Neural networks for insurance pricing with frequency and severity data:

a benchmark study from data preprocessing to technical tariff

Freek Holvoet
LRisk - KU Leuven

7 November 2023

Presenter + Authors

Freek Holvoet

Katrien Antonio

Roel Henckaerts

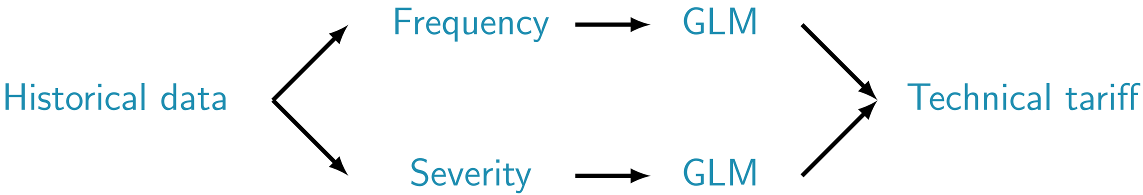
Holvoet, F., Antonio, K., & Henckaerts, R. (2023). Neural networks for insurance pricing with frequency and severity data: a benchmark study from data preprocessing to technical tariff. *arXiv preprint arXiv:2310.12671*.

Introduction

Pricing in non-life insurance

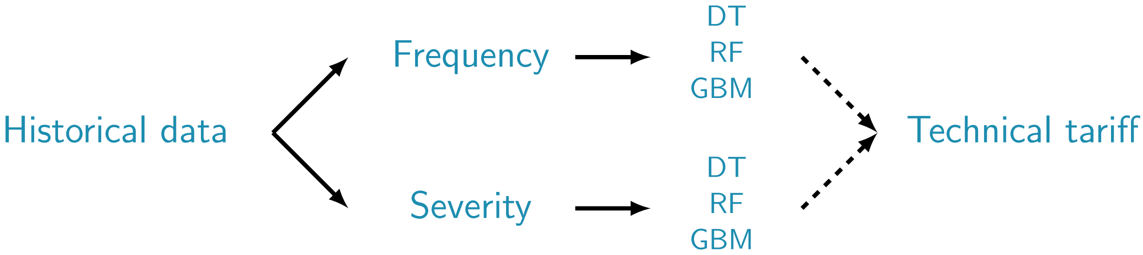
Historical data → Modelling tools → Technical tariff

Pricing in non-life insurance: frequency-severity modelling with GLMs



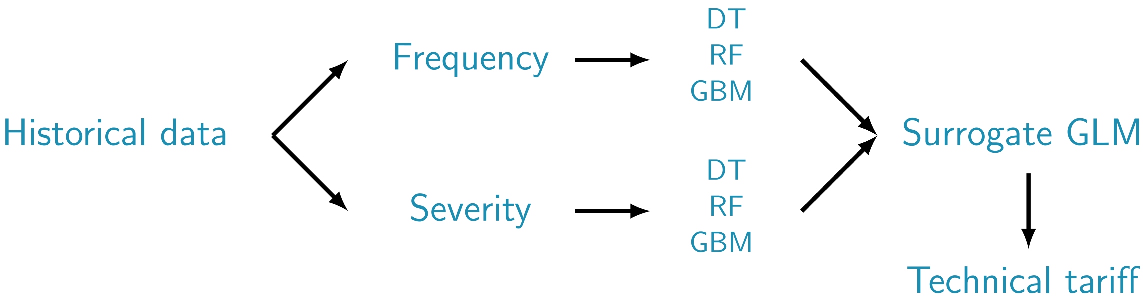
Henckaerts, R., Antonio, K., Clijsters, M. & Verbelen, R. (2018). *A data driven binning strategy for the construction of insurance tariff classes.* Scandinavian Actuarial Journal, 8, 681-705.

Pricing in non-life insurance: machine learning techniques



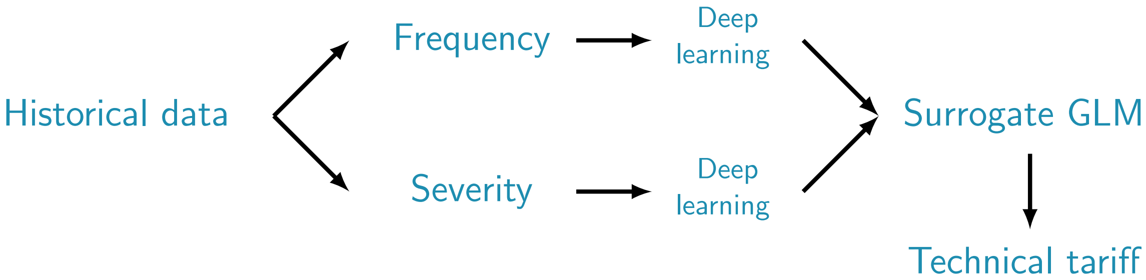
Henckaerts, R., Cote, M-P., Antonio, K. & Verbelen, R. (2021) *Boosting insights in insurance tariff plans with tree-based machine learning methods.* North American Actuarial Journal, 25, 255-285.

Pricing in non-life insurance: machine learning techniques and GLM as global surrogate model



Henckaerts, R., Antonio, K. & Cote, M-P. (2022). *When stakes are high: balancing accuracy and transparency with model-agnostic interpretable data-driven surrogates.* Expert Systems with Applications, 202, 117230.

