JOURNAL SUBMISSION

Classification and Regression via Integer Optimization for Neighbourhood Change

Alexander W. Olson[1], Kexin Zhang[1], Fernando Calderon-Figueroa[2], Ronen Yakubov[1], and Scott Sanner[1], Daniel Silver[2], Dani Arribas-Bel[3]
[1] Data-Driven Decision Making Lab, University of Toronto, Toronto, Canada
[2] Urban Genome Project, University of Toronto, Toronto, Canada
[3] Geographic Data Science Lab, University of Liverpool, Liverpool, UK

ARTICLE HISTORY
Compiled September 15, 2020

This paper applies a method we term "predictive clustering" to cluster neighbourhoods. Much of the literature in this direction is based on groupings built using intrinsic characteristics of each observation. Our approach departs from this framework by delineating clusters based on how the neighbourhood's features respond to a particular outcome of interest (e.g. income change). To do so, we leverage a classification and regression via integer optimization (CRIO) method that groups neighbourhoods according to their predictive characteristics and consistently outperforms traditional clustering methods along several metrics. The CRIO methodology contributes a novel methodological and conceptual capability to the literature on neighbourhood dynamics that can provide useful insights for policymaking.

Correspondence: Alexander W. Olson, Data-Driven Decision Making Lab, Department of Mechanical and Industrial Engineering, University of Toronto, Bahen Centre for Information Technology, St George St Toronto, ON M5S 2E4, Canada
alex.olson@mail.utoronto.ca

Introduction

The quantitative study of neighbourhoods has a long pedigree, going back at least to the Chicago School. A large proportion of these applications, at least until recently, fall under the umbrella of geodemographic research — the study of neighbourhoods through the characteristics of the populations which inhabit them (Harris, Sleight, and Webber 2005; Webber and Burrows 2018). Geodemographic applications thus characterize neighbourhoods through a sample of socioeconomic characteristics and then apply clustering techniques — typically K-Means — to derive neighbourhood typologies.

Cluster-based methods have gained significant popularity as computing power has increased, and user-friendly software has spread. In the process, researchers have expanded original, static approaches in several directions to refine how the theoretical notion of contextually varying urban areas is operationalized. For example, spatio-temporal classification (Singleton, Pavlis, and Longley 2016) and sequence analysis (Delmelle 2015; Delmelle 2016; Delmelle 2017) have been proposed. Much of this work takes a two-stage approach. First, a classification (or clustering) is built across neighbourhoods for each period, or all periods. An additional classification is then built to characterize the changes in category over time, for example, using sequence analysis. In essence, we first determine what is there, and then we determine how it changes.

This paper proposes the use of an alternative approach we term "predictive clustering". The key distinguishing feature of our proposed method is to cluster neighbourhoods based on the relationship between outcome and characteristics, rather than from the characteristics themselves. We define what *is* based upon how it *becomes.* More prosaically, we group neighbourhoods to maximize the quality of prediction made about their change, according to a set of linear regressions. The predictive clustering approach makes at least three noteworthy contributions to neighbourhood research:

- Conceptually, predictive clustering is highly distinct from traditional methods — in particular, linear regression. Because it relies on a predictive model, it can uncover how specific characteristics influence an outcome researchers are interested in studying. We can then observe heterogeneity in those influences and use it to cluster observations. This perspective makes neighbourhood dynamics fundamental to the definition of urban space.
- Methodologically, in contrast to traditional neighbourhood classifications, predictive clustering is based on a specific outcome to which areas similarly "respond". In other words, our proposal can be considered model-based clustering (Fraley and Raftery 2002). This framework defines the current state of a neighbourhood as a set of variables thought to influence trajectories of change. In this paper, following a long tradition of research, we use income change to illustrate the potential of this methodological feature of our approach (for a recent example, see Hochstenbach and Gent 2015).
- Practically, predictive clustering has clear implications that differentiate it from the traditional K-Means approach. Our approach can identify neighbourhoods that exhibit different processes of change even if they *look* similar at any given point in time, and similar dynamics even when they *look* different. Not only is this potentially useful for urban policymakers when designing interventions, it cannot be readily examined when one only considers characteristics of the neighbourhood, as in the traditional K-Means approach.

In developing our proposal, we build upon established frameworks of clustering and regression in machine learning. These models are used to study the relationship between a series of predictive variables (i.e. features) and a response variable (i.e. label). In predictive modelling for urban settings, rarely does a single model work well for all regions, and hence it can often be beneficial to cluster regions of similar traits or behaviour and to fit a separate model per cluster. Such approaches are traditionally referred to as cluster-wise linear regression (CLR) (Späth 1979; Späth 1981).

Unfortunately, CLR methods relying on traditional approaches like K-Means clustering are not guaranteed to produce the minimum error. This is in part because the K-Means clustering which groups together the most similar areas does not guarantee clustering which groups areas that *change* similarly. Furthermore, CLR models produce results dependent on their initialization, which can risk damaging reproducibility. In contrast, we propose to use a more recent variant of CLR termed Classification and Regression via Integer Optimization (CRIO) (Bertsimas and Shioda 2007) to ensure optimality, consistency, and reproducibility for CLR.

While the *optimal* CRIO framework has been proposed in the technical literature, to the best of our knowledge, it has not been widely applied either in general or specifically to analyzing neighbourhood dynamics; in this article, we aim to demonstrate that CRIO provides a powerful new tool for understanding neighbourhood change. By employing CRIO, which simultaneously optimizes the clusters generated for the data and the (robust) linear regression models used to predict income change, our model is better suited to create clusters within which census tracts *behave* similarly. This contrasts with more traditional models, which treat the clustering step and the prediction step as two separate stages, as in K-Means followed by per-cluster linear regression (KM+LR), each with different objectives. In simultaneously optimizing these two steps, we thus leverage a model that generates clusters better suited to the prediction task at hand, while also improving reliability over successive replications of the system.

In the following sections, we develop the notion of CRIO for CLR and present a case study for its application to prediction of neighbourhood income change. First, we review previous work on clustering-based regression models in the general literature, followed by a review of clustering approaches applied specifically in geographical analysis applications. Then we focus specifically on defining the CRIO methodology for CLR along with more traditional CLR approaches like KM+LR that rely on K-Means for clustering. Having outlined our methodology, we then proceed to apply it to a case study for the prediction of neighbourhood income change. Next, we discuss key differences between CRIO and KM+LR in this case study and the potential benefits and insights offered by the CRIO method for predictive clustering. We conclude with a discussion of future refinements of the CRIO methodology as well as further potential applications in neighbourhood effect and dynamics modelling.

In summary, our study argues for a novel approach to examining neighbourhood change. We deviate from existing methods by focusing on the similarity of change between neighbourhoods rather than the similarity of features. By employing a globally optimal method for predictive clustering, an existing technology, we ensure that (a) our method generates reproducible results and (b) the clustering and regression stages occur together, rather than separately. In our results section, we examine precisely how this difference in methodology achieves our underlying goal of finding clusters that are predictive of neighbourhood change.

Literature Review

The proposed contributions in this article represent the intersection of two mostly independent threads of research: general cluster-wise regression (CLR) algorithms and the general application of clustering methodologies in geographical analysis. We survey the literature in both areas in the following subsections.

Clustering in Geographical Analysis

There is a longstanding lineage of clustering applications in the context of geography. Clustering methods stem both from the geodemographics and the urban regional science traditions. In broad strokes, these methods can be classified according to their engagement with spatial constraints and temporal dynamics. Given that a comprehensive review exceeds the scope of this paper (see Knaap et al. 2019), our focus is limited to recent work related to our contribution.

Traditional clustering identifies typologies of places based on a set of shared attributes of their residents. These typologies are both static, in that they are based on cross-sectional data, and nonspatial since they do not impose formal spatial constraints. In the context of neighbourhoods, much of this work is known as geodemographics (Webber and Burrows 2018; Spielman and Folch 2015). The most common method used in this line of work, often combined with others, is cluster analysis via K-Means (Lloyd 1982). For instance, Wei and Knox (2014) use it to cluster census data tracts in three different tract-years before applying discriminant analysis to identify seven clusters, which they then qualitatively analyze to produce neighbourhood types. Using survey instead of census data, Spielman and Singleton (2015) combine K-Means with Ward's hierarchical clustering algorithm to identify 250 clusters that get grouped in ten neighbourhood types at the highest level. One of the main drawbacks of traditional clustering methods is their unsupervised nature — there may not be a clear alignment with the underlying mathematical objective of the clustering algorithms and the end goals of the intended research. Furthermore, these traditional unsupervised clustering methods are not well-suited to identify neighbourhood change and its underlying sociodemographic and spatial causes (Webber and Burrows 2018; Longley 2012).

