Advances in Data Analysis and Classification (2020) 14:5–27
https://doi.org/10.1007/s11634-019-00358-7

REGULAR ARTICLE

Count regression trees

Nan-Ting Liu[1] · Feng-Chang Lin[2] · Yu-Shan Shih[1]

Received: 24 September 2018 / Revised: 24 April 2019 / Accepted: 4 May 2019 / Published online: 10 May 2019

Abstract
Count data frequently appear in many scientific studies. In this article, we propose a regression tree method called CORE for analyzing such data. At each node, besides a Poisson regression, a count regression such as hurdle, negative binomial, or zero-inflated regression which can accommodate over-dispersion and/or excess zeros is fitted. A likelihood-based procedure is suggested to select split variables and split sets. Node deviance is then used in the tree pruning process to avoid overfitting. CORE is able to eliminate variable selection bias. In the simulations and real data studies, we show that CORE has some advantages over the existing method, MOB.

Keywords Hurdle model · GUIDE · MOB · Negative binomial model · Score residual · Zero-inflated model

Mathematics Subject Classification 62G08 · 62J12

1 Introduction

Count data commonly appear in economics, health services and social science. The classical Poisson regression model is frequently used to analyze such data. However, it does not perform well when data present over-dispersion and/or excess zeros. Several models, like the negative binomial regression (see Eq. 2), are usually employed to capture over-dispersion. However, in many cases, they fail to handle excess zeros. To deal with the circumstances, the hurdle (zero-altered) model (Mullahy 1986) and

✉ Yu-Shan Shih
mthyss@ccu.edu.tw

Nan-Ting Liu
x88776544pc@gmail.com

Feng-Chang Lin
flin33@email.unc.edu

[1] Department of Mathematics, National Chung Cheng University, Chiayi 621, Taiwan

[2] Department of Biostatistics, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

the zero-inflated model (Mullahy 1986; Lambert 1992) were proposed. An overview of models on count data can be found in Cameron and Trivedi (2013). Neelon et al. (2016) gave recent reviews of modeling zero-modified count data in health services.

Classical tree-structured methods including classification and regression trees are common analytic tools in Statistics and machine learning among others. For example, the method of CART (Breiman et al. 1984) is used frequently in many scientific areas. Easy interpretation and good prediction performance are two key features of these tree methods (Loh 2014). Starting at the root node, the methods recursively divide the sample into two or more subnodes, if splitting conditions are satisfied. A split is produced to divide the sample in each node. The splitting process proceeds until one of the tree's stopping criteria is reached. A pruning method is then employed in choosing of the final right-sized tree which assigns a class label or fits a regression function at each terminal node (Breiman et al. 1984; Loh 2014). For count data, several tree methods which fit Poisson regression to the sample in each node were proposed. The early Poisson tree method was proposed by Ciampi (1991). Later, Loh (2006) extended the GUIDE (Generalized, Unbiased, Interaction Detection and Estimation) tree method (Loh 2002) to count data. However, these approaches did not pay much attention to data with excess zeros. Alternatively, tree methods which considered over-dispersion and/or excess zeros for count data were suggested. Among them, Choi et al. (2005) extended the GUIDE approach to build a regression tree for over-dispersed count data. Meanwhile, Lee and Jin (2006) extended the CART approach to overcome the appearance of excess zeros. But, it only fits a constant model at each node. Zeileis et al. (2008a) proposed the MOB (MOdel-Based recursive partitioning) method. Its algorithm is designed as a generic tool that can be integrated with different types of parametric models (Rusch and Zeileis 2013; Hothorn and Zeileis 2015). For count data, it can fit a Poisson, negative binomial, hurdle or zero-inflated regression model at each node (Zeileis et al. 2008b).

In this paper, we propose a new regression tree method named CORE (for *COunt REgression tree*). It allows us to fit regression models which accommodates data with over-dispersed and/or excess zeros count at each node. The rest of the paper is organized as follows. Some count regression models including the hurdle and zero-inflated regression models and their corresponding partial score residuals are introduced in Sect. 2. The MOB method for count data is briefly described in Sect. 3. Section 4 presents our CORE method and Sect. 5 demonstrates its application. Section 6 compares the two tree methods in simulation experiments. We demonstrate the versatility of our method by analyzing the article counts data in Sect. 7. Finally, conclusions are given in Sect. 8.

2 Count regression models

The Poisson regression which assumes its (conditional) mean and variance are the same is a classical model for count data. To overcome over-dispersion phenomenas in some count data, the negative binomial regression is suggested as an alternative. Later, two parametric models which combine a zero component and a count component to model data with excess zeros were proposed. They are the hurdle model (Mullahy 1986) and the zero-inflated model (Mullahy 1986; Lambert 1992). The former combines a zero hurdle model and a count model while the later combines a point mass at zero with a count regression. In general, both models link the probability of zero point mass or zero hurdle with the covariates using the classical logistic regression. In addition, both models typically assume that the count component follows the Poisson or negative binomial regression model. The definitions of the regression models are given in the following.

Let count response variable be Y with covariate vectors $Z = (1, Z_1, \ldots, Z_{K_0})'$ and $X = (1, X_1, \ldots, X_{K_c})'$, where Z is associated with probability of zeros and X is associated with counts. Let the probability function of zero part be $f_0(y; z, \gamma)$ and the probability function of count part be $f_c(y; x, \beta)$ where $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{k_0})$ and $\beta = (\beta_0, \beta_1, \ldots, \beta_{k_c})$ are the parameter vectors respectively. Denote I_A be the indicator function of set A.

Definition 1 The probability function of the hurdle regression model of Y on Z and X is defined as follows

$$f_0(0; z, \gamma) + \frac{1 - f_0(0; z, \gamma)}{1 - f_c(0; x, \beta)} f_c(y; x, \beta) I_{\{y>0\}}.$$

Definition 2 The probability function of the zero-inflated regression model of Y on Z and X is defined as follows

$$f_0(0; z, \gamma) I_{\{y=0\}} + [1 - f_0(0; z, \gamma)] f_c(y; x, \beta).$$

The logistic link

$$\log\left(\frac{\pi}{1-\pi}\right) = \gamma z \tag{1}$$

is frequently applied to model $\pi = f_0(0; z, \gamma)$ on Z. The probability function of the Poisson regression model is

$$\frac{e^{-\mu}\mu^y}{y!}$$

where $\log(\mu) = \beta x$. The probability function of the negative binomial regression model is

$$\frac{\Gamma(y+\theta)}{\Gamma(\theta)\cdot y!} \cdot \frac{\mu^y \theta^\theta}{(\mu+\theta)^{(y+\theta)}} \tag{2}$$

where $\log(\mu) = \beta x$ and θ is treated as a nuisance shape parameter. Both models are commonly used to model the count probability.

In the following, we give the definition of partial score residuals and a related proposition which plays an important rule in our algorithms of selecting the split variable at each node. Without lost of generality, we demonstrate the proposition for the zero-inflated Poisson regression. The proposition also applies to other count regressions, like the negative binomial regression, the zero-inflated negative binomial regression and the hurdle regression models.

Let $\log(\frac{\pi}{1-\pi}) = \gamma_0+\gamma_1 Z_1+\cdots+\gamma_{K_0} Z_{K_0}$ and $\log(\mu) = \beta_0+\beta_1 X_1+\cdots+\beta_{K_c} X_{K_c}$. Given the observed data $\mathbf{y}' = (y_1, \ldots, y_n)$, $\mathbf{x}'_{k_c} = (x_{1k_c}, \ldots, x_{nk_c})$, $k_c = 1, \ldots, K_c$ and $\mathbf{z}'_{k_0} = (z_{1k_0}, \ldots, z_{nk_0})$, $k_0 = 1, \ldots, K_0$. The log likelihood function of the zero-inflated Poisson regression model is

$$\begin{aligned}\ell(\beta,\gamma) &\equiv \log \mathcal{L}(\beta,\gamma|\mathbf{y},\{\mathbf{x}_{k_c}\},\{\mathbf{z}_{k_0}\})\\ &= \sum_{i=1}^{n}\Big\{\log(\pi_i+(1-\pi_i)e^{-\mu_i})I_{\{y_i=0\}}\\ &\quad+\big[\log(1-\pi_i)-\mu_i+y_i\log\mu_i-\log y_i!\big]I_{\{y_i>0\}}\Big\}.\end{aligned}$$