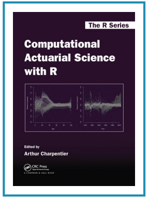
Contribution	Categorical treatment	Model architecture	# Data sets	Case study	Interpretational tools
Dugas et al. (2003)	–	LR, GLM, DT, NN, SVM	1	Tech. tariff	–
Yang et al. (2018)	–	TDBoost	1	Tweedie compound	PDP, VIP
Henckaerts et al. (2018)	DT binning	GLM	1	Freq, sev	–
Wüthrich (2019)	Dummy encoding, Embedding layers	GLM, NN, CANN	1	Freq	avg. neuron activation
Schelldorfer and Wüthrich (2019)	Embedding layers	CANN	1	Freq	–
Ferrario et al. (2020)	One-hot encoding	Boosted trees, NN	1	Freq	–
Henckaerts et al. (2021)	–	DT, RF, GBM	1	Freq, sev, tech. tariff	PDP, VIP, ICE
Delong and Kozak (2021)	Autoencoders	NN	1	Freq	–
Meng et al. (2022)	Convolutional autoencoder	GLM	1	Freq	–
Henckaerts et al. (2022)	–	GBM	6	Freq	PDP, SHAP, Surrogates



+ benchmarking study with multiple data sets
+ comparison of different embedding techniques
+ interpreting the results with a variety of interpretation tools

Used data sets

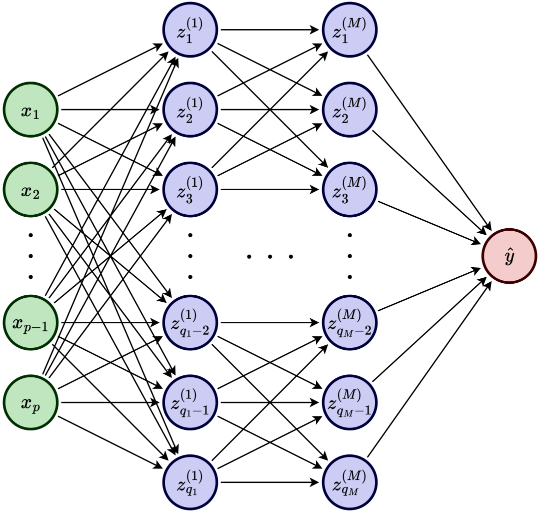
[github/henckr/maidrr] [Henckaerts et al., 2022] [Charpentier, 2015]



	Australian MTPL	Belgian MTPL	French MTPL	Norwegian MTPL
Number of observations				
Frequency	67 856	163 212	668 897	183 999
Severity	4 624	18 276	24 944	8 444
Covariates: number and type				
Continuous	1	4	2	0
Categorical	4	5	5	3
Spatial	0	1	2	1

Deep learning architecture

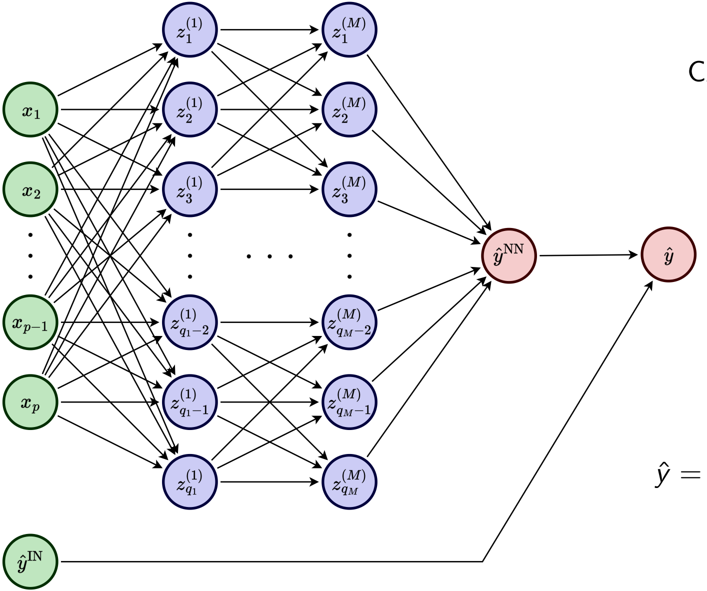
Feed-forward neural network



Feed-forward neural network characteristics:

- $\boldsymbol{x}$: numerical input
- $\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(M)}$: hidden layers
- $q_1, \ldots, q_M$: number of neurons for each hidden layer
- exponential activation in the output layer $\hat{y}$
- Poisson deviance for frequency modelling, gamma deviance for severity modelling

Combined actuarial neural network (CANN)



CANN characteristics:

- $\hat{y}^{IN}$: initial model input, i.e., a GLM or GBM
- $\hat{y}^{NN}$: neural network adjustment on the $\hat{y}^{IN}$
- fixed CANN:

  $$\hat{y} = \exp\left(\hat{y}^{\mathrm{NN}} + \ln\left(\hat{y}^{\mathrm{IN}}\right)\right)$$

- flexible CANN:

$$\hat{y} = \exp\left(w_{\mathrm{NN}} \cdot \hat{y}^{\mathrm{NN}} + w_{\mathrm{IN}} \cdot \ln\left(\hat{y}^{\mathrm{IN}}\right) + b\right)$$

Pre-processing steps

Continuous input variables:

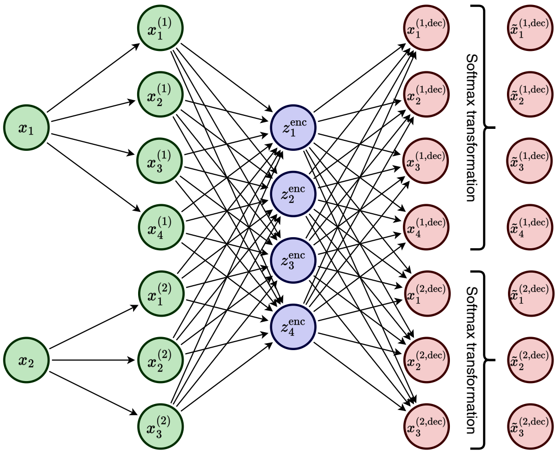
- we use normalization around zero
- for each continuous variable x_j we replace the value $x_{i,j}$ with

$$x_{i,j} \mapsto \tilde{x}_{i,j} = \frac{x_{i,j} - \mu_{x_j}}{\sigma_{x_j}}$$

where μ_{x_j} and σ_{x_j} are calculated on the training data

Spatial input variables

- AUS, FR, NOR, low number of levels: categorical
- BE, very high number of levels: continuous latitude & longitude



Autoencoder embedding

- create one-hot encoding for each categorical variable
- construct autoencoder with all one-hot encodings as input
- encoded layer $\boldsymbol{z}^{\text{enc}}$ of lower dimension than the input layer
- apply softmax transformation on the output layer
- train the autoencoder using cross-entropy loss function

Delong and Kozak (2023)

After the autoencoder is trained, we can calculate the embedding of all categorical variables as

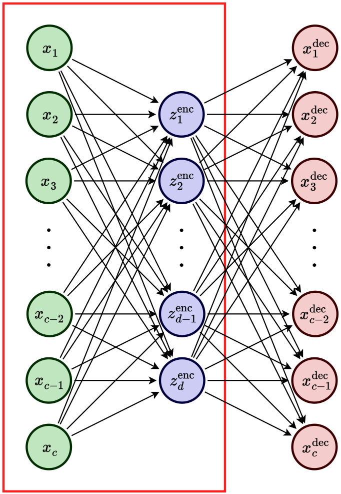
$$\boldsymbol{z}_i^{\text{enc}} = \sigma^{(\text{enc})}\left(W_{\text{enc}} \cdot \boldsymbol{x}_i + \boldsymbol{b}_{\text{enc}}\right).$$

The vector $\boldsymbol{z}_i^{\text{enc}}$ is a **compact**, **accurate** and **numerical** representation of $\boldsymbol{x}_i$.

Therefore we need to normalise the values in $\boldsymbol{z}^{\text{enc}}$ to be used in our FFNN and CANN models.

$$W_{\text{enc}} \mapsto \tilde{W}_{\text{enc}} = \begin{pmatrix} \frac{w_{11}}{\sigma_1} & \frac{w_{12}}{\sigma_1} & \cdots & \frac{w_{1c}}{\sigma_1} \\ \frac{w_{21}}{\sigma_2} & \frac{w_{12}}{\sigma_2} & \cdots & \frac{w_{1c}}{\sigma_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_{d-1,1}}{\sigma_{d-1}} & \frac{w_{d-1,2}}{\sigma_{d-1}} & \cdots & \frac{w_{d-1,c}}{\sigma_{d-1}} \\ \frac{w_{d1}}{\sigma_d} & \frac{w_{d2}}{\sigma_d} & \cdots & \frac{w_{dc}}{\sigma_d} \end{pmatrix}, \boldsymbol{b}_{\text{enc}} \mapsto \tilde{\boldsymbol{b}}_{\text{enc}} = \begin{pmatrix} \frac{b_1 - \mu_1}{\sigma_1} \\ \frac{b_2 - \mu_2}{\sigma_2} \\ \vdots \\ \frac{b_{d-1} - \mu_{d-1}}{\sigma_{d-1}} \\ \frac{b_d - \mu_d}{\sigma_d} \end{pmatrix}$$

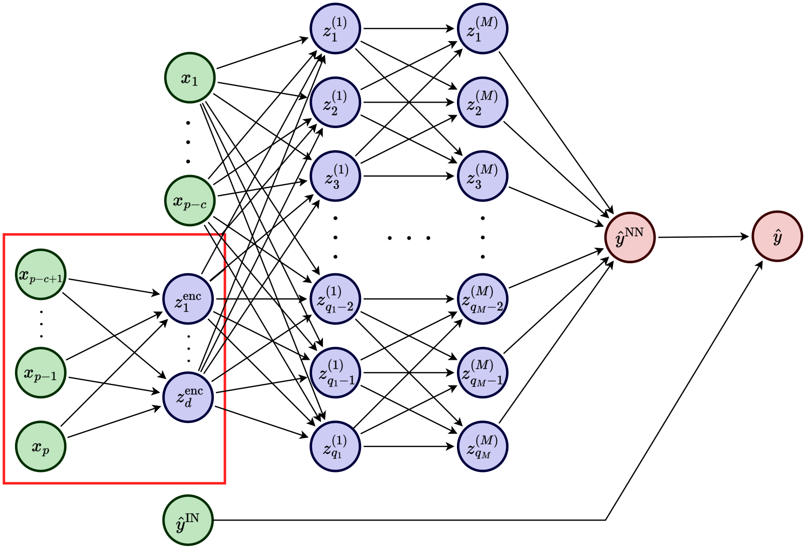
Categorical input variables



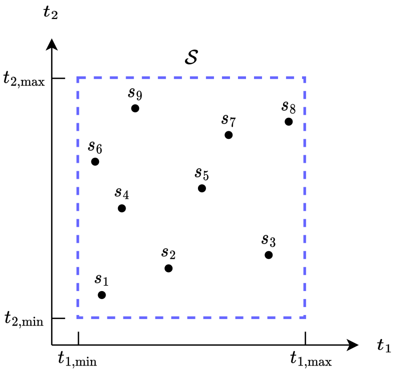
Advantages of autoencoder embedding:

- allows for cross-effects between variables in the embedding
- tunable autoencoded dimension
- unsupervised learning, so the autoencoder can be trained on the entire training set, and learned encoding can be used for both frequency and severity modelling.