Unlike geodemographics, regionalization methods inherently impose spatial constraints on clustering with the goal of aggregating subregions into a predetermined number of distinct contiguous regions with underlying similar features. Duque, Ramos and Surinach (2007) extensively discuss the work in this tradition. The goal of the regionalization algorithm may be spatial, sociodemographic, or both. Notable work in this area includes Duque, Anselin, and Rey's (2012) Network-Max-P Regions model. This is a global optimization model that attempts to group areas into a maximum number of regions while satisfying a threshold constraint, minimizing heterogeneity within the groups, and including spatial proximity constraints. Since Network-Max-P prohibits non-contiguous regions from sharing a label, it would be inappropriate for establishing generic neighbourhood types that may correspond to geographically distributed "islands". Similarly, the work of Rey and colleagues (2011) addresses the issue of neighbourhood change using a Max-P regionalization algorithm. They apply the algorithm on the same set of census tracts at two different time periods, 1990 and 2000. Then, they measure the changes between both spatial solutions as a proxy to changes in the spatial boundaries of neighbourhoods between the two periods. This line of work faces similar limitations to geodemographics in that it may inherently fail to find meaningful regions that explain label changes over time (Knaap et al. 2019).

A third kind of clustering methods are those that engage with changes over time but do not impose spatial constraints on the models as in regionalization. Such temporal clustering methods follow changes in neighbourhood composition (e.g., gentrification), their spatial boundaries (e.g., service coverage), or both (S. J. Rey et al. 2011). The general logic is, then, to identify the trajectories of specific urban spaces (Knaap et al. 2019). One widely used method is to run an initial geodemographic analysis that segments an urban space into neighbourhood classes. Then, the change in neighbourhood classes is modelled. In her recent work, Delmelle (2016; 2017) clusters sequential patterns of class label changes over time. The extended method combines a self-organizing map to project the feature space onto a 2D surface with K-Means to group the resulting areas (Delmelle 2017). Then, the sequences are clustered using Ward's hierarchical algorithm. An important advantage of the method is that it is asynchronous, i.e., it allows to identify similar trajectories for different neighbourhoods at different segments of time (Knaap et al. 2019). Other recent research uses different clustering methods and data to assess spatial changes over time. Reades, De Souza and Hubard (2019) aim to both classify and predict gentrifying neighbourhoods in London through Principal Component Analysis and Random Forests on census data. Using data from the venue rating application "Yelp," Glaeser, Kim, and Luca (2018) apply linear regression models to predict various metrics of gentrification. Unlike unsupervised clustering, these methods do have a clear success metric that can be used to verify whether the model adequately captures the data. However, clustering and prediction remain as separate processes in these models rather than sharing a single mathematical objective used to globally optimize both.

The clustering method we leverage in this article aims to tackle some of the limitations discussed above by combining clustering together with prediction of temporal change. Mohamed and colleagues (2013) explored a similar method by initially using K-Means to create sets of clusters, and then applying separate Ordered Probit Regressions to each cluster. While this two-step process of clustering followed by regression proved to be a step in the right direction, the clustering is independent of the regression and therefore does not necessarily yield clusters that provide the optimal regression. In contrast, our CRIO approach to CLR produces clusters that directly minimize regression error, thus providing the optimal clusters for prediction.

Cluster-wise Regression Models

Cluster-wise regression (CLR) is a type of regression from a vector of predictive variables (features) to a response variable (label), where each observation (consisting of predictive variables) is assigned to a cluster that determines the regression model that is used to predict the response variable across all models. Researchers have primarily explored CLR algorithms in two directions that we discuss next: heuristic solution approaches and globally optimal mathematical programming approaches.

To obtain an approximate solution in a relatively short amount of computation time, researchers have proposed heuristic approaches to solve the CLR problem. One obvious approach to CLR is simply to cluster the data with K-Means (Lloyd 1982) according to the predictors and then perform a linear regression on the data in each cluster to obtain a per cluster regression model. We refer to this model as K-Means plus linear regression (KM+LR) and note that an analysis of clusters of KM+LR is simply an analysis of K-Means clusters themselves — the regression problem has no impact on the clustering. In order to unify clustering and regression under a single objective to jointly minimize regression error, Späth (1979) proposed an exchange method in which after starting with an initial cluster assignment, two items from different clusters are exchanged if the total squared error is reduced. Although this method is efficient, Späth (1981) later improved the exchange method to achieve faster convergence by moving a single item from one cluster to the other if the total squared error is reduced. These heuristic methods produce acceptable solutions in many contexts; however, they are highly dependent on the initialization, and thus, do not guarantee a globally optimal solution.

Given the drawbacks of heuristic approaches, many researchers started to explore mathematical programming formulations of the CLR problem. Although these approaches were not applied to the neighbourhood analysis problem we examine here, these methods nonetheless provide the fundamental global optimization approach that we leverage in this article. Lau, Leung, and Tse (1999) proposed one of the first non-linear programming procedures to solve a variant of the CLR problem. However, since their optimization model is non-linear, their solution is subject to local optima and thus dependent on initialization. Zhang (2003) developed a centre-based clustering algorithm to reduce dependence on initialization, called K-Harmonic Means, which converges to a better local optimum than the heuristic approaches. To facilitate computationally efficient solutions that are robust to outliers, other researchers have looked at a variation of the CLR problem: minimizing total absolute error instead of the total squared error. Bertsimas and Shioda (2007) proposed a mixed-integer linear programming (MILP) model called Classification and Regression via Integer Optimization (CRIO) to solve this variation of the CLR problem, which is the basis upon which we develop our approach. Follow-on work by Zhu, Li, and Kong (2012) explored a similar MILP approach and explored symmetry breaking approaches to enhancing scalability. Recent work by Park et al. (2017) switched back to a less computationally efficient squared error objective, but also provided column generation techniques to improve the scalability of his mathematical programming framework.

Each of the above approaches focused on a unique algorithm to solve a CLR problem. However, while addressing similar tasks, these papers did not apply their methods to neighbourhood change analysis, nor our specific problem set. We built both a heuristic model based on K-Means followed by per-cluster linear regression (KM+LR), which has fast convergence, as well as a CRIO model that guarantees global optimality to solve our CLR problem. We aim to highlight the differences of these two CLR approaches and the advantages offered by CRIO for neighbourhood change analysis.

Data

Our experiments attempt to predict income change between 2000 and 2010 in New York City. To facilitate this, we use the Longitudinal Tract Database (LTDB), which harmonizes census data to 2010 boundaries (Logan, Xu, and Stults 2014). This study focuses on data from 2000 and 2010 in Manhattan to obtain a dataset where accurate results can be obtained in reasonable computation time, and results can be more readily interpreted qualitatively.

The census features that were considered in our study were chosen to represent basic population characteristics, economic measures, educational attributes, geographic and migration information and housing aspects of the census tract. We normalize within each census tract according to the total population of that tract. The value for prediction is change in per capita income from 2000 to 2010. It is calculated as the log of per capita income in 2010 divided by the per capita income in 2000.

We log transform all variables (feature and label). The main reasons to log transform are due to the wide dynamic range of the data, where a small number of extreme values can disproportionately affect the regression; furthermore, for the label that is computed as a ratio, we would expect change to be linear and additive in terms of a log-ratio (Gelman and Hill 2006).

Before we use the dataset in our analysis, we performed several pre-processing steps and cleaning to prepare the data. We started with 288 census tracts total. Firstly, since we used per capita income in 2000 and 2010 to create the labels, we only kept tracts that exist in both years. We only removed one tract this way — Liberty Island, which has no registered inhabitants, is recorded in the 2010 dataset but not the 2000 dataset. Next, we removed rows where any value was recorded as -999, indicating a missing entry, a step which removed seven individual entries — all in the income per capita feature. We then divided the 2010 data by the 2000 data to obtain the change between the two censuses. Where a value was 0 in the 2000 census, this provides a division by zero error. For columns that now contain $>10\%$ NaN values, we remove the entire column. For the remainder, we move the census tract row containing this NaN value. The reason for this hybrid removal process is to (a) prevent removing entire columns because they simply contain one invalid entry; and (b) prevent removing entire rows simply because they have one column, which is frequently invalid. This hybrid process allowed us to retain a larger proportion of both the census tracts and the features than entirely removing by column or by row. We repeat this process after taking the log of all columns. After cleaning, we are left with 236 rows of the original 288, with only valid entries remaining.

