Supplementary Material
A Tweedie Compound Poisson Model in Reproducing Kernel Hilbert Space

Yi Lian[*], Archer Yi Yang[†], Boxiang Wang[‡], Peng Shi[§], Robert William Platt[¶]

A Algorithms

A.1 Bisection Line-search for BFGS

This line-search is performed in each (inverse) BFGS update iteration. It aims to find an appropriate positive step size t that satisfies the Wolfe conditions in (17).

[*]Department of Epidemiology, Biostatistics and Occupational Health, McGill University
[†]Department of Mathematics and Statistics, McGill University (archer.yang@mcgill.ca)
[‡]Department of Statistics and Actuarial Science, University of Iowa
[§]Risk and Insurance Department, Wisconsin School of Business, University of Wisconsin-Madison
[¶]Corresponding author, Department of Epidemiology, Biostatistics and Occupational Health, McGill University

Algorithm S1: Bisection line-search for the (inverse) BFGS

```
Input: α, p
Output: t
Constants: c_1 = 10^-4, c_2 = 0.9, a = 0
Initialization: t = 1, phase = A, accept = False;
repeat phase A
    if Condition 1 holds then
        if Condition 2 holds then
            accept = True
        else
            t = 2t
        end
    else
        phase = B
        exit;
    end
until accept;
if phase = B then
    b = t
    repeat phase B
        t_old = t
        t = (a + b)/2
        if t_old = t then
            cannot find proper t
            exit;
            /* exit BFGS                                  */
            /* switch to GD                               */
        end
        if Condition 1 holds then
            if Condition 2 holds then
                accept = True
            else
                a = t
            end
        else
            b = t
        end
    until accept;
end
```

A.2 Backtracking Line-search for Gradient Descent

This line-search is performed in each gradient descent update iteration. It aims to find an appropriate positive step size t that satisfies the Armijo-Goldstein condition

$$g(\boldsymbol{\xi} - t\nabla g(\boldsymbol{\xi})) \leq g(\boldsymbol{\xi}) - ct\|\nabla g(\boldsymbol{\xi})\|_2^2$$

where $\boldsymbol{\xi}$ is the parameter of interest ($\boldsymbol{\alpha}$ or $\mathbf{w}$ in our case) and $c \in (0, 1/2]$ is some constant.

Algorithm S2: Backtracking line-search for gradient descent

```
Input: ξ
Output: t
Constants: c = 0.5
Initialization: t = 1, accept = False;
repeat
    if g(ξ − t∇g(ξ)) ≤ g(ξ) − ct‖∇g(ξ)‖₂² then
        accept = True
    else
        t = 0.9t
    end
until accept;
```

A.3 Gradient Descent Update for the Weights in the SKtweedie

Algorithm S3: Gradient descent for weight

Input: $\mathbf{X}, \mathbf{y}, \lambda_1, \lambda_2, \boldsymbol{\alpha}^{(m)}, \mathbf{w}^{(m)}$
Output: $\mathbf{w}^{(m+1)}$

Initialization: $k = 0, \mathbf{w}^{(m,0)} = \mathbf{w}^{(m)}$;
repeat gradient descent loop
  Generate new kernel matrix $\mathbf{K}(\mathbf{w}^{(m,k)})$ as defined in (12)
  call Algo. S2 to find step size $t^{(m,k)}$
  for $j = 1, \ldots, p$ **do**
    Compute $w_j^{(m,k+1)}$ using (18)
  end
  $k := k + 1$
  if $\mathbf{w}^{(m,k+1)} = \mathbf{0}_p$ **then** exit;
until *convergence*;
$\mathbf{w}^{(m+1)} = \mathbf{w}^{(m,k)}$

B Fitting the Ktweedie Model with an Intercept

This section discusses the implementation details when there is an intercept term in the model. Denote by $g(\alpha_0, \boldsymbol{\alpha})$ the objective function in (10). It is convex in $(\alpha_0, \boldsymbol{\alpha})$, which allows convenient alternating minimization. Based on Algorithm 1, after updating $\boldsymbol{\alpha}^{(k)}$ to $\boldsymbol{\alpha}^{(k+1)}$ with α_0 fixed at $\alpha_0^{(k)}$ in each iteration k (Line 6), we update $\alpha_0^{(k)}$ to $\alpha_0^{(k+1)}$. This can be done by solving the equation $\frac{\partial g(\alpha_0, \boldsymbol{\alpha}^{(k+1)})}{\partial \alpha_0} = 0$ analytically,

$$\alpha_0^{(k+1)} \leftarrow \log \frac{\sum_{i=1}^n y_i \exp[(1-\rho)\mathbf{K}_i^\top \boldsymbol{\alpha}^{(k+1)}]}{\sum_{i=1}^n \exp[(2-\rho)\mathbf{K}_i^\top \boldsymbol{\alpha}^{(k+1)}]}.$$

C Proof of Theorem 1

Proof. According to Theorem 6.5 (Nocedal and Wright, 2006), in order to show **the global convergence of BFGS** in our algorithm, we only need to check the following two conditions (Assumption 6.1 Nocedal and Wright, 2006) are satisfied:

1. The objective function g is twice continuously differentiable.

2. There exist positive constants m and M such that, for all $\boldsymbol{\alpha}$,

$$m\mathbf{I}_n \preceq \nabla^2 g\left(\boldsymbol{\alpha}\right) \preceq M\mathbf{I}_n.$$

   where $\mathbf{I}_n$ is an $n \times n$ identity matrix.

Since Algorithm 1 is descending along its iterations thus we can restrict the domain of $\boldsymbol{\alpha}$ to the sublevel set $\mathcal{L}_0 = \left\{\boldsymbol{\alpha} \in \mathbb{R}^n : g(\boldsymbol{\alpha}) \leq g(\boldsymbol{\alpha}^{(0)})\right\}$. Since g is a convex function, set $\mathcal{L}_0$ is convex compact. Without loss of generality, assume not all y_i's are zero. Define $\tau_i = \mathbf{K}_i^\top \boldsymbol{\alpha}$ for $i = 1, \ldots, n$. It follows that the set

$$\mathcal{C}_0 = \left\{\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)^\top : \boldsymbol{\alpha} \in \mathcal{L}_0\right\}$$

is convex compact. Therefore for all $\boldsymbol{\alpha} \in \mathcal{L}_0$, η_i is bounded by $\eta_{\max}$, where

$$\eta_{\max} = \max_{1 \leq i \leq n} \sup_{\boldsymbol{\alpha} \in \mathcal{L}_0} |\eta_i| < \infty.$$

Also y_i's are bounded by $v_{\max} = \max_{1 \leq i \leq n} v_i$ and $y_{\max} = \max_{1 \leq i \leq n} y_i$. Let

$$\bar{w}_i = v_i \left((\rho - 1) y_i e^{(1-\rho)\tau_i} + (2 - \rho) e^{(2-\rho)\tau_i}\right)$$

Note that $\bar{w}_i$ is bounded by

$$\max_{1 \leq i \leq n} \sup_{\boldsymbol{\alpha} \in \mathcal{L}_0} |\bar{w}_i| \leq v_{\max} \left(y_{\max}(\rho - 1) e^{(\rho-1)\tau_{\max}} + (2 - \rho) e^{(2-\rho)\tau_{\max}}\right) \equiv w_{\max}.$$

