# Imbalanced Learning using Actuarial Modified Loss Function in Tree-Based Models

## The 56th Actuarial Research Conference

Changyue Hu, Zhiyu Quan, Alfred Chong
University of Illinois at Urbana-Champaign

# Outline

- Why - Motivation
    - Imbalance problem in insurance claim prediction
    - Pitfall of CART
- How - Modification to loss function of CART
    - WSSE loss function
    - Canberra loss function
- What - Results on simulated dataset

# Motivation

- Insurance loss datasets usually contain a high percentage of zero claims.
- Problem of imbalance:
    - Majority (zero claims), minority (nonzero claims).
    - Standard algorithms fail to properly depict data characteristics and therefore yield poor prediction accuracy.

# Imbalanced learning techniques

- Resampling: rebalance the sample space.
    - Over-sampling: add more samples from the minority.
    - Under-sampling: removing samples from the majority.
- Ensemble methods: combine weak learners to improve prediction ability.
    - Parallel-based ensembles: bagging.
    - Iterative-based ensembleg: boosting.
- Cost-sensitive learning: different costs for different prediction errors.

We borrow the idea of cost-sensitive learning to modify the loss function of CART.

# Overview of CART algorithm and notation

- Step 1: Grow a large tree.
  - Recursive binary splitting.
- Step 2: Prune the large tree.
  - Cost-complexity pruning.
- Notation:
  - Response variable as $\mathbf{Y}$ from the sample space as $\mathcal{Y}$.
  - $N$ denotes the number of observations.
  - $i$th sample with $p$-dimensional explanatory variables is denoted as $\mathbf{X}_i = (x_{i1}, x_{i2} \ldots, x_{ip}), \quad i = 1 \ldots N,$ which is sampled from the space $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_p$ .

# CART - grow a large tree

- Regression tree, denoted by $T(\mathbf{X}, \Theta)$, is produced by partitioning the space of the explanatory variables into $M$ disjoint regions $R_1, R_2, \ldots, R_M$ and then assigning a constant $c_m$ for each region $R_m$, for $m = 1, 2, \ldots, M$.

$$T(\mathbf{X}_i, \Theta) = \sum_{m=1}^{M} c_m \mathbf{1}_{R_m}(\mathbf{X}_i) \text{ where } \Theta = \{R_m, c_m\}_{m=1}^{M}$$

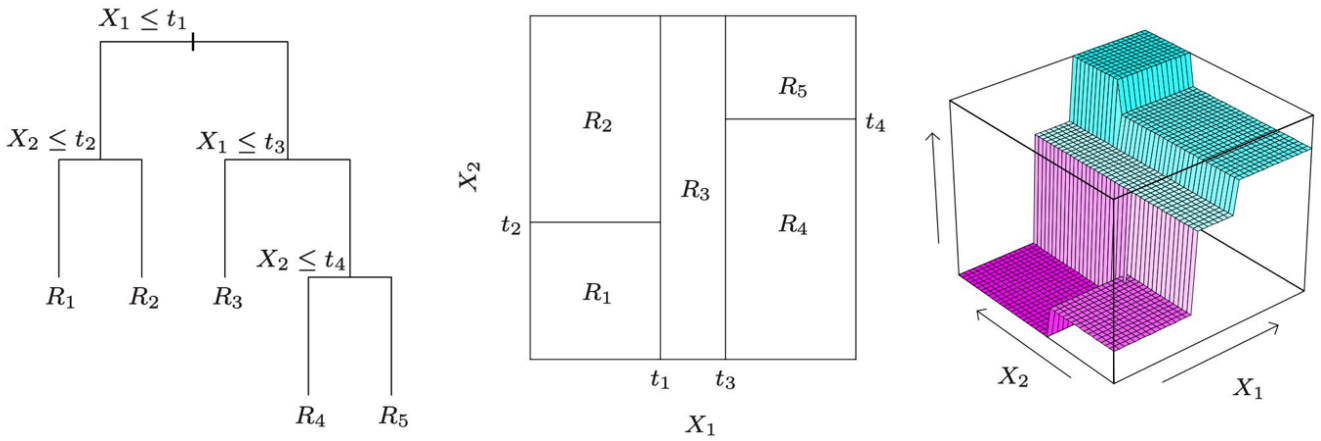



FIGURE 9.2. Partitions and CART. Hastie, T., Tibshirani, R., and Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction.