The last step was to standardize all features and labels. We first subtracted each variable by its mean and then divided it by the standard deviation of that variable. Then, each standardized value represented the number of standard deviations away from the mean of that variable. This step was vital since we wanted to compare weights on features and mutual information score of each feature.

One limitation of the CRIO model comes from the nature of MILP optimization tasks. The computation time required to solve the problem and hence produce optimal clusters increases dramatically with each variable included in the model. As a result of this limitation, we employed a maximum of 5 census features at a time in prediction.

To determine the features best suited for inclusion in our model, we calculated the mutual information (MI) of each feature with the prediction variable (Cover and Thomas 1991). This resulted in the following selection of the highest MI features:

(1) **prof: Professional Employees:** The proportion of residents in professional occupations.
(2) **col: College Degree+:** The proportion of people in the census tract with at least a four-year college degree.
(3) **flabf: Females in Labor Force:** The proportion of working women.
(4) **multi: Multi-Family Units:** The proportion of single homes containing more than one family.
(5) **own: Owner-Occupied Housing Units:** The proportion of people living in homes that they own.

Methods

In this section, we first begin with notation definitions and then proceed to outline our Cluster-wise Linear Regression (CLR) methodology for the two methods we compare in this article: KM+LR as well as the technical definition of the CRIO approach that motivates this research investigation.

Optimization Metrics for Linear Regression

We assume there are n observations consisting of f predictive variables $X_{i,j}$ (for $i \in \{1,\dots,n\}$ and $j \in \{1,\dots,f\}$) used to predict response variable Y_i. Specifically, the ith observation can be written as a tuple of the form ($X_{i,1}$,...,$X_{i,f}$, Y_i).

In linear regression, we use a linear weighting of the predictive variables to predict the response variable; formally, letting $\mathbf{w} = (w_1, \dots, w_f)$ be a vector of f regression coefficients and b a bias term, we can obtain a prediction $\hat{Y_i}$ of the true value Y_i as follows: $\hat{Y_i} = \sum_{j=1}^{f} w_j X_{i,j}$. Given our n observations, we can optimize our linear regression according to either mean absolute error (MAE) or mean squared error (MSE) as defined below.

(1) Mean Squared Error (MSE):

$$\min_{\mathbf{w},b} \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{f} w_j X_{i,j} - b \right)^2 \tag{1}$$

(2) Mean Absolute Error (MAE):

$$\min_{\mathbf{w},b} \frac{1}{n} \sum_{i=1}^{n} \left| Y_i - \sum_{j=1}^{f} w_j X_{i,j} - b \right| \tag{2}$$

While mostly similar in calculation, MAE differs from MSE in that it penalizes outliers less substantially – the squared term in MSE causes the cost of outliers to increase quadratically. For this reason, the use of MAE is considered to lead to *robust* linear regression that is less sensitive to outliers and which also has useful sparsity (zero weight) inducing properties that lead to improved interpretability that can be observed in the experimental results for the CRIO method that uses MAE. We explicitly optimize the MAE metric in both our KM+LR and CRIO methods, although we report performance in terms of both MAE and MSE for each model.

General Cluster-wise Linear Regression (CLR)

As discussed previously, cluster-wise linear regression (CLR) is a general framework for piecewise linear regression, where we aim to find a fixed number of cluster assignments for the data and then apply a linear regression within each cluster. Formally, if we assume there are n observations and that we define each observation as ($X_{i,1}$,...,$X_{i,f}$, Y_i), the goal is to look for K clusters, C_1, ..., C_K given n observations, such that the following properties hold:

- each cluster contains a set of the observations: $C_k \subset N = \{1, ..., n\}$,

- each cluster must contain at least one observation: $|C_k| > 0$,
- clusters do not overlap: $C_j \cap C_k = 0$ for $j \neq k$, and
- each observation is assigned to a cluster: $\cup_{k=1}^{K} C_k = K$.

For each cluster k, we will find f cluster-specific weights $(w_{1,k}, w_{2,k}, ..., w_{f,k})$ and K bias terms. How we find the clusters and optimize the linear regression models depends on the specific approach we take.

K-Means plus Linear Regression (KM+LR)

As previously done in Mohamed et al. (2013), we approach CLR in two stages. In the first stage, we apply the popular K-Means algorithm (Lloyd 1982) to find K clusters. In the second stage, we simply apply standard linear regression using MAE as the objective to fit the data assigned to each cluster, hence resulting in one MAE-based linear regression model per cluster. We refer to this method as K-Means plus Linear Regression (KM+LR). We note that MAE is used as the regression optimization metric not only for its robustness properties, but also to match CRIO, discussed next.

Classification and Regression via Integer Optimization (CRIO)

We now provide the mathematical derivation of our Classification and Regression via Integer Optimization (CRIO) model that reflects the final result presented in Bertsimas and Shioda (2007, see Eq (15)). Unlike KM+LR, we aim to do both clustering and robust (MAE-based) linear regression *simultaneously* in this model.

In the objective function (3), the first term is the MAE. The indicator, $c_{i,k}$ =1 when tract i belongs to cluster k; and $c_{i,k} = 0$, otherwise. The per-datum absolute error measure, $e_{i,k} = \left|\sum_{j=1}^{f} w_{j,k}\ X_{i,j} + b_k - y_i\right|$, is multiplied by a binary cluster assignment indicator $c_{i,k}$ to ensure that we are only considering error resulting from the linear regression that each tract is assigned to. Also, we use the constraint in (4) to ensure each tract can only be assigned to one of the K clusters.

$$\min \qquad \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{m} c_{i,k}\Big|\sum_{j=1}^{f} w_{j,k}\ X_{i,j} + b_k - y_i\Big| \tag{3}$$

$$\begin{aligned}
s.t. \quad & \sum_{k=1}^{m} c_{i,k} = 1 && \forall\ i = 1, ..., n && (4)\\
& c_{i,k} \in \{0, 1\} && \forall\ i = 1, ..., n,\ \ k = 1, ..., m && (5)\\
& w_{j,k} \in \mathbf{R} && \forall\ j = 1, ..., f,\ \ k = 1, ..., m && (6)\\
& b_k \in \mathbf{R} && \forall\ k = 1, ..., m && (7)
\end{aligned}$$

In this model, the objective function is nonlinear because of the absolute value and multiplication of absolute value with a binary cluster assignment indicator $c_{i,k}$. Therefore, the model was reformulated as a mixed integer linear program (MILP) for computational efficiency and implementation. Though we initially tested three encoding methods, we ultimately selected the following "Big-M" encoding as this method produced the best results in our preliminary testing.

$$\min \quad \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{m} e_{i,k} \tag{8}$$

$$s.t. \quad \sum_{k=1}^{m} c_{i,k} = 1 \qquad \forall\ i = 1, ..., n \tag{9}$$

$$\sum_{j=1}^{f} w_{j,k}\ X_{i,j} + b_k - y_i - e_{i,k} \leq M(1 - c_{i,k}) \qquad \forall\ i = 1, ..., n,\ \ k = 1, ..., m \tag{10}$$

$$\sum_{j=1}^{f} -w_{j,k}\ X_{i,j} - b_k + y_i - e_{i,k} \leq M(1 - c_{i,k}) \qquad \forall\ i = 1, ..., n,\ \ k = 1, ..., m \tag{11}$$

$$e_{i,k} \geq 0 \qquad \forall\ i = 1, ..., n,\ \ k = 1, ..., m \tag{12}$$

$$c_{i,k} \in \{0, 1\} \qquad \forall\ i = 1, ..., n,\ \ k = 1, ..., m \tag{13}$$

$$w_{j,k} \in \mathbf{R} \qquad \forall\ j = 1, ..., f,\ \ k = 1, ..., m \tag{14}$$

$$b_k \in \mathbf{R} \qquad \forall\ k = 1, ..., m \tag{15}$$

In this Big-M encoding, we replaced the absolute values in the objective function by enforcing $e_{i,k} = \left|\sum_{j=1}^{f} w_{j,k}\ X_{i,j} + b_k - y_i\right|$ with constraints (10) and (11) (Bertsimas and Tsitsiklis 1997). We then used the Big-M encoding to remove $c_{i,k}$ in the objective function and M was set to the max distance between all pairs of data points. We added constraint (12) to ensure $e_{i,k} = 0$ when $c_{i,k} = 0$ in constraint (10) and (11). To see that this compact encoding is correct, when $c_{i,k} = 1$, the RHS of constraint (10) and 11 becomes 0 and $e_{i,k} = \left|\sum_{j=1}^{f} w_{j,k}\ X_{i,j} + b_k - y_i\right|$. In contrast, if $c_{i,k} = 0$, $e_{i,k} = 0$ due to constraint (12) and the minimization of the objective function.