Combining neural networks with autoencoder embedding



Out-of-sample performance



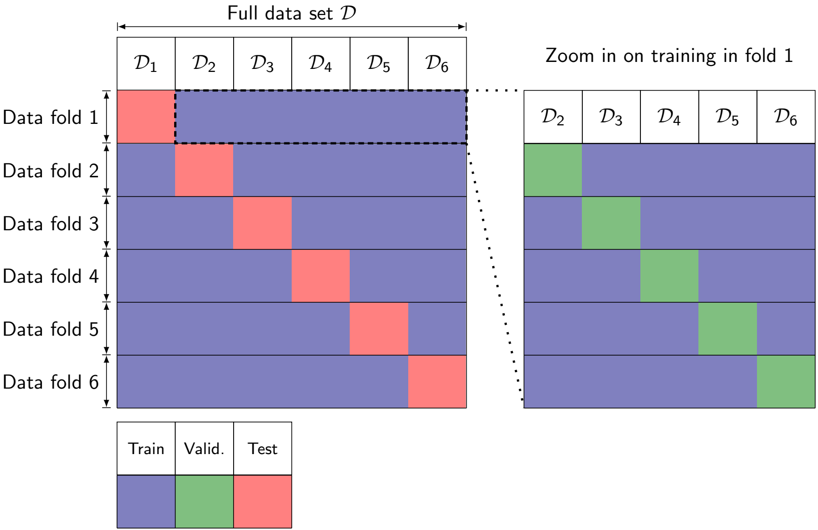
We use random grid search as tuning strategy (Bergstra and Bengio, 2012):

- for each tuning parameter t_k we define a range $[t_{k,\text{min}}, t_{k,\text{max}}]$
- the search space $\mathcal{S}$ is defined as

$$\mathcal{S} = [t_{1,\text{min}}, t_{1,\text{max}}] \times \ldots \times [t_{K,\text{min}}, t_{K,\text{max}}]$$

- we draw a random grid $\mathcal{R} \subset \mathcal{S}$ of candidate tuning parameters.

Cross-validation scheme



Henckaerts et al. (2020)

Out-of-sample performance: frequency

	Austalian data	Belgian data	French data	Norwegian data
GLM*	0.3816	0.5314	0.2762	0.2779
GBM*	0.3804	0.5295	0.2714	0.2778
Neural Network	0.3816	0.5319	0.2706	0.2799
CANN GLM fixed	0.3820	0.5307	0.2765	0.2778
CANN GLM flexible	0.3793	0.5283	0.2743	0.2779
CANN GBM fixed	0.3805	0.5295	0.2711	**0.2777**
CANN GBM flexible	**0.3782**	**0.5279**	**0.2695**	0.2778

- Measured in Poisson deviance, lowest deviance in bold
- All results averaged over three runs to avoid local minima solutions

*GLM as constructed in Henckaerts et al. (2018)
*GBM as constructed in Henckaerts et al. (2020)

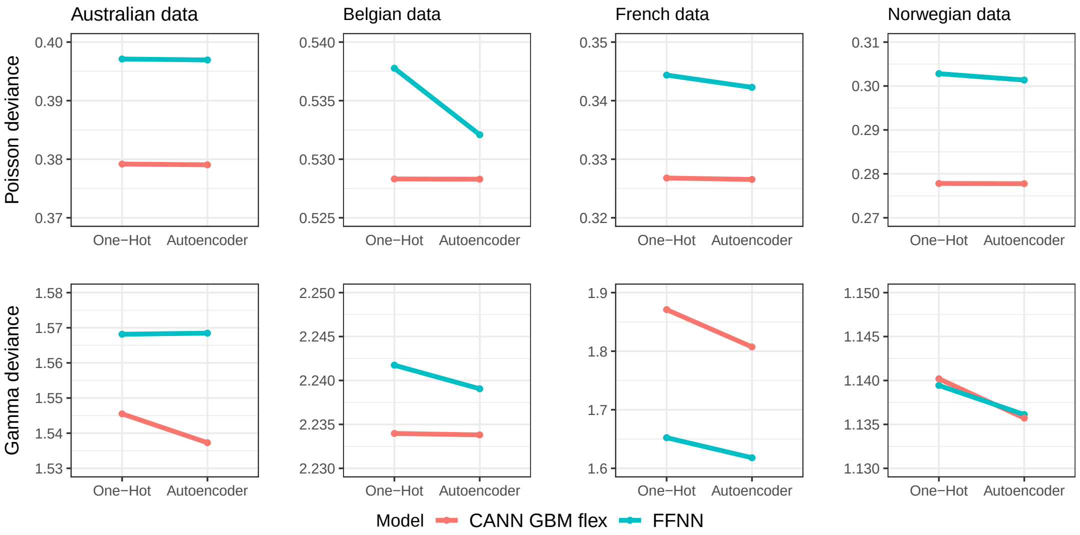
Out-of-sample performance: severity

	Austalian data	Belgian data	French data	Norwegian data
GLM*	1.5562	**2.2280**	1.7093	1.1355
GBM*	1.5359	2.2365	1.6471	1.1370
Neural Network	1.5752	2.2436	**1.6104**	**1.1353**
CANN GLM fixed	1.5414	2.2284	1.7132	1.1373
CANN GLM flexible	1.5508	2.2284	1.7124	1.1358
CANN GBM fixed	**1.5357**	2.2364	1.6472	1.1374
CANN GBM flexible	1.5395	2.2365	1.7153	1.1378

