Neural Network Embedding of the Over-Dispersed Poisson Reserving Model

Andrea Gabrielli
RiskLab
ETH Zürich

Schweizerische Aktuarvereinigung SAV
110. Mitgliederversammlung

Luzern
30. August 2019

Idea

- CANN (Combined Actuarial Neural Network) approach
- Embedding of cross-classified over-dispersed Poisson (ccODP) reserving model into neural network architecture
- Starting point of neural network calibration: ccODP model

$\Longrightarrow$ Learning model structure beyond ccODP model (boosting)

Example Data

- Simulated from Individual Claims History Simulation Machine
- $1 \leq i \leq I$: accident years, $0 \leq j \leq J$: development delays
- Aggregated (incremental) payments $Y_{i,j}$ for all claims in LoB 1:

Accident year i	Development delay j											
	0	1	2	3	4	5	6	7	8	9	10	11
1	9'416	4'850	1'596	871	594	446	322	242	188	177	159	130
2	9'822	5'293	1'826	1'026	657	457	364	292	228	191	146	
3	9'613	4'903	1'665	970	594	443	325	263	212	176		
4	9'788	5'250	1'823	1'086	744	550	431	303	226			
5	9'955	5'722	2'089	1'159	791	558	458	354				
6	10'453	6'122	2'214	1'311	859	630	497					
7	11'130	6'476	2'401	1'356	890	677						
8	11'268	6'629	2'504	1'493	1'008							
9	11'475	6'953	2'648	1'478								
10	12'172	7'084	2'746									
11	12'816	8'028										
12	13'239											

Cross-Classified Over-Dispersed Poisson (ccODP) Model

- ODP model: $Y_{i,j}/\phi \overset{\text{ind.}}{\sim} \text{Poi}(\mu_{i,j}/\phi), \quad \phi > 0$
- Cross-classification: $\log \mu_{i,j} = \alpha_i + \beta_j$

$$\Longrightarrow \mathbb{E}[Y_{i,j}] = \text{Var}(Y_{i,j})/\phi = \mu_{i,j} = \exp\{\alpha_i + \beta_j\}$$

Cross-Classified Over-Dispersed Poisson (ccODP) Model

- ODP model: $Y_{i,j}/\phi \overset{\text{ind.}}{\sim} \text{Poi}(\mu_{i,j}/\phi), \quad \phi > 0$

- Cross-classification: $\log \mu_{i,j} = \alpha_i + \beta_j$

$$\Longrightarrow \mathbb{E}[Y_{i,j}] = \text{Var}(Y_{i,j})/\phi = \mu_{i,j} = \exp\{\alpha_i + \beta_j\}$$

- Minimize Poisson deviance statistics $\Longrightarrow$ MLEs $(\widehat{\alpha}_i)_i, (\widehat{\beta}_j)_j$

- Estimates: $\widehat{Y}_{i,j}^{\text{ODP}} = \widehat{\mu}_{i,j}^{\text{ODP}} = \exp\left\{\widehat{\alpha}_i + \widehat{\beta}_j\right\}$

- ODP reserves $= \sum_{i+j>I} \widehat{\mu}_{i,j}^{\text{ODP}} =$ Chain-ladder (CL) reserves

Results

- 6 LoBs from Individual Claims History Simulation Machine
- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	true claims reserves	39'689	37'037	16'878	71'630	72'548	31'117	268'899
(ii)	CL/ccODP reserves	38'569	35'460	15'692	67'574	70'166	29'409	256'870
(iii)								
(iv)								
(v)	bias CL/ccODP	-2.8%	-4.3%	-7.0%	-5.7%	-3.3%	-5.5%	-4.5%
(vi)								
(vii)								

Results

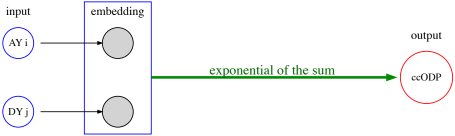
- 6 LoBs from Individual Claims History Simulation Machine
- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	true claims reserves	39'689	37'037	16'878	71'630	72'548	31'117	268'899
(ii)	CL/ccODP reserves	38'569	35'460	15'692	67'574	70'166	29'409	256'870
(iii)								
(iv)								
(v)	bias CL/ccODP	-2.8%	-4.3%	-7.0%	-5.7%	-3.3%	-5.5%	-4.5%
(vi)								
(vii)								

- Question: Can we do better?

