Prediction with high dimensional regression via hierarchically structured Gaussian mixtures and latent variables

Chun-Chen Tu, Florence Forbes, Benjamin Lemasson, Naisyin Wang

Prediction with High-Dimensional Regression via Hierarchically Structured Gaussian Mixtures and Latent Variables †

Chun-Chen Tu
University of Michigan, Ann Arbor, USA
Florence Forbes
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP‡, LJK, 38000 Grenoble, France
Benjamin Lemasson
Univ. Grenoble Alpes, Inserm, U1216, GIN, F-38000, Grenoble, France.
and Naisyin Wang
University of Michigan, Ann Arbor, USA

Summary.
In this paper we propose a structured mixture model, named Hierarchical Gaussian Locally Linear Mapping (HGLLiM), to predict low-dimensional responses based on high-dimensional covariates when the associations between the responses and the covariates are non-linear. For tractability, HGLLiM adopts inverse regression to handle the high dimension and locally-linear mappings to capture potentially non-linear relations. Data with similar associations are grouped together to form a cluster. A mixture is composed of several clusters following a hierarchical structure. This structure enables shared covariance matrices and latent factors across smaller clusters to limit the number of parameters to estimate. Moreover, HGLLiM adopts a robust estimation procedure for model stability. We use three real-world datasets to demonstrate different features of HGLLiM. With the face dataset, HGLLiM shows the ability of modeling non-linear relationship through mixtures. With the orange juice dataset, we show the prediction performance of HGLLiM is robust to the presence of outliers. Moreover, we demonstrate that HGLLiM is capable of handling large-scale complex data using the data acquired from a magnetic resonance vascular fingerprinting (MRvF) study. These examples illustrate the wide applicability of HGLLiM on handling different aspects of a complex data structure in prediction.

Keywords: Expectation-Maximization, High dimension, Mixture of regressions, Magnetic resonance vascular fingerprinting, Robustness.

1. Introduction

Building a regression model for the purpose of prediction is widely used in all disciplines. A large number of applications consists of learning the association between responses and predictors and focusing on predicting responses for the newly observed samples. In

† *Address for correspondence*: Chun-Chen Tu, Department of Statistics, University of Michigan, 311 West Hall 1085 South University, Ann Arbor, MI 48109-1107, USA
E-mail: timtu@umich.edu
‡ Institute of Engineering Univ. Grenoble Alpes

this work, we go beyond simple linear models and focus on predicting low-dimensional responses using high-dimensional covariates when the associations between responses and covariates are non-linear. Non-linear mappings can be handled through different techniques such as kernel methods (Elisseeff and Weston, 2002; Wu, 2008) or local linearity (De Veaux, 1989; Frühwirth-Schnatter, 2006; Goldfeld and Quandt, 1973). In general, the conventional methods adopting local linearity assume assignment independence and are considered as not adequate for regression (Hennig, 2000). Alternatively, one can adopt a mixture-modeling strategy and let the membership indicator of a mixture component depend on the values of the covariates. The Gaussian Locally Linear Mapping (GLLiM Deleforge et al., 2015) follows this principle.

GLLiM groups data with similar regression associations together. Within the same cluster, the association can be considered as locally linear, which can then be resolved under the classical linear regression setting. Besides adopting the framework of model-based clustering (Banfield and Raftery, 1993; Fraley and Raftery, 2002), GLLiM also takes on a factor-model based parameterization (Baek et al., 2010; Bouveyron et al., 2007; McLachlan and Peel, 2000; Xie et al., 2010) to accommodate the high-dimensional and potentially dependent covariates (see Equation (20) in Deleforge et al. (2015)). In particular, the high-dimensional variables were postulated as a sum of two components: the one that is linearly related with the low-dimensional responses, and the other which can be projected onto a factor model and then be presented as augmented latent variables. This data augmentation approach is applicable in many application scenarios, whenever certain variables are only partially observed or corrupted with irrelevant information. The augmentation step, with added latent variables, acts as a factor analyzer modeling for the noise covariance matrix in the regression model. GLLiM is based on a joint modeling of both the responses and covariates, observed or latent. This joint modeling framework allows for the use of an inverse regression strategy to handle high-dimensional data.

However, when the covariate dimension is much higher than the response dimension, GLLiM may result in erroneous clusters at the low dimension, leading to potentially inaccurate predictions. Specifically, when the clustering is conducted at a high joint dimension, the distance at low dimension between two members of the same cluster could remain large. As a result, a mixture component might contain several sub-clusters and/or outliers, violating the Gaussian assumption of the model. This results in a model misspecification effect that can seriously impact prediction performance. We demonstrate this phenomenon with a numerical example in Section 2. A natural way to lessen this effect is to increase the number of components in the mixture making each linear mapping even more local. But this practice also increases the number of parameters to be estimated. Estimating parameters in a parsimonious manner is required to avoid over-parameterization. In addition, increasing the number of clusters could isolate some data points or lead to singular covariance matrices. Hence, a robust estimation procedure for model stability is also necessary.

In this work, we propose a parsimonious approach combined with a robust estimation procedure which we refer to as Hierarchical Gaussian Locally Linear Mapping (HGLLiM) to construct a stable model for predicting low-dimensional responses. Parsimonious models generally refer to some model instances where the number of parameters is reduced

compared to the full parameterization. The goal of parsimonious models is to find a good compromise between model flexibility and parsimony. HGLLiM inherits the advantages from GLLiM on handling high-dimensional, non-linear regression with partially-latent variables. In terms of the number of parameters, the largest costs usually come from high-dimensional covariance matrices. On this front, HGLLiM follows a two-layer hierarchical clustering structure in which we reduce the number of covariance parameters in the model. HGLLiM also includes a pruning algorithm for eliminating outliers as well as determining an appropriate number of clusters. The number of clusters and training outliers determined by HGLLiM can be further used by GLLiM for improving prediction performance.

With the goal of investigating the flexibility in accommodating data structure and the ability to protect from influences of outliers, we evaluate our method on three datasets with different characteristics. The face dataset contains face images, the associated angles of faces and the source of the light. There is no obvious cluster structure at first glance nor the existence of real outliers. We use this dataset to evaluate the ability of HGLLiM on modeling regression through local linear approximations. The orange juice dataset contains continuous spectrum predictors and some abnormal observations. Using this dataset, we aim to show that HGLLiM is robust and can effectively identify outlying observations. We use these two moderate size datasets to demonstrate how the method works on data with different features and the insensitivity of tuning parameter selection on a wide range of selection domain. Finally, in our last data analysis, we study a problem where researchers are interested in predicting the microvascular properties using the so-called magnetic resonance vascular fingerprinting (MRvF). Hereafter we refer to this dataset as the fingerprint data. We use this dataset to demonstrate the power of HGLLiM on modeling complex associations over a large number of observations. Results show that HGLLiM can provide comparable prediction performance on one case and much smaller prediction errors on the other, compared to the dictionary matching method in Lemasson et al. (2016) with only 25% of the computational time.

This paper is organized as follows. In Section 2 we explain and illustrate the issue encountered with unstructured GLLiM in high-dimensional settings. In Section 3, we present the structured alternative that we propose. The experiment results on three real datasets are provided in Section 4. Finally, Section 5 concludes with a discussion and potential future directions.

2. Unstructured Gaussian Locally Linear Mapping (GLLiM)

To predict low-dimensional data $Y \in \mathbb{R}^L$ using high-dimensional data $X \in \mathbb{R}^D$, $D >> L$, GLLiM elegantly copes with several challenging issues simultaneously. The high-to-low mapping difficulty is circumvented by inverse regression. And then the desired high-to-low relationship can be easily converted from the low-to-high associations under a proper model construction. Non-linearity is approximated by locally linear associations (Chapter 6, Scott, 2015). The parameter estimation is carried out by an Expectation-Maximization algorithm, which nicely incorporates estimation of latent variables.

The original GLLiM model groups data into K clusters. For cluster k, the data follow the distributions below:

$$p(X = x|Y = y, Z = k; \theta) = \mathcal{N}(x; A_k y + b_k, \Sigma_k). \quad (1)$$
$$p(Y = y|Z = k; \theta) = \mathcal{N}(y; c_k, \Gamma_k), \quad (2)$$
$$p(Z = k; \theta) = \pi_k,$$

where the latent variable Z represents the cluster assignment, and $\theta = \{c_k, \Gamma_k, \pi_k, A_k, b_k, \Sigma_k\}_{k=1}^K$ is a vector denoting the model parameters. For the k-th cluster, the center and the covariance matrix for the mixture at low dimension are c_k and Γ_k. The parameter A_k and b_k are the transformation matrix and intercept, mapping data from low dimension to high dimension with Σ_k capturing the reconstruction errors.

