L}_{\mathcal{D}}(\boldsymbol{\beta}).$$

Example: car insurance Poisson frequencies

After pre-processing the covariates $\boldsymbol{x}$:

	# param.	in-sample loss (in 10^{-2})	out-of-sample loss (in 10^{-2})
homogeneous ($\mu \equiv \text{const.}$)	1	32.935	33.861
Model GLM (Poisson)	48	31.257	32.149

Note for low frequency examples of, say, 5%: we have in the true model $\mathcal{L}_{\mathcal{D}} \approx 30.3 \cdot 10^{-2}$.

- This convex optimization problem has a unique optimal solution.
- The solution satisfies the balance property (under the canonical link choice)

$$\sum_{i=1}^{n} Y_i \;=\; \sum_{i=1}^{n} \exp\langle \widehat{\boldsymbol{\beta}}^{\mathrm{MLE}}, \boldsymbol{x}_i \rangle.$$

From GLMs to neural networks

- Example of a GLM (with log-link $\Rightarrow$ exponential output activation):

$$\boldsymbol{x} \;\mapsto\; \mu_{\boldsymbol{\beta}}^{\mathrm{GLM}}(\boldsymbol{x}) \;=\; \exp\langle \boldsymbol{\beta}, \boldsymbol{x} \rangle.$$

- Choose network of depth $d \in \mathbb{N}$ with network parameter $\theta = (\theta_{1:d}, \theta_{d+1})$:

$$\boldsymbol{x} \;\mapsto\; \mu_{\theta}^{\mathrm{NN}}(\boldsymbol{x}) \;=\; \exp\langle \theta_{d+1}, \boldsymbol{z} \rangle,$$

with neural network function (covariate pre-processing $\boldsymbol{x} \mapsto \boldsymbol{z}$)

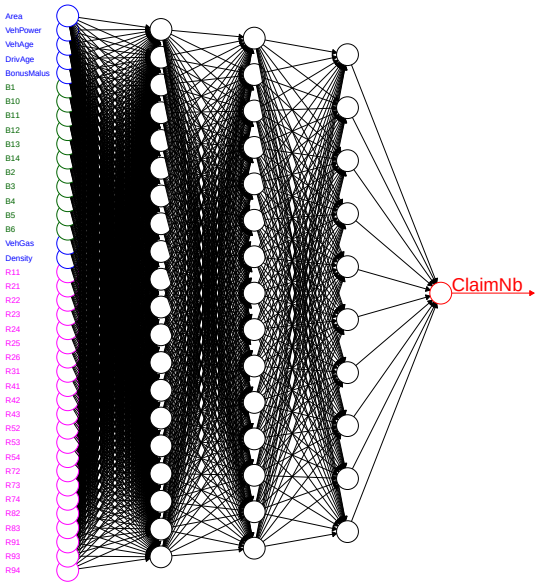
$$\boldsymbol{x} \;\mapsto\; \boldsymbol{z} \;=\; \boldsymbol{z}_{\theta_{1:d}}^{(d:1)}(\boldsymbol{x}) \;=\; \left(\boldsymbol{z}^{(d)} \circ \cdots \circ \boldsymbol{z}^{(1)} \right)(\boldsymbol{x}).$$

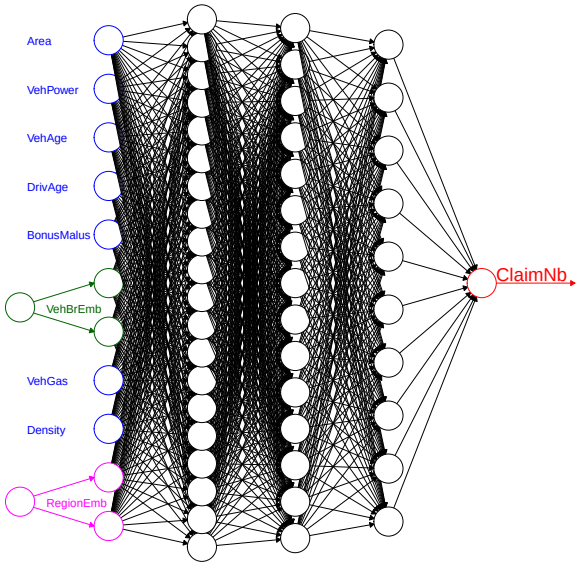
- For GLMs, actuary pre-processes $\boldsymbol{x}$; network does self-pre-processing $\boldsymbol{x} \mapsto \boldsymbol{z}$.

