1-1-2016

Collective estimation of multiple bivariate density functions with application to angular-sampling-based protein loop modeling

Mehdi Maadooliat
Marquette University, mehdi.maadooliat@marquette.edu

Lan Zhou
Texas A & M University - College Station

Seyed Morteza Najibi
Persian Gulf University

Xin Gao
King Abdullah University of Science and Technology

Jianhua Z. Huang
Texas A & M University - College Station

Accepted version. *Journal of the American Statistical Association*, Vol. 111, No. 513 (2016): 43-56. DOI. © 2016 American Statistical Association. Used with permission.

Collective Estimation of Multiple Bivariate Density Functions with Application to Angular-Sampling-Based Protein Loop Modeling

Mehdi Maadooliat
Department of Mathematics, Statistics and Computer Science, Marquette University
Milwaukee, WI

Lan Zhou
Department of Statistics, Texas A&M University,
College Station, TX

Seyed Morteza Najibi
Department of Statistics, Persian Gulf University, Bushehr

Xin Gao
Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST),
Thuwal, Saudi Arabia.

Jianhua Z. Huang
Department of Statistics, Texas A&M University,
College Station, TX

Journal of the American Statistical Association, Vol. 111, No. 513 (2016): pg. 43-56. DOI.

Abstract: This paper develops a method for simultaneous estimation of This paper develops a method for simultaneous estimation of density functions for a collection of populations of protein backbone angle pairs using a data-driven, shared basis that is constructed by bivariate spline functions defined on a triangulation of the bivariate domain. The circular nature of angular data is taken into account by imposing appropriate smoothness constraints across boundaries of the triangles. Maximum penalized likelihood is used to fit the model and an alternating blockwise Newton-type algorithm is developed for computation. A simulation study shows that the collective estimation approach is statistically more efficient than estimating the densities individually. The proposed method was used to estimate neighbor-dependent distributions of protein backbone dihedral angles (i.e., Ramachandran distributions). The estimated distributions were applied to protein loop modeling, one of the most challenging open problems in protein structure prediction, by feeding them into an angular-sampling-based loop structure prediction framework. Our estimated distributions compared favorably to the Ramachandran distributions estimated by fitting a hierarchical Dirichlet process model; and in particular, our distributions showed significant improvements on the hard cases where existing methods do not work well.

Keywords: Bivariate splines, Log-spline density estimation, Protein structure, Ramachandran distribution, Roughness penalty, Triangulations

1 Introduction

An important topic in the field of structural biology is the determination of the three-dimensional (3D) structure of a protein. A protein is a linear chain of amino acids, each of which is composed of an amino group (−NH2), a central carbon atom (C_α), a carboxyl group (−COOH), and a side-chain group that is attached to C_α and is specific to each amino acid. When amino acids are chained into a peptide, the carboxyl group of the previous amino acid reacts with the amino group of the following one, releases a water molecule and forms a peptide bond. In a protein, each amino acid is called a residue and the chain of carbon, nitrogen and oxygen atoms is referred to as the backbone. While the side-chain structures determine local structures and interactions of the amino acids of the protein, the backbone structure describes the overall shape of the protein and is the focus of much research.

The backbone structure can be either specified by the 3D coordinates of the backbone atoms or the backbone angles when the peptide bonds are assumed to have the same length. Although most problems in the protein structure field depend on coordinate-based methods, backbone-angle-based methods have provided an attractive alternative approach in various protein structure-related problems, such as protein structure prediction (Simons et al., 1999; Hamelryck et al., 2006; Boomsma et al., 2008; Zhao et al., 2010), protein loop modeling (Ting et al., 2010), model quality assessment (Benkert et al., 2008; Gao et al., 2009; Archie and Karplus, 2009), prediction server ranking (Qiu et al., 2008; Maadooliat et al., 2013a), protein structure alignment (Miao et al., 2008; Challis and Schmidler, 2012), free energy function learning (Mu et al., 2005; Altis et al., 2008; Riccardi et al., 2009), and molecular dynamics simulation (Altis et al., 2007). In this paper, we focus on statistical modeling of the bivariate distribution of protein backbone angles.

There are two typical ways to represent backbone angles of proteins, i.e., the (ϕ, ψ) representation and the (θ, τ) representation. The (ϕ, ψ) representation is defined by dihedral angles along the chain of all backbone atoms, whereas the (θ, τ) representation is defined by planar and torsion angles along the C_α trace; see Figure 1 and also Oldfield and Hubbard (1994). Shortly after Kendrew et al. (1960) solved the first protein structure at atomic resolution, Ramachandran et al. (1963) studied the corresponding angular distribution. Since then, it has been found that different amino acids and secondary structures have different distributions in both the (ϕ, ψ) space (Ramachandran et al., 1963) and the (θ, τ) space (Hamelryck et al., 2006).

To understand the protein angular distributions, the circular nature of the angular data (i.e., −180° and 180° corresponds to the same configuration) demands non-traditional statistical methods. Parametric families for angular data have been proposed in Mardia (1975), Rivest (1988), and Singh et al. (2002), but they usually do not fit the actual protein data well (Mardia et al., 2007). There have been sufficient interests in developing more flexible models for bivariate protein angular data sets. In particular, Pertsemlidis et al. (2005) used a finite number of Fourier basis functions. Mardia et al. (2007) considered a finite mixture of bivariate von Mises distributions. Built on the work by Dahl et al. (2008), Lennox et al. (2009) developed a nonparametric Bayesian model consisting of a Dirichlet process mixture of bivariate von Mises distributions. These studies have provided excellent starting points for applying sophisticated statistical methods on protein structure related scientific problems.

The purpose of this paper is to develop a flexible density estimation method for collectively estimating multiple bivariate angular densities. By "collective estimation", we mean putting data from multiple distributions into one model and estimating all distributions together. We assume that multiple probability densities have some common features so that the log density functions can be represented using a common set of basis functions while each log density has its own coefficient vector in the basis expansion. The basis shared by the collection of density functions is not pre-specified but rather estimated as a low-dimensional manifold of a large space spanned by a rich basis. The functions in the rich basis are modeled as bivariate splines on a triangulation and roughness penalties are introduced to regularize the estimated bivariate splines. The circular nature of the angular data is respected by imposing appropriate smoothness constraints.

Though there is a large literature on nonparametric density estimation (Silverman, 1986; Stone, 1990; Scott, 1992; Gu, 1993; Hansen et al., 1998), existing methods have focused on estimation of a single density. Compared with estimating each density individually, the proposed collective estimation approach has several advantages. Firstly, the collective estimation approach allows pooling data and borrowing strength across distributions to achieve better estimation efficiency. Secondly, by using a common basis, the dimensionality of the parameter space for characterizing all distributions is significantly reduced. Furthermore, each estimated density has a concise representation using the coefficients of the basis expansion and these coefficients can be used for visualization, clustering, and classification purposes. Finally, this collective density estimation approach likely has unique advantages for protein angles due to the physical constraints on conformation. The proposed method is most useful in estimating multiple densities when the sample sizes are small. We shall demonstrate using a simulation study that our collective estimation approach can substantially improve estimation efficiency over a non-collective estimation approach using the kernel density estimators.

Ramachandran plot (Ramachandran and Sasisekharan, 1968) is a scatter plot commonly used to visualize the backbone angle pairs, (ϕ, ψ). The estimated probability density function of the Ramachandran plot is referred to as the Ramachandran distribution, which has become a fundamental concept in various protein structure-related problems, such as structural model checking (Laskowski et al., 1993; Hooft et al., 1997; Davis et al., 2004), protein structure prediction (Rohl et al., 2004; Zhao et al., 2010), side chain rotamer library (Bhuyan and Gao, 2011; Shapovalov and Dunbrack, 2011), and empirical energy functions (Buck et al., 2006). Ramachandran distributions are known to be affected by the secondary structure (Hovmöller et al., 2002; Jha et al., 2005) and the amino acid type (Berkholz et al., 2009) of the residue from which ϕ and ψ angles are calculated, as well as the neighboring amino acids (Keskin et al., 2004; Lennox et al., 2009; Ting et al., 2010). The neighbor-dependent Ramachandran distributions can reveal detailed relationships between protein sequences and structures, and provide significantly more accurate distributions to the aforementioned structure-related problems. However, density estimation of the neighbor-dependent Ramachandran distributions is difficult because when we focus on a specific amino acid while conditioning on the neighboring amino acids, the data are fractionated into groups each of which may contain only a small number of data points. This issue becomes more severe when the distributions are further conditioned on different secondary structures, i.e., α-helices, β-strands, and loops. By pooling the fractionated data together, our method can overcome the data sparsity problem and therefore improves the accuracy of density estimation. More accurate estimation of the probability density functions for the Ramachandran distributions can help improve protein structure prediction (Ting et al., 2010).

We applied the proposed collective density estimation method to estimate the neighbor-dependent Ramachandran distributions of protein loop regions and used the estimated distributions for angular-sampling-based protein loop modeling. Protein loop modeling remains as one of the most challenging problems in protein structure prediction, and is a key step in comparative modeling, protein design and structure refinement problems (Mandell et al., 2009; Stein and Kortemme, 2013; Ting et al., 2010). Although the flexible nature of loop structures makes modeling backbone angular distributions for protein loops much more difficult than that for regular secondary structures including α-helices and β-strands, our collective density estimation approach has shown promises. On a benchmark data set of reconstructing short loops, we compared our method with the state-of-art angular-sampling-based protein loop modeling procedures and observed competitive performance in terms of the ability to sample high-quality loops and the ability to select good loops by using the energy function.

The rest of the paper is organized as follows. Section 2 presents the core of the proposed method and Section 3 provides implementation details. Section 4 reports simulation results to illustrate the proposed collective estimation approach and to compare it with a non-collective estimation approach. Application to neighbor-dependent Ramachandran distributions of loop regions and sampling-based protein loop modeling is given in Section 5. Section 6 concludes the paper and Appendices collect some technical details.

2 Collective estimation of multiple probability density functions

This section presents the main components of the proposed collective density estimation approach, including the probabilistic model, model identifiability, and penalized likelihood estimation. Throughout the rest of this paper, the Greek letters ϕ, ψ, θ, τ will be used in mathematical equations. Such use of notation should not be confused with the names of protain backbone angles, as can be easily seen from the context.

2.1 A model for multiple density functions using a shared basis

Consider a collection of m probability distributions with density functions $f_i, i = 1, \cdots, m$. We have data observed from each distribution and we would like to estimate the density functions together. The rationale of this collective density estimation approach is the assumption that the density functions in the collection can be represented by a shared basis.

Assume that, up to a constant, each log density function can be represented by a linear combination of a common set of basis functions $\phi_k(x), k = 1, \cdots, K$, and each has its own set of coefficients. Specifically, we assume that $\log\{f_i(x)\} = \omega_i(x) - c_i$ with

$$\omega_i(x) = \sum_{k=1}^{K} \phi_k\,(x)\alpha_{ik}, \quad i = 1, \cdots, m, \tag{1}$$

and $c_i = \log\{\int \exp \omega_i(x)dx\}$ is a normalizing constant to ensure that the integral of the density function is 1. Equivalently, the density functions can be written as

$$f_i(x) = \frac{\exp\omega_i(x)}{\int \exp \omega_i\,(x)dx} = \exp\left\{\sum_{k=1}^{K} \phi_k(x)\alpha_{ik} - c_i\right\}, \tag{2}$$

For identifiability, we require that $1, \phi_k, k = 1, \ldots, K$, are linearly independent. We would like K to be a small number so that the number of parameters to be estimated is kept at a manageable scale even when we estimate a large number of density functions (i.e., m is large).

If the basis functions $\{\phi_k(x), k = 1, \cdots, K\}$ were given, the density functions would belong to an exponential family of order K. However, in our setting the basis functions are not pre-specified and need to be determined by the data. To this end, we suppose that these basis functions fall in a low-dimensional subspace of a function space spanned by a rich family of fixed basis functions, $\{b_l(x), l = 1, \cdots, L\}(L \gg K)$, such that

$$\phi_k(x) = \sum_{l=1}^{L} b_l(x)\theta_{lk}\,, \quad k = 1\,\dots, K. \tag{3}$$

For identifiability, we require that $1, b_l, l = 1, \dots, L$, are linearly independent. A large enough L ensures the needed flexibility in representing the unknown densities. For univariate densities, the fixed basis can be the monomials, B-splines, or the Fourier basis. Bivariate splines can be used as the fixed basis functions for bivariate densities; the details, including various complications for the specific application we consider, will be given in Section 3.