# CART - splitting criteria

- Sum of Squared Errors (SSE) as loss function.

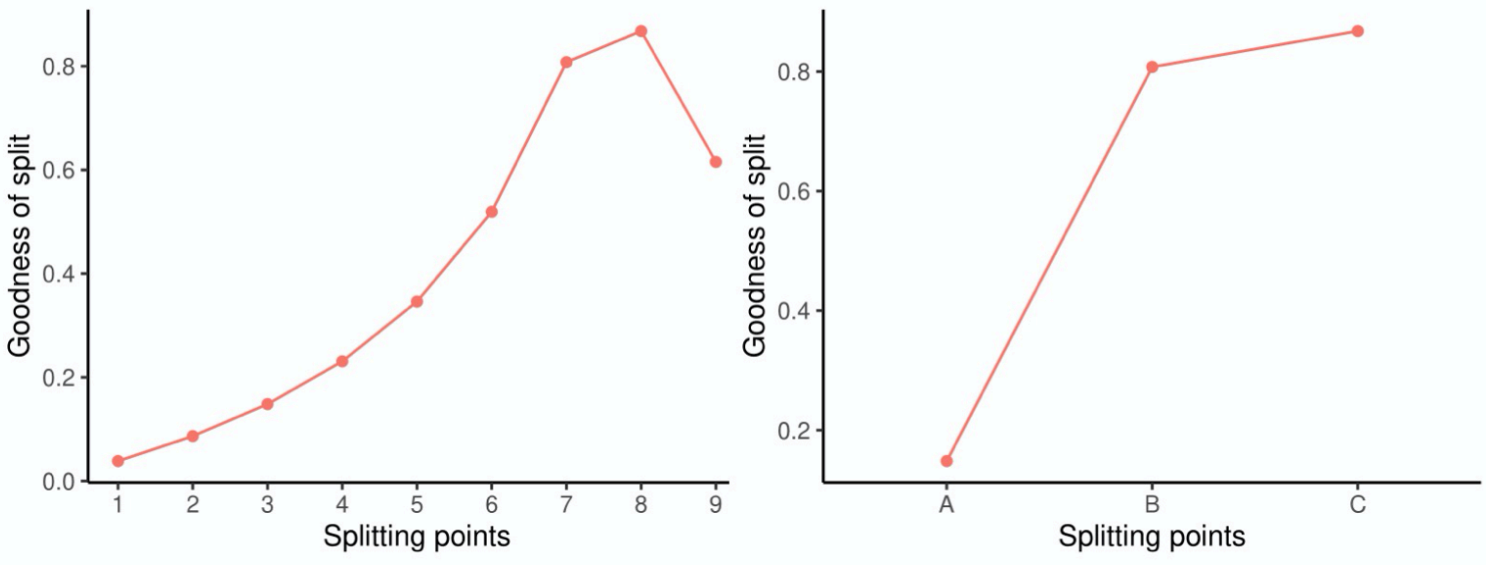
$$L(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2$$

- Two daughter nodes: left node $R_L(j, s) = \{\mathbf{X}_i | X_{.j} < s\}$ and right node $R_R(j, s) = \{\mathbf{X}_i | X_{.j} \geq s\}$ in the case of a continuous explanatory variable,
- Exhaustive search to find the best split： checks all potential splitting points on all possible splitting explanatory variables.
- The best split is the one selected corresponds to the minimum of the sum of loss at two subregions,

$$\underset{j,s}{\operatorname{argmin}} \sum_{i:\mathbf{X}_i \in R_l(j,s)} \left(y_i - \hat{c}_{R_l(j,s)}\right)^2 + \sum_{i:\mathbf{X}_i \in R_r(j,s)} \left(y_i - \hat{c}_{R_r(j,s)}\right)^2.$$

# Motivating example - pitfall of the default split

| OBS. | $X_1$ | $X_2$ | $Y$ |
|---|---|---|---|
| 1 | 1 | A | 0 |
| 2 | 2 | A | 0 |
| 3 | 3 | A | 0 |
| 4 | 4 | B | 0 |
| 5 | 5 | B | 0 |
| 6 | 6 | B | 0 |
| 7 | 7 | B | 0 |
| 8 | 8 | C | 1 |
| 9 | 9 | D | 2 |
| 10 | 10 | D | 3 |




- The default method (ANOVA) under SSE fails to separate zeros and nonzeroes as anticipated.
- The zeros will be grouped together with certain small but non-zero values.
- The sum of squared errors is heavily influenced by the prediction error of the nonzero responses.