$\Longrightarrow$ Embed ccODP model into neural network architecture

ccODP Model as Neural Network



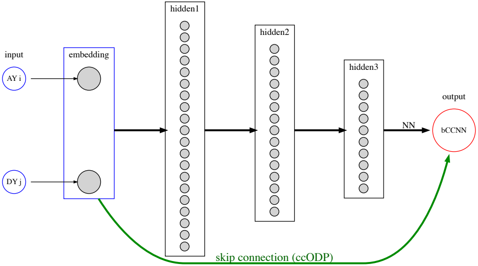
- Input layer: $(i,j) \in \{1,\ldots,I\} \times \{0,\ldots,J\}$
- Embedding layers:

$$\alpha(\cdot):\ \{1,\ldots,I\} \to \mathbb{R}, \qquad i \mapsto \alpha(i) = \widehat{\alpha}_i,$$

$$\beta(\cdot):\ \{0,\ldots,J\} \to \mathbb{R}, \qquad j \mapsto \beta(j) = \widehat{\beta}_j.$$

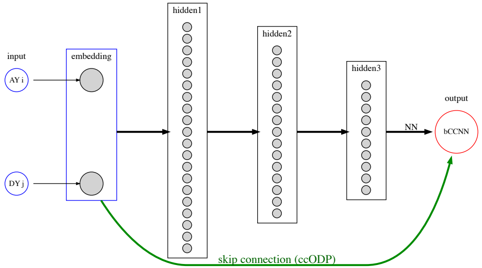
- ccODP: $\widehat{\mu}_{i,j}^{\mathrm{ODP}} = \exp\left\{\widehat{\alpha}_i + \widehat{\beta}_j\right\}$

Neural Network Embedding (1/3)



- Neural network: (non-linear) parametric regression function
- Input layer: $(i, j) \in \{1, \ldots, I\} \times \{0, \ldots, J\}$
- Embedding layers: $(i, j) \mapsto \left(\widehat{\alpha}_i, \widehat{\beta}_j\right)$

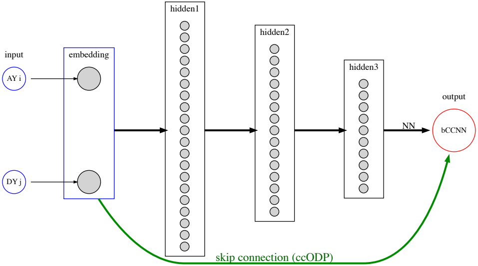
Neural Network Embedding (2/3)



- Three hidden layers with $(q_1, q_2, q_3) = (20, 15, 10)$
- First hidden layer: $\boldsymbol{z}^{(1)} = \left(z_1^{(1)}, \ldots, z_{q_1}^{(1)}\right) \in \mathbb{R}^{q_1}$, where

$$z_l^{(1)} = \tanh\left(b_l^{(1)} + w_{l,1}^{(1)}\,\widehat{\alpha}_i + w_{l,2}^{(1)}\,\widehat{\beta}_j\right) \in (-1, 1)$$

Neural Network Embedding (3/3)

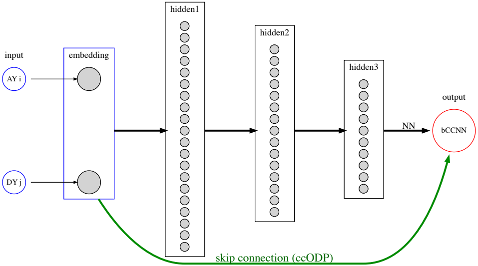


- Second hidden layer: $\boldsymbol{z}^{(2)} = \left(z_1^{(2)}, \ldots, z_{q_2}^{(2)}\right) \in \mathbb{R}^{q_2}$, where

$$z_l^{(2)} = \tanh\left(b_l^{(2)} + \langle \boldsymbol{w}_l^{(2)}, \boldsymbol{z}^{(1)} \rangle\right)$$