To simplify the presentation, we now introduce some vectors and matrices to denote the quantities of interest. Denote $\phi(x) = (\phi_1(x), \cdots, \phi_K(x))^\top$, $\boldsymbol{\alpha}_i = (\alpha_{i1}, \cdots, \alpha_{iK})^\top$, $\mathbf{b}(x) = (b_1(x), \cdots, b_L(x))^\top$, $\theta_k = (\theta_{1k}, \cdots, \theta_{Lk})^\top$, and $\Theta = (\theta_1, \cdots, \theta_K)$. Then, from (1) and (3) we can rewrite $\omega_i(x)$ in the vector-matrix form as

$$\omega_i(x) = \phi(x)^\top \boldsymbol{\alpha}_i = \mathbf{b}x^\top \Theta\alpha_i, \quad i = 1, \dots, m. \tag{4}$$

We also denote $\mathbf{A} = (\alpha_1, \dots, \alpha_m)^\top$. The unknown parameters can then be collectively written as the pair $(\boldsymbol{\Theta}, \mathbf{A})$. There is an identifiability issue caused by the non-uniqueness of the parametrization of $(\boldsymbol{\Theta}, \mathbf{A})$. This issue can be resolved by introducing some restrictions on the parameterization; see Appendix A.

We could have used the fixed basis $\{b_l(x), l = 1, \cdots, L\}$ in (1) and (2), however that would be either too restrictive (if L is small) or incur a large number of parameters (if L is large). Alternatively, if we were to model the individual density functions separately using the fixed basis $\{b_l(x), l = 1, \cdots, L\}$, we would write

$$\omega_i(x) = \boldsymbol{b}(x)^\top \psi_i, \quad i = 1, \dots, m. \tag{5}$$

Let $\Psi = (\psi_1, \ldots, \psi_m)^\top$ be the $m \times L$ matrix of coefficients from the basis expansions given in (5). Comparing (4) and (5), we obtain that $\Psi = \boldsymbol{A}\Theta^\top$, which is a rank-$K$ matrix. Thus, the collective modeling approach introduces a low-rank structure to the coefficient matrix in the basis expansion of the log densities. This dimensionality reduction allows us to significantly reduce the number of parameters to be estimated and thus gain estimation efficiency.

2.2 Penalized likelihood estimation

Suppose we have available data $x_{i\,j}, j = 1, \cdot\cdot\cdot, n_i$, from the ith distribution, $i = 1 \ldots, m$. The log likelihood is

$$\ell(\Theta, \mathrm{A}) = \sum_{i=1}^{m} \sum_{j=1}^{n_i} \left\{ \omega_i(x_{ij}) - \log \int \exp \omega_i(x)\, dx \right\},$$

(6)

where $\omega_i(x)$ are defined in (3). It is concave in α_i when other parameters are fixed and also concave in θ_k when other parameters are fixed. Applying the roughness penalty approach of function estimation (Green and Silverman, 1994), we estimate the model parameters by minimizing the following penalized likelihood criterion

$$-2\ell(\boldsymbol{\Theta}, \mathbf{A}) + \lambda \sum_{k=1}^{K} \mathbf{PEN}(\phi_k)$$

(7)

where **PEN**(ϕ_k) is a roughness penalty function that regularizes the estimated basis function ϕ_k to ensure that it is a smooth function, and $\lambda > 0$ is a penalty parameter. The penalty function can usually be written as a quadratic form

$$\sum_{k=1}^{K} \mathbf{PEN}\left(\phi_k\right) = \sum_{i=k}^{K} \theta_k^{\top} \boldsymbol{R} \theta_k = \mathrm{tr}\{\boldsymbol{\Theta}^{\top} \boldsymbol{R} \boldsymbol{\Theta}\}. \tag{8}$$

For univariate density estimation, noticing that $\phi_k(x) = \mathbf{b}(x)^{\top}\, \theta_k$, we have that $\mathbf{R} = \int \ddot{\mathbf{b}}\,(x) \ddot{\mathbf{b}}(x)^{\top}\, dx$ with $\ddot{\boldsymbol{b}}(t) = (b_1''\,(t), \ldots, b_L''\,(t))^{\top}$ if we use the usual squared-second-derivative penalty $\mathrm{PEN}(\phi_k) = \int\{\phi_k''(x)\}^2 dx$. The form of **R** for bivariate density estimation is given in Section 3.

We use an alternating blockwise Newton-Raphson algorithm to minimize the penalized log likelihood. Our algorithm cycles through updating of $\alpha_i, i = 1, \ldots, m$, and $\theta_k, k = 1, \ldots, K$, until convergence. Following the usual step-halving strategy for the Newton-Raphson iteration, the updating formulas are

$$\alpha_i^{new} = \alpha_i^{old} - \tau \left[\frac{\partial^2}{\partial \alpha_i \partial \alpha_i^{\top}} \{\ell(\Theta, \mathrm{A})\}\right]^{-1} \left[\frac{\partial}{\partial \alpha_i} \{\ell(\Theta, \mathrm{A})\}\right] \Big|_{\Theta = \Theta^{old}, \mathrm{A} = \mathrm{A}^{old}} \tag{9}$$

and

$$\alpha_k^{new} = \alpha_k^{old} - \tau \left[\frac{\partial^2}{\partial \alpha_k \partial \alpha_k^{\top}} \{\ell(\Theta, \mathrm{A})\} - \lambda R\right]^{-1} \left[\frac{\partial}{\partial \alpha_k} \{\ell(\Theta, \mathrm{A})\}\right] \Big|_{\Theta = \Theta^{old}, \mathrm{A} = \mathrm{A}^{old}} \tag{10}$$

where τ is taken as the first one from the sequence $\{(1/2)^t, t = 0, 1, \ldots\}$ such that the objective function in (7) is reduced. The expressions of the gradient and Hessian of the log likelihood are given in Appendix B. The initial values of the Newton-Raphson iteration can be obtained by projecting some raw density estimates such as KDE to the model space of (2).

We select the penalty parameter by minimizing the AIC (Akaike, 1973):

$$\mathrm{AIC}(\lambda) = -2\ell\left(\widehat{\Theta}, \widehat{\mathrm{A}}\right) + 2\mathrm{df}, \tag{11}$$

where $\ell(\Theta,\mathbf{A})$ is the log likelihood defined in (6), and the degrees of freedom df is defined as

$$\mathrm{df} = \sum_{k=1}^{K} \operatorname{trace}\left\{\left[\sum_{i=1}^{m} n_i \alpha_{ik}^2 \operatorname{var}_i\{\mathrm{b}(\mathrm{X})\} + \lambda \mathbf{R}\right]^{-1} \left[\sum_{i=1}^{m} n_i \alpha_{ik}^2 \operatorname{var}_i\{\mathbf{b}(\mathrm{X})\}\right]\right\}. \tag{12}$$

The parameters in these formulas are replaced by their estimated values. The AIC can be derived as an approximation to the leave-one-out cross-validation (O'Sullivan, 1988; Gu, 2002).

2.3 The number of basis functions

We identify the number of basis functions, K, using the scree plot as typically used in principal component analysis (Jolliffe, 2002). Using the fit from an initial model with a large K (i.e., $K = \min\{m, L\}$), we plot the sum of squares of the component coefficients as the function of the component index, that is, $\sum_i \alpha_{ik}^2$ vs k, and find the "elbow" that locates a suitable value of K.

3 Implementation details for bivariate density estimation

The neighbor-dependent Ramachandran distributions encountered in our application are bivariate distributions. This section discusses details for implementing the proposed method for this bivariate case, including construction of the fixed basis using bivariate splines, imposition of various constraints on the basis functions, and formation of the roughness penalty.

3.1 Triangulation and bivariate splines

We assume that the densities are defined on a polygonal set $\Omega \subset \mathbb{R}^2$. To construct a suitable fixed basis $\{b_l(x), l = 1, \ldots, L\}$ to be used in (3), we apply bivariate splines on a triangulation (Lai and Schumaker, 2007). A triangulation of Ω partitions the domain into triangles; see Figure 3 for some examples. A bivariate spline is a piecewise bivariate polynomial (i.e., being a polynomial on each triangle) with the polynomial pieces joining together smoothly. Unlike for univariate splines where B-splines are available and easy to compute, constructing a locally supported basis for bivariate splines is complicated. We thus take a different strategy: we first represent bivariate polynomials on each triangle in the Bernstein-Bézier form (B-form), and then join together the polynomials on adjacent triangles by imposing smoothness constraints across the edges. As shown in Lai and Schumaker (2007), the smoothness constraints can be written as a linear system of equations on the coefficients of the B-form representation. To take into account the circular nature of the angular data, we need to put identical triangle edges at the angles of −180° and 180° and impose smoothness constraints across these edges. In next subsection, we show how to construct a basis under these constraints, along with the identifiability constraint mentioned earlier.

3.2 Construction of the fixed basis functions to satisfy constraints

Let $\mathbb{G}_0 = \{\tilde{\mathbf{b}}(x)^\top \tilde{\beta}\}$ be an L_0-dimensional linear space spanned by the basis $\tilde{\mathbf{b}}(x)$. Let $\mathbb{G} = \{\tilde{\boldsymbol{b}}(x)^\top \tilde{\beta}, \mathbf{H}\tilde{\beta} = \mathbf{0}\}$ be the L-dimensional linear subspace of $\mathbb{G}_0$ obtained by imposing the constraints $\mathbf{H}\tilde{\beta} = 0$ on coefficients of the basis expansion, where **H** is a given $(L^0 - L) \times L_0$ matrix. In our application, $\mathbb{G}_0$ is the space of piecewise bivariate polynomials on a triangulation and is easy to construct, and $\mathbb{G}$ is the space of splines with smoothness constraints written in the form of a set of linear equations. Consider the QR decomposition

$$\mathbf{H}^\top = \underset{L_0 \times L_0}{\mathbf{Q}} \begin{bmatrix} \mathbf{R} \\ (L_0 - L) - L \times (L_0 - L) \\ \mathbf{0} \\ \mathrm{L} \times (\mathrm{L}_0 - \mathrm{L}) \end{bmatrix} = \begin{bmatrix} \underset{L_0 \times (L_0 - L)}{\mathbf{Q}_1} & \vdots \underset{L_0 \times L}{\mathbf{Q}_2} \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{R}$ is an upper triangular matrix. Then $\tilde{\beta} = \boldsymbol{Q}_2\beta$ for an unconstrained β will satisfy the constraints $\mathbf{H}\tilde{\beta} = 0$. In fact,

$$\mathbf{H}\tilde{\beta} = [\mathbf{R}^\top \mathbf{0}] \begin{bmatrix} \mathbf{Q}_1^\top \\ \mathbf{Q}_2^\top \end{bmatrix} \mathbf{Q}_2 \beta = [\mathbf{R}^\top \mathbf{0}] \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \beta = \mathbf{0}.$$

It follows that $\mathbf{b}(x) = \boldsymbol{Q}_2^\top \tilde{\mathbf{b}}(x)$ is a desired basis for $\mathbb{G}$.

The method in the previous paragraph can also be used to construct a fixed basis $b_1(x),\ldots,\, b_L(x)$ such that 1, $b_1(x),\ldots, b_L(x)$ are linearly independent, required by identifiability (Section 2.1). To be more specific, assume we start with a basis $\tilde{\mathbf{b}}\,(x) = (\tilde{b}_1(x),\ldots,.\tilde{b}_{L_0}\,(x))^\top$ but it is not linearly independent with the constant function 1. There is a vector β_0 such that $1 = \tilde{\mathbf{b}}(x)^\top \beta_0$. Let $\mathbf{h}_0$ be a vector such that $\mathbf{h}_0^\top\, \beta_0 \neq 0$. Applying the construction method, we can obtain a basis $\mathbf{b}(x)$ for the linear space $\mathbb{G} = \{\tilde{\mathbf{b}}(x)^\top\beta,\, \mathbf{h}_0^\top\, \beta = 0\}$. We claim that $\mathbf{b}(x)$ is linearly independent of the constant function 1 and thus is the desired basis. See Appendix C for a proof of this claim. When we use the bivariate spline basis discussed in Section 3.1, $\beta_0 = \mathbf{1}$, the vector of 1's. For convenience, we used $\mathbf{h}_0 = \mathbf{1}$ in our implementation of the method.