We note that the above CRIO MILP closely relates to a specific type of finite mixture model (see, e.g., McLachlan, Lee, and Rathnayake (2019) for recent work) assuming Laplacian noise (MAE) as previously pointed out by Lau, Leung, and Tse (1999); however, in our case, the latent class membership variables (the $c_{i,k}$) are conditioned on and directly optimized rather than being optimized in a marginal likelihood framework typically used in mixture model fitting. The advantage of our CRIO model is that MILP-based solvers can provide a *globally optimal* solution that *guarantees reproducibility* (up to symmetries) as opposed to standard finite mixture modelling approaches that rely on non-convex optimization algorithms like Expectation Maximization (EM) known to be subject to local optima (Zhang 2003).

In terms of implementation of the above CRIO MILP, we used the Gurobi solver (a commercial solver that is free for academic use) on a desktop computer with a six-core Intel i7-8700 CPU at 3.20GHz and 32GB of RAM. We limited running time to 4 hours per MILP solution and found that this was sufficient to support up to 5 features and 5 clusters for our 236 census tracts. This was sufficient for our initial investigation of

CRIO for neighbourhood analysis in this article, but we discuss avenues for potential enhanced scalability for CRIO in our concluding future work discussion.

Results

In this section, we compare the CLR performance of our CRIO method to the KM+LR baseline method and measure performance using both MAE and MSE. We qualitatively examine the weights and feature averages for the clusters of both KM+LR and CRIO and also visualize the cluster assignments on a map. Finally, for KM+LR (the only method examined here with outcomes dependent on initialization), we additionally examine the reproducibility of its K-Means clusterings across multiple runs.

Illustrative Example

Before we apply KM+LR and CRIO to real data, we first begin with an illustrative example that highlights key differences between the two approaches. For this example, we refer to Figure 1, where we have generated a synthetic dataset consisting of samples of points i (with Gaussian distributed error) from three different regression lines of the form $Y_i = w_1 X_{i,1} + b$ (nb. $f = 1$ according to our notation from the previous section) with randomly generated w_1 and b parameters. These plots show the clusters and regression lines (in colour) for $K \in \{1, 2, 3\}$ for each of KM+LR and CRIO.

The optimal result for either model would be to assign each point to the same cluster for the line that generated it and to recover the optimal w_1 and b for each line. However, we chose this example specifically to demonstrate a limitation of KM+LR to achieve this optimal result. Because KM+LR uses a two-stage clustering algorithm and only clusters each sample i on the feature (i.e., the x-axis $X_i, 1$) in the first stage, it is restricted to assigning clusters by partitioning the x-axis and then to determining optimal regressions for each of these x-axis partitions at the second stage. In contrast, CRIO's joint CLR optimization has the ability to determine the cluster assignment for each i considering both its feature $X_i, 1$ and it's label Y_i and how well both fit a potential regression line. In this way, CRIO is clearly able to recover the optimal underlying generative model for the data without "knowledge" of the existence of these generating models or their parameters. Hence, Figure 1 demonstrates that CRIO can capture clusters that are predictive of the label (and change, if this is the chosen label), whereas KM+LR is limited to capture clusters based on feature similarity alone.

Average Error per Tract

Leveraging insight from the illustrative example, we now proceed to compare KM+LR and CRIO on our previously described New York census data. The first evaluation of performance is through comparing the Mean Absolute Error (MAE) and Mean Squared Error (MSE) per tract displayed in Table 2. Here we show results for KM+LR and CRIO for four different values of K, as well as the values for LR at a K of 1 (where KM+LR reduces to standard linear regression). Both KM+LR and CRIO get progressively better as K increases since they have more clusters and linear regression models to fit, thus reducing error. In terms of comparison, we observe that CRIO performs better on both metrics due to its global optimization approach. Additionally, both of these models outperform standard LR, particularly at higher values of K.

KM+LR Consistency

After executing the KM+LR model across a broad range of random initializations, we were able to produce the consistency results shown in the CRC column of Table 2. The cross-run consistency (CRC) value is calculated as, for every pair of runs, the proportion of model assignments that are the same in both runs divided by the total number of model assignments possible. As can be seen from this table, a low k value does produce stable clusters as indicated by high CRC, but as the number of clusters increases, the stability (i.e., CRC) substantially diminishes. This is representative of a critical problem with clustering methods such as K-Means that underlies KM+LR — re-running the same model on the same data is not guaranteed to reproduce results.

Exploring the Nature of Clusters Produced by KM+LR and CRIO

After comparing the results quantitatively, we now visualize the cluster assignment on the New York City map to analyze the results qualitatively in Figure 4 as well as the individual weights and feature averages for each cluster in Table 1. In the cluster analysis that follows, it is critical to recall that the clusterings for KM+LR are simply due to K-Means since linear regression (LR) is only applied after clustering in the KM+LR methodology.

In Figure 4, we show KM+LR in the top row and CRIO in the bottom row while columns correspond to different K. It is evident that for the same K, there is a substantial qualitative difference in the clusterings produced by KM+LR and CRIO. This is clear already when K = 2. K-Means divides Manhattan primarily between North and South, whereas CRIO produces clusters that span this division. This is strong initial evidence that areas that look “the same” from the traditional K-Means point of view might “behave” very differently, while “different” areas might change according to a similar model.

Table 1 helps to unpack what the clusters consist of. It shows the individual weights and feature averages for each cluster for differing numbers of clusters K. Since our main objective is to articulate the generic differences between the two methods, highlighting K = 2 provides a straightforward illustration, especially when we primarily focus on education, which at this level seems to be a key variable. K-Means classifies neighbourhoods into high and low educated areas (average standardized college degree values of 0.99 and -0.54 respectively), which have correspondingly high and low levels of homeownership (average homeownership of 0.44 and -0.24 respectively). Higher-educated areas tend to exhibit greater income growth; growth in the less-educated areas is slower. Within highly educated areas, the influence of homeownership is lower, making minimal impact on the prediction. In low educated areas, however, the impact of homeownership is double.

For K=2, CRIO takes an entirely different approach. It divides Manhattan into areas in which the factors driving income change between 2000 and 2010 are similar, in the first cluster, and areas which can be predicted based on a combination of all features in the second. It includes this ’static’ cluster in every increasing value for K. For example, at K=5, the final cluster also contains zero weightings, although this time, the participation in this group is lower. Returning to K=2, the exception to this is that homeownership is again considered important as a factor for income change.

It is additionally worth noting the number of census tracts assigned to each cluster between the models. For the KM+LR models, cluster assignment is lopsided, with some clusters containing a very small number of tracts. For K=4 and K=5, KM+LR has a cluster containing only a single tract, which is a poor utilization of the additional clustering opportunity. On the other hand, CRIO assigns roughly even proportions of tracts to each cluster, even as the number of clusters increases. This demonstrates better utilization of the models available.

Cluster-based Regression Analysis

In Figure 3, we show the linear regressions per-feature for KM+LR (top) and CRIO (bottom) for each of the five features (five columns) with $K = 5$ for both algorithms. Each graph shows a linear regression overlaid on a scatterplot. The scatterplot shows each tract as a point providing both its feature value (x-axis) and corresponding income change (y-axis); the CRIO models assign each tract to one of 5 cluster models, which is indicated by the colour of the point. The regression plots for the five cluster models were generated by zeroing out all features except for the feature being shown in the column and showing the predicted change in income per capita as a linear response to that feature. We remark that the slope of this line does reflect the weight of that feature in the model, but that the y-intercept (vertical offset) of the line should be ignored since all other features were zeroed out for this analysis of individual feature response.

