TR-No. 24-02, Hiroshima Statistical Research Group, 1–23

Generalized Fused Lasso for Grouped Data in Generalized Linear Models

Mineaki Ohishi

Center for Data-driven Science and Artificial Intelligence, Tohoku University
Kawauchi 41, Aoba-ku, Sendai, 980-8576, Japan

Abstract

Generalized fused Lasso (GFL) is a powerful method based on adjacent relationships or the network structure of data. It is used in a number of research areas, including clustering, discrete smoothing, and spatio-temporal analysis. When applying GFL, the specific optimization method used is an important issue. In generalized linear models, efficient algorithms based on the coordinate descent method have been developed for trend filtering under the binomial and Poisson distributions. However, to apply GFL to other distributions, such as the negative binomial distribution, which is used to deal with overdispersion in the Poisson distribution, or the gamma and inverse Gaussian distributions, which are used for positive continuous data, an algorithm for each individual distribution must be developed. To unify GFL for distributions in the exponential family, this paper proposes a coordinate descent algorithm for generalized linear models. To illustrate the method, a real data example of spatio-temporal analysis is provided.

(Last Modified: February 8, 2024)

Key words: Grouped data, Coordinate descent algorithm, Generalized fused Lasso, Generalized linear models, Multivariate trend filtering.

E-mail address: mineaki.ohishi.a4@tohoku.ac.jp

1. Introduction

Assume we have grouped data such that $y_{j1}, \ldots, y_{jn_j}$ are observations of the jth group ($j \in \{1, \ldots, m\}$) for m groups. Further, assume the following generalized linear models (GLMs; Nelder & Wedderburn, 1972) with canonical parameter θ_{ji} and dispersion parameter $\phi > 0$:

$$y_{ji} \sim p_{ji}(\theta_{ji}, \phi) = \exp\left[\frac{a_{ji}}{a(\phi)}\{\theta_{ji} y_{ji} - b(\theta_{ji})\} + c(y_{ji}, \phi)\right] \quad (i \in \{1, \ldots, n_j\}), \tag{1.1}$$

where y_{ji} is independent with respect to j and i, a_{ji} is a constant defined by

$$a_{ji} = \begin{cases} (\text{the number of trials}) & (y_{ji} \text{ follows a binomial distribution}) \\ 1 & (\text{otherwise}) \end{cases},$$

$a(\cdot) > 0$, $b(\cdot)$, and $c(\cdot)$ are known functions, and $b(\cdot)$ is differentiable. The θ_{ji} has the following structure:

$$\theta_{ji} = h(\eta_{ji}), \quad \eta_{ji} = \beta_j + q_{ji},$$

where $h(\cdot)$ is a known differentiable function, β_j is an unknown parameter, and q_{ji} is a known term called the offset, which is zero in many cases. Although θ_{ji} depends not only on the group but also on the individual, the jth group is characterized by a common parameter β_j. We are thus interested in describing the relationship among the m groups. Here, the expectation of y_{ji} is given by

$$\mathrm{E}[y_{ji}] = \mu(\eta_{ji}) = \dot{b}(\theta_{ji}),$$

where $\mu(\cdot)$ is a known function and $\dot{b}(\cdot)$ is a derivative of $b(\cdot)$, i.e., $\dot{b}(\theta) = db(\theta)/d\theta$. Furthermore, $\mu^{-1}(\cdot)$ is a link function, and $h(\cdot)$ is an identify function, i.e., $h(\eta) = \eta$, when $\mu^{-1}(\cdot)$ is a canonical link. Tables 1, 2, and 3 summarize the relationships between model (1.1) and each individual distribution. In this paper, we consider clustering for m groups or discrete smoothing via generalized fused Lasso (GFL; e.g., Höfling *et al.*, 2010; Ohishi *et al.*, 2021).

Table 1. Expectation (ζ (= E[y])) and dispersion (ϕ)

Gaussian	$N(\zeta, \phi)$
Binomial	$B(a, \zeta)/a$ ($\phi = 1$)
Poisson	$P(\zeta)$ ($\phi = 1$)
Negative binomial	$NB(\zeta, 1/\phi)$ (Poisson-Gamma mixture: $y \sim P(\zeta_0)$ and $\zeta_0 \sim Ga(1/\phi, \zeta\phi)$)
Gamma	$Ga(1/\phi, \zeta\phi)$ (shape: $1/\phi$; scale: $\zeta\phi$)
Inverse Gaussian	$IG(\zeta, 1/\phi)$ (shape: $1/\phi$)

GFL is an extension of fused Lasso (Tibshirani *et al.*, 2005) which can incorporate relationships among multiple variables, such as adjacent relationships and network structure, into parameter estimation. For example, Xin *et al.* (2014) applied GFL to the diagnosis of Alzheimer's disease by expressing the structure of structural magnetic resonance images of human brains as a 3D grid graph; Ohishi *et al.* (2021) applied GFL to model spatial data based on geographical adjacency. Although the GFL in these particular instances is based on one factor (brain structure or a geographical relationship), it can deal with relationships based on multiple factors. For example, we can define an adjacent relationship

Table 2. GLM components for canonical link ($h(\eta) = \eta$)

	$\mu(\eta)$	link ($\mu^{-1}(\zeta)$)	$a(\phi)$	$b(\theta)$	$V(\zeta)^{(1)}$
Gaussian	η	ζ	ϕ	$\theta^2/2$	1
Binomial	$\exp(\eta)/\{1+\exp(\eta)\}$	$\log\{\zeta/(1-\zeta)\}$	1	$-\log\{1-\mu(\theta)\}$	$\zeta(1-\zeta)$
Poisson	$\exp(\eta)$	$\log\zeta$	1	$\exp(\theta)$	ζ
Negative binomial	$\phi^{-1}\exp(\eta)/\{1-\exp(\eta)\}$	$\log\{\zeta/(\phi^{-1}+\zeta)\}$	1	$\phi^{-1}\log\{\phi^{-1}+\mu(\theta)\}$	$\zeta+\phi\zeta^2$
Gamma	$-1/\eta$	$-1/\zeta$	ϕ	$-\log(-\theta)$	ζ^2
Inverse Gaussian	$(-\eta)^{-1/2}$	$-1/\zeta^2$	2ϕ	$-2\sqrt{-\theta}$	ζ^3

(1) $V(\cdot)$ is a variance function.

Table 3. GLM components for log-link ($\mu^{-1}(\zeta) = \log\zeta$)

	$h(\eta)$	$h^{-1}(\theta)$	$\dot{h}(\eta)$
Negative binomial	$\log[\exp(\eta)/\{\phi^{-1}+\exp(\eta)\}]$	$\log[\phi^{-1}\exp(\theta)/\{1-\exp(\theta)\}]$	$\{1+\phi\exp(\eta)\}^{-1}$
Gamma	$-\exp(-\eta)$	$-\log(-\theta)$	$\exp(-\eta)$
Inverse Gaussian	$-\exp(-2\eta)$	$-\log(-\theta)/2$	$2\exp(-2\eta)$

for spatio-temporal cases based on two factors by combining geographical adjacency and the order of time. Yamamura *et al.* (2021), Ohishi *et al.* (2022), and Yamamura *et al.* (2023) dealt with multivariate trend filtering (e.g., Tibshirani, 2014) based on multiple factors via GFL and applied it to the estimation of spatio-temporal trends. Yamamura *et al.* (2021) and Ohishi *et al.* (2022) used a logistic regression model, which coincides with model (1.1) when $n_j = 1, q_{ji} = 0$ ($\forall j \in \{1,\ldots,m\}; \forall i \in \{1,\ldots,n_j\}$) under a binomial distribution. Since this relationship holds by the reproductive property of the binomial distribution, their methods can also be applied to grouped data. Yamamura *et al.* (2023) used a Poisson regression model, which coincides with model (1.1) when $n_j = 1$ ($\forall j \in \{1,\ldots,m\}$) under a Poisson distribution. As is the case for Yamamura *et al.* (2021) and Ohishi *et al.* (2022), the method of Yamamura *et al.* (2023) can also be applied to grouped data from the reproductive property of the Poisson distribution. Yamamura *et al.* (2021), Ohishi *et al.* (2022) and Yamamura *et al.* (2023) proposed coordinate descent algorithms to obtain the GFL estimator. Although optimization problems for GLMs, such as logistic and Poisson regression models, are generally solved by linear approximation, Ohishi *et al.* (2022) and Yamamura *et al.* (2023) directly minimize coordinate-wise objective functions and derive update equations of a solution in closed form. Although Yamamura *et al.* (2021) minimized the coordinate-wise objective functions using linear approximation, Ohishi *et al.* (2022) showed numerically that direct minimization can provide the solution faster and more accurately than minimization using a linear approxima-

tion. Ohishi *et al.* (2021) also derived an explicit update equation for the coordinate descent algorithm, which corresponds to model (1.1) under the Gaussian distribution. As described, coordinate descent algorithms have been developed to produce GFL estimators for three specific distributions; however, none have been proposed for other distributions. For example, we have an option of using the negative binomial distribution to deal with overdispersion in the Poisson distribution (e.g., Gardner *et al.*, 1995; Ver Hoef & Boveng, 2007), or the gamma or inverse Gaussian distribution for positive continuous data. To apply GFL to these distributions, it is necessary to derive update equations for each distribution individually.