3.3 Roughness penalty

For a function $g(x), x = (x_1, x_2)$, defined on a region Ω of $\mathbb{R}^2$, denote the partial derivatives as

$$g_{ij}(x_1, x_2) = \frac{\partial g(x_1,x_2)}{\partial x_i \partial x_j}, \qquad i, j = 1, 2.$$

The thin-plate penalty (Wahba, 1990; Green and Silverman, 1994) is defined as

$$\mathbf{PEN}(g) = \iint_\Omega (g_{11}^2 + 2\, g_{12}^2 + g_{22}^2)\, dx_1 dx_2.$$

Suppose that there is a basis expansion $g(x) = \mathbf{b}(x)^\top \beta$, where $\mathbf{b}(x)$ is a vector of basis functions. Let $\mathbf{b}_{ij}(x) = (b_{1,ij}(x),\ldots, b_{L,ij}(x))^\top$ be a vector of partial derivatives of the component functions of $\mathbf{b}(x)$ for i, j

$= 1, 2$. Then $g_{ij}(x) = \mathbf{b}_{ij}^{\top}(x)\beta$, and the penalty function can be written in quadratic form as

$$\mathrm{PEN}(g) = \beta^{\top}\mathbf{R}\beta \tag{13}$$

with the penalty matrix

$$\mathbf{R} = \iint_{\Omega}\{\mathbf{b}_{11}(x)\mathbf{b}_{11}^{\top}(x) + 2\,\mathbf{b}_{12}(x)\mathbf{b}_{12}^{\top}(x) + \mathbf{b}_{22}(x)\mathbf{b}_{22}^{\top}(x)\}\,dx. \tag{14}$$

When Ω is a collection of triangles, the above integration can be computed as the summation of integrations over all triangles. We apply the penalty matrix defined in (14) when we compute the penalty function in (8) for bivariate density estimation.

As shown in Subsection 3.2, it is convenient to construct a desirable basis $\mathbf{b}(x)$ by projecting a larger basis $\widetilde{\boldsymbol{b}}(x)$ onto a constrained space, using $\mathbf{b}(x) = \mathbf{Q}_2^{\top}\tilde{\mathbf{b}}(x)$. Suppose that the penalty matrix corresponding to the basis $\tilde{\mathbf{b}}(x)$ is $\widetilde{\mathbf{R}}$ (defined as in (14) with an obvious change of notation), then the penalty matrix for $\mathbf{b}(x)$ can be obtained as $\mathbf{R} = \mathbf{Q}_2^{\top}\widetilde{\mathbf{R}}\mathbf{Q}_2$.

4 Simulations

We conducted a simulation study to evaluate the proposed collective density estimation method and compared it with a non-collective density estimation approach using the kernel density estimator. From now on, we refer to our proposed method as PSCDE (penalized spline collective density estimator). The simulation setups were designed to mimic actual protein angular distributions. Hamelryck et al. (2006) reported that there exists a very strong concentration of the dihedral/planar $(\theta - \tau\,)$ angles around $\theta_1^* = 95$ and $\tau_1^* = 50$ for α-helices, and a relatively strong concentration around $\theta_2^* = 120$ and $\tau_2^* = -165$ for β-strands. Motivated by this observation, we considered bivariate distributions in the following form

$$\begin{pmatrix}\theta\\ \\ \tau\end{pmatrix} \sim \delta \begin{pmatrix}\mathrm{WN}(\theta_1^*, \rho_\theta)\\ \\ \mathrm{WN}(\tau_1^*, \rho_\tau)\end{pmatrix} + (1-\delta)\begin{pmatrix}\mathrm{WN}(\theta_2^*, \rho_\theta)\\ \\ \mathrm{WN}(\tau_2^*, \rho_\tau)\end{pmatrix} \tag{15}$$

with $\rho_0 = 0.99$ and $\rho_\tau = 0.975$, where $\mathrm{WN}(\mu, \rho)$ is the wrapped normal distribution on the unit circle with mean direction μ and concentration parameter ρ (Jammalamadaka and SenGupta, 2001). Data generated from (15) with a large value of δ are similar to angles from α-helices, and with a small value of δ are similar to angles from β-strands.

We used the following model of m bivariate distributions to generate simulated data. The m distrbutions are clustered into three sets, each of which contains $m/3$ distributions from (15) and corresponds to respective mixture parameters δ = 0.96 (mimicking α-helices), δ = 0.20, and δ = 0.04 (mimicking β-strands). We generated n pairs of angles from each distribution. Different values of m and n were considered. Note that for simplicity in generating the data the distributions in each cluster were chosen to be the same, but they were treated as different distributions when we applied the estimation methods.

The kernel density estimator (KDE, Wand and Jones, 1995) is a widely used nonparametric density estimator. A typical form of the *p*-dimensional kernel density estimator is

$$\hat{f}_h(\mathbf{X}) = \frac{1}{n}\left(\prod_{k=1}^{p} h_k\right)^{-1} \sum_{i=1}^{n} K\left(\frac{x_1 - x_{i1}}{h_1}, \ldots, \frac{x_p - x_{ip}}{h_p}\right),$$

where *K* is a *p*-variate kernel function satisfying $\int K(\mathbf{x})d\mathbf{x} = 1$ and $\mathbf{h} = (h_1, \cdots, h_p)$ is known as the bandwidth vector that controls the smoothness of the density estimate. To take into account the circular nature of the angular data, Maadooliat et al. (2013a) suggested the following modified bivariate kernel estimator

$$\hat{f}_{h_1,h_2}(x_1, x_2) = \frac{\sum_{i=1}^{n} \varphi\left(\frac{x_1 \ominus x_{i1}}{h_1}\right) \varphi\left(\frac{x_2 \ominus x_{i2}}{h_2}\right)}{nh_1h_2} \tag{16}$$

where $\varphi(\bullet)$ is the standard Gaussian density function, and the notation $\ominus$ is used to denote the distance between two points on a unit circle. For example, if $\omega_1 = 359°$ and $\omega_2 = 1°$, the Euclidean distance $|\omega_1 - \omega_2| = 358$, but the difference on the unit circle is $|\omega_1 \ominus \omega_2| = 2$. Our use of distance on the circle yields a smooth density on the manifold of angular space and prevents the boundary effect from the naïve use of the kernel density estimation.

The bandwidth is an important tuning parameter for the KDE. We considered the following five methods for bandwidth selection: a well-supported rule-of-thumb method for choosing the bandwidth of a Gaussian KDE (rKDE, Venables and Ripley, 2002), plug-in bandwidth selector (Hpi, Chacón and Duong, 2010), biased cross-validation (Hlscv, Sain et al., 1994), smoothed cross-validation (Hlscv, Jones et al., 1991), and least-squares cross-validation (Hlscv, Sain et al., 1994). The first bandwidth selector is implemented in the MASS package of R and the other four bandwidth selectors are implemented in the ks package of R (Duong, 2007; Chacón and Duong, 2011).

Figure 2 shows the perspective plots for the true bivariate density, the estimated density by the rKDE and the PSCDE for data generated from a density corresponding to $\delta = 0.04$ with $m = 42$ and $n = 50$. The plots were drawn on the same scale for ease of direct visual comparison. We observe that the KDE obtains more peaks than that exist in the true density, while the PSCDE is closer to the truth. This is also clearly seen in the contour plot of Figure 3. Figure 2 also shows the scatter plot of the first two coefficients ($\mathbf{A}_{\cdot 2}$ vs $\mathbf{A}_{\cdot 1}$) in the distributions fitted by the PSCDE. We observe a clear separation into three classes, indicating that these coefficients are also useful for clustering purposes.

Next we present the results from a systematic simulation study. For each of the two models, we considered six different cases from all possible combinations of $m = 6, 18$ and $n = 30, 50, 100$. To evaluate the

DOI.

performance of a method in estimating angular densities, we used three distance measures, namely, the integrated absolute distance (IAD), the Hellinger distance (HLD), and the symmetrized Kullback-Leibler divergence (SKLD). For distribution functions F and G with corresponding densities f and g, these distances are defined as

$$\mathrm{IAD}(F,G) = \int |f(\mathbf{x}) - g(\mathbf{x})| d\mathbf{x}, \tag{17}$$

$$\mathrm{HLD}(F,G) = \left[\int \left\{\sqrt{f(\mathbf{x})} - \sqrt{g(\mathbf{x})}\right\}^2 d\mathbf{x}\right]^{1/2}, \tag{18}$$

$$\mathrm{SKLD}(F,G) = \int \{f(\mathbf{x}) - g(\mathbf{x})\} log \left\{\frac{f\ (\mathbf{x})}{g(\mathbf{x})}\right\} d\mathbf{x}. \tag{19}$$

More details about these metrics can be found in DasGupta (2011).

We generated data from each simulation setup, applied both the PSCDE (with $K = 2$, suggested by the scree plot) and the KDE (with five different bandwidth selectors) on the generated data. For each data set and a given method, we computed the distance between the estimated density and the true density using the three distance measures mentioned above. We ran the simulation 100 times for each setup. The empirical means and standard errors of the distances are reported in Table 1. For both methods the distance between the estimated and the true densities decreases as the number of observations (n) increases. This is due to the increment of estimation efficiency by increasing the sample size. When the number of distributions (m) gets larger, the performance of the proposed PSCDE improves, while the performance of KDE does not change. The PSCDE clearly outperforms KDE in all setups no matter which bandwidth selector is used and its superiority gets enhanced when m gets larger. This result suggests that the PSCDE can improve estimation efficiency by borrowing strength across distributions while the non-collective estimation method of KDE does not have such ability.

5 Application: Neighbor-dependent Ramachandran distributions for protein loop modeling by Rosetta

5.1 Background on angular-sampling-based protein structure prediction

In nature, a protein folds into its native structure, which is the key to understanding its functions and behaviors in complex biological networks. Although high-throughput sequencing technologies have been advanced in recent years, experimental protein structure determination by X-ray crystallography or nuclear magnetic resonance spectroscopy remains a costly and time-consuming process, causing an increasing gap between the number of known protein sequences and that of known structures. Therefore, computational protein structure prediction has become an important alternative to experimental methods.

One successful approach for computational protein structure prediction is angular-sampling based methods (Rohl et al., 2004; Bystroff et al., 2000; Tuffery and Derreumaux, 2005; Hamelryck et al., 2006; Sellers et al., 2008; Boomsma et al., 2008; Mandell et al., 2009; Ting et al., 2010; Lennox et al., 2010; Zhao et al., 2010; Stein and Kortemme, 2013; Maadooliat et al., 2013b; Källberg et al., 2014). Compared with other sampling-based methods, angular-sampling-based methods have the advantage of being able to model the continuous conformational space of proteins. Every angular-sampling-based method has two key steps: (1) sampling realistic and nativelike conformations; (2) identifying the good conformations. The sampling step requires accurate estimation of torsion angle distributions that captures the local relationships between sequences and structures. The identification step selects good conformations from the sampled ones or direct searches the comformation space by minimizing a suitable energy function, which in turn is chosen to distinguish correct, native-like structures from incorrect ones. An energy function is often specified as a weighted linear combination of a number of statistical and empirical terms, such as that encode bond lengths, bond angles, torsion angles, van-der-Waals interactions, and electrostatic interactions.

Among existing computational protein structure prediction softwares, Rosetta is one of the most accurate and commonly used. It provides a flexible library of functionality to accomplish a diverse set of biomolecular modeling tasks. The kinematic inversion closure (KIC) protocol in Rosetta was developed by Mandell et al. (2009) to reconstruct high-resolution loop structures. Loop structures are irregular parts of proteins which play important roles in protein function, stability and folding (Fetrow, 1995). They are often conformationally flexible and cannot be modeled using standard homology modeling techniques. In KIC, the torsion angles, (ϕ, ψ), are sampled from an estimated Ramachandran distribution of the associated amino acids to effectively explore the conformational space. The sampling step is followed by a Monte-Carlo minimization step that involves an empirical energy function. Mandell et al. (2009) demonstrated that Rosetta with this KIC procedure can accurately predict native-like structures of protein loop regions.

