A tree-based Varying Coefficient Model

Henning Zakrisson

June 17, 2024

Department of Mathematics, Stockholm University
Insurance Data Science Conference 2024
Based on joint work with Mathias Lindholm

Problem statement

Consider a random variable

$$Y|x \sim F(\mu(x))$$

where F is a member of the exponential dispersion family. The mean function

$$\mathbb{E}[Y|x] = \mu(x)$$

for feature vector $x \in \mathcal{X}$ is unknown. Let $\mathcal{L}(\mu(x_i), y_i)$ denote the negative log-likelihood of the model for a given observation (x_i, y_i).

Example

Assume $Y \sim \text{Poisson}(\mu(x_1, x_2, z))$, represents the total number of claims for a car insurance policyholder, and that

- x_1 is the policyholder's age,
- x_2 is the policyholder's current bonus level,
- z is the name of the region where the policyholder lives.

Then, $\mu(x_1, x_2, z)$ could represent the expected claim amount for a policyholder in region z of age x_1 and with bonus level x_2.

A model that can be used for a problem like this is the Generalized Linear Model (GLM) as

$$g\left(\mu(x)\right) = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$$

where g is a link function, and $\beta_0, \beta_1, \dots, \beta_p$ are model parameters, all in $\mathbb{R}$.

Since $\mathcal{L}(\mu_i, y_i)$ is convex in μ_i, the loss for a set of parameters,

$$\mathcal{L}(\beta; y) = \sum_{i=1}^{n} \mathcal{L}(\mu(x_i), y_i; \beta)$$

is also convex in β, where $\beta = (\beta_0, \beta_1, \dots, \beta_p)$, meaning that the GLM can be fit as

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^{p+1}} \mathcal{L}(\beta; y).$$

This yields a flexible and easy-to-interpret model, where predictions on out-of-sample data point x can be made as

$$\widehat{\mu}(x) = g^{-1}\left(\hat{\beta}_0 + \sum_{j=1}^{p} \hat{\beta}_j x_j\right).$$

Example

In the example, a GLM with parameters $\beta_0, \beta_1, \beta_2$ could be interpreted as

- $\exp(\beta_0)$ is the expected claim amount for a policyholder at age 0 and bonus level 0
- $\exp(\beta_1)$ is the expected increase factor in claim amount for a one-year increase in age
- $\exp(\beta_2)$ is the expected increase factor in claim amount for a one-unit increase in bonus level

However, the effect of age and bonus level can only be modeled linearly, and the categorical variable z cannot be included in the model without further feature engineering.

An alternative is to use a machine learning model, such as a Gradient Boosting Machine (GBM) (Friedman 2001), which assumes the more flexible form

$$g\left(\mu(x)\right) = b(x;\psi) = \sum_{k=1}^{K} f(x;\nu_k)$$

where f is a regression tree parameterized by ν_k, and K is the number of trees in the model. ψ is the set of all parameters in the model.

The GBM is fit by iteratively fitting trees to the negative gradient of the loss function, i.e.

$$d_i = -\frac{\partial}{\partial u}\mathcal{L}(\mu(u), y_i)\bigg|_{u=b(x_i;\hat{\psi}^{(-1)})}$$

$$\hat{\nu}_k = \arg\min_{\nu_k} \sum_{i=1}^{n} \left(f(x_i;\nu_k) - d_i\right)^2$$

where $b(x;\hat{\psi}^{(-1)})$ is the current model.

The GBM functional form is very flexible, and can model complex relationships between input variables and the response.

However, it is less interpretable than a GLM, as the model parameters are not directly interpretable.

Example

Say that the effect of age is significant in years $20 - 30$, but not after that. A GBM could capture this relationship, whereas a GLM could not.

Also, a GBM can model the effect of the categorical variable z without further feature engineering, given a tree structure that can split on the categorical variable.

However, the nice interpretation of the GLM is lost, since the model parameters are not directly interpretable.

A Varying Coefficient Model (VCM) (Hastie and Tibshirani 1993), allows the model parameters to vary with the input variables, i.e.

$$g\left(\mu(x)\right) = \beta_0(z) + \sum_{j=1}^{p} \beta_j(z) x_j$$

where $\beta_0(z), \beta_1(z), \ldots, \beta_p(z)$ are model parameter functions of modifier features z.

