KØBENHAVNS UNIVERSITET

Detection of interacting variables for generalized linear models using neural networks

Yevhen Havrylenko[1,2], Julia Heger[2,3]

[1]Department of Mathematical Sciences, University of Copenhagen, Denmark (current)

[2]TUM School of Computation, Information and Technology, Technical University of Munich, Germany (former)

[3]Chair of Analytics and Optimization, University of Augsburg, Germany (current)

DAV/DGVFM Jahrestagung

Berlin, 26 April 2024

DAV/DGVFM
Jahrestagung
2024

Presentation overview

Motivation and context

- Actuaries have huge data sets, especially in motor third party liability (MTPL) insurance
- Insurers use generalized linear models (GLMs) due to interpretability of these models and companies' IT legacy
- GLMs are improved by actuaries via **sophisticated choice** of significant variables and their **interactions**
- Search for strong interactions is more **time-consuming**, is mostly **visual** and depends much on **expert judgement**
- Example: for 20 variables $\approx$ 200 pairwise interactions, for 50 variables $\approx$ 1200 pairwise interactions
- A **recommendation engine** for the next-best interaction missing in a GLM **may save actuaries hours/days**
- Why next-best? GLMs for tariffs cannot be drastically changed/replaced but should be adjusted gradually
- For automatic tree-based construction of a GLM from scratch, see, e.g., Henckaerts et al. (2022)

Problem setting and business requirements

- **Given**:
  - the predictions of a benchmark GLM (e.g., claim counts)
  - training data (e.g., driver's age, profession, car brand, postcode)
- **Find**:
  - Next-best pairwise interaction missing in the benchmark GLM (e.g., interaction between postcode and bonus malus)
- **Subject to** business requirements:
  1. Avoid retraining the benchmark GLM, use only its predictions
  2. "Next-best" means in terms of key performance indicators (KPIs) that actuaries rely on, e.g., lift plots
  3. Minimize the need for visual evaluation of KPIs

Presentation overview

Notation

- $\mathcal{D} = \{(N_i, \boldsymbol{x}_i, v_i)\}_{i=1}^n$ – data set, where
  - $n \in \mathbb{N}$ is the number of observations
  - $v_i \in [0, 1]$ corresponds to the exposure time in years for the i-th observation
  - $N_i \in \mathbb{N} \cup \{0\}$ is the number of claims observed for the i-th observation within exposure time v_i
  - $\boldsymbol{x}_i \in \mathcal{X} \subset \{1\} \times \mathbb{R}^p$ represents the vector of covariates for the i-th observation excluding v_i
  - $p \in \mathbb{N}$ is the number of covariates
- $x_{\cdot,j}$ denotes covariate $j = 1, \ldots, p$

Poisson GLM

- Poisson GLM with the canonical link assumes

$$N_i \sim \text{Poisson}(v_i \exp(\eta(\boldsymbol{\beta}, \boldsymbol{x}_i))),$$

  where $\eta(\boldsymbol{\beta}, \boldsymbol{x}_i) = \boldsymbol{\beta}^T \boldsymbol{x}_i$ – the linear component and β is the vector of the GLM parameters

- Denote by $\hat{\lambda}_i^{GLM} := \exp(\eta(\hat{\boldsymbol{\beta}}, \boldsymbol{x}_i))$ the annualized claim frequency predicted by a GLM

- An interaction is a term $I(x_{i,1}, x_{i,2})$ added to the component $\eta(\boldsymbol{\beta}, \boldsymbol{x}_i)$ such that $I(x_{i,1}, x_{i,2})$ is not additively separable, e.g.:

  – for numerical $x_{\cdot,1}$, $x_{\cdot,2}$ we can have:

$$I(x_{i,1}, x_{i,2}) = \beta_{1,2} \cdot x_{i,1} \cdot x_{i,2}$$

  – for numerical $x_{\cdot,1}$ and categorical $x_{\cdot,2}$ with $J \in \mathbb{N}$ categories and J as a reference category:

$$I(x_{i,1}, x_{i,2}) = \sum_{j=1}^{J-1} \beta_j \cdot x_{i,1} \cdot \mathbb{1}_{\{x_{i,2}=j\}}$$