Following the promising results of the KIC in obtaining accurate predictions for the local protein structures, Stein and Kortemme (2013) developed a new protocol, called "next-generation KIC" (NGK), to further improve Rosetta's KIC protocol. NGK consists of a combination of several strategies, including: (a) intensification: aim to intensify sampling of certain regions by sampling (ϕ, ψ) from neighbor-dependent Ramachandran distributions (referred to as Rama2b sampling); and (b) annealing: modulate the energy function and gradually ramp the weight of terms in the Rosetta energy function to overcome energy barriers. In both of these strategies, the amino-acid-specific Ramachandran distributions used in KIC are replaced by distributions specific to the amino acid and its immediate left or right neighbor. The neighbor-dependent Ramachandran distributions used in NGK are provided by fitting a hierarchical Dirichlet process (HDP) model (Ting et al., 2010). From here on we refer to the Stein and Kortemme procedure as NGK.HDP.

5.2 Comparison of estimations of neighbor-dependent Ramachandran distributions

Below we demonstrate that our proposed collective density estimation method offers a competitive alternative to HDP in estimating the neighbor-dependent Ramachandran distributions. We replaced the neighbor-dependent Ramachandran distributions in NGK obtained by applying HDP with those obtained from applying PSCDE, and refer to the new protocol as NGK.PSCDE. To facilitate a fair comparison, except the replacement of the neighbor-dependent Ramachandran distributions, all other components of NGK remain the same. We evaluated the performance of different density estimation methods using the task of protein loop structure prediction.

We used the same data set provided in Ting et al. (2010) to obtain the neighbor-dependent Ramachandran distributions when applying the proposed PSCDE method. The data set is generated from 3, 038 proteins with available electron densities from the Uppsala Electron Density Server (Kleywegt et al., 2004). As in Ting et al. (2010), we considered a set of 62,345 residues after removing those with electron density in the bottom 20th percentile and restricting the set to loop residues with no missing backbone atoms and at least three residues away from α-helices or β-strands. For each amino acid type, we applied PSCDE (with $K = 4$) to collectively estimate the $m = 20$ left-neighbor-dependent Ramachandran distributions; we also applied PSCDE to collectively estimate the $m = 20$ right-neighbor-dependent Ramachandran distributions. Since there are 20 possible amino acid types, we obtained 800 (= 20 × (20 + 20)) neighbor-dependent estimated density functions. The number of data points available for each of these 800 density functions, i.e., n_i in (6), ranges from 6 to 620, with median 131 and quartiles 68.75 and 213.20.

In our comparison of NGK.HDP and NGK.PSCDE for protein loop modeling, we also included Rosetta's KIC protocol as a benchmark. It is noteworthy that Rosetta is the most comprehensive method as well as one of the most accurate protein structure prediction methods, which has consistently won the CASP (Critical Assessment of Protein Structure Prediction) competitions (Cozzetto et al., 2009; Kryshtafovych et al., 2011, 2014). Thus, any improvement on the performance upon Rosetta is considered significant in the protein structure prediction community.

We assessed the three methods by reconstructing the structures of short loops (12-residue segments) from an established benchmark data set with 20 proteins. This benchmark was compiled by Zhu et al. (2006) to allow direct comparisons among studies by Jacobson et al. (2004), Zhu et al. (2006), and Sellers et al. (2008). It was selected from high quality structures (resolution ≤ 2.0Å, R < 0.25) for loops with diverse sequences (< 40% sequence identity), low temperature factors (< 35), lack of contacts to heteroatom groups (> 4.0Å for neutral ligands, > 6.5Å for metal ions), lack of secondary structure within the loop, lack of more than 4 loop residues adjacent to either loop endpoint, and pH 6.5.7.5; see Mandell et al. (2009) for more details. Although KIC and NGK.HDP have demonstrated considerable success in sampling and identifying near-native conformations on this benchmark (Stein and Kortemme, 2013), for some of the proteins, sub-angstrom conformations (i.e., reconstructed loops that are within 1Å to the native structure) were either not sampled or not identified correctly by the energy function.

In our comparative study, the assigned loop in each protein was deleted and then "reconstructed" using KIC, NGK.HDP and NGK.HDP methods, respectively. For each of the 20 benchmark proteins, we reconstructed 500 loop structures for the associated assigned loop (12-residue) using different methods (KIC, NGK.HDP and NGK.PSCDE). Figure 4 provides a sketch of five randomly selected KIC reconstructions of a loop for one of the benchmark proteins, "PDB id: 1CB0". The fact that none of the five reconstructions match the true structure very well indicates the difficulty of the problem.

Following Stein and Kortemme (2013), we used two metrics to evaluate the performance of each method: The first metric is the percentage of reconstructed loops that are within 1Å to the native structure (i.e., sub-angstrom cases), denoted as $\%sA$. The second metric is the lowest root mean square deviation (RMSD) of the backbone atoms between the 10 lowest energy reconstructed loops and the native structure, denoted as RMSD*. The first metric measures the ability to sample high-quality loops, whereas the second metric measures the ability to select good loops by using the energy function. Using these two metrics, we obtained an overall comparison of the three methods based on the reconstructed 500 structures for each case. The results are summarized in Table 2. For "1BN8", none of the three methods were able to generate any structure predictions (probably due to some internal issues of Rosetta), so both $\%\text{sA}$ and RMSD* are unavailable for this protein. For two out of the rest of 19 proteins ("1CNV" and "1CS6"), we did not obtain any sub-angstrom structure using any of the three methods, as indicated by $\%\text{sA}$ being 0.0.

Table 2 indicates that the KIC obtains the highest $\%\text{sA}$ for 3 proteins ("1I7P", "1MS9" and "1MY7"), while both NGK.HDP and NGK.PSCDE obtain the highest $\%\text{sA}$ for 7 proteins each. Therefore, the two NGK methods seem to be competitive with respect to the percentage of sub-angstrom structures criterion on this benchmark. However, an important observation from Table 2 is that for all six hard cases, i.e., the $\%\text{sA}$ is less than 10% for KIC, the proposed NGK.PSCDE clearly outperforms the other two methods by giving the highest $\%\text{sA}$ value. This is significant because angular-sampling-based methods are most useful for such hard protein targets, whereas for relatively easy targets with close homologs, non-sampling-based methods such as template-based modeling methods are often sufficiently accurate. In terms of the second evaluation criterion, the lowest RMSD among the top ten reconstructed loops selected by the energy function, the proposed NGK.PSCDE significantly outperforms both KIC and NGK.HDP. In fact, NGK.PSCDE obtains the smallest RMSD* for 14 out of the 19 proteins, and obtains the second smallest RMSD* for 4 of the rest 5 proteins.

Figure 5 illustrates the relationship between the empirical energy function and the RMSD that we obtained for reconstructing the 500 12-residue loops for the PBD entry "1OYC" using the three methods (KIC, NGK.HDP and NGK.PSCDE). It is clear that NGK.PSCDE not only generates more high-quality loops than KIC and NGK.HDP, but also has a higher correlation between the energy value of the predicted loop and the RMSD to the native structure, especially in the sub-angstrom region. Note that a lower energy value does not necessarily imply a lower RMSD and vice versa, however, a higher correlation between the energy value and the RMSD indicates that searching a structure with lower energy will likely find a structure that is closer to the native structure in terms of RMSD.

Finally, Figure 6 presents the best model fits (in terms of RMSD*) by the three methods and the native structure, for four proteins ("1CB0", "1F46", "ICS6" and "1OYC") arbitrarily selected out of 14 cases where NGK.PSCDE outperforms the other two methods in terms of RMSD*. It is clear that the loops predicted by NGK.PSCDE match the native structures very well, while the predictions from KIC and NGK.HDP do not match well for three and two proteins respectively among those four proteins.

6 Conclusion

This paper develops a novel approach for collectively estimating multiple bivariate densities. By pooling data from different distributions and using a shared basis, the collective estimation approach is statistically more efficient than non-collective estimation approaches. The proposed method uses penalized bivariate splines on a triangulation to yield a flexible family of bivariate densities. As an output of applying the new method, each estimated log density is expressed in a basis expansion where the basis is estimated from the data, assuming that the densities lie in a low-dimensional manifold of the large space spanned by a pre-specified rich basis. The collective density estimation approach is widely applicable when there is a need to estimate multiple density functions from different populations. Moreover, the coefficients of the basis expansion for the fitted densities provide a low-dimensional representation that could be useful for visualization, clustering, and classification of the densities. We applied the new method to estimate the neighbor-dependent Ramachandran distributions and the estimated distributions show competitive performance for angular-sampling-basis protein loop modeling.

One limitation of the our approach is that the possible dependence of data from the same density is not modeled and thus our likelihood should be interpreted as a composite likelihood if dependence exists. The consequence of not modeling the dependence is the potential loss of efficiency and the incorrect degrees of freedom used in AIC. Dependence should not be a serious problem for the application of modeling the neighbor-dependent Ramachandran distributions, because when conditioning on the neighboring amino acids, subsetting the data substantially reduces the dependence. Extending the proposed method to dependent data is an interesting research topic.

Acknowledgment

We are grateful to Dr. Roland L. Dunbrack for providing the data set for the neighbor-dependent Ramachandran distribution application, and to Amelie Stein for help with implementation of Rosseta. We would also like to thank the editor, associate editor and referees for their constructive and thoughtful comments which helped us tremendously in improving the manuscript. Zhou's work was partially supported by NSF grant DMS-0907170. Huang's work was partially supported by NSF grants DMS-0907170, DMS-1007618, DMS-1208952. Gao's work was supported by competitive research funding from King Abdullah University of Science and Technology (KAUST). This work was also supported by Award KUS-CI-016-04 and FCC/1/1976-04-01 by KAUST.

Appendix A: Identifiability of (Θ,A)

The non-uniqueness of the parametrization of $(\boldsymbol{\Theta}, \boldsymbol{A})$ causes an identifiability issue. Specifically, if $\boldsymbol{U}$ is a $K\times K$ orthogonal matrix, then $\boldsymbol{\Theta\alpha}_{\mathrm{i}} = (\boldsymbol{\Theta}\mathbf{U})(\mathbf{U}^\top\boldsymbol{\alpha}_{\mathrm{i}})$. Thus $\widetilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta}\mathbf{U}$ and $\widetilde{\boldsymbol{\alpha}}_i = \mathbf{U}^\top\boldsymbol{\alpha}_{\mathrm{i}}$ give the same representation (4). To gain identifiability, we require that (i) $\boldsymbol{\Theta}^\top\boldsymbol{\Theta} = \mathbf{I}$, (ii) $\mathbf{A}^\top\mathbf{A} = \mathbf{D}^2$ be a diagonal matrix, (iii) the columns of $\mathbf{A}$ be ordered such that the diagonal elements of $\mathbf{D}^2$ are in strictly decreasing order, and (iv) the first non-zero element of each column of $\boldsymbol{\Theta}$ be positive. With such $\boldsymbol{\Theta}$ and $\mathbf{A}$, if the diagonal elements of $\mathbf{D}$ are all different, setting $\overline{\mathbf{A}} = \mathbf{A}\mathbf{D}^{-\mathbf{1}}$ and so $\overline{\mathbf{A}}^\top \overline{\mathbf{A}} = \mathbf{I}$, we have that $\boldsymbol{\Theta}\boldsymbol{A}^\top = \boldsymbol{\Theta}\boldsymbol{D}\overline{\boldsymbol{A}}^\top$ which is a uniquely defined singular value decomposition (SVD). The desired identifiability of $(\boldsymbol{\Theta},\mathbf{A})$ then follows from the uniqueness of the SVD.