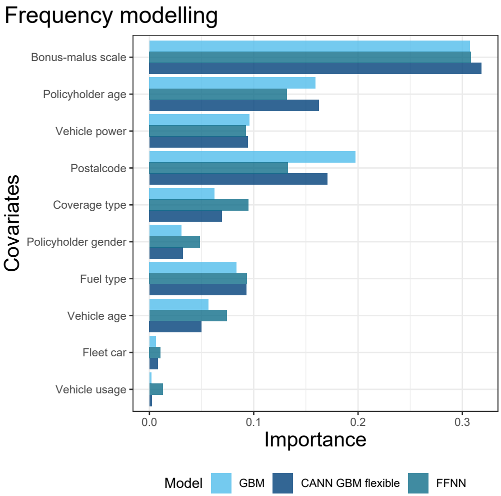
- Measured in gamma deviance, lowest deviance in bold
- All results averaged over three runs to avoid local minima solutions

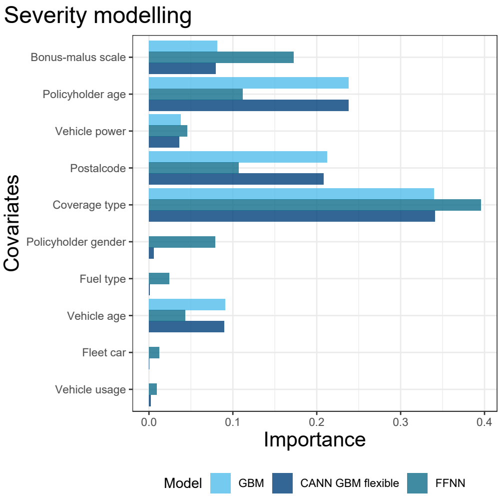
*GLM as constructed in Henckaerts et al. (2018)
*GBM as constructed in Henckaerts et al. (2020)

Effect of autoencoder embedding



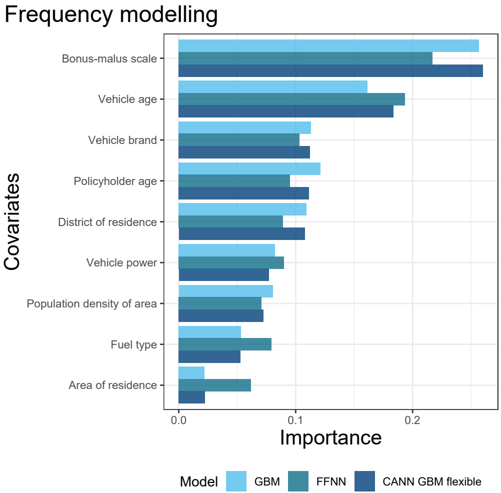
Interpretation tools

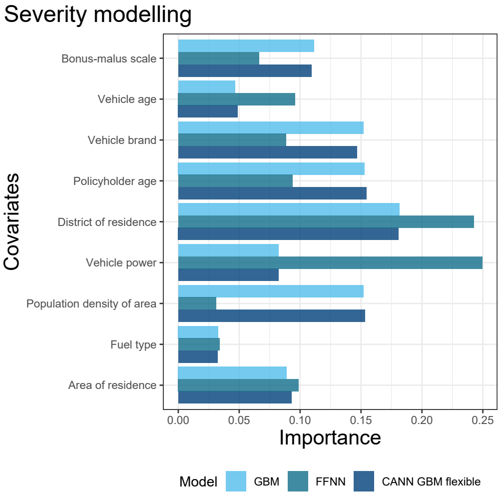




Results for the Belgian MTPL data set.

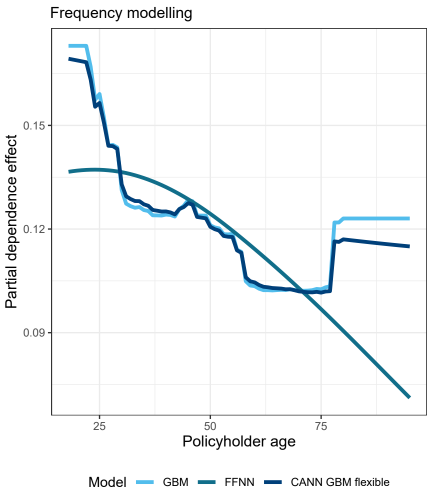
Variable importance

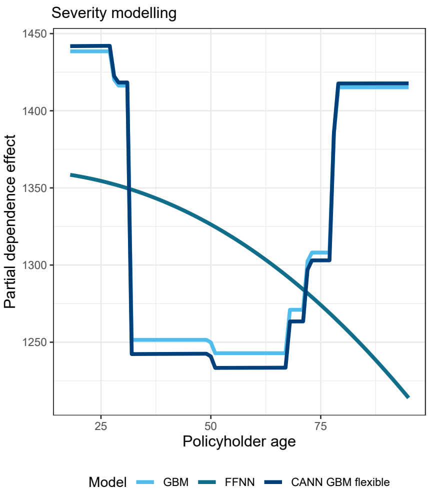




Results for the French MTPL data set.

