Computational Statistics and Data Analysis 159 (2021) 107217

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Generalized k-means in GLMs with applications to the outbreak of COVID-19 in the United States

Tonglin Zhang [a,*], Ge Lin [b]

[a] Department of Statistics, Purdue University, 250 North University Street, West Lafayette, IN 47907-2066, USA
[b] Department of Environmental and Occupational Health, University of Nevada Las Vegas, Las Vegas, NV 89154, USA

ARTICLE INFO

Article history:
Received 9 August 2020
Received in revised form 28 February 2021
Accepted 28 February 2021
Available online 10 March 2021

Keywords:
Clustering
COVID-19
Exponential family distributions
Generalized k-means
Generalized information criterion (GIC)
Generalized linear models (GLMs)

ABSTRACT

Generalized k-means can be combined with any similarity or dissimilarity measure for clustering. Using the well known likelihood ratio or F-statistic as the dissimilarity measure, a generalized k-means method is proposed to group generalized linear models (GLMs) for exponential family distributions. Given the number of clusters k, the proposed method is established by the uniform most powerful unbiased (UMPU) test statistic for the comparison between GLMs. If k is unknown, then the proposed method can be combined with generalized lformation criterion (GIC) to automatically select the best k for clustering. Both AIC and BIC are investigated as special cases of GIC. Theoretical and simulation results show that the number of clusters can be correctly identified by BIC but not AIC. The proposed method is applied to the state-level daily COVID-19 data in the United States, and it identifies 6 clusters. A further study shows that the models between clusters are significantly different from each other, which confirms the result with 6 clusters.

1. Introduction

Generalized k-means, including both k-means and k-medians as special cases, can be incorporated with any similarity or dissimilarity measure for grouping objects. The similarity or dissimilarity measure can be very general. In this work, we choose the dissimilarity measure as the well known likelihood ratio or F-statistic and the objects as statistical models for exponential family distributions, such that the resulting method can be used to group generalized linear models (GLMs). In particular, we assume that each object is composed by a vector for a response and a design matrix for explanatory variables, and a GLM has been established within each object. The linear component of the GLM provides the relationship between the expected value of the response and the explanatory variables within the objects. The significance of regression coefficients for the explanatory variables is determined by the likelihood ratio statistic, which means that we can combine the likelihood ratio test with the generalized k-means. The current research develops the method and uses it to group the patterns for the state-level daily confirmed cases of COVID-19 in the United States.

The outbreak of COVID-19 has become a worldwide ongoing pandemic since March 2020. According to the website of the World Health Organization (WHO), until January 31 2021, the outbreak has affected over 200 countries and territories with more that 100 million confirmed cases and 2 million deaths in the entire world. The most serious country is the United States. It has over 25 million confirmed cases and 440 thousand deaths. To understand the outbreak in the United

[*] Corresponding author.
E-mail addresses: tlzhang@purdue.edu (T. Zhang), ge.kan@unlv.edu (G. Lin).
URL: https://www.stat.purdue.edu/~tlzhang (T. Zhang).

https://doi.org/10.1016/j.csda.2021.107217
0167-9473/© 2021 Elsevier B.V. All rights reserved.

States in the early period, we compare daily patterns of new cases in the fifty states and Washington DC until July 31 2020. We find that some of these patterns are similar to each other and some of these are far away from each other, implying that we can carry out a clustering analysis to group these patterns. As statistical models are involved, we use the generalized k-means. We adopt the likelihood ratio or F-statistic because it is induced by the standard uniformly most powerful unbiased (UMPU) test for exponential family distributions. Based on theory of the UMPU test, the proposed method should be more powerful than the convenient method based on k-means directly on regression coefficients. This is confirmed by our simulation studies.

Clustering is one of the most popular unsupervised statistical learning methods for unknown structures. Clustering methods are often carried out by similarity or dissimilarity measures between objects. Their goal is to group the objects into a few clusters. The definition of objects can be very general. They can be observations, images, or statistical models. The purpose of clustering is to make objects within clusters mostly homogeneous and objects between clusters mostly heterogeneous. In the literature, one of the most well known clustering methods is the k-means. For objects from a Euclidean space, the method assigns each of them to the cluster with the nearest mean. Based on a given k, it provides k clusters according to k centers. The k centers are solved by minimizing the sum-of-squares (SSQ) criterion, formulated by the Euclidean distance between the objects. Theoretically, the SSQ criterion in the k-means can be replaced by any similarity or dissimilarity measure, leading to the generalized k-means (Bock, 2008; Soheily-Khah et al., 2016). Because the choice of the dissimilarity measure is flexible, generalized k-means can be combined with any divergence measure, including the UMPU test statistics.

Many clustering methods have been proposed in the literature. Examples include hierarchical clustering (Zhao and Karypis, 2005), fuzzy clustering (Trauwaert et al., 1991), density-based clustering (Kriegel et al., 2001), model-based clustering, and partitioning clustering. Model-based clustering is usually carried out by EM algorithms or Bayesian methods under the framework of mixture models (Fraley and Raftery, 2002; Lau and Green, 2007). Partitioning clustering can be interpreted by the centroidal Voronoi tessellation method in mathematics (Du and Wong, 2002). It can be further specified to k-means (Forgy, 1965; Hartigan and Wong, 1979; Lloyd, 1982; MacQueen, 1967), k-medians (Charikar and Guha, 2002), and k-modes (Goyal and Aggarwal, 2017), where k-means is the most popular. To implement those, one needs to express observations of the data in a metric space, such that a distance measure can be defined. Several approaches have been developed to specify the distance measure. A review of these can be found in Johnson and Wichern (2002), p. 670.

Challenges appear in grouping daily patterns for the state-level COVID-19 data in the United States. Suppose that the daily patterns have been fitted by statistical models (e.g., GLMs) with the response as daily confirmed cases and explanatory variables as certain functions of time. The interest is to know whether models for individual states can be grouped into a few clusters. At least, two other methods can be used. The first is the direct usage of an existing clustering method on estimates of coefficients. A concern may arise because it is hard to address variability in estimates of coefficients. The second is the usage of mixture models, which often leads to EM algorithms for mixture structures (Qin and Self, 2006). Here, we propose another method. We use a likelihood ratio or an F-statistic as the dissimilarity measure in the generalized k-means. Because they are formulated by the UMPU test, the resulting method should be more powerful than any other method theoretically. To verify this, we compare our method with the other two methods by simulation studies. We find that our method has lower clustering object error (OE) rates than our competitors.

We propose our method based on a known k at the beginning. When k is unknown, we use GIC to select the best k. We specify it to both BIC and AIC. We find that BIC is more reliable than AIC in selecting number of clusters. Therefore, we recommend using our BIC selector. To implement our method to the COVID-19 data in the United States, we have to define an unsaturated clustering problem. In particular, we partition the coefficient vector into two sub-vectors. The first sub-vector does not contain any information of time. Therefore, we only need to study the second sub-vector. The goal is to know whether time variations between these models are similar. This problem can be partially reflected by Fig. 1. Suppose that six regression lines are compared. The intercepts do not contain any time information. We allow them to vary within clusters. We restrict the generalized k-means on the slopes only, leading to two clusters. Based on our intuition, we believe that the unsaturated clustering problem can also be carried out by mixture models with EM algorithms. Because our method is developed based on the UMPU test, it should be more powerful than any other methods.

The article is organized as follows. In Section 2, we propose our method. In Section 3, we study theoretical properties of our method. In Section 4, we evaluate our method with the comparison to a few previous methods by simulation studies. In Section 5, we implement our method to the state-level COVID-19 data in the United States. In Section 6, we provide a discussion.

2. Method

We propose our method based on a known k in Section 2.1. The method is combined with GIC (Zhang et al., 2010) to select the best k when k is unknown, and this is introduced in Section 2.2. In Section 2.3, we specify our method to regression models for normal data and loglinear models for Poisson data. These models are treated as special cases of GLMs. The loglinear model for Poisson data can be extended to models with overdispersion for quasi-Poisson data, and this is used in analysis of the state-level COVID-19 data in the United States.

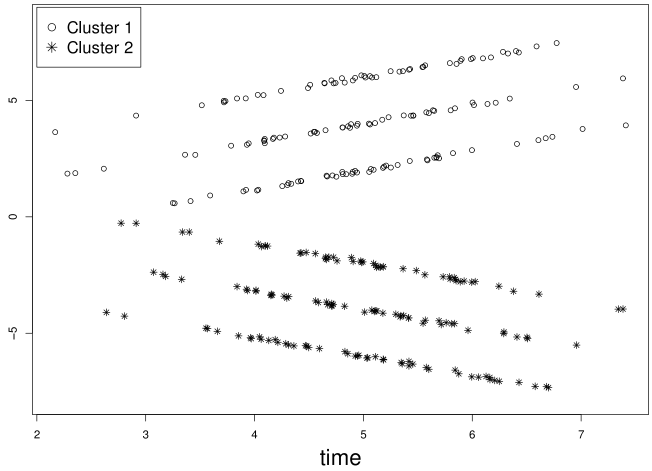

Fig. 1. Generalized k-means clustering for six regression lines.

2.1. Generalized k-means in GLMs

The goal of clustering is to partition a set of N objects, denoted by $\mathcal{S} = \{z_1, \ldots, z_N\}$, into several non-empty subsets or clusters, such that the objects within clusters are mostly homogeneous and the objects between clusters are mostly heterogeneous. If the objects are points from a Euclidean space, then the k-means can be used. It partitions $\mathcal{S}$ into k distinct clusters denoted by $\mathcal{C} = \{C_1, \ldots, C_k\}$ with $\mathcal{C}$ given by

$$\mathcal{C} = \operatorname*{argmin}_{\mathcal{C}} \sum_{s=1}^{k} \sum_{i \in C_s} \|z_i - c_s\|^2, \tag{1}$$

where c_s is the center of C_s. The right-hand side of (1) is called the SSQ criterion in the k-means. The generalized k means is induced if the SSQ criterion is replaced by any similarity or dissimilarity measure. In particular, let $d(z, C)$ be a selected dissimilarity measure with z representing an object and C representing a cluster. The generalized k-means solves $\mathcal{C}$ by

$$\mathcal{C} = \operatorname*{argmin}_{\mathcal{C}} \sum_{s=1}^{k} \sum_{i \in C_s} d(z_i, C_s). \tag{2}$$

If z_i are points in a Euclidean space, then generalized k-means becomes the k-means by choosing $d(z_i, C_s) = \|z_i - c_s\|^2$. This can also be the k-medians if $d(z_i, C_s) = \|z_i - c_s\|_1$ is used. Furthermore, the generalized k-means can also be implemented by adding a penalty function in the SSQ criterion. This can induce the convex clustering problem studied by Chen et al. (2016), Chi and Lange (2015), Lindsten et al. (2011) and Hocking et al. (2011) in the literature.

We find that $d(z, C)$ in (2) can be specified as the UMPU test statistic for grouping statistical models. In this work, we restrict our attention on GLMs for exponential family distributions, which can be linear models for normal data or loglinear models for Poisson data. The task of our method is to group the GLMs into a number of clusters.