Appendix B: Gradient and Hessian of the log likelihood function

Let $E^{\omega}(\cdot)$ and $\mathrm{var}^{\omega}(\cdot)$ denote respectively the expectation and covariance operator with respect to the density ω. Then

$$E^{\omega i}\{\mathbf{b}(X)\} = \frac{\int \exp\omega_i(x)\mathbf{b}(x)dx}{\int \exp\omega_i(x)dx}, \tag{20}$$

and

$$\mathrm{var}^{\omega_i}\{\mathbf{b}(X)\} = \frac{\int \exp\,\omega_i(x)\mathbf{b}(x)\mathbf{b}(x)^{\top}dx}{\int \exp\,\omega_i(x)\,dx} - \frac{\{\int \exp\,\omega_i(x)\mathbf{b}(x)dx\}\{\int \exp\,\omega_i(x)\mathbf{b}(x)dx\}^{\top}}{\{\int \exp\,\omega_i(x)dx\}^2} \tag{21}$$

Here the exponential function when applied to a vector is treated as a component-wise operation; the expectation and integration operators are interpreted in the same manner.

Denote $\beta_i = \boldsymbol{\Theta}\boldsymbol{\alpha}_i$ so that $\omega_i(x) = \mathbf{b}(x)^{\top}\beta_i$. Some simple calculation yields

$$\frac{\partial}{\partial\beta_i}\log\int \exp\,\omega_i(x)dx = E^{\omega_i}\{\mathbf{b}(X)\}, \tag{22}$$

and

$$\frac{\partial^2}{\partial\beta_i\partial\beta_i^{\top}}\log\int\ \exp\omega_i(x)dx = \mathrm{var}^{\omega_i}\{\mathbf{b}(X)\}. \tag{23}$$

These facts are properties of the exponential family and are useful when computing the gradient and Hessian of the log likelihood.

Journal of the American Statistical Association, Vol. 111, No. 513 (2016): pg. 43-56. DOI. This article is © American Statistical Association and permission has been granted for this version to appear in e-Publications@Marquette. American Statistical Association does not grant permission for this article to be further copied/distributed or hosted elsewhere without the express permission from American Statistical Association.

To compute the log likelihood, we need the following expressions

$$\sum_{j=1}^{n_i} \omega_i\,(xij) \;=\sum_{j=1}^{n_i} \mathbf{b}(x_{ij})^\top \boldsymbol{\Theta}\boldsymbol{\alpha}_i \;=\sum_{j=1}^{n_i} \mathbf{b}(x_{ij})^\top(\boldsymbol{\theta}_1\alpha_{i1} + \cdots + \boldsymbol{\theta}_K\alpha_{iK}),$$

and

$$\int \exp\omega_i(x)dx = \int \exp\mathbf{b}(x)^\top \Theta\alpha_i dx \;= \int \exp\mathbf{b}(x)^\top(\theta_1\alpha_{i1} + \cdots + \theta_K\alpha_{iK})dx.$$

Using these expressions we obtain that

$$\frac{\partial}{\partial\alpha_i}\sum_{j=1}^{n_i}\omega_i(x_{ij}) \;=\; \Theta^\top\sum_{j=1}^{n_i}\boldsymbol{b}(x_{ij}), \quad i \;=\; 1,\ldots,m$$

$$\frac{\partial}{\partial\theta_k}\sum_{j=1}^{n_i}\omega_i(xij) \;=\; \alpha_{ik}\sum_{j=1}^{n_i}\mathbf{b}(xij), \quad k \;=\; 1,\ldots,K$$

$$\frac{\partial}{\partial\alpha_i}\log\int\exp\omega_i(x)dx \;=\; \boldsymbol{\Theta}^\top E^{\omega_i}\{\mathbf{b}(X)\}, \;\; i \;=\; 1,\ldots,m$$

$$\frac{\partial}{\partial\theta_k}\log\int\exp\omega_i(x)dx = \;\alpha_{ik}E^{\omega_i}\{\mathbf{b}(X)\}, \qquad k \;=\; 1,\ldots,K.$$

The last two equations follow from (22) and the chain rule. Equation (23) and the chain rule together gives the following useful expressions

∂2∂□*i*∂□T*i*log   expω*i*(*x*) *dx* = □Tvarω*i* {**b**(*X*)}□, *i* = 1, . . . ,*m*

∂2∂ *k*∂ T*k*log   expω*i*(*x*) *dx* = α2*ik* varω*i* {**b**(*X*)}, *k* = 1, . . . , *K*.

Using the expressions in previous paragraph, we obtain that the gradient vector of the log likelihood is given by

$$\frac{\partial}{\partial \alpha_i}\{\ell(\boldsymbol{\Theta}, \boldsymbol{A})\} = \boldsymbol{\Theta}^\top \sum_{j=1}^{n_i} [\mathbf{b}(x_{ij}) - E^{\omega_i}\{\mathbf{b}(X)\}], \tag{24}$$

$$\frac{\partial}{\partial \theta_k}\{\ell(\boldsymbol{\Theta}, \boldsymbol{A})\} = \sum_{i=1}^{m} \alpha_{ik} \sum_{j=1}^{n_i} [\mathbf{b}(x_{ij}) - E^{\omega_i}\{\mathbf{b}(X)\}], \tag{25}$$

and the diagonal blocks of the Hessian are given by

$$\frac{\partial^2}{\partial \alpha_i \partial \alpha_i^\top}\{\ell(\boldsymbol{\Theta}, \mathbf{A})\} = -n_i \boldsymbol{\Theta}^\top \mathrm{var}^{\omega_i}\{\mathbf{b}(X)\}\boldsymbol{\Theta}, \quad i = 1, \dots, m \tag{26}$$

$$\frac{\partial^2}{\partial \theta_k \partial \theta_k^\top}\{\ell(\boldsymbol{\Theta}, \mathbf{A})\} = -\sum_{i=1}^{m} n_i \alpha_{ik}^2 \, \mathrm{var}^{\omega_i}\{\mathbf{b}(X)\}, \quad k = 1, \dots, K. \tag{27}$$

Note that the quantities in (26) and (27) are non-positive definite. It follows that the log likelihood function is concave in α_i when other parameters are fixed and also concave in θ_k when other parameters are fixed. The expectation and variance appeared in the gradient and Hessian can be computed using numerical integration.

Appendix C: Proof of the Claim in Section 3.2

Suppose

$$1 \cdot c_0 + \mathbf{b}(x)^\top \boldsymbol{c} = 0, \tag{28}$$

we show that $c_0 = 0$ and $\mathbf{c} = \mathbf{0}$. Since $\mathbf{b}(x)^\top\mathbf{c} \in \mathbb{G}$, there is a vector β with $\boldsymbol{h}_0^\top \beta = 0$ such that $\mathbf{b}(x)^\top\mathbf{c} = \tilde{\mathbf{b}}(x)^\top\beta$. This together with (28) and $1 = \tilde{\mathbf{b}}(x)^\top\beta_0$ yields that $\tilde{\mathbf{b}}(x)^\top(c_0\beta_0 + \beta) = 0$, which in turn implies that $c_0\beta_0 + \beta = \mathbf{0}$, because $\tilde{\mathbf{b}}(x)$ is a basis. Thus,

$$0 = \mathbf{h}_0^\top(c_0\beta_0 + \beta) = c_0\mathbf{h}_0^\top\beta_0 + \mathbf{h}_0^\top\beta = c\mathbf{h}_0^\top\beta_0.$$

Since $\mathbf{h}_0^\top\beta_0 \neq 0$, we conclude that $c_0 = 0$. Plugging this into (28), we obtain $\mathbf{b}(x)^\top\mathbf{c} = 0$. Because $\mathbf{b}(x)$ is a basis, we have that $\mathbf{c} = \mathbf{0}$. This completes the proof of the claim.

References

Altis, A., Nguyen, P. H., Hegger, R., and Stock, G. (2007), "Dihedral angle principal component analysis of molecular dynamics simulations," *Journal of Chemical Physics*, 126, 244111.

Altis, A., Otten, M., Nguyen, P. H., Hegger, R., and Stock, G. (2008), "Construction of the free energy landscape of biomolecules via dihedral angle principal component analysis," *Journal of Chemical Physics*, 128, 245102.

Archie, J. and Karplus, K. (2009), "Applying undertaker cost functions to model quality assessment," *Proteins*, 75, 550.555.

Benkert, P., Tosatto, S. C. E., and Schomburg, D. (2008), "QMEAN: A comprehensive scoring function for model quality assessment," *Proteins*, 71, 261-277.

Berkholz, D. S., Shapovalov, M. V., Dunbrack, Jr, R. L., and Karplus, P. A. (2009), "Conformation dependence of backbone geometry in proteins," *Structure*, 17, 1316-1325.

Bhuyan, M. S. I. and Gao, X. (2011), "A protein-dependent side-chain rotamer library," *BMC Bioinformatics*, 12(Suppl 14):S10, 1-12.

Boomsma, W., Mardia, K. V., Taylor, C. C., Ferkinghoff-Borg, J., Krogh, A., and Hamelryck, T. (2008), "A generative, probabilistic model of local protein structure." *Proc Natl Acad Sci U S A*, 105, 8932-8937.

Buck, M., Bouguet-Bonnet, S., Pastor, R. W., and MacKerell, Jr, A. D. (2006), "Importance of the CMAP correction to the CHARMM22 protein force field: dynamics of hen lysozyme," *Biophysical Journal*, 90, L36-L38.

Bystroff, C., Thorsson, V., and Baker, D. (2000), "HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins." *J Mol Biol*, 301, 173-190.

Chacón, J. and Duong, T. (2010), "Multivariate plug-in bandwidth selection with unconstrained pilot bandwidth matrices," *TEST: An Official Journal* *of the Spanish Society of Statistics and Operations Research*, 19, 375-398.

— (2011), "Unconstrained pilot selectors for smoothed cross-validation," *Australian* & *New Zealand Journal of Statistics*, 53, 331-351.

Challis, C. J. and Schmidler, S. C. (2012), "A stochastic evolutionary model for protein structure alignment and phylogeny," *Molecular Biology and Evolution*, 29, 3575-3587.

Cozzetto, D., Kryshtafovych, A., and Tramontano, A. (2009), "Evaluation of CASP8 model quality predictions." *Proteins*, 77 Suppl 9, 157-166.

Dahl, D. B., Bohannan, Z., Mo, Q., Vannucci, M., and Tsai, J. W. (2008), "Assessing Side-chain Perturbations of the Protein Backbone: A Knowledge Based Classification of Residue Ramachandran Space," *Journal of Molecular Biology*, 378, 749-758.

DasGupta, A. (2011), *Probability for Statistics and Machine Learning: Fundamentals and Advanced Topics*, Springer Texts in Statistics, Springer.

Davis, I. W., Murray, L. W., Richardson, J. S., and Richardson, D. C. (2004), "MOLPROBITY: structure validation and all-atom contact analysis for nucleic acids and their complexes," *Nucleic Acids Research*, 32, W615-W619.

Duong, T. (2007), "ks: Kernel density estimation and kernel discriminant analysis for multivariate data in R," *Journal of Statistical Software*, 21, 1-16.

Fetrow, J. S. (1995), "Omega loops: nonregular secondary structures significant in protein function and stability." *The FASEB Journal*, 9, 708-717.

Gao, X., Xu, J., Li, S. C., and Li, M. (2009), "Predicting local quality of a sequence-structure alignment," *Journal of Bioinformatics and Computational Biology*, 7, 789-810.

Green, P. and Silverman, B. (1994), *Nonparametric regression and generalized linear models: a roughness penalty approach*, Chapman & Hall/CRC.

Gu, C. (1993), "Smoothing spline density estimation: A dimensionless automatic algorithm," *Journal of the American Statistical Association*, 88, 495-504.

— (2002), *Smoothing Spline ANOVA Models*, Springer Series in Statistics, Springer.

Hamelryck, T., Kent, J. T., and Krogh, A. (2006), "Sampling realistic protein conformations using local structural bias." *PLoS Comput Biol*, 2, e131.

Hansen, M., Kooperberg, C., and Sardy, S. (1998), "Triogram models," *Journal of the American Statistical Association*, 93, 101-119.

Hooft, R. W., Sander, C., and Vriend, G. (1997), "Objectively judging the quality of a protein structure from a Ramachandran plot," *Computer applications in the biosciences : CABIOS*, 13, 425-430.