A distinct feature of GLLiM is that Y needs not be a completely observable vector. In fact, it is set to have $Y^\top = (T^\top, W^\top)$, where T contains the observable variables, which one intends to predict, and W, being latent, absorbs the remaining dependency and variation in the high-dimensional X. The inclusion of W strengthens the chance to reach validity of Equation (1).

The issue with GLLiM is that the high dimensionality of the data may have an unexpected impact on the posterior probability of the cluster assignment. When the dimensions of X and Y are satisfying $D >> L$, this comes from the following observation: in the E-step the posterior probabilities r_{nk} (Equation (27) in Deleforge et al. (2015)) is computed as:

$$r_{nk} = p(Z_n = k|x_n, y_n; \theta) = \frac{\pi_k p(x_n, y_n|Z_n = k; \theta)}{\sum_{j=1}^K \pi_j p(x_n, y_n|Z_n = j; \theta)} \quad (3)$$

for all n and all k, where $p(x_n, y_n|Z_n = k; \theta)$ can be computed as $p(x_n|y_n, Z_n = k; \theta)p(y_n|Z_n = k; \theta)$. The first term is a density with much higher dimension (D) so that its value could dominate the product. In addition, y_n can be decomposed into two parts: the observed variable t_n and the latent variable w_n. The component w_n reflects the remaining variation in x_n that cannot be explained by x_n's association with t_n. When w_n accounts for explaining most of the variation in x_n, the clustering outcome would highly depend on w_n and weaken the ability of detecting sub-clusters in T.

Therefore, although GLLiM assumes that within each cluster $p(Y = y|Z = k; \theta)$ is Gaussian and centered on c_k, in practice, the model groups data according to the high dimension term and could fail in imposing the Gaussian shape on the t_n's. In other words, the model rather chooses the clusters to satisfy the assumption in Equation (1). And this induces a clustering of the (x_n, y_n)'s into groups within which the same affine transformation holds. Thus, a cluster could contain several sub-clusters and/or outliers since the Gaussian assumption on T, as part of the Y, in Equation (2) is sometimes neglected. This may cause a serious impact on the estimation of c_k and Γ_k and consequently on the prediction step.

We illustrate this issue by presenting an example using a face dataset (Tenenbaum et al., 2000). This dataset contains 698 images (of size 64×64 and being further condensed to 32×32). The pose of each image is defined by three variables in T: *Light*, *Pan* and *Tilt*, as shown in Figure 1 (a). We adopt GLLiM to predict these T's (low-dimensional) using the image (high-dimensional). The superiority of GLLiM in

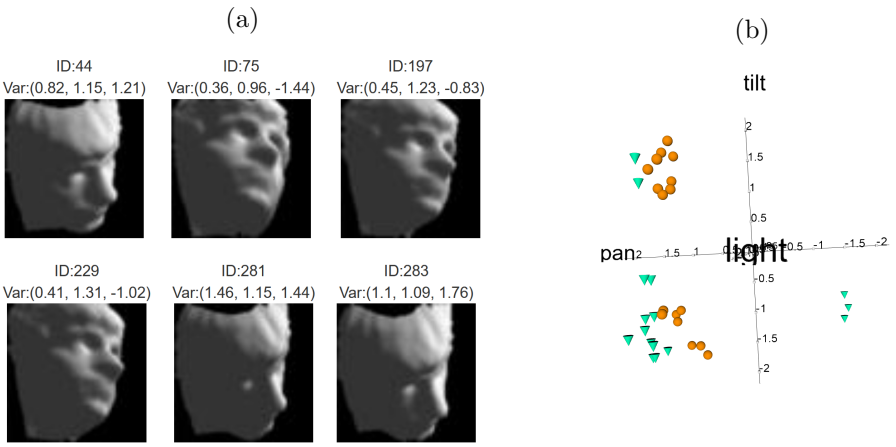

Fig. 1: The clustering results of the face dataset obtained from GLLiM: (a) six face images from Cluster 7; (b) scatter plot of T for points within Cluster 7 and 13 clustered by GLLiM. Data points are from Cluster 7 (circle) and Cluster 13 (triangle). The three variables are (*Light*, *Pan*, *Tilt*).

Table 1: The comparison of original and post cluster-division Squared Error (SE). The improved ratio is calculated as the ratio of difference of SE from pre- to post cluster-division over the original SE. The value is positive if the procedure reduces the SE, and negative, otherwise.

Image ID	GLLiM cluster	Original SE	Post-Division SE	Improved ratio
56	7	0.306	0.043	86.03%
223	7	0.016	0.180	-1039.83%
293	7	0.060	0.023	61.27%
302	7	0.087	0.003	96.99%
114	13	0.114	0.118	-2.93%
204	13	0.307	0.073	76.19%
294	13	3.119	0.120	96.15%

prediction, comparing to multiple existing approaches, for this data set was numerically illustrated in Deleforge et al. (2015).

Figure 1(b) shows the scatter plot of T within Clusters 7 and 13, grouped by GLLiM. By visual inspection, both clusters seem to consist of two or more sub-clusters. In GLLiM, samples within the same cluster are assumed to follow Gaussian distributions. This sub-cluster structure, however, violates the assumption and potentially increases the prediction errors. We demonstrate the difference of prediction performance before and after accounting for the sub-cluster structure in Table 1. We use prediction Squared Error (SE) for testing data pre- and post cluster-division. We observe that the prediction errors are mostly reduced if we account for the sub-cluster structure.

Dividing samples at low dimension is an effective and straightforward solution for this issue. However, we could obtain small sub-clusters after division and then increase the prediction variance. In Table 1, Images 114 and 223 were assigned to small and/or

tight local clusters and the prediction of T for these two images become worse after cluster-division. Conceptually, small clusters could damage the prediction performances for several reasons: the small number of observations in such a cluster leads to estimates with large variation; a small cluster with a small covariance matrix determinant (volume) could lead to instability of the whole likelihood-based algorithm, and a small/tight cluster could consider a close-by testing sample unfit and force it to be predicted by another less suitable cluster with a larger within-cluster covariance. The last consideration is not relevant to model building but plays an important role in prediction precision.

This observation motivates us to look into enhancing prediction stability by eliminating small clusters and outliers in the training samples. We further explore both issues in Section 4.

3. Hierarchical Gaussian Locally Linear Mapping (HGLLiM)

In our proposed work, we intend to strike a balance between model flexibility and variation reduction in the estimated predictive model, with the goal of predicting the low-dimensional observable variables, T, using the high-dimensional X. This predictive model needs not to be the true model but should be effective in prediction. To present the fundamental concepts with clarity, we will first describe the model structure when $Y = T$, with minimum required notations. The scenario of Y containing W is easily extended in Section 3.2.

3.1. Model description

The joint probability, $p(X = x, Y = y; \theta)$, of high-dimensional predictor X and low-dimensional response Y can be written as:

$$\sum_{k=1}^{K}\sum_{l=1}^{M} p(X = x|Y = y, Z = k, U = l; \theta)p(Y = y|Z = k, U = l; \theta)p(Z = k, U = l; \theta),$$

where θ denotes the vector of parameters; Z and U are, respectively, latent global and local cluster assignment. The locally linear relationship between X and Y is given by the mixture model below:

$$X = \sum_{k=1}^{K}\sum_{l=1}^{M} \mathbb{I}(Z = k, U = l)(A_{kl}Y + b_{kl} + E_k),$$

where $\mathbb{I}$ is the indicator function, $A_{kl} \in \mathbb{R}^{D\times L}$ and $b_{kl} \in \mathbb{R}^D$ map Y onto X, and $E_k \in \mathbb{R}^{D\times D}$ is the error term that absorbs the remaining uncertainty. Recall that D and L are dimensions of X and Y, respectively, and $D >> L$. Here, we let the local cluster size $M(k) \equiv M$ for notation simplicity only. We assume, within the k-th global cluster, all local clusters share the same error structure which follows a zero-mean Gaussian distribution with covariance matrix Σ_k. That is, we have,

$$p(X = x|Y = y, Z = k, U = l; \theta) = \mathcal{N}(x; A_{kl}y + b_{kl}, \Sigma_k).$$

As in (2), the model is completed by assuming that the low-dimensional Y, given the clustering-assignment indicators $(Z,\ U) = (k,\ \ell)$, follows a Gaussian distribution with mean c_{kl} and variance Γ_{kl}, and by defining a prior for clustering assignment: $p(Z = k, U = l \mid \theta) = \rho_{kl}$, where $c_{kl} \in \mathbb{R}^L$, $\Gamma_{kl} \in \mathbb{R}^{L\times L}$ and $\sum_{k=1}^{K}\sum_{l=1}^{M}\rho_{kl} = 1$. The vector of parameters in the inverse regression model, θ, is given by

$$\theta = \{c_{kl}, \Gamma_{kl}, \rho_{kl}, A_{kl}, b_{kl}, \Sigma_k\}_{k=1,l=1}^{K,M}. \tag{4}$$

The remaining task is to use the inverse conditional density to construct the exact formulations of the conditional density of Y given X in the forward regression model and the resulting expression of $E[Y|X = x]$. Equivalent to how we define θ in the model of X given Y, we let θ^* denote the parameter vector in the forward regression model of Y given X. We then derive the closed-form expression of θ^* as a function of θ, as given in Appendix A. The prediction of Y given X can then be done by taking the expectation over the forward conditional density, $E[Y|X = x]$, given in (1). The use of the closed-form expressions provided in Appendix A makes it computationally efficient in conducting prediction.