We can see that

$$\begin{aligned}\nabla^2 g\left(\boldsymbol{\alpha}\right) &= \mathbf{K}\mathrm{diag}\left[\bar{w}_1, \bar{w}_2, \ldots, \bar{w}_n\right]\mathbf{K} + \lambda\mathbf{K} \\ &\preceq (w_{\max}\Lambda_{\max}(\mathbf{KK}) + \Lambda_{\max}(\mathbf{K}))\mathbf{I}_n, \qquad \forall \boldsymbol{\alpha} \in \mathcal{L}_0.\end{aligned}$$

where $\Lambda_{\max}(\mathbf{A})$ represents the largest eigenvalue of matrix $\mathbf{A}$. Thus $g\left(\boldsymbol{\alpha}\right)$ is strongly smooth on the sublevel set $\mathcal{L}_0$. We can also show that $g\left(\boldsymbol{\alpha}\right)$ is strongly convex on $\mathcal{L}_0$. It can be shown that $\bar{w}_i$ can be lower-bounded on $\mathcal{L}_0$,

$$\bar{w}_i \geq \left(\frac{\rho-1}{2-\rho}\right)^{3-2\rho} v_i \left(y_i\right)^{2-\rho} I\left(y_i > 0\right) + (2-\rho)e^{-(2-\rho)\eta_{\max}} I\left(y_i = 0\right) > 0$$

for all $\boldsymbol{\alpha} \in \mathcal{L}_0$ and $i = 1, \ldots, n$. Let

$$w_{\min} = \min\left\{\left(\frac{\rho-1}{2-\rho}\right)^{3-2\rho} \min_{i:y_i>0} w_i \left(y_i\right)^{2-\rho}, (2-\rho)e^{-(2-\rho)\eta_{\max}}\right\}.$$

We see that $\bar{w}_i \geq w_{\min} > 0$. Therefore

$$\begin{aligned}\nabla^2 g\left(\boldsymbol{\alpha}\right) &= \mathbf{K}\mathrm{diag}\left[\bar{w}_1, \bar{w}_2, \ldots, \bar{w}_n\right]\mathbf{K} + \lambda\mathbf{K} \\ &\succeq (w_{\min}\Lambda_{\min}(\mathbf{KK}) + \Lambda_{\min}(\mathbf{K}))\mathbf{I}_n, \qquad \forall \boldsymbol{\alpha} \in \mathbb{R}^n.\end{aligned}$$

This shows that $g(\boldsymbol{\alpha})$ is strongly convex. We have proved that Assumption 6.1 in Theorem 6.5 (Nocedal and Wright, 2006) holds so that Algorithm 1 has global convergence.

By Theorem 6.6 (Nocedal and Wright, 2006), in order to show that the update $\boldsymbol{\alpha}^{(k)}$ generated by Algorithm 1 converges to $\boldsymbol{\alpha}^*$ at a superlinear rate, we only need to show that g is twice continuously differentiable and that the Hessian matrix $\nabla^2 g$ is Lipschitz continuous (Assumption 6.2 Nocedal and Wright, 2006), i.e. for all $\boldsymbol{\alpha}, \boldsymbol{\alpha}' \in \mathrm{dom}g$, there exists a positive constant L such that,

$$\left\|\nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}')\right\|_2 \leq L \left\|\boldsymbol{\alpha} - \boldsymbol{\alpha}'\right\|_2,$$

where the norm applied to the matrix is the spectral norm.

We consider a vector-valued function $h(t) : \mathbb{R} \to \mathbb{R}^n$ satisfying $h_{\mathbf{b}}(t) = \mathbf{b}^\top \nabla^2 f(\boldsymbol{\alpha} + t(\boldsymbol{\alpha}' - \boldsymbol{\alpha}))$, then by the mean value theorem

$$\begin{aligned}
\mathbf{b}^\top[\nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}')] &= \frac{h_{\mathbf{b}}(1) - h_{\mathbf{b}}(0)}{1-0} \\
&= h'_{\mathbf{b}}(\tilde{t}) \qquad \text{(mean value theorem, } \tilde{t} \in (0,1)\text{)} \\
&= \begin{bmatrix} \sum_i \sum_j \frac{\partial^3 g(\widetilde{\boldsymbol{\alpha}})}{\partial \alpha_1 \partial \alpha_i \partial \alpha_j} b_i(\alpha'_j - \alpha_j) \\ \vdots \\ \sum_i \sum_j \frac{\partial^3 g(\widetilde{\boldsymbol{\alpha}})}{\partial \alpha_n \partial \alpha_i \partial \alpha_j} b_i(\alpha'_j - \alpha_j) \end{bmatrix}. \qquad (\widetilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha} + \tilde{t}(\boldsymbol{\alpha}' - \boldsymbol{\alpha}))
\end{aligned} \tag{1}$$

In the sublevel set $\mathcal{L}_0$, the values of third derivatives of g in (1) can be upper-bounded

$$\left| \frac{\partial^3 g(\widetilde{\boldsymbol{\alpha}})}{\partial \alpha_1 \partial \alpha_i \partial \alpha_j} \right| \le D, \tag{2}$$

where $D > 0$ is a constant. Therefore the L_2 norm of the vector $\mathbf{b}^\top[\nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}')]$ can also be upper-bounded

$$\begin{aligned}
\|\mathbf{b}^\top[\nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}')]\|_2 &\le D\sqrt{n} \Big| \sum_i \sum_j b_i(\alpha'_j - \alpha_j) \Big| \\
&\le D\sqrt{n} \cdot n \|\mathbf{b}\|_2 \|\boldsymbol{\alpha}' - \boldsymbol{\alpha}\|_2.
\end{aligned}$$

The above inequality indicates that $\nabla^2 g$ is Lipschitz continuous, since that

$$\begin{aligned}
\left\| \nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}') \right\|_2 &= \max_{\|\mathbf{b}\|_2 = 1} \|\mathbf{b}^\top[\nabla^2 g(\boldsymbol{\alpha}) - \nabla^2 g(\boldsymbol{\alpha}')]\|_2 \\
&\le \max_{\|\mathbf{b}\|_2 = 1} D\sqrt{n} \cdot n \|\mathbf{b}\|_2 \|\boldsymbol{\alpha}' - \boldsymbol{\alpha}\|_2 \\
&= D\sqrt{n} \cdot n \|\boldsymbol{\alpha}' - \boldsymbol{\alpha}\|_2,
\end{aligned}$$

where the first line follows by the definition of the spectral norm. Therefore Assumption 6.1 in Theorem 6.5 (Nocedal and Wright, 2006) holds. □

D The Derivative of the SKtweedie Objective Function

The objective function is

$$
\begin{aligned}
g(\boldsymbol{\alpha}, \mathbf{w}) &= l_1 + l_2 + p_1 + p_2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i \exp\left[-(\rho-1)\mathbf{K}(\mathbf{w})_i^\top \boldsymbol{\alpha}\right]}{\rho-1}\right) \ldots\ldots\ldots\ldots\ldots\ldots (l_1) \\
&+ \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\exp\left[(2-\rho)\mathbf{K}(\mathbf{w})_i^\top \boldsymbol{\alpha}\right]}{2-\rho}\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (l_2) \\
&+ \lambda_1 \boldsymbol{\alpha}^\top \mathbf{K}(\mathbf{w}) \boldsymbol{\alpha} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (p_1) \\
&+ \lambda_2 \mathbf{1}^\top \mathbf{w} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (p_2) \\
\text{s.t. } & w_j \in [0,1],\ j = 1, \ldots, p,
\end{aligned}
$$

where

$$
\mathbf{K}(\mathbf{w}) = \begin{bmatrix} \mathbf{K}(\mathbf{w})_1 \\ \mathbf{K}(\mathbf{w})_2 \\ \vdots \\ \mathbf{K}(\mathbf{w})_n \end{bmatrix} = \begin{bmatrix} \mathbf{K}(\mathbf{w})_{11} & \mathbf{K}(\mathbf{w})_{12} & \cdots & \mathbf{K}(\mathbf{w})_{1n} \\ \mathbf{K}(\mathbf{w})_{21} & \mathbf{K}(\mathbf{w})_{22} & \cdots & \mathbf{K}(\mathbf{w})_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}(\mathbf{w})_{n1} & \mathbf{K}(\mathbf{w})_{n2} & \cdots & \mathbf{K}(\mathbf{w})_{nn} \end{bmatrix}
$$

