Longitudinal Deep Kernel Gaussian Process Regression

Junjie Liang, Yanting Wu, Dongkuan Xu, Vasant G Honavar
Pennsylvania State University
{jul672, yxw514, dux19, vhonavar}@psu.edu

Abstract

Gaussian processes offer an attractive framework for predictive modeling from longitudinal data, i.e., irregularly sampled, sparse observations from a set of individuals over time. However, such methods have two key shortcomings: (i) They rely on ad hoc heuristics or expensive trial and error to choose the effective kernels, and (ii) They fail to handle multilevel correlation structure in the data. We introduce Longitudinal deep kernel Gaussian process regression (L-DKGPR) to overcome these limitations by fully automating the discovery of complex multilevel correlation structure from longitudinal data. Specifically, L-DKGPR eliminates the need for ad hoc heuristics or trial and error using a novel adaptation of deep kernel learning that combines the expressive power of deep neural networks with the flexibility of non-parametric kernel methods. L-DKGPR effectively learns the multilevel correlation with a novel additive kernel that simultaneously accommodates both time-varying and the time-invariant effects. We derive an efficient algorithm to train L-DKGPR using latent space inducing points and variational inference. Results of extensive experiments on several benchmark data sets demonstrate that L-DKGPR significantly outperforms the state-of-the-art longitudinal data analysis (LDA) methods.

Introduction

Longitudinal studies, which involve repeated observations, taken at irregularly spaced time points, for a set of individuals over time, are ubiquitous in many applications, e.g., in health, cognitive, social, and economic sciences. Such studies are used to identify the time-varying as well as the time-invariant factors associated with a particular outcome of interest, e.g., health risk (Hedeker and Gibbons 2006), air pollution (Tang et al. 2020; Hsieh et al. 2020), etc. Longitudinal data typically exhibit longitudinal correlation (LC), i.e., correlations among the repeated observations of a given individual over time; and cluster correlation (CC), i.e., correlations among observations across individuals, e.g., due to the characteristics that they share among themselves e.g., age, demographics factors; or both, i.e., multilevel correlation (MC). In general, the structure of MC can be complex and a priori unknown. Failure to adequately account for the structure of MC in predictive modeling from longitudinal data can lead to misleading statistical inferences (Gibbons and Hedeker 1997; Liang et al. 2020). It can be non-trivial to choose a suitable correlation structure that reflects the correlations present in the data. The relationships between the covariates and outcomes of interest can be highly complex and non-linear. Furthermore, modern applications often call for LDA methods that scale gracefully with increasing number of variables, the number of individuals, and the number of longitudinal observations per individual.

Related Work

Conventional LDA Methods LDA methods have been extensively studied for decades (Hedeker and Gibbons 2006; Verbeke et al. 2014). Conventional LDA methods fall into two broad categories: (i) marginal models and (ii) conditional models. Marginal models rely on assumptions about the marginal association among the observed outcomes. The generalized estimating equations (GEE) (Liang and Zeger 1986), where a working correlation matrix is specified to model the marginal association among the observed outcomes, offer an example of marginal models. The parameters of marginal models are often shared by all individuals in the population, yielding population-averaged effects or fixed effects. Conditional models on the other hand avoid directly specifying the full correlation matrix by distinguishing random effects, i.e., parameters that differ across individuals, from fixed effects, so as to estimate the individual parameters conditioned on the population parameters. A popular example of conditional models is the generalized linear mixed-effects models (GLMM) (McCulloch 1997). Despite much work on both marginal and conditional models (Fitzmaurice, Laird, and Ware 2012; Wang 2014; Xiong, Kim, and Singh 2019; Liang et al. 2020), many of the challenges, especially the choice of correlation structure, and the selection of variables to model random versus fixed effects, and the scalability of the methods remain to be addressed.