3.2. HGLLiM model with partially-latent responses

Recall that the low-dimensional data $Y \in \mathbb{R}^L$ contains a latent component W. Namely, $Y^\top = (T^\top, W^\top)$, where $T \in \mathbb{R}^{L_t}$ is observed and $W \in \mathbb{R}^{L_w}$ is latent and thus $L = L_t + L_w$. It is assumed that T and W are independent given Z, and so are W and U. According to the decomposition of Y, the corresponding mean (c_{kl}), variance (Γ_{kl}) and regression parameters (A_{kl}) of Y, at the local-cluster level, are given as:

$$c_{kl} = \begin{bmatrix} c_{kl}^t \\ c_k^w \end{bmatrix},\ \Gamma_{kl} = \begin{bmatrix} \Gamma_{kl}^t & 0 \\ 0 & \Gamma_k^w \end{bmatrix},\ \text{ and } A_{kl} = [A_{kl}^t\ A_k^w]. \tag{5}$$

That is, when $Z = k,\ U = l$, at the local-cluster level, $T \sim \mathcal{N}(c_{kl}^t, \Gamma_{kl}^t)$; when $Z = k$, at the global-cluster level, $W \sim \mathcal{N}(c_k^w, \Gamma_k^w)$. It follows that, locally, the association function between the high-dimensional Y and low-dimensional X can be written as:

$$X = \sum_{k=1}^{K}\mathbb{I}(Z = k)\left\{\sum_{l=1}^{M}\mathbb{I}(U = l)(A_{kl}^t T + b_{kl}) + A_k^w W + E_k\right\}. \tag{6}$$

Finally, the parameter vector θ in the inverse regression model is rewritten as: $\theta = \{\rho_{kl}, c_{kl}^t, \Gamma_{kl}^t, A_{kl}^t, b_{kl}, c_k^w, \Gamma_k^w, A_k^w, \Sigma_k\}_{k=1,l=1}^{K,M}$.

It follows that (6) rewrites equivalently as

$$X = \sum_{k=1}^{K}\mathbb{I}(Z = k)\left\{\sum_{l=1}^{M}\mathbb{I}(U = l)(A_{kl}^t T + b_{kl}) + A_k^w c_k^w + E_k'\right\}, \tag{7}$$

where the error vector E_k' is modeled by a zero-centered Gaussian variable with a $D \times D$ covariance matrix given by

$$\Sigma_k' = \Sigma_k + A_k^w \Gamma_k^w A_k^{w\top}. \tag{8}$$

Considering realizations of variables T and X, the addition of the latent W naturally leads to a covariance structure, namely (8), where $A_k^w \Gamma_k^w A_k^{w\top}$ is at most of rank L_w. When Σ_k is diagonal, this structure is that of factor analysis with at most L_w factors, and represents a flexible compromise between a full covariance with $O(D^2)$ parameters on one side, and a diagonal covariance with $O(D)$ parameters on the other.

Using the same number of total clusters and considering the fact that Σ_k and A_k^w are only estimated at the global-cluster level, we note that the total number of parameters needed to model the covariances, Σ_k, and the latent transformation coefficients, A_k^w, using HGLLiM is $1/M$ of that required by using GLLiM. In addition, the key emphasis of HGLLiM is to conduct prediction. As shown in (5), (7), and (8) at the local vs. global-cluster levels, we now separate the estimation of the mean association functions, which play a key role in prediction, from that of high-dimensional covariance matrices, so that the means can be obtained even more locally. Together with the current dependence structures, being stably estimated at the global-cluster level using more data points per cluster, the HGLLiM provides a strong prediction tool built on a structure facilitating sensible approximations to the true underlying distribution of low-dimensional T and high-dimensional X.

3.3. Robust estimation procedure

The HGLLiM model contains three sets of latent variables: $Z_{1:N} = \{Z_N\}_{n=1}^N$, $U_{1:N} = \{U_n\}_{n=1}^N$ and $W_{1:N} = \{W_n\}_{n=1}^N$. The first two sets of variables indicate the global and the local cluster assignments and the last one is the latent covariates. The model parameters, θ, as defined in Equation (4) can be estimated using the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). However, even with the inversion step, the prediction procedure still involves a high-dimensional predictor and elevated variation in estimated parameters, induced by small clusters or abnormal observations, that could lead to deteriorated prediction quality. The stability can be achieved by constraining the sizes of the clusters (control both covariance volume and prediction variance) and trimming outliers. We design a robust estimation procedure to refine the standard EM algorithm with the purpose of enhancing model stability, which consequently leads to improved prediction performance.

Define the posterior probability of observation n being assigned to cluster (k, l) as

$$r_{nkl} = p(Z_n = k, U_n = l | t_n, x_n; \theta), \tag{9}$$

and let $\sum_{n=1}^N r_{nkl}$ represent the cluster size for cluster (k, l). Each data point in a cluster whose cluster size is smaller than a pre-determined *minSize* is reassigned to other clusters. The point is kept when the prediction squared error is less than a pre-determined *dropThreshold*; otherwise, it would be excluded from the current EM iteration when updating the estimated parameters. With the data points within a cluster playing a dominating role in estimating within-cluster parameters, the cluster-size plays the role of the sample size in estimation: when the sample size is too small, the prediction quality deteriorates even if the assumed structure is true. An improved prediction performance might be achieved by assigning such a data point within a small cluster to another cluster that shares similar structures. If no such a cluster can be identified, then the data point is excluded from the construction of the prediction model.

An EM algorithm for HGLLiM directly constructed according to models given in Sections 3.1 and 3.2 is described in Appendix A.1. The algorithm iterates between E-steps to update latent W, Z, U and M-steps that update θ. Here, we describe the robust estimation procedure, tailored to ensure stability and outlier trimming. The algorithm is described as follows:

(a) The algorithm is initialized by adopting the parameters θ, mean and covariance, $\tilde{\mu}_{nk}^w$, $\tilde{S}_k^w$, of latent W of the k-th cluster, and cluster assignment r_{nkl} obtained from the EM algorithm described in Appendix A.1.

(b) The estimating procedure iterates through the following substeps until the algorithm converges:

(i) Trimming step: In order to remove outliers, we scan through all local clusters and remove all samples whose in-sample prediction squared errors are greater than a pre-determined *dropThreshold*. The prediction squared error for the n-th sample is calculated as:

$$E_n^2 = ||t_n^{pred} - t_n||_2^2, \tag{10}$$

where t_n is the true value and t_n^{pred} is the prediction from Equation (1). Note that the low dimension data $\{t_n\}_{n=1}^N$ are standardized before training so that each dimension would be equally weighted. The samples with in-sample prediction squared error larger than *dropThreshold* are considered as outliers and are temporarily removed by assigning r_{n^*kl} to be 0 at that iteration of M-step, where n^* indicates the training sample whose $E_{n^*}^2 > dropThreshold$.

(ii) Maximization step with a cluster size constraint: The estimation of θ is the same as the Maximization step described in Appendix A.1 but with an additional cluster size constraint. Before estimating parameters for each local cluster (k, l), we first check the associated cluster size. If the cluster size is smaller than the given $minSize$, we force the training data originally assigned to this cluster to either be assigned to other clusters during the E-step in updating cluster-assignment Z, and U, or, if no appropriate cluster could be found, be trimmed during the next Trimming Step.

(iii) Update step for the latent variables: Estimation of $\tilde{\mu}_{nk}^w$, $\tilde{S}_k^w$ and r_{nkl} are done using E-W and E-Z, U step described in Appendix A.1.