  – for categorical $x_{\cdot,1}$ with $J \in \mathbb{N}$ categories and categorical $x_{\cdot,2}$ with $K \in \mathbb{N}$ categories:

$$I(x_{i,1}, x_{i,2}) = \sum_{j=1}^{J-1} \sum_{k=1}^{K-1} \beta_{j,k} \cdot \mathbb{1}_{\{x_{i,1}=j\}} \cdot \mathbb{1}_{\{x_{i,2}=k\}}$$

Presentation overview

Popular existing approaches and our contribution

- Approach known from practitioners:
  - train a Gradient Boosting Machine (GBM) with trees of depth 2 and benchmark GLM predictions as offset
  - compute Friedman's H-statistic (Friedman and Popescu (2008)) for each pair of features

Popular existing approaches and our contribution

- Approach known from practitioners:
  - train a Gradient Boosting Machine (GBM) with trees of depth 2 and benchmark GLM predictions as offset
  - compute Friedman's H-statistic (Friedman and Popescu (2008)) for each pair of features
- Approach in Wüthrich (2020):
  - train a Combined Actuarial Neural Network (CANN) for each pair of features
  - for each CANN, compute the decrease of loss function in comparison to benchmark GLM

Popular existing approaches and our contribution

- Approach known from practitioners:
  - train a Gradient Boosting Machine (GBM) with trees of depth 2 and benchmark GLM predictions as offset
  - compute Friedman's H-statistic (Friedman and Popescu (2008)) for each pair of features
- Approach in Wüthrich (2020):
  - train a Combined Actuarial Neural Network (CANN) for each pair of features
  - for each CANN, compute the decrease of loss function in comparison to benchmark GLM
- Our contribution:
  1. our approach is **computationally faster**, especially for data sets with many variables
  2. we address the question of automating the "best" **functional form of the interaction**

Overview of our approach

- Algorithm:

  1 **Train Combined Actuarial Neural Network** (CANN) using all variables

  2 **Calculate strength** of all pairwise interactions **via Neural Interaction Detection** (NID), sort from strongest to weakest

  3 **Train mini-GLMs** for top ranked interaction(s), **identify the best** mini-GLM, **recommend** the corresponding interaction

Overview of our approach

- Algorithm:
  1. **Train Combined Actuarial Neural Network** (CANN) using all variables
  2. **Calculate strength** of all pairwise interactions **via Neural Interaction Detection** (NID), sort from strongest to weakest
  3. **Train mini-GLMs** for top ranked interaction(s), **identify the best** mini-GLM, **recommend** the corresponding interaction
- Motivation for each component:
  1. **CANN** captures well **non-linear interactions** missing in an actuarial model and allows for **embedding layers**
  2. **NID** is **fast**, does not rely on partial dependence plots or data reshuffling, **easy to implement**
  3. **mini-GLMs** help identify **next-best** interaction **among top-ranked interations** and its **“optimal” functional form**

Algorithm Step 1: CANN

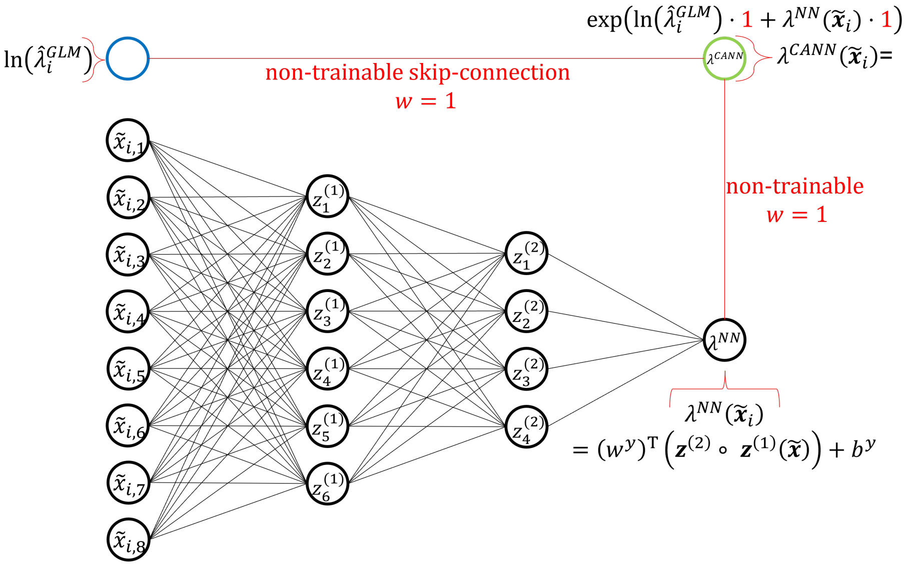

Figure: Figure taken from Havrylenko and Heger (2023)