Neural network with embeddings

- Network of depth $d \in \mathbb{N}$ with network parameter θ

$$\boldsymbol{x} \;\mapsto\; \mu_\theta^{\mathrm{NN}}(\boldsymbol{x}) \;=\; \exp\langle \theta_{d+1}, \boldsymbol{z}\rangle \;=\; \exp\left\langle \theta_{d+1}, \left(\boldsymbol{z}^{(d)} \circ \cdots \circ \boldsymbol{z}^{(1)}\right)(\boldsymbol{x})\right\rangle.$$





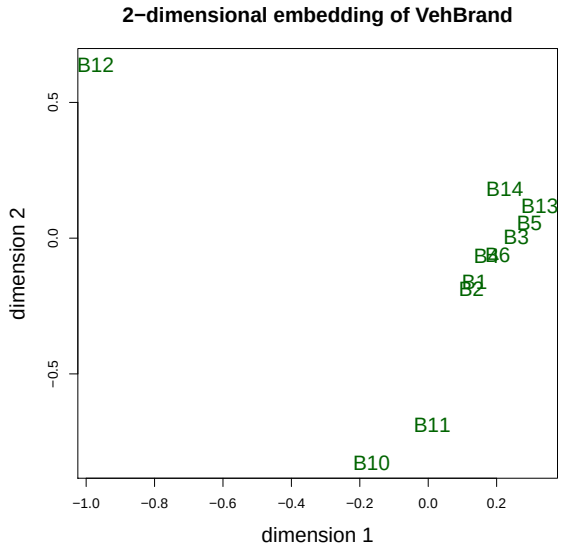
- Gradient descent method (GDM) provides $\widehat{\theta}$ w.r.t. deviance loss $\theta \mapsto \mathcal{L}_{\mathcal{D}}(\theta)$.

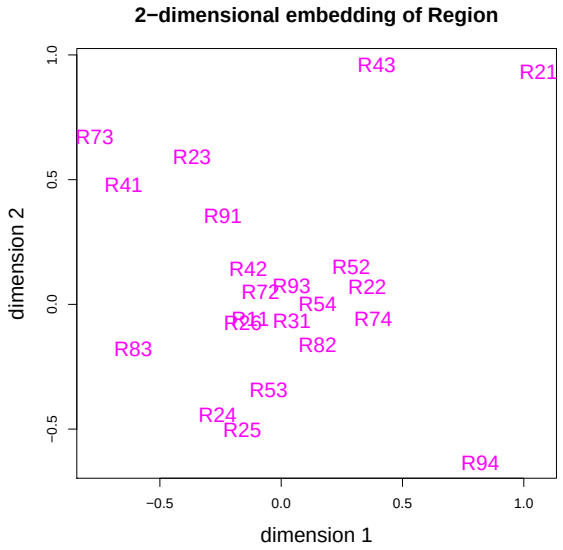
- Exercise early stopping of GDM because MLE over-fits (in-sample).

Network example: car insurance frequencies

	# param.	in-sample loss (in 10^{-2})	out-of-sample loss (in 10^{-2})
homogeneous ($\mu \equiv \text{const.}$)	1	32.935	33.861
Model GLM (Poisson)	48	31.257	32.149
network (2-dim. embeddings)	792	30.411	31.503

Note for low frequency examples of, say, 5%: we have in the true model $\mathcal{L}_{\mathcal{D}} \approx 30.3 \cdot 10^{-2}$.





Remarks on the neural network approach

+ Use embedding layers for categorical variables.

+ *Typically*, the neural network outperforms the GLM approach in terms of out-of-sample prediction accuracy.

– The neural network does not build on improving the GLM.

– Resulting prices are not unique, but depend on seeds.