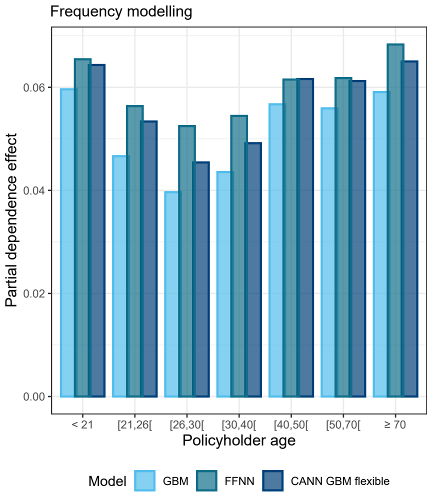
Partial dependency plots

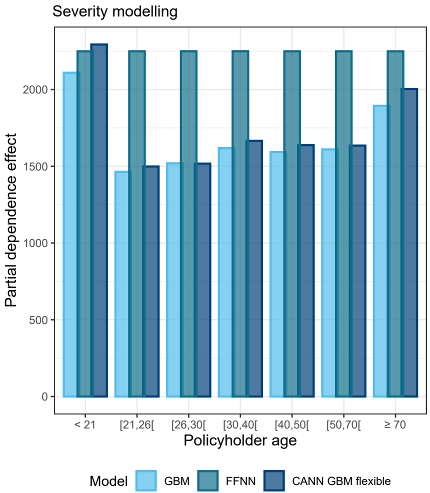




Results for the Belgian MTPL data set.

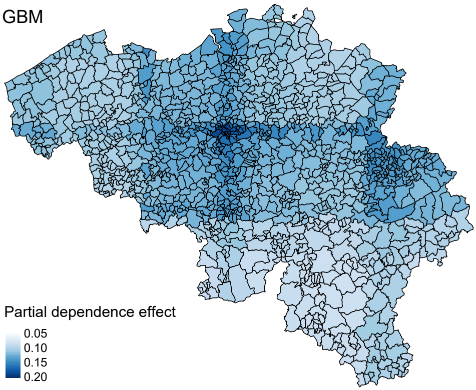
Partial dependency plots





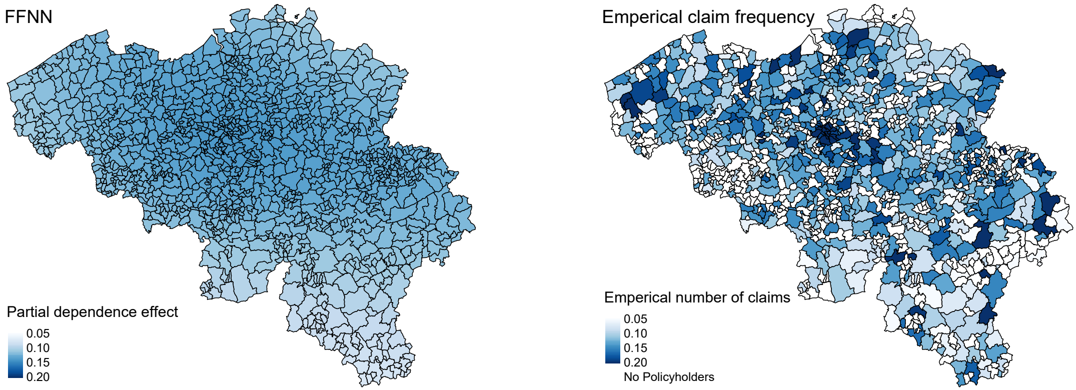
Results for the French MTPL data set.

Partial dependency plots



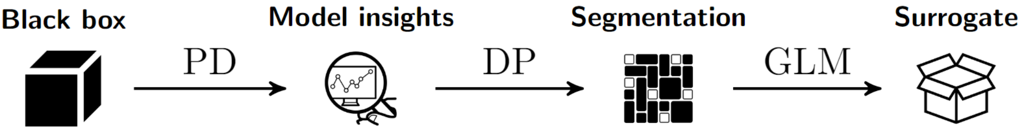
Results for the Belgian MTPL data set.

Partial dependency plots



Results for the Belgian MTPL data set.

Premium structure



Henckaerts, R., Antonio, K., & Cote, M-P. (2022). *When stakes are high: balancing accuracy and transparency with model-agnostic interpretable data-driven surrogates.* Expert Systems with Applications, 202, 117230.

Surrogate models

Covariates → Partial dependency effects → Segmentation

Covariates

Bonus-malus score
50
51
52
53
54
55
⋮
168
169
170
171
172
173

Vehicle power
4
5
6
7
8
9
10
11
12
13
14
15

Region
Rhone-Alpes
Picardie
Aquitaine
⋮
Franche-Comte
Limousin
Haute-Normandie

Partial dependency effects

Bonus-malus score
0.0445
0.0431
0.0547
0.0550
0.0489
0.0729
⋮
0.1910
0.1912
0.1918
0.1919
0.1926
0.1941

Vehicle power
0.0515
0.0589
0.0623
0.0576
0.0492
0.0684
0.0634
0.0620
0.0587
0.0605
0.0600
0.0581