Suppose that z_i contains a response vector $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{in_i})^\top$ and a design matrix $\mathbf{X}_i = (\boldsymbol{x}_{i1}^\top, \ldots, \boldsymbol{x}_{in_i}^\top)^\top$, such that the sample size of the entire data is $n = \sum_{i=1}^{N} n_i$. In z_i, $y_{i1}, \ldots, y_{in_i}$ are independently collected from an exponential family distribution with the probability mass function (PMF) or the probability density function (PDF) as

$$f(y_{ij}) = \exp\left[\frac{y_{ij}\theta_{ij} - b(\theta_{ij})}{a(\phi)} + c(y_{ij}, \phi)\right], \tag{3}$$

where θ_{ij} is a canonical parameter representing the location and ϕ is a dispersion parameter representing the scale. The linear component η_{ij} is related to explanatory variables by $\eta_{ij} = \boldsymbol{x}_{ij}^\top \boldsymbol{\beta}_i$. The link function $g(\cdot)$ connects $\mu_{ij} = \mathrm{E}(y_{ij}) = b'(\theta_{ij})$ and η_{ij} through

$$\eta_{ij} = g(\mu_{ij}) = g[b'(\theta_{ij})] = \boldsymbol{x}_{ij}^\top \boldsymbol{\beta}_i, \tag{4}$$

for all $i \in \{1, \ldots, N\}$ and $j \in \{1, \ldots, n_i\}$, where $\theta_{ij} = h(\boldsymbol{x}_{ij}^\top \boldsymbol{\beta}_i)$ is the inverse function obtained by (4). In (3), there is $\mathrm{V}(y_{ij}) = a(\phi)v(\mu_{ij})$, where $v(\mu) = b''\{h^{-1}[g(\mu)]\}$ is the variance function. If the canonical link is used, then (4) becomes $\eta_{ij} = \theta_{ij} = g(\mu_{ij}) = \boldsymbol{x}_{ij}^\top \beta_i$, implying that $h(\cdot)$ is the identity function.

The MLEs of $\boldsymbol{\beta}_i$, denoted by $\hat{\boldsymbol{\beta}}_i$, can only be solved numerically if the distribution is not normal. A popular and well known algorithm is the iteratively reweighted least squares (IRWLS) (Green, 1984). The IRWLS is equivalent to the

Fisher scoring algorithm. It is identical to the Newton–Raphson algorithm under the canonical link. After $\hat{\boldsymbol{\beta}}_i$ is derived, a straightforward method is to estimate ϕ by moment estimation (McCullagh, 1983) as

$$a(\hat{\phi}) = \frac{1}{df}\sum_{i=1}^{N}\sum_{j=1}^{n_i}\frac{(y_{ij}-\hat{\mu}_{ij})^2}{b''[h(\boldsymbol{x}_{ij}^\top\hat{\boldsymbol{\beta}}_i)]}, \tag{5}$$

where $\hat{\mu}_{ij} = b'[h(\boldsymbol{x}_{ij}^\top\hat{\boldsymbol{\beta}}_i)]$ and df is the residual degrees of freedom. If ϕ is not present in (3), then (5) is not needed. This occurs in Bernoulli, binomial, and Poisson models. The IRWLS is the standard algorithm for fitting GLMs, which has been adopted by many software packages, such as R, SAS, and Python.

Our interest is to group $\boldsymbol{\beta}_i$ into a few clusters, such that we have $\boldsymbol{\beta}_i = \boldsymbol{\beta}_{i'}$ if objects i and i' are in the same cluster or $\boldsymbol{\beta}_i \neq \boldsymbol{\beta}_{i'}$ otherwise. The regression version of this problem has been previously investigated in gene expressions by an EM algorithm for Gaussian mixture models (Qin and Self, 2006). Their interest is to know whether the entire coefficient vectors can be partitioned into a few clusters. In our method, we allow a few components of $\boldsymbol{\beta}_i$ to be different within clusters. Therefore, we only need to partition the objects based on the remaining components.

Suppose that (4) is expressed as

$$\eta_{ij} = \boldsymbol{x}_{ij1}^\top\boldsymbol{\beta}_{i1} + \boldsymbol{x}_{ij2}^\top\boldsymbol{\beta}_{i2}, \tag{6}$$

where $\boldsymbol{x}_{ij} = (\boldsymbol{x}_{ij1}^\top, \boldsymbol{x}_{ij2}^\top)^\top$ and $\boldsymbol{\beta}_i = (\boldsymbol{\beta}_{i1}^\top, \boldsymbol{\beta}_{i2}^\top)^\top$. We want to know whether $\boldsymbol{\beta}_{i2}$ can be grouped into a few clusters, such that we only need $\boldsymbol{\beta}_{i2} = \boldsymbol{\beta}_{i'2}$ if objects i and i' are in the same cluster or $\boldsymbol{\beta}_{i2} \neq \boldsymbol{\beta}_{i'2}$ otherwise. Based on a given $\mathcal{C}$, our clustering model is

$$g(\mu_{ij}) = \boldsymbol{x}_{ij1}^\top\boldsymbol{\beta}_{i1} + \boldsymbol{x}_{ij2}^\top\boldsymbol{\beta}_{s2}, \tag{7}$$

for $z_i \in C_s$. We call (7) the *unsaturated clustering problem*. The *saturated clustering problem* is induced if $\boldsymbol{\beta}_{i1}$ is absent in (7). As the choice of $\boldsymbol{x}_{ij1}$ and $\boldsymbol{x}_{ij2}$ is flexible in (7), our method can be used to group GLMs based on any arbitrary sub-vectors of $\boldsymbol{\beta}_i$. In practice, the choice of $\boldsymbol{\beta}_{i1}$ and $\boldsymbol{\beta}_{i2}$ depends on interpretations or the interest of the applications. If no information is provided, then we can simply study the saturated clustering problem.

The best measure for the difference between statistical models under (6) or (7) is the UMPU test statistic. The UMPU test is optimal in finite samples for the comparison between two statistical models. It is also optimal in the simultaneous comparison between many statistical models. The UMPU test is more powerful than any other method with the same type I error probabilities. This motivates us to use the UMPU test statistic to define the similarity measure in (2).

We want to start with a nice initial $\mathcal{C}$ in our generalized k-means. We do not follow the usual k-means algorithms, as they select the initial $\mathcal{C}$ randomly. Instead, we want to choose the initial $\mathcal{C}$ as heterogeneous as possible. This has been previously used in the initialization of traditional k-means with observations from a Euclidean space based on a complete weighted graph (Gonzalez, 1985). It has also been used in the initialization of the k-means++ proposed by Arthur and Vassilvitskii (2007), who point out that k-means++ generally outperforms k-means with random initial centers in terms of both accuracy and speed by substantial margins.

The goal of our initialization can be achieved by selecting k most dissimilar seeds first and then using them to generate the entire initial $\mathcal{C}$. We use a sequential approach to obtain the k seeds. At the beginning, we randomly choose the first seed z_i from $\mathcal{S}$. We denote it as z_{i_1}. We treat it as the seed for C_1. To obtain the second seed z_{i_2} for C_2, we calculate the UMPU test statistic for

$$H_0 : \boldsymbol{\beta}_i = \boldsymbol{\beta}_{i_1 2} \leftrightarrow H_1 : \boldsymbol{\beta}_{i2} \neq \boldsymbol{\beta}_{i_1 2}, \tag{8}$$

for any $i \neq i_1$. A larger value of the UMPU test statistic indicates more dissimilar between z_i and z_{i_1}. The UMPU test statistic can be either a likelihood ratio or an F-statistic. It is the likelihood ratio statistic if ϕ is absent in (3) (e.g., in binomial or Poisson regressions) or the F-statistic if ϕ is present (e.g., in linear regressions). We want z_{i_2} to be the most dissimilar to z_{i_1}. This can be achieved by maximizing the tail distribution of the UMPU test statistic, which is equivalent to minimizing the p-value. Therefore, the resulting z_{i_2} has the lowest p-value in (8).

Now, we have two seeds z_{i_1} and z_{i_2}. We want to derive the third seed z_{i_3} for C_3. We cannot use the simple UMPU test given by (8) to select z_{i_3}. Then, we incorporate the minimax principle. For each $i \neq i_1, i_2$, we calculate the UMPU test statistic for

$$H_0 : \boldsymbol{\beta}_{i2} = \boldsymbol{\beta}_{j2} \leftrightarrow H_1 : \boldsymbol{\beta}_{i2} \neq \boldsymbol{\beta}_{j2}. \tag{9}$$

For a given i, (9) contains two testing problems by taking $j = i_1$ and $j = i_2$, respectively. We want z_{i_3} to be the most dissimilar to both z_{i_1} and z_{i_2}. We can do this by minimizing the maximum of the two p-values. Then, we have z_{i_3}. Using this idea, we can obtain all seeds $z_{i_1}, \ldots, z_{i_k}$ for $C_1, \ldots, C_k$, respectively.

To finalize our initial $\mathcal{C}$, the next task is to assign the remaining objects to one of $C_1, \ldots, C_k$. We assign z_i to cluster s if it is the most similar to C_s. We need this for all $i \neq z_{i_1}, \ldots, z_{i_k}$, which can also be achieved by the UMPU test statistic given by (9) with $j \in \{i_1, \ldots, i_k\}$, respectively. We claim that z_i is the most similar to C_s if the p-value of the UMPU test is maximized at $j = s$. Then, we have our initial $\mathcal{C}$.

We next carry out an iterative method to update $\mathcal{C}$. We want to reassign every z_i to the cluster candidates with an improved result. This can also be achieved by the UMPU test. In particular, let $\tilde{\mathcal{C}}$ be the result given by the previous iteration. Then, for each $z_i \in \mathcal{S}$, there exists a unique $C_s \in \tilde{\mathcal{C}}$ such that $z_i \in C_s$. In the current iteration, we need to determine whether z_i should be kept in C_s or moved to another $C_{s'}$ with $s' \neq s$. After we do this for all z_i, we obtain an updated result. It is denoted as $\mathcal{C}$ in the current iteration. The notation will be changed to be $\tilde{\mathcal{C}}$ in the next iteration.

To derive the updated $\mathcal{C}$ based on the previous $\tilde{\mathcal{C}}$, for each $z_i \in \mathcal{S}$, we need to know whether z_i should be kept in the current C_s or moved to another $C_{s'}$. To fulfill the task, we calculate the UMPU test statistic for

$$H_0 : \boldsymbol{\beta}_{i2} = \boldsymbol{\beta}_{s''2} \leftrightarrow H_1 : \boldsymbol{\beta}_{i2} \neq \boldsymbol{\beta}_{s''2}, \tag{10}$$

for every $C_{s''} \in \tilde{\mathcal{C}}$. Because there are k cluster candidates in $\tilde{\mathcal{C}}$, we obtain k p-values of z_i. We want to reassign z_i to the most similar $C_{s'}$ by using these p-values. For every $z_i \in \mathcal{S}$, we reassign z_i to cluster candidate $C_{s'}$ if the p-value of the UMPU test statistic given by (10) is maximized at $s'' = s'$. This can involve two cluster candidates. After we use the method for all $z_i \in \mathcal{S}$, we obtain the updated $\mathcal{C}$, which becomes $\tilde{\mathcal{C}}$ in the next iteration. Although it is very unlikely, to ensure each C_s non-empty theoretically, we do not move the object with the largest p-value in current C_s to any other $C_{s'}$. Then, we have the following algorithm.

Algorithm 1 Generalized k-means in GLMs

```
Input: $\mathcal{S} = \{z_1, \dots, z_N\}$ with $z_i = \{\boldsymbol{y}_i, \mathbf{X}_i\}$
Output: $\mathcal{C} = \{C_1, \dots, C_k\}$ and the value of UMPU test statistic based on $\mathcal{C}$
1: Initialization: find distinct $z_{i_1}, \dots, z_{i_k}$ such that they are the most dissimilar, and use those to generate the initial $\mathcal{C}$.
2: procedure UPDATE ITERATIVELY
3:     For each $C_s$, compute the $p$-value of $z_i$ under (10) for every $z_i \in C_s$. The object with the largest $p$-value will be remained in $C_s$.
4:     For every other $z_i \in C_s$ that will not be remained, compute its $p$-values under (10) for every $C_{s''} \in \mathcal{C}$. Assign $z_i$ to $C_{s'}$ if the largest $p$ value is attained at $s'' = s'$.
5: end procedure
6: Output.
```

Algorithm 1 has two major stages. The second stage is given by Step 2 to Step 5, which is common in many k-means algorithms. The goal of the first stage given by Step 1 is to find the best initial $\mathcal{C}$. We want it to be as heterogeneous as possible. In the end, the algorithm provides k non-empty clusters with the value and the p-value of the UMPU test statistic based on the final partition.