In this paper, we propose a coordinate descent algorithm to obtain GFL estimators for model (1.1) in order to unify the GFL approach for distributions in the exponential family. The negative log-likelihood function for model (1.1) is given by

$$\begin{aligned}&\frac{1}{a(\phi)}\sum_{j=1}^{m}\sum_{i=1}^{n_j}\left[a_{ji}\left\{b(h(\beta_j+q_{ji}))-y_{ji}h(\beta_j+q_{ji})\right\}-c(y_{ji},\phi)\right]\\&\quad=\frac{1}{a(\phi)}\sum_{j=1}^{m}\sum_{i=1}^{n_j}a_{ji}\left\{b(h(\beta_j+q_{ji}))-y_{ji}h(\beta_j+q_{ji})\right\}-\frac{1}{a(\phi)}\sum_{j=1}^{m}\sum_{i=1}^{n_j}c(y_{ji},\phi).\end{aligned}$$

We estimate parameter vector $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_m)'$ by minimizing the following function defined by removing terms that do not depend on $\boldsymbol{\beta}$ from the above equation and by adding a GFL penalty:

$$L(\boldsymbol{\beta})=\sum_{j=1}^{m}\sum_{i=1}^{n_j}a_{ji}\left\{b(h(\beta_j+q_{ji}))-y_{ji}h(\beta_j+q_{ji})\right\}+\lambda\sum_{j=1}^{m}\sum_{\ell\in D_j}w_{j\ell}|\beta_j-\beta_\ell|, \tag{1.2}$$

where λ is a non-negative tuning parameter, $D_j\subseteq\{1,\ldots,m\}\backslash\{j\}$ is an index set expressing adjacent relationship among groups and satisfying $\ell\in D_j\Leftrightarrow j\in D_\ell$, and $w_{j\ell}$ is a positive weight satisfying $w_{j\ell}=w_{\ell j}$. The GFL penalty shrinks the difference between two adjacent groups $|\beta_j-\beta_\ell|$ and often gives a solution satisfying $|\beta_j-\beta_\ell|=0$ ($\Leftrightarrow\beta_j=\beta_\ell$). That is, GFL can estimate some parameters to be exactly equal, thus enabling the clustering of m groups or the accomplishment of discrete smoothing. To obtain the GFL estimator for $\boldsymbol{\beta}$, we minimize the objective function (1.2) via a coordinate descent algorithm. As Ohishi *et al.* (2022) and Yamamura *et al.* (2023), we directly minimize coordinate-wise objective functions without the use of approximations. For ordinary situations, where a canonical link ($h(\eta)=\eta$) is used and there is no offset ($q_{ji}=0$), and for several other situations, the update equation of a solution can be derived in closed form. Table 4 summarizes relationships between an individual distribution and an update equation. Here, ◯ indicates that the update equation can be obtained in closed form, and × indicates that it cannot. Even when the update equation cannot be obtained in closed form, the proposed algorithm can specify an interval that includes the solution, which

Table 4. Whether the update equation can be obtained in closed form

Using canonical link	offset	
	without	with
Gaussian	○	○
Binomial	○	×
Poisson	○	○
Negative binomial	○	×
Gamma	○	×
Inverse Gaussian	○	×

Using log-link	offset	
	without	with
Negative binomial	○	×
Gamma	○	○
Inverse Gaussian	×	×

means we can easily obtain the solution by a simple numerical search.

The remainder of the paper is organized as follows: In Section 2, we give an overview of coordinate descent algorithm and derive the objective functions for each step. In Section 3, we discuss coordinate-wise minimization of the coordinate descent algorithm and derive update equations in closed form in many cases. In Section 4, we evaluate the performance of the proposed method via numerical simulation. In Section 5, we provide a real data example. Section 6 concludes the paper. Technical details are given in the Appendix.

2. Preliminaries

As in Ohishi *et al.* (2022) and Yamamura *et al.* (2023), we minimize the objective function (1.2) using a coordinate descent algorithm. Algorithm 1 gives an overview of the algorithm. The descent cycle updates the parameters separately, and several parameters are often updated

Algorithm 1 Overview of the coordinate descent algorithm

Require: λ and initial vector for $\boldsymbol{\beta}$
  repeat
    execute descent cycle
    if some parameters are exact equal **then**
      execute fusion cycle
    end if
  until solution converges

to be exactly equal. If several parameters are exactly equal, their updates can become stuck. To avoid this, the fusion cycle simultaneously updates equal parameters (Friedman *et al.*, 2007). In each cycle of the coordinate descent, the following function is essentially minimized:

$$f(x) = \sum_{i=1}^{d} a_i \{b(h(x+q_i)) - y_i h(x+q_i)\} + 2\lambda \sum_{\ell=1}^{r} w_\ell |x - z_\ell|, \tag{2.1}$$

where a_i and w_ℓ are positive constants and z_ℓ ($\ell = 1, \ldots, r$) are constants satisfying $z_1 < \cdots < z_r$. The minimization of $f(x)$ is described in Section 3, and the following subsections show that an objective function in each cycle is essentially equal to $f(x)$.

2.1. Descent cycle

The descent cycle repeats coordinate-wise minimizations of the objective function $L(\boldsymbol{\beta})$ in (1.2). To obtain a coordinate-wise objective function, we extract terms that depend on β_j ($j \in \{1, \ldots, m\}$) from $L(\boldsymbol{\beta})$. As described in Ohishi *et al.* (2021), the penalty term can be decomposed as

$$\sum_{l=1}^{m} \sum_{\ell \in D_l} w_{l\ell} |\beta_l - \beta_\ell| = 2 \sum_{\ell \in D_j} w_{j\ell} |\beta_j - \beta_\ell| + \sum_{l \neq j}^{m} \sum_{\ell \in D_l \setminus \{j\}} w_{l\ell} |\beta_l - \beta_\ell|.$$

Then, only the first term depends on β_j. By regarding terms that do not depend on β_j as constants and removing them from $L(\boldsymbol{\beta})$, the coordinate-wise objective function is obtained as

$$L_j(\beta) = \sum_{i=1}^{n_j} a_{ji} \left\{ b(h(\beta + q_{ji})) - y_{ji} h(\beta + q_{ji}) \right\} + 2\lambda \sum_{\ell \in D_j} w_{j\ell} |\beta - \hat{\beta}_\ell|, \tag{2.2}$$

where $\hat{\beta}_\ell$ indicates β_ℓ is given. By sorting elements of D_j in increasing order of $\hat{\beta}_\ell$ ($\forall \ell \in D_j$), we can see that $L_j(\beta)$ essentially equals $f(x)$ in (2.1). If there exist $\ell_1, \ell_2 \in D_j$ ($\ell_1 \neq \ell_2$) such that $\hat{\beta}_{\ell_1} = \hat{\beta}_{\ell_2}$, we can temporarily redefine D_j and $w_{j\ell}$ as

$$D_j \longleftarrow D_j \setminus \{\ell_2\}, \quad w_{j\ell_1} \longleftarrow w_{j\ell_1} + w_{j\ell_2}.$$

Since GFL estimates several parameters as being equal, this redefinition is required in most updates.