In the first row of Figure 3 we show the results for KM+LR. These regressions are highly stochastic and include many extremely steep regression slopes that are highly unlikely in the context of the points assigned to that model (same colour) and strongly indicative of overfitting and response to individual outliers. Additionally, it is possible to observe how KM+LR simply splits its clusters on feature values as indicated by the horizontal (colour) separation of the clusters with regard to the y-axis for many of the features. For this reason, we note that there is a very high variance of income change (y-axis) per feature, indicating that the clusters do not correspond to tracts that behave similarly with regard to income change. Finally, KM+LR does not optimally use all five models to produce different predictions. Instead, many of the models are used for one or a few noisy data points (as reflected in Table 1). Hence, these models are highly overfit and leave the rest of the points to be covered by the few remaining regression models.

In sharp contrast, the CRIO plots in the second row of Figure 3 demonstrate how each cluster corresponds to a different range of incomes, as indicated by the vertical separation of cluster models (colours) on the y-axis. We remark that the CRIO models opt for much more reasonable feature coefficients in the context of the points assigned to that model (same colour), and where the slopes are more extreme it does appear to fit well to the associated data. Across plots in the bottom row, we can see how each of the five models focuses on responses to different features — flat lines indicate that a model is ignoring that feature, and likely focusing on others. This indicates that CRIO is more robust to noise. Overall, these models are visibly more plausible for the data, indicating the clusters much better reflect the predictive process than the corresponding results in the first row KM+LR plots.

Case Study

While it is beyond the scope of this comparative methodological article to provide a comprehensive analysis of income change in New York City, we now highlight a case study that elucidates key differences between the clustering and regression approaches of the KM+LR and CRIO models. Specifically, to highlight the different underlying approaches employed by KM+LR and CRIO, we look at the example of two census tracts in New York — 148.01 and 148.02. These tracts existed separately since at least 1970, which is the first decade included in the LTDB, but due to their decimal difference, they would have at one point existed as a single tract. Indeed, they are directly adjacent to one another (see Figure 2). In addition to being directly adjacent, both tracts' census features are mostly in alignment with one another, as seen in Table 3. Due to their highly similar features, the KM+LR model naturally predicts very similar outcomes for the two tracts. However, despite this, the two tracts have very different outcomes in reality. By contrast, our CRIO model can model the change in the two tracts differently (Table 4). In tract 148.01, the model identifies homeownership as the critical factor in changing income, whereas 148.02 appears largely to remain static in this prediction. This demonstrates a fundamental difference in the two approaches to income prediction. KM+LR is only capable of identifying groups of tracts that are similar in the present features. This does result in spatially cohesive clustering, but as we see from this example, spatial coherence is not necessarily a desirable quality for the prediction task of understanding neighbourhoods' susceptibility to change by similar processes. On the other hand, CRIO produces non-spatially coherent clustering, but does so as it can capture the underlying similarity in change, rather than just high-level feature similarity.

Conclusion

In this comparative methodology article, we applied the "Classification and Regression via Integer Optimization" (CRIO) methodology to group census tracts into neighbourhood clusters. Whereas traditional approaches using K-Means and Linear Regression (KM+LR) group neighbourhoods based upon shared static attributes, CRIO groups neighbourhoods based upon common dynamic processes. In short, what superficially looks like highly related clustering and regression methods turn out to produce very different analyses. This highlights the critical importance of carefully considering the assumptions underlying combined clustering and regression approaches when using them to analyze neighbourhood change.

Not only does CRIO constitute a novel conceptual approach to neighbourhood clustering, but it also outperforms traditional methods along several metrics. By using a mixed integer linear programming (MILP) model, CRIO ensures optimal, reproducible solutions. Moreover, CRIO achieved both lower mean absolute error and mean squared error than the KM+LR baseline method for CLR. Though CRIO does involve a longer computation time than existing methods, it produces a robust, reproducible solution even when the problem size is large.

We used income change in Manhattan to demonstrate the potential of CRIO in contrast to standard methods. Quantitatively, CRIO produces a solution with a lower absolute error than the baseline KM+LR method and more consistent cluster assignments. Qualitatively, CRIO reveals patterns that KM+LR obscures. In particular, it cuts across static divisions to uncover areas held together by their potential to change.

Most fundamentally, CRIO provides a quantitative methodology for perceiving the city not only as a collection of common attributes but as an evolving space of processes. It offers a way to identify what and where these processes are rigorously. Beyond this conceptual and methodological reorientation, it has significant practical potential. A policymaker can use CRIO-based tools to identify areas that, despite any current differences, tend to change in similar ways. This can help in both evaluating and designing interventions.

The application of CRIO applied to neighbourhood analysis is new, and as such it, presents challenges and opportunities. As noted previously, the basic version of CRIO that we implemented is not highly scalable, which limited us to regression models over five features and five clusters for 236 census tracts in this analysis; quite simply, this is the computational price to be paid for global optimality that previous approaches lacked. Future work should look at incorporating one or more of the scalability enhancements for the CRIO methodology (Bertsimas and Shioda 2007; Zhu, Li, and Kong 2012; Park et al. 2017) that involve a range of techniques from pre-clustering to symmetry breaking to column generation methods for optimization.

Even with its current moderate scalability, CRIO still represents a valuable contribution to cluster analysis for neighbourhoods. This article only managed to introduce the idea of predictive clustering for neighbourhood analysis, compare key differences of CRIO compared to KM+LR from a few perspectives, and present a brief case study predicting income change in Manhattan. Future work should continue to explore the application and extension of CRIO to a variety of neighbourhood response variables to better understand whether neighbourhood clusterings are consistent across multiple response variables and the underlying (causal) mechanisms that explain the common "behavioural" evolution underlying these response-dependent clusterings.

Acknowledgments

We kindly thank the anonymous reviewers whose suggestions have helped to improve the article. We also thank the reviewer who suggested the analysis in Figures 1 and 3 that has substantially helped with the interpretation and comparison of the two models.

References

Bertsimas, Dimitris and Romy Shioda (2007). "Classification and Regression via Integer Optimization". In: *Operations Research* 55.2, pp. 252–271. URL: https://doi.org/10.1287/opre.1060.0360.

Bertsimas, Dimitris and John Tsitsiklis (1997). *Introduction to Linear Optimization.* 1st. Athena Scientific. ISBN: 1886529191.

Cover, TM and JA Thomas (1991). "Elements of information theory: Wiley Online Library". In:

Delmelle, Elizabeth C (2015). "Five decades of neighborhood classifications and their transitions: A comparison of four US cities, 1970–2010". In: *Applied Geography* 57, pp. 1–11.

– (2016). "Mapping the DNA of urban neighborhoods: clustering longitudinal sequences of neighborhood socioeconomic change". In: *Annals of the American Association of Geographers* 106.1, pp. 36–56.

– (2017). "Differentiating pathways of neighborhood change in 50 US metropolitan areas". In: *Environment and planning A* 49.10, pp. 2402–2424.

Duque, Juan C, Luc Anselin, and Rey (2012). "The max-p-regions problem". In: *Journal of Regional Science* 52.3, pp. 397–419.

Duque, Juan Carlos, Raúl Ramos, and Jordi Surinach (2007). "Supervised regionalization methods: A survey". In: *International Regional Science Review* 30.3, pp. 195–220. ISSN: 01600176. DOI: 10.1177/0160017607301605.

Fraley, Chris and Adrian E Raftery (2002). "Model-based clustering, discriminant analysis, and density estimation". In: *Journal of the American statistical Association* 97.458, pp. 611–631.

Gelman, Andrew and Jennifer Hill (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press.

Glaeser, Edward L, Hyunjin Kim, and Michael Luca (2018). "Nowcasting gentrification: using yelp data to quantify neighborhood change". In: *AEA Papers and Proceedings*. Vol. 108, pp. 77–82.

Harris, Richard, Peter Sleight, and Richard Webber (2005). *Geodemographics, GIS and neighbourhood targeting*. Vol. 8. John Wiley & Sons.

Hochstenbach, Cody and W. P. Van Gent (2015). "An Anatomy of Gentrification Processes: Variegating Causes of Neighbourhood Change". In: *Environment and Planning A: Economy and Space* 47.7, pp. 1480–1501. URL: `https://doi.org/10.1177/0308518X15595771`.