Region
0.0595
0.0610
0.0507
⋮
0.0491
0.0566
0.0524

Segmentation

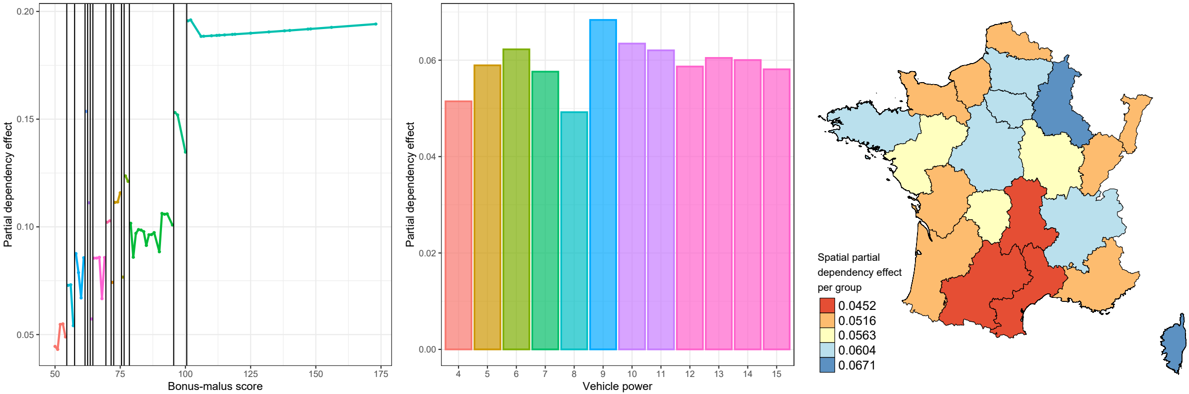
Bonus-malus score
$[50, 54]$
$[55, 57]$
$[58, 61]$
62
63
64
⋮
$[73, 75]$
76
$[77, 78]$
$[79, 95]$
$[96, 100]$
$[101, 173]$

Vehicle power
4
5
6
7
8
9
$[10, 11]$
$[12, 15]$

Region
{Zone 1}
{Zone 2}
{Zone 3}
{Zone 4}
{Zone 5}

Results for French MTPL data set; CANN GBM flexible model.

Surrogate models



Results for French MTPL data set; CANN GBM flexible model.

Surrogate models

Surrogate GLM fitted on the segmented data

Vehicle power	Bonus-malus score	Policyholder age	$\cdots$	Region	Frequency surrogate
1	64	$[30, 40)$	$\cdots$	{Champagne-Ardenne, Corse}	0.0099
3	76	$[21, 26)$	$\cdots$	{Bourgogne, Limousin, Pays-de-la-Loire}	0.0857
4	$[79, 95]$	$[40, 70)$	$\cdots$	{Bretagne, Centre, Ile-de-France, Picardie, Rhone-Alpes}	0.1709

Out-of-sample performance comparison

	Austalian data	Belgian data	French data
Binned GLM*	0.3817	0.5314	0.2761
Surrogate GLM	0.3805	0.5308	0.2738

*GLM as constructed in Henckaerts et al. (2018)

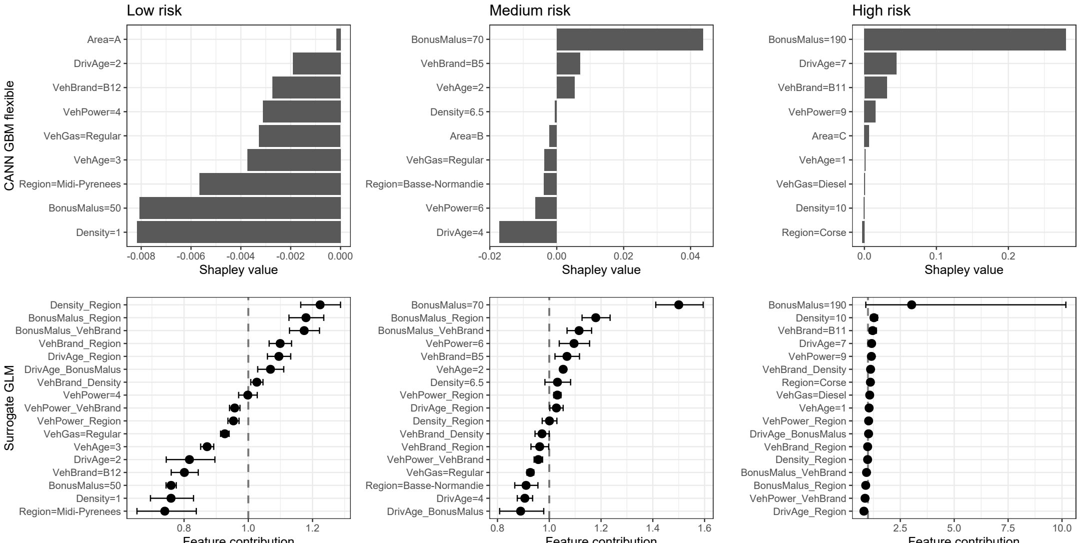
Risk profile comparison

Variables	Low risk	Medium risk	High risk
Vehicle power	4	6	9
Vehicle age	3	2	1
Policyholder age	$[21, 26[$	$[30, 40[$	≥ 70
Bonus-malus scale	50	70	190
Vehicle brand	B12	B5	B11
Fuel type	Regular	Regular	Diesel
Population density of area	2.71	665.14	22 026.47
District of residence	Midi-Pyrenees	Basse-Normandie	Corse
Predicted number of claims			
Surrogate GLM	0.020	0.106	0.361
CANN GBM flexible	0.021	0.101	0.519

Results for French MTPL data set.