– The neural network fails to have the balance property

$$\sum_{i=1}^{n} Y_i \overset{??}{=} \sum_{i=1}^{n} \exp\langle \widehat{\boldsymbol{\beta}}^{\mathrm{MLE}}, \boldsymbol{x}_i \rangle.$$

The modeling cycle: the optimization step

(1) data collection, data cleaning and data pre-processing ($\geq 80\%$ of total time)

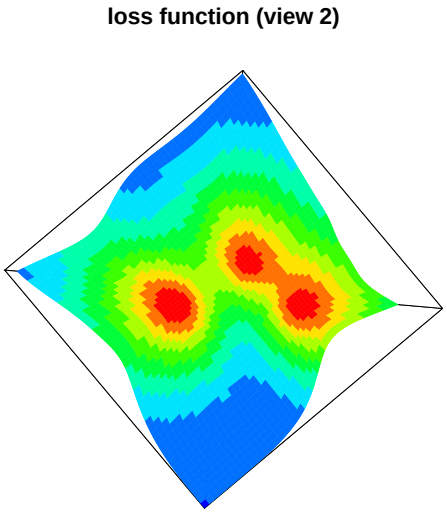
(2) selection of model class (data or algorithmic modeling culture, Breiman 2001)

(3) choice of objective function

(4) 'solving' a (non-convex) optimization problem

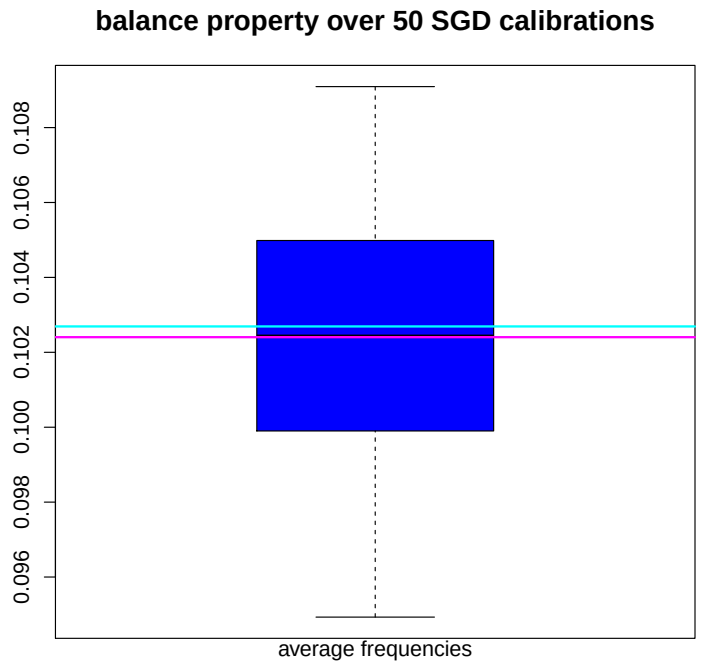
(5) model validation

(6) possibly go back to (1)



- ▷ 'solving' involves:
  - ⋆ choice of algorithm
  - ⋆ choice of stopping criterion, step size, etc.
  - ⋆ choice of seed (starting value)

Failure of balance property



- Boxplot of 50 gradient descent calibrations
- Cyan line: balance property
- Magenta line: average of 50 gradient descent calibrations
- Balance property fails to hold.

Regularization step for the balance property

- Apply an additional GLM step on the learned representation

$$\boldsymbol{x} \;\mapsto\; \boldsymbol{z} = \boldsymbol{z}_{\theta_{1:d}}^{(d:1)}(\boldsymbol{x}) = \left(\boldsymbol{z}^{(d)} \circ \cdots \circ \boldsymbol{z}^{(1)}\right)(\boldsymbol{x}),$$

  keeping the learned representation $\boldsymbol{z}$ fixed, ...

- ... that is, calculate MLE $\widehat{\theta}_{d+1}^{\mathrm{MLE}}$ of θ_{d+1} from regression function

$$\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{x}) \;\mapsto\; \exp\langle \theta_{d+1}, \boldsymbol{z} \rangle.$$

- This works for the canonical link of the chosen GLM.

- Regularization step is important, in particular, when there is a class imbalance!