The usage of Step 1 in Algorithm 1 can increase the accuracy and the speed compared to the method with the random assignment of the initial $\mathcal{C}$. This has been found in the k-means++ when data are collected from a Euclidean space (Arthur and Vassilvitskii, 2007). Because our Step 1 can be treated as an extension of the initialization in k-means++, we treat k-means++ as our method if we want to compare it with our competitors for data from Euclidean spaces. This is used in Section 4.2.

2.2. Generalized information criterion

The generalized k-means proposed in Section 2.1 cannot be used if k is unknown. To overcome the difficulty, we use the likelihood function given by Algorithm 1 to construct a penalized likelihood function, which is used in determining k if it is unknown. The penalized likelihood approach has been widely applied in variable selection problems. It is also used in clustering analysis problems (Chen et al., 2016; Chi and Lange, 2015; Hocking et al., 2011). Here, we adopt the well known GIC approach (Zhang et al., 2010) to construct our objective function with the best k obtained by optimizing the corresponding criterion.

Let $\ell(\boldsymbol{\omega}_{\mathcal{C}})$ be the loglikelihood of (7), where $\boldsymbol{\omega}_{\mathcal{C}}$ represents all of the parameters involved in the model. If the dispersion parameter is not present, then $\boldsymbol{\omega}$ is composed by $\boldsymbol{\beta}_{i1}$ and $\boldsymbol{\beta}_{s2}$ for all $i \in \{1, \dots, N\}$ and $s \in \{1, \dots, k\}$ only. It is enough for us to use $\ell(\boldsymbol{\omega}_{\mathcal{C}})$ to define the objective function in GIC. If the dispersion parameter is present, then we need to address the impact of the estimator of $a(\phi)$, because variance can be seriously underestimated in the penalized likelihood approach under the high-dimensional setting (Fan et al., 2012). We introduce our approach based on (3) without $a(\phi)$ first. We then modify it to the case when $a(\phi)$ is present.

Assume that $a(\phi)$ does not appear in (3). The GIC for (7) is defined as $\mathrm{GIC}_\kappa(\mathcal{C}) = -2\ell(\hat{\boldsymbol{\omega}}_{\mathcal{C}}) + \kappa df_{\mathcal{C}}$, where $\hat{\omega}_{\mathcal{C}}$ is the MLE of ω and $df_{\mathcal{C}}$ is the model degrees of freedom under $\mathcal{C}$, and κ is a positive number that controls the properties of GIC. If q_1 is the dimension of $\boldsymbol{\beta}_{i1}$ and q_2 is the dimension of $\boldsymbol{\beta}_{i2}$, then $df_{\mathcal{C}} = Nq_1 + kq_2$. Because N does not vary with k, we define the objective function in our GIC as

$$\mathrm{GIC}_\kappa(\mathcal{C}) = -2\ell(\hat{\boldsymbol{\omega}}_{\mathcal{C}}) + \kappa k q_2. \tag{11}$$

The best k is solved by

$$\hat{k}_\kappa = \underset{k}{\mathrm{argmin}}\{\mathrm{GIC}_\kappa(\hat{\mathcal{C}}_k)\}, \tag{12}$$

where $\hat{\mathcal{C}}_k$ is the best grouping based on the current k. The GIC given by (11) includes AIC if we choose $\kappa = 2$ or BIC if we choose $\kappa = \log n$. If these are adopted, then the solutions given by (12) are denoted by $\hat{k}_{AIC}$ and $\hat{k}_{BIC}$, respectively.

We need to estimate the dispersion parameter if it is present. Because the estimator based on the current k can be seriously biased, we recommending using $k + 1$ as the number of clusters in the computation of the estimate of $a(\phi)$. In particular, we calculate the best $\mathcal{C}$ based on the current k in the generalized k means. We use it to compute $\hat{\boldsymbol{\beta}}_{i1}$ and $\hat{\boldsymbol{\beta}}_{s2}$ for all $i \in \{1, \ldots, N\}$ and $s \in \{1, \ldots, k\}$. Next, we calculate the best $\mathcal{C}$ by setting the number of clusters equal to $k+1$ with $a(\phi)$ estimated by (5). This is analogous to the full model versus the reduced model approach in linear regression, where the variance parameter is always estimated under the full model. We treat the model with $k+1$ clusters in (7) as the full model, and the model with k clusters as the reduced model. We estimate $a(\phi)$ based on the full model but not the reduced model. After $a(\hat{\phi})$ is derived, we put it into (11) in the computation of GIC. We then use (12) to calculate the best k when $a(\phi)$ is present. This is used in our method for regression models.

2.3. Specification

In regression, (6) becomes

$$\boldsymbol{y}_i = \mathbf{X}_{i1}\boldsymbol{\beta}_{i1} + \mathbf{X}_{i2}\boldsymbol{\beta}_{i2} + \boldsymbol{\epsilon}_i, \tag{13}$$

where $\mathbf{X}_{i1} = (\boldsymbol{x}_{i11}^\top, \ldots, \boldsymbol{x}_{in_i1}^\top)^\top$, $\mathbf{X}_{i2} = (\boldsymbol{x}_{i12}^\top, \ldots, \boldsymbol{x}_{in_i2}^\top)^\top$, and $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_{n_i})$. With a given $\mathcal{C}$, our generalized k-means model becomes

$$\boldsymbol{y}_i = \mathbf{X}_{i1}\boldsymbol{\beta}_{i1} + \mathbf{X}_{i2}\boldsymbol{\beta}_{s2} + \boldsymbol{\epsilon}_i, \tag{14}$$

for $z_i \in C_s$. We treat (14) as a special case of (13). Because the second stage in Algorithm 1 is common, we only discuss the first stage.

We select seed z_{i_1} for C_1 randomly. Suppose that $z_{i_1}, \ldots, z_{i_{\tilde{k}}}$ have been selected as the seeds for $C_1, \ldots, C_{\tilde{k}}$, for any $\tilde{k} < k$, respectively. To determine $z_{i_{\tilde{k}+1}}$ for $C_{\tilde{k}+1}$, we calculate the dissimilarity measure between z_s and z_i for $s \in \tilde{S}_{\tilde{k}} = \{z_{i_1}, \ldots, z_{i_{\tilde{k}}}\}$ and $i \notin \tilde{S}_{\tilde{k}}$ based on $\boldsymbol{y}_v = \mathbf{X}_{v1}(\boldsymbol{\beta}_{s1} + \delta_v\boldsymbol{\xi}_{s1}) + \mathbf{X}_{v2}(\boldsymbol{\beta}_{s2} + \delta_v\boldsymbol{\xi}_{s2}) + \boldsymbol{\epsilon}_v$, where $v = s$ or $v = i$, δ_v is the dummy variable defined as $\delta_v = 0$ if $v = s$ or $\delta_v = 1$ if $v = i$, and $\boldsymbol{\epsilon}_v \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_{n_i})$ is the error vector. As the UMPU test statistic becomes an F-statistic, we calculate the F-statistic for

$$H_0 : \boldsymbol{\xi}_{i2} = \mathbf{0} \leftrightarrow H_1 : \boldsymbol{\xi}_{i2} \neq \mathbf{0}. \tag{15}$$

Let p_{si} be the p-value of the F-statistic. We define the p-value of the dissimilarity between z_i and $\tilde{S}_{\tilde{k}}$ as $p_i = \max_{s \in \tilde{S}_{\tilde{k}}} p_{si}$. We choose z_i as the seed for $C_{\tilde{k}+1}$ if it has the lowest p_i value among all objects in $\tilde{S}_{\tilde{k}}$. Therefore, $z_{i_{\tilde{k}+1}}$ is given by the minimax principal as

$$i_{\tilde{k}+1} = \operatorname*{argmin}_i p_i = \operatorname*{argmin}_i \max_s p_{si}. \tag{16}$$

After we obtain $\tilde{S}_k$, which is the set of all of the seeds for $\mathcal{C}$, we calculate the p-value of the F-statistic for (15) for every $s \in \tilde{S}_k$ and $i \notin \tilde{S}_k$. We assign z_i to C_s if p_{si} is maximized at s. Then, we have the initial $\mathcal{C}$. By iterating the second stage in Algorithm 1, we obtain the final $\hat{\mathcal{C}}_k$ based on a given k.

Because $\sigma^2 = a(\phi)$ is present, we follow the GIC in variable selection for regression models (Zhang et al., 2010), and propose our GIC based on a known σ^2 as

$$\mathrm{GIC}_{\sigma^2,\kappa}(\mathcal{C}) = \frac{\mathrm{SSE}}{\sigma^2} + \kappa k q_2, \tag{17}$$

where SSE is the sum of squares of errors given by (14).

Because σ^2 cannot be known, we need to estimate σ^2 in our method. We use the full versus reduced model approach. If the current k is used, then the estimate of σ^2 is SSE divided by residual degrees of freedom. The first term on the right-hand side of (17) is always equal to $n - Nq_1 - kq_2$, implying that this cannot be used. To overcome the difficulty, we use $k+1$ in (14) to estimate σ^2, denoted as $\hat{\sigma}^2_{k+1}$. Therefore, our GIC based on an unknown σ^2 becomes

$$\mathrm{GIC}_{\kappa}(\mathcal{C}) = \frac{\mathrm{SSE}_k}{\hat{\sigma}^2_{k+1}} + \kappa k q_2, \tag{18}$$

where SSE_k is the SSE with k clusters in (14). This is appropriate. If the number of true clusters is less than or equal to k, then slightly increasing the number of clusters would not significantly change the estimate of σ^2, implying that the second term dominates the right-hand side of (18). Otherwise, the estimate of σ^2 would be significantly reduced, implying that the first term dominates the right-hand side of (18). Therefore, the objective function in our GIC provides a nice trade-off between the SSE and the penalty function.

In loglinear models for Poisson data, (6) becomes

$$\log(\mu_{ij}) = \boldsymbol{x}_{ij1}^\top\boldsymbol{\beta}_{i1} + \boldsymbol{x}_{ij2}^\top\boldsymbol{\beta}_{i2}. \tag{19}$$

With a given $\mathcal{C}$, it reduces to

$$\log(\mu_{ij}) = \boldsymbol{x}_{ij1}^\top\boldsymbol{\beta}_{i1} + \boldsymbol{x}_{ij2}^\top\boldsymbol{\beta}_{s2}, \tag{20}$$

for $i \in C_s$. Analogous to the regression models, after selecting z_{i_1} randomly, we investigate

$$\log(\mu_{vj}) = x_{vj1}^{\top}(\boldsymbol{\beta}_{s1} + \delta_v \boldsymbol{\xi}_{i1}) + x_{vj2}^{\top}(\boldsymbol{\beta}_{s2} + \delta_v \boldsymbol{\xi}_{i2}), \tag{21}$$

with $v = s$ or $v = i$. We measure the dissimilarity between z_s and z_i by the likelihood ratio statistic. We derive the initial $\mathcal{C}$ by the same idea that we have displayed in regression models. With the second stage in Algorithm 1, we obtain $\hat{\mathcal{C}}_k$ based on a given k. To determine the best k, we choose $-2\ell(\hat{\omega}_{\mathcal{C}_k})$ as the residual deviance of (20). As the dispersion parameter is not present, the implementation of GIC is straightforward.