$$
= \begin{bmatrix} K(\mathbf{w}\odot\mathbf{x}_1, \mathbf{w}\odot\mathbf{x}_1) & K(\mathbf{w}\odot\mathbf{x}_1, \mathbf{w}\odot\mathbf{x}_2) & \cdots & K(\mathbf{w}\odot\mathbf{x}_1, \mathbf{w}\odot\mathbf{x}_n) \\ K(\mathbf{w}\odot\mathbf{x}_2, \mathbf{w}\odot\mathbf{x}_1) & K(\mathbf{w}\odot\mathbf{x}_2, \mathbf{w}\odot\mathbf{x}_2) & \cdots & K(\mathbf{w}\odot\mathbf{x}_2, \mathbf{w}\odot\mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ K(\mathbf{w}\odot\mathbf{x}_n, \mathbf{w}\odot\mathbf{x}_1) & K(\mathbf{w}\odot\mathbf{x}_n, \mathbf{w}\odot\mathbf{x}_2) & \cdots & K(\mathbf{w}\odot\mathbf{x}_n, \mathbf{w}\odot\mathbf{x}_n) \end{bmatrix},
$$

and $K(\cdot,\cdot)$ is the RBF kernel function with tuning parameter σ. For $i, j = 1, 2, \ldots, n$,

$$
\mathbf{K}(\mathbf{w})_{ij} = k(\mathbf{w}\odot\mathbf{x}_i, \mathbf{w}\odot\mathbf{x}_j) = \exp(-\sigma \cdot \|\mathbf{w}\odot\mathbf{x}_i - \mathbf{w}\odot\mathbf{x}_j\|_2^2).
$$

For clarity, divide the objective function into four parts $g(\boldsymbol{\alpha}, \mathbf{w}) = l_1 + l_2 + p_1 + p_2$ and derive individually. First, we take derivative of l_1 with respect to $\mathbf{w}$,

$$\frac{\partial l_1}{\partial \mathbf{w}} = \frac{1}{n} \sum_{i=1}^{n} \frac{\partial l_1}{\partial \mathbf{K}(\mathbf{w})_i} \cdot \frac{\partial \mathbf{K}(\mathbf{w})_i}{\partial \mathbf{w}},$$

where

$$\begin{aligned} \frac{\partial l_1}{\partial \mathbf{K}(\mathbf{w})_i} &= -y_i \exp\left[-(\rho - 1)\mathbf{K}(\mathbf{w})_i^\top \boldsymbol{\alpha}\right] \cdot \boldsymbol{\alpha} \\ &= \eta_i \cdot \boldsymbol{\alpha} \in \mathbb{R}^n, \end{aligned}$$

with $\eta_i = -y_i \exp\left[-(\rho - 1)\mathbf{K}(\mathbf{w})_i^\top \boldsymbol{\alpha}\right]$ is a scalar, and

$$\frac{\partial \mathbf{K}(\mathbf{w})_i}{\partial \mathbf{w}} = \frac{\partial \left[\mathbf{K}(\mathbf{w})_{i1}, \mathbf{K}(\mathbf{w})_{i2}, \ldots, \mathbf{K}(\mathbf{w})_{in}\right]}{\partial \mathbf{w}} \in \mathbb{R}^{n \times p},$$

with

$$\begin{aligned} \frac{\partial \mathbf{K}(\mathbf{w})_{ij}}{\partial \mathbf{w}} &= \frac{\partial k(\mathbf{w} \odot \mathbf{x}_i, \mathbf{w} \odot \mathbf{x}_j)}{\partial \mathbf{w}} \\ &= \frac{\partial \exp(-\sigma \cdot \|\mathbf{w} \odot \mathbf{x}_i - \mathbf{w} \odot \mathbf{x}_j\|_2^2)}{\partial \mathbf{w}} \\ &= \exp(-\sigma \cdot \|\mathbf{w} \odot \mathbf{x}_i - \mathbf{w} \odot \mathbf{x}_j\|_2^2) \cdot (-2\sigma) \cdot (\mathbf{x}_i - \mathbf{x}_j) \odot (\mathbf{x}_i - \mathbf{x}_j) \odot \mathbf{w} \\ &= c_{ij} \cdot (\mathbf{x}_i - \mathbf{x}_j) \odot (\mathbf{x}_i - \mathbf{x}_j) \odot \mathbf{w}, \end{aligned}$$

for the scalar $c_{ij} = -2\sigma \cdot \exp(-\sigma \cdot \|\mathbf{w} \odot \mathbf{x}_i - \mathbf{w} \odot \mathbf{x}_j\|_2^2)$. Therefore,

$$\frac{\partial \mathbf{K}(\mathbf{w})_i}{\partial \mathbf{w}} = \begin{bmatrix} c_{i1} \cdot (\mathbf{x}_i - \mathbf{x}_1) \odot (\mathbf{x}_i - \mathbf{x}_1) \odot \mathbf{w} \\ c_{i2} \cdot (\mathbf{x}_i - \mathbf{x}_2) \odot (\mathbf{x}_i - \mathbf{x}_2) \odot \mathbf{w} \\ \vdots \\ c_{in} \cdot (\mathbf{x}_i - \mathbf{x}_n) \odot (\mathbf{x}_i - \mathbf{x}_n) \odot \mathbf{w} \end{bmatrix}.$$

Put it together,

$$\frac{\partial \ell_1}{\partial \mathbf{w}} = \frac{1}{n} \sum_{i=1}^{n} \eta_i \cdot \boldsymbol{\alpha}^\top \cdot \begin{bmatrix} c_{i1} \cdot (\mathbf{x}_i - \mathbf{x}_1) \odot (\mathbf{x}_i - \mathbf{x}_1) \odot \mathbf{w} \\ c_{i2} \cdot (\mathbf{x}_i - \mathbf{x}_2) \odot (\mathbf{x}_i - \mathbf{x}_2) \odot \mathbf{w} \\ \vdots \\ c_{in} \cdot (\mathbf{x}_i - \mathbf{x}_n) \odot (\mathbf{x}_i - \mathbf{x}_n) \odot \mathbf{w} \end{bmatrix} \in \mathbb{R}^p.$$

Next, we derive l_2. Similar to the above,

$$\frac{\partial l_2}{\partial \mathrm{w}} = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial l_2}{\partial \mathbf{K}(\mathbf{w})_i} \cdot \frac{\partial \mathbf{K}(\mathbf{w})_i}{\partial \mathbf{w}}$$

$$= \frac{1}{n}\sum_{i=1}^{n} \zeta_i \cdot \boldsymbol{\alpha}^\top \cdot \begin{bmatrix} c_{i1} \cdot (\mathbf{x}_i - \mathbf{x}_1) \odot (\mathbf{x}_i - \mathbf{x}_1) \odot \mathbf{w} \\ c_{i2} \cdot (\mathbf{x}_i - \mathbf{x}_2) \odot (\mathbf{x}_i - \mathbf{x}_2) \odot \mathbf{w} \\ \vdots \\ c_{in} \cdot (\mathbf{x}_i - \mathbf{x}_n) \odot (\mathbf{x}_i - \mathbf{x}_n) \odot \mathbf{w} \end{bmatrix},$$

where $\zeta_i = \exp\left[(2-\rho)\mathbf{K}(\mathbf{w})_i^\top \boldsymbol{\alpha}\right], i = 1, 2, \ldots, n$.

Next, take the derivative of the first penalty p_1 w.r.t. $\mathbf{w}$,

$$\frac{\partial p_1}{\partial \mathbf{w}} = \lambda_1 \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial p_1}{\partial \mathbf{K}(\mathbf{w})_{ij}} \cdot \frac{\partial \mathbf{K}(\mathbf{w})_{ij}}{\partial \mathbf{w}}$$

$$= \lambda_1 \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j \frac{\partial \mathbf{K}(\mathbf{w})_{ij}}{\partial \mathbf{w}}$$

$$= \lambda_1 \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j c_{ij} \cdot (\mathbf{x}_i - \mathbf{x}_j) \odot (\mathbf{x}_i - \mathbf{x}_j) \odot \mathbf{w}.$$

Finally, $\partial p_2 / \partial \mathbf{w}$ has the following form,

$$\frac{\partial p_2}{\partial \mathbf{w}} = \lambda_2.$$

Note that the gradient is scaled by the weights except for the last term, thus $\frac{\partial g(\boldsymbol{\alpha}, \mathbf{w})}{\partial w_j} = \lambda_2$, for all $w_j = 0$.