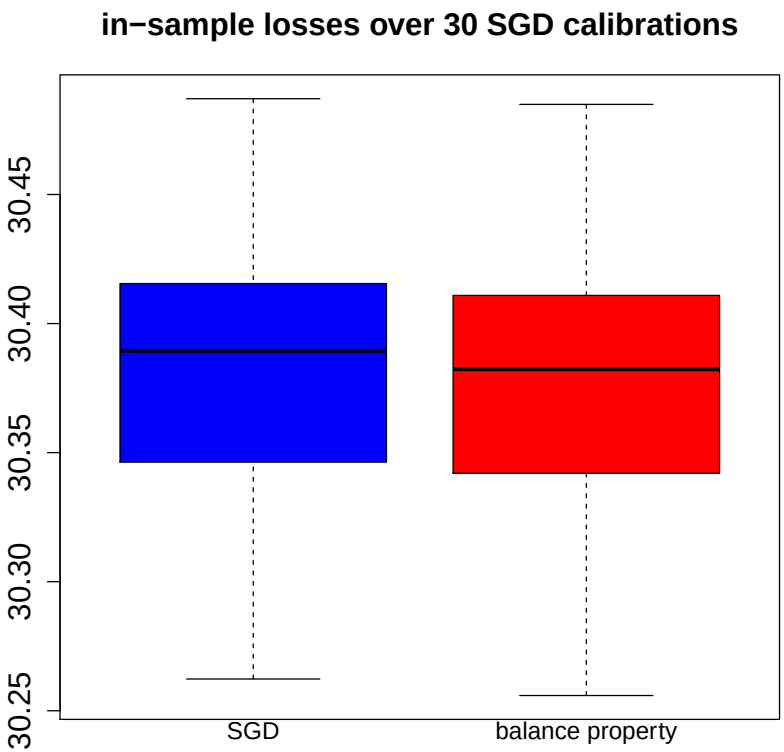
Network example: car insurance frequencies

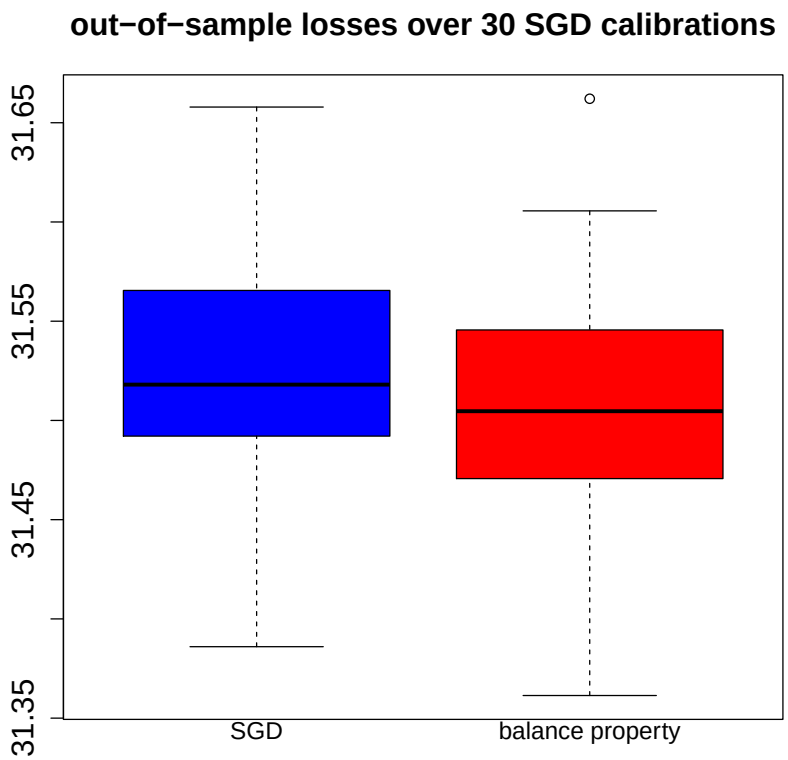
	# param.	in-sample loss (in 10^{-2})	out-of-sample loss (in 10^{-2})	average frequency
homogeneous ($\mu \equiv \text{const.}$)	1	32.935	33.861	10.02%
Model GLM (Poisson)	48	31.257	32.149	10.02%
network (2-dim. embeddings)	792	30.411	31.503	9.90%
regularized network, seed 1	792	30.408	31.488	10.02%
regularized network, seed 2	792	30.346	31.418	10.02%
regularized network, seed 3	792	30.303	31.462	10.02%