- CANN was proposed in Wüthrich and Merz (2019)
- Two parts: GLM (non-trainable), NN (trainable)
- $W^{(l)}$ weight matrix, $\boldsymbol{b}^{(l)}$ bias vector for hidden layer $l = 1, \ldots, d$, where $d \in \mathbb{N}$ is number of hidden layers
- $\boldsymbol{w}^y$ coefficient vector, b^y bias for output neuron
- $\phi_l(\cdot)$ activation function of neurons in layer l, $\phi_{d+1}(z) = z$
- Vector of activation values in hidden layers (HLs):

$$\boldsymbol{z}^{(l)} = \vec{\phi_l}\left(W^{(l)}\boldsymbol{z}^{(l-1)} + \boldsymbol{b}^{(l)}\right), \quad l = 1, \ldots, d,$$

  $q_l \in \mathbb{N}$, $\boldsymbol{z}^{(0)} := \tilde{\boldsymbol{x}}$ input features
- Assumptions of a Poisson CANN:
  - $N_i \sim \text{Poisson}(v_i \cdot \lambda^{\text{CANN}}(\tilde{\boldsymbol{x}}_i))$
  - $\boldsymbol{w}^y = (0, 0, \ldots, 0)^\top \in \mathbb{R}^{q_d}$, $b^y = 0$ at initialization

Algorithm Step 2: NID

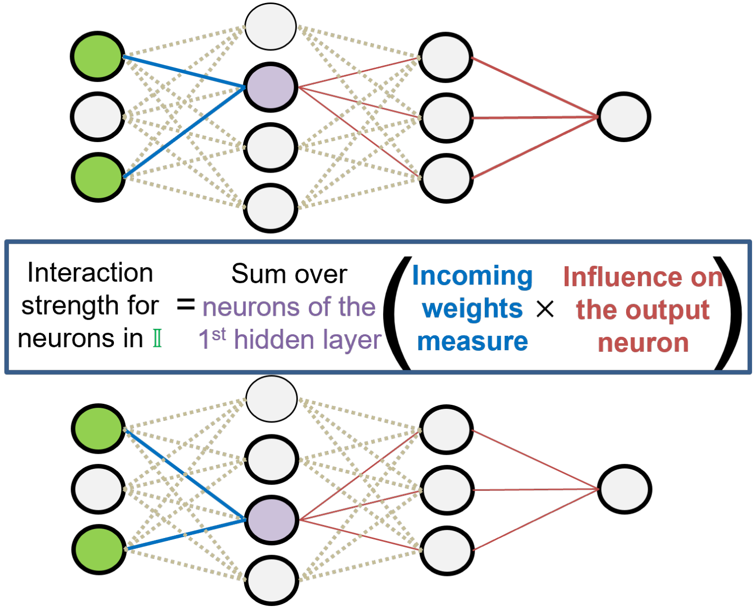

Figure: Figure taken from Havrylenko and Heger (2023)

- NID proposed in Tsang et al. (2018) for NN and num. ftrs.
- We modify NID to CANN and categorical variables
- Assumption: interactions are captured in the 1-st HL
- Interaction strength between input neurons in $\mathbb{I}$ measured at j-th neuron in 1-st HL:

$$s_j(\mathbb{I}) = \min(|W^{(1)}_{j,\mathbb{I}}|) \cdot \zeta^{(1)}_j, \quad s_j(\mathbb{I}) \in \mathbb{R}$$