E Parameter Orthogonality

Following (5), $g(y|\mu,\phi,\rho)$ is the density function, for y, we have $\int g(y|\mu,\phi,\rho)dy = 1$. Therefore

$$\begin{aligned}
0 &= \frac{\partial}{\partial\mu}\int g(y|\mu,\phi,\rho)dy \\
&= \int \frac{g(y|\mu,\phi,\rho)}{g(y|\mu,\phi,\rho)}\frac{\partial g(y|\mu,\phi,\rho)}{\partial\mu}dy \\
&= \int g(y|\mu,\phi,\rho)\frac{\partial \log g(y|\mu,\phi,\rho)}{\partial\mu}dy \\
&= \mathbb{E}_Y\left[\frac{\partial \log g(y|\mu,\phi,\rho)}{\partial\mu}\right].
\end{aligned}$$

Since

$$g(y|\mu,\phi,\rho) = a(y,\phi,\rho)\exp\left\{\frac{1}{\phi}\left(\frac{y\mu^{1-\rho}}{1-\rho} - \frac{\mu^{2-\rho}}{2-\rho}\right)\right\},$$

the density satisfies

$$\frac{\partial \log g\left(y|\mu,\rho,\phi\right)}{\partial\mu} = \frac{y-\mu}{\phi\mu^{\rho}}.$$

Therefore

$$\begin{aligned}
\mathbb{E}\left[\frac{\partial^2 \log g(y|\mu,\phi,\rho)}{\partial\mu\partial\phi}\right] &= \mathbb{E}\left[\frac{\partial}{\partial\phi}\left(\frac{y-\mu}{\phi\mu^{\rho}}\right)\right] \\
&= \mathbb{E}\left[-\frac{1}{\phi^2}\cdot\frac{y-\mu}{\mu^{\rho}}\right] \\
&= -\frac{1}{\phi}\mathbb{E}\left[\frac{y-\mu}{\phi\mu^{\rho}}\right] \\
&= -\frac{1}{\phi}\mathbb{E}\left[\frac{\partial \log g(y|\mu,\phi,\rho)}{\partial\mu}\right] \\
&= 0,
\end{aligned}$$

also

$$
\begin{aligned}
\mathbb{E}\left[\frac{\partial^2 \log g(y|\mu,\phi,\rho)}{\partial\mu\partial\rho}\right] &= \mathbb{E}\left[\frac{\partial}{\partial\rho}\left(\frac{y-\mu}{\phi\mu^\rho}\right)\right] \\
&= \mathbb{E}\left[\log\mu \cdot \frac{y-\mu}{\phi\mu^\rho}\right] \\
&= \log\mu \cdot \mathbb{E}\left[\frac{y-\mu}{\phi\mu^\rho}\right] \\
&= \log\mu \cdot \mathbb{E}\left[\frac{\partial \log g(y|\mu,\phi,\rho)}{\partial\mu}\right] \\
&= 0.
\end{aligned}
$$

Therefore μ is orthogonal to both ϕ and ρ (Cox and Reid, 1987, 1989; Jørgensen and Knudsen, 2004). The statistical consequences of this orthogonality is that the maximum likelihood estimates $\hat{\mu}$ is asymptotically independent to $\hat{\phi}$ and $\hat{\rho}$.

F Additional Tables and Figures

Table S1: The mean computation times for Case I Model 1 based on 20 replications for different values of ϕ.

ϕ	MGCV	TDboost	TGLM	RBF	Laplace
0.1	0.020	0.971	0.001	0.638	1.273
0.5	0.055	0.994	0.001	0.672	1.340
1.0	0.019	0.970	0.001	0.687	1.457
2.0	0.022	0.982	0.001	0.682	1.611

Table S2: The mean computation times for Case I Model 2 based on 20 replications for different values of ϕ.

ϕ	MGCV	TDboost	TGLM	RBF	Laplace
0.1	0.130	4.211	0.002	0.915	2.095
0.5	0.037	4.369	0.001	0.958	3.444
1.0	0.064	4.230	0.001	0.976	4.469
2.0	0.130	4.132	0.001	1.027	5.822

Table S3: The mean and standard errors of MADs, $\widehat{\rho}$ and $\widehat{\phi}$ based on 20 independent replications. True $\rho = 1.5$ and true $\phi = 0.5$

Model	MAD	$\widehat{\rho}$	$\widehat{\phi}$
1	0.096 (0.004)	1.503 (0.0126)	0.497 (0.008)
2	0.088 (0.003)	1.441 (0.024)	0.505 (0.013)

Table S4: The mean computation times for Case II based on 20 replications for different values of ϕ.

ϕ	MGCV	TDboost	RBF	Laplace
0.1	0.703	0.088	0.417	0.436
0.5	0.686	0.088	0.672	0.706
1.0	0.679	0.088	0.202	0.276
2.0	0.756	0.088	0.236	0.274

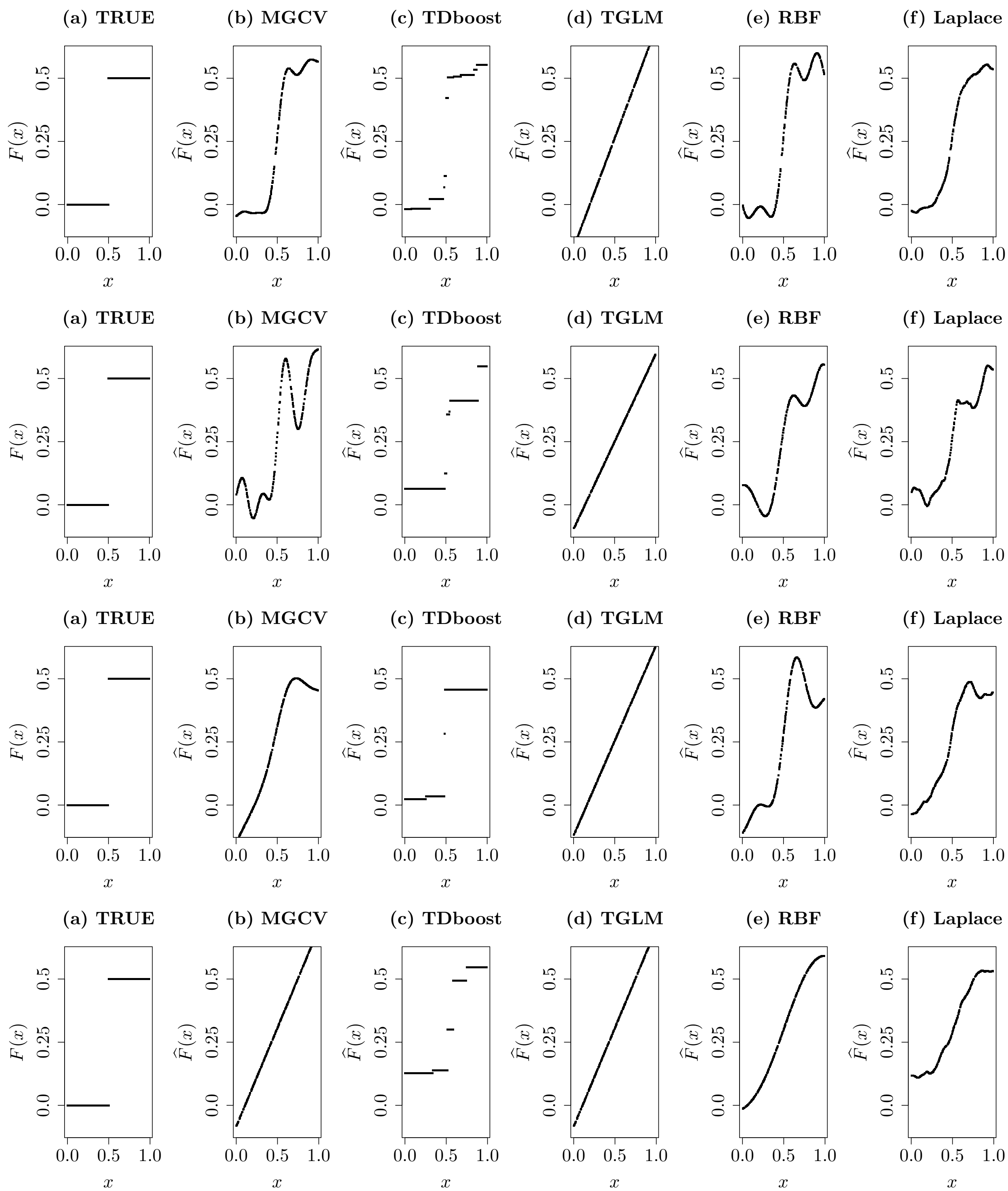
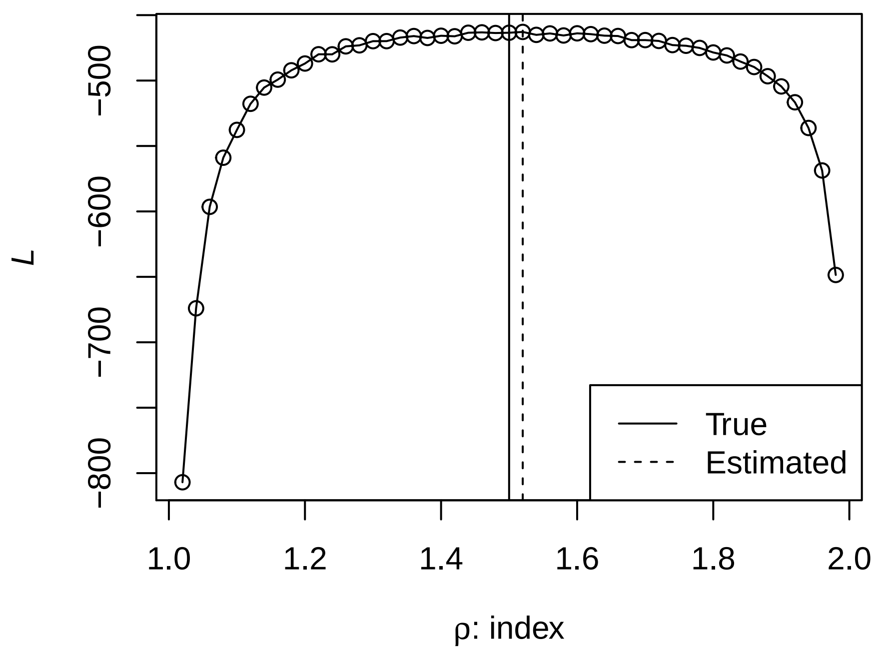