For quasi-Poisson data, there is $\mathrm{V}(y_{ijj}) = \phi \mathrm{E}(y_{ij}) = \phi\mu_{ij}$, implying that $a(\phi) = \phi$. We can still use (19), (20), and (21) to find the best $\mathcal{C}$ with a given k. To determine the best k when it is unknown, we estimate ϕ by (5), which is the Pearson goodness-of-fit statistic under (20) divided by its residual degrees of freedom. For the same reason, we choose the number of clusters equal to $k+1$ in (20) in estimating ϕ. It is denoted as $\hat{\phi}_{k+1}$. This induces

$$\mathrm{GIC}_{\kappa}(\mathcal{C}) = \frac{G_k^2}{\hat{\phi}_{k+1}} + \kappa k q_2, \tag{22}$$

where G_k^2 is the residual deviance (i.e., deviance goodness-of-fit) with k clusters in (20).

3. Asymptotic properties

We evaluate asymptotic properties of our method under $n = \sum_{i=1}^{N} n_i \to \infty$, achieved by letting $n_{\min} = \min_i(n_i) \to \infty$. To simplify our notations, we assume that n_i are all equal to n_0 and $|C_s|$ are all equal to c such that we have $N = kc$ and $n = kcn_0$ in our data. The case with distinct n_i and $|C_s|$ can be proven under their minimums going to infinity with bounded ratios between the minimums and the maximums, where the idea is the same.

The asymptotic properties are evaluated under $n_0 \to \infty$ possibly with $k, c \to \infty$, which includes the case when both k and c are constants. For any $i \neq i'$, let $\Lambda_{ii'}$ be the likelihood ratio statistic for

$$H_0 : \boldsymbol{\beta}_{i2} = \boldsymbol{\beta}_{i'2} \leftrightarrow H_1 : \boldsymbol{\beta}_{i2} \neq \boldsymbol{\beta}_{i'2}. \tag{23}$$

As $n_0 \to \infty$, $-2\log\Lambda$ is asymptotically $\chi^2_{q_2}$ distributed if z_i and $z_{i'}$ are in the same cluster, or goes to ∞ with rate n_0 otherwise. Because (23) is applied to all pairs (i, i') in $\mathcal{S}$, the multiple testing problem must be addressed. This can be solved by the method of higher criticisms (Donoho and Jin, 2004). Because we restrict our methods on exponential family distributions, all usual regularity conditions (e.g., all those listed in Chapters 17, 18, and 22 in Ferguson (1996)) for consistency and asymptotic normality of the MLE and the asymptotic χ^2-distribution of the likelihood ratio statistic hold. Therefore, we do not need to impose any other conditions.

Lemma 1. *Assume that $(y_{ij}, \mathbf{x}_{ij}^{\top})^{\top}$ for $j \in \{1, \ldots, n_0\}$ are iid copies of (7) with PDF or PMF given by (3) based on a non-degenerate common distribution of $\mathbf{x}_{ij}$ for any given $i \in \mathcal{S}$. If z_i and $z_{i'}$ are in the same cluster, then $-2\log\Lambda_{ii'} \xrightarrow{L} \chi^2_{q_2}$. If z_i and $z_{i'}$ are in different clusters, then exists a positive constant $A = A(\boldsymbol{\beta}_i, \boldsymbol{\beta}_{i'}, \phi)$, such that the limiting distribution of $-2\log\Lambda - n_0 A$ is non-degenerate as $n_0 \to \infty$.*

Proof. The conclusion can be proven by the standard approach to the asymptotic properties of maximum likelihood and M-estimation. Please refer to Chapter 22 in Ferguson (1996) and Chapter 5 in van der Vaart (1998). □

Theorem 1. *If the assumption of Lemma 1 holds, and $N = o(e^{n_0^{\alpha}})$ for some $\alpha \in (0, 1)$ when $n_0 \to \infty$, then $\hat{\mathcal{C}}_k \xrightarrow{P} \mathcal{C}$.*

Proof. Note that the likelihood ratio test based on $\Lambda_{ii'}$ is applied to distinct $i, i' \in \mathcal{C}$. We need to evaluate the impact of the multiple testing problem. We examine the distribution of the $-2\log\max_{i\neq i'}\Lambda_{ii'}$ based on Lemma 1. According to Donoho and Jin (2004), it is asymptotically bounded by a constant times $2\log N$ if z_i and $z_{i'}$ are in same clusters or increases to ∞ with rate n_0 if z_i and $z_{i'}$ are in different clusters. Thus, with probability 1, the increasing rate of $-2\log\Lambda_{ii'}$ with z_i and $z_{i'}$ in different clusters is faster than that of $\Lambda_{ii'}$ with z_i and $z_{i'}$ in same clusters, implying the conclusion. □

Theorem 2. *Assume that $a(\phi)$ is not present in (3) or $a(\phi)$ is consistently estimated by $a(\hat{\phi})$ used in the construction of GIC, and the assumption of Theorem 1 holds. If $\kappa^{-1}\log c \to 0$ as $n_0 \to \infty$, then $\hat{k}_{\kappa} \xrightarrow{P} k$ and $\hat{\mathcal{C}}_{\hat{k}_{\kappa}} \xrightarrow{P} \mathcal{C}$.*

Proof. If $\hat{k}_{\kappa} < k$, then we can find at least one pair of z_i and $z_{i'}$, such that they are not in the same cluster but they are grouped to the same cluster. By Lemma 1, the first term on the right-hand side of (11) goes to ∞ with rate n_0. It is faster than the rate of GIC under $\hat{k}_{\kappa} = k$, implying that $P(\hat{k}_{\kappa} < k) = 0$ as $n_0 \to \infty$. Therefore, we only need to study the case when $\hat{k}_{\kappa} \geq k$. The loglikelihood function of (7) based on a given $\mathcal{C}$ is equal to the sum of the loglikelihood functions obtained from each $C_s \in \mathcal{C}$. By Theorem 1, we can restrict our attention on the case when all objects in C_s are in the same cluster. By Donoho and Jin (2004), with probability 1, the loglikelihood function (7) in C_s is not higher than that under the true cluster plus $2\log c$. By the property of the χ^2-approximation of the likelihood ratio statistic under the true $\mathcal{C}$, with probability 1, the first term on the right-hand side of (11) is not higher than $n_0N - (Nq_1 + kq_2) + 2kq_2 \log c$. Combining it with the second term, we conclude that $\hat{k}_\kappa \xrightarrow{P} k$. Finally, we draw the conclusion by Theorem 1. □

Theorem 1 implies that both c and k can increase exponentially fast in n_0 when k is known, but the rate is significantly reduced when k is unknown. If $c \to \infty$, then we cannot choose $\kappa = 2$ in our method, implying that $\hat{k}_{AIC}$ is not consistent, but we can still show that $\hat{k}_{BIC}$ is consistent.

Corollary 1. *Suppose that all of assumptions of Theorem 2 are satisfied. If $k \to \infty$ or k is constant, and $c/n_0 \to 0$ when $n_0 \to \infty$, then $\hat{k}_{BIC} \xrightarrow{P} k$.*

Proof. Note that the increasing rate of $\log n$ cannot be lower than the increasing rate of $\log c$. We draw the conclusion by Theorem 2. □

Corollary 1 implies that BIC can be used to determine the number of clusters if k is unknown. This is consistent with many findings for BIC in tuning parameter determinations. Examples include variable selection (Zhang et al., 2010) and dimension reduction (Bai et al., 2018) problems. In clustering analysis, if data are collected from a Euclidean space, then it is generally hard to provide a consistent estimator of σ^2 (or $a(\phi)$) based on an unknown k, implying that it is unlikely to implement GIC to determine the number of clusters. This issue can be easily solved in our method because σ^2 can be consistently estimated by statistical models. Therefore, we can use GIC to determine the number of clusters, but this approach cannot be migrated to data from Euclidean spaces.

4. Simulation

We carried out simulations to evaluate our methods. For an estimated cluster assignment $\hat{\mathcal{C}}$ and the true clustering assignment $\mathcal{C}$, we define the clustering error (CE) of $\hat{\mathcal{C}}$ as $CE_{\hat{\mathcal{C}}} = \binom{N}{2}^{-1} \#\{(i, i') : \hat{\delta}_{ii'} = \delta_{ii'}, 1 \le i < i' \le N\}$, where $\hat{\delta}_{ii'} = 1$ if z_i and $z_{i'}$ belong to the same clusters in $\hat{\mathcal{C}}$, or $\hat{\delta}_{ii'} = 0$ otherwise, and similarly for $\delta_{ii'}$ in $\mathcal{C}$. For estimated clustering assignments $\hat{\mathcal{C}}_1, \ldots, \hat{\mathcal{C}}_R$ obtained from R simulation replications, we calculate the percentage of clustering object errors (OE) by

$$OE = \frac{100}{R} \sum_{j=1}^{R} CE_{\hat{\mathcal{C}}_j}. \tag{24}$$

This is a commonly used criterion in the clustering literature (Wang, 2010). We also study the percentage of numbers of clusters identified correctly (IC) as

$$IC = \frac{100}{R} \sum_{j=1}^{R} I(\hat{k}_j = k), \tag{25}$$

where $\hat{k}_1, \ldots, \hat{k}_R$ are the estimated numbers of clusters obtained from R simulation replications, and k is the true number of clusters. We compare methods based on CE and IC.

4.1. Regression models with a few explanatory variables

We generated data from regression models with $k = 2, 3$ clusters and 2 explanatory variables. This was treated as the implementation of our method under the low-dimensional setting. Each cluster had $c = 10, 20$ objects. Each object contained $n_0 = 50, 100$ observations. We generated explanatory variables x_{ij1} from $\mathcal{U}[18, 70]$ and x_{ij2} from $\mathcal{N}(0, 9)$ independently. For each selected k, c, and n_0, we generated the normal response from

$$y_{ij} = \beta_{i0} + x_{ij1}\beta_{s1} + x_{ij2}\beta_{s2} + \epsilon_{ij}, \tag{26}$$

for $j = 1, \ldots, n_0$ and $i = 1, \ldots, N$, where $N = cn_0$ and $\epsilon_{ij} \sim^{iid} \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.5, 1.0$. We set $\beta_{i0} = \beta_{i'0}$ if z_i and $z_{i'}$ were in the same cluster in (26). If $k = 2$, we chose $\beta_{i0} = 1$, $\beta_{i1} = -0.06$, and $\beta_{i2} = -0.01$ when z_i was in the first cluster, or $\beta_{i0} = 1$, $\beta_{i1} = 0.06$, and $\beta_{i2} = 0.01$ when z_i was in the second cluster. If $k = 3$, we added one more cluster by choosing $\beta_{i0} = 1$, $\beta_{i1} = -0.02$, and $\beta_{i2} = 0.01$ when z_i was in the third cluster. Then, we obtained data from (26) with either 2 or 3 clusters.

We evaluated our method based on AIC and BIC with the comparison to the previous EM algorithm proposed by Qin and Self (2006). We implemented our AIC and BIC given by (18) by choosing $\kappa = 2$ and $\log(n)$, respectively. The EM algorithm was implemented by R package RegClust. To implement RegClust, we had to consider the saturated clustering problem, where we chose β_{i0} not varied within clusters. We also considered two other competitors. The first was the usual k-means directly on regression coefficients. The second was the convex clustering (Chi and Lange, 2015) directly on regression coefficients. Following Tibshirani et al. (2001), we estimated the number of clusters by maximizing the gap statistic.