The maximum likelihood estimates of the parameters are obtained by the following score equations

$$\begin{aligned}\frac{\partial\ell(\beta,\gamma)}{\partial\beta_0} &= \sum_{i=1}^{n}\left\{\frac{-\mu_i(1-\pi_i)e^{-\mu_i}}{\pi_i+(1-\pi_i)e^{-\mu_i}}I_{\{y_i=0\}}+(y_i-\mu_i)I_{\{y_i>0\}}\right\}=0,\\ \frac{\partial\ell(\beta,\gamma)}{\partial\beta_{k_c}} &= \sum_{i=1}^{n}x_{ik_c}\left\{\frac{-\mu_i(1-\pi_i)e^{-\mu_i}}{\pi_i+(1-\pi_i)e^{-\mu_i}}I_{\{y_i=0\}}+(y_i-\mu_i)I_{\{y_i>0\}}\right\}=0,\\ &\text{for }\ k_c=1,\ldots,K_c,\\ \frac{\partial\ell(\beta,\gamma)}{\partial\gamma_0} &= \sum_{i=1}^{n}\left\{\pi_i(1-\pi_i)\left[\frac{(1-e^{-\mu_i})}{\pi_i+(1-\pi_i)e^{-\mu_i}}I_{\{y_i=0\}}-\frac{I_{\{y_i>0\}}}{(1-\pi_i)}\right]\right\}=0,\\ \frac{\partial\ell(\beta,\gamma)}{\partial\gamma_{k_0}} &= \sum_{i=1}^{n}z_{ik_0}\left\{\pi_i(1-\pi_i)\left[\frac{(1-e^{-\mu_i})}{\pi_i+(1-\pi_i)e^{-\mu_i}}I_{\{y_i=0\}}-\frac{I_{\{y_i>0\}}}{(1-\pi_i)}\right]\right\}=0,\\ &\text{for }\ k_0=1,\ldots,K_0.\end{aligned}$$

Let the MLE be $\hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_{K_c})$ and $\hat{\gamma} = (\hat{\gamma}_0, \ldots, \hat{\gamma}_{K_0})$. Denote

$$\log(\hat{\mu}_i) = \hat{\beta}_0+\hat{\beta}_1 x_{i1}+\cdots+\hat{\beta}_{K_c}x_{iK_c}$$

and

$$\log\left(\frac{\hat{\pi}_i}{1-\hat{\pi}_i}\right) = \hat{\gamma}_0+\hat{\gamma}_1 z_{i1}+\cdots+\hat{\gamma}_{K_0}z_{iK_0}.$$

Definition 3 The partial score residuals associated with β_0 and γ_0, denoted by R^{β} and R^{γ}, are defined as follows. For $i = 1, \ldots, n$,

$$\begin{aligned}R_i^{\beta} &= \frac{-\hat{\mu}_i(1-\hat{\pi}_i)e^{-\hat{\mu}_i}}{\hat{\pi}_i+(1-\hat{\pi}_i)e^{-\hat{\mu}_i}}I_{\{y_i=0\}}+(y_i-\hat{\mu}_i)I_{\{y_i>0\}},\\ R_i^{\gamma} &= \hat{\pi}_i(1-\hat{\pi}_i)\left[\frac{(1-e^{-\hat{\mu}_i})}{\hat{\pi}_i+(1-\hat{\pi}_i)e^{-\hat{\mu}_i}}I_{\{y_i=0\}}-\frac{I_{\{y_i>0\}}}{(1-\hat{\pi}_i)}\right].\end{aligned}$$

The score equations imply that $\sum_{i=1}^{n} R_i^{\beta} = 0$, $\sum_{i=1}^{n} x_{ik_c} R_i^{\beta} = 0$, $k_c = 1, \ldots, K_c$, $\sum_{i=1}^{n} R_i^{\gamma} = 0$, and $\sum_{i=1}^{n} z_{ik_0} R_i^{\gamma} = 0$, $k_0 = 1, \ldots, K_0$. These identities yield the following proposition.

Proposition 1 *The sample correlation coefficient between* $\mathbf{R}^{\beta}$ *and* $\mathbf{x}_k'$ *is zero, where* $\mathbf{R}^{\beta} = (R_1^{\beta}, \ldots, R_n^{\beta})$*, for* $k = 1, \ldots, K_c$*.*
The sample correlation coefficient between $\mathbf{R}^{\gamma}$ *and* $\mathbf{z}_k'$ *is zero, where* $\mathbf{R}^{\gamma} = (R_1^{\gamma}, \ldots, R_n^{\gamma})$*, for* $k = 1, \ldots, K_0$ *.*

This proposition indicates that if covariate X (Z) is fitted well in the model, the sample correlation coefficient between its values and the associated partial score residuals $\mathbf{R}^{\beta}$ $(\mathbf{R}^{\gamma})$ should be zero. A violation of this identity suggests that such covariate should be split. A test of independence between $X_{k_c}, k_c = 1, \ldots, K_C$ or $Z_{k_0}, k_0 = 1, \ldots, K_0$ and their associated partial score residuals could tell which covariate is the split variable. This is the basic principle that we use to select split variables and the detailed algorithm is given in Sect. 4.1.

3 MOB tree

The MOB method fits a parametric model to the sample in current node and assesses the stability of the parameters across each split variable. The testing for parameter instability is carried out by considering the empirical fluctuation process (Zeileis and Hornik 2007) corresponding to each split variable. Under the null hypothesis of parameter stability, the process converges to a Brownian bridge. A test statistic is obtained by applying a scalar functional to the process that captures deviations from zero. For ordered variable, the limiting distribution of the test statistic is the supremum of a tied-down Bessel process. For categorical variable, its limiting distribution is a chi-squared distribution. The test statistics and the associated p values are computed. If the smallest p values is significant, the associated variable is selected as the split variable. Otherwise, the algorithm stops splitting and declares the current node terminal.

MOB separates split variable selection from split set selection. After the split variable is selected, the split set is chosen by considering all possible binary split sets. The set that maximizes the partitioned likelihood is selected. The procedure is applied recursively until the node size is too small or the related instability tests are not significant. In each terminal node, the fitted parametric regression function is used to predict future observations. Detailed description of the MOB method is given in Zeileis et al. (2008a) and Rusch and Zeileis (2013). The `R` packages `partykit` (Hothorn and Zeileis 2015) and `countreg` (Zeileis et al. 2008b) with the default options are used to obtain the MOB results for count data.

4 The CORE method

For count data, CORE provides versatile models at each node. It accommodates overdispersion and/or excess zeros count by fitting a Poisson, negative binomial, hurdle or zero-inflated regression model to the sample in each node. Once the model is decided, it selects the split variable at each node first. After the split variable is determined, the associated split set is chosen. For split variable selection, it basically follows the GUIDE algorithms which utilize the residuals of the fitted model at each node to select the split variable (Loh 2006, 2009). The partial score residuals of the fitted model are used in our algorithms. Based on Proposition 1 and the signs patterns of its partial score residuals, we employ the Pearson chi-squared test of independence to select split variables. The usage of the signs patterns allows us to discover the split variables when the hurdle or zero-inflated model is fitted (steps 4 and 6 of Procedure 1) . It also helps us detect pairwise interactions between covariates (Procedure 2). In order to use the test, we follow GUIDE's approach which divides the values of the ordered covariate(s) into four groups in its split variable selection (Loh 2002). The algorithm is given in Sect. 4.1. Furthermore, the likelihood of the fitted model facilitates the finding of the split sets and the right-sized tree. The selection method of the split sets is given in Sect. 4.2. The tree pruning method is given in Sect. 4.3.

4.1 Split variable selection

The following two procedures are applied to the covariates at each node. Let ω be a x-covariate or z-covariate. To keeping away from the difficulties of computing very small p values of the Pearson chi-squared test statistic, we use the Wilson-Hilferty approximation which is adopted by GUIDE (Loh 2009).

Procedure 1. Main effect detection:

1. Fit a count regression, like the Poisson, negative binomial, hurdle or zero-inflated model to the current sample.
2. Obtain the partial score residuals R_i^{β} and R_i^{γ}, $i = 1, \ldots, n$ from the fitted model.
3. Divide each set of the residuals into two groups according to their signs: positive versus non-positive. Totally, there are at most 4 possible patterns for the residual sign vector.
4. For each ordered covariate, divide the data into four levels at the sample quartiles; construct a two-way 2×4 contingency table, if the Poisson or negative binomial regression is used to fit the sample; otherwise, fit the zero-inflated or hurdle model to the sample and construct a 4×4 contingency table with the levels as columns and the signs of the partial score residuals as rows; count the number in each cell.
5. After removing entries with zero column totals, compute the Pearson chi-squared test statistic χ_ν^2 of independence. If $\nu > 1$, use the following Wilson-Hilferty approximation (Wilson and Hilferty 1931) to convert χ_ν^2 to the 1-d.f. chi-squared

$$W_1(\omega) = \max\left(0, \left[\frac{7}{9} + \sqrt{\nu}\left\{\left(\frac{\chi_\nu^2}{\nu}\right)^{\frac{1}{3}} - 1 + \frac{2}{9\nu}\right\}\right]^3\right). \tag{3}$$

6. Do the same for each categorical covariate, using the categories of the covariate to form the columns of the contingency table and omitting entries with zero column totals.