2.2. Fusion cycle

In the fusion cycle, equal parameters are replaced by a common parameter and $L(\boldsymbol{\beta})$ is minimized with respect to the common parameter. Let $\hat{\beta}_1, \ldots, \hat{\beta}_m$ be current solutions for $\beta_1, \ldots, \beta_m$, and $\hat{\xi}_1, \ldots, \hat{\xi}_t$ ($t < m$) be their distinct values. The relationship among the current solutions and their distinct values is specified as

$$E_k = \left\{ j \in \{1, \ldots, m\} \mid \hat{\beta}_j = \hat{\xi}_k \right\} \quad (k = 1, \ldots, t).$$

That is, the following statements are true:

$$j_1, j_2 \in E_k \Longleftrightarrow \hat{\beta}_{j_1} = \hat{\beta}_{j_2} = \hat{\xi}_k, \quad j_1 \in E_{k_1}, j_2 \in E_{k_2}\ (k_1 \neq k_2) \Longleftrightarrow \hat{\xi}_{k_1} = \hat{\beta}_{j_1} \neq \hat{\beta}_{j_2} = \hat{\xi}_{k_2}.$$

Then, the β_j ($\forall j \in E_k$) are replaced by a common parameter ξ_k and $L(\boldsymbol{\beta})$ is minimized with respect to ξ_k. Hence, to obtain a coordinate-wise objective function, we extract terms that depend on ξ_k ($k = 1, \ldots, t$) from $L(\boldsymbol{\beta})$.

We can decompose the first term of $L(\boldsymbol{\beta})$ as

$$\sum_{j=1}^{m}\sum_{i=1}^{n_j} a_{ji}\left\{b(h(\beta_j + q_{ji})) - y_{ji}h(\beta_j + q_{ji})\right\}$$
$$= \sum_{j\in E_k}\sum_{i=1}^{n_j} a_{ji}\left\{b(h(\xi_k + q_{ji})) - y_{ji}h(\xi_k + q_{ji})\right\} + \sum_{j\notin E_k}\sum_{i=1}^{n_j} a_{ji}\left\{b(h(\beta_j + q_{ji})) - y_{ji}h(\beta_j + q_{ji})\right\}.$$

Furthermore, as Ohishi *et al.* (2021), the penalty term of $L(\boldsymbol{\beta})$ can be decomposed as

$$\sum_{j=1}^{m}\sum_{\ell\in D_j} w_{j\ell}|\beta_j - \beta_\ell| = 2\sum_{j\in E_k}\sum_{\ell\in D_j\setminus E_k} w_{j\ell}|\xi_k - \beta_\ell| + \sum_{j\notin E_k}\sum_{\ell\in D_j\setminus E_k} w_{j\ell}|\beta_j - \beta_\ell|.$$

By regarding terms that do not depend on ξ_k as constants and removing them from $L(\boldsymbol{\beta})$, the coordinate-wise objective function is obtained as

$$L_k^*(\xi) = \sum_{j\in E_k}\sum_{i=1}^{n_j} a_{ji}\left\{b(h(\xi + q_{ji})) - y_{ji}h(\xi + q_{ji})\right\} + 2\lambda\sum_{j\in E_k}\sum_{\ell\in D_j\setminus E_k} w_{j\ell}|\xi - \hat{\beta}_\ell|. \tag{2.3}$$

As in the descent cycle, we can see that $L_k^*(\xi)$ essentially equals $f(x)$ in (2.1).

3. Main results

In this section, to obtain update equations for the descent and fusion cycles of the coordinate descent algorithm, we describe the minimization of $f(x)$ in (2.1). Following Ohishi *et al.* (2022) and Yamamura *et al.* (2023), we directly minimize $f(x)$. One of the difficulties of the minimization of $f(x)$ is that $f(x)$ has multiple non-differentiable points $z_1, \ldots, z_r$. We cope with this difficulty by using a subdifferential. The subdifferential of $f(x)$ at $\tilde{x} \in \mathbb{R}$ is given by

$$\partial f(\tilde{x}) = \{u \in \mathbb{R} \mid f(x) \geq f(\tilde{x}) + u(x - \tilde{x})\ (\forall x \in \mathbb{R})\} = [g_-(\tilde{x}), g_+(\tilde{x})],$$

where $g_-(x)$ and $g_+(x)$ are left and right derivatives defined by

$$g_-(x) = \lim_{\delta\to-0} g(x,\delta), \quad g_+(x) = \lim_{\delta\to+0} g(x,\delta), \quad g(x,\delta) = \frac{f(x+\delta) - f(x)}{\delta}.$$

Then, $\tilde{x}$ is a stationary point of $f(x)$ if $0 \in \partial f(\tilde{x})$. For details of a subdifferential, see, e.g., Rockafellar (1970), Parts V and VI. In the following subsections, we separately describe the minimization of $f(x)$ in cases where a canonical link and a general link are used.

3.1. Canonical link

We first describe the minimization of $f(x)$ in (2.1) with a canonical link, i.e., $h(\eta) = \eta$. That is, the update equation of the coordinate descent algorithm is given by minimizing the following function:

$$f(x) = \sum_{i=1}^{d} a_i \{b(x + q_i) - y_i(x + q_i)\} + 2\lambda \sum_{\ell=1}^{r} w_\ell |x - z_\ell|. \tag{3.1}$$

Notice that $f(x)$ in (3.1) is strictly convex. Hence, $\tilde{x}$ is the minimizer of $f(x)$ if and only if $0 \in \partial f(\tilde{x})$. First, based on this relationship, we derive the condition that $f(x)$ attains the minimum at a non-differentiable point z_ℓ.

The subdifferential of $f(x)$ at z_ℓ is given by

$$\begin{aligned}
\partial f(z_\ell) &= [g_-(z_\ell), g_+(z_\ell)], \\
g_-(z_\ell) &= \sum_{i=1}^{d} a_i \{\mu(z_\ell + q_i) - y_i\} - 2\lambda w_\ell + 2\lambda \sum_{l \neq \ell}^{r} w_l \operatorname{sign}(z_\ell - z_l), \\
g_+(z_\ell) &= \sum_{i=1}^{d} a_i \{\mu(z_\ell + q_i) - y_i\} + 2\lambda w_\ell + 2\lambda \sum_{l \neq \ell}^{r} w_l \operatorname{sign}(z_\ell - z_l).
\end{aligned}$$

Hence, if there exists $\ell_\star \in \{1, \ldots, r\}$ such that $0 \in \partial f(z_{\ell_\star})$, $f(x)$ attains the minimum at $x = z_{\ell_\star}$ and $\ell_\star$ uniquely exists because of the strict convexity of $f(x)$.

On the other hand, when $\ell_\star$ does not exist, we can specify an interval that includes the minimizer by checking the signs of the left and right derivatives at each non-differentiable point. Let $s(x) = (\operatorname{sign}(g_-(x)), \operatorname{sign}(g_+(x)))$. From $z_1 < \cdots < z_r$ and the strict convexity of $f(x)$, we have

$$\exists! \ell_* \in \{0, 1, \ldots, r\} \ s.t. \ \forall \ell \in \{1, \ldots, r\}, \ s(z_\ell) = \begin{cases} (-1, -1) & (\ell \leq \ell_*) \\ (1, 1) & (\ell > \ell_*) \end{cases}.$$

Then, the minimizer of $f(x)$ exists in the following interval:

$$R_* = (z_{\ell_*}, z_{\ell_*+1}); \quad z_0 = -\infty, \ z_{r+1} = \infty.$$

Hence, it is sufficient to search for the minimizer in R_*. For all $x \in R_*$, the following equation holds:

$$|x - z_\ell| = \begin{cases} x - z_\ell & (\ell \leq \ell_*) \\ -x + z_\ell & (\ell > \ell_*) \end{cases}.$$

This result allows us to rewrite the penalty term in $f(x)$ as

$$\sum_{\ell=1}^{r} w_\ell |x - z_\ell| = \sum_{\ell=1}^{\ell_*} w_\ell (x - z_\ell) - \sum_{\ell=\ell_*+1}^{r} w_\ell (x - z_\ell) = \tilde{w}_1 x - \tilde{w}_2,$$
$$\tilde{w}_1 = \sum_{\ell=1}^{\ell_*} w_\ell - \sum_{\ell=\ell_*+1}^{r} w_\ell, \quad \tilde{w}_2 = \sum_{\ell=1}^{\ell_*} w_\ell z_\ell - \sum_{\ell=\ell_*+1}^{r} w_\ell z_\ell.$$

Hence, $f(x)$ is rewritten in non-absolute form as

$$f(x) = \sum_{i=1}^{d} a_i \{b(x + q_i) - y_i(x + q_i)\} + 2\lambda(\tilde{w}_1 x - \tilde{w}_2) \quad (x \in R_*).$$

The $f(x)$ is differentiable when $x \in R_*$ and its derivative is given by

$$\frac{d}{dx} f(x) = \sum_{i=1}^{d} a_i \{\mu(x + q_i) - y_i\} + 2\lambda\tilde{w}_1 = \sum_{i=1}^{d} a_i \mu(x + q_i) + u, \quad u = 2\lambda\tilde{w}_1 - \sum_{i=1}^{d} a_i y_i.$$

Then, the solution x_* of $df(x)/dx = 0$ is the minimizer of $f(x)$. Hence, we have the following theorem.