Note for low frequency examples of, say, 5%: we have in the true model $\mathcal{L}_{\mathcal{D}} \approx 30.3 \cdot 10^{-2}$

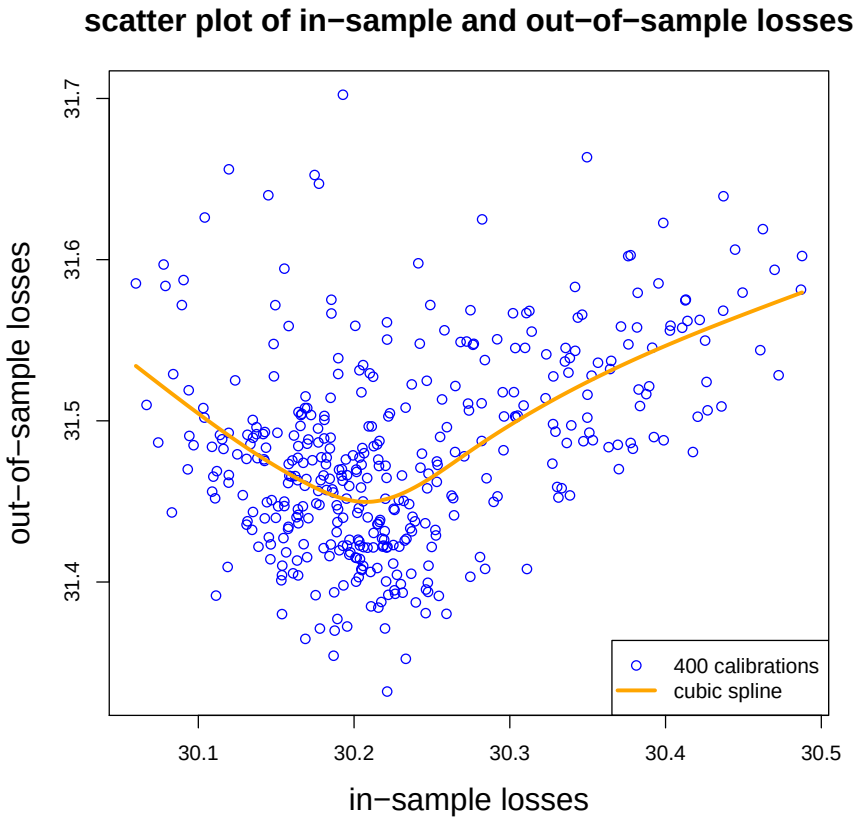
The average exposure is roughly 0.5 accounting years.

Uniqueness of optimal networks (1/2)





Uniqueness of optimal networks (2/2)



Aggregating predictors

- Similar to bootstrapping and bagging, aggregating over different predictors typically improves a predictive model, and it reduces the noise in the model.

- Every network provides an i.i.d. predictor, conditionally given the data $\mathcal{D}_I$. Conditionally i.i.d. because we choose i.i.d. seeds for every run of the gradient descent algorithm. Note: Bootstrap is different!