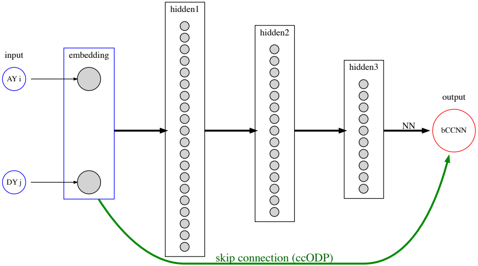
- Third hidden layer: $\boldsymbol{z}^{(3)} \in \mathbb{R}^{q_3}$

Blended Cross-Classified Neural Network (bCCNN)



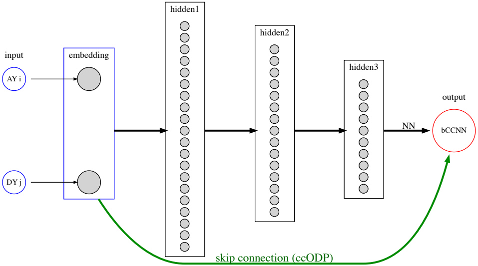
- **Output:** $\mu_{i,j}^{\mathrm{bCCNN}} = \exp\left\{ \qquad\qquad b + \langle \boldsymbol{w}, \boldsymbol{z}^{(3)}(i,j)\rangle \right\}$

Blended Cross-Classified Neural Network (bCCNN)



- Output: $\mu_{i,j}^{\mathrm{bCCNN}} = \exp\left\{\widehat{\alpha}_i + \widehat{\beta}_j + b + \langle \boldsymbol{w}, \boldsymbol{z}^{(3)}(i,j)\rangle\right\}$

Blended Cross-Classified Neural Network (bCCNN)



- Output: $\mu_{i,j}^{\text{bCCNN}} = \exp\left\{\widehat{\alpha}_i + \widehat{\beta}_j + b + \langle \boldsymbol{w}, \boldsymbol{z}^{(3)}(i,j)\rangle\right\}$
- Initialization: $b = 0,\ \boldsymbol{w} = \boldsymbol{0} \implies \mu_{i,j}^{\text{bCCNN}} = \exp\left\{\widehat{\alpha}_i + \widehat{\beta}_j\right\} = \widehat{\mu}_{i,j}^{\text{ODP}}$

$\implies$ Starting point of neural network calibration: ccODP model

Neural Network Calibration

- Neural network parameter: $\boldsymbol{\theta} \in \mathbb{R}^q$ with $q = 547$
- Minimize Poisson deviance statistics $\mathcal{L}(\boldsymbol{\theta})$ with gradient descent:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \rho \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}), \qquad \rho > 0$$

Neural Network Calibration

- Neural network parameter: $\boldsymbol{\theta} \in \mathbb{R}^q$ with $q = 547$
- Minimize Poisson deviance statistics $\mathcal{L}(\boldsymbol{\theta})$ with gradient descent:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \rho \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}), \qquad \rho > 0$$

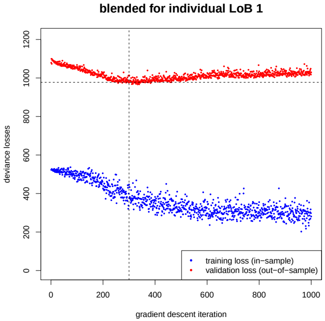
- Problem: For how long should we run gradient descent?
- Idea: Split claims $\Longrightarrow$ training triangle and validation triangle

Neural Network Calibration

- Neural network parameter: $\boldsymbol{\theta} \in \mathbb{R}^q$ with $q = 547$
- Minimize Poisson deviance statistics $\mathcal{L}(\boldsymbol{\theta})$ with gradient descent:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \rho \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}), \qquad \rho > 0$$

- Problem: For how long should we run gradient descent?
- Idea: Split claims $\Longrightarrow$ training triangle and validation triangle



Results

- 6 LoBs from Individual Claims History Simulation Machine
- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	true claims reserves	39'689	37'037	16'878	71'630	72'548	31'117	268'899
(ii)	CL/ccODP reserves	38'569	35'460	15'692	67'574	70'166	29'409	256'870
(iii)	bCCNN reserves							
(iv)								
(v)	bias CL/ccODP	-2.8%	-4.3%	-7.0%	-5.7%	-3.3%	-5.5%	-4.5%
(vi)	bias bCCNN							
(vii)								