Theorem 1. *Let $\hat{x}$ be the minimizer of $f(x)$ in* (3.1)*. Then, $\hat{x}$ is given by*

$$\hat{x} = \begin{cases} z_{\ell_\star} & (\ell_\star \textit{ exists}) \\ x_* & (\ell_* \textit{ exists}) \end{cases},$$

where ℓ_* exists if and only if $\ell_\star$ does not exist.

Algorithm 2 The coordinate descent algorithm for a canonical link

Require: λ and initial vector for $\boldsymbol{\beta}$
  repeat
    (descent cycle)
    for $j \in \{1, \ldots, m\}$ **do**
      update β_j by applying Theorem 1 to (2.2)
    end for
    define E_k $(k = 1, \ldots, t)$
    if $t < m$ **then**
      (fusion cycle)
      for $k \in \{1, \ldots, t\}$ such that $\#(E_k) > 1$ **do**
        update ξ_k $(= \beta_j; j \in E_k)$ by applying Theorem 1 to (2.3)
      end for
    end if
  until solution converges

We can execute Algorithm 1 by applying Theorem 1 to (2.2) and (2.3) in the descent and fusion

cycles, respectively. Thus, a detailed implementation of Algorithm 1 when using a canonical link is provided in Algorithm 2. To apply Theorem 1, we need to obtain x_*. In many cases, x_* can be obtained in closed form according to the following proposition.

Proposition 1. *Let x_* be the solution of $df(x)/dx = 0$ and q_0 be a value such that $q_1 = \cdots = q_d = q_0$. Then, x_* is given as follows:*

- *When q_0 exists, x_* is given in a general form as*

$$x_* = \mu^{-1}\left(-u/\sum_{i=1}^{d} a_i\right) - q_0.$$

- *Even when q_0 does not exist, x_* for the Gaussian and Poisson distributions is given by*

$$x_* = \begin{cases} \dfrac{1}{d}\left(-\displaystyle\sum_{i=1}^{d} q_i - u\right) & (\textit{Gaussian}) \\ \log(-u) - \log\left(\displaystyle\sum_{i=1}^{d} \exp(q_i)\right) & (\textit{Poisson}) \end{cases}.$$

For example, q_0 exists and $q_0 = 0$ holds for GLMs without an offset. When q_0 does not exist, x_* can be obtained for each distribution. For the Gaussian and Poisson distributions, since $\mu(x+q)$ can be divided with respect to x and q, x_* can be obtained in closed form. Note that x_* for a Gaussian distribution when q_0 exists and equals 0 coincides with the result in Ohishi *et al.* (2021). For distributions for which such a decomposition is impossible, such as the binomial distribution, a numerical search is required to obtain x_*. However, we can easily obtain x_* by a simple algorithm, such as a line search, because $f(x)$ is strictly convex and has its minimizer in the interval R_*.

3.2. General link

Here, we consider the minimization of $f(x)$ in (2.1) with a general link, i.e., $h(\cdot)$ is a generally differentiable function. Then, although strict convexity of $f(x)$ is not guaranteed, its continuity is maintained. This means the uniqueness of the minimizer of $f(x)$ is not guaranteed, but we can obtain minimizer candidates by using the same procedure as in the previous subsection.

The subdifferential of $f(x)$ at z_ℓ is given by

$$\begin{aligned} &\partial f(z_\ell) = [g_-(z_\ell), g_+(z_\ell)], \\ &g_-(z_\ell) = \sum_{i=1}^{d} a_i \dot{h}(z_\ell + q_i)\{\mu(z_\ell + q_i) - y_i\} - 2\lambda w_\ell + 2\lambda \sum_{l\neq\ell}^{r} w_l \operatorname{sign}(z_\ell - z_l), \end{aligned}$$

$$g_+(z_\ell) = \sum_{i=1}^{d} a_i \dot{h}(z_\ell + q_i)\{\mu(z_\ell + q_i) - y_i\} + 2\lambda w_\ell + 2\lambda \sum_{l\neq\ell}^{r} w_l \operatorname{sign}(z_\ell - z_l),$$

where $\dot{h}(x) = dh(x)/dx$. Since z_ℓ satisfying $0 \in \partial f(z_\ell)$ is a stationary point of $f(x)$, such points are minimizer candidates of $f(x)$. Next, we define intervals as $R_\ell = (z_\ell, z_{\ell+1})$ $(\ell = 0, 1, \ldots, r)$. For $x \in R_\ell$, $f(x)$ can be written in non-absolute form as

$$f(x) = f_\ell(x) = \sum_{i=1}^{d} a_i \{b(h(x + q_i)) - y_i h(x + q_i)\} + 2\lambda(\tilde{w}_{1,\ell} x - \tilde{w}_{2,\ell}),$$
$$\tilde{w}_{1,\ell} = \sum_{l=1}^{\ell} w_l - \sum_{l=\ell+1}^{r} w_l, \quad \tilde{w}_{2,\ell} = \sum_{l=1}^{\ell} w_l z_l - \sum_{l=\ell+1}^{r} w_l z_l.$$

We can then search for minimizer candidates of $f(x)$ by piecewise minimization. That is, $x \in R_\ell$ minimizing $f_\ell(x)$ is a minimizer candidate. Hence, we have the following theorem.

Theorem 2. *Let $\hat{x}$ be the minimizer of $f(x)$ in* (2.1) *and define a set $\mathcal{S}$ by*

$$\mathcal{S} = \{z \in \{z_1, \ldots, z_r\} \mid 0 \in \partial f(z)\} \cup \bigcup_{\ell=0}^{r} \left\{ \arg\min_{x \in R_\ell} f_\ell(x) \right\}.$$

Now, suppose that

$$\exists Z_- \in \mathbb{R} \; s.t. \; x \leq Z_- \Longrightarrow \dot{f}_0(x) < 0, \quad \exists Z_+ \in \mathbb{R} \; s.t. \; x \geq Z_+ \Longrightarrow \dot{f}_r(x) > 0, \tag{3.2}$$

where $\dot{f}_\ell(x) = df_\ell(x)/dx$. Then, $\mathcal{S}$ is the set of minimizer candidates of $f(x)$ and $\hat{x}$ is given by

$$\hat{x} = \arg\min_{x \in \mathcal{S}} f(x).$$

The assumption (3.2) excludes the case in which $f(x)$ attains the minimum at $x = \pm\infty$. Moreover, we have the following corollary (the proof is given in Appendix A.1).

Corollary 1. *Suppose that for all $\ell \in \{0, 1, \ldots, r\}$,*

$$\forall x \in R_\ell, \; \ddot{f}_\ell(x) = \frac{d^2}{dx^2} f_\ell(x) > 0,$$

is true, and that (3.2) *holds. Then, $f(x)$ is strictly convex and $\#(\mathcal{S}) = 1$, where $\mathcal{S}$ is given in Theorem 2. Moreover, the unique element of $\mathcal{S}$ is the minimizer of $f(x)$ and is given as in Theorem 1.*

To execute Algorithm 1 for GLMs with a general link, we can replace Theorem 1 with Theorem 2 or Corollary 1 in Algorithm 2. The next subsection gives specific examples of using a general link.