Procedure 2. Interaction effect detection:

1. For a pair of ordered covariates (ω_i, ω_j), where both covariates are x-covariate or z-covariate, divide the (ω_i, ω_j)-space into four quadrants by splitting the range of each variable into two halves at the sample median; construct a two-way 2×4 or 4×4 contingency table with the quadrants as columns and the sign patterns of the partial score residual vectors as rows; count the number in each cell.
2. Do the same for each pair of categorical covariates, using their categorical pairs to form the columns of the contingency table and omitting entries with zero column totals.
3. For each pair of covariates (ω_i, ω_j) where ω_i is ordered and ω_j is categorical, divide the ω_i-space into two categories at the sample median; use their categorical pairs to form the columns of the contingency table and omitting entries with zero column totals.
4. Compute its chi-squared test statistic and use (3) to transform it to a 1-d.f. chi-squared value $W_2(\omega_i, \omega_j)$.

Based on the two procedures, the selection method of the split variable at each node is given in the following.

Algorithm (Split variable selection) Let K be the number of distinct covariates. Let $\alpha_1 = \frac{.05}{K}, \alpha_2 = \frac{0.1}{K(K-1)}$ and $\chi^2_{1,\alpha}$ be the $100(1-\alpha)$th percentile of the chi-squared distribution with one degree of freedom.

1. Apply Procedure 1 to obtain $W_1(\omega_i), i = 1, \ldots, K$.
2. If $\max_i W_1(\omega_i) > \chi^2_{1,\alpha_1}$, choose the variable associated with the largest value of $W_1(\omega_i)$ and exit.
3. Otherwise, apply Procedure 2 to obtain $W_2(\omega_i, \omega_j)$ for each pair of covariates
   (a) If $\max_{i \neq j} W_2(\omega_i, \omega_j) > \chi^2_{1,\alpha_2}$, select ω_i if $W_1(\omega_i) > W_1(\omega_j)$, otherwise select ω_j and exit.
   (b) Otherwise, select the variable associated with the largest value of $W_1(\omega_i)$.

4.2 Split set selection

After the split variable, say ω_s, is selected, the split set is determined by comparing all possible binary splits generated by ω_s on node deviance (impurity). Let $\mathbf{y}_t$ be the response vector and $\mathbf{x}_{k_c,t}, k_c = 1, \ldots, K_C$ and $\mathbf{z}_{k_0,t}, k_0 = 1, \ldots, K_0$ be the covariates associated with the observations in node t. Define the node deviance based on the fitted likelihood function $\mathcal{L}$

$$I(t) = -\log \left\{ \max_{\pi,\mu} \left\{ \mathcal{L}\left(\pi, \mu | \mathbf{y}_t, \{\mathbf{x}_{k_c,t}, k_c = 1, \ldots, K_c\}, \{\mathbf{z}_{k_0,t}, k_0 = 1, \ldots, K_0\}\right)\right\}\right\}.$$

Let S be the collection of all the binary splits based on ω_s. Define $\phi(\zeta, t) = I(t) - I(t_{\zeta_l}) - I(t_{\zeta_r})$, where t_{ζ_l} and t_{ζ_r} are the left and right child node of node t based on $\zeta \in S$. The split set s^* which satisfies

$$\phi(s^*, t) = \max_{\zeta \in S} \phi(\zeta, t)$$

is chosen. That is, the set which reduces the maximum amount of the node impurity is selected. Besides, we do not consider the split sets which separate zero responses to one node and all non-zero responses to the other node.

As an example, we derive the explicit formula for $\phi(\zeta, t)$ when the Poisson regression model is fitted. Suppose $\mathbf{y}'_t = (y_{1,t}, y_{2,t}, \ldots, y_{n_t,t})$ and $\hat{\mu}_t$ is the MLE of parameter μ_t. Then

$$I(t) = n_t\hat{\mu}_t - \log \hat{\mu}_t \sum_{i=1}^{n_t} y_{i,t} + \sum_{i=1}^{n_t} \log(y_{i,t}!).$$

Denote

$$i(t) = n_t(\hat{\mu}_t - \bar{y}_t \log \hat{\mu}_t)$$

where $\bar{y}_t$ is the mean of the response values at node t. We have

$$\phi(\zeta, t) = i(t) - i(t_{\zeta_l}) - i(t_{\zeta_r}).$$

Thus, we may use $i(t)$ as the impurity function at node t and find the split set accordingly.

4.3 Tree pruning

The aforementioned splitting method is applied recursively until one of the stopping criteria is reached. At each node, common stopping criterion, like all covariates are the same or all the response values are the same, is adopted. Moreover, node t is considered terminal, if the number of cases in t is less than 5% of the total cases. Afterwards, the cost-complexity pruning method (Breiman et al. 1984) is adopted to find the right-sized tree. At each node, the node deviance is used as the error measure in the pruning algorithm (Breiman et al. 1984, Chapter 10). The tree is then pruned by the cost-complexity method with ten-fold cross-validation. Node deviance has been used to prune trees in Chan and Loh (2004) among others.

5 Solder data

As illustration, a CORE tree which is the result of analyzing the solder data is shown in Fig. 1. The solder data is originally reported in Comizzoli et al. (1990) and is obtained from the `R` package `rpart`. It contains 720 observations which are the result of an experiment on wave-soldering of electronic components in a printed circuit board. The response variable is the number of solder skips. The covariates are Mask, type and thickness of the material for the solder mask (4 levels), Opening, thickness of clearance around a mounting pad (3 levels), PadType, geometry and size of the mounting pad (10 levels), Panel, panel position on a board (3 levels) and Solder, thickness of solder (2 levels). The response values contain 242 zeros.

Fig. 1 CORE tree for the solder data. At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample size and the expected frequency of the responses are printed below nodes

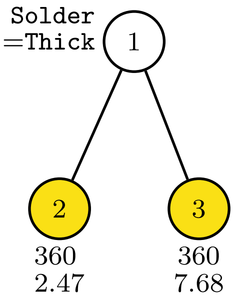

Table 1 Regression coefficients and the associated z scores of the negative binomial regression model in each terminal node of Fig. 1

	Node 2		Node 3	
	Coefficient	$\lvert z \rvert$	Coefficient	$\lvert z \rvert$
Intercept	− 2.459	10.35	0.062	0.51
OpeningM	0.860	5.44	0.116	1.37
OpeningS	2.455	17.68	1.829	26.17
MaskA3	0.482	2.41	0.439	4.62
MaskB3	1.836	10.75	1.020	11.57
MaskB6	2.517	15.25	1.794	21.57
PadTypeD6	− 0.332	1.88	− 0.451	4.09
PadTypeD7	0.145	0.91	− 0.162	1.54
PadTypeL4	0.737	5.09	0.177	1.77
PadTypeL6	− 0.355	2.00	− 0.741	6.34
PadTypeL7	0.059	0.36	− 0.673	5.84
PadTypeL8	0.210	1.34	− 0.388	3.56
PadTypeL9	− 0.563	3.00	− 0.635	5.55
PadTypeW4	0.004	0.02	− 0.184	1.75
PadTypeW9	− 1.289	5.45	− 1.548	10.79
Panel2	0.231	2.52	0.394	6.00
Panel3	0.063	0.67	0.394	6.00
θ	34.8		18.56	

Table 2 Distributions of X variables used in the simulation studies

$X_1 \sim G$
$X_2 \sim U$
$X_3 \sim W$
$X_4 \sim B$
$X_5 \sim C$

G, U, W, B, and C are mutually independent; G has a standard normal distribution, U has a discrete uniform distribution on integers over $[-2, 2]$, W has a Poisson distribution with mean 1, B has a beta distribution, $\mathcal{B}(a, b)$ where $a = 5$ and $b = 2$ and C has a uniform distribution on $(-1, 1)$

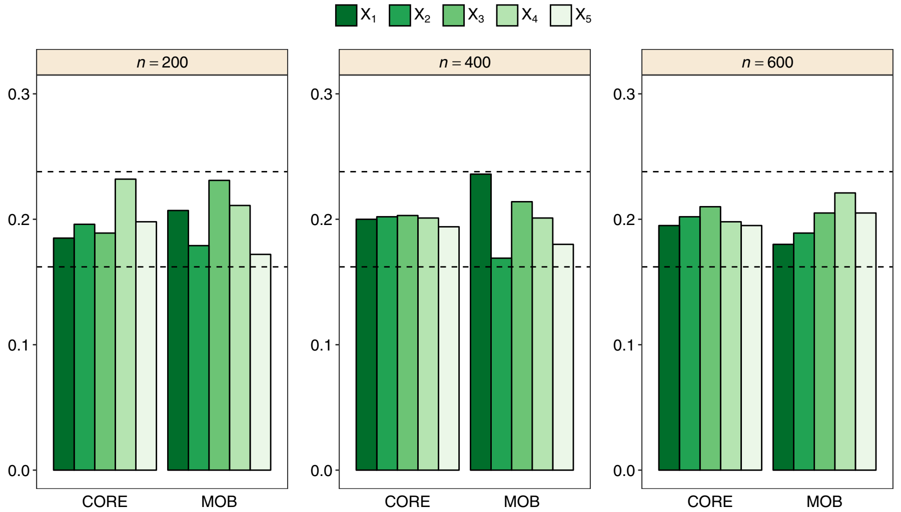

Fig. 2 Estimated probabilities of split variable selection of the two methods. The response variable is independent of the covariates. The response variable follows a zero-inflated negative binomial regression model with $\log(\mu) = 1$, $\log(\frac{\pi}{1-\pi}) = -1.5$ and shape parameter $\theta = 10$. The distributions of Xs follow the distributions given in Table 2. The dotted lines are three simulation standard errors of 0.2