Hovmöller, S., Zhou, T., and Ohlson, T. (2002), "Conformations of amino acids in proteins," *Acta Crystallogr D Biol Crystallogr*, 58, 768-776.

Jacobson, M. P., Pincus, D. L., Rapp, C. S., Day, T. J., Honig, B., Shaw, D. E., and Friesner, R. A. (2004), "A hierarchical approach to all-atom protein loop prediction," *Proteins: Structure, Function, and Bioinformatics*, 55, 351-367.

Jammalamadaka, S. and SenGupta, A. (2001), *Topics in Circular Statistics*, Series on Multivariate Analysis, World Scientific.

Jha, A., Colubri, A., Zaman, M., Koide, S., Sosnick, T., and Freed, K. (2005), "Helix, sheet, and polyproline II frequencies and strong nearest neighbor effects in a restricted coil library," *Biochemistry*, 44, 9691-9702.

Jolliffe, I. (2002), *Principal Component Analysis*, Springer Series in Statistics, Springer.

Jones, M. C., Marron, J. S., and Park, B. U. (1991), "A simple root *n* bandwidth selector," *The Annals of Statistics*, 19, 1919-1932.

K¨allberg, M., Margaryan, G., Wang, S., Ma, J., and Xu, J. (2014), "RaptorX server: a resource for template-based protein structure modeling." *Methods Mol Biol*, 1137, 17-27.

Kendrew, J. C., Dickerson, R. E., Strandberg, B. E., Hart, R. G., Davies, D. R., Phillips, D. C., and Shore, V. C. (1960), "Structure of myoglobin: A three-dimensional Fourier synthesis at 2 A. resolution," *Nature*, 185, 422-427.

Keskin, O., Yuret, D., Gursoy, A., Turkay, M., and Erman, B. (2004), "Relationships between amino acid sequence and backbone torsion angle preferences," *Proteins*, 55, 992-998.

Kleywegt, G. J., Harris, M. R., Zou, J.-y., Taylor, T. C., Wälby, A., and Jones, T. A. (2004), "The Uppsala Electron-Density Server," *Acta Crystallographica Section D*, 60, 2240-2249.

Kryshtafovych, A., Barbato, A., Fidelis, K., Monastyrskyy, B., Schwede, T., and Tramontano, A. (2014), "Assessment of the assessment: evaluation of the model quality estimates in CASP10." *Proteins*, 82 Suppl 2, 112-126.

Kryshtafovych, A., Fidelis, K., and Tramontano, A. (2011), "Evaluation of model quality predictions in CASP9." *Proteins*, 79 Suppl 10, 91-106.

Lai, M. and Schumaker, L. (2007), *Spline Functions on Triangulations*, no. v. 13 in Encyclopedia of Mathematics and Its Applications, Cambridge University Press.

Laskowski, R., MacArthur, M., Moss, D., and Thornton, J. (1993), "PROCHECK: a program to check the stereochemical quality of protein structures," *Journal of Applied Crystallography*, 26, 283-291.

Lennox, K. P., Dahl, D. B., Vannucci, M., Day, R., and Tsai, J. W. (2010), "A Dirichlet Process Mixture of HiddenMarkovModels for Protein Structure Prediction," *Ann Appl Stat*, 4, 916-942.

Lennox, K. P., Dahl, D. B., Vannucci, M., and Tsai, J. W. (2009), "Density Estimation for Protein Conformation Angles Using a Bivariate von Mises Distribution and Bayesian Nonparametrics," *Journal of the American Statistical Association*, 104, 586-596.

Maadooliat, M., Gao, X., and Huang, J. Z. (2013a), "Assessing protein conformational sampling methods based on bivariate lag-distributions of backbone angles," *Briefings in Bioinformatics*,14, 724-736.

— (2013b), "Assessing protein conformational sampling methods based on bivariate lagdistributions of backbone angles." *Brief Bioinform*, 14, 724-736.

Mandell, D. J., Coutsias, E. A., and Kortemme, T. (2009), "Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling," *Nature methods*, 6, 551-552.

Mardia, K. V. (1975), "Statistics of Directional Data," *Journal of the Royal Statistical Society, Series B: Methodological*, 37, 349-393.

Mardia, K. V., Taylor, C. C., and Subramaniam, G. K. (2007), "Protein Bioinformatics and Mixtures of Bivariate Von Mises Distributions for Angular Data," *Biometrics*, 63, 505-512.

Miao, X.,Waddell, P. J., and Valafar, H. (2008), "TALI: local alignment of protein structures using backbone torsion angles," *Journal of Bioinformatics and Computational Biology*, 6, 163-181.

Mu, Y., Nguyen, P. H., and Stock, G. (2005), "Energy landscape of a small peptide revealed by dihedral angle principal component analysis," *Proteins*, 58, 45-52.

Oldfield, T. J. and Hubbard, R. E. (1994), "Analysis of $C\alpha$ geometry in protein structures," *Proteins*, 18, 324-337.

O.Sullivan, F. (1988), "Fast computation of fully automated log-density and log-hazard estimators," *SIAM Journal on Scientific and Statistical Computing*, 9, 363-379.

Pertsemlidis, A., Zelinka, J., Fondon, J. W., Henderson, R. K., and Otwinowski, Z. (2005), "Bayesian Statistical Studies of the Ramachandran Distribution," *Statistical Applications in Genetics and Molecular Biology*, 4, 1-18.

Qiu, J., Sheffler, W., Baker, D., and Noble, W. S. (2008), "Ranking predicted protein structures with support vector regression," *Proteins*, 71, 1175-1182.

Ramachandran, G. and Sasisekharan, V. (1968), "Conformations of polypeptides and proteins," *Advances in Protein Chemistry*, 23, 283.

Ramachandran, G. N., Ramakrishnan, C., and Sasisekharan, V. (1963), "Stereochemistry of polypeptide chain configurations," *Journal of Molecular Biology*, 7, 95-99.

Riccardi, L., Nguyen, P. H., and Stock, G. (2009), "Free-energy landscape of RNA hairpins constructed via dihedral angle principal component analysis," *The Journal of Physical Chemistry B*, 113, 16660-16668.

Rivest, L. P. (1988), "A Distribution for Dependent Unit Vectors," *Communications in Statistics: Theory and Methods*, 17, 461-483.

Rohl, C. A., Strauss, C. E.M., Misura, K.M. S., and Baker, D. (2004), "Protein structure prediction using Rosetta." *Methods Enzymol*, 383, 66-93.

Sain, S. R., Baggerly, K. A., and Scott, D. W. (1994), "Cross-validation of multivariate densities," *Journal of the American Statistical Association*, 89, 807-817.

Scott, D. (1992), *Multivariate Density Estimation: Theory, Practice, and Visualization*, Wiley Series in Probability and Statistics, Wiley.

Sellers, B. D., Zhu, K., Zhao, S., Friesner, R. A., and Jacobson, M. P. (2008), "Toward better refinement of comparative models: predicting loops in inexact environments," *Proteins: Structure, Function, and Bioinformatics*, 72, 959-971.

Shapovalov, M. V. and Dunbrack, Jr, R. L. (2011), "A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions." *Structure*, 19, 844-858.

Silverman, B. (1986), *Density estimation for statistics and data analysis*, vol. 26, Chapman & Hall/CRC.

Simons, K. T., Bonneau, R., Ruczinski, I., and Baker, D. (1999), "Ab initio protein structure prediction of CASP III targets using ROSETTA," *Proteins*, 37 Suppl 3, 171-176.

Singh, H., Hnizdo, V., and Demchuk, E. (2002), "Probabilistic Model for Two Dependent Circular Variables," *Biometrika*, 89, 719-723.

Stein, A. and Kortemme, T. (2013), "Improvements to robotics-inspired conformational sampling in rosetta," *PloS one*, 8, e63090.

Stone, C. (1990), "Large-sample inference for log-spline models," *The Annals of Statistics*, 18, 717-741.

Ting, D., Wang, G., Shapovalov, M., Mitra, R., Jordan, M. I., and Dunbrack, Jr, R. L. (2010), "Neighbor-dependent Ramachandran probability distributions of amino acids developed from a hierarchical Dirichlet process model," *PLOS Computational Biology*, 6, e1000763.

Tuffery, P. and Derreumaux, P. (2005), "Dependency between consecutive local conformations helps assemble protein structures from secondary structures using Go potential and greedy algorithm." *Proteins*, 61, 732-740.

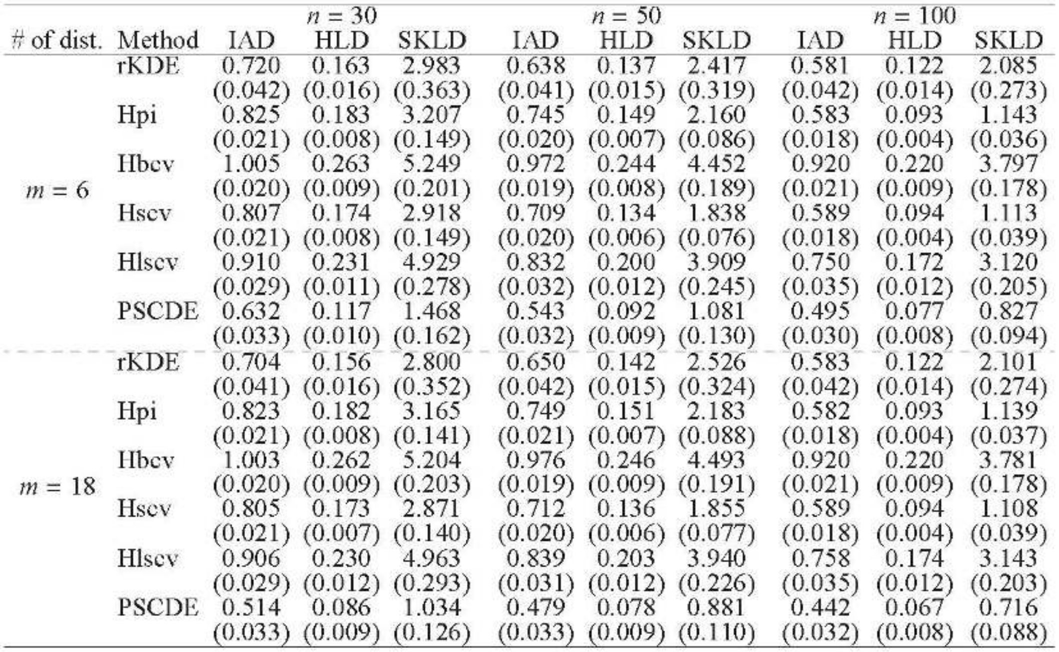
Venables, W. N. and Ripley, B. D. (2002), *Modern Applied Statistics with S*, New York: Springer, 4th ed.

Wahba, G. (1990), *Spline Models for Observational Data*, SIAM [Society for Industrial and Applied Mathematics].

Wand, P. and Jones, C. (1995), *Kernel Smoothing*, Monographs on Statistics and Applied Probability, Taylor & Francis.