Results

- 6 LoBs from Individual Claims History Simulation Machine
- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	true claims reserves	39'689	37'037	16'878	71'630	72'548	31'117	268'899
(ii)	CL/ccODP reserves	38'569	35'460	15'692	67'574	70'166	29'409	256'870
(iii)	bCCNN reserves	39'233	35'899	15'815	70'219	70'936	30'671	262'773
(iv)								
(v)	bias CL/ccODP	-2.8%	-4.3%	-7.0%	-5.7%	-3.3%	-5.5%	-4.5%
(vi)	bias bCCNN	-1.1%	-3.1%	-6.3%	-2.0%	-2.2%	-1.4%	-2.3%
(vii)								

Multiple LoB Model

- Input layer: $(i, j, m) \in \{1, \dots, I\} \times \{0, \dots, J\} \times \{1, \dots, 6\}$

- Embedding layers:

$$\alpha(\cdot) : \{1, \dots, I\} \to \mathbb{R}^6, \qquad i \mapsto \alpha(i) = \left(\widehat{\alpha}_{i|1}, \dots, \widehat{\alpha}_{i|6}\right),$$

$$\beta(\cdot) : \{0, \dots, J\} \to \mathbb{R}^6, \qquad j \mapsto \beta(j) = \left(\widehat{\beta}_{i|1}, \dots, \widehat{\beta}_{i|6}\right),$$

$$\gamma(\cdot) : \{1, \dots, 6\} \to \mathbb{R}, \qquad m \mapsto \gamma(m) = \gamma_m.$$

- Output: $\mu_{i,j,m}^{\mathrm{LoB}} = \exp\left\{\widehat{\alpha}_{i|m} + \widehat{\beta}_{j|m} + b + \langle \boldsymbol{w}, \boldsymbol{z}^{(3)}(i, j, m)\rangle\right\}$

Results

- 6 LoBs from Individual Claims History Simulation Machine
- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	true claims reserves	39'689	37'037	16'878	71'630	72'548	31'117	268'899
(ii)	CL/ccODP reserves	38'569	35'460	15'692	67'574	70'166	29'409	256'870
(iii)	bCCNN reserves	39'233	35'899	15'815	70'219	70'936	30'671	262'773
(iv)	multiple LoB reserves	40'271	37'027	16'400	70'563	73'314	30'730	268'305
(v)	bias CL/ccODP	-2.8%	-4.3%	-7.0%	-5.7%	-3.3%	-5.5%	-4.5%
(vi)	bias bCCNN	-1.1%	-3.1%	-6.3%	-2.0%	-2.2%	-1.4%	-2.3%
(vii)	bias multiple LoB	1.5%	0.0%	-2.8%	-1.5%	1.1%	-1.2%	-0.2%

Prediction Uncertainty (with Bootstrap)

- Conditional root mean square error of prediction (rmsep):

$$\begin{aligned}\mathrm{rmsep}\left(R^{\mathrm{true}}, R^{\mathrm{ODP}} \,\middle|\, \mathcal{D}_I\right) &= \sqrt{\mathbb{E}\left[\left(R^{\mathrm{true}} - R^{\mathrm{ODP}}\right)^2 \,\middle|\, \mathcal{D}_I\right]} \\ &= \sqrt{\mathrm{Var}\left(R^{\mathrm{true}} \middle| \mathcal{D}_I\right) + \left(R^{\mathrm{ODP}} - \mathbb{E}\left[R^{\mathrm{true}} \middle| \mathcal{D}_I\right]\right)^2}\end{aligned}$$

Prediction Uncertainty (with Bootstrap)

- Conditional root mean square error of prediction (rmsep):

$$\mathrm{rmsep}\left(R^{\mathrm{true}}, R^{\mathrm{ODP}} \middle| \mathcal{D}_I\right) = \sqrt{\mathbb{E}\left[\left(R^{\mathrm{true}} - R^{\mathrm{ODP}}\right)^2 \middle| \mathcal{D}_I\right]}$$

$$= \sqrt{\mathrm{Var}\left(R^{\mathrm{true}} \middle| \mathcal{D}_I\right) + \left(R^{\mathrm{ODP}} - \mathbb{E}\left[R^{\mathrm{true}} \middle| \mathcal{D}_I\right]\right)^2}$$

- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	rmsep CL/ccODP	1'076	1'316	475	2'150	1'938	975	3'528
(ii)	bias CL/ccODP	-1'120	-1'577	-1'186	-4'056	-2'382	-1'708	-12'029
(iii)								
(iv)								
(v)								
(vi)								

Prediction Uncertainty (with Bootstrap)

- Conditional root mean square error of prediction (rmsep):

$$\text{rmsep}\left(R^{\text{true}}, R^{\text{ODP}} \,\middle|\, \mathcal{D}_I\right) = \sqrt{\mathbb{E}\left[\left(R^{\text{true}} - R^{\text{ODP}}\right)^2 \,\middle|\, \mathcal{D}_I\right]}$$

$$= \sqrt{\text{Var}\left(R^{\text{true}} \,\middle|\, \mathcal{D}_I\right) + \left(R^{\text{ODP}} - \mathbb{E}\left[R^{\text{true}} \,\middle|\, \mathcal{D}_I\right]\right)^2}$$

- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	rmsep CL/ccODP	1'076	1'316	475	2'150	1'938	975	3'528
(ii)	bias CL/ccODP	-1'120	-1'577	-1'186	-4'056	-2'382	-1'708	-12'029
(iii)	rmsep bCCNN	1'171	1'299	508	2'105	2'029	1'072	3'607
(iv)	bias bCCNN	-456	-1'138	-1'063	-1'411	-1'612	-446	-6'126
(v)								
(vi)								

Prediction Uncertainty (with Bootstrap)

- Conditional root mean square error of prediction (rmsep):

$$\mathrm{rmsep}\left(R^{\mathrm{true}}, R^{\mathrm{ODP}} \middle| \mathcal{D}_I\right) = \sqrt{\mathbb{E}\left[\left(R^{\mathrm{true}} - R^{\mathrm{ODP}}\right)^2 \middle| \mathcal{D}_I\right]}$$
$$= \sqrt{\mathrm{Var}\left(R^{\mathrm{true}} \middle| \mathcal{D}_I\right) + \left(R^{\mathrm{ODP}} - \mathbb{E}\left[R^{\mathrm{true}} \middle| \mathcal{D}_I\right]\right)^2}$$

- Results:

		LoB 1	LoB 2	LoB 3	LoB 4	LoB 5	LoB 6	total
(i)	rmsep CL/ccODP	1'076	1'316	475	2'150	1'938	975	3'528
(ii)	bias CL/ccODP	-1'120	-1'577	-1'186	-4'056	-2'382	-1'708	-12'029
(iii)	rmsep bCCNN	1'171	1'299	508	2'105	2'029	1'072	3'607
(iv)	bias bCCNN	-456	-1'138	-1'063	-1'411	-1'612	-446	-6'126
(v)	rmsep multiple LoB	1'102	1'357	498	2'098	1'989	1'033	3'757
(vi)	bias multiple LoB	582	-10	-478	-1'067	766	-387	-594

Relative Model Differences

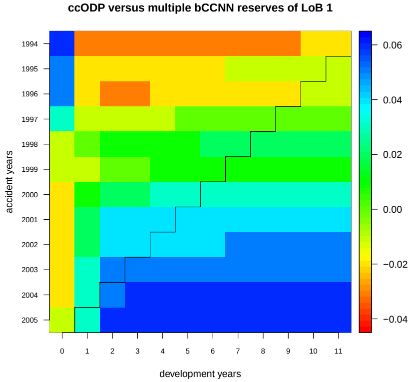
- For each cell (i, j):

$$\frac{\widehat{\mu}_{i,j,\cdot}^{\mathrm{LoB}} - \widehat{\mu}_{i,j}^{\mathrm{ODP}}}{\widehat{\mu}_{i,j}^{\mathrm{ODP}}}$$