Knaap, Elijah et al. (2019). "The Dynamics of Urban Neighborhoods : A Survey of Approaches for Modeling Socio-Spatial Structure". In: *GIS & Quantitative Geography*, pp. 1–28. DOI: `10.31235/OSF.IO/3FRCZ`. URL: `https://osf.io/preprints/socarxiv/3frcz/`.

Lau, Kin-nam, Pui-lam Leung, and Ka-kit Tse (1999). "A mathematical programming approach to clusterwise regression model and its extensions". In: *European Journal of Operational Research* 116.3, pp. 640–652. URL: `https://doi.org/10.1016/S0377-2217(98)00052-6`.

Lloyd, Stuart P. (1982). "Least squares quantization in pcm". In: *IEEE Transactions on Information Theory* 28, pp. 129–137.

Logan, John R., Zengwang Xu, and Brian J. Stults (2014). "Interpolating U.S. Decennial Census Tract Data from as Early as 1970 to 2010: A Longitudinal Tract Database". In: *The Professional Geographer* 66.3, pp. 412–420. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4134912/`.

Longley, Paul A. (2012). "Geodemographics and the practices of geographic information science". In: *International Journal of Geographical Information Science* 26.12, pp. 2227–2237. DOI: `10.1080/13658816.2012.719623`.

McLachlan, Geoffrey J, Sharon X Lee, and Suren I Rathnayake (2019). "Finite mixture models". In: *Annual review of statistics and its application* 6, pp. 355–378.

Mohamed, Mohamed Gomaa et al. (2013). "A clustering regression approach: A comprehensive injury severity analysis of pedestrian–vehicle crashes in New York, US and Montreal, Canada". In: *Safety science* 54, pp. 27–37.

Park, Young Woong et al. (2017). "Algorithms for Generalized Clusterwise Linear Regression". In: *INFORMS Journal on Computing* 29, pp. 301–317.

Reades, Jonathan, Jordan De Souza, and Phil Hubbard (2019). "Understanding urban gentrification through machine learning". In: *Urban Studies* 56.5, pp. 922–942.

Rey, Sergio J. et al. (2011). "Measuring spatial dynamics in metropolitan areas". In: *Economic Development Quarterly* 25.1, pp. 54–64. ISSN: 08912424. DOI: `10.1177/0891242410383414`.

Singleton, Alexander, Michail Pavlis, and Paul A. Longley (2016). "The stability of geodemographic cluster assignments over an intercensal period". In: *Journal of Geographical Systems* 18.2, pp. 97–123.

Späth, Helmuth (1979). "Algorithm 39 Clusterwise linear regression". In: *Computing* 22.4, pp. 367–373. URL: `https://doi.org/10.1007/BF02265317`.

– (1981). "A fast algorithm for clusterwise linear regression". In: *Computing* 29.2, pp. 175–181. URL: `https://doi.org/10.1007/BF02249940`.

Spielman, Seth E. and David C. Folch (2015). “Reducing uncertainty in the American Community Survey through data-driven regionalization”. In: *PloS one* 10.2, e0115626.

Webber, Richard and Roger Burrows (2018). *The Predictive Postcode: The Geodemographic Classification of British Society*. SAGE.

Wei, Fang and Paul L Knox (2014). “Neighborhood change in metropolitan America, 1990 to 2010”. In: *Urban Affairs Review* 50.4, pp. 459–489.

Zhang, Bin (2003). “Regression clustering”. In: *Third IEEE International Conference on Data Mining Research*, pp. 451–458. URL: https://ieeexplore.ieee.org/document/1250952.

Zhu, Zhen, Yan Li, and Nan Kong (2012). “Clusterwise Linear Regression with the Least Sum of Absolute Deviations – An MIP Approach”. In: *International Journal of Operations Research* 9.3, pp. 162–172. URL: https://www.semanticscholar.org/paper/Clusterwise-Linear-Regression-with-the-Least-Sum-of-Zhu-Li/84e47b61abcb0ec5743b180ed734978d76583c0a.

			KM+LR							CRIO					
K	**id**	**type**	#	**professional**	**college degree +**	**females in labor force**	**multi family units**	**own home**	**median Δ income**	#	**professional**	**college degree +**	**females in labor force**	**multi family units**	**own home**	**median Δ income**
2	**1**	ftr	83	1.02	0.99	0.80	0.48	0.44	0.56	100	-0.02	-0.01	0.06	0.01	0.01	0.01
		wgt		0.45	0.27	-0.27	-0.04	0.07			0.00	0.00	0.00	0.00	0.21	
	2	ftr	153	-0.56	-0.54	-0.43	-0.26	-0.24	-0.35	136	0.02	0.00	-0.05	-0.01	-0.01	-0.5
		wgt		0.28	0.35	-0.25	0.15	0.14			0.54	0.28	0.00	0.40	0.15	
3	**1**	ftr	8	2.83	2.84	2.19	2.98	2.72	2.05	68	-0.04	-0.06	-0.05	-0.15	-0.10	0.41
		wgt		-23.16	6.16	10.83	-2.12	0.57			0.38	0.23	0.00	1.33	0.00	
	2	ftr	135	-0.62	-0.63	-0.53	-0.28	-0.29	-0.52	79	0.16	0.13	0.17	0.05	0.02	0
		wgt		0.26	0.47	-0.34	0.17	0.15			0.00	0.04	0.00	0.00	0.20	
	3	ftr	93	0.66	0.67	0.58	0.15	0.19	0.45	89	-0.11	-0.06	-0.12	0.07	0.06	-0.59
		wgt		0.50	0.17	-0.16	0.33	0.12			0.99	0.04	0.00	0.12	0.00	
4	**1**	ftr	137	-0.58	-0.60	-0.47	-0.22	-0.23	-0.48	64	-0.05	-0.04	-0.09	-0.13	0.02	-0.44
		wgt		0.25	0.40	-0.20	0.50	0.30			0.28	0.00	0.02	0.65	0.38	
	2	ftr	90	0.67	0.70	0.60	0.15	0.20	0.46	69	-0.16	-0.11	-0.11	0.02	-0.04	-0.65
		wgt		0.51	0.15	-0.16	0.32	0.12			1.32	0.00	0.00	0.00	0.00	
	3	ftr	8	2.83	2.84	2.19	2.98	2.72	2.05	53	0.20	0.18	0.24	0.11	0.11	-0.13
		wgt		-23.16	6.16	10.83	-2.12	0.57			0.02	0.28	0.00	0.00	0.09	
	4	ftr	1	-3.93	-3.41	-7.43	-7.48	-8.11	-1.74	50	0.07	0.02	0.01	0.02	-0.09	0.73
		wgt		0.00	0.00	0.00	0.00	0.00			0.04	0.36	0.00	0.41	0.34	
5	**1**	ftr	79	0.17	0.18	0.26	0.10	0.22	0.21	46	0.08	0.15	-0.04	-0.14	-0.09	0.16
		wgt		0.51	0.46	-0.05	0.23	0.18			0.52	0.19	0.00	0.00	0.00	
	2	ftr	114	-0.67	-0.69	-0.54	-0.28	-0.29	-0.58	48	-0.20	-0.20	-0.13	-0.05	-0.04	-0.69
		wgt		0.00	0.00	0.00	0.00	0.00			0.11	0.00	0.00	0.78	2.04	
	3	ftr	34	1.30	1.32	0.91	0.23	0.05	0.87	33	-0.22	-0.27	-0.19	0.04	-0.17	-0.57
		wgt		0.10	0.56	-0.06	0.26	0.33			0.73	1.47	0.00	0.30	0.99	
	4	ftr	1	-3.93	-3.41	-7.43	-7.48	-8.11	-1.74	53	0.20	0.21	0.17	0.02	0.08	-0.15
		wgt		-23.16	6.16	10.83	-2.12	0.57			0.30	0.00	0.27	0.32	0.31	
	5	ftr	8	2.83	2.84	2.19	2.98	2.72	2.05	56	0.04	0.01	0.10	0.12	0.13	0.02
		wgt		0.51	-0.03	-0.36	0.48	0.04			0.00	0.00	0.00	0.00	0.00	

Table 1.: Features and weights for the KM+LR and CRIO CLR approaches. *K* is the number of clusters used in the model, id is the index of the model, and type refers to whether cluster feature averages (ftr) or linear regression weights (wgt) are shown. # indicates the number of tracts assigned to each cluster. We remark that the feature averages displayed for KM+LR clusters are solely the result of the K-Means algorithm.