All outcomes in Section 4 are obtained using the algorithm presented in this section. The procedure to select all tuning parameters, aiming at obtaining better prediction performances, is given in Appendix A.2.

4. Numerical Investigation

We analyze three datasets and use the outcomes to illustrate the usage of the proposed method. Key features of each dataset, thus the type of data they represent, are reported in the corresponding subsections. Throughout, we use squared error (Equation (10)) to evaluate the prediction performance for each data point. We also calculate the prediction

mean squared error (MSE) among all testing samples with $MSE = \sum_{n=1}^{N_{test}} E_n^2/N_{test}$, where N_{test} is the total number of testing samples.

We calculate and compare the MSE or the quantiles of squared errors over several methods:

(a) HGLLiM: This is the proposed method. The user-defined parameters K and L_w are set to values using the method described in Appendix A.2. The number of local clusters M is set to 5 to reflect the possible sub-cluster structure. In each global cluster, the number of local clusters varies and depends on the structure of the dataset. Some of the local clusters would be dissolved so the number of local clusters could be less than M. The initial cluster assignment is done by dividing the GLLiM clustering outcomes at the low dimension using the R package *mclust* (R Core Team, 2018; Scrucca et al., 2017). As stated before, the robust version of the EM algorithm is used throughout the experiments. We set $minSize = 5$ for all of the analyses and post-analysis checks at the neighborhood of 5 suggest this is an appropriate choice. The prediction MSE using different values of $dropThreshold$ would be calculated and compared.

(b) GLLiM: The original GLLiM algorithm. GLLiM is compared to other methods under the same settings of K and L_w. The initial cluster assignment is done by applying a Gaussian Mixture Model to a dataset that combines the low-dimensional T and high-dimensional X together.

(c) GLLiM-structure: This method adopts the number of clusters learned structurally by HGLLiM. In addition, outliers identified by HGLLiM are removed from the training dataset. We adopt the same tuning parameters as GLLiM and the initial conditions are obtained from the outcomes of HGLLiM. The key difference between GLLiM-structure and HGLLiM is that GLLiM-structure uses local estimated covariance, which may be more appropriate for a large dataset with more local dependence features. Its effectiveness also suggests an additional usage of HGLLiM, in terms of structure learning and identification of outliers.

4.1. The face dataset

The face dataset, consisting of 698 samples, was analyzed in the original GLLiM paper (Deleforge et al., 2015). For this dataset, we are interested in predicting the pose parameters ($L_t = 3$) using the image information. The size of each image is condensed to 32×32, and thus $D = 1024$. In addition, T is standardized so that all three dimensions are equally weighted. The histograms of the three T variables bear no clustering structure. Consequently, the mixture modeling serves the purpose of local linear approximation and the inverse regression is utilized to circumvent the difficulties encountered in high-dimensional regression.

In each run of cross-validation investigation, we follow the procedure in Deleforge et al. (2015) and select 100 testing samples and keep the remaining 598 as training samples. We repeated this procedure 20 times to establish 2000 testing samples. According to the approach described in Appendix A.2, we run cross-validation on K from 10 to 25, L_w from 1 to 15. The cross-validation results in Figure 2(a) suggest that $K = 20$,

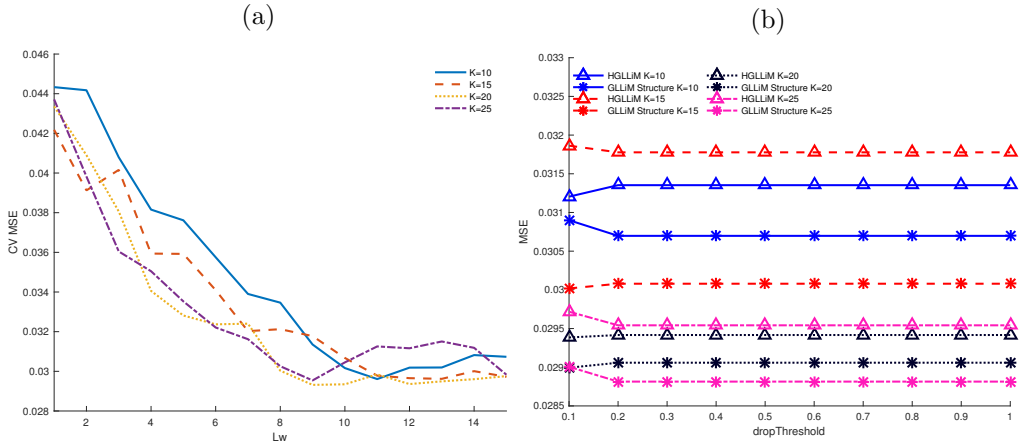

Fig. 2: Results for different user-defined parameters of the face dataset. (a) The HGLLiM cross-validation results for different K and L_w. (b) The prediction MSE of different K and different methods against different $dropThreshold$.

$L_w = 9$. It is noted that the prediction errors decrease with increasing values of L_w. This phenomenon suggests that the high-dimensional X are dependent and that accounting for such dependency via the latent W leads to improvement in prediction. It is also observed that the change of prediction error is relatively small when L_w exceeds a certain value. Therefore, we fix the number of latent factors and compare the prediction performance under $K = 10$, $K = 15$ and $K = 25$.

Figure 2(b) shows prediction outcomes under different values of $dropThreshold$. We observe that for different methods and different K, the prediction MSE's are not sensitive to the values of $dropThreshold$. Thus, we compare the prediction MSE of HGLLiM, GLLiM-structure when $dropThreshold = 0.5$ to GLLiM in Table 2. The prediction MSE for GLLiM decreases as K increases, which indicates that more clusters could be helpful to capture the non-linear relationship between X and T. For HGLLiM, we observe that the prediction MSE is not sensitive to the choice of K. In addition, the numbers of clusters are similar under different choices of K. This indicates that HGLLiM could adjust itself to reach the number of clusters suitable to its setting. As for GLLiM-structure, the prediction MSE's are slightly smaller than those of HGLLiM. This is because GLLiM-structure estimates all parameters using local clusters, local covariances, and the prediction would be less biased when the local structures sufficiently differ. In the face dataset, there is no obvious cluster structure and, as a result, clustering only serves the purpose of improving local approximation. Thus, the prediction MSE for GLLiM-structure would be smaller. However, the differences of prediction MSE's between HGLLiM and GLLiM-structure are small, which implies that the settings learned from HGLLiM are appropriate, even though HGLLiM imposes a global-cluster structure when there is none. Overall, the prediction performance for HGLLiM is similar when $K = 20$ and $K = 25$. As for GLLiM-structure, the MSE is smaller when $K = 25$ but the difference is negligible.

We further investigate the phenomenon described in Section 2. Specifically, as the

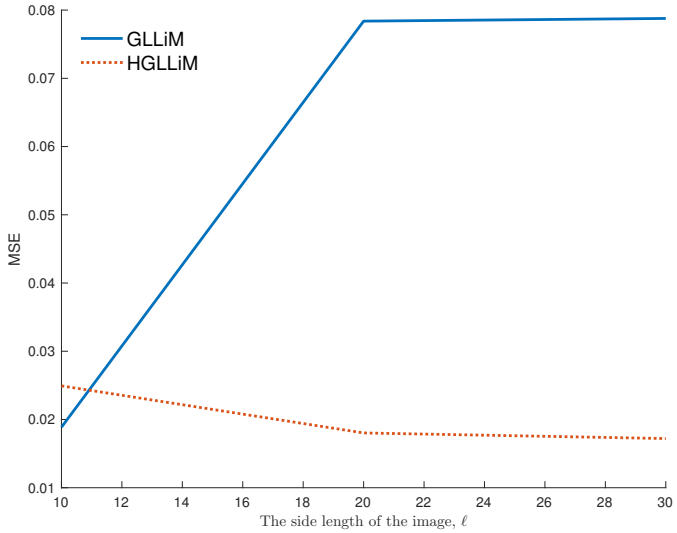

Fig. 3: The prediction MSE of the face dataset under different dimension of X. Each image in the face dataset consists of $\ell \times \ell$ pixels, where ℓ, the side length of the image, is the square root of the dimension of X.

Table 2: The prediction MSE and the average cluster number of the face dataset when $dropThreshold = 0.5$.