Figure S1: Fitted $\widehat{F}(\mathbf{x})$ vs. true $F(\mathbf{x})$ in Model 1 from a sample run (top to bottom $\phi = 0.1, 0.5, 1.0, 2.0$).



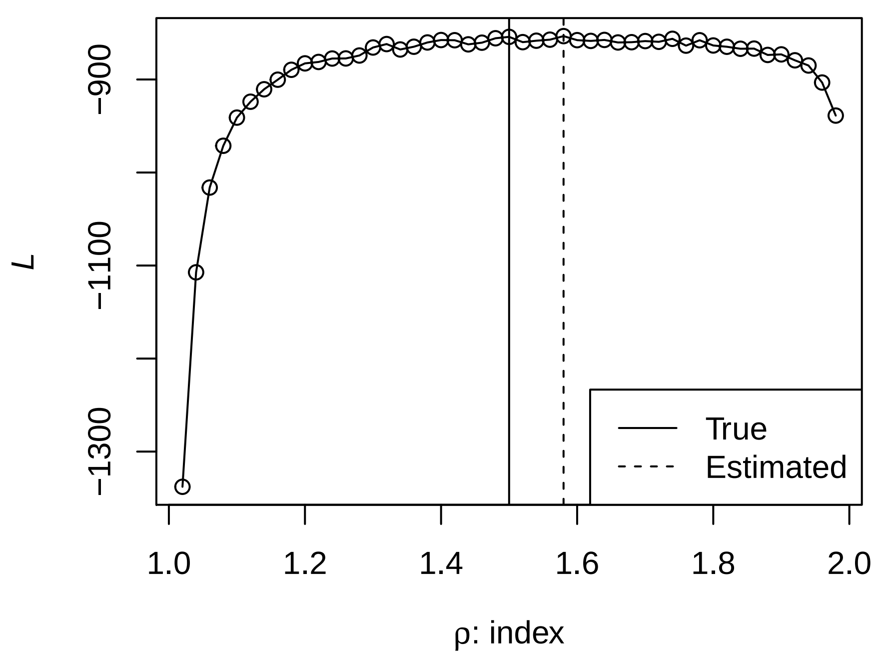


Figure S2: The profile likelihood of ρ from a sample run. Model 1 (left): true $\rho = 1.5$, $\widehat{\rho} = 1.52$; Model 2 (right): true $\rho = 1.5$, $\widehat{\rho} = 1.58$.

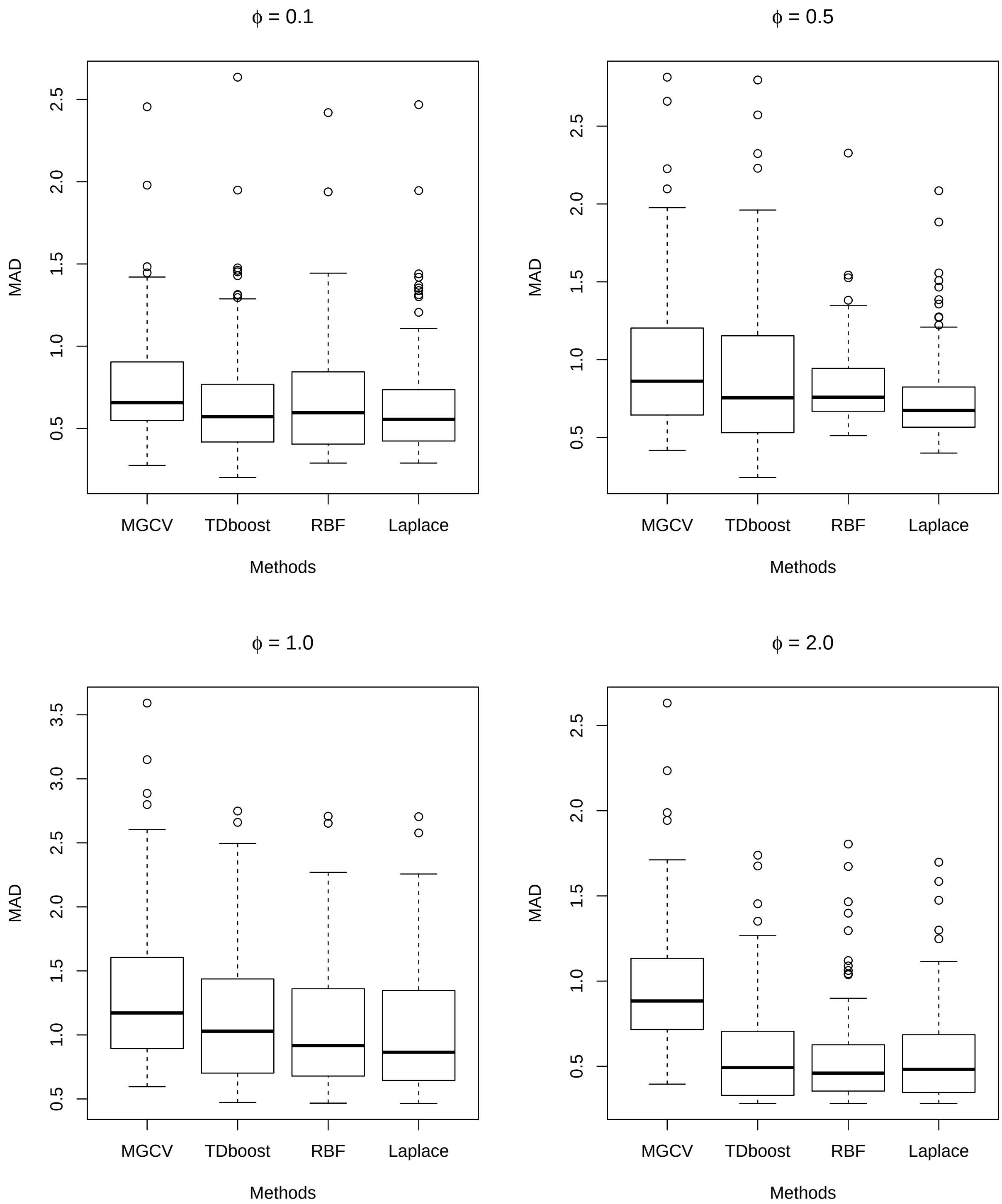


Figure S3: Distribution of the mean absolute deviations from the MGCV, TDboost, and Ktweedie (RBF and Laplace kernel) in Case II based on 100 independent replications.