- $|W^{(1)}_{j,\mathbb{I}}|$ absolute value of incoming weights from input neurons in $\mathbb{I}$ to j-th neuron in 1-st HL
- Influence of j-th neuron in 1-st HL is j-th element of

$$\zeta^{(1)} = |\boldsymbol{w}^y|^\top \cdot |W^{(d)}| \cdot |W^{(d-1)}| \cdot \ldots \cdot |W^{(2)}|, \; \zeta^{(1)} \in \mathbb{R}^{q_1}$$

- Total interaction strength score for input neurons in $\mathbb{I}$

$$s(\mathbb{I}) = \sum_{j=1}^{q_1} s_j(\mathbb{I}) = \sum_{j=1}^{q_1} \min(|W^{(1)}_{j,\mathbb{I}}|) \cdot \zeta^{(1)}_j$$

Algorithm Step 3: mini-GLMs and recommendation

- One cannot blindly recommend the interaction first-ranked by NID:

  1. a categorical variable in the interaction may require clustering of its categories (e.g., postcode, car brand)
  2. another top-ranked interaction may have a very similar NID score
  3. the functional form of the interaction for a GLM is not known in general

- Point 1: Cluster num. representations of cat. variables that appear in top-ranked interaction(s) and have many categories

- Points 2 and 3 are dealt as follows:

  1. Fit "mini-GLMs" for top-ranked pair(s) of interacting variables with different $I(x_{\cdot,j}, x_{\cdot,k})$, bench. GLM prediction is an offset:

$$N_i \sim \text{Poisson}\left(v_i \cdot \hat{\lambda}_i^{GLM} \cdot \exp(I(x_{i,j}, x_{i,k}))\right)$$

  2. Recommend the interaction that corresponds to mini-GLM with the best performance on relevant KPIs

Presentation overview

KØBENHAVNS UNIVERSITET

Case study 1: artificial data and algorithm step 1

- Artificially generated $\mathcal{D}$ with $(x_{i,1}, \dots, x_{i,8}) \sim N(0, \Sigma)$, where Σ is identity matrix except $\Sigma_{2,8} = \Sigma_{8,2} = 0.5$, $x_{i,9} \sim \mathit{Binomial}(2, 0.3)$, $x_{i,10} \sim \mathit{Binomial}(5, 0.2)$, $v_i = 1$:

$$\begin{aligned} \boldsymbol{x}_i \in \mathbb{R}^{10} \mapsto \eta(\boldsymbol{x}_i) = & -3 + 0.5 \cdot x_{i,1} - 0.25 \cdot x_{i,2}^2 + 0.5 \cdot |x_{i,3}| \cdot \sin(2 \cdot x_{i,3}) + 0.5 \cdot x_{i,4} \cdot x_{i,5} \\ & + 0.125 \cdot x_{i,5}^2 \cdot x_{i,6} - 0.1 \cdot 1_{\{x_{i,9}=1\}} - 0.2 \cdot 1_{\{x_{i,9}=2\}} + 0.1 \cdot 1_{\{x_{i,10}=1\}} \\ & + 0.2 \cdot 1_{\{x_{i,10}=2\}} + 0.3 \cdot 1_{\{x_{i,10}=3\}} + 0.4 \cdot 1_{\{x_{i,10}=4\}} + 0.5 \cdot 1_{\{x_{i,10}=5\}}. \end{aligned}$$

$$N_i \sim \text{Poisson}(\exp(\eta(\boldsymbol{x}_i))), \quad i = 1, \dots, 2 \cdot 10^6$$

- $\eta(\boldsymbol{x}_i)$ is taken from Richman and Wüthrich (2023) and modified such that the portfolio distribution looks realistic
- $\lambda^{GLM}(\boldsymbol{x}_i)$ with all main effects, but no interactions
- In step 1, a CANN with 3 HLs is trained using:
  - pre-processed variables (min-max scaling for numerical, one-hot encoding for $x_{\cdot,9}$ and 2-dim. embedding for $x_{\cdot,10}$)
  - fine-tuning of activation functions and the number of neurons in each HL