Table 1
Percentage of numbers of clusters identified correctly (*IC*) based on 1000 simulation replications when data are generated from (26) with respect to *k*-means (K), convex clustering (Convex), the EM algorithm, and our AIC and BIC selectors in generalized *k*-means.

σ	c	k	$n_0 = 50$					$n_0 = 100$				
			K	Convex	EM	AIC	BIC	K	Convex	EM	AIC	BIC
0.5	10	2	53.2	59.7	81.1	15.4	**97.6**	70.2	73.9	72.4	21.4	**98.1**
		3	23.3	19.2	0.0	5.6	**97.8**	10.7	7.4	0.0	6.2	**98.9**
	20	2	85.1	87.2	75.1	0.5	**88.6**	92.3	91.7	75.2	0.3	**93.2**
		3	6.7	5.6	0.0	0.0	**95.9**	1.7	1.5	0.0	0.0	**97.3**
1.0	10	2	22.6	25.0	75.1	17.3	**96.8**	35.9	39.2	71.6	15.9	**98.6**
		3	27.3	21.6	0.0	7.3	**95.7**	24.4	21.5	0.0	4.9	**98.4**
	20	2	52.4	52.7	74.2	0.5	**93.0**	69.5	71.5	72.2	0.3	**93.8**
		3	15.6	13.0	0.0	0.3	**87.6**	15.1	14.0	0.0	0.0	**96.5**

Table 2
Percentage of clustering object errors (*OE*) based on 1000 simulation replications when data are generated from (26) with respect to *k*-means (K), convex clustering (Convex), the EM algorithm, and our AIC and BIC selectors in generalized *k*-means.

σ	c	k	$n_0 = 50$				$n_0 = 100$			
			K	Convex	EM	BIC	K	Convex	EM	BIC
0.5	10	2	48.7	46.9	10.0	**0.3**	47.4	46.9	33.6	**0.0**
		3	47.7	47.1	14.5	**0.2**	49.5	49.2	33.1	**0.3**
	20	2	49.7	49.5	12.8	**1.3**	49.5	49.4	33.1	**0.3**
		3	49.8	49.7	12.8	**0.8**	50.1	50.0	31.8	**0.2**
1.0	10	2	48.8	48.5	13.1	**0.4**	49.0	48.6	33.6	**3.1**
		3	45.6	44.5	14.9	**0.2**	46.7	45.8	36.4	**0.2**
	20	2	49.8	49.6	13.3	**0.8**	49.7	49.6	34.5	**3.8**
		3	48.4	47.8	14.3	**0.7**	49.0	48.7	36.5	**0.4**

Table 1 displays the simulation results for the percentage of number errors correctly identified by the EM algorithm, the *k*-means and convex clustering directly on regression coefficients, and our AIC and BIC selectors. Although it was also based on BIC for numbers of clusters, in all of the simulations that we ran, we found that the number of clusters reported by the EM algorithm based on RegClust was either 1 or 2, implying that it could not identify the true number of clusters when $k > 2$. The performance of *k*-means and convex clustering was slightly better than that of the EM algorithm, but it was not as good as our BIC selector. The true k could be detected by our BIC not our AIC.

Table 2 displays the simulation results for the percentage of clustering object errors by the EM algorithm, the *k*-means and convex clustering directly on regression coefficients, and our BIC selector. We did not include AIC in the table because BIC was better. Our result shows that our BIC was always better than our competitors. It was able to find the true number of clusters with lower clustering object errors. This is an advantage of our generalized *k*-means for regression models under the low-dimensional setting.

4.2. Regression models with many explanatory variables

We still generated data from regression models with $k = 2, 3$ clusters, but we increased the number of explanatory variables to 15 such that it could reflect our method under the high-dimensional setting. We studied the unsaturated problem. We also chose $c = 10, 20$ objects in each cluster, and each object contained $n_0 = 50, 100$ observations. We generated the 15th explanatory variables independently from $\mathcal{N}(0, 1)$. For each selected k, c, and n_0, we generated the normal response from

$$y_{ij} = \beta_{i0} + \sum_{t=0}^{5} x_{ijt}\beta_{it} + \sum_{t=6}^{15} x_{ijt}\beta_{it} + \epsilon_{ij}, \tag{27}$$

for $j = 1, \ldots, n_0$ and $i = 1, \ldots, N$, where $N = cn_0$ and $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.1, 0.2, 0.5, 1.0$. We generated $\beta_{i0}, \ldots, \beta_{i5}$ independently $\mathcal{N}(0, 0.2^2)$ for each i. If $k = 2$, we chose $\beta_{it} = 0.05$ for $6 \le t \le 15$ when z_i was in the first cluster, or $\beta_{it} = -0.05$ when z_i was in the second cluster. If $k = 3$, we added one more cluster by choosing $\beta_{it} = \cdots = \beta_{it} = -0.05$ for $6 \le t \le 10$ or $\beta_{it} = 0.05$ for $11 \le t \le 15$. We obtained data from (27) with $k = 2, 3$ clusters.

We discarded our AIC and only used our BIC selector for number of clusters (Table 3). We wanted to group the statistical models based on the last 10 regression coefficients (i.e., it is an unsaturated clustering problem). We could not use RegClus because it has not been formulated for an unsaturated clustering problem yet. Therefore, we compared our method with the other two competitors: the *k*-means and the convex clustering directly on regression coefficients. We also included

Table 3
Percentage of numbers of clusters identified correctly (IC) based on 1000 simulation replications when data are generated (27) with respect to k-means (K), convex clustering (Convex), and k-means++ (KPP) directly on regression coefficients based on the gap statistic and our BIC selector in generalized k-means.

σ	c	k	$n_0 = 50$				$n_0 = 100$			
			K	Convex	KPP	BIC	K	Convex	KPP	BIC
0.1	10	2	92.6	98.5	88.2	**100.0**	96.5	**100.0**	93.9	**100.0**
		3	72.3	99.2	90.2	**100.0**	72.4	99.9	93.1	**100.0**
	20	2	99.7	**100.0**	98.9	**100.0**	**100.0**	**100.0**	99.9	**100.0**
		3	73.3	**100.0**	99.2	**100.0**	72.9	**100.0**	99.9	**100.0**
0.2	10	2	94.6	97.8	90.9	**100.0**	96.0	99.8	93.3	**100.0**
		3	36.8	47.5	26.9	**100.0**	75.6	99.2	95.0	**100.0**
	20	2	99.7	**100.0**	99.0	**100.0**	**100.0**	**100.0**	99.9	**100.0**
		3	24.6	24.4	10.6	**100.0**	77.2	**100.0**	99.8	**100.0**
0.5	10	2	95.7	99.9	93.5	**100.0**	96.0	99.1	95.4	**100.0**
		3	1.1	1.3	1.2	**88.7**	1.3	1.4	1.4	**100.0**
	20	2	99.9	**100.0**	99.7	**100.0**	99.9	**100.0**	99.8	**100.0**
		3	0.0	0.0	0.0	**95.2**	0.0	0.0	0.0	**100.0**
1.0	10	2	96.2	**100.0**	93.0	**100.0**	97.7	**100.0**	95.8	**100.0**
		3	0.9	0.8	1.6	**16.2**	0.3	0.2	0.3	**57.5**
	20	2	99.6	**100.0**	99.6	**100.0**	**100.0**	**100.0**	99.9	**100.0**
		3	0.0	0.0	0.0	**39.8**	0.0	0.0	0.0	**84.3**

Table 4
Percentage of clustering object errors (OE) based on 1000 simulation replications when data are generated from (27) with respect to k-means (K), convex clustering (Convex), and k-means++ (KPP), directly on regression coefficients and our BIC selector in the generalized k-means.

σ	c	k	$n_0 = 50$				$n_0 = 100$			
			K	Convex	KPP	BIC	K	Convex	KPP	BIC
0.1	10	2	0.4	0.1	0.6	**0.0**	0.2	**0.0**	0.3	**0.0**
		3	1.4	**0.0**	0.2	**0.0**	1.5	**0.0**	0.1	**0.0**
	20	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
		3	1.5	**0.0**	**0.0**	**0.0**	1.5	**0.0**	**0.0**	**0.0**
0.2	10	2	0.3	0.1	0.5	**0.0**	0.2	**0.0**	0.3	**0.0**
		3	14.0	12.4	16.1	**0.0**	1.4	**0.0**	0.1	**0.0**
	20	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
		3	17.5	17.6	20.6	**0.0**	1.5	**0.0**	**0.0**	**0.0**
0.5	10	2	10.8	10.3	10.4	**0.1**	0.9	0.6	0.9	**0.0**
		3	33.1	32.6	32.2	**2.5**	26.0	26.8	25.7	**0.1**
	20	2	8.4	8.2	8.3	**0.1**	0.5	0.5	0.5	**0.0**
		3	31.4	31.3	31.2	**1.5**	25.9	26.3	25.6	**0.1**
1.0	10	2	41.1	40.2	40.0	**8.3**	22.8	21.7	20.5	**1.2**
		3	46.8	45.7	46.4	**22.6**	38.9	38.1	38.0	**11.1**
	20	2	39.4	38.9	38.1	**5.3**	18.6	18.0	17.7	**1.2**
		3	45.9	45.2	45.1	**18.0**	36.4	36.1	35.8	**4.5**

the k-means++ in our comparison because Step 1 in Algorithm 1 was motivated by initialization of k-means++. Similar to Section 4.1, we still estimated the number of clusters by maximizing the gap statistic. It was used to the k-means, the convex clustering, and the k-means++. Our results showed that all of the four methods were able to identify the number of clusters when σ was small (i.e., $\sigma = 0.1, 0.2$), but our BIC selector could still be used to identify the number of clusters even when σ was large (i.e., $\sigma = 0.5, 1.0$). This was because our method was formulated by the UMPU test, which was optimal in measuring the difference between statistical models. The performance of the convex clustering was better than that of the k-means and the k-means++, indicting that it is more appropriate than the other two methods in grouping regression models.

We also evaluated the percentage of clustering object errors (Table 4). We found that our method was also better than our competitors, as it had the lowest clustering object errors. The performance of the convex clustering was better than that of the k-means and the k-means++. Notice that many penalties could be used to determine the number of clusters and the gap statistic was just one of those. Examples can be found in Koepke and Clarke (2013). To know whether the performance of our competitors could be significantly improved if other penalties were adopted, we compared our BIC selector based on an unknown k with our competitors based on a known k. In this case, the impact of the penalties was completely removed in our competitors. Our results (not shown) indicated that the percentage of clustering object errors was almost the same as those displayed in Table 4. This means that our method based on an unknown k was better than our competitors based on a known k. Thus, our method can significantly enhance the precision and accuracy in grouping statistical models compared to our competitors. It is more appropriate to use our method than our competitors in grouping statistical models (see Table 5).

Table 5
Percentage of number of clusters identified correctly (IC) based on 1000 simulation replications when data are generated from a Euclidean space (i.e., $\mathbb{R}^5$) with respect to the k-means (K), the convex clustering (Convex), and the k-means++ (KPP) based on the gap statistic.

σ	k	$n_0 = 100$			$n_0 = 200$		
		K	Convex	KPP	K	Convex	KPP
0.001	2	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	3	62.1	89.4	**100.0**	60.2	91.7	**100.0**
	4	44.9	80.4	**98.8**	44.5	85.7	**98.9**
0.002	2	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	3	62.0	88.9	**100.0**	59.6	93.4	**99.9**
	4	46.2	85.0	**98.7**	48.9	86.7	**98.5**
0.005	2	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	3	65.3	89.7	**100.0**	64.5	93.5	**100.0**
	4	45.3	83.9	**98.6**	47.8	85.7	**98.4**
0.01	2	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	3	67.0	89.6	**100.0**	67.1	90.7	**100.0**
	4	46.9	83.3	**98.9**	48.4	84.1	**98.5**

Table 6
Percentage of clustering object errors (OE) based on 1000 simulation replications when data are generated from a Euclidean space (i.e, $\mathbb{R}^5$) with respect to the k-means (K), the convex clustering (Convex), and the k-means++ (KPP) based on the gap statistic.