	K=10		K=15		K=20		K=25	
	MSE	#Cluster	MSE	#Cluster	MSE	#Cluster	MSE	#Cluster
GLLiM	0.0711	10.00	0.0441	15.00	0.0369	20.00	0.0321	25.00
HGLLiM	0.0314	43.90	0.0318	51.35	0.0294	53.75	0.0295	53.45
GLLiM-structure	0.0307	43.90	0.0301	51.35	0.0291	53.75	0.0288	53.45

dimension of X becomes higher, not only the number of covariance parameters increases, but there is also a higher chance the clusters formed by GLLiM could contain sub-clusters and/or outliers, which could deteriorate the prediction quality. We use Cluster 7 as our reference to create two clusters. There are two sub-clusters within Cluster 7. We first identify the center of each sub-cluster using the low-dimensional T and find 30 nearest samples to each center. We randomly select 25 data points from each sub-cluster as the training data and use the rest of the data points as the testing samples. Thus, there will be 50 training samples and 10 testing samples. The procedure is repeated 20 times and the results are aggregated together to evaluate the model performance.

To investigate the prediction performance under the different dimensions of X, we resize the face image to $\ell \times \ell$ pixels, where we denote ℓ the side length of the image so that the dimension of X is $D = \ell \times \ell$. For GLLiM, we set the number of clusters, K, to be 2 and the dimension of the latent variables, L_w, to be 9. For HGLLiM, we have one global cluster and two local clusters, that is, $K = 1, M = 2$. As suggested in Figure 2(a), we let $L_w = 9$ since this setting results in smaller cross-validation MSE. We disable the robust estimation step, which is equivalent to setting $minSize = 0$, $dropThreshold = \infty$, as described in Section 3.3; also see Appendix A.2. Figure 3 shows

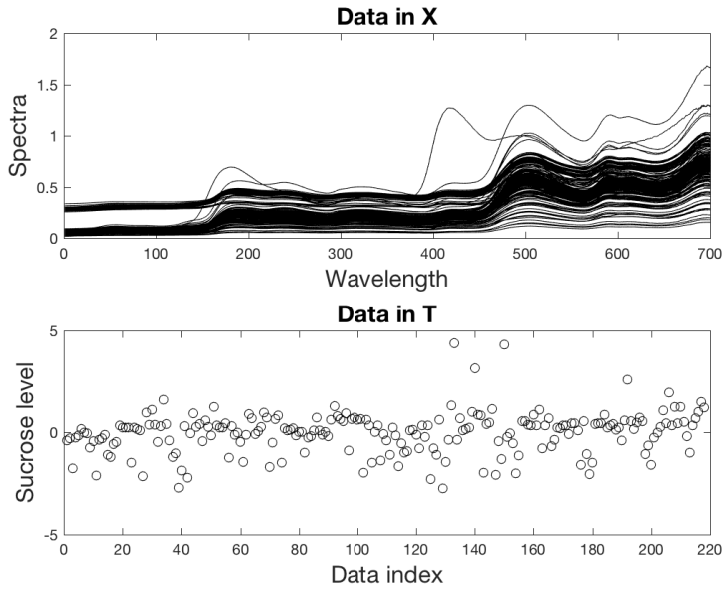

Fig. 4: The orange juice dataset. The upper panel shows the high-dimensional data (X) and the lower one shows the low-dimensional data (T).

the result of prediction MSE under different dimensions of X. When the dimension of X is low, GLLiM can outperform HGLLiM. However, as the dimension of X increases, we observe that the prediction error of GLLiM increases, suffering from the potentially less suitable cluster-assignments. On the other hand, HGLLiM maintains appropriate clustering results and thus the prediction performance remains similar for all image sizes, if not slightly improved with the increasing dimension of X and the enhanced information in the images with higher resolution.

4.2. The orange juice dataset

The orange juice dataset contains the spectra measured on different kinds of orange juice ($N = 218$). The goal is to use the spectra to predict the level of sucrose ($L_t = 1$). We follow the step described in Perthame et al. (2018) and decompose the spectra on a spline basis with ($D = 134$) to make $D \approx N$. This dataset is known for the presence of outliers; the realization of X and T is given in Figure 4.

We setup the following prediction evaluation procedure. In each run, we randomly select 20 testing samples from the main population (excluding outliers). The remaining 198 samples (including outliers, unless otherwise specified) are used for training. These outliers were identified through Leave One Out Cross Validation (LOOCV) using GLLiM, with $K = 10$ and $L_w = 2$. Although the set of outliers may differ for different selection of K, L_w, the severe outliers are always selected and they are included here. We identify 11 points, which are the observations with top 5% of the prediction E^2's (above 4.8) among all data points, as outliers. Removing outliers from testing data pre-

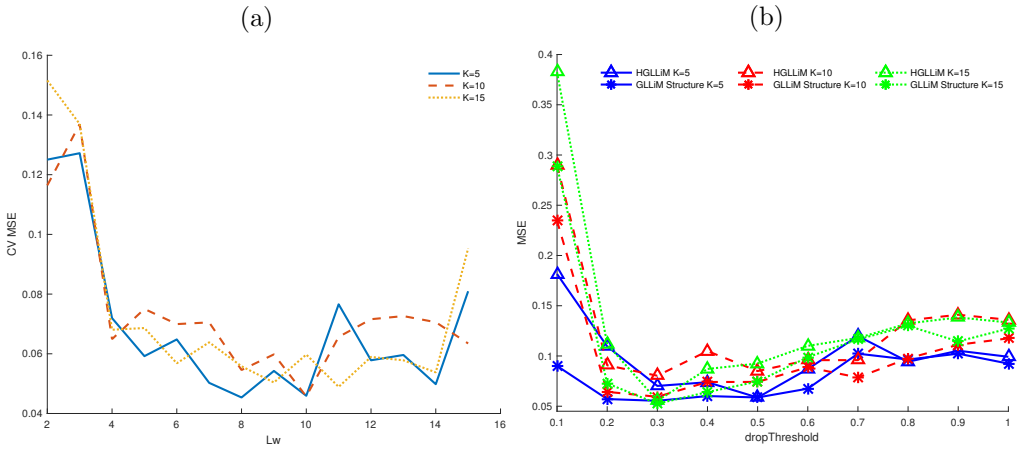

Fig. 5: Results for the user-defined parameters of the orange juice dataset. (a) The HGLLiM cross-validation results for different K and L_w. (b) The prediction MSE of different K and different methods against different $dropThreshold$.

vents the summarized outcomes being overwhelmed by prediction results of few points, which potentially makes the differences among methods less obvious. All methods were evaluated using the same settings.

Figure 5(a) shows the cross-validation results, which suggests the use of $K = 5$, $L_w = 8$. For comparison purpose, we also provide MSE results for $K = 10$ and $K = 15$. The rest of the setting is the same as the experiment setting used for the face dataset.

To evaluate the influence of outliers on GLLiM, we conduct an analysis in which we use the same cluster number as in GLLiM-structure but without removing training outliers. This method is referred to as GLLiM-outlier. In addition, we consider SLLiM in Perthame et al. (2018) provided by the R package *xLLiM* (Perthame et al., 2017). SLLiM is a counterpart of GLLiM that accommodates abnormal samples using Student's t-distributions. Precisely, the high-dimensional X is modeled by a mixture of K generalized multivariate Student's t-distributions, using the structure given in Section 5.5 (p.94) of Kotz and Nadarajah (2004). We also compare SLLiM performances by using the same cluster number learned structurally by HGLLiM. We refer to the resulting procedure as "SLLiM-structure". We use the default settings in *xLLiM* for the remaining SLLiM configurations.

Figure 5(b) shows the prediction MSE for different $dropThreshold$'s. The prediction MSE's vary, mainly reflecting the high variation in this data set, partially due to outliers. For a small $dropThreshold$, the number of identified training outliers is more than expected. This reduces the training data size and makes the prediction unreliable. As $dropThreshold$ reaches a reasonable value, the prediction performance becomes better. However, more and more abnormal training samples are included in the training dataset as $dropThreshold$ keeps increasing. These outlying data enlarge the model variance and downgrade the prediction performance. Table 3 shows the results for $dropThreshold = 0.5$. We observe that for $K = 5$, the cluster number is not sufficiently large for GLLiM to capture the non-linear trend in the data, which results in a relatively large prediction

Table 3: The prediction MSE and the average cluster number of the orange juice dataset when $dropThreshold = 0.5$.