Case study 1: algorithm step 2

Interaction rank	Feature 1 name	Feature 2 name	Interaction strength score
1	$x_{\cdot,4}$	$x_{\cdot,5}$	70.0263
2	$x_{\cdot,5}$	$x_{\cdot,6}$	37.3492
3	$x_{\cdot,4}$	$x_{\cdot,6}$	34.7608
4	$x_{\cdot,5}$	$x_{\cdot,10}$	24.3280
5	$x_{\cdot,4}$	$x_{\cdot,10}$	23.9654

CANN + NID results

computation time:
170 sec. for CANN +
1.2 sec. for NID

Interaction rank	Feature 1 name	Feature 2 name	Interaction strength score
1	$x_{\cdot,4}$	$x_{\cdot,5}$	0.8495
2	$x_{\cdot,5}$	$x_{\cdot,6}$	0.2223
3	$x_{\cdot,3}$	$x_{\cdot,5}$	0.0156
4	$x_{\cdot,3}$	$x_{\cdot,6}$	0.0055
5	$x_{\cdot,3}$	$x_{\cdot,4}$	0.0001

GBM + H-statistic results

computation time:
120 sec. for GBM +
40 sec. for H-statistic on 0.5% of data
(350 sec. for H-statistic on 5% of data)

Case study 1: algorithm step 3

- Since the functional form of $I(x_{\cdot,4}, x_{\cdot,5})$ is not known for non-categorical variables, one can:
  - categorize $x_{\cdot,4} \to x^c_{\cdot,4}$ as well as $x_{\cdot,5} \to x^c_{\cdot,5}$ and recommend $I(x^c_{\cdot,4}, x^c_{\cdot,5}) = \sum_{j=1}^{J-1} \sum_{k=1}^{K-1} \beta_{j,k} \cdot \mathbb{1}_{\{x^c_{i,4}=j\}} \cdot \mathbb{1}_{\{x^c_{i,5}=k\}}$ or
  - train mini-GLMs with various “reasonable” $I(x_{\cdot,4}, x_{\cdot,5})$ (from a selection of elementary functions or inferred from data visualization) and identify the best-performing one
- In our example, mini-GLM with $I(x_{\cdot,4}, x_{\cdot,5}) = \beta_{4,5} \cdot x_{\cdot,5} \cdot x_{\cdot,5}$ has the best KPIs, so this $I(\cdot, \cdot)$ is recommended for bench. GLM
- When $I(x_{\cdot,4}, x_{\cdot,5}) = \beta_{4,5} \cdot x_{\cdot,4} \cdot x_{\cdot,5}$ is added to the benchmark GLM, out-of-sample Poisson deviance $\downarrow$ from 0.3314 to 0.3134
- Repeating steps 1 & 2 for updated GLM ranks $(x_{\cdot,5}, x_{\cdot,6})$ as (by far) 1-st ranked
- In step 3, mini-GLMs with $I(x_{\cdot,5}, x_{\cdot,6}) = \beta_{5,6} \cdot x^a_{\cdot,5} \cdot x^b_{\cdot,6}$ for $a \in \{1, 2, 3\}$ and $b \in \{1, 2, 3\}$ are trained and KPIs are evaluated
- $I(x_{\cdot,5}, x_{\cdot,6}) = \beta_{5,6} \cdot x^2_{\cdot,5} \cdot x_{\cdot,6}$ is recommended for the benchmark GLM

Case studies with open-source data and big proprietary data

- *freMTPL2freq* data set with $\approx$ 680000 observations, 9 variables, so 36 potential pairwise interactions:
  - Alg. step 1 takes 90 sec. for 1 CANN (comparable for 1 GBM)
  - Alg. step 2 takes $<$ 1 sec., whereas the computation of Friedman's H-statistics takes 5 minutes
  - Alg. step 3 recommends $I(VehAge, BonusMalus)$ or $I(VehAge, VehGas)$, depending on KPI for mini-GLM
  - Both interactions are among top ones as per GBM + Friedman's H-statistic
  - Schelldorfer and Wüthrich (2019) does not have $I(VehAge, VehGas)$ in the list of detected interactions