Non-parametric LDA Methods More recently, there is a growing interest in Gaussian processes (GP) (Quintana et al. 2016; Cheng et al. 2019; Wang et al. 2019) for LDA because of their advantages over conventional parametric LDA methods: (i) GP make fewer assumptions about the underlying data distribution by dispensing with the need to choose a particular parametric form of the nonlinear predictive model;
we improve the inducing points technique by introducing
(ii) GP permit the use of parameterized kernels to model the
correlation between observed outcomes, to cope with data
sampled at irregularly spaced time points, by interpolating
between samples; (iii) The interpretability of GP models can be
enhanced by choosing modular kernels that are composed of
simpler kernels that capture the shared correlation struc-
ture of a subset of covariates, and (iv) GP models can flexibly
account for both longitudinal and cluster correlations in the
data. For example, Cheng et al. (2019) utilize an additive ker-
nel for Gaussian data and employ a step-wise search strategy
to select the kernel components and covariates that optimize
the predictive accuracy of the model. Timonen et al. (2019)
consider a heterogeneous kernel to model individual-specific
(random) effects in the case of non-Gaussian data. Despite
their advantages, existing GP based approaches to LDA suf-
f er from several shortcomings that limit their applicability in
real-world settings: (i) The choice of an appropriate kernel
often involves a tedious, often expensive and unreliable, pro-
cess of trial and error (Rasmussen 2003) or ad hoc heuristics
for identifying a kernel or selecting a subset of kernels from
a pool of candidates (Cheng et al. 2019). (ii) Suboptimal
choice of kernels can fail to adequately model the complex
MC structure in the data. (iii) They do not scale to thousands
of covariates and/or millions of data points that are common
in modern LDA applications.

Overview of Contributions
A key challenge in predictive modeling of longitudinal data
has to do with modeling the complex correlation structure in
the data. We posit that the observed correlation structure is in-
duced by the interactions between time-invariant, individu-
specific effects, and time-varying population effects. Hence,
we can divide the task of predictive modeling from longi-
dinal data into two sub-tasks: (i) Given an observed data
set, how do we estimate the time-varying and time-invariant
effects and the correlation structure present in the data? (ii)
Given the correlation structure, how do we predict as yet unob-
seved, e.g., future outcomes?

We introduce Longitudinal deep kernel Gaussian process
regression (L-DKPR) to fully automate the discovery of
complex multilevel correlation structure from longitudinal
data. L-DKPR inherits the attractive features of GP while
overcoming their key limitations. Specifically, L-DKPR
eliminates the need for ad hoc heuristics or trial and error by
using a deep kernel learning method (Wilson et al. 2016a) that
combines the expressive power of deep neural networks with
the flexibility of non-parametric kernel methods. L-DKPR
extends (Wilson et al. 2016a) by introducing a novel addi-
tive kernel that includes two components, one for modeling
the time-varying (fixed) effects and the other for modeling the
time-invariant (random) effects, to compensate for the
multilevel correlation structure in longitudinal data. To en-
hance the effectiveness and efficiency of model inference,
we improve the inducing points technique by introducing
inducing points directly in the latent space. Our formula-
tion permits a tractable ELBO, which not only eliminates
the need for Monte Carlo sampling, but also dramatically
reduces the number of parameters and iterations needed to
achieve state-of-the-art regression performance.

Preliminaries
Notations. We denote a longitudinal data set by \( D = (X, y) \),
where \( X \in \mathbb{R}^{N \times P} \) is the covariate matrix and \( y \in \mathbb{R}^{N \times 1} \)
is the vector of measured outcomes. We denote a row in \( X \)
by \( x_{it} \), with \( i, t \) indexing the individual and the time for the
observation respectively. Because the observations for each
individual are irregularly sampled over time, we have for each
individual \( i \), a submatrix \( X_i \in \mathbb{R}^{N_i \times P} \subset X \), where \( N_i \) is
the number of observations available for the individual \( i \). If we
 denote by \( I \) be the number of individuals in \( D \), the covariate
matrix \( X \) is given by \( X^\top = (X_1^\top, \ldots, X_I^\top)^\top \). Accordingly,
the outcomes \( y \) are given by \( y^\top = (y_1^\top, \ldots, y_I^\top)^\top \).