K	Method	MSE	MAE	CRC
1	LR	0.59	0.57	
2	KM+LR	0.57	0.57	1.0
	CRIO	0.32	0.39	
3	KM+LR	0.49	0.53	0.98
	CRIO	0.27	0.34	
4	KM+LR	0.47	0.51	0.92
	CRIO	0.19	0.25	
5	KM+LR	0.45	0.50	0.90
	CRIO	0.14	0.22	

Table 2.: Comparison of MSE and MAE for Linear Regression, KM+LR and CRIO CLR methods for k clusters. CRIO performs best on both MSE and MAE for all k. For KM+LR, the Cross-Run Consistency (CRC) is also shown, where the maximum CRC of 1.0 indicates complete reproducibility. CRC is not shown for the other two methods as they are deterministic and so produce the same results each time (technically equivalent to a CRC of 1.0).

Tract	prof	col	flabf	multi	own	incpc	Model	Prediction
148.01	-0.94	-0.91	-1.49	-0.45	-0.25	0.24	2	-0.61
148.02	-0.69	-0.79	-0.52	-0.20	-0.25	-1.68	2	-0.51

Table 3.: Census features and KM+LR predictions for 148.01 and 148.02. Note that highly similar census features results in KM+LR predicting very similar income change for both tracts, while in reality the change in the two tracts is quite different. It would in fact be nearly impossible for a well-trained KM+LR model to correctly predict the income change for both of these tracts at the same time.

Method	Tracts	prof	col	flabf	multi	own	Bias
KM+LR	Both	0.10	0.56	0.05	-0.06	0.26	0.33
CRIO	148.01	0.00	0.00	0.00	0.00	0.00	0.01
CRIO	148.02	0.73	1.47	0.00	0.30	0.99	0.55

Table 4.: Prediction weights for 148.01 and 148.02 in KM+LR and CRIO. CRIO assigns different weights to the features for the two tracts, which reflects underlying differences in the way that these two tracts are changing. KM+LR is incapable of this type of observation.

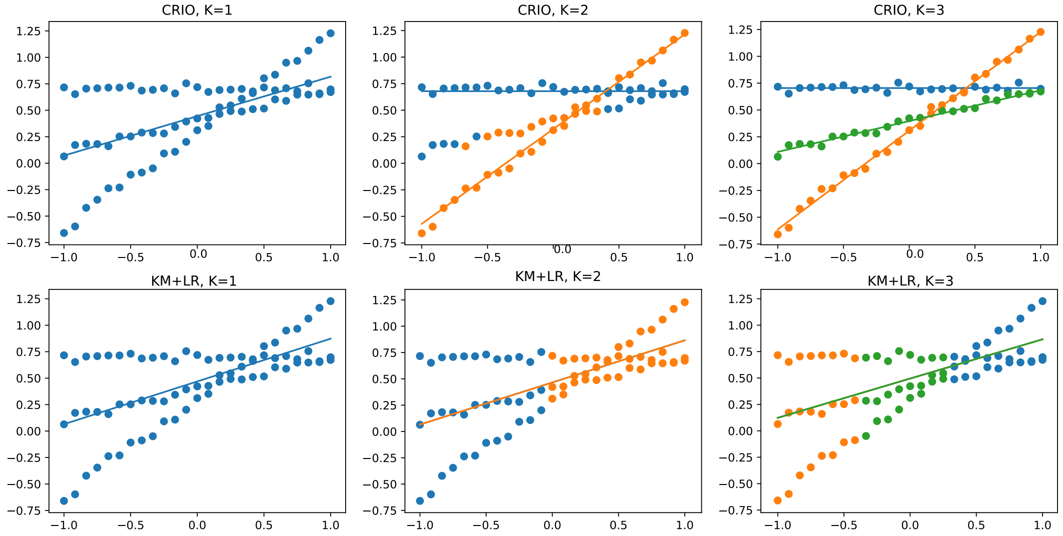


Figure 1.: Illustrative comparison of CRIO and KM+LR on a simple synthetic dataset. Raw data (individual points) corresponding to three distinct regression lines (dependent y-axis as a linear function of the independent x-axis with a small amount of Gaussian noise added) and ground truth cluster labels and regressions (respectively, the points and lines shown in orange, green, and blue) are recovered by CRIO for K=3 clusters (top right). CRIO results are shown in the top row and KM+LR results are shown in the bottom row. Columns provide results for each method for a different number of clusters $K \in \{1, 2, 3\}$. While CRIO for K=3 (top right) is able to accurately assign each point to the correct cluster and recover the ground truth regression line per cluster, KM+LR fails to identify the ground truth clusters and regressions for K=3 (bottom right) since it can only use x-axis features for clustering.

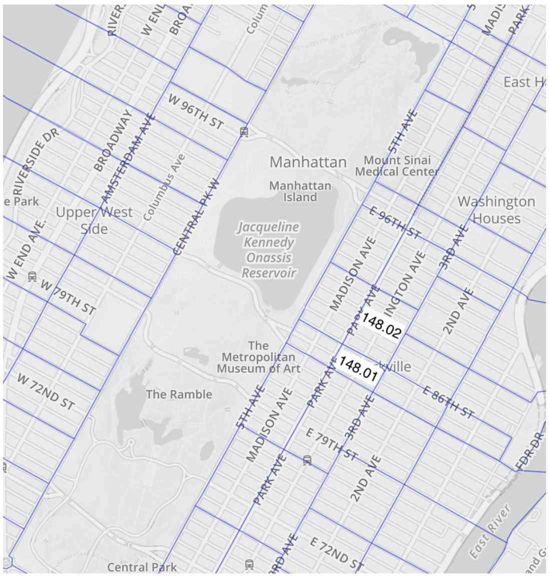

148.02
148.01
Manhattan
Upper West Side
Jacqueline Kennedy Onassis Reservoir
The Metropolitan Museum of Art
The Ramble
Central Park
Washington Houses
Mount Sinai Medical Center

Figure 2.: The location of census tracts 148.01 and 148.02 in New York City.