# Actuarial modified loss function- WSSE

- Modify the sum of squared errors by assigning a larger weight to the prediction errors of observations with zero responses.
- We define the **weighted sum of squared errors (WSSE)** as a loss function:

$$L_w(\mathbf{y}, \hat{\mathbf{y}}) = w_0 \underbrace{\sum_{i:y_i=0} \left(y_i - \hat{y}_i\right)^2}_{\text{zero claims}} + w_1 \underbrace{\sum_{i:y_i\neq 0} \left(y_i - \hat{y}_i\right)^2}_{\text{nonzero claims}}$$

- $w_0$ and $w_1$ are the hyperparameters denoting the weights for the observations with zero response and non-zero response respectively.
- The default size of the weights, $w_0$ and $w_1$, are determined by the percentage of zero responses in the data.

# Actuarial modified loss function - Canberra loss

- Prediction error between 0 & 1 = Prediction error between 100 & 101 ?
- Canberra distance between $y_i$ and $\widehat{y}_i$ is given as follows:

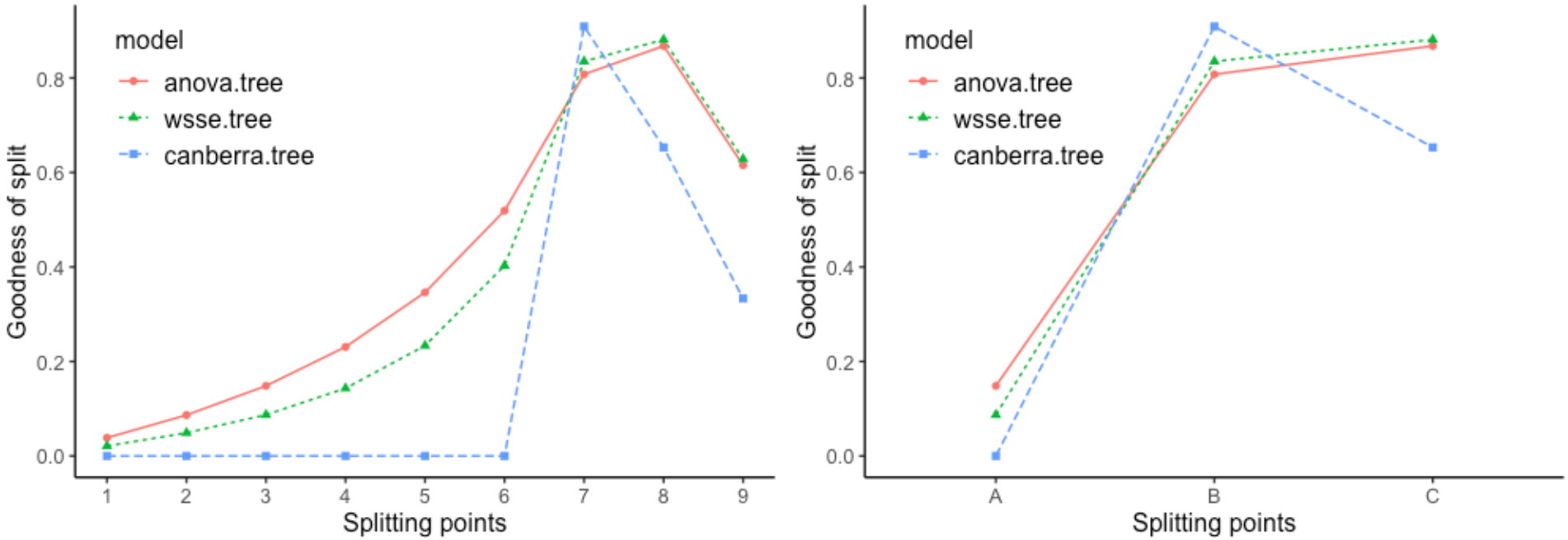
$$d_{\text{CAD}} : (y_i, \widehat{y}_i) \mapsto \frac{|y_i - \widehat{y}_i|}{|y_i| + |\widehat{y}_i|},$$