Gaussian Process. A Gaussian process (GP) is a stochas-
tic process, i.e., a distribution over functions or an infinite
collection of (real-valued) random variables, such that any
finite subset of random variables has a multivariate Gauss-
ian distribution (Williams and Rasmussen 2006). A kernel
describes the covariance of the random variables that make
up the GP. More precisely, if a function \( f: \mathcal{X} \to \mathbb{R} \) has a GP prior \( f \sim GP(\mu, k_\gamma) \)
where \( \mu \) is the mean function and \( k_\gamma(\cdot, \cdot) \) is a (positive semi-definite) kernel function parameter-
ized by \( \gamma \), then any finite collection of components of \( f \) (denoted as \( f \)) has a multivariate Gaussian distribution
\( f(X) \sim \mathcal{N}(\mu(X), K_{XX}) \), where \( \mu(X) \) is the mean vector,
and \( (K_{XX})_{ij} = k_\gamma(x_i, x_j) \) is the covariance matrix. In the
regression setting, the function \( f \) is treated as an unobserved
signal linked to the outcomes through a (typically Gaussian)
likelihood function, such that \( (y|f) \sim \mathcal{N}(f, \sigma_i^2) \).

Additive GP is a special case of GP where unobserved
signal is expressed as the sum of \( J \) independent signal com-
ponents, i.e., \( f = \sum_{j=1}^J \alpha_j f(j) \), where \( \alpha = \{\alpha_j\}_{j=1}^J \)
are the coefficients associated with the individual compo-
ents (Duvenaud, Nickisch, and Rasmussen 2011). In prac-
tice, each signal component is computed on a (typically small
(Cheng et al. 2019; Timonen et al. 2019)) subset of the ob-
erved covariates in \( x \). The fact that each signal component
has a GP prior ensures that the joint signal \( f \) is also GP.
Additive GP allows using different kernel functions for dif-
ferent signal components, so to model the shared correlation
structure of a subset of covariates, thus enhancing the inter-
pretability of the resulting GP. More importantly, it permits
the time-varying and time-invariant effects to be modeled
different kernel functions, which is especially attractive
in modeling longitudinal data.

Longitudinal Deep Kernel Gaussian Process
Regression
Predictive modeling from longitudinal data typically requires
solving two sub-problems: (i) Extracting the time-varying
and time-invariant information from the observed data to
estimate the underlying multilevel correlation structure; and
(ii) using the estimated correlation structure to predict the
future outcomes. In what follows, we describe our solutions
to both sub-problems.
Time-varying Kernel

\[ \begin{align*}
&\mathbf{x}_{it} \quad \text{Encoder} \quad \mathbf{z}_{it}^0 \quad \rightarrow \quad k_{SE} \left( \mathbf{s}_{it}^0, \mathbf{s}_{it}^0 \right) \\
&\mathbf{x}_{ij} \quad \mathbf{s}_{it}^0 \quad \mathbf{s}_{ij}^0 \quad \rightarrow \quad k_{SE} \left( \mathbf{s}_{it}^0, \mathbf{s}_{ij}^0 \right)
\end{align*} \]

Time-invariant Kernel

\[ \begin{align*}
&\mathbf{s}_t \quad \text{Individual} \quad \mathbf{z}_t^0 \quad \rightarrow \quad k_{SE} \left( \mathbf{s}_t^0, \mathbf{s}_j^0 \right) \\
&\mathbf{s}_i \quad \mathbf{z}_t^0 \quad \mathbf{z}_j^0 \quad \rightarrow \quad k_{SE} \left( \mathbf{s}_i^0, \mathbf{s}_j^0 \right)
\end{align*} \]

Figure 1: Structure of the deep kernels.