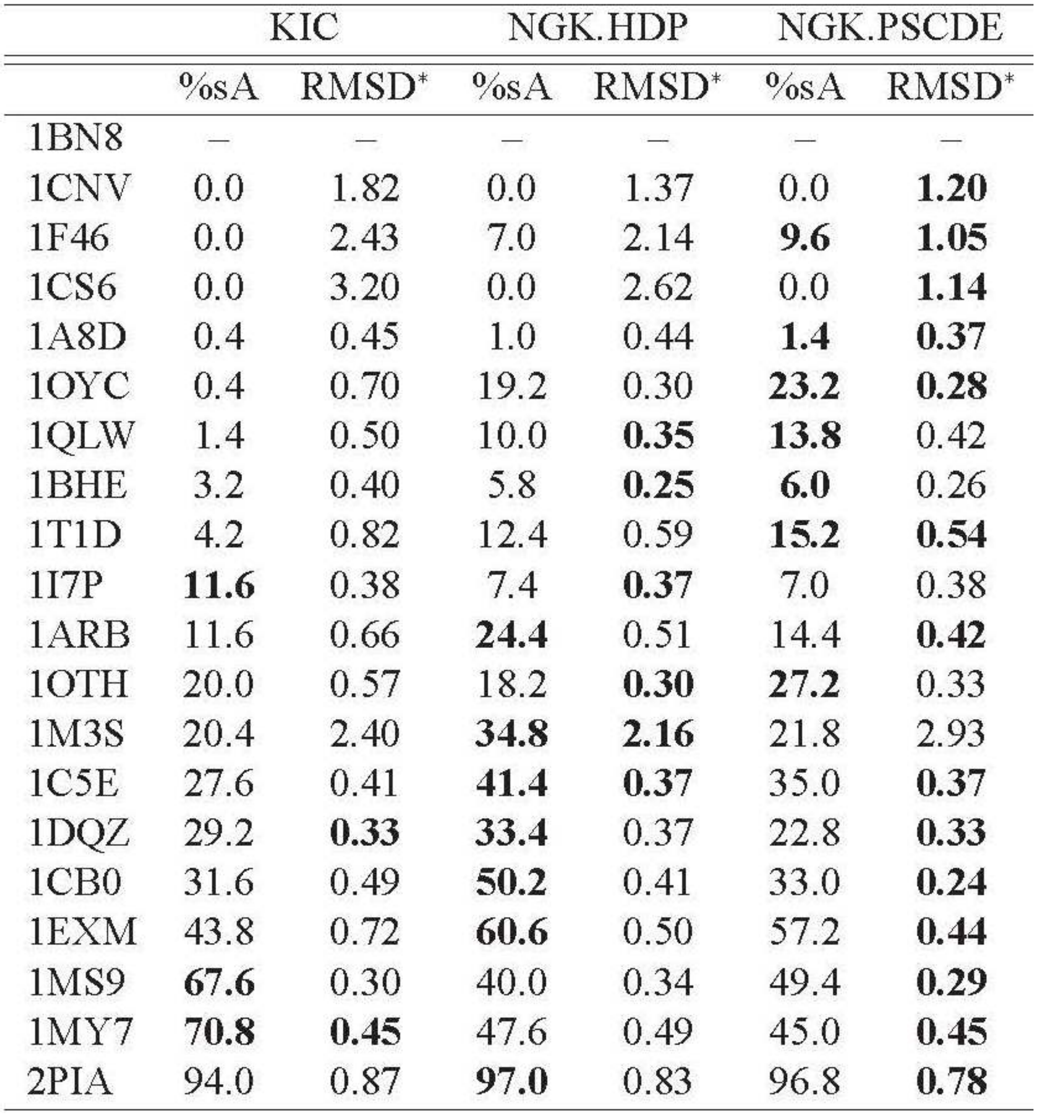
Zhao, F., Peng, J., Debartolo, J., Freed, K. F., Sosnick, T. R., and Xu, J. (2010), "A probabilistic and continuous model of protein conformational space for template-free modeling." *J Comput Biol*, 17, 783-798.

Zhu, K., Pincus, D. L., Zhao, S., and Friesner, R. A. (2006), "Long loop prediction using the protein local optimization program," *Proteins: Structure, Function, and Bioinformatics*, 65, 438-452.



		$n = 30$			$n = 50$			$n = 100$		
# of dist.	Method	IAD	HLD	SKLD	IAD	HLD	SKLD	IAD	HLD	SKLD
$m = 6$	rKDE	0.720	0.163	2.983	0.638	0.137	2.417	0.581	0.122	2.085
		(0.042)	(0.016)	(0.363)	(0.041)	(0.015)	(0.319)	(0.042)	(0.014)	(0.273)
	Hpi	0.825	0.183	3.207	0.745	0.149	2.160	0.583	0.093	1.143
		(0.021)	(0.008)	(0.149)	(0.020)	(0.007)	(0.086)	(0.018)	(0.004)	(0.036)
	Hbcv	1.005	0.263	5.249	0.972	0.244	4.452	0.920	0.220	3.797
		(0.020)	(0.009)	(0.201)	(0.019)	(0.008)	(0.189)	(0.021)	(0.009)	(0.178)
	Hscv	0.807	0.174	2.918	0.709	0.134	1.838	0.589	0.094	1.113
		(0.021)	(0.008)	(0.149)	(0.020)	(0.006)	(0.076)	(0.018)	(0.004)	(0.039)
	Hlscv	0.910	0.231	4.929	0.832	0.200	3.909	0.750	0.172	3.120
		(0.029)	(0.011)	(0.278)	(0.032)	(0.012)	(0.245)	(0.035)	(0.012)	(0.205)
	PSCDE	0.632	0.117	1.468	0.543	0.092	1.081	0.495	0.077	0.827
		(0.033)	(0.010)	(0.162)	(0.032)	(0.009)	(0.130)	(0.030)	(0.008)	(0.094)
$m = 18$	rKDE	0.704	0.156	2.800	0.650	0.142	2.526	0.583	0.122	2.101
		(0.041)	(0.016)	(0.352)	(0.042)	(0.015)	(0.324)	(0.042)	(0.014)	(0.274)
	Hpi	0.823	0.182	3.165	0.749	0.151	2.183	0.582	0.093	1.139
		(0.021)	(0.008)	(0.141)	(0.021)	(0.007)	(0.088)	(0.018)	(0.004)	(0.037)
	Hbcv	1.003	0.262	5.204	0.976	0.246	4.493	0.920	0.220	3.781
		(0.020)	(0.009)	(0.203)	(0.019)	(0.009)	(0.191)	(0.021)	(0.009)	(0.178)
	Hscv	0.805	0.173	2.871	0.712	0.136	1.855	0.589	0.094	1.108
		(0.021)	(0.007)	(0.140)	(0.020)	(0.006)	(0.077)	(0.018)	(0.004)	(0.039)
	Hlscv	0.906	0.230	4.963	0.839	0.203	3.940	0.758	0.174	3.143
		(0.029)	(0.012)	(0.293)	(0.031)	(0.012)	(0.226)	(0.035)	(0.012)	(0.203)
	PSCDE	0.514	0.086	1.034	0.479	0.078	0.881	0.442	0.067	0.716
		(0.033)	(0.009)	(0.126)	(0.033)	(0.009)	(0.110)	(0.032)	(0.008)	(0.088)

Table 1: Comparison of PSCDE and KDE with 5 different bandwidth selectors (rKDE, Hpi, Hbcv, Hscv, Hlscv) for the simulation study, with three sample sizes (n = 30, 50, 100) and two different numbers of distributions (m = 6, 18) using integrated absolute distance (IAD), Hellinger distance (HLD), and symmetrized Kullback-Leibler distance (SKLD). The empirical mean and standard errors (in parentheses) are reported, based on 100 simulation runs.



	KIC		NGK.HDP		NGK.PSCDE	
	%sA	RMSD*	%sA	RMSD*	%sA	RMSD*
1BN8	–	–	–	–	–	–
1CNV	0.0	1.82	0.0	1.37	0.0	**1.20**
1F46	0.0	2.43	7.0	2.14	**9.6**	**1.05**
1CS6	0.0	3.20	0.0	2.62	0.0	**1.14**
1A8D	0.4	0.45	1.0	0.44	**1.4**	**0.37**
1OYC	0.4	0.70	19.2	0.30	**23.2**	**0.28**
1QLW	1.4	0.50	10.0	**0.35**	**13.8**	0.42
1BHE	3.2	0.40	5.8	**0.25**	**6.0**	0.26
1T1D	4.2	0.82	12.4	0.59	**15.2**	**0.54**
1I7P	**11.6**	0.38	7.4	**0.37**	7.0	0.38
1ARB	11.6	0.66	**24.4**	0.51	14.4	**0.42**
1OTH	20.0	0.57	18.2	**0.30**	**27.2**	0.33
1M3S	20.4	2.40	**34.8**	**2.16**	21.8	2.93
1C5E	27.6	0.41	**41.4**	**0.37**	35.0	**0.37**
1DQZ	29.2	**0.33**	**33.4**	0.37	22.8	**0.33**
1CB0	31.6	0.49	**50.2**	0.41	33.0	**0.24**
1EXM	43.8	0.72	**60.6**	0.50	57.2	**0.44**
1MS9	**67.6**	0.30	40.0	0.34	49.4	**0.29**
1MY7	**70.8**	**0.45**	47.6	0.49	45.0	**0.45**
2PIA	94.0	0.87	**97.0**	0.83	96.8	**0.78**

Table 2: Comparing the performance of KIC, NGK.HDP and NGK.PSCDE for reconstructing short loops with the length of 12-residue for 20 benchmark proteins based on 500 simulation runs. The stars indicate the proteins whose sub-angstrom reconstruction is not seen. For each of the benchmark proteins, the method produces the highest percentage of sub-angstrom structures (%sA) is denoted as **bold**. Similarly, the method that produces the smallest RMSD☐ on the energy score is indicated in **bold**.

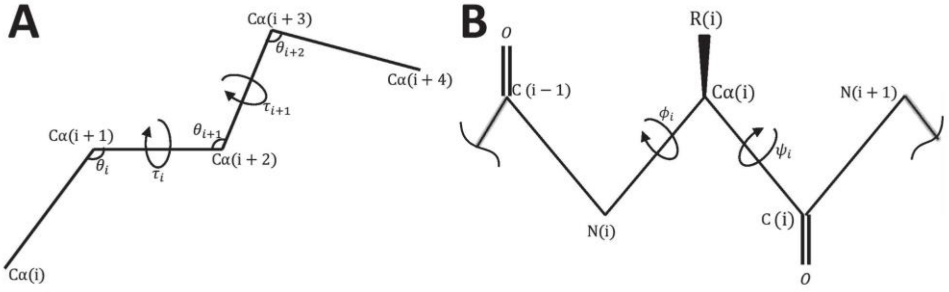


Figure 1: Schematic representation of the protein backbone angles. The atoms on the chain are labeled. (a) Angles along the Cα trace is denoted by (θ_i, τ_i), where θ_i is the pseudo-bond angle of three consecutive Cα atoms $(C_\alpha(i), C_\alpha(i + 1), C_\alpha(i + 2)$, and τ_i is the pseudo-torsion angle of four consecutive Cα atoms $(C_\alpha(i), \cdots, C_\alpha(i + 3)$. The term pseudo is used for (θ, τ) here because the consecutive Cα atoms are not actually connected by a single chemical bond. (b) Angles along the chain of all backbone atoms is denoted by (ϕ_i, ψ_i), where ϕ_i is the torsion angle formed by $C(i - 1), \mathrm{N}(i), C_\alpha(i), C(i)$ and ψ_i is the torsion angle formed by $\mathrm{N}(i), C_\alpha(i), C(i), \mathrm{N}(i + 1)$. A bond or planar angle is the angle formed between three consecutive atoms. For four atoms bonded together in a chain, the torsion or dihedral angle is the angle between the plane formed by the first three atoms and the plane formed by the last three atoms.

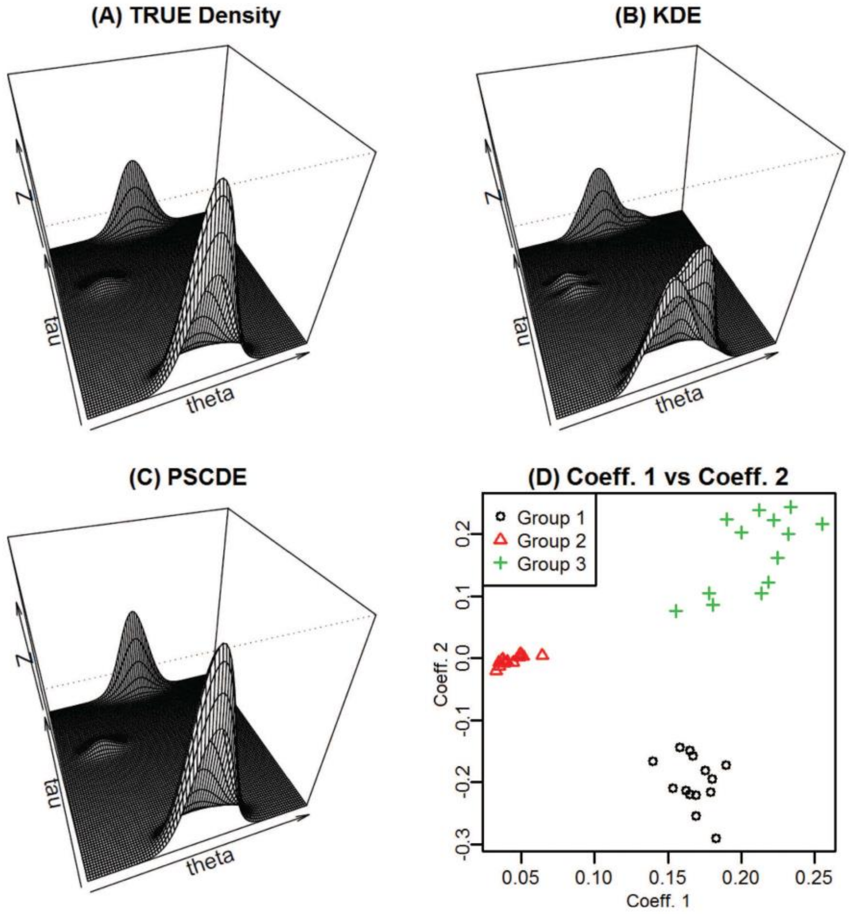


Figure 2: One of the $m/3 = 14$ densities corresponding to $\delta = 0.04$ from the simulation model with $m = 42$ and $n = 50$. (A)-(C): Perspective plots of the true density, rKDE estimate, and PSCDE; (D) scatter plot of the first two coefficients from the fitted PSCDE model.