- Canberra distance is a biased measure and very sensitive to values close to zero.
- We define the **Canberra loss function** as

$$L_c(\mathbf{y}, \widehat{\mathbf{y}}) = \sum_{i=1}^{N} f_{\text{SCAD}}(y_i, \widehat{y}_i), \text{ where } f_{\text{SCAD}}(y_i, \widehat{y}_i) = \begin{cases} 0 & y_i = \widehat{y}_i = 0, \\ \dfrac{(y_i - \widehat{y}_i)^2}{y_i^2 + \widehat{y}_i^2} & \text{otherwise.} \end{cases}$$

| PREDICTION | SQUARED ERROR | CANBERRA | SQUARED CANBERRA |
|---|---|---|---|
| (0, 1) | $(0-1)^2 = 1$ | $\frac{\lvert 0-1\rvert}{\lvert 0\rvert+\lvert 1\rvert} = 1$ | $\frac{(0-1)^2}{0^2+1^2} = 1$ |
| (100, 101) | $(100-101)^2 = 1$ | $\frac{\lvert 100-101\rvert}{\lvert 100\rvert+\lvert 101\rvert} \approx 0.005$ | $\frac{(100-101)^2}{100^2+101^2} \approx 0.00005$ |

# Splitting choice of WSSE tree & Canberra tree




- WSSE tree and the Canberra tree provide better splitting performance than the ANOVA tree when the data contains a large proportion of zero response.
- Canberra tree can effectively separate zeros and nonzeroes as anticipated.