	K=5		K=10		K=15	
	MSE	#Cluster	MSE	#Cluster	MSE	#Cluster
GLLiM	0.1259	5.00	0.1210	10.00	0.0918	15.00
HGLLiM	0.0587	9.95	0.0681	11.85	0.0692	12.80
GLLiM-structure	0.0621	9.95	0.0742	11.85	0.0746	12.80
GLLiM-outlier	0.0976	9.95	0.1171	11.85	0.1044	12.80
SLLiM	0.1039	5.00	0.0788	10.00	0.0706	15.00
SLLiM-structure	0.0907	9.95	0.0747	11.85	0.0721	12.80

MSE. HGLLiM, on the other hand, adjusts the cluster number automatically and the prediction errors are smaller. In addition, HGLLiM removes training outliers that would deteriorate the model performance. This explains why even though the cluster number is as large as $K = 15$ (larger than the average size of 12.8 used in GLLiM-structure), GLLiM still suffers from large prediction errors. We further observe the benefit of removing outliers by comparing GLLiM-structure and GLLiM-outlier. The prediction errors for GLLiM-structure are smaller than those produced by GLLiM-outlier and the only difference between GLLiM-structure and GLLiM-outlier is whether training outliers, identified by HGLLiM, are removed. There are 11 outliers in the training dataset. HGLLiM could effectively identify and remove all of them. In addition to these outliers, some potential outlying samples that could result in unstable model are trimmed as well. Overall, about 6% to 10% of the training samples would be removed by HGLLiM.

SLLiM and SLLiM-structure use t-distributions to accommodate the existence of outliers. They are expected to perform better than their Gaussian counterparts (GLLiM and GLLiM-outlier). When K, the cluster number, is small, there would be more samples in each cluster and thus the cluster size, $\sum_{n=1}^{N} r_{nkl}$ for cluster (k, l), would be large. On the contrary, when K is large, samples would be divided into more clusters, which decreases the cluster size. It is observed that when K is small, accommodating outliers with t-distributions is not as effective as removing them by comparing SLLiM-structure and GLLiM-structure. When the number of clusters becomes larger, outliers could be assigned to a cluster with less influence on the prediction and thus we can obtain similar prediction performance from SLLiM-structure and GLLiM-structure. However, removing outliers would reduce the cluster size and result in unstable prediction performance. To provide reliable model performance, HGLLiM controls the cluster size via the tuning parameter $minSize$. In addition, HGLLiM estimates the covariance matrices under global cluster level and this estimation is more reliable compared to GLLiM-structure, which estimates covariance matrices locally. SLLiM does not remove any samples and thus the performance would be better than GLLiM-structure when the cluster number, K, is large. Although removing outliers is more effective, accommodating outliers may still be an alternative to combat outliers when the cluster size is the concern.

4.3. A magnetic resonance vascular fingerprint dataset

It is of great interest to the scientific community to be able to efficiently assess microvascular properties, such as, blood volume fraction, vessel diameter, and blood oxygenation, in brain so that the ability in diagnosis and management of brain diseases can be improved. Recently, a new approach called magnetic resonance vascular fingerprinting (MRvF) was proposed as an alternative to overcome the limitations of analytical methods in measuring microvascular properties. The approach was built on a system in which the signal acquired in each voxel, also called "fingerprint", was compared to a dictionary obtained from numerical simulations. Finding the closest match to a fingerprint record in the dictionary allows a direct link between the parameters of the simulations and the microvascular variables (also referred to as parameters in these studies) at the image voxel (Lemasson et al., 2016; Ma et al., 2013).

A synthetic MRv fingerprint (hereafter referred to as fingerprint) dataset composed of $1,383,648$ observations was created to serve as a "search/match" library. Each observation in the library consists of a fingerprint measurement and associated parameters: mean vessel radius (Radius), Blood Volume Fraction (BVf) and a measurement of blood oxygenation (DeltaChi). One goal is to predict these parameters ($L_t = 3$) using the fingerprint measurement ($D = 32$). In addition to these three parameters, other parameters (variables) that have influence over the fingerprint measurements include Apparent Diffusion Coefficient (ADC), vessel direction (Dir) and vessel geometry (Geo).

In Lemasson et al. (2016), numerical performances of a dictionary matching method were presented. For comparison purpose, we implement the dictionary matching method adopted in Lemasson et al. (2016). The coefficient of determination (r^2) is used to measure the similarity between a testing sample and the training samples (dictionary). The coefficient of determination, r^2, between a testing sample y^{test} and a training sample y^{train} is calculated as:

$$r^2 = 1 - \frac{\sum_{d=1}^{D}(y_d^{test} - y_d^{train})^2}{\sum_{d=1}^{D}(y_d^{test} - \bar{y}^{test})^2}, \tag{11}$$

where $\bar{y}^{test} = \frac{1}{D}\sum_{d=1}^{D} y_d^{test}$. The matched fingerprint is the training fingerprint with the largest r^2 and we predict the parameters of the testing data using the matched fingerprint.

Computation time is a critical issue when analyzing large datasets. To speed up the computation, we could take advantage of the hierarchical structure of HGLLiM by subsetting the dataset into smaller groups and applying HGLLiM on the resulting groups in parallel. Our current study consists of two components. Through cross-validation, we first evaluate the feasibility and effectiveness of the parallel computation algorithms and utilize it to compare the performance of different methods on the synthetic dataset. We then apply these methods to a fingerprint dataset collected at an animal study; in which, besides predicting the variable BVf (a main goal of Lemasson et al. (2016)), we focus on predicting another variable ADC, a more challenging scenario which has not been reported before.

We divide the synthetic library into 20 groups and apply the parallelization techniques to accelerate the model building process (see Appendix B and Appendix C for more details). When conducting the analysis at the animal study, we add a small amount

Table 4: The mean predicted values within ROIs of different vascular parameters from different categories.

		Dictionary matching	GLLiM	HGLLiM	GLLiM-structure
9L	Radius	21.85	20.14	22.12	21.52
	BVf	14.49	14.33	14.71	14.25
	DeltaChi	0.98	0.93	1.03	0.94
C6	Radius	13.59	16.01	13.67	13.81
	BVf	4.17	4.01	4.25	4.52
	DeltaChi	0.77	0.76	0.79	0.74
F98	Radius	11.56	13.14	11.13	11.23
	BVf	3.86	3.96	4.01	3.97
	DeltaChi	0.65	0.66	0.62	0.61
Stroke	Radius	14.69	13.51	14.31	14.41
	BVf	4.22	4.49	4.13	4.25
	DeltaChi	0.60	0.63	0.62	0.63
Healthy	Radius	8.16	7.96	8.54	8.34
	BVf	3.58	3.51	3.63	3.56
	DeltaChi	0.76	0.72	0.74	0.80

of the *in vivo* data in the training dataset. We noted that fingerprint samples from the real world are noisier than their synthetic counterparts and thus this practice, as a calibration step, enables the training model to readily accommodate the real fingerprint samples in prediction. The ratio of the synthetic samples to the real image samples is 4 to 1. The cluster number and latent factor number are selected using the method described in Appendix A.2 and are set to $K = 1240$ and $L_w = 9$, respectively. We evaluate and compare the performance of different methods on the synthetic dataset through cross-validation. The cross-validation results in predicting Radius, BVf and DeltaChi demonstrate that the model-based methods (GLLiM/HGLLiM/GLLiM-structure) can achieve comparative prediction performance (Appendix D). Next, we apply these methods to a fingerprint data set collected at an animal study.

This animal study dataset contains samples from 115 rats categorized into 5 different groups: healthy, 3 kinds of tumors (9L, C6 and F98) and stroke. For each rat, there are 5 brain slices of 128×128 voxels and each voxel contains 32 dimension fingerprint information. For each slice, the lesion (unhealthy) and the striatum (healthy) areas are labeled and they form the region of interest (ROI). Figure 6 shows the predicted BVf image using different methods. As indicated in Lemasson et al. (2016), the values of true BVf is not available at the voxel level, and instead, they are measured over the whole ROI's. Nevertheless, the comparison between the true values and those obtained by the dictionary matching method, at the ROI level, indicates that the method has successfully provided close-to-truth match; see Lemasson et al. (2016). Table 4 shows the mean prediction results within the ROI's obtained by different methods. The three additional methods considered here, besides the dictionary-matching method used in Lemasson et al. (2016), are GLLiM, HGLLiM and GLLiM-structure. All four methods provide similar results in predicting BVf.