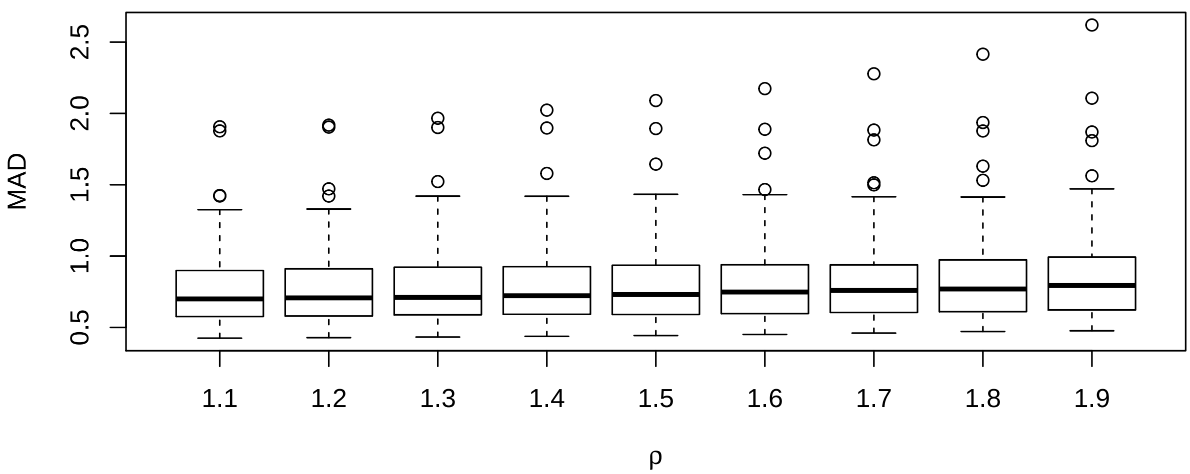


Figure S4: Boxplot of the mean absolute deviations for different values of the index parameter $\rho \in \{1.1, 1.2, \ldots, 1.9\}$ used during model fitting when the true value ($\rho = 1.5$) is unknown. The estimation accuracy is almost unaffected by ρ.

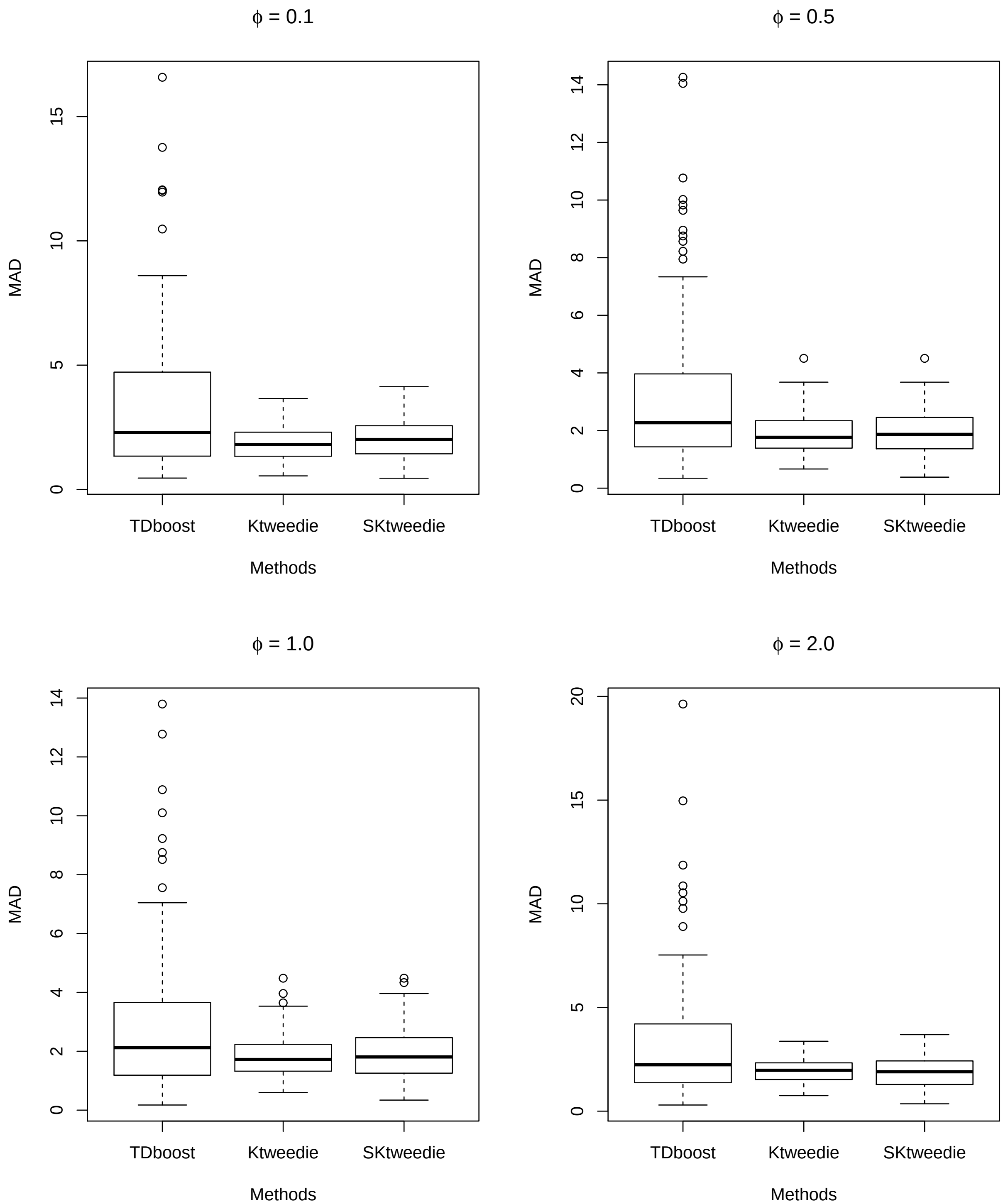


Figure S5: Distribution of the mean absolute deviations from the TDboost, Ktweedie, and SKtweedie in Case III based on 100 independent replications.