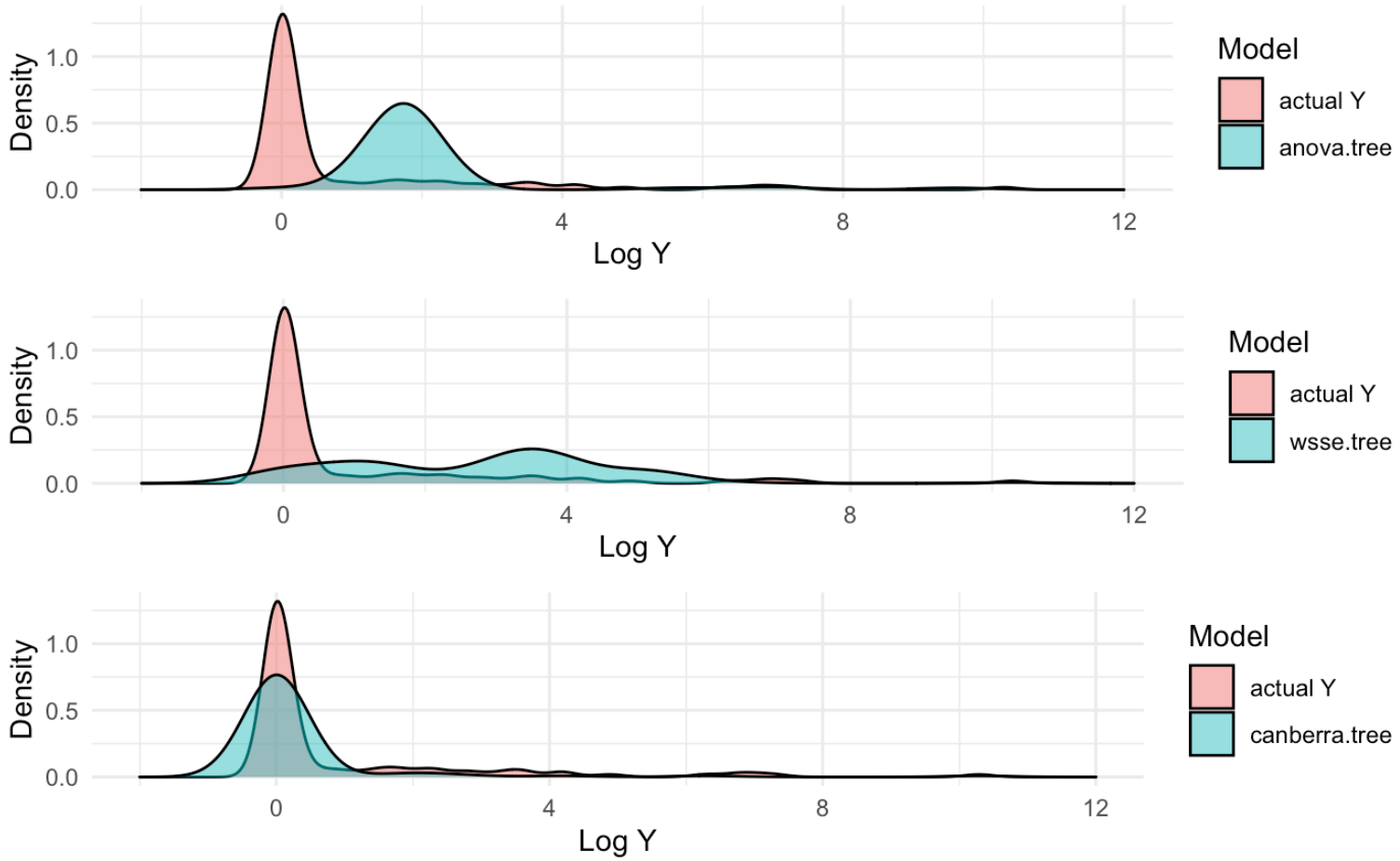
# Simulation study - data generation

- To mimic the real-life insurance datasets, we generate a simulated dataset that contains 50% of zero responses.
- Simulation design:
  - Explanatory variables: $\mathbf{X} = (\mathbf{X}_{categorical}, \mathbf{X}_{continuous})$.
  - $\mathbf{X}_{continuous} \sim \boldsymbol{N}_p(0, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}_{ij} = Cov(\mathbf{X}_i, \mathbf{X}_j) = (0.8)^{i-j}$. $N = 100, p = 5$.
  - $\mathbf{X}_{categorical}$, random sampling from the set of integers $(-3, -2, 1, 4)$, with respective probabilities of $(0.1, 0.2, 0.2, 0.5)$.
  - Linear coefficients: $\boldsymbol{\beta} = (-0.1, \underbrace{1.0, 1.0}_{\text{2 cat}}, \underbrace{0.5, 0.5}_{\text{2 catl}}, \underbrace{0}_{\text{1 cat}}, \underbrace{1.0, 1.0}_{\text{2 con}}, \underbrace{0.5, 0.5}_{\text{2 con}}, \underbrace{0}_{\text{1 con}})^T$
  - Response variable: $\mathbf{Y}$, generated from a Tweedie GLM framework,