There are 1,385,509 samples in the real image dataset. For the dictionary match-

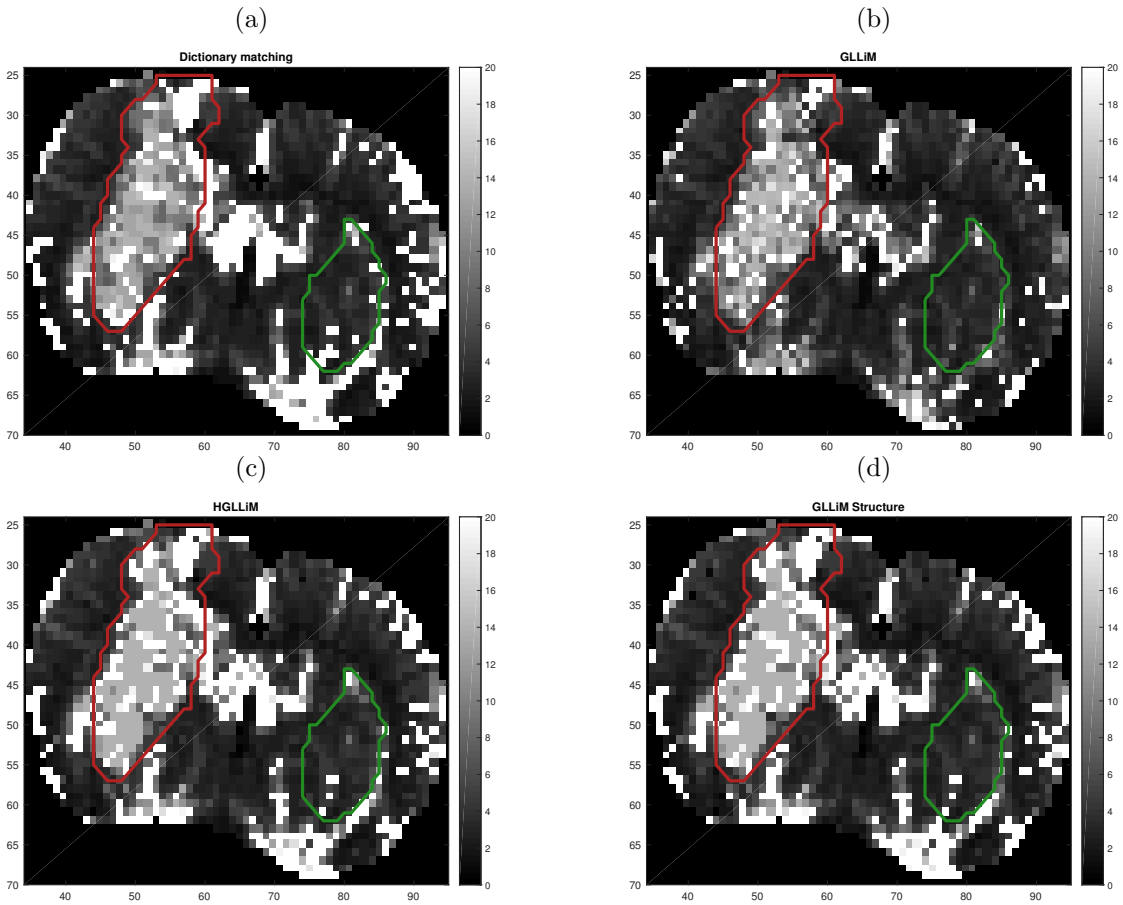

Fig. 6: The predicted BVf images of one animal from the 9L group using either (a) dictionary matching, (b) GLLiM, (c) HGLLiM or (d) GLLiM-structure. In each plot, the ROI on the left marks the lesion region and the ROI on the right is from the healthy striatum.

ing method, using a parallel for-loop (*parfor*) and a pre-processing technique (Lemasson et al. (2016)), it took about 2.4 hours (precisely 8639.53 seconds) to match the whole animal image samples to the training dataset ($N^{train} = 1,383,648$). A direct computation without *parfor* and pre-processing took 429507.79 seconds and reach the same outcomes. For the model-based method, utilizing the grouped structure and the parallel computing technique, it takes 1058.32/2133.51/1922.37 seconds for GLLiM/HGLLiM/GLLiM-structure to process the animal image dataset. Thus, the prediction procedure of GLLiM/HGLLiM/GLLiM-structure is much more efficient than the dictionary matching method.

The parameter ADC was not thoroughly investigated in Lemasson et al. (2016). The main reason is that the predicted ADC values, obtained using the dictionary matching approach, were not comparable to the ones measured *in vivo* by MRI. With the *in vivo* ADC values available at the voxel-level, being able to understand how the synthetic and

Table 5: The 50%, 90% 99% quantiles of squared errors of predicting ADC for different methods under different image categories.

	Dictionary matching			GLLiM			HGLLiM			GLLiM-structure		
	50%	90%	99%	50%	90%	99%	50%	90%	99%	50%	90%	99%
9L	1.1180	3.9803	10.6829	0.2392	0.5684	14.5668	0.1132	0.7613	11.8721	0.1018	0.7154	10.9574
C6	1.1208	4.4719	14.4888	0.3043	2.6091	26.7575	0.3252	1.9840	22.5427	0.3138	1.7764	20.0213
F98	1.0994	4.2373	14.4888	0.3802	3.4129	55.4479	0.2951	2.3672	35.3199	0.2801	2.4129	50.8133
Stroke	1.1663	5.8045	14.8888	0.4779	4.5668	66.1164	0.3218	3.0975	55.7821	0.3192	3.1424	53.9315
Health	1.0931	3.8086	7.7912	0.2131	1.2510	14.5668	0.1527	1.1087	11.9597	0.1054	1.1145	13.2165

real measurements differ for a given parameter is scientifically important to developing new instruments and to future knowledge advancements. Here, we study ADC and use it to evaluate the prediction performances of different methods. Figure 7 shows the true ADC image and the images of the differences between the true and predicted ADC values. The differences are shown in the ratio against the signal levels for each ROI. Most of the predictions made by dictionary matching are deviated from the true values. On the other hand, HGLLiM and GLLiM-structure provide better ADC images. There are some voxels with extreme differences that all methods cannot predict well. When no suitable training information could be provided by the synthetic fingerprint data, the prediction quality on these voxels tends to be dreadful regardless which method is used.

Table 5 shows the 50%, 90% and 99% quantiles of the ADC squared errors. The outcomes reported under the 50th and 90th percentiles give the indication of "average" and "almost-all" prediction performances for each method. The 99th percentile values allow the comparisons of worse-case scenarios. We still obtain some predictions with large errors using GLLiM/HGLLiM/GLLiM-structure. However, for majority of the data, the squared errors are smaller than those obtained by the dictionary matching method. We figure out that here is no suitable cluster to conducting prediction for these data. For GLLiM/HGLLiM/GLLiM-structure, if a suitable cluster for conducting prediction does not exist, the cluster with the closest Mahalanobis distance is applied for prediction. However, the largest membership posterior probability r_{nkl} among all k, l in Equation (9) would be smaller than the majority of the data. This information could be utilized to identify unreliable prediction results. The worst case of dictionary matching seems to produce smaller prediction error when being compared to other methods. Nevertheless, this is due to the nature of the difference among approaches. The dictionary matching method always predict using values obtained from a member in the dictionary, so that its prediction error cannot go beyond what would be provided by the possible values in the dictionary. This phenomenon does not apply to model-based methods. When prediction is conducted on the data outside of the range of the training dataset, the prediction error could become considerably large, as shown by the outcome of 99 percentiles of prediction squared errors. As a result, even though dictionary matching seems to outperform other model-based methods at these extreme cases, it does not necessary indicates that dictionary method is practically useful for these cases, with the outcomes being so much worse than predicting the rest of the dataset. Our model-based approaches, on the other hand, do have the advantage of identifying these troublesome cases for further considerations.