Relative Model Differences

- For each cell (i, j):

$$\frac{\widehat{\mu}^{\text{LoB}}_{i,j,\cdot} - \widehat{\mu}^{\text{ODP}}_{i,j}}{\widehat{\mu}^{\text{ODP}}_{i,j}}$$



$\Longrightarrow$ slower payout pattern in more recent accident years

Cumulative Development Factors

- ccODP model:

$$f_j^{\mathrm{ODP}} = \frac{\sum_{l=0}^{j} \widehat{\mu}_{i,l}^{\mathrm{ODP}}}{\widehat{\mu}_{i,0}^{\mathrm{ODP}}} = \frac{\sum_{l=0}^{j} \exp\left\{\widehat{\beta}_l\right\}}{\exp\left\{\widehat{\beta}_0\right\}}$$

- Multiple LoB model:

$$f_{i,j}^{\mathrm{LoB}} = \frac{\sum_{l=0}^{j} \widehat{\mu}_{i,l,\cdot}^{\mathrm{LoB}}}{\widehat{\mu}_{i,0,\cdot}^{\mathrm{LoB}}}$$

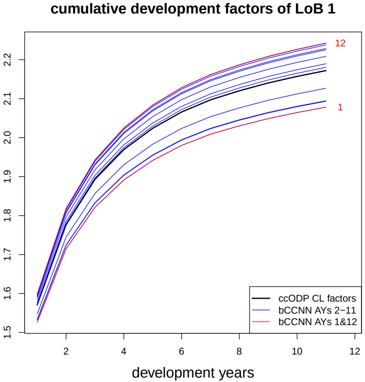
Cumulative Development Factors

- ccODP model:

$$f_j^{\mathrm{ODP}} = \frac{\sum_{l=0}^{j} \widehat{\mu}_{i,l}^{\mathrm{ODP}}}{\widehat{\mu}_{i,0}^{\mathrm{ODP}}} = \frac{\sum_{l=0}^{j} \exp\left\{\widehat{\beta}_l\right\}}{\exp\left\{\widehat{\beta}_0\right\}}$$

- Multiple LoB model:

$$f_{i,j}^{\mathrm{LoB}} = \frac{\sum_{l=0}^{j} \widehat{\mu}_{i,l,\cdot}^{\mathrm{LoB}}}{\widehat{\mu}_{i,0,\cdot}^{\mathrm{LoB}}}$$



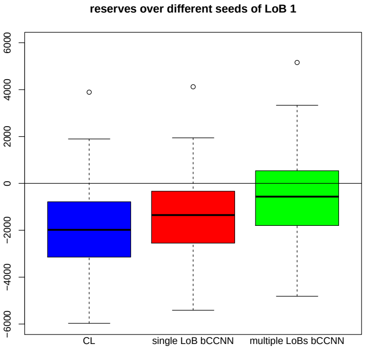
$\Longrightarrow$ accident year dependent cumulative development factors

Bias (100 Datasets)

- Simulate 100 datasets from Individual Claims History Simulation Machine
- 100 biases for:
  - ccODP/CL model
  - bCCNN model
  - multiple LoB model

Bias (100 Datasets)

- Simulate 100 datasets from Individual Claims History Simulation Machine
- 100 biases for:
  - ccODP/CL model
  - bCCNN model
  - multiple LoB model



$\Longrightarrow$ learning additional model structure

Conclusions

- Learning additional model structure through embedding
- Number of training iterations has to be chosen carefully
- Small number of iterations allows us to apply bootstrap
- Extension: embedding of numbers of claims and payments

References

1. Gabrielli, A. (2019). A neural network boosted double over-dispersed Poisson claims reserving model. *SSRN Manuscript*, ID 3365517.
2. Gabrielli, A., Richman, R., Wüthrich, M.V. (2018). Neural network embedding of the over-dispersed Poisson reserving model. *SSRN Manuscript*, ID 3288454 (to be published in the *Scandinavian Actuarial Journal*).
3. Gabrielli, A., Wüthrich, M.V. (2018). An individual claims history simulation machine. *Risks* **6/2**, 29.
4. Wüthrich, M.V., Merz, M. (2019). Editorial: Yes, we CANN! *ASTIN Bulletin* **49/1**, 1-3.