Surrogate models

Risk profile comparison



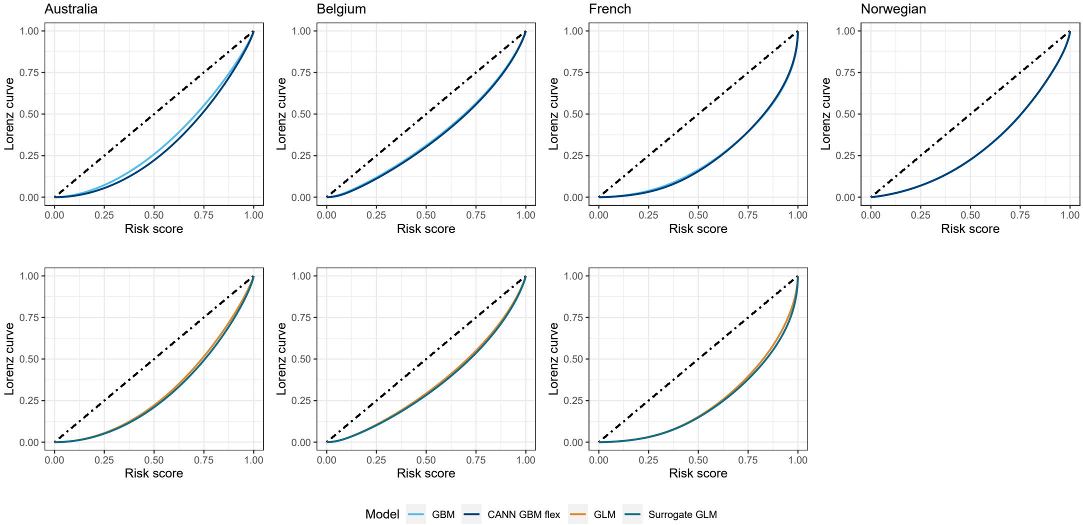
Managerial insights

Portfolio comparison on predicted losses versus observed losses

	Observed	GLM	GBM	CANN GBM flex	Surrogate GLM
Observed and predicted losses					
Australia (AU$)	9 314 604	9 345 113	9 136 324	9 154 467	9 355 718
Belgium (€)	26 464 970	26 399 027	26 079 709	25 720 143	26 345 969
France (€)	58 872 147	56 053 341	56 207 993	58 629 584	57 048 375
Norway (NOK)	206 649 080	206 634 401	206 475 980	206 494 683	-
Ratio of predicted losses over observed losses					
Australia	-	1.00	0.98	0.98	1.00
Belgium	-	1.00	0.99	0.97	1.00
France	-	0.95	0.95	1.00	0.97
Norway	-	1.00	1.00	1.00	-

Managerial insights

Risk classification comparison using Lorenz curve



Conclusions

- Our deep learning structures have a higher performance on multiple data sets, in both frequency and severity modelling
- With interpretation tools we can get insights into the deep learning models
- We can construct interpretable and easy to use surrogate GLMs, based on the insights of the deep learning models, including the predictive power of the deep learning models

Bergstra, J. & Bengio, Y. (2012). *Random search for hyper-parameter optimization.* Journal of Machine Learning Research, 13, 281-305.

Delong, Ł. & Kozak, A. (2023). *The use of autoencoders for training neural networks with mixed categorical and numerical features.* ASTIN Bulletin: The Journal of the IAA, 1–20.

Dugas, C., Bengio, Y., Chapados, N., Vincent, P., Denoncourt, G. & Fournier, C. (2003). *Statistical learning algorithms applied to automobile insurance ratemaking.* Intelligent and Other Computational Techniques in Insurance, 137-197.

Garavaglia, S. & Sharma, A. (1998). *A smart guide to dummy variables: four applications and a macro.* Proceedings of the northeast SAS users group conference, 43.

Guo, C. & Berkhahn, F. (2016). *Entity embeddings of categorical variables.* arXiv preprint arXiv:1604.06737.

Henckaerts, R., Antonio, K., Clijsters, M. & Verbelen, R. (2018). *A data driven binning strategy for the construction of insurance tariff classes.* Scandinavian Actuarial Journal, 8, 681-705.

Henckaerts, R., Cote, M-P., Antonio, K. & Verbelen, R. (2021). *Boosting insights in insurance tariff plans with tree-based machine learning methods.* North American Actuarial Journal, 25, 255-285.

Henckaerts, R., Antonio, K. & Cote, M-P. (2022). *When stakes are high: Balancing accuracy and transparency with model-agnostic interpretable data-driven surrogates.* Expert Systems with Applications, 202, 117230.

Meng, S., Wang, H., Shi, Y. & Gao, G. (2022). *Improving automobile insurance claims frequency prediction with telematics car driving data.* ASTIN Bulletin: The Journal of the IAA, 52, 2, 363–391.

Schelldorfer, J. & and Wuthrich, M. (2019). *Nesting classical actuarial models into neural networks.* URL www.ssrn.com/abstract=3320525.

Wuthrich, M. (2019). *From generalized linear models to neural networks, and back.* URL www.ssrn.com/abstract=3491790.

Yang, Y., Qian, W. & Zou, H. (2018). *Insurance premium prediction via gradient tree-boosted Tweedie compound Poisson models.* Journal of Business & Economic Statistics, 36, 3, 456-470.

Thank you for your attention!

Paper on Arxiv

Code via Github