Table 3 Experiments for variable selection studies of the two tree methods. In each scenario, a zero-inflated negative binomial regression model with parameters π, μ, and shape parameter $\theta = 10$ is generated

Scenario	$\log(\mu)$	$\log(\frac{\pi}{1-\pi})$
J–J	$1 + \beta_1 I_{\{X_5>0\}}$	$-1.5 + 3\gamma_1 I_{\{X_2>0\}}$
Q–Q	$1 + \beta_1 X_5^2$	$-1.5 + \gamma_1 X_2^2$
J–Q	$1 + \beta_1 I_{\{X_5>0\}}$	$-1.5 + \gamma_1 X_2^2$
Q–J	$1 + \beta_1 X_5^2$	$-1.5 + 3\gamma_1 I_{\{X_2>0\}}$
INT	$1 + 0.5(X_2 + X_5 + \beta_1 X_2 X_5)$	$-1.5 + 0.5(X_2 + X_5 + \gamma_1 X_2 X_5)$

The link function $\log(\frac{\pi}{1-\pi})$ depends X_2 with coefficient $\gamma_1 \in \{0.0, 0.2, 0.5, 1.0\}$ and $\log(\mu)$ is a function of X_5 with coefficient $\beta_1 \in \{0.0, 0.2, 0.5, 1.0, 2.0\}$ where γ_1 and β_1 are not zero simultaneously. The distributions of Xs are given in Table 2

Two regular tests (Cameron and Trivedi 2013, Sect. 3.4.1) on the null hypothesis of equidispersion in Poisson regression against the alternative of over-dispersion strongly indicate that the data are over-dispersed (p value $< 10^{-7}$). Thus, a negative binomial regression model is more suitable for analyzing this data. We use all the covariates to split the sample and to fit a negative binomial regression model at each node except that Solder is used for splitting only. The resulting tree is given in Fig. 1. Our tree method divides the whole sample into two groups (terminal nodes). The samples with thick solder form one group (node 2). The samples with thin solder are in another group (node 3). In each terminal node, a negative binomial regression model is fitted and the estimated coefficients are given in Table 1. In general, the expected skip counts of the thin solder group is larger than that of the thick solder group.

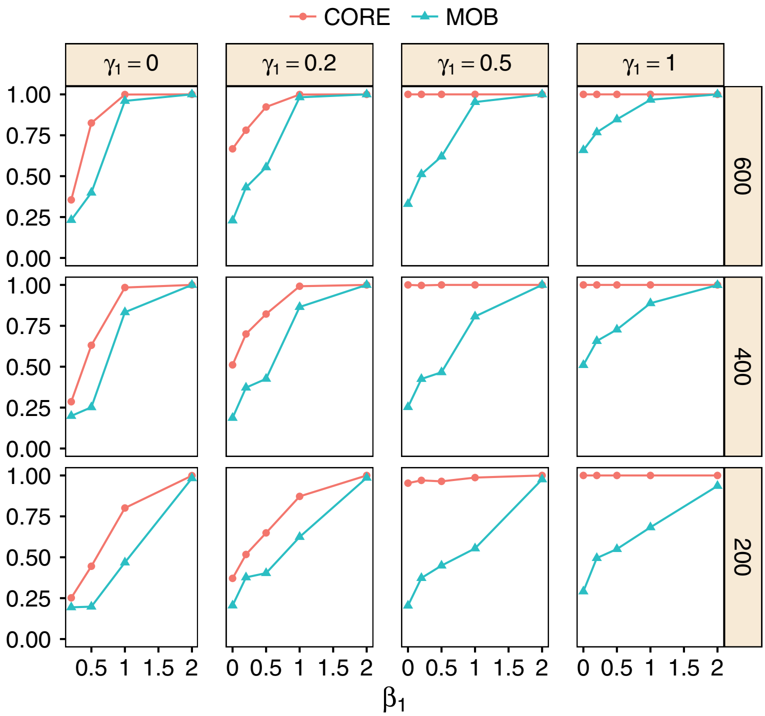

Fig. 3 Estimated probabilities of split variable selection of the two methods under Scenario J–J in Table 3 with sample size 200, 400 or 600. The maximum value of the estimated standard error is about 0.016

To assess its prediction accuracy, we use ten-fold cross validation to obtain its average prediction MAE or MSE (see Eq. 4). Cross-validation is a re-sampling technique for assessing model performance (Breiman et al. 1984; Kuhn and Johnson 2013). The data set is randomly divided into ten roughly equal pieces. One piece is held out in turn and a regression estimate is constructed from the remaining nine pieces. The held-out piece is then used to estimate the prediction MAE or MSE. The average of the ten estimated MAE or MSE is used to assess prediction accuracy. For the CORE tree, the values are 1.76 and 10.0 respectively. The corresponding values for the classical negative binomial regression model are 2.01 and 15.92 respectively. Overall, the former has better prediction power than the latter!

6 Simulations

We compare our CORE method with the MOB method on split variable selection and prediction accuracy in this section. Five mutually independent covariates are simulated and their distributions are given in Table 2. Covariates X_1, X_3 and X_4 are different noise variables. Covariates X_2 and X_5 act as potential informative variables. Besides, X_2 has smaller number of possible split points than the others. The response variable follows

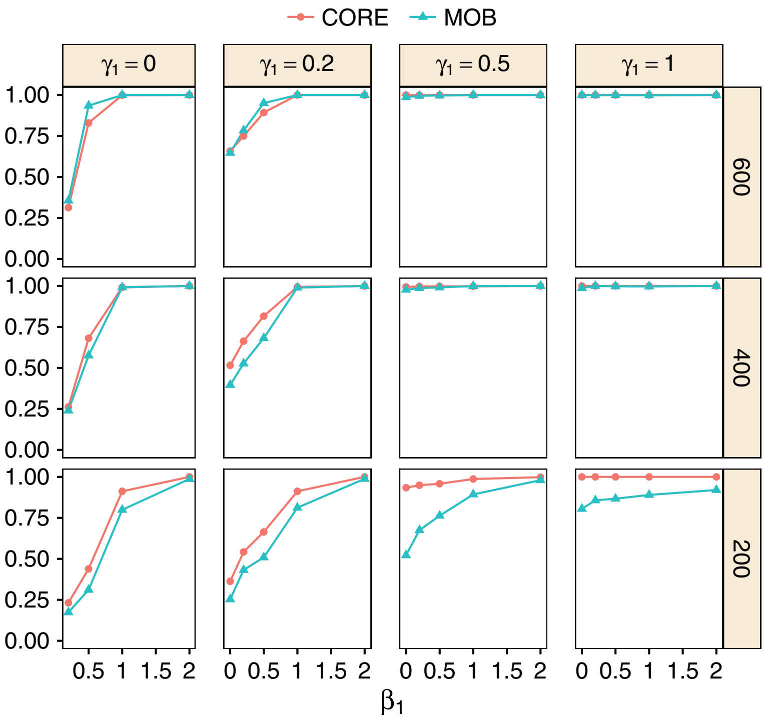

Fig. 4 Estimated probabilities of split variable selection of the two methods under Scenario Q–Q in Table 3 with sample size 200, 400 or 600. The maximum value of the estimated standard error is about 0.016

the zero-inflated negative binomial distribution with parameters μ and π which may depend on X_2 or X_5 or both. Similar simulation designs can be found in the literature [see for example, Loh (2002); Chan and Loh (2004); Zeileis et al. (2008a)]. The R package VGAM (Yee 2015) is used to simulate the distributions. Both CORE and MOB build trees by fitting the zero-inflated negative binomial regression model at each node.