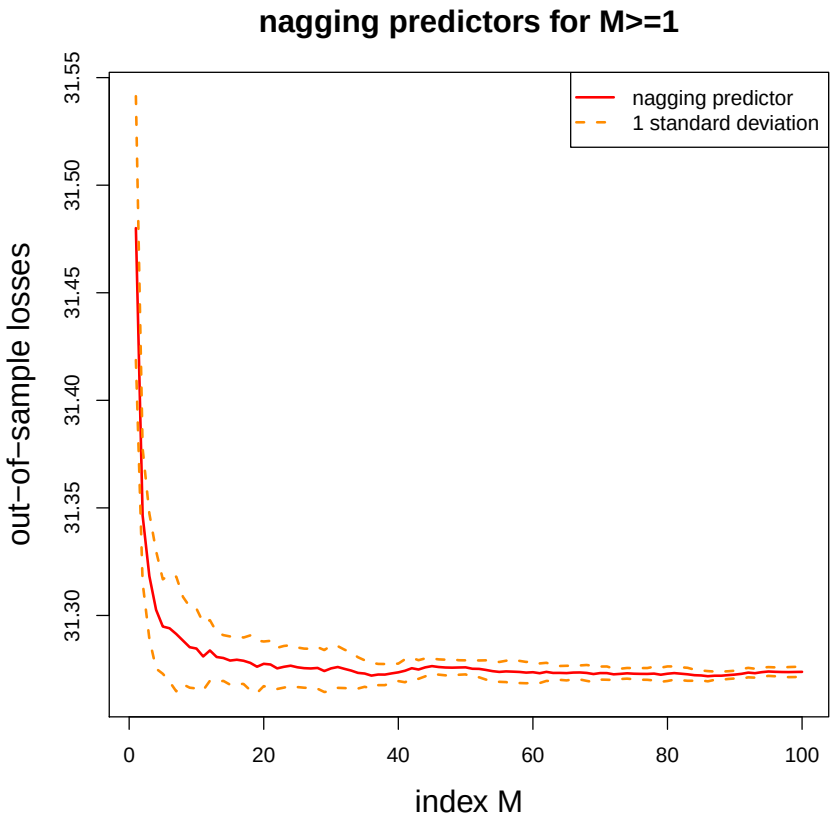
- This motivates predictor

$$\bar{\mu}^{(M)}(\boldsymbol{x}) = \frac{1}{M}\sum_{j=1}^{M} \widehat{\mu}^{\mathrm{NN}}_{\widehat{\theta}(j)}(\boldsymbol{x}),$$

  where $\widehat{\theta}^{(j)}$ are the network parameter estimates for different seeds $j = 1, \ldots, M$.

- We have law of large numbers (LNN) and central limit theorem (CLT) for $M \to \infty$.

Rate of convergence of aggregated predictor



- Decrease of out-of-sample losses of predictors $\bar{\mu}^{(M)}$ as a function of $M \to \infty$.
- We should average over $M = 20$ networks to have stability in portfolio results.

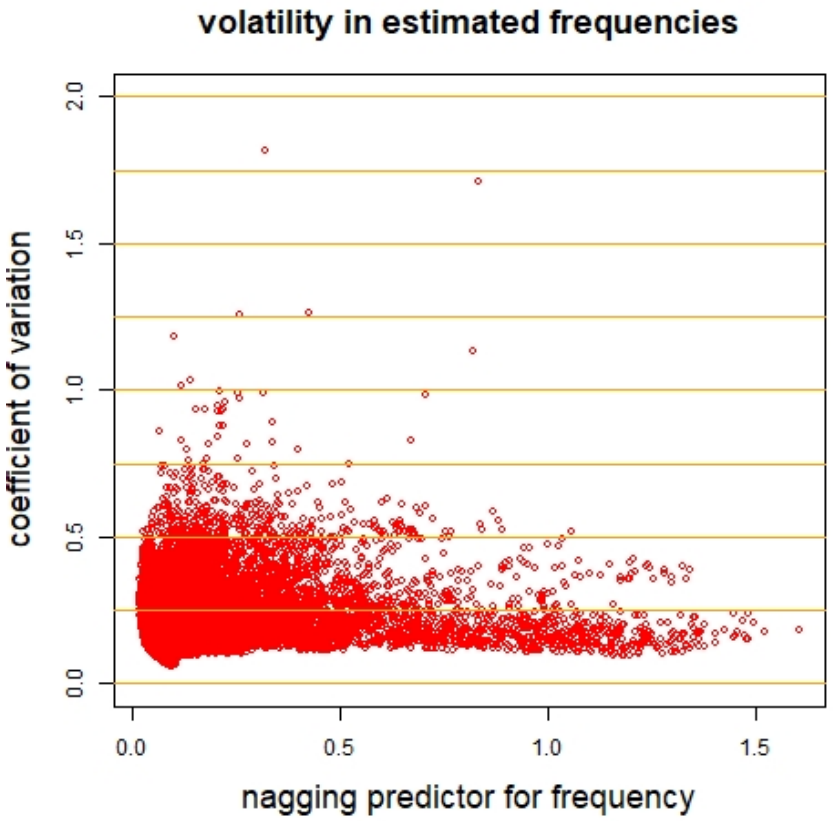
Network example: car insurance frequencies