Modeling the Multilevel Correlation using Deep Kernels

Recall that longitudinal data exhibit complex correlations arising from the interaction between time-varying effects and time-invariant effects. Hence, we decompose the signal function \( f \) into two parts, i.e., \( f^{(v)} \) which models the time-varying effects and \( f^{(i)} \), which models the time-invariant effects. The result is a probabilistic model that can be specified as follows:

\[
(y|f) \sim \mathcal{N}(f, \sigma^2 I) \quad \quad f = \alpha^{(v)} f^{(v)} + \alpha^{(i)} f^{(i)} \\
\mathbb{E}[f|X] \sim \mathcal{N}(\mu^{(v)}(X), k^{(v)}(X, X)) \quad \quad \mathbb{E}[f^{(i)}|X] \sim \mathcal{N}(\mu^{(i)}(X), k^{(i)}(X, X))
\]

We denote the kernel parameters for time-varying effects and time-invariant effects respectively by \( \gamma \) and \( \phi \). The mean functions \( \mu^{(v)}, \mu^{(i)} \), if unknown, can be estimated from data. In this study, without loss of generality, following (Williams and Rasmussen 2006; Wilson et al. 2016a,b; Cheng et al. 2019; Timonen et al. 2019), we set \( \mu^{(v)} = \mu^{(i)} = 0 \). Assuming that \( f^{(v)} \) and \( f^{(i)} \) are conditionally independent given \( X \), we can express the joint signal distribution \( f \) as follows:

\[
(f|X) \sim \mathcal{N} \left( 0, k_\theta = \alpha^{(v)}^2 k^{(v)} + \alpha^{(i)}^2 k^{(i)} \right)
\]

Time-varying Kernel \( k^{(v)}_\gamma \). We introduce a time-varying kernel to capture the longitudinal correlation in the data. The structure of our time-varying kernel \( k^{(v)}_\gamma \) is shown in Figure 1 (top). Let \( e_\gamma : \mathcal{X} \rightarrow S^{(v)} \subset \mathbb{R}^{D_v} \) be a non-linear encoder function given by a deep architecture parameterized by \( \gamma \). Given a pair of data points \( \mathbf{x}_{it}, \mathbf{x}_{ij} \), where \( i, j \) index the individuals and \( t, q \) index the time-dependent observations, the time-varying kernel is given by:

\[
k^{(v)}_\gamma(x_{it}, x_{ij}) = k_{SE}(e_\gamma(x_{it}), e_\gamma(x_{ij}))
\]

with \( k_{SE} \) denoting the squared exponential kernel (Williams and Rasmussen 2006). Note that SE kernel is based on Euclidean distance, which is not a useful measure of distance in the high dimensional input space (Aggarwal, Hinneburg, and Keim 2001). Hence, we use a deep neural network (Goodfellow, Bengio, and Courville 2016), specifically, a nonlinear encoder to map the input space to a low-dimensional latent space and then apply the SE kernel to the latent space.

Time-invariant Kernel \( k^{(i)}_\phi \). We introduce a time-invariant kernel to capture cluster correlation, i.e., time-invariant correlations among individuals that share similar characteristics. The structure of time-invariant kernel is shown in Figure 1 (bottom). Let \( i(x_{it}) = i \) be a mapping function that identifies the individuals, and \( g_\phi : \mathcal{X} \rightarrow S^{(i)} \subset \mathbb{R}^{D_i} \) be an embedding function that maps each individual to a vector in the latent space. Then for any pair of data samples \( x_{it}, x_{jq} \), with arbitrary observation indices, the time-invariant kernel is given by:

\[
k^{(i)}_\phi(x_{it}, x_{jq}) = k_{SE}(g_\phi \circ i(x_{it}), g_\phi \circ i(x_{jq}))
\]

Learning L-DKGPR Model from Data

We now proceed to describe how to efficiently learn an L-DKGPR model and use it to make predictions. Because of space constraints, the details of the derivations are relegated to the Appendix1.