6.1 Selection of split variables

The performance of the two tree methods are investigated under two settings. First, the response and the covariates are generated independently. Here, the tree methods are tested against selection bias. The split selection method of a regression tree is said to be free of selection bias, if it selects each covariate with equal probability when the response is independent of the covariates (Loh 2014). Later, the response which depends on some informative covariates are simulated. The tree methods are evaluated on how often they select the informative covariates. For each run, a random sample of 200, 400 or 600 observations was generated and the experiment was repeated 1000 times. The random samples are forced to split and the relative frequency of each covariate selected by the two methods was then recorded. The maximum value of the estimated standard error is $0.5/\sqrt{1000} \simeq 0.016$.

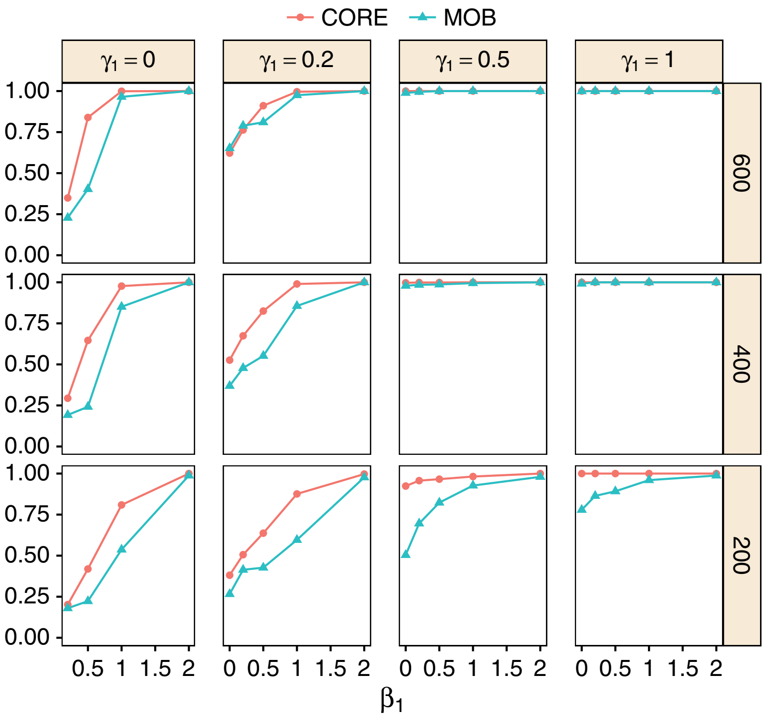

Fig. 5 Estimated probabilities of split variable selection of the two methods under Scenario J–Q in Table 3 with sample size 200, 400 or 600. The maximum value of the estimated standard error is about 0.016

In the first setting, the response variable following a zero-inflated negative binomial regression model (Definition 2) with $\log(\mu) = 1$, $\log(\frac{\pi}{1-\pi}) = -1.5$ and shape parameter $\theta = 10$ is generated while the five covariates follow the distributions given in Table 2. Figure 2 shows the results. The plots indicate that both the CORE method and the MOB method select each covariate with probability about 0.2. Thus, both methods are shown to be free of selection bias.

Next, we simulate some zero-inflated negative binomial models with shape parameter $\theta = 10$ where the response depends on covariate X_2 or X_5 or both. The scenarios given in Table 3 are under consideration. For the first four experiments, $\log(\frac{\pi}{1-\pi})$ is either a jump (J) or quadratic (Q) function of X_5 and $\log(\mu)$ is either a jump or quadratic function of X_2 in the respective scenarios. Because they are non-linear functions of a single covariate, CORE or MOB should split the simulated data. For example, the mean of the count component, μ, depends on X_5 and the probability of zero point mass, π, depends on X_2 under Scenario J–J. Hence, covariate X_2 or X_5 is the informative split variable. When $\gamma_1 = 0$, X_5 is the only informative covariate. So is X_2 when $\beta_1 = 0$. For Scenario INT, both the main effects and the interaction between X_2 and X_5 affect the outcomes. Equations (1) and (2) with the corresponding links in Table 3 are used to simulate the data. We record the number of times each method selects the informative covriate(s) and compute the relative frequency (esti-

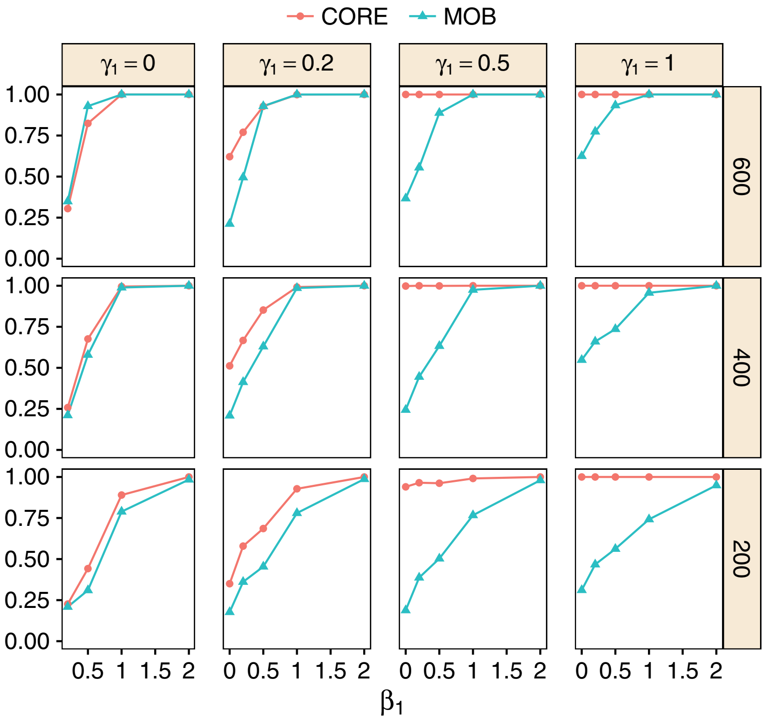

Fig. 6 Estimated probabilities of split variable selection of the two methods under Scenario Q–J in Table 3 with sample size 200, 400 or 600. The maximum value of the estimated standard error is about 0.016

mated probability) of such selection. The results are given in Figs. 3, 4, 5, 6, and 7 respectively.

Under Scenario J–J, Fig. 3 shows that our method has higher chance of selecting the informative covariates than the MOB method for various β_1 and γ_1 values across all three sample sizes. For some β_1 and γ_1 values, the differences are significant. Under Scenario Q–Q, Fig. 4 reveals that two methods are competitive when the sample size is 400 or 600. The CORE method has some edges over the MOB method when γ_1 is 0.5 or 1 and the sample size is 200. Under Scenario J–Q, Fig. 5 shows that our method has either higher or equal chance of selecting the informative covariates. In addition, we find that the CORE method performs significantly better than the MOB method when $\beta_1 = 0.5$ or 1 and $\gamma_1 = 0$ or 0.2 with sample size 200 or 400. Similarly, Under Scenario Q–J, Fig. 6 exhibits that our method is significantly better than the MOB method for various β_1 values when $\gamma_1 = 0.2, 0.5,$ or 1. Under Scenario INT, Fig. 7 demonstrates that our method performs better than the MOB method for small β_1 values across all sample sizes and they are competitive when β_1 is large and the sample size is 400 or 600. When the sample size is 200, the MOB method is better than the CORE method if β_1 is large.

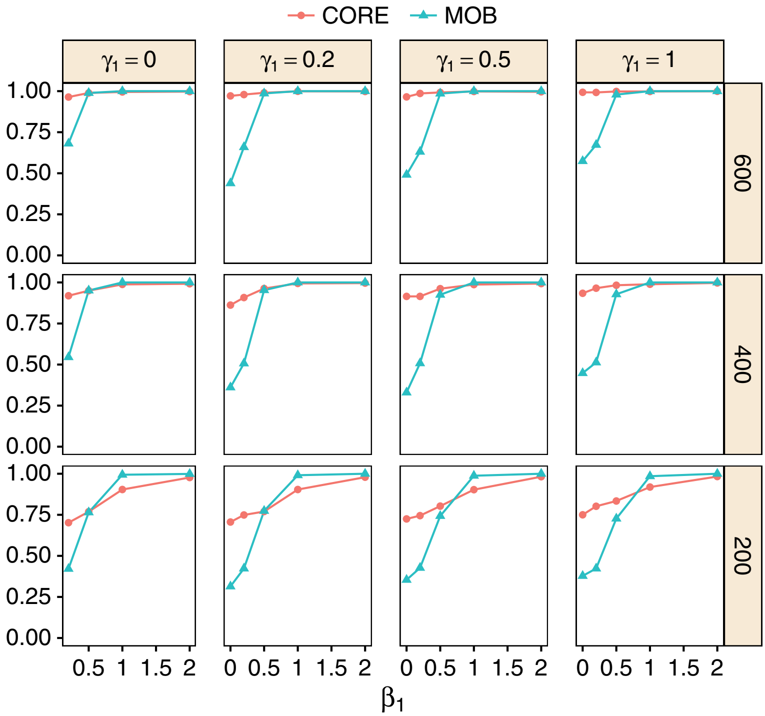

Fig. 7 Estimated probabilities of split variable selection of the two methods under Scenario INT in Table 3 with sample size 200, 400 or 600. The maximum value of the estimated standard error is about 0.016