This gives the model plenty of flexibility over a standard GLM, since the model parameter functions can be fit using any regression method. It also retains some local interpretability.

Example

For an individual in region z, the VCM parameter function values can be interpreted as

- $\exp(\beta_0(z))$ is the average claim amount for a policyholder in region z
- $\exp(\beta_1(z))$ is the expected increase factor in claim amount for a one-year increase in age for a policyholder in region z
- $\exp(\beta_2(z))$ is the expected increase factor in claim amount for a one-unit increase in bonus level for a policyholder in region z

However, potential non-linear relationships between age and claim amount would not be captured by this model.

Example

If one however adds x_1 as an input to the parameter functions, i.e.,

$$g(\mu(x)) = \beta_0 + \sum_{j=1}^{p} \beta_j(z, x_1, x_2)x_j$$

the model can capture non-linear relationships between age and claim amount. Note though, that the interpretability of β_1 is no longer as clear, as we can no longer guarantee that the the value of β_1 is constant when x_1 changes.

Examples of VCMs

One example of a VCM is the LocalGLMNet model (Richman and Wüthrich 2023), which uses a neural network with a skip connection to model the parameter functions. It assumes that the feature sets x and z are equal, making the model very flexible.

Another example of a VCM can be found in Decision tree bossted VCMs (Zhou and Hooker 2022), where the model parameters functions are modeled using regression trees.

In this work, we propose a tree-based VCM, with the following model architecture:

$$g\left(\mu(x,z)\right) = a(x,z;\theta) = \beta_0 + \sum_{j=1}^{p} \beta_j(z_j;\psi_j)x_j$$

where

$$\beta_j(z_j;\psi_j) = \sum_{k=1}^{K_j} f(z_j;\nu_{jk})$$

for $j = 1, \ldots, p$, where f is a regression tree with parameters ν_{jk}, and K_j is the number of trees for parameter function β_j. Here, z_j represents a set of modifier features for input variable x_j.

The model is fit by iteratively fitting trees to the negative gradient of the loss function in a cyclic manner, i.e.

$$d_{ij} = -x_{ij} \frac{\partial}{\partial \mu} \mathcal{L}(\mu, y_i) \Big|_{\mu = \mu(u_i)} \cdot \frac{\partial g^{-1}(u)}{\partial u} \Big|_{u = a(x_i; \hat{\theta}^{(-1)})}$$

$$\hat{\nu}_{jk} = \arg\min_{\nu_{jk}} \sum_{i=1}^{n} \left(f_j(z_{ij}; \nu_{jk}) - d_{ij} \right)^2$$

where $a(x; \hat{\theta}^{(-1)})$ is the current model.

Some advantages of this tree-based VCM include:

- For disjoint feature sets z and x, the model is highly interpretable locally.
- Using parameter-wise early stopping (see On cyclic gradient boosting by Delong et al. 2023), different parameter function complexity is allowed, allowing for e.g. fully linear relationships for some input variables.
- Using parameter-wise feature importance scores, the relationship between modifiers z and input variables x can be mapped more clearly, allowing the models to be tuned to smaller feature sets.

Consider the following model from Richman and Wüthrich 2023:

Let $Y|x \sim \mathcal{N}(\mu(x), 1)$ have mean function

$$\mu(x) = \frac{1}{2}x_1 - \frac{1}{4}x_2^2 + \frac{1}{2}|x_3|\sin(2x_3) + \frac{1}{2}x_4x_5 + \frac{1}{8}x_5^2x_6$$

where x are drawn from a Normal distribution with mean 0 and variance 1.

Features x_2 and x_8 have a correlation of 0.5. Note that x_7 and x_8 are not used in the model.

The VCM model captures the structure far better than a GLM...