3.2.1. Examples

This subsection focuses on the negative binomial, gamma, and inverse Gaussian distributions with a log-link as examples of using a general link. In the framework of regression, the negative binomial distribution is often used to deal with overdispersion in Poisson regression, making it natural to use a log-link. Note that NB-C and NB2 indicate negative binomial regression with canonical and log-links, respectively (for details, see, e.g., Hilbe, 2011). The gamma and inverse Gaussian distributions are used to model positive continuous data. Their expectations must be positive. However, their canonical links do not guarantee that their expectations will, in fact, be positive. Hence, a log-link rather than a canonical link is often used for these distributions (e.g., Algamal, 2018; Dunn & Smyth, 2018, Chap. 11). Here, we consider coordinate-wise minimizations for the three distributions with a log-link.

For $x \in R_\ell$, $f(x)$ in (2.1) is given by

$$
\begin{aligned}
f(x) &= f_\ell(x) \\
&= \begin{cases}
\displaystyle\sum_{i=1}^{d} (\phi^{-1} + y_i) \log\left\{\phi^{-1} + \exp(x + q_i)\right\} + v_\ell x - 2\lambda\tilde{w}_{2,\ell} - \sum_{i=1}^{d} y_i q_i & \text{(NB2)} \\
u\exp(-x) + v_\ell x - 2\lambda\tilde{w}_{2,\ell} + \displaystyle\sum_{i=1}^{d} q_i & \text{(Gamma)} \\
\exp(-x)\left\{u_1\exp(-x) - 2u_2\right\} + v_\ell x - 2\lambda\tilde{w}_{2,\ell} & \text{(Inverse Gaussian)}
\end{cases}, \\
v_\ell &= \begin{cases}
2\lambda\tilde{w}_{1,\ell} - \sum_{i=1}^{d} y_i & \text{(NB2)} \\
2\lambda\tilde{w}_{1,\ell} + d & \text{(Gamma)} \\
2\lambda\tilde{w}_{1,\ell} & \text{(Inverse Gaussian)}
\end{cases}, \quad u = \sum_{i=1}^{d} \frac{y_i}{\exp(q_i)}, \\
u_1 &= \sum_{i=1}^{d} y_i \exp(-2q_i), \quad u_2 = \sum_{i=1}^{d} \exp(-q_i).
\end{aligned}
$$

Hence, the first- and second-order derivatives of $f_\ell(x)$ are given by

$$
\dot{f}_\ell(x) = \begin{cases}
\displaystyle\sum_{i=1}^{d} \left(\phi^{-1} + y_i\right) \exp(x + q_i) \left\{\phi^{-1} + \exp(x + q_i)\right\}^{-1} + v_\ell & \text{(NB2)} \\
-u\exp(-x) + v_\ell & \text{(Gamma)} \\
2\exp(-x)\{u_2 - u_1\exp(-x)\} + v_\ell & \text{(Inverse Gaussian)}
\end{cases},
$$

$$
\ddot{f}_\ell(x) = \begin{cases}
\phi^{-1} \displaystyle\sum_{i=1}^{d} \left(\phi^{-1} + y_i\right) \exp(x + q_i) \left\{\phi^{-1} + \exp(x + q_i)\right\}^{-2} & \text{(NB2)} \\
u\exp(-x) & \text{(Gamma)} \\
2\exp(-x)\{2u_1\exp(-x) - u_2\} & \text{(Inverse Gaussian)}
\end{cases}.
$$

We can see that $\ddot{f}_\ell(x) > 0$ holds for all $\ell \in \{0, 1, \ldots, r\}$, for NB2 and the gamma distribution. Hence, the minimizers of $f(x)$ can be uniquely obtained from Corollary 1. On the other hand, the uniqueness of the minimizer for the inverse Gaussian distribution is not guaranteed; however, we have $v_0 < 0$, $v_r > 0$, and

$$u_2 - u_1 \exp(-x) = \begin{cases} < 0 & (x < \log(u_1/u_2)) \\ > 0 & (x > \log(u_1/u_2)) \end{cases}.$$

This implies $x < \min\{\log(u_1/u_2), z_1\} \Rightarrow \dot{f}_0(x) < 0$ and $x > \max\{\log(u_1/u_2), z_r\} \Rightarrow \dot{f}_r(x) > 0$. Hence, the minimizer for the inverse Gaussian distribution can be obtained by Theorem 2.

We now give specific solutions. From above, we have the following proposition.

Proposition 2. *Let $\tilde{x}_\ell$ be a stationary point of $f_\ell(x)$. If $\tilde{x}_\ell$ exists, it is given by*

$$\tilde{x}_\ell = \begin{cases} \log(-v_\ell/\phi) - \log\left(2\lambda\tilde{w}_{1,\ell} + d/\phi\right) - q_0 & \textit{(NB2 only when } \exists q_0 \textit{ s.t. } q_1 = \cdots = q_d = q_0\textit{)} \\ \log(u/v_\ell) & \textit{(Gamma)} \\ \log\left(\left\{-u_2 + \sqrt{u_2^2 + 2u_1 v_\ell}\right\}/v_\ell\right) & \textit{(Inverse Gaussian)} \end{cases}.$$

Moreover, a relationship between $\tilde{x}_\ell$ and the minimizer of $f(x)$ is given by

$$\tilde{x}_\ell \textit{ exists in } R_\ell \Longrightarrow \begin{cases} \tilde{x}_\ell \textit{ is the unique minimizer of } f(x) & \textit{(NB2, Gamma)} \\ \tilde{x}_\ell \textit{ is a minimizer candidate of } f(x) & \textit{(Inverse Gaussian)} \end{cases}.$$

3.3. Some comments regarding implementation

3.3.1. Dispersion parameter estimation

In the previous subsections, we discussed the estimation of β_j which corresponds to the estimation of the canonical parameter θ_{ji}. The GLMs in (1.1) also have dispersion parameter ϕ. Although ϕ is fixed at one for the binomial and Poisson distributions, it is unknown for other distributions, and, hence, we need to estimate the value of ϕ. The Pearson estimator is often used as a suitable estimator (e.g., Dunn & Smyth, 2018, Chap. 6). Let $\hat{\beta}_1, \ldots, \hat{\beta}_m$ be estimators of $\beta_1, \ldots, \beta_m$, t be the number of distinct values of them, and $\hat{\zeta}_{ji} = \mu(\hat{\beta}_j + q_{ji})$. Then, the Pearson estimator of ϕ is given by

$$\hat{\phi} = \frac{\chi^2}{n - t}, \quad \chi^2 = \sum_{j=1}^{m} \sum_{i=1}^{n_j} \frac{(y_{ji} - \hat{\zeta}_{ji})^2}{V(\hat{\zeta}_{ji})},$$

where $V(\cdot)$ is a variance function (see Table 2). For distributions other than the negative binomial distribution, the estimator of ϕ can be obtained after $\boldsymbol{\beta}$ is estimated since the estimation of $\boldsymbol{\beta}$ does not depend on ϕ. For the negative binomial distribution, the estimation of $\boldsymbol{\beta}$ depends

on ϕ because $\mu(\cdot)$ and $b(\cdot)$ depend on ϕ. Hence, we need to add a step updating ϕ and repeat updates of $\boldsymbol{\beta}$ and ϕ alternately. Moreover, this Pearson estimator is used for the diagnosis of overdispersion in the binomial and Poisson distributions. If $\hat{\phi} > 1$, it is doubtful that the model is appropriate.

3.3.2. Penalty weights

The objective function $L(\boldsymbol{\beta})$ in (1.2) includes penalty weights, and the GFL estimation proceeds with the given weights. Although setting $w_{j\ell} = 1$ is usual, this may cause a problem of over-shrinkage because all pairs of parameters are shrunk uniformly by the common λ. As one option to avoid this problem, we can use the following weight based on adaptive-Lasso (Zou, 2006):

$$w_{j\ell} = 1/|\tilde{\beta}_j - \tilde{\beta}_\ell|,$$

where $\tilde{\beta}_j$ is an estimator of β_j and the maximum likelihood estimator (MLE) may be a reasonable choice for it. If there exists q_{j0} such that $q_{j1} = \cdots = q_{jn_j} = q_{j0}$, the MLE is given in the following closed form:

$$\tilde{\beta}_j = \mu^{-1}\left(\sum_{i=1}^{n_j} a_{ji} y_{ji} / \sum_{i=1}^{n_j} a_{ji}\right) - q_{j0}.$$

For other cases, see Appendix A.2.