Table 4 Experiments for prediction accuracy studies of the two tree methods. In each scenario, a zero-inflated negative binomial regression model with parameters π, μ, and shape parameter $\theta = 10$ is generated where $\gamma_1 \in \{0.2, 0.5, 0.8, 1.0\}$ and $\beta_1 \in \{1, 2, 3, 4\}$

Scenario	$\log(\mu)$	$\log(\frac{\pi}{1-\pi})$
A	$1 + \beta_1 I_{\{X_5>0\}}$	-1.5
B	$1 + \beta_1 X_5^2$	-1.5
C	1	$-1.5 + 3\gamma_1 I_{\{X_2>0\}}$
D	1	$-1.5 + \gamma_1 X_2^2$

The distributions of Xs are given in Table 2

6.2 Prediction

In this study, we evaluate our CORE method and the MOB method in terms of prediction accuracy. Two metrics are considered. They are mean square error (MSE) and mean absolute error (MAE) which are defined as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \quad \text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \tag{4}$$

where $\hat{y}_i$ is the predicted value of y_i, $i = 1, \ldots, n$ in the test sample.

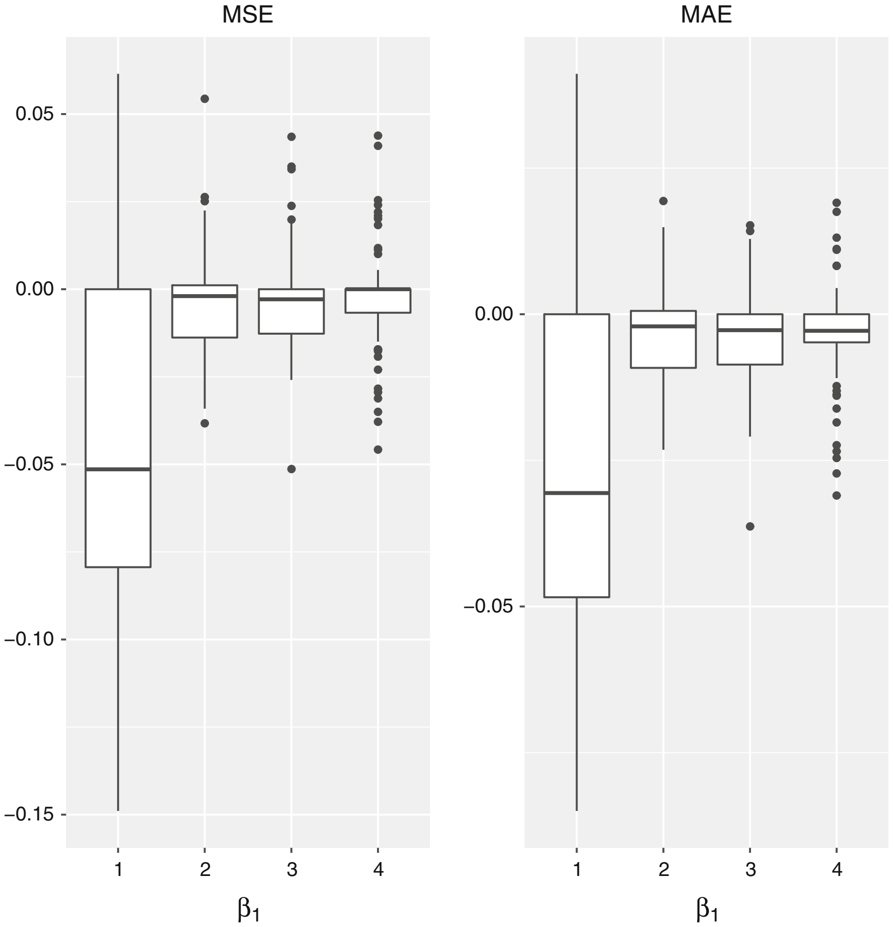

Fig. 8 Box plots of log $\frac{\mathrm{MSE}_C}{\mathrm{MSE}_M}$ (left) and log $\frac{\mathrm{MAE}_C}{\mathrm{MAE}_M}$ (right) under Scenario A in Table 4 with learning sample of 400 and test sample of 1000 observations. MSE_C and MSE_M are the prediction mean square errors of the CORE tree and the MOB tree respectively. MAE_C and MAE_M are the prediction mean absolute errors of the CORE tree and the MOB tree respectively

At first, a learning sample of size 400 was generated under the models listed in Table 4. Both tree methods were then applied to the learning sample and the corresponding trees were obtained. Later, a test sample of size 1000 was generated under the same model which creates the learning sample. Lastly, each resulting tree was applied to the test sample and the predicted values were obtained. This procedure was repeated 100 times and the box plots of log MSE ratio or MAE ratio (CORE vs. MOB) are given in Figs. 8, 9, 10 and 11 respectively.

Under Scenario A or B, only X_5 affects the count part of the regression model, while only X_2 affects the zero part of the regression model under Scenario C or D. Under Scenario A, the MSE or MAE box plots center around zero if the regression coefficient $\beta_1 > 1$. These results show that both methods are competitive in terms of prediction

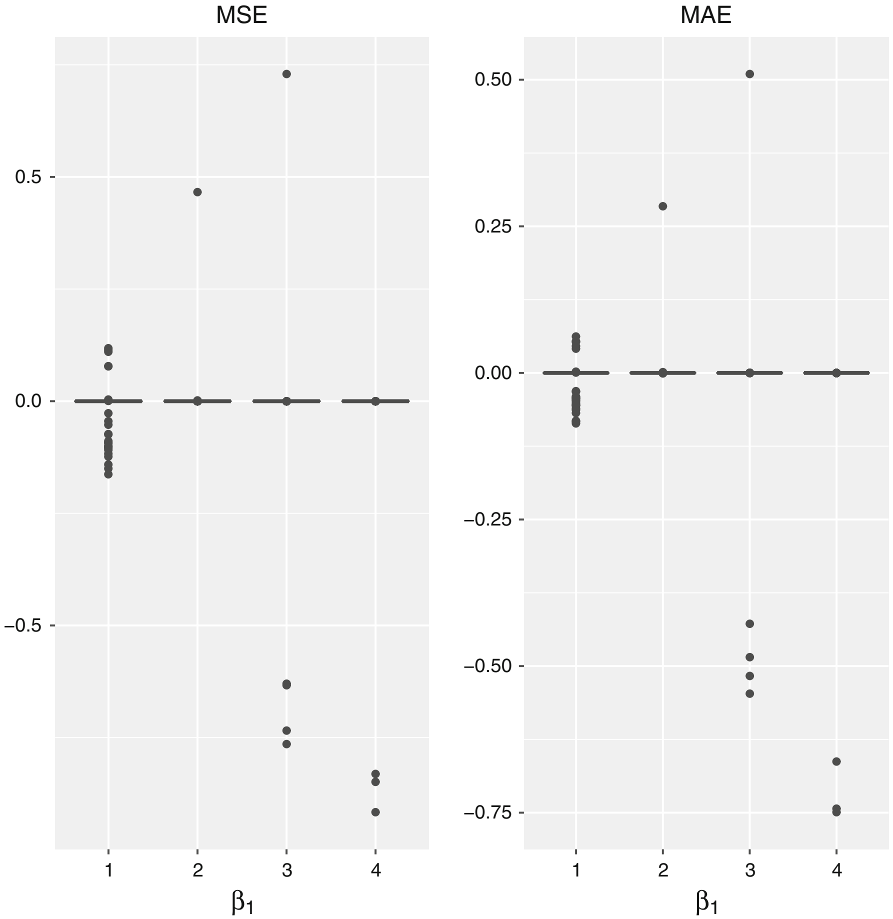

Fig. 9 Box plots of $\log \frac{\text{MSE}_C}{\text{MSE}_M}$ (left) and $\log \frac{\text{MAE}_C}{\text{MAE}_M}$ (right) under Scenario B in Table 4 with learning sample of 400 and test sample of 1000 observations. MSE_C and MSE_M are the prediction mean square errors of the CORE tree and the MOB tree respectively. MAE_C and MAE_M are the prediction mean absolute errors of the CORE tree and the MOB tree respectively

MSE or MAE if $\beta_1 > 1$. When $\beta_1 = 1$, about 75% of the ratios are away from zero. Thus, our method has an edge over the MOB method in prediction accuracy. Under Scenario B, all box plots center around zero and their inter-quartile ranges are rather small. These plots suggest that both methods have similar performances in prediction accuracy. Under Scenario C, the box plots which center around zero if the regression coefficient $\gamma_1 = 0.2$ show that both methods are competitive. If $\gamma_1 = 0.5$, on one hand, the MSE box plot suggests that the MOB method performs better than our method. On the other hand, the MAE box plot indicates that both methods perform similarly. If $\gamma_1 = 0.8$ or 1, the box plots center away from zero and near negative values. These results demonstrate that our method has better accuracy than the MOB method. Under Scenario D, all the box plots center around zero and their inter-quartile