Case studies with open-source data and big proprietary data

- *freMTPL2freq* data set with $\approx$ 680000 observations, 9 variables, so 36 potential pairwise interactions:
  - Alg. step 1 takes 90 sec. for 1 CANN (comparable for 1 GBM)
  - Alg. step 2 takes $<$ 1 sec., whereas the computation of Friedman's H-statistics takes 5 minutes
  - Alg. step 3 recommends $I(VehAge, BonusMalus)$ or $I(VehAge, VehGas)$, depending on KPI for mini-GLM
  - Both interactions are among top ones as per GBM + Friedman's H-statistic
  - Schelldorfer and Wüthrich (2019) does not have $I(VehAge, VehGas)$ in the list of detected interactions
- Confidential data with $\approx$ 11 mln. observations, $\approx$ 50 variables, so $\approx$ 1225 potential pairwise interactions:
  - Alg. step 2 takes $<$ 3 sec., whereas calculating Friedman's H-statistics is too costly even for 5% of data
  - Dimensionality reduction of large cat. variables like *postcode* or *carBrand* is also beneficial for actuaries

Presentation overview

Summary and conclusions

- Detection of interactions missing in GLMs can be very time-consuming
- We contribute to the academic literature on **the detection of interacting variables for GLMs by**
  - proposing an interaction-detection **methodology that is signif. faster** than alernatives and has a comparable quality
  - pioneering the usage of **NID in actuarial science**
  - **modifying NID to CANNs** and **categorical variables**
- Advantages of proposed methodology:
  - **almost fully automatable** with little to no need for actuarial intervention
  - **faster** than other approaches $\Rightarrow$ especially suitable for big data
  - represents large **categorical variables** as **low-dimensional num. vectors**
- Research outlook: analyze robustness of the algorithm, improve automation of Step 3

Acknowledgments

Both authors acknowledge the financial support of the ERGO Center of Excellence in Insurance at the Technical University of Munich for conducting this research project.

Both authors thank Kay Adam (ERGO) for project initiation and feedback, Noel Stein and Samarth Mehrotra for their input at project beginning.

Yevhen Havrylenko thanks Kay Adam (ERGO) and Frank Ellgring (ERGO) for the opportunity to gain insights into actuarial pricing during a 3-month project at Global P&C Pricing Department of ERGO in 2022.

References I

Friedman, J. H. and B. E. Popescu (2008). Predictive learning via rule ensembles. *The Annals of Applied Statistics 2*(3), 916–954.

Havrylenko, Y. and J. Heger (2023). Detection of interacting variables for generalized linear models via neural networks. *European Actuarial Journal*.

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

Richman, R. and M. V. Wüthrich (2023). LocalGLMnet: interpretable deep learning for tabular data. *Scandinavian Actuarial Journal 2023*(1), 71–95.

Schelldorfer, J. and M. V. Wüthrich (2019). Nesting classical actuarial models into neural networks. *Available at SSRN 3320525*.

Tsang, M., D. Cheng, and Y. Liu (2018). Detecting statistical interactions from neural network weights. *ICLR 2018*.

Wüthrich, M. V. (2020). From generalized linear models to neural networks, and back. *Available at SSRN 3491790*.

Wüthrich, M. V. and M. Merz (2019). Editorial: Yes, we CANN! *ASTIN Bulletin 49*(1).

I thank you for your attention...

... and look forward to your questions!

(a) QR code to Havrylenko and Heger (2023)

(b) QR code to https://war.ukraine.ua/support-ukraine/

NID vs Friedman’s H-statistic

NID	**Friedman’s H-statistic**
• Model-specific, only applicable for feed-forward NNs	• Model-agnostic, i.e., applicable to any model
• Based on the learned weights incoming in 1st hidden layer of NN and outgoing paths to the output neuron	• Based on partial dependence decomposition and calculates share of variance explained by interaction
• Assumes that the interactions are learned by the neural network and happen in the first hidden layer	• Assumes that features can be shuffled, which is violated if features are strongly correlated
• Does not allow for comparison across different NNs	• Allows for comparison across different models
• Always leads to the same result for a fixed NN	• May lead to unstable results depend. on used data
• Computationally fast	• Computationally time-consuming

Both methods

- Can detect interactions of order higher than 2 (more computationally demanding)
- Can detect various forms of interactions, independently of their specific structure
- Do not provide the functional form of the interaction
- Do not clearly indicate whether the interaction is statistically significant