σ	k	$n_0 = 100$			$n_0 = 200$		
		K	Convex	KPP	K	Convex	KPP
0.001	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
	3	2.4	**0.0**	**0.0**	2.5	**0.0**	**0.0**
	4	4.6	0.3	**0.2**	4.2	0.4	**0.1**
0.002	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
	3	2.4	**0.0**	**0.0**	2.6	**0.0**	**0.0**
	4	4.3	0.4	**0.2**	4.3	0.4	**0.2**
0.005	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
	3	2.1	**0.0**	**0.0**	2.2	0.1	**0.0**
	4	4.1	0.3	**0.2**	4.5	0.4	**0.2**
0.01	2	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
	3	2.0	0.1	**0.0**	2.0	0.5	**0.0**
	4	4.1	0.4	**0.1**	4.3	0.4	**0.2**

To understand the importance of the UMPU test statistic, we compared the k-means, the convex clustering, and the k-means++ when they were applied to data from Euclidean spaces. In this case, we generated data from $\mathbb{R}^5$ with $k = 2, 3, 4$ clusters. At the beginning, for each k, we generated cluster centers by the uniform distribution on $[0, 1]^5$. Then, for each cluster center, we generated n_0 points by the multivariate normal distribution with mean vector to be the cluster center and variance–covariance matrix to be $\sigma^2\mathbf{I}$. After that, we used the three methods to group the data with the number of clusters to be determined by the gap statistic. We found that based on the gap statistic, all of the three methods can correctly identify the true number of clusters. The performance of the k-means++ was better than the other two, because of its initialization. Because Step 1 in our algorithm is motivated by the k-means++, we conclude that this step could also increase the precision and accuracy compared to the method based on random initialization. Our result for the percentage of clustering object errors (Table 6) indicated that the three methods were all precise even if they did not find the correct number of clusters. To confirm this, we looked at the information contained by additional clusters. We found that they were all small and did not significantly affect the results of the percentage of clustering object errors.

In summary, our simulation shows that all of the k-means, the convex clustering, and the k-means++ are precise and accurate in grouping data from Euclidean spaces, but our method is more precise and accurate than those in grouping statistical models. This is because our method is formulated by the UMPU test, which is the best in measuring the difference between statistical models. Initialization of clustering methods is important. A good initialization can increase precision and accuracy of the results.

4.3. Loglinear models

Similar to the regression models, we also chose $k = 2, 3$ clusters in loglinear models for Poisson data. Each cluster had $c = 10, 20$ objects. Each object contained $n_0 = 50, 100$ observations. We generated explanatory variables x_{ij1} and x_{ij2} from $\mathcal{N}(0, 4)$ independently. For each selected k, c, and n_0, we independently generated the response y_{ij} from $\mathcal{P}(\lambda_{ij})$ with

$$\log \lambda_{ij} = \beta_{i0} + x_{ij1}\beta_{s1} + x_{ij2}\beta_{s2}, \tag{28}$$

Table 7
Percentage of number clusters identified correctly (IC) in loglinear models based on 1000 simulation replications when data are generated from (28).

τ	c	$n_0 = 50$				$n_0 = 100$			
		$k = 2$		$k = 3$		$k = 2$		$k = 3$	
		AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC
0.5	10	1.6	**91.7**	0.3	**90.9**	1.6	**94.9**	0.4	**93.9**
	20	0.1	**71.2**	0.0	**68.7**	0.0	**80.1**	0.0	**77.1**
1.0	10	2.1	**92.5**	1.2	**92.1**	1.3	**95.3**	0.6	**95.2**
	20	0.0	**78.1**	0.0	**76.5**	0.1	**83.4**	0.0	**83.0**

Table 8
BIC for percentage of clustering object errors (OE) in loglinear models based on 1000 simulation replications with data generated from (28).

τ	c	k for $n_0 = 50$		k for $n_0 = 100$	
		2	3	2	3
0.5	10	1.3	0.6	0.8	0.4
	20	4.4	2.1	3.0	1.5
1.0	10	1.1	0.5	0.7	0.3
	20	3.3	1.5	2.5	1.1

for $j = 1, \ldots, n_0$ and $i = 1, \ldots, N$, where $N = cn_0$. We generated β_{i0} independently from $\mathcal{N}(10, 1)$. We set $(\beta_{11}, \beta_{12}) = (1, 1)$ in the first cluster and $(\beta_{21}, \beta_{22}) = (-1, -1)$ in the second cluster. This was used if $k = 2$. If $k = 3$, we chose $(\beta_{31}, \beta_{32}) = (1, -1)$ in the third cluster. We evaluated our method based on AIC and BIC for the unsaturated clustering problem, where we varied β_{i0} within clusters.

Table 7 displays the simulation results for the percentage of clustering number errors. We also found that the true k could be identified by our BIC but not by our AIC. Table 8 displays the results for the percentage of clustering object errors based on BIC. It shows that the percentage of clustering object errors was still low, indicating that BIC can be used to find the correct number of cluster with the low error rate. Therefore, we recommend using BIC in our generalized k-means if the number of clusters is unknown.

5. Application

We implemented our method to the state-level daily COVID-19 data in the United States. The state-level daily COVID-19 data are reported by the United States Centers for Disease Control and Prevention (CDC). The data set contains confirmed disease counts, deaths, and recoveries with the information updated everyday. Data reported by CDC are based on the most recent numbers reported by states and territories in the United States. COVID-19 can cause mild symptoms, which can induce delays in reporting and testing, leading to difficulties in reporting the exactly numbers of COVID-19 cases. The accuracy of the data has been discussed by CDC. CDC attempts to provide more accurate data by updating previous information. The detail interpretation of the accuracy of the data can be found in the website of the CDC.

In the global pandemic of COVID-19, many countries in the Northern Hemisphere have encountered dramatically increased cases and deaths since August 2020, leading to the appearance of the second wave (which they called). Patients in the second wave were younger than those in the first wave, but the impact was still unclear (Iftimie et al., 2020). We applied our method to the data until July 31 2020 to avoid this problem.

The outbreak of COVID-19 has occurred and become the ongoing pandemic in the world since March 2020. More than 200 countries and territories have affected. The most serious country is the United States. Until July 31, it had over 4.7 million confirmed cases and one hundred sixty thousand deaths. Both were the highest in the world. After briefly looking at the patterns of the data (Fig. 2), we found that some of the curves were similar to each other (e.g., California and North Carolina) but some of those were far away from each other (e.g., California and Michigan). To address this issue, a straightforward approach is to group these curves by a clustering method. This can help us understand the connection of outbreaks between individual states. We found significant changes in the daily patterns before May 31 and after June 1. Two possible issues were identified based on social medias. The first was the George Floyd issue, that occurred on May 25 in Minneapolis. The second was the economy reopening issue. Most states reopened their economy or released their restrictions for the prevention of the spread at the end of May.

The first patient of COVID-19 appeared in Wuhan, China, on December 1 2019. In late December, a cluster of pneumonia cases of unknown causes was reported by local health authorities in Wuhan with clinical presentations greatly resembling viral pneumonia (Chen et al., 2020; Sun et al., 2020). Deep sequencing analysis from lower respiratory tract samples indicated a novel coronavirus (Feng, 2020; Huang et al., 2020). The virus of COVID-19 primarily spreads between people via respiratory droplets from breathing, coughing, and sneezing (World Health Organization (WHO), 2020). This can cause cluster infections in society. To avoid cluster infections, many countries have imposed travel restrictions, which affected

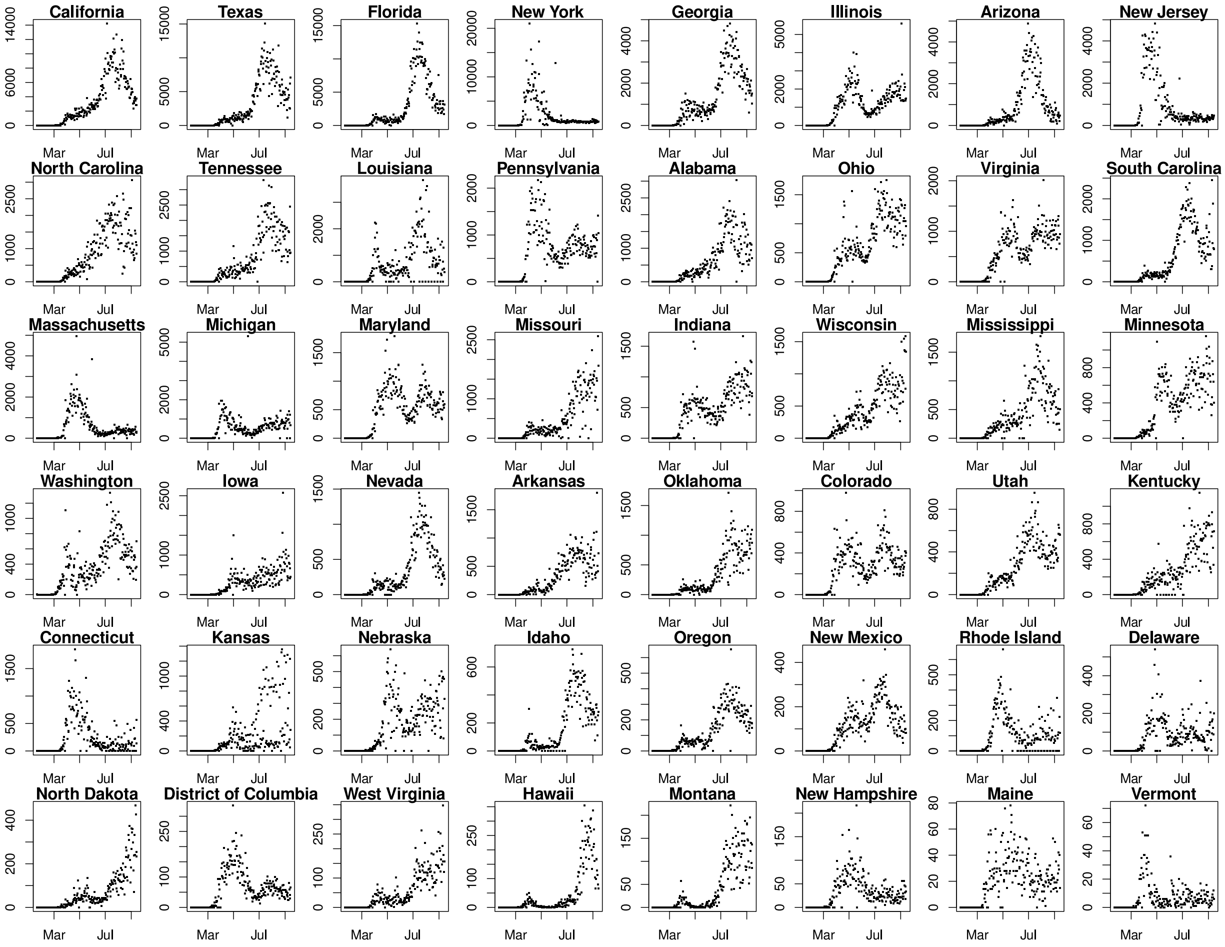

Fig. 2. Daily new cases of COVID-19 in 48 states in the mainland United States.

over 91% of the total population of the world with three billion people living in countries with restrictions on people arriving from other countries borders completely closed to noncitizens and nonresidents (Pew Research Center, 2020).