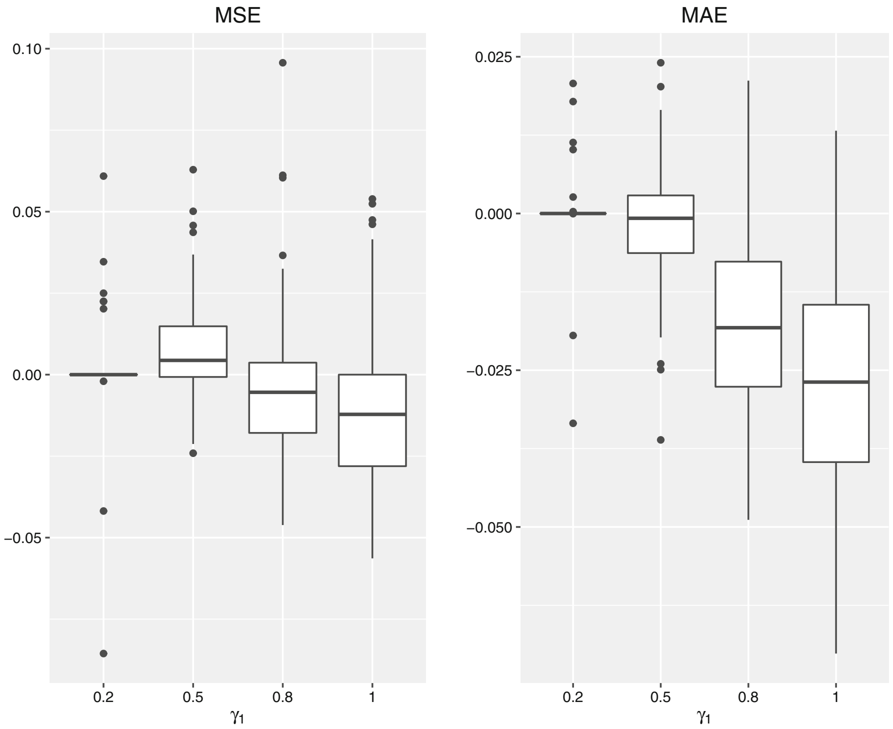

Fig. 10 Box plots of log $\frac{\text{MSE}_C}{\text{MSE}_M}$ (left) and log $\frac{\text{MAE}_C}{\text{MAE}_M}$ (right) under Scenario C in Table 4 with learning sample of 400 and test sample of 1000 observations. MSE_C and MSE_M are the prediction mean square errors of the CORE tree and the MOB tree respectively. MAE_C and MAE_M are the prediction mean absolute errors of the CORE tree and the MOB tree respectively

ranges are rather small. These results suggest that both methods are rather competitive in prediction accuracy.

Exceptional values appear in some of the box plots. However, they are few in each of those plots. As results, they do not affect the overall comparisons between the two methods.

7 Article data

Long (1990) examines the scientific productivity of Ph.D. students in biochemistry by analyzing this data set. We obtained it from the R package AER (Kleiber and Zeileis 2008). It consists of 915 observations where the response is the number of articles published during last 3 years of PhD. The covariates are FEM, female, MAR, married PhD student, KIDS, number of children less than 6 years old, PES, prestige of the graduate program, and MENT, number of articles published by student's mentor. The responses have 275 zero values. Long (1997, Chapter 8) presents a comparison of count regression models on this data and finds that the zero-inflated negative binomial regression model listed in Table 5 gives the best fit.

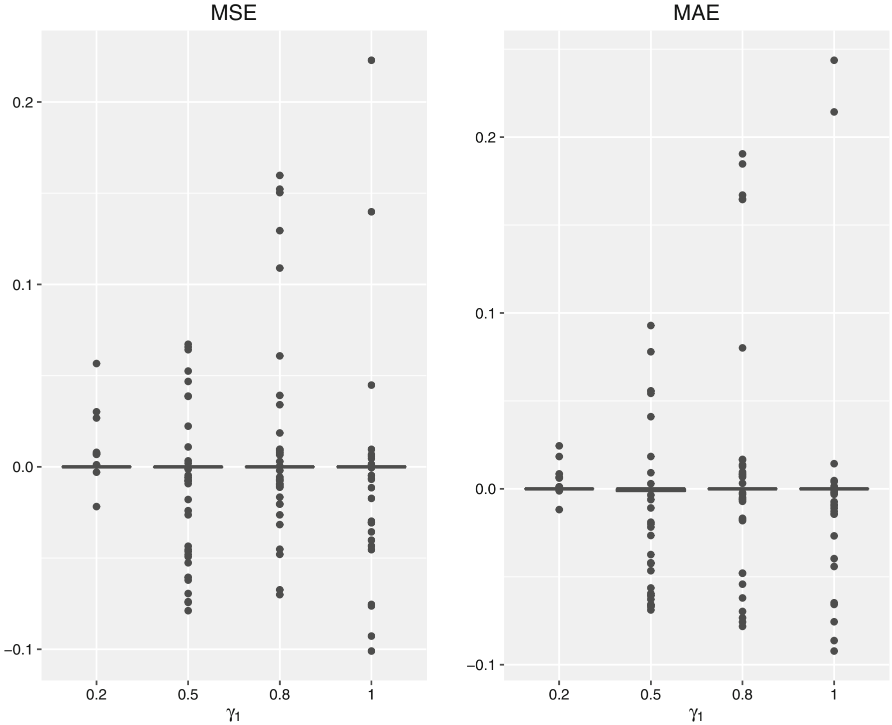

Fig. 11 Box plots of $\log \frac{\text{MSE}_C}{\text{MSE}_M}$ (left) and $\log \frac{\text{MAE}_C}{\text{MAE}_M}$ (right) under Scenario D in Table 4 with learning sample of 400 and test sample of 1000 observations. MSE_C and MSE_M are the average prediction mean square errors of the CORE tree and the MOB tree respectively. MAE_C and MAE_M are the average prediction mean absolute errors of the CORE tree and the MOB tree respectively

We apply the CORE and MOB methods to the data where all the covariates are used to fit a negative binomial regression model at each node and to split nodes. On one hand, the resulting CORE tree is given in Fig. 12 and the coefficient estimates are given in Table 6. Our method splits the whole samples into two groups. The PhD students whose mentors published two or less articles form one group (node 2). The rest students form another group (node 3). This tree model can be treated as a mixture of two different negative binomial regression models. We use ten-fold cross validation to assess prediction accuracy of our model, the zero-inflated negative binomial regression model (ZINB) and later the MOB model. The average of the ten estimated MAE or MSE is reported. The MAE values (CORE vs. ZINB) are 1.303 versus 1.313 while the MSE values are 3.294 versus 3.342. These values show that our tree method performs better in these two prediction categories.

On the other hand, the corresponding MOB tree is given in Fig. 13 and its coefficient estimates are given in Table 7. It uses KIDS twice to split the data. Later, it splits on MENT and then on PES twice. Figure 13 shows that the students with KIDS > 1 are in Node 11 and the students with KIDS $= 1$ are in Node 10. For these two groups, the students are all married. Thus, in both node, the corresponding estimated coeffi-

Table 5 Regression coefficients and the associated z scores of the zero-inflated negative binomial regression model on the article data

	$\log(\mu)$		$\log(\frac{\pi}{1-\pi})$	
	Coefficient	$\|z\|$	Coefficient	$\|z\|$
Intercept	0.417	2.90	– 0.192	0.15
FEM	– 0.196	2.59	0.636	0.75
MAR	0.098	1.16	– 1.499	1.60
KIDS	– 0.152	2.80	0.628	1.42
PES	– 0.001	0.02	– 0.038	0.12
MENT	0.025	7.10	– 0.882	2.79
$\log\theta$	0.976	7.21		

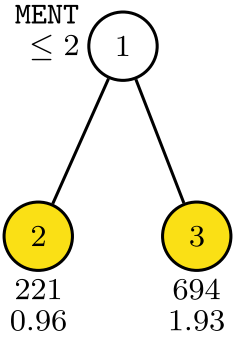

Fig. 12 CORE tree for the article data. At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample size and the expected frequency of the responses are printed below nodes

Table 6 Regression coefficients and the associated z scores of the negative binomial regression model in each terminal node of the CORE tree (Fig. 12)

	Node 2		Node 3	
	Coefficient	$\|z\|$	Coefficient	$\|z\|$
Intercept	– 0.670	1.89	0.556	3.71
FEM	– 0.174	0.91	– 0.221	2.86
MAR	0.699	2.99	0.083	0.96
KIDS	– 0.209	1.66	– 0.167	2.91
PES	0.091	0.99	– 0.021	0.53
MENT	0.033	0.31	0.023	6.64
θ	1.469		2.649	

The estimate of the shape parameter of the negative binomial model, θ, is given for each model

cients of covariate MAR are not available (NA). So is the the corresponding estimated coefficients of covariate KIDS in NODE 10. Similarly, the corresponding estimated coefficient of covariate KIDS is NA in Node 4, 5, 8, or 9. Based on the ten-fold cross validation method, its MAE and MSE values are 1.274 and 3.104 respectively. Although the value is even smaller than that of the CORE tree respectively, it comes with a price. The MOB tree has six terminal nodes. This could make interpreting the tree difficult.