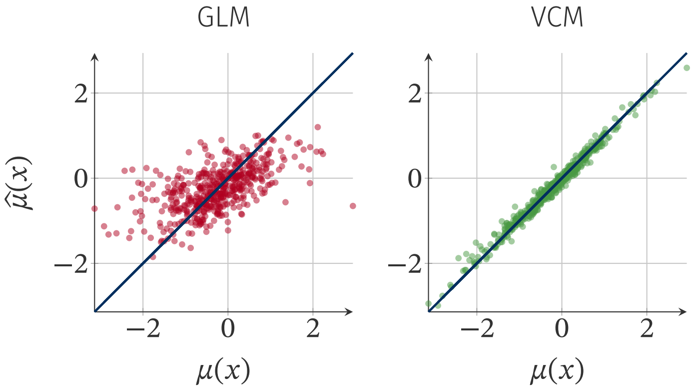

Figure 1: GLM vs VCM predictions

…while maintaining some of the local interpretability

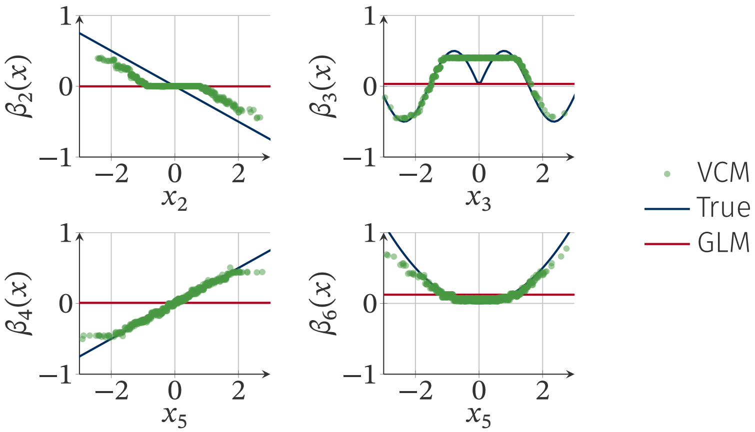

Figure 2: GLM vs VCM coefficient predictions

Also, like in Richman and Wüthrich 2023, the model is tested on real life insurance data.

The `freMTPL2freq` dataset contains $678,013$ observations of the number of claims for French car insurance policies as well as various features.

	Model	Train	Test
	GLM	24.18	24.22
	GBM	23.85	23.89
	Tree-based VCM	23.75	23.84
	LocalGLMNet	23.73	23.95

Table 1: Poisson deviance for the models

Real data example

The feature importance scores for the parameter functions allows for further analysis and feature selection.

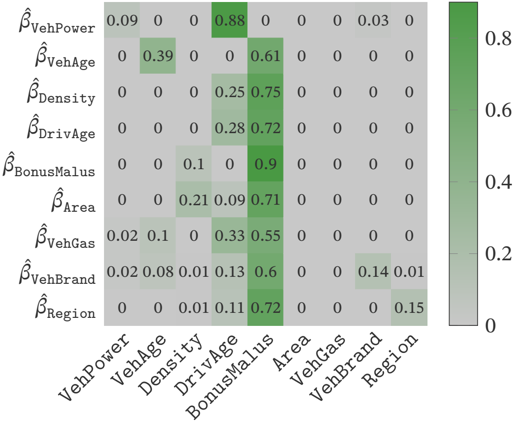

Figure 3: Feature importance scores for the parameter functions

Thank you for your attention!

Interested? Check out

- Preprint at `arxiv.org/pdf/2401.05982`
- Code at `github.com/henningzakrisson/local-glm-boost`

References

- Delong, Łukasz, Lindholm, Mathias, and Zakrisson, Henning (2023). **On Cyclic Gradient Boosting Machines.** *Available at SSRN 4352505.*
- Friedman, Jerome H (2001). **Greedy function approximation: a gradient boosting machine.** *Annals of statistics*, pp. 1189–1232.
- Hastie, Trevor and Tibshirani, Robert (1993). **Varying-coefficient models.** *Journal of the Royal Statistical Society Series B: Statistical Methodology* 55.4, pp. 757–779.

- Richman, Ronald and Wüthrich, Mario V (2023). **LocalGLMnet: interpretable deep learning for tabular data.** *Scandinavian Actuarial Journal* 2023.1, pp. 71–95.
- Zakrisson, Henning and Lindholm, Mathias (2024). **A tree-based varying coefficient model.** *arXiv: 2401.05982.*
- Zhou, Yichen and Hooker, Giles (2022). **Decision tree boosted varying coefficient models.** *Data Mining and Knowledge Discovery* 36.6, pp. 2237–2271.