3.3.3. Tuning parameter selection

It is important for a penalized estimation, such as GFL estimation, to select a tuning parameter, which, in this paper, is represented as λ in (1.2). Because λ adjusts the strength of the penalty against a model fitting, we need to select a good value of λ in order to obtain a good estimator. The optimal value of λ is commonly selected from candidates based on the minimization of, e.g., cross-validation and a model selection criterion. For a given $\lambda_{\max}$, candidates for λ are selected from the interval $[0, \lambda_{\max}]$. Following Ohishi *et al.* (2021), $\lambda_{\max}$ is defined by a value such that all β_j ($j \in \{1, \ldots, m\}$) are updated as $\hat{\beta}_{\max}$ when a current solution of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}}_{\max} = \hat{\beta}_{\max}\mathbf{1}_m$, where $\hat{\boldsymbol{\beta}}_{\max}$ is the MLE under $\boldsymbol{\beta} = \beta\mathbf{1}_m$ (see Appendix A.2) and $\mathbf{1}_m$ is the m-dimensional vector of ones. When a current solution of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}}_{\max}$, the discussion in Subsection 3.2 gives the condition that β_j is updated as $\hat{\beta}_{\max}$ as

$$0 \in \partial L_j(\hat{\beta}_{\max}) \Longleftrightarrow A_j - 2\lambda w_j^\dagger \leq 0 \leq A_j + 2\lambda w_j^\dagger,$$
$$A_j = \sum_{i=1}^{n_j} a_{ji}\dot{h}(\hat{\beta}_{\max} + q_{ji})\left\{\mu(\hat{\beta}_{\max} + q_{ji}) - y_{ji}\right\}, \quad w_j^\dagger = \sum_{\ell \in D_j} w_{j\ell}.$$

Hence, $\lambda_{\max}$ is given by

$$\lambda_{\max} = \max_{j \in \{1,\ldots,m\}} \frac{|A_j|}{2w_j^{\dagger}}.$$

4. Simulation

In this section, we focus on modeling using count data and establish whether our proposed method can select the true cluster from the clustering of groups through simulation. For count data, Poisson regression and NB2 are often used. Hence, we compare the performance of the two approaches for various settings of the dispersion parameter. Note that GFL for Poisson regression has already been proposed by Yamamura *et al.* (2023) and that our contribution is to apply GFL to NB2. Note, too, that simulation studies were not conducted in Yamamura *et al.* (2023).

Let m^* be the number of true clusters and $E_k^* \subset \{1,\ldots,m\}$ ($k \in \{1,\ldots,m^*\}$) be an index set specifying groups in the kth true cluster. Then, we generate simulation data from

$$y_{ji} \sim NB(k, 1/\phi) \quad (j \in E_k^*,\ i \in \{1,\ldots,n_j\}).$$

We consider four cases of m and m^* as $(m, m^*) = (10, 3), (10, 6), (20, 6), (20, 12)$, and use the same settings as Ohishi *et al.* (2021) for adjacent relationships of m groups and true clusters. The sample sizes for each group are common, i.e., $n_1 = \cdots = n_m = n_0$. Furthermore, the estimation of ϕ, the definition of the penalty weights, and the candidates for λ follow Subsection 3.3, and the optimal value of λ is selected based on the minimization of BIC (Schwarz, 1978) from 100 candidates. Here, the performance of the two methods is evaluated by Monte Carlo simulation with 1,000 iterations.

Tables 5 and 6 summarize the results for $m = 10, 20$, respectively, in which SP is the selection probability (%) of the true cluster, $\hat{\phi}$ is the Pearson estimator of ϕ, and time is runtime (in seconds). First, focusing on $\phi = 0$, i.e., the true model according to the Poisson distribution, the value of SP using Poisson regression approaches 100% as n_0 increases. Furthermore, we can say that Poisson regression provides good estimation since $\hat{\phi}$ is approximately 1. On the other hand, NB2 is unable to select the true cluster. The reason for this may be that the dispersion parameter in the negative binomial distribution is positive. Next, we focus on $\phi > 0$. Here, Poisson regression produced overdispersion since $\hat{\phi}$ is larger than 1, and, hence, it is unable to select the true cluster. On the other hand, the SP value for NB2 approaches 100% as n_0 increases. Furthermore, $\hat{\phi}$ is roughly the true value, indicating that NB2 can provide good estimation. Finally, it is apparent that Poisson regression is always faster than NB2. The reason for this may be that Poisson regression requires only the estimation of $\boldsymbol{\beta}$, whereas NB2

Table 5. Simulation results when $m = 10$

			SP (%)		$\hat{\phi}$		time (s)	
m^*	ϕ	n_0	Poisson	NB2	Poisson	NB2	Poissosn	NB2
3	0	100	67.7	**70.0**	0.99	0.03	**0.31**	1.37
		500	**92.2**	77.7	1.00	0.02	**0.50**	2.48
		1,000	**97.6**	65.3	1.00	0.02	**0.70**	3.72
		5,000	**99.6**	48.9	1.00	0.01	**2.56**	14.42
		10,000	**99.7**	52.3	1.00	0.01	**5.49**	30.84
	1	100	20.5	**43.3**	2.87	0.99	**0.30**	0.51
		500	49.3	**84.8**	2.89	0.99	**0.48**	0.89
		1,000	60.3	**93.1**	2.90	1.00	**0.68**	1.34
		5,000	76.1	**98.8**	2.90	1.00	**2.52**	5.05
		10,000	80.0	**99.3**	2.90	1.00	**5.27**	10.15
	3	100	3.6	**18.0**	6.54	2.97	**0.29**	0.57
		500	11.4	**68.5**	6.67	2.99	**0.47**	1.04
		1,000	18.8	**84.9**	6.69	3.00	**0.67**	1.64
		5,000	32.4	**97.8**	6.70	3.00	**2.46**	6.73
		10,000	36.2	**98.5**	6.70	3.00	**5.16**	14.05
6	0	100	52.4	**52.5**	0.99	0.02	**0.34**	1.20
		500	**87.0**	67.9	1.00	0.02	**0.56**	2.28
		1,000	**95.0**	61.9	1.00	0.01	**0.80**	3.51
		5,000	**99.5**	64.2	1.00	0.01	**2.99**	14.51
		10,000	**99.9**	70.2	1.00	0.01	**6.48**	30.93
	1	100	8.4	**12.2**	4.23	0.99	**0.32**	0.57
		500	31.3	**64.4**	4.28	0.99	**0.54**	1.06
		1,000	44.8	**81.7**	4.29	1.00	**0.77**	1.63
		5,000	64.1	**97.0**	4.30	1.00	**2.92**	6.01
		10,000	71.0	**99.1**	4.30	1.00	**6.10**	11.95
	3	100	1.5	**2.6**	10.57	2.96	**0.31**	0.61
		500	9.0	**29.2**	10.85	2.99	**0.50**	1.22
		1,000	15.1	**57.6**	10.86	2.99	**0.72**	1.91
		5,000	26.8	**92.3**	10.89	3.00	**2.74**	7.57
		10,000	31.2	**96.6**	10.90	3.00	**5.77**	15.72

requires repeatedly estimating $\boldsymbol{\beta}$ and ϕ alternately. We can conclude from this simulation that Poisson regression is better when the true model is according to a Poisson distribution and that NB2 can effectively deal with overdispersion in Poisson regression.