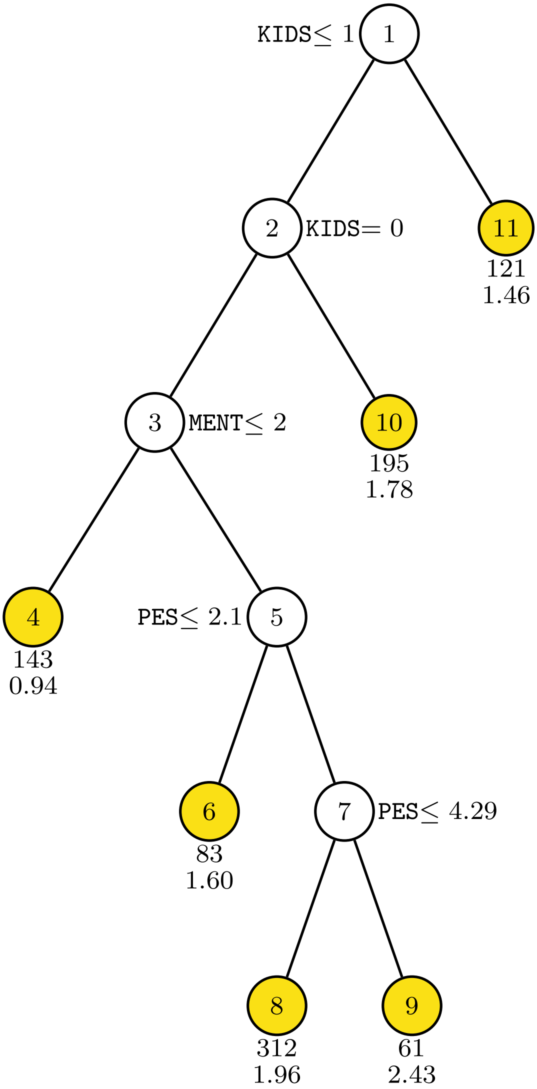

Fig. 13 MOB tree for the article data. At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample size and the expected frequency of the responses are printed below nodes

8 Conclusion

In this paper, we propose a regression tree method for count data, CORE. At each node, it provides flexible fitting models to account for over-dispersion and/or excess zeros. It can fit the classical Poisson, negative binomial, hurdle or zero-inflated regression model to the sample in each node. It then uses the partial score residuals derived from the chosen model to determine the split variable. The related maximum log likelihood values are used to choose the split sets and to select the right-size tree. Our split selection method is free of selection bias. Compared with the MOB method, the simulation results show that our method has better performance in selecting infor-

Table 7 Regression coefficients and the associated z scores of the negative binomial regression model in each terminal node of the MOB tree (Fig. 13)

	Node 4		Node 6		Node 8	
	Coefficient	$\|z\|$	Coefficient	$\|z\|$	Coefficient	$\|z\|$
Intercept	− 0.619	1.58	0.877	1.08	1.013	3.46
FEM	− 0.199	0.93	− 0.464	1.76	− 0.151	1.45
MAR	0.667	2.97	0.331	1.21	0.019	0.85
KIDS	NA	NA	NA	NA	NA	NA
PES	0.099	0.93	− 0.492	1.15	− 0.151	1.84
MENT	− 0.024	0.19	0.047	3.14	0.021	3.42
θ	2.240		1.712		3.388	

	Node 9		Node 10		Node 11	
	Coefficient	$\|z\|$	Coefficient	$\|z\|$	Coefficient	$\|z\|$
Intercept	11.862	2.47	0.252	0.99	1.956	2.52
FEM	− 0.127	0.67	− 0.155	0.88	− 0.518	1.98
MAR	− 0.140	0.71	NA	NA	NA	NA
KIDS	NA	NA	NA	NA	− 0.514	1.54
PES	− 2.421	2.29	0.021	0.26	− 0.253	2.39
MENT	0.010	1.15	0.025	4.35	0.032	4.02
θ	10.3		1.903		2.90	

The estimate of the shape parameter of the negative binomial model, θ, is given for each model

mative covariates under various settings. The prediction results also reveal that our method is competitive with or better than the MOB method in most studied cases. The usefulness of CORE is further demonstrated in two real data studies. Our tree method is implemented in R and its software can be downloaded from http://discovery.ccu.edu.tw/Site/nu26786/.

Some future works are needed to make CODE more flexible. At each node, CODE considers four count regression models: the Poisson, negative binomial, hurdle and zero-inflated models. Other parametric models like those mentioned in Cameron and Trivedi (2013) can be considered as well. Real data may contain missing values. The CODE algorithm needs modifications in order to accommodate such values.

Acknowledgements The authors thank Dr. Chen-Hsin Chen of Academia Sinica for his insightful comments and suggestions during our discussions. We are very grateful to the two reviewers for the helpful comments. This research is partly supported by Taiwan MOST grant 105-2118-M-194-001 and Domestic Visiting Scholar Program of Academica Sinica, Taiwan, contract 106-1-1-06-18.

References

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Wadsworth and Brooks, Monterey

Cameron AC, Trivedi PK (2013) Regression analysis of count data, 2nd edn. Cambridge University Press, New York

Chan KY, Loh WY (2004) LOTUS: an algorithm for building accurate and comprehensible logistic regression trees. J Comput Graph Stat 13(4):826–852

Choi Y, Ahn H, Chen JJ (2005) Regression trees for analysis of count data with extra Poisson variation. Comput Stat Data Anal 49(3):893–915

Ciampi A (1991) Generalized regression trees. Comput Stat Data Anal 12(1):57–78

Comizzoli RB, Landwehr JM, Sinclair JD (1990) Robust materials and processes: key to reliability. AT & T Tech J 69(6):113–128

Hothorn T, Zeileis A (2015) partykit: a modular toolkit for recursive partytioning in R. J Mach Learn Res 16:3905–3909

Kleiber C, Zeileis A (2008) Applied econometrics with R. Springer, New York

Kuhn M, Johnson K (2013) Applied predictive modeling. Springer, New York

Lambert D (1992) Zero-inflated Poisson regression, with an application to defects in manufacturing. Technometrics 34(1):1–14

Lee SK, Jin S (2006) Decision tree approaches for zero inflated count data. J Appl Stat 33(8):853–864

Loh WY (2002) Regression tree with unbiased variable selection and interaction detection. Stat Sin 12(2):361–386

Loh WY (2006) Regression tree models for designed experiments. In: Rojo J (ed) Second E. L. Lehmann Symposium, IMS Lecture Notes-Monograph Series, vol 49, pp 210–228

Loh WY (2009) Improving the precision of classification trees. Ann Appl Stat 3(4):1710–1737

Loh WY (2014) Fifty years of classification and regression trees. Int Stat Rev 82(3):329–348

Long JS (1990) The origins of sex differences in science. Soc Forces 68(4):1297–1316

Long JS (1997) Regression models for categorical and limited dependent variables. Sage Publications, Thousand Oaks

Mullahy J (1986) Specification and testing of some modified count data models. J Econ 33(3):341–365

Neelon B, O'Malley AJ, Smith VA (2016) Modeling zero-modified count and semicontinuous data in health services research part 1: background and overview. Stat Med 35(27):5070–5093

Rusch T, Zeileis A (2013) Gaining insight with recursive partitioning of generalized linear models. J Stat Comput Simul 83(7):1301–1315

Wilson EB, Hilferty MM (1931) The distribution of chi-square. Proc Natl Acad Sci USA 17:684–688

Yee TW (2015) Vector generalized linear and additive models: with an implementation in R. Springer, New York

Zeileis A, Hornik K (2007) Generalized M-fluctuation tests for parameter instability. Statistica Neerlandica 61(4):488–508

Zeileis A, Hothorn T, Hornik K (2008a) Model-based recursive partitioning. J Comput Graph Stat 17(2):492–514

Zeileis A, Kleiber C, Jackman S (2008b) Regression models for count data in R. J Stat Softw Articles 27(8):1–25

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.