Exponential increasing trends are expected at the beginning of outbreaks in any infectious disease. This has been observed in the 2009 Influenza A (H1N1) pandemic (de Picoli et al., 2011) and the 2014 Ebola outbreak in West Africa (Hunt, 2014). Without any prevention efforts, the exponential trend will be continuing for a long time until a large portion of people is infected. This trend can be changed by government prevention (Maier and Brockmann, 2020). This is the reason why we study the data until July 31, 2020.

To obtain a more appropriate model, we investigate a few candidate models. We choose the response as the number of daily new cases and explanatory variables as certain functions of time. We obtain two candidate models. The first is the exponential model given by

$$\log \lambda_j = \mu + \beta(t_j - t_0), \tag{29}$$

where t_0 is the starting date, t_j is the current date, $\lambda_j = \mathrm{E}(y_j)$, and y_j is the number of daily new cases observed on the current date. The second is the Gamma model given by

$$\log \lambda_j = \mu + \alpha \log(t_j - t_0) + \beta(t_j - t_0). \tag{30}$$

The Gamma model assumes that the expected number of daily new cases is proportional to the density of a Gamma distribution. If the second term is absent, then the Gamma model reduces to the exponential model, implying that (29) is a special case of (30).

In the case when $\alpha > 0$, if $\beta > 0$, then the third term dominates the right-hand side of (30). The expected value of the response goes to infinity as time goes to infinity, leading to an exponential increasing trend in the outbreak in the study period. If $\beta < 0$, then the peak of the model is attained at $t_{\max} = t_0 - \alpha/\beta$. An increasing trend is expected if $t < t_{\max}$, and a decreasing trend is expected otherwise. Therefore, we can use the sign of β to determine whether the outbreak is under control or not.

Table 9
Fitting results of the exponential and the Gamma models for the outbreak of COVID-19 in eleven selected countries between January 11 to May 31, 2020.

Country	Exponential			Gamma				
	μ	β	R^2	μ	α	β	R^2	Peak
China	7.91	−0.032	0.368	−9.6	7.77	−0.290	0.813	02/07
USA	7.23	0.025	0.582	−64.7	20.56	−0.195	0.939	04/26
Canada	4.19	0.026	0.561	−75.1	22.61	−0.215	0.920	04/26
Russia	3.67	0.044	0.840	−135.2	37.90	−0.308	0.993	05/13
Spain	6.59	0.013	0.164	−77.0	24.72	−0.283	0.899	04/08
UK	5.53	0.023	0.446	−82.9	25.25	−0.248	0.857	04/22
Italy	6.66	0.010	0.110	−58.0	19.53	−0.238	0.945	04/03
France	6.26	0.012	0.096	−96.9	30.47	−0.353	0.694	04/07
Germany	6.33	0.011	0.103	−83.7	26.80	−0.317	0.862	04/05
Switzerland	4.90	0.006	0.030	−116.0	36.50	−0.463	0.853	03/30
Sweden	3.28	0.026	0.626	−43.75	13.6	−0.123	0.876	04/30

We chose t_0 as January 11, 2020 in both (29) and (30). We assumed that y_i followed the quasi-Poisson model, such that we could fit the two models by the traditional loglinear model with dispersion parameter $a(\phi) = \phi$ to be estimated by (5). We assessed the two models by their R^2 values, where the R^2 value of a GLM was defined as one minus residual deviance divided by the null deviance. We verified (29) and (30) by implementing them in eleven countries in the world (Table 9), where the peak was estimated by $\hat{t}_{\max} = t_0 - \hat{\alpha}/\hat{\beta}$ with $\hat{\alpha}$ and $\hat{\beta}$ as the MLEs of α and β in the model. We found that the results given by the Gamma model were significantly better than those given by the exponential model.

We used our generalized k-means to group models for the 50 states and Washington DC. We modified the model given by (30) as

$$\log \lambda_{ij} = \mu_i + \alpha_s \log(t_j - t_0) + \beta_s (t_j - t_0), \tag{31}$$

where $\lambda_{ij} = \mathrm{E}(y_{ij})$, y_{ij} was the number of daily new cases from the ith state on the jth date, and α_s and β_s were the coefficients given by the sth cluster.

Because we allowed μ_i to be different within clusters, we were able to account for many state-level variables simultaneously by μ_i only in (31). For instance, if the population sizes of two states are different but we conclude that they belong to the same cluster, then the impact of the population sizes can be completely accounted for by μ_i in (31). Using this idea, we can account for the combined effects of governmental restrictions, policies, population densities, and population demographics only by μ_i in (31), and this is the advantage of the unsaturated clustering method used in the data analysis. It is not necessary to develop additional statistical models to account for their separate effects in the clustering analysis.

After briefly looking at the data, we found that many of daily new cases were zero in January and February and the United States only had 6 total number of confirmed cases until February 24. We decided to exclude data before February 24 in our analysis. We then applied (31) to the data between February 24 and May 31 and the data between February 24 and July 31, respectively. We looked at their differences because we wanted to know the impact of the two issues mentioned at the beginning of this section. Both AIC and BIC showed that there were six clusters in the data (Fig. 3). We then calculated the cluster maps based on $k = 6$ (Fig. 4). To compare, we also directly used k-means to group estimates of regression coefficients given by (31) (i.e., based on $\hat{\alpha}_i$ and $\hat{\beta}_i$ for the ith state). We found that we were not able to identify the number of clusters based on the gap statistic (Fig. 5). This means that it is hard to use k-means to group statistical models for the patterns of COVID-19 data in the United States. Similar issues also appeared in convex clustering directly on regression coefficients. Our generalized k-means can overcome the difficulty because it is more powerful than methods directly on regression coefficients.

To verify our result, we examined three models. The first was the main effect model. It had only one cluster in (31). The second was the resulting (31) with 6 clusters. The third was the interaction effect model, which assumed that each state was an individual cluster in (31). We calculated the differences of residual deviance between the first and second models, and between the first and the third model, respectively. We obtained the partial R^2 by the ratio of the two differences. The partial R^2 value interpreted the ratio of residual deviance reduced by the model with k clusters. When $k = 6$, the partial R^2 was 0.9235 for data between February 24 and May 31, and 0.9606 for data between February 24 and July 31, implying that the model with six clusters was good enough to interpret the differences among the 50 states and Washington DC.

We evaluated properties of identified clusters by the MLEs of α_s and β_s with $k = 6$ in (31) (Table 10). These coefficients were directly reported by the generalized k-means. We found that the situation in the entire United States was under control before May 31 as the signs of $\hat{\boldsymbol{\beta}}_s$ were all negative. For data between February 24 and July 31, the situations in the states contained by the first, the fourth, and the sixth cluster became worse, as they were out-of-control. The situations in the states contained by the second, the third, and the fifth clusters were still under control. This was caused by the issue that a lot of people did not keep social distance or stay-at-home order in the summer in these states (according to social media).

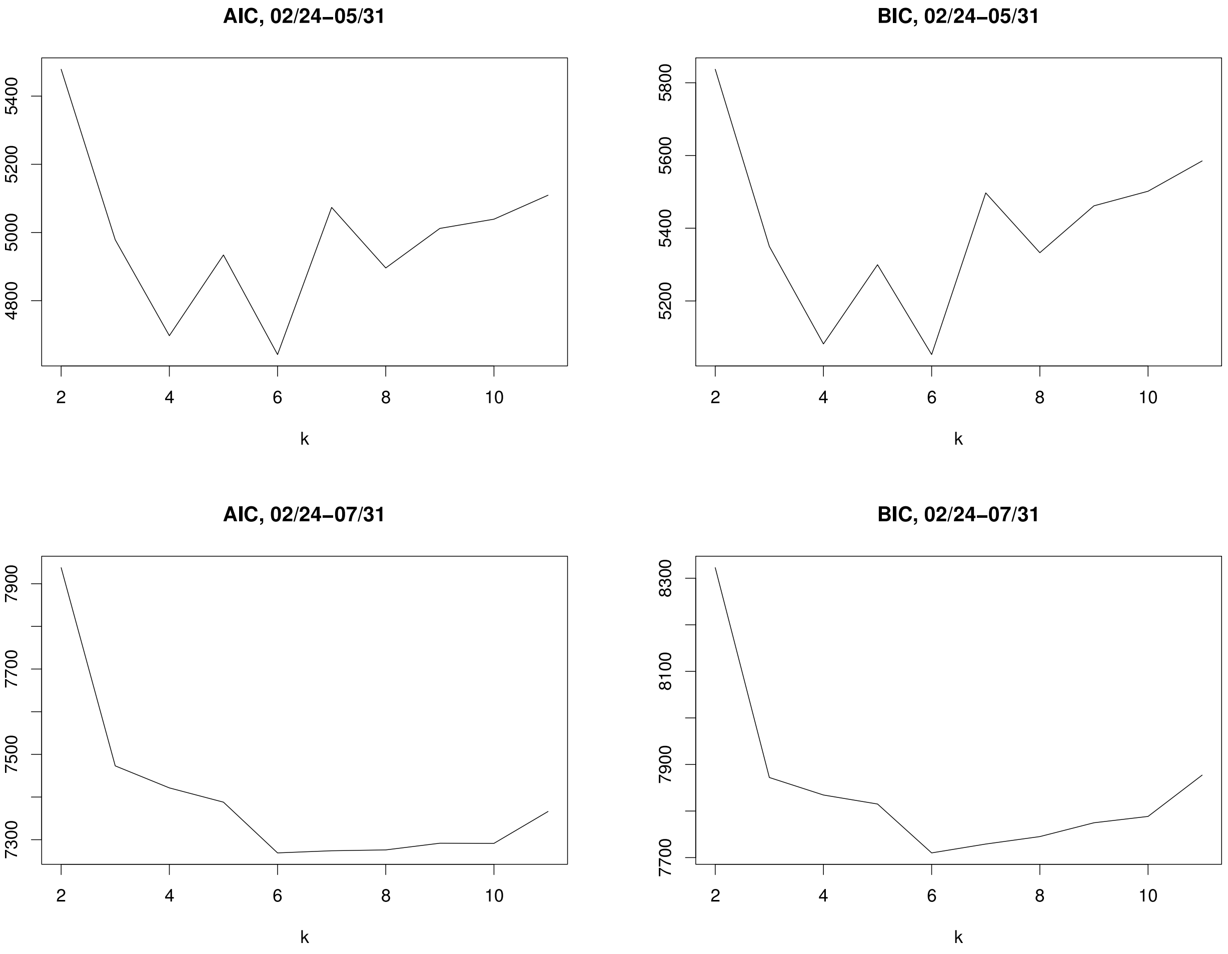

Fig. 3. AIC and BIC for number of clusters in generalized k-means based on (30).

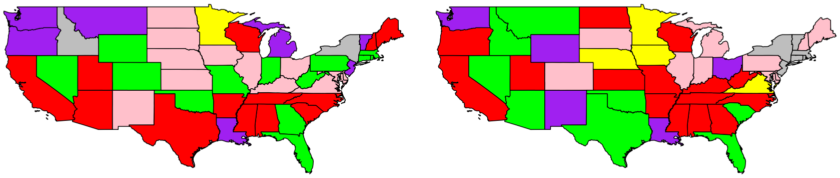

Fig. 4. Six clusters identified by BIC in generalized k-means for the period between February 24 to May 31 (left) and the period between February 24 to July 31 (right), respectively.

Table 10
Parameter estimates in the six clusters with a selected state (State) for each cluster based on the Gamma model for the outbreak of COVID-19 in the United States, where the standard errors are given inside the parenthesis and ×means out of control.