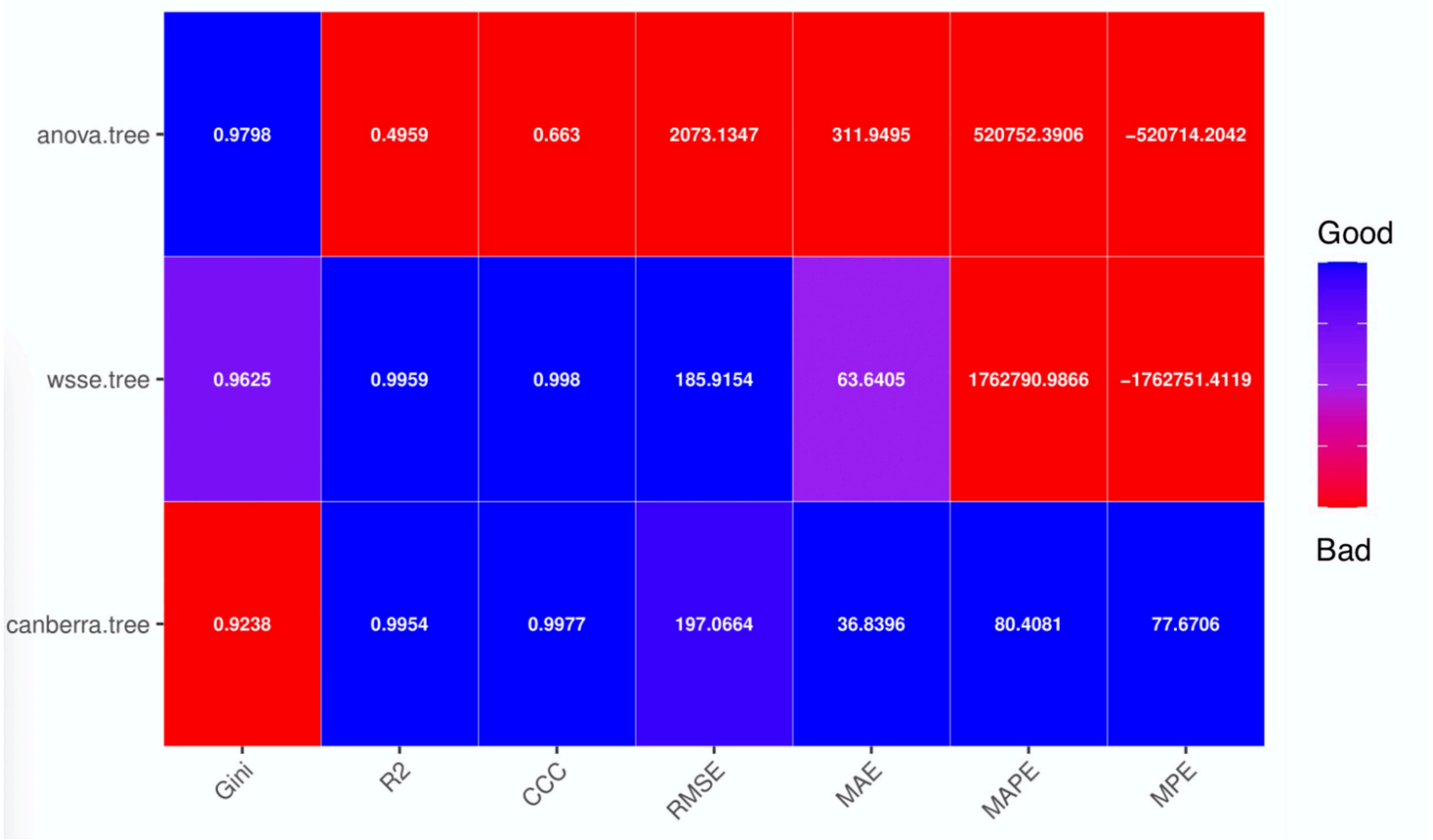
$$y_i \sim Tweedie(\mu_i, \phi, \xi),$$

with the log link function $g(\mu_i) = \log(\mu_i) = \mathbf{X}_i\boldsymbol{\beta}$, the dispersion parameter $\phi = 2$, and the variance power parameter $\xi = 1.7$.

# Result - density plots



# Result - heatmap



# Concluding remarks

- Motivation:
    - The default CART is insufficient to handle insurance datasets that contains a high percentage of zeros.
- Modification:
    - The WSSE tree and the Canberra tree is more effective at separating zero claims from nonzero claims observations than the default CART.
- Simulation Study:
    - The WSSE tree and the Canberra tree offer better prediction performance than the default CART.
- Details about the implementation are given in the paper.

# Selected references


- Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). Classification and Regression Trees.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009).The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer
- Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., and Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications.Expert Systems withApplications, 73:220–239.
- He, H. and Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21(9):1263–1284.
- Lance, G. N. and Williams, W. T. (1966). Computer programs for hierarchical polythetic classification (“similarity analyses”).The Computer Journal, 9(1):60–64.
- Therneau, T. (2019). User written splitting functions for RPART. Technical report, MayoClinic.
- Therneau, T. M. and Atkinson, E. J. (1997). An introduction to recursive partitioning usingthe RPART routines. Technical report, Mayo Foundation

# Q&A

Thank you for your attention!