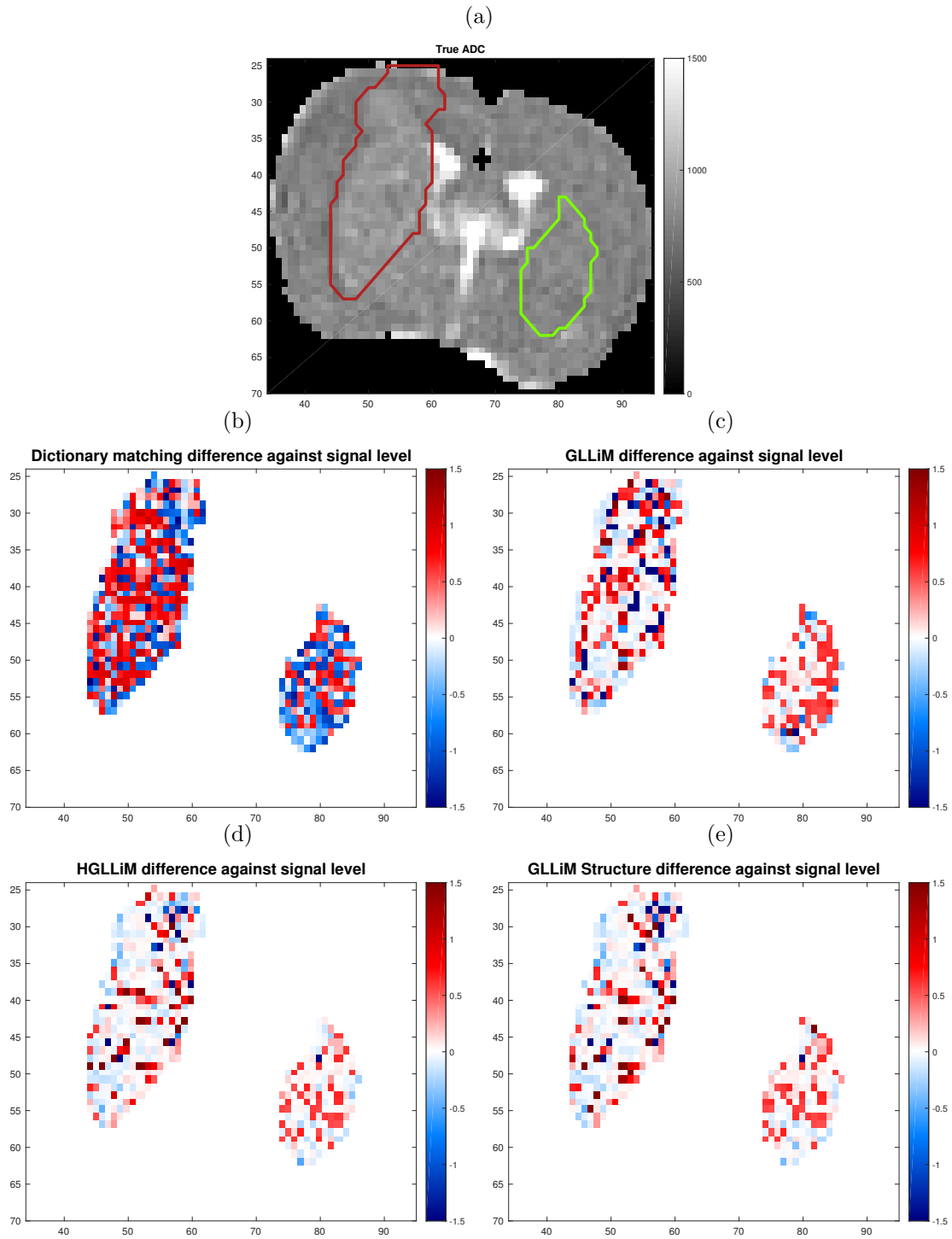

Fig. 7: The true ADC image and the differences between the true values and the predicted values against the signal levels of one animal from the 9L group. Differences are normalized by the average true ADC values in each ROI. (a) The true ADC image. Difference maps between true values and predicted values against the signal levels using either (b) dictionary matching method, (c) GLLiM, (d) HGLLiM or (e) GLLiM-structure.

5. Discussion and conclusion

We propose HGLLiM as a parsimonious and structured version of GLLiM. HGLLiM adopts a two level hierarchical structure of clusters. The assumed structure enables us to assess the parameters in the mean association functions more locally without suffering from the clustering outcomes being dominated by the dependence structures in the high-dimensional predictors. Under the same construction, we also estimate the reduced number of covariance parameters with more data points. In addition, we implement a robust version of HGLLiM to enhance model stability and reduce prediction variation. HGLLiM further leads to a post-learning version of GLLiM, called GLLiM-structure. By using local means and local variances, while with unfitted points removed, GLLiM-structure tends to reach improved empirical performances.

The motivation behind HGLLiM and GLLiM-structure is to obtain precise predictions by constructing stable training models. Eliminating the existence of small clusters and removing outliers assist to achieving this goal. The fact that HGLLiM only focuses on preserving primary structures learned from the training dataset may reduce the quality of its predictions of rare data points, which are insufficiently presented therein. Nevertheless, by utilizing the largest membership posterior probability r_{nkl} among all clusters (k, l) and by recognizing when this maximum be much smaller than those obtained from the majority of the data, we can identify such testing samples with unreliable prediction results.

In the analysis of the fingerprint dataset, we experimented predicting these testing samples using the average predicted values based on nearest neighbor matching (results not shown). At the cost of slightly elevated computation time, such replacement does have the predicted values being within the range of training measurements. The prediction quality for these difficult-to-predict cases is similar to that of the dictionary matching method. However, it is important to note that such quality, as for the dictionary matching method, remains unsatisfactory. This outcome is not a surprise due to the fact that these data points are either not present or poorly represented in the training samples.

Albeit this drawback that the resulted training model obtained by HGLLiM may or may not reflect the exact true model that generates all the data, it nevertheless captures the critical structure and establishes a model that can be stably estimated using the data available. The size and complexity of this model would be determined by the data. Finally, the learning procedure could be accelerated by dividing data into groups and adopting parallelization computation technique. We illustrate that this practice is readily accommodated by HGLLiM's hierarchical model structure.

References

Baek, J., McLachlan, G. J. and Flack, L. K. (2010) Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data. *IEEE transactions on pattern analysis and machine intelligence*, **32**, 1298–1309.

Banfield, J. D. and Raftery, A. E. (1993) Model-Based Gaussian and Non-Gaussian Clustering. *Biometrics*, **49**, 803–821.

Bouveyron, C., Girard, S. and Schmid, C. (2007) High-dimensional data clustering. *Computational Statistics & Data Analysis*, **52**, 502–519.

De Veaux, R. D. (1989) Mixtures of linear regressions. *Computational Statistics & Data Analysis*, **8**, 227–245.

Deleforge, A., Forbes, F. and Horaud, R. (2015) High-dimensional regression with Gaussian mixtures and partially-latent response variables. *Statistics and Computing*, **25**, 893–911.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, 1–38.

Elisseeff, A. and Weston, J. (2002) A kernel method for multi-labelled classification. In *Advances in neural information processing systems*, 681–687.

Fraley, C. and Raftery, A. E. (2002) Model-Based Clustering, Discriminant Analysis, and Density Estimation. *Journal of the American Statistical Association*, **97**, 611–631.

Frühwirth-Schnatter, S. (2006) *Finite Mixture and Markov Switching Models*. Springer Science & Business Media.

Goldfeld, S. M. and Quandt, R. E. (1973) A Markov model for switching regressions. *Journal of Econometrics*, **1**, 3–15.

Hennig, C. (2000) Identifiablity of Models for Clusterwise Linear Regression. *Journal of Classification*, **17**, 273–296.

Kotz, S. and Nadarajah, S. (2004) *Multivariate t-distributions and their applications*. Cambridge University Press.

Lemasson, B., Pannetier, N., Coquery, N., Boisserand, L. S. B., Collomb, N., Schuff, N., Moseley, M., Zaharchuk, G., Barbier, E. L. and Christen, T. (2016) MR Vascular Fingerprinting in Stroke and Brain Tumors Models. *Scientific Reports*, **6**, 37071.

Ma, D., Gulani, V., Seiberlich, N., Liu, K., Sunshine, J. L., Duerk, J. L. and Griswold, M. A. (2013) Magnetic Resonance Fingerprinting. *Nature*, **495**, 187–192.

McLachlan, G. and Peel, D. (2000) Mixtures of factor analyzers. In *In Proceedings of the Seventeenth International Conference on Machine Learning*.

Perthame, E., Forbes, F. and Deleforge, A. (2018) Inverse regression approach to robust nonlinear high-to-low dimensional mapping. *Journal of Multivariate Analysis*, **163**, 1–14.

Perthame, E., Forbes, F., Deleforge, A., Devijver, E. and Gallopin, M. (2017) *xLLiM: High Dimensional Locally-Linear Mapping*. R package version 2.1.

R Core Team (2018) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Scott, D. W. (2015) *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.

Scrucca, L., Fop, M., Murphy, T. B. and Raftery, A. E. (2017) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, **8**, 205–233.

Tenenbaum, J. B., De Silva, V. and Langford, J. C. (2000) A global geometric framework for nonlinear dimensionality reduction. *science*, **290**, 2319–2323.

Wu, H.-M. (2008) Kernel sliced inverse regression with applications to classification. *Journal of Computational and Graphical Statistics*, **17**, 590–610.

Xie, B., Pan, W. and Shen, X. (2010) Penalized mixtures of factor analyzers with application to clustering high-dimensional microarray data. *Bioinformatics*, **26**, 501–508.