Cluster	State	02/24–05/31			02/24–07/31		
		α	β	Peak	α	β	Peak
1	California	10.49(0.62)	−0.8750(0.0066)	5/10(2.27)	1.958(0.25)	0.0069(0.0020)	×
2	New York	24.63(0.65)	−0.2962(0.0078)	4/3(0.28)	11.05(0.29)	−0.1206(0.0030)	4/12
3	Illinois	19.22(0.87)	−0.1780(0.0090)	4/28(0.81)	6.48(0.30)	−0.0538(0.0026)	5/11
4	Louisiana	21.00(0.72)	−0.2378(0.0082)	4/8(0.39)	1.179(0.39)	0.0044(0.0034)	×
5	Minnesota	19.50(4.26)	−0.1545(0.0425)	5/17(7.3)	8.010(0.69)	−0.0548(0.0056)	6/5
6	Florida	19.39(0.63)	−0.2011(0.0068)	4/26(0.42)	1.57(0.31)	0.0178(0.0024)	×

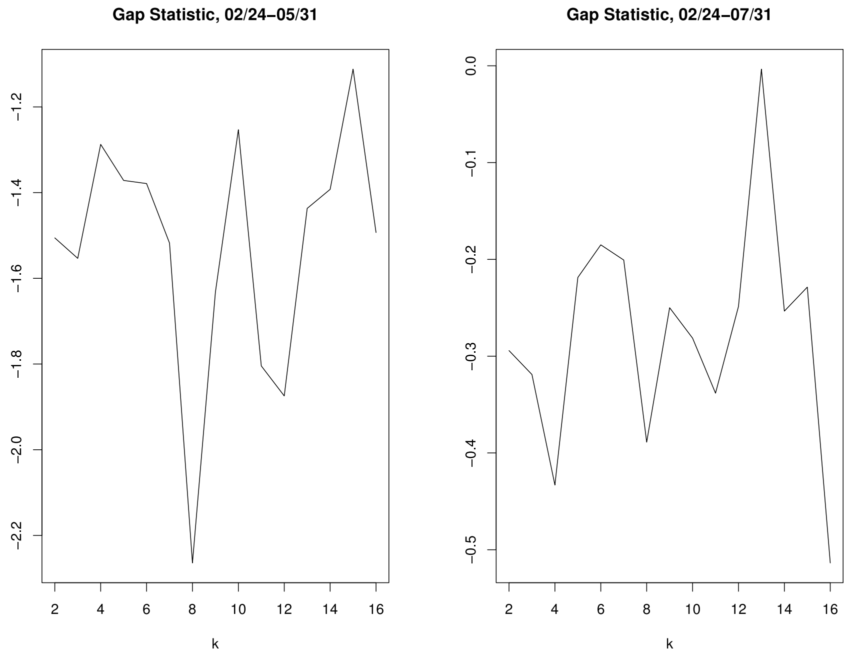

Fig. 5. Gap statistics for number of cluster in k-means for coefficients.

6. Discussion

We propose a new clustering method under the framework of the generalized k-means to group GLMs for exponential family distributions. The method can automatically select the number of clusters if it is combined with GIC. Our theoretical and simulation results show that the number of clusters can be identified by BIC but not by AIC. Therefore, we recommend using BIC in finding the number of clusters. As the choice of the dissimilarity measure is flexible, our method can be extended to other models beyond GLMs. We implement our method to partition loglinear models for the state-level COVID-19 data until July 31 2020 in the United States and finally we have identified six clusters. In Fall 2020, the situations in United States and many European countries became worse and the outbreaks became out-of-control. This is left to future research.

Basically, our generalized k-means can be treated as a modification of k-means++ after the Euclidean distance for dissimilarity between points is replaced by the likelihood ratio statistic for dissimilarity between statistical models. The basic approach in our method can be migrated to any existing clustering methods, including convex clustering, such that the modified methods can be used to group statistical models. The idea is to simply replace the distance measure by a UMPU test statistic. Therefore, the impact of our research is not limited to generalizations of k-means or k-means++.

Theoretically, generalized k-means can also be combined with the penalized likelihood approach. This is useful when the number of explanatory variables exceeds the number of observations within objects. As it is impossible to estimate coefficients of regression parameters, a variable selection procedure is needed to reduce the number of explanatory variables. This is also left to future research.

Acknowledgments

The authors appreciate great comments from Associate Editor and two anonymous referees that significantly improved the quality of the article.

References

Arthur, D., Vassilvitskii, S., 2007. K-means++: the advantage of careful seeding. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA, pp. 1027–1035.

Bai, Choi, Fujikoshi, 2018. Consistency of AIC and BIC in estimating the number of significant components in high-dimensional principal component analysis. Ann. Statist. 46, 1050–1076.

Bock, H., 2008. Origins and extensions of the k-means algorithm in cluster analysis electron. Electron. J. Hist. Probab. Stat. 4, Article 14.

Charikar, M., Guha, S., 2002. A constant-factor approximation algorithm for the k-median problem. J. Comput. Syst. Sci. 65, 129–149.

Chen, Y., Iyengar, R., Ivengar, G., 2016. Modeling multimodal continuous heterogeneity in conjoint analysis–a sparse learning approach. Mark. Sci. 36, 140–156.

Chen, N., Zhou, M., Dong, X., Qu, J., Gong, F., Han, Y., Qiu, Y., Wang, J., Liu, Y., Wei, Y., Xia, J., Yu, T., Zhang, X., Zhang, L., 2020. Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study. Lancet 395, 507–513.

Chi, E.C., Lange, K., 2015. Splitting methods for convex clustering. J. Comput. Graph. Statist. 24, 994–1013.

Donoho, D., Jin, J., 2004. Higher criticism for detecting sparse heterogeneous mixtures. Ann. Statist. 32, 962–994.

Du, Q., Wong, T.W., 2002. Numerical studies for MacQueen's *k*-means algorithms for computing the centroidal Voronoi tessellations. Comput. Math. Appl. 44, 511–523.

Fan, J., Guo, S., Hao, N., 2012. Variance estimation using refitted cross-validation in ultrahigh dimensional regression. J. R. Stat. Soc. Ser. B 74, 37–55.

Feng, Z., 2020. Urgent research agenda for the novel coronavirus epidemic: transmission and non-pharmaceutical mitigation strategies. Chin. J. Epidemiol. 41, 135–138.

Ferguson, T.S., 1996. A Course in Large Sample Theory. CRC Press, New York.

Forgy, E.W., 1965. Cluster analysis of multivariate data: efficiency vs interpretability of classifications. Biometrics 21, 768–769.

Fraley, C., Raftery, A.E., 2002. Model-based clustering, discriminant analysis, and density estimation. J. Amer. Statist. Assoc. 97, 611–631.

Gonzalez, T.F., 1985. Clustering to minimize the maximum intercluster distance. Theoret. Comput. Sci. 38, 293–306.

Goyal, M., Aggarwal, S., 2017. A review on *k*-mode clustering algorithm. Int. J. Adv. Res. Comput. Sci. 8, 725–729.

Green, P.J., 1984. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternative. J. R. Stat. Soc. Ser. B 46, 149–192.

Hartigan, J.A., Wong, M.A., 1979. A *k*-means clustering algorithm. Appl. Stat. 28, 100–108.

Hocking, T.D., Joulin, A., Back, F., Vert, J.P., 2011. Clusterpath: an algorithm for clustering using convex fusion penalties. In: Proceedings of the 28th International Conference on International Conference on Machine Laerning. ICML2011. pp. 745–752.

Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y, Zhang, L., Fan, G., Xu, J., Gu, X, Cheng, Z., Yu, T., Xia, J., Wei, Y., Wu, W., Xie, X., Yin, W., Li, H., Liu, M., Xiao, Y., Gao, H., Guo, L., Xie, J., Wang, G., Jiang, R., Gao, Z., Jin, Q., Wang, J., Cao, B., 2020. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. Lancet 395, 497–506.

Hunt, A.G., 2014. Exponential growth in Ebola outbreak since May 14, 2014. Complexity 20, 8–11.

Iftimie, S., López-Azcone, A.F., Vallverdu, I., Hernánde-Flix, S., de Febrer, G., Parra, S., Hernández-Aguilera, A., Riu, F., Joven, J., Camps, J., Castro, A., 2020. First and second waves of coronavirus disease-19: a comparative study in hospitalized patients in Reus, Spain. http://dx.doi.org/10.1101/2020.12.10.20246959, MedRxiv.

Johnson, R.A., Wichern, D.W., 2002. Applied Multivariate Statistical Analysis. Prentice Hall, New Jersey.

Koepke, H., Clarke, B., 2013. A Bayesian criterion for cluster stability. Stat. Anal. Data Min. 6, 346–374.

Kriegel, H.P., Kröger, P., Sander, J., Zimek, A., 2001. Density-based clustering. WIREs Data Min. Knowl. Discov. 1, 231–240.

Lau, J.W., Green, P.J., 2007. Bayesian model-based clustering procedures. J. Comput. Graph. Statist. 16, 526–558.

Lindsten, F., Ohisson, G., Ljung, L., 2011. Just Relax and Come Clustering! A Convexication of *k*-Means Clustering. Technical Report, Linköpings Universitet.

Lloyd, S.P., 1982. Least squares quantization in PCM. IEEE Trans. Inform. Theory 28, 128–137.

MacQueen, J.B., 1967. Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, pp. 281–297.

Maier, B.F., Brockmann, D., 2020. Effective containment explains subexponential growth in recent confirmed COVID-19 cases in China. Science http://dx.doi.org/10.1126/science.abb4557.

McCullagh, P., 1983. Quasi-likelihood functions. Ann. Statist. 11, 59–67.

Pew Research Center, 2020. More than nine-in-ten people worldwide live in countries with travel restrictions amid COVID-19. https://www/pewreseach.org/fact-tank/2020/04/01.

de Picoli, S., Teixeira, J.J., Ribeiro, H.V., Malacarne, L.C., dos Santos, R.P., dos Santos Mendes, R., 2011. Spreading patterns of the influenza A (H1N1) pandemic. PLoS One 6, e17823.

Qin, L.X., Self, S.G., 2006. The Clustering of regression models method with applications in gene expression data. Biometrics 62, 526–533.

Soheily-Khah, S., Douzal-Chouakria, A., Gaussie, E., 2016. Generalized *k*-means-based clustering for temporal data under weighted and kernel time warp. Pattern Recognit. Lett. 75, 63–69.

Sun, K., Chen, J., Viboud, C., 2020. Early epidemiological analysis of the coronavirus disease 2019 outbreak based on crowdsourced data: a population-level observational study. Lancet Digit. Health.

Tibshirani, R., Walter, G., Hastie, T., 2001. Estimating the number of clusters in a dataset via the gap statistic. J. R. Stat. Soc. Ser. B Stat. Methodol. 63, 411–423.

Trauwaert, E., Kaufman, L., Rousseeuw, P., 1991. Fuzzy clustering algorithms based on the maximum likelihood principle. Fuzzy Sets and Systems 42, 213–227.

van der Vaart, A.W., 1998. Asymptotic Statistics. Cambridge University Press, Cambridge, UK.

Wang, J., 2010. Consistent selection of the number of clusters via cross validation. Biometrika 97, 893–904.

World Health Organization (WHO), 2020. Getting your workplace ready for COVID-19. February 27 2020.

Zhang, Y., Li, R., Tsai, C., 2010. Regularization parameter selections via generalized information criterion. J. Amer. Statist. Assoc. 105, 312–323.

Zhao, Y., Karypis, G., 2005. Hierarchical clustering algorithms for document datasets. Data Min. Knowl. Discov. 10, 141–168.