Table 6. Simulation results when $m = 20$

			SP (%)		$\hat{\phi}$		time (s)	
m^*	ϕ	n_0	Poisson	NB2	Poisson	NB2	Poissosn	NB2
6	0	100	**42.1**	29.1	0.99	0.02	**0.74**	3.86
		500	**86.1**	57.7	1.00	0.01	**1.14**	5.13
		1,000	**94.6**	47.8	1.00	0.01	**1.59**	7.35
		5,000	**98.6**	46.7	1.00	0.01	**6.08**	30.11
		10,000	**99.7**	47.2	1.00	0.01	**13.68**	68.28
	1	100	1.1	**6.2**	4.41	0.99	**0.72**	1.15
		500	13.5	**53.8**	4.44	1.00	**1.12**	2.00
		1,000	22.0	**75.5**	4.44	1.00	**1.59**	2.98
		5,000	42.8	**97.1**	4.45	1.00	**5.79**	11.40
		10,000	46.5	**99.4**	4.45	1.00	**12.96**	23.71
	3	100	0.0	**0.2**	11.02	2.98	**0.74**	1.33
		500	0.5	**19.6**	11.28	2.99	**1.16**	2.41
		1,000	1.4	**44.5**	11.31	2.99	**1.68**	3.77
		5,000	4.8	**90.2**	11.35	3.00	**6.25**	15.81
		10,000	5.1	**97.1**	11.35	3.00	**13.81**	34.25
12	0	100	**12.3**	9.7	0.99	0.01	**0.75**	3.87
		500	**72.9**	68.2	1.00	0.01	**1.19**	4.76
		1,000	**89.4**	78.7	1.00	0.01	**1.67**	6.36
		5,000	**98.5**	90.4	1.00	0.01	**6.52**	26.50
		10,000	**99.4**	93.4	1.00	0.01	**14.93**	61.25
	1	100	**0.1**	**0.1**	7.44	0.99	**0.69**	1.04
		500	1.3	**3.8**	7.54	1.00	**1.11**	1.90
		1,000	5.3	**15.0**	7.54	1.00	**1.58**	2.91
		5,000	19.1	**78.6**	7.55	1.00	**6.02**	11.56
		10,000	25.8	**95.1**	7.55	1.00	**13.97**	24.21
	3	100	**0.0**	**0.0**	20.07	2.99	**0.69**	1.19
		500	**0.0**	**0.0**	20.55	3.00	**1.09**	2.23
		1,000	0.2	**1.3**	20.59	3.00	**1.56**	3.47
		5,000	2.0	**37.6**	20.64	3.00	**5.71**	14.34
		10,000	2.5	**68.3**	20.65	3.00	**12.94**	31.52

5. Real data example

In this section, we apply our method to the estimation of spatio-temporal trend using real crime data. The data consist of the number of recognized crimes committed in the Tokyo area as collected by the Metropolitan Police Department, available at TOKYO OPEN DATA

(https://portal.data.metro.tokyo.lg.jp/)[1]. Although these data were aggregated for each chou-chou (level 4), the finest regional division, we integrate the data for each chou-oaza (level 3) and apply our method by regarding level 3 as individuals and the city (level 2) as the group (see Figure 1). There are 53 groups as a division of space, and spatial adjacency is

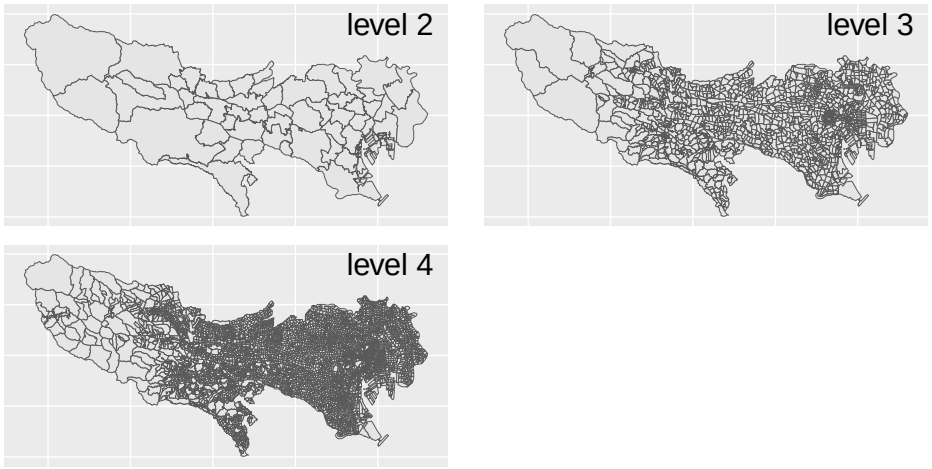

Figure 1. Divisions of Tokyo

defined by the regional relationships of level 2. We use six years of data, from 2017 to 2022. The sample size is $n = 9{,}570$. Temporal adjacency is defined using a chain graph for the six time points. According to Yamamura *et al.* (2021), we can define adjacent spatio-temporal relationships for $m = 318$ $(= 53 \times 6)$ groups by combining spatial and temporal adjacencies. Furthermore, following Yamamura *et al.* (2023), we use population as a variable for the offset. The population data were obtained from the results of the population census, as provided in e-Stat (https://www.e-stat.go.jp/en). Since the population census is conducted every five years, we use the population in 2015 for the crimes in 2017 to 2019 and the population in 2020 for the crimes in 2020 to 2022.

In this analysis, we apply our method to the above crime data, with $n = 9{,}570$ individuals aggregated into $m = 318$ groups, and estimate the spatio-temporal trends in the data. Specifically, y_{ji}, the number of crimes in the ith region of the jth group, is modeled based on the Poisson and negative binomial distributions, respectively, as

[1] We arranged and used the following production: Tokyo Metropolitan Government & Metropolitan Police Department. The number of recognized cases by region, crime type, and method (yearly total; in Japanese), https://creativecommons.org/licenses/by/4.0/deed.en.

$$y_{ji} \sim P\left(\exp(\beta_j + q_{ji})\right), \quad y_{ji} \sim NB\left(\exp(\beta_j + q_{ji}), \psi\right),$$

where q_{ji} is a logarithm transformation of the population and canonical and log-links are used, respectively. Estimation of the dispersion parameter, the setting of penalty weights, and the candidates for the tuning parameter follow Subsection 3.3. The optimal tuning parameter is selected from 100 candidates based on the minimization of BIC. Table 7 summarizes the estimation results. The $\hat{\phi}$ indicates the Pearson estimator of the dispersion parameter. Since the

Table 7. Summary of real data example

	$\hat{\phi}$	cluster	time (s)
Poisson	2728.28	160	12.48
NB2	18.77	109	111.92

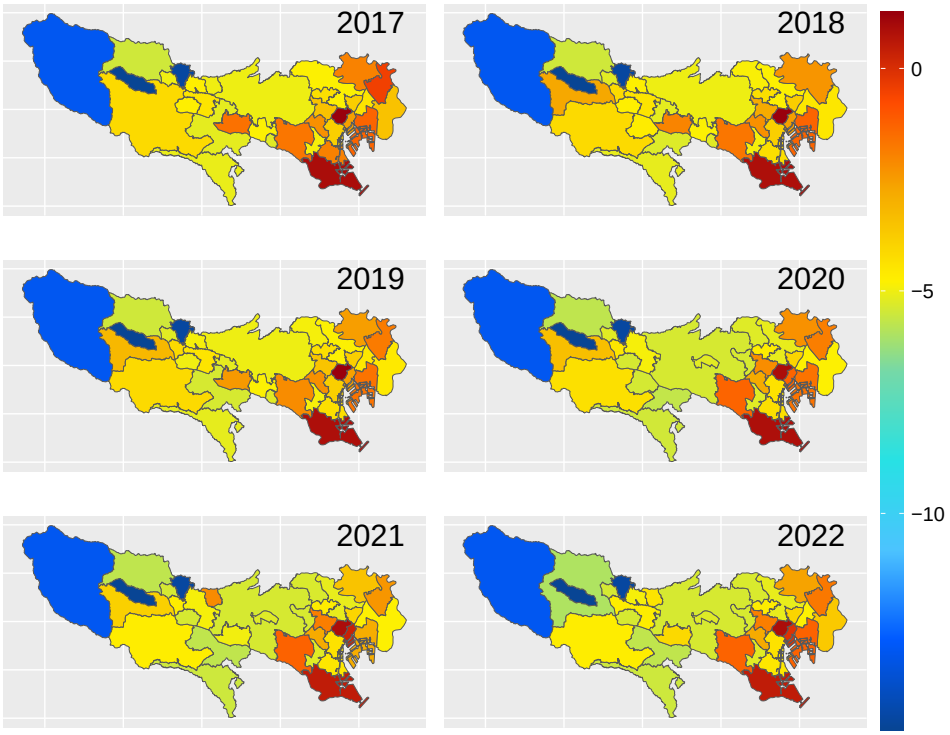

Figure 2. GFL estimates of β by NB2

value of $\hat{\phi}$ in the Poisson regression is far larger than 1, there is overdispersion, and we can say that using Poisson regression is inappropriate. To cope with this overdispersion, we adopted NB2. The cluster value in the table indicates the number of clusters using GFL. Poisson regression and NB2 clustered the $m = 318$ groups into 160 and 109 groups, respectively. Figure 2 is a yearly choropleth map of the GFL estimates of $\boldsymbol{\beta}$ using NB2. The map shows that the larger the value, the easier it is for crime to occur, and that the smaller the value, the harder it is. As in this figure, we can visualize the variation of trend with respect to time and space.