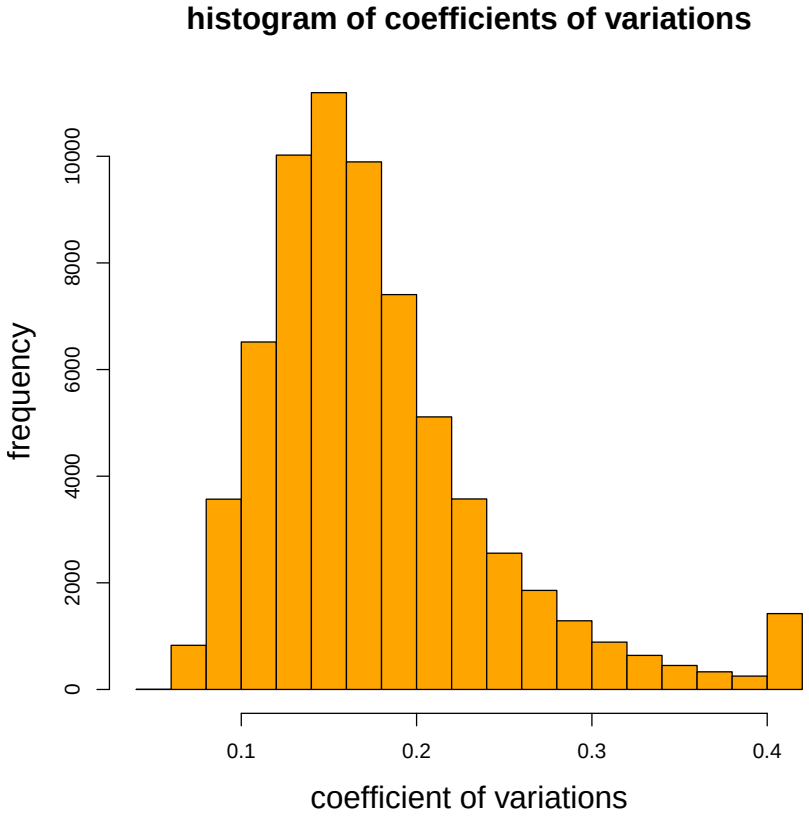
	# param.	in-sample loss (in 10^{-2})	out-of-sample loss (in 10^{-2})	average frequency
homogeneous ($\mu \equiv \text{const.}$)	1	32.935	33.861	10.02%
Model GLM (Poisson)	48	31.257	32.149	10.02%
network (2-dim. embeddings)	792	30.411	31.503	9.90%
regularized network, seed 1	792	30.408	31.488	10.02%
regularized network, seed 2	792	30.346	31.418	10.02%
regularized network, seed 3	792	30.303	31.462	10.02%
aggregated predictor for $M = 400$	–	30.060	31.272	10.02%

Note for low frequency examples of, say, 5%: we have in the true model $\mathcal{L}_{\mathcal{D}} \approx 30.3 \cdot 10^{-2}$

The average exposure is roughly 0.5 accounting years.

Stability on individual insurance policies





- Some insurance policies $\boldsymbol{x}_i$ have a coefficient of variation in individual estimates $\widehat{\mu}^{\mathrm{NN}}_{\widehat{\theta}^{(j)}}(\boldsymbol{x}_i)$ of more than 100%!

- This illustrates the difficulty and uncertainty involved in working with networks.

Take Aways

- A neural network as an extension of a GLM.
- Neural networks do covariate pre-processing themselves.
- Categorical covariates can be embedded in Euclidean spaces.
- 'Sufficiently good' network regression models are not unique.
- An additional GLM step allows us to comply with the balance property.
- Aggregating helps to improve the models.
- Individual network predictors involve a lot of uncertainty on insurance policy level and aggregating should be used to reduce this.
- There is a vastly growing literature on explaining networks.

This presentation is based on:

- From generalized linear models to neural networks, and back.
  SSRN Manuscript 3491790, (2019).

  `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3491790`

- Nesting classical actuarial models into neural networks (with J. Schelldorfer).
  SSRN Manuscript 3320525, (2019).

  `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3320525`

- Nagging predictors (with R. Richman).
  Risks 8/3, (2020), 83.

  `https://doi.org/10.3390/risks8030083`

- Interpreting deep learning models with marginal attribution by conditioning on quantiles (with M. Merz, R. Richman, A. Tsanakas).
  SSRN Manuscript 3809674, (2021).

  `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3809674`

Outlook: interpretability of neural networks