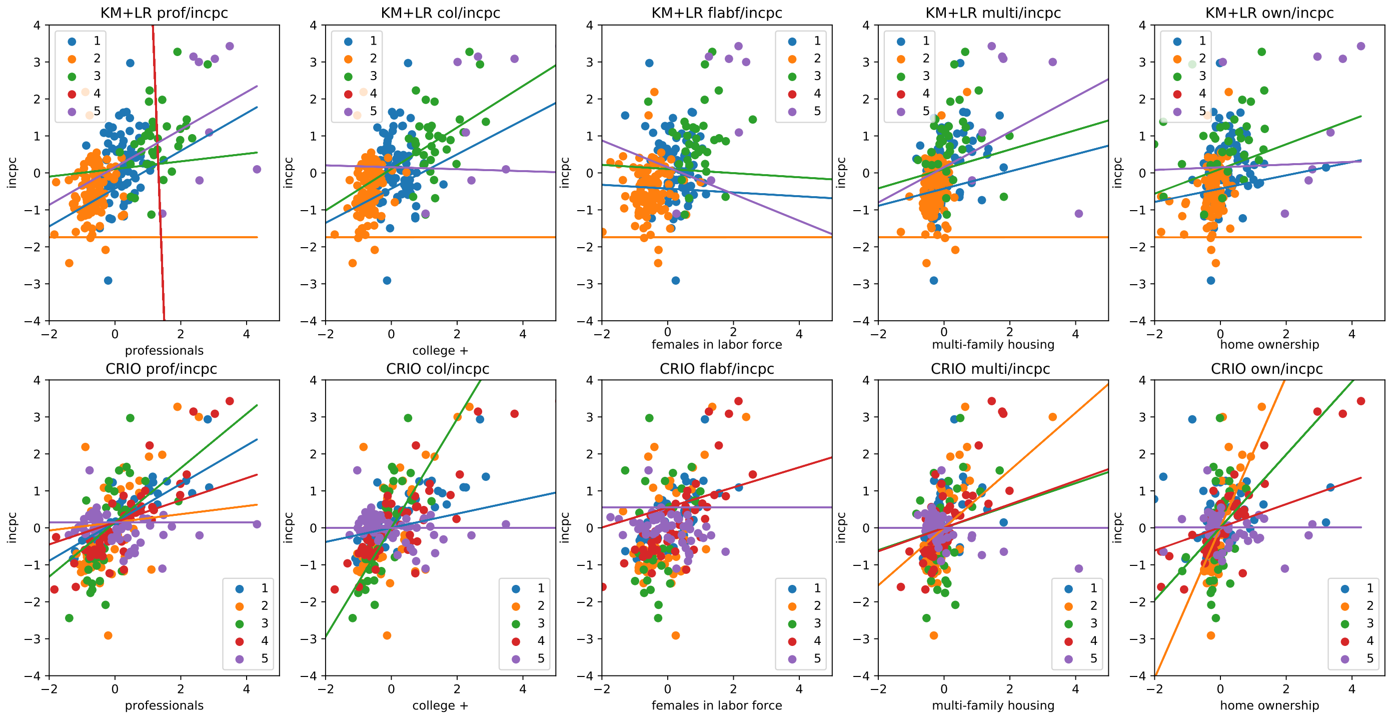
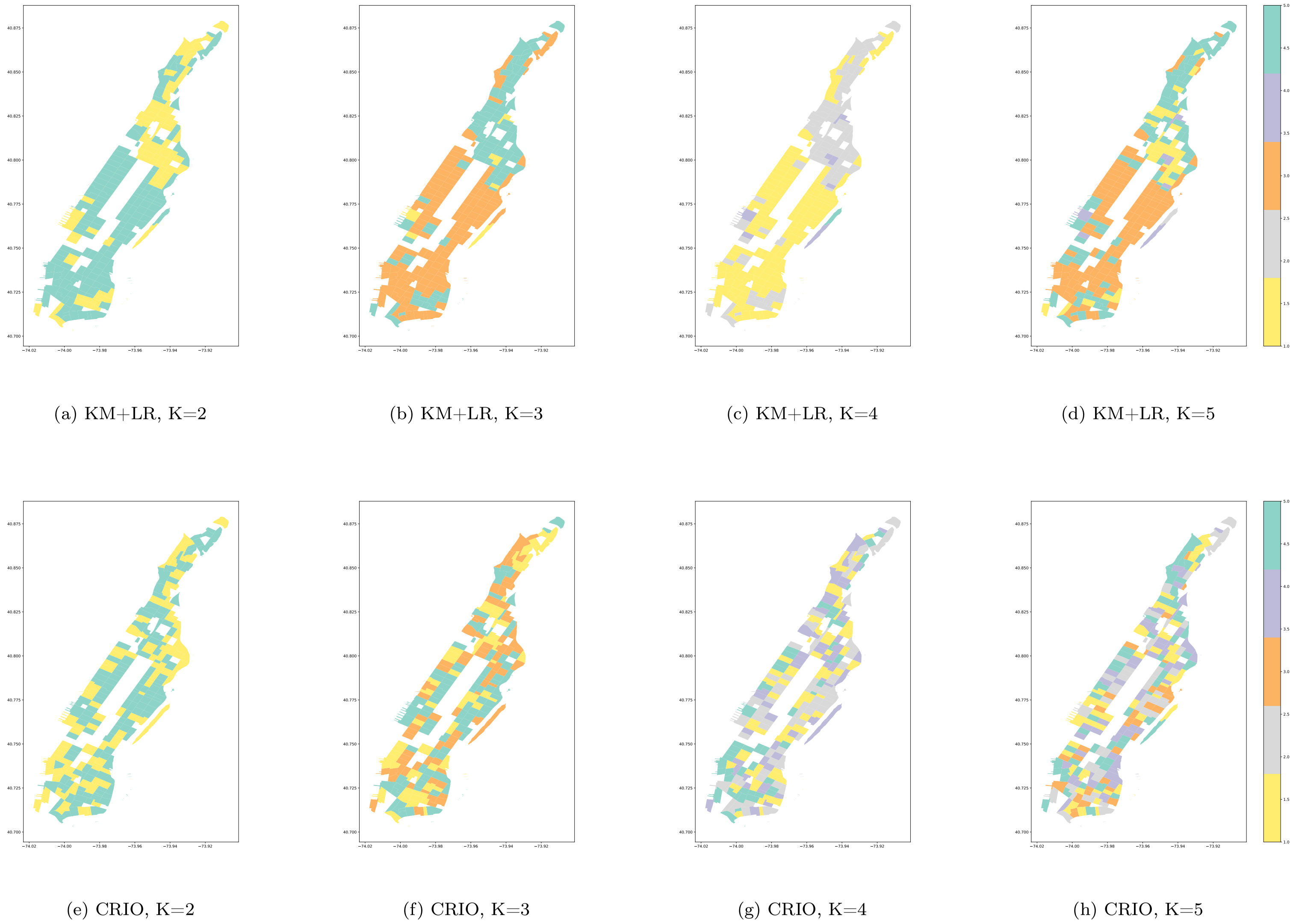


Figure 3.: Comparison of KM+LR and CRIO CLR clustering by feature. The first row contains the plots for KM+LR, while the second row shows CRIO. Each colour corresponds to one cluster in the model. Each column is one of the five features. The models maintain continuity along each row — that is to say, the green line for Model 3 represents a single Model 3 across the entire row, which considers all the features together with the label variable in six dimensional space. Note that in certain plots, most notably the CRIO plot of females in labor force, multiple regressions are horizontal and overlap. As a result, they appear obscured.



(a) KM+LR, K=2

(b) KM+LR, K=3

(c) KM+LR, K=4

(d) KM+LR, K=5

(e) CRIO, K=2

(f) CRIO, K=3

(g) CRIO, K=4

(h) CRIO, K=5

Figure 4.: Comparison of KM+LR and CRIO CLR clustering methods in terms of clusters (colours) for each number of clusters k. Note that the clusters displayed for KM+LR are solely the result of the K-Means algorithm.

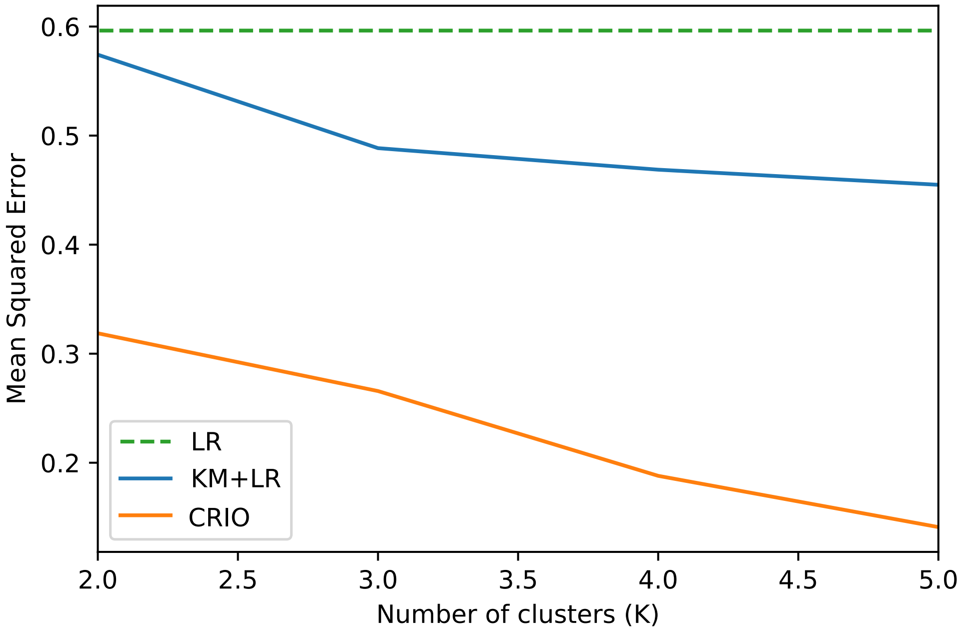


Figure 5.: Mean Squared Error (MSE) for all three compared methods. Note that a dashed line is shown for Linear Regression (LR) as this method does not use clustering and is not dependent on K. Clearly, both CLR models (KM+LR and CRIO) reduce prediction error over LR. CRIO substantially outperforms KM+LR at each K as well as LR in terms of MSE regression error.