6. Conclusion

To unify models based on a variety of distributions, we proposed a coordinate descent algorithm to obtain GFL estimators for GLMs. Although Yamamura *et al.* (2021), Ohishi *et al.* (2022), and Yamamura *et al.* (2023) dealt with GFL for the binomial and Poisson distributions, our method is more general, covering both these distributions and others. The proposed algorithm repeats the partial update of parameters and directly solves sub-problems without any approximations of the objective function. In many cases, the solution can be updated in closed form. Indeed, in the ordinary situation where a canonical link is used and there is no offset, we can always update the solution in closed form. Moreover, even when an explicit update is impossible, we can easily update the solution using a simple numerical search since the interval containing the solution can be specified. Hence, our algorithm can efficiently search the solution.

Acknowledgment

The author thank Prof. Hirokazu Yanagihara of Hiroshima University for his many helpful comments and FORTE Science Communications (`https://www.forte-science.co.jp/`) for English language editing. This work was partially supported by JSPS KAKENHI Grant Number JP20H04151, JP21K13834, JSPS Bilateral Program Grant Number JPJSBP120219927, and ISM Cooperative Research Program (2023-ISMCRP-4105).

References

Algamal, Z. Y. (2018). Developing a ridge estimator for the gamma regression model. *J. Chemometr.*, **32**, e3054.

Dunn, P. K. & Smyth, G. K. (2018). *Generalized Linear Models With Examples in R*. Springer Nature. New York.

Friedman, J., Hastie, T., Höfling, H. & Tibshirani, R. (2007). Pathwise coordinate optimization. *Ann. Appl. Stat.*, **1**, 302–332.

Gardner, W., Mulvey, E. P. & Shaw, E. C. (1995). Regression analyses of counts and rates: Poisson, overdispersed poisson, and negative binomial models. *Psychol. Bull.*, **118**, 392–404.

Hilbe, J. M. (2011). *Negative Binomial Regression 2nd Edition*. Cambridge University Press. Cambridge.

Höfling, H., Binder, H. & Schumacher, M. (2010). A coordinate-wise optimization algorithm for the fused Lasso. arXiv 1011.6409v1.

Nelder, J. A. & Wedderburn, R. W. M. (1972). Generalized linear models. *J. Roy. Statist. Soc. Ser. A*, **135**, 370–384.

Ohishi, M., Fukui, K., Okamura, K., Itoh, Y. & Yanagihara, H. (2021). Coordinate optimization for generalized fused Lasso. *Comm. Statist. Theory Methods*, **50**, 5955–5973.

Ohishi, M., Yamamura, M. & Yanagihara, H. (2022). Coordinate descent algorithm of generalized fused Lasso logistic regression for multivariate trend filtering. *Jpn. J. Stat. Data Sci.*, **5**, 535–551.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press. New Jersey.

Schwarz, G. (1978). Estimating the dimension of a model. *Ann. Statist.*, **6**, 461–464.

Tibshirani, R. J. (2014). Adaptive piecewise polynomial estimation via trend filtering. *Ann. Statist.*, **42**, 285–323.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J. & Knight, K. (2005). Sparsity and smoothness via the fused Lasso. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, **67**, 91–108.

Ver Hoef, J. M. & Boveng, P. L. (2007). Quasi-Poisson vs. negative binomial regression: how should we model overdispersed count data?. *Ecology*, **88**, 2766–2772.

Xin, B., Kawahara, Y., Wang, Y. & Gao, W. (2014). Efficient generalized fused Lasso and its application to the diagnosis of Alzheimer's disease. Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence. AAAI press. California. 2163–2169.

Yamamura, M., Ohishi, M. & Yanagihara, H. (2021). Spatio-temporal adaptive fused Lasso for proportion data. I. Czarnowski, R. J. Howlett & L. C. Jain, eds, Intelligent Decision Technologies. Springer Singapore. Singapore. 479–489.

Yamamura, M., Ohishi, M. & Yanagihara, H. (2023). Spatio-temporal analysis of rates derived from count data using generalized fused Lasso. I. Czarnowski, R. J. Howlett & L. C. Jain, eds, Intelligent Decision Technologies. Springer Singapore. Singapore. 225–234.

Zou, H. (2006). The adaptive Lasso and its oracle properties. *J. Amer. Statist. Assoc.*, **101**, 1418–1429.

Appendix

A.1. Proof of Corollary 1

Suppose that for all $\ell \in \{0, 1, \ldots, r\}$, the statement

$$\forall x \in R_\ell,\ \ddot{f}_\ell(x) > 0,$$

is true and that (3.2) holds. Then, $\dot{f}_\ell(x)$ is strictly increasing on R_ℓ and hence, $f_\ell(x)$ is strictly convex. Moreover, there is the following relationship among a derivative and one-sided derivatives:

$$\lim_{x \to z_\ell - 0} \dot{f}_{\ell-1}(x) = g_-(z_\ell) < g_+(z_\ell) = \lim_{x \to z_\ell + 0} \dot{f}_\ell(x) \quad (\ell = 1, \ldots, r).$$

This fact and (3.2) imply the strict convexity of $f(x)$ on $\mathbb{R}$ and hence, the minimizer uniquely exists.

A.2. Derivation of MLEs

We first describe the derivation of the MLE of β_j. For distributions with a convex likelihood function, the MLE is obtained by solving

$$\sum_{i=1}^{n_j} a_{ji} \dot{h}(\beta_j + q_{ji}) \left\{ \mu(\beta_j + q_{ji}) - y_{ji} \right\} = 0.$$

In Tables 2 and 3, all distributions, with the exception of the inverse Gaussian distribution with log-link, have convexity. The MLE of β_j is given in closed form in the following cases:

$$\tilde{\beta}_j = \begin{cases} \mu^{-1}\left(\sum_{i=1}^{n_j} a_{ji} y_{ji} / \sum_{i=1}^{n_j} a_{ji} \right) - q_{j0} & (q_{j1} = \cdots = q_{jn_j} = q_{j0}) \\ n_j^{-1} \sum_{i=1}^{n_j} (y_{ji} - q_{ji}) & \text{(Gaussian)} \\ \log \sum_{i=1}^{n_j} y_{ji} \dot{h}(q_{ji}) - \log \sum_{i=1}^{n_j} \dot{h}(q_{ji}) \exp(q_{ji}) & \text{(Poisson or Gamma with log-link)} \end{cases}.$$

Other distributions, including the inverse Gaussian distribution with log-link, require a numerical search. Furthermore, the negative binomial distribution requires the repeated updating of $\boldsymbol{\beta}$ and ϕ alternately.

Next, we describe the derivation of $\beta_{\max}$. The $\beta_{\max}$ is the MLE of β under $\boldsymbol{\beta} = \beta \mathbf{1}_m$, and for distributions with a convex likelihood function, its value is obtained by solving

$$\sum_{j=1}^{m} \sum_{i=1}^{n_j} a_{ji} \dot{h}(\beta + q_{ji}) \left\{ \mu(\beta + q_{ji}) - y_{ji} \right\} = 0.$$

Notice that this is essentially equal to the derivation of the MLE of β_j. Hence, $\beta_{\max}$ is given in closed form in the following cases:

$$\hat{\beta}_{\max} = \begin{cases} \mu^{-1}\left(\sum_{j=1}^{m} \sum_{i=1}^{n_j} a_{ji} y_{ji} / \sum_{j=1}^{m} \sum_{i=1}^{n_j} a_{ji} \right) - q_0 & (q_{ji} = q_0 \ (\forall j, i)) \\ n^{-1} \sum_{j=1}^{m} \sum_{i=1}^{n_j} (y_{ji} - q_{ji}) & \text{(Gaussian)} \\ \log \sum_{j=1}^{m} \sum_{i=1}^{n_j} y_{ji} \dot{h}(q_{ji}) - \log \sum_{j=1}^{m} \sum_{i=1}^{n_j} \dot{h}(q_{ji}) \exp(q_{ji}) & \text{(Poisson or Gamma with log-link)} \end{cases}.$$