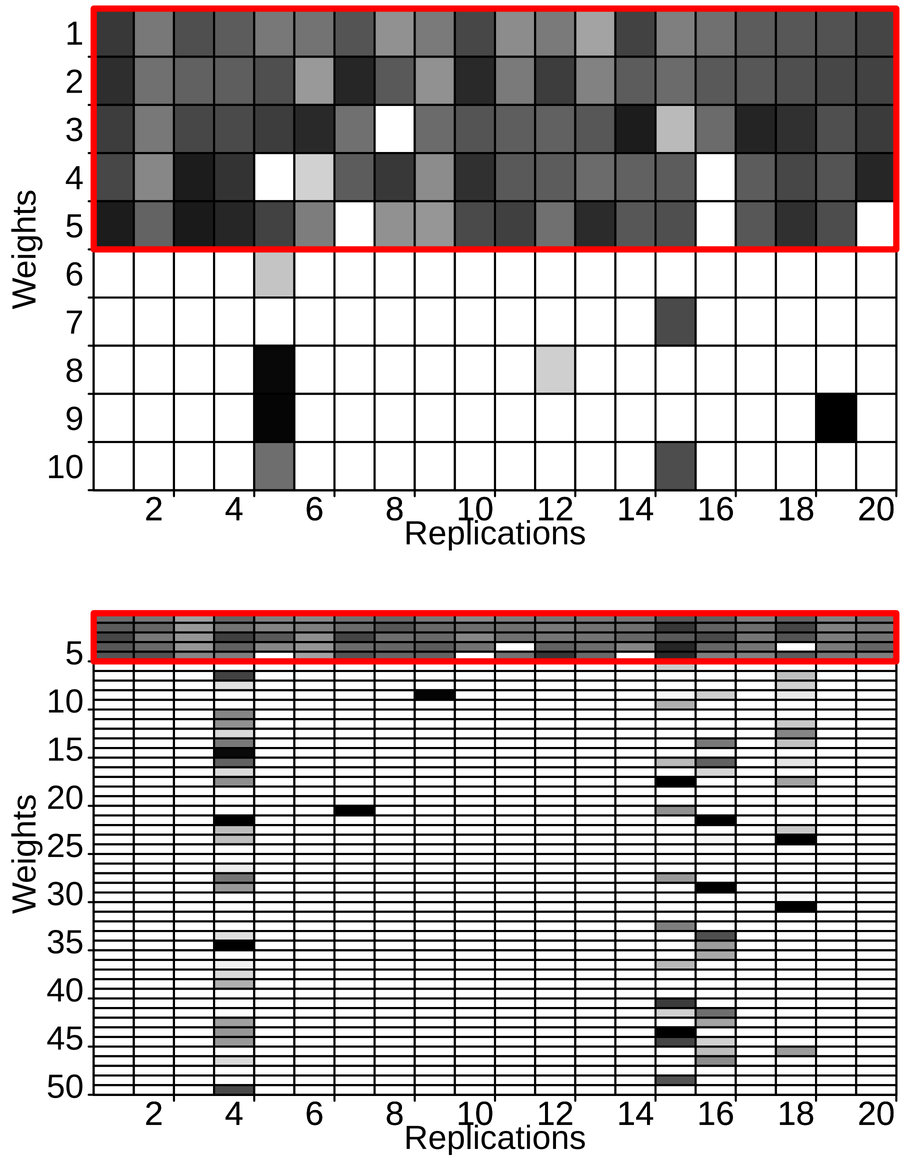


Figure S6: Variable selection results using SKtweedie with the Gaussian RBF kernel (left: $p = 10$, right: $p = 50$). Each column corresponds to a replication and each row corresponds to a variable, thus within the red rectangles are the true signal variables. The grayscale represents the magnitude of the estimated weights with a value between 0 and 1.

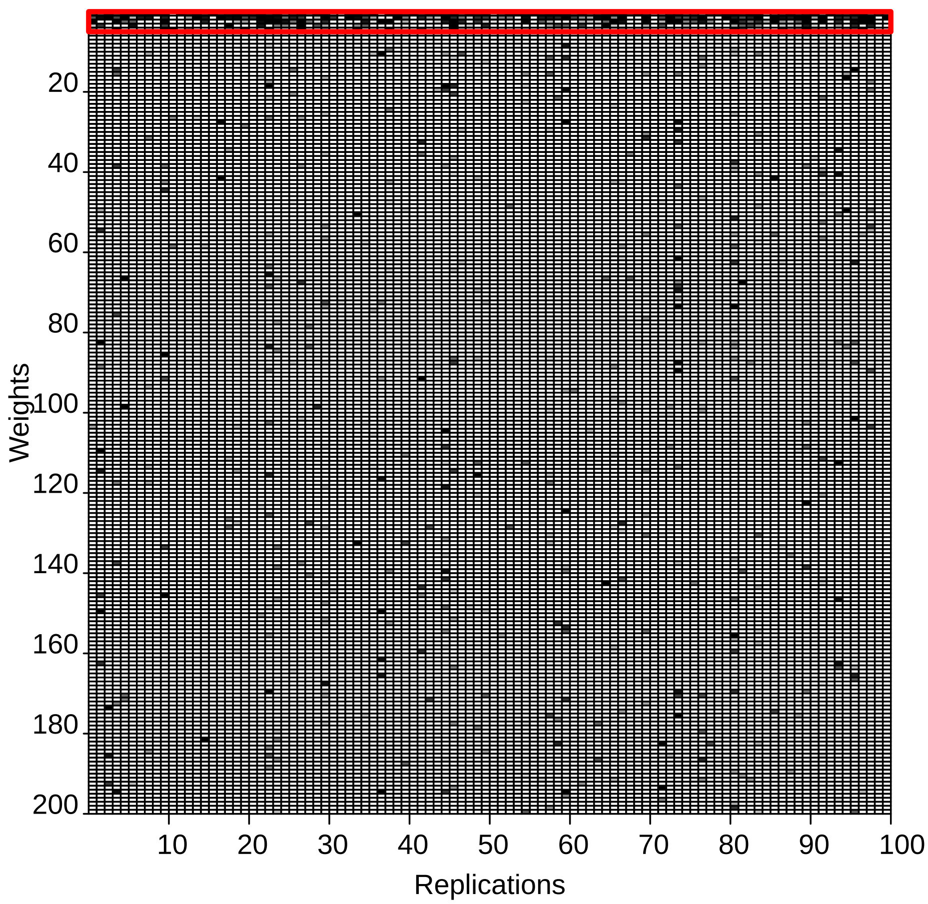


Figure S7: Variable selection results using the SKtweedie with Gaussian RBF kernel ($p = 200$)

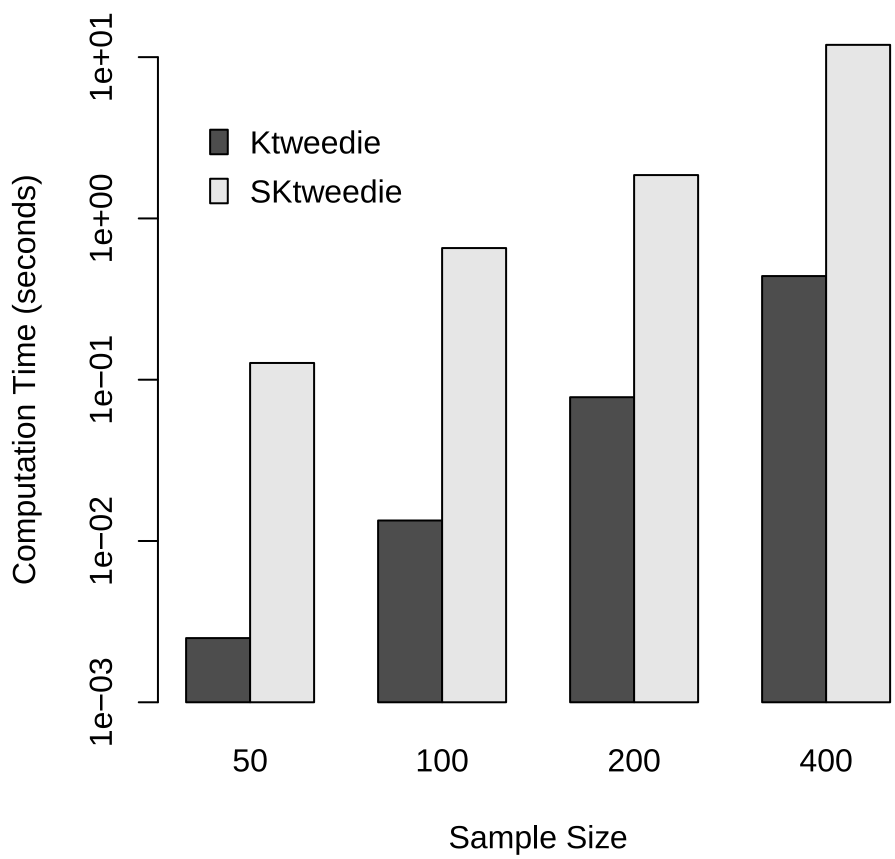


Figure S8: Computation times needed to fit a Ktweedie model and an SKtweedie model for sample size $n = 50, 100, 200, 400$ and $p = 10$ in simulation Case IV.