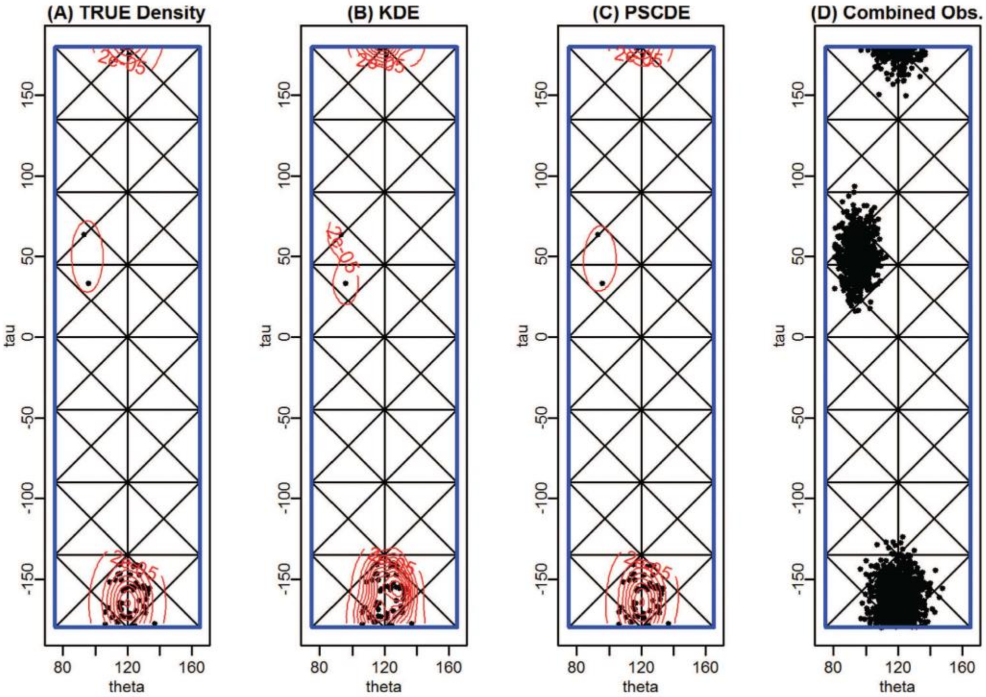


Figure 3: Contour plots of the densities shown in panels (A)-(C) of Figure 2, presented on the triangulation used by the PSCDE. Panel (D) shows all data points from the 42 densities in one simulation run.

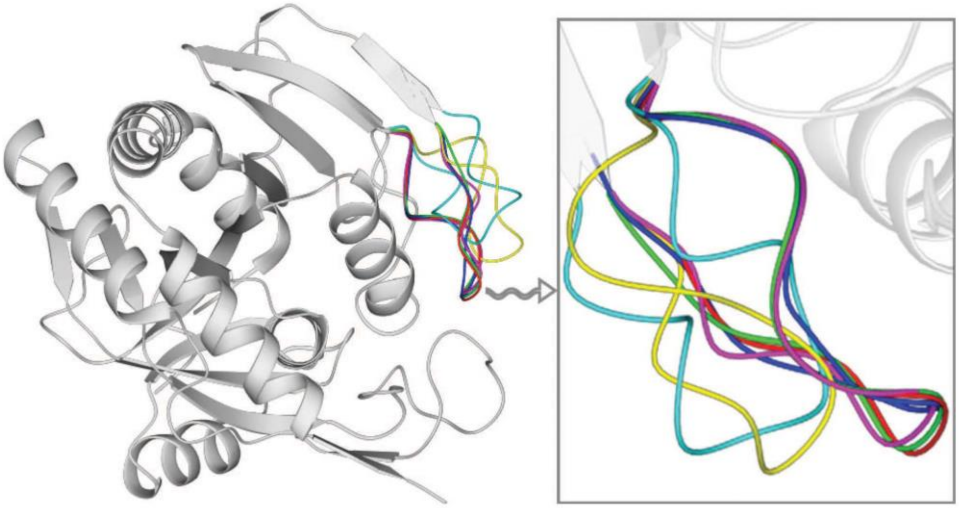


Figure 4: Five randomly selected reconstructed loop models for protein "1CB0" plus the native structure of the associated loop (in blue).

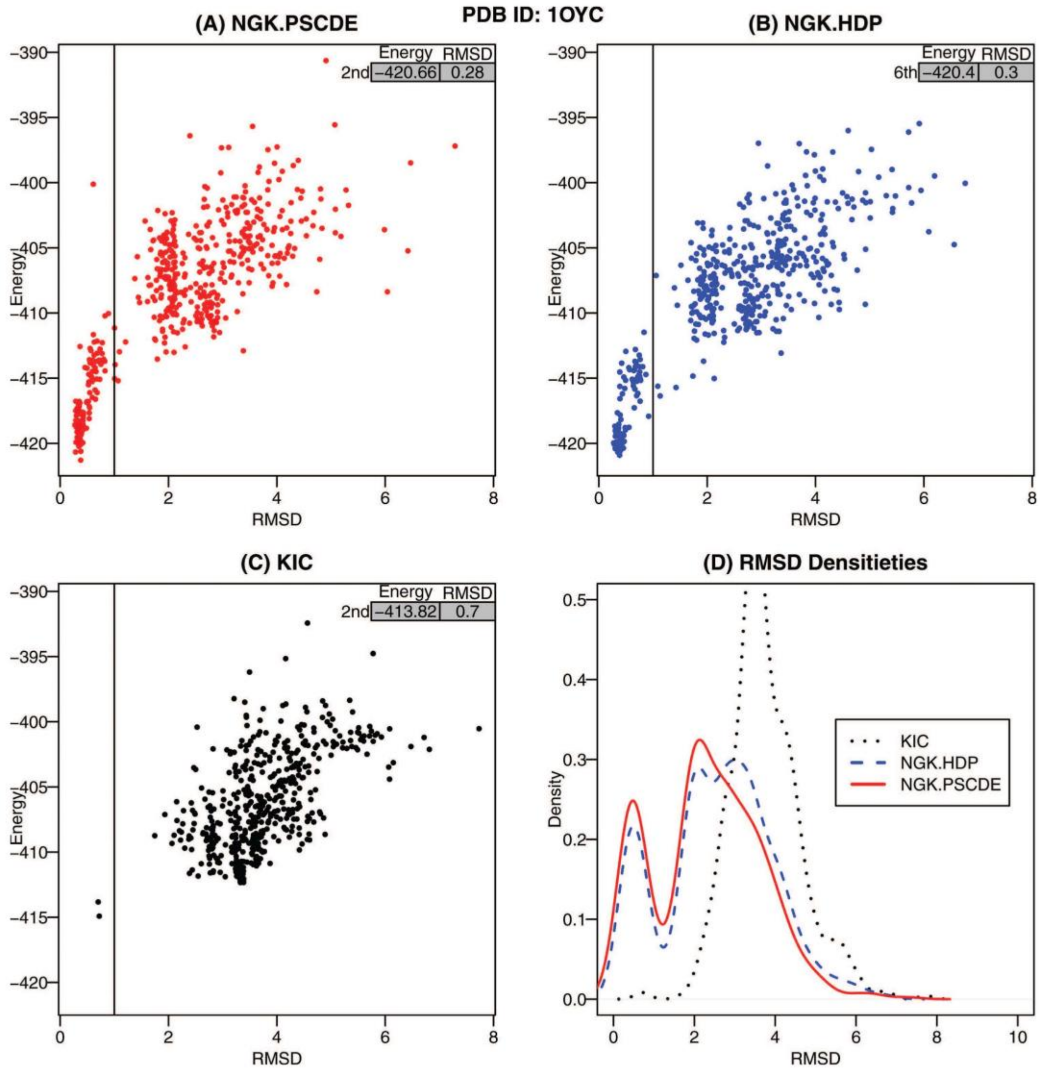
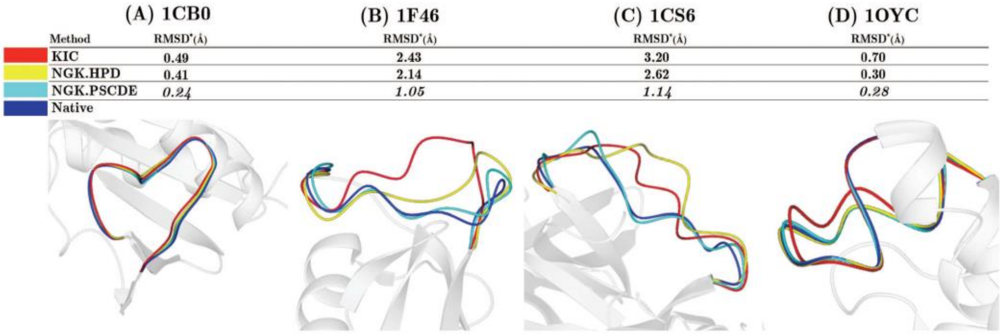


Figure 5: Energy vs. RMSD for "1OYC" based on 500 loops predicted by NGK.PSCDE (A), NGK.HDP (B), and KIC (C). Kernel density estimates of the RMSD obtained for the predicted loops by three different methods (panel D).



	(A) 1CB0	(B) 1F46	(C) 1CS6	(D) 1OYC
Method	RMSD*(Å)	RMSD*(Å)	RMSD*(Å)	RMSD*(Å)
KIC	0.49	2.43	3.20	0.70
NGK.HPD	0.41	2.14	2.62	0.30
NGK.PSCDE	*0.24*	*1.05*	*1.14*	*0.28*
Native				

Figure 6: The best model fit (in terms of RMSD*) predicted by the three methods for "1CB0" (A), "1F46" (B), "ICS6" (C), and "1OYC" (D).

Author's Footnote:

Mehdi Maadooliat is Assistant Professor (Email: mehdi@mscs.mu.edu), Department of Mathematics, Statistics and Computer Science, Marquette University, Wisconsin, 53201-1881, USA. Lan Zhou is Associate Professor (Email: lzhou@stat.tamu.edu), Jianhua Z. Huang is Professor (Email: jianhua@stat.tamu.edu), Department of Statistics, Texas A&M University, Texas, 77843-3143, USA. Seyed Morteza Najibi is Assistant Professor (Email: adius12@gmail.com), Department of Statistics, Persian Gulf University, Bushehr, 75169, Iran. Xin Gao (Email: xin.gao@kaust.edu.sa) is Assistant Professor, Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal, 23955-6900, Saudi Arabia. Part of Maadooliat's work was done when he was a Postdoctoral Fellow at Institute of Applied Mathematics and Computational Science, Texas A&M University. Part of Najibi's work was done when he was a visiting scholar at Department of Mathematics, Statistics and Computer Science, Marquette University. The first two authors, Maadooliat and Zhou, made equal contributions to the paper. Corresponding authors: Xin Gao and Jianhua Huang.