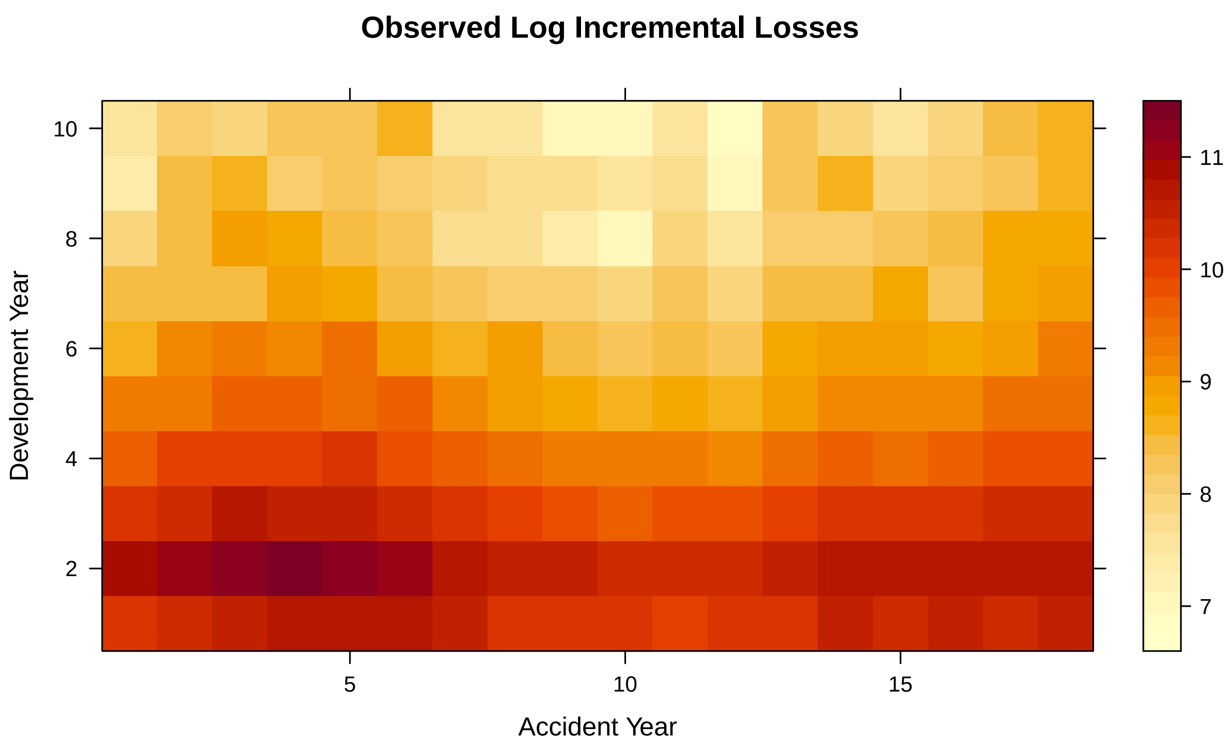


Figure S9: A heatmap of the log incremental losses by accident year and development year.

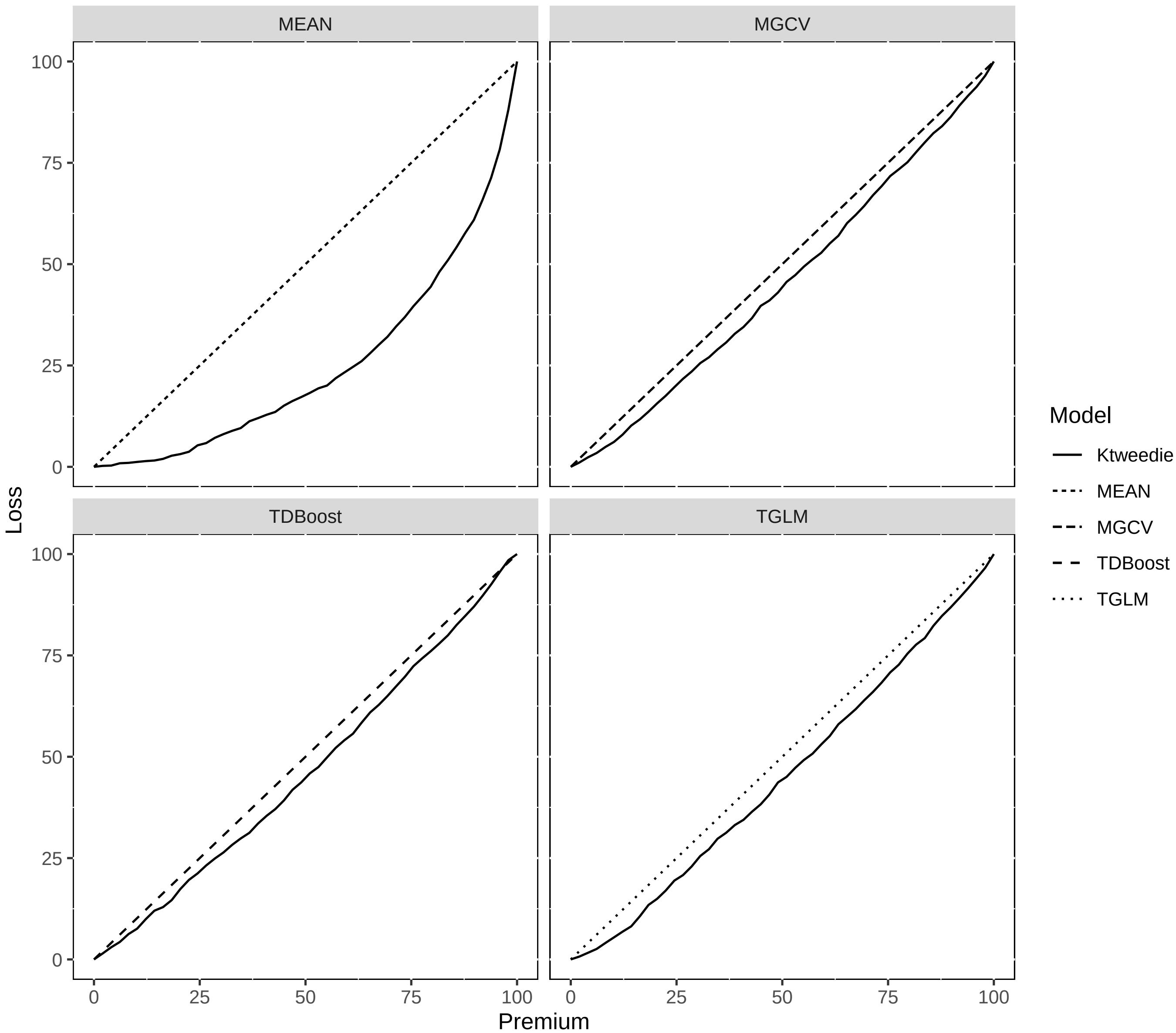


Figure S10: The ordered Lorenz curves for the auto-insurance claim data. In all four plots, the Ktweedie serves as the competing model.

References

Cox, D. and Reid, N. (1989) On the stability of maximum-likelihood estimators of orthogonal parameters. *Canadian Journal of Statistics*, **17**, 229–233. E

Cox, D. R. and Reid, N. (1987) Parameter orthogonality and approximate conditional inference. *Journal of the Royal Statistical Society: Series B (Methodological)*, **49**, 1–18. E

Jørgensen, B. and Knudsen, S. J. (2004) Parameter orthogonality and bias adjustment for estimating functions. *Scandinavian Journal of Statistics*, **31**, 93–114. E

Nocedal, J. and Wright, S. (2006) *Numerical optimization*. Springer Science & Business Media. C, C, C