Model Inference. Our approach to efficiently learning an L-DKGPR model draws inspiration from (Wilson et al. 2016b), to greatly simplify the computation of the GP posterior by reducing the effective number of rows in \( X \), from \( N \) to \( M \) \((M \ll N)\), where \( M \) is the number of inducing points. However, unlike (Wilson et al. 2016b), which uses inducing points in the input space, we use inducing points in a low-dimensional latent space. Let \( Z = \{z_m\}_{m=1}^M \) be the collection of inducing points, and \( \mathbf{u} \) their corresponding signal. The kernel computations based on the inducing points are given by:

\[
k^{(v)}_\gamma(z_i, z_j) = k_{SE}(e_\gamma(z_i), z_j) \quad \quad k^{(i)}_\phi(z_i, z_j) = k_{SE}(z_i, z_j)
\]

Replacing inducing points in the input space with those in a low-dimensional latent space offers several advantages. First, we no longer need to use the encoder network \( e_\gamma(\cdot) \) to transform the inducing points \( Z \), thus increasing the computational efficiency of the model. Second, the latent space is dense, continuous, and usually is of much lower dimension than the input space \((D_o \ll P)\). The resulting parameterization of inducing points directly in the latent space, results in a reduction in the number of parameters that describe the inducing points \((i.e., Z)\) from \( O(MP) \) to \( O(MD_o) \). Third, the latent space simplifies the optimization of L-DKGPR, especially when the input space is defined by heterogeneous data types subject to domain-specific constraints, because the latent space is always continuous regardless the constraints in the input space. We define \( i(z_m) = i + m \) to distinguish the inducing points from the input data. We can now express the joint signal distribution as follows:

\[
(f, \mathbf{u}|X, Z) \sim \mathcal{N} \left( \begin{bmatrix} 0 & 0 \\ \mathbf{K}_{XX} & \mathbf{K}_{XZ} \end{bmatrix} \right)
\]

Therefore, the signal distribution conditioned on the inducing points is given by:

\[
(f|u, X, Z) \sim \mathcal{N}(\mathbf{K}_{XZ} K_{ZZ}^{-1} u, \mathbf{K}_{XX} - \mathbf{K}_{XZ} K_{ZZ}^{-1} K_{XZ}^T)
\]

1Appendix can be found at https://github.com/unjieliang672/L-DKGPR/blob/master/Appendix_LDKGPR.pdf

8558
Let \( \Theta = \{ \alpha^{(v)}, \alpha^{(i)}, \gamma, \phi, \sigma^2, Z \} \) be the model parameters. We aim to learn the parameters by maximizing the log of marginal likelihood \( p(y|X, Z) \). By assuming a variational posterior over the joint signals \( q(f|u(X, Z) = q(u|X, Z)p(f|u, X, Z) \), we can derive the evidence lower bound (see e.g., (Wilson et al. 2016b)):

\[
\mathcal{L} = \mathbb{E}_{q(u|X,Z)}[\log p(y|f)] - KL[q(u|X,Z)||p(u|X,Z)] \tag{6}
\]

We define the proposal posterior \( q(u|X,Z) = \mathcal{N}(\mu_q, L_q \sigma^2) \). To speed up the computation, we follow the deterministic training conditional (DTC) (Seeger, Williams, and Lawrence 2003), an elegant sparse method for accurate computation of the Gaussian process posterior by retaining exact likelihood coupled with an approximate posterior (Liu et al. 2020), rendering \( q(u|X,Z) \) deterministic during the training phase. Letting \( A = K_{XZ} K_{ZZ}^{-1} \) and reparameterizing \( u = \mu_q + L_q \epsilon \) with \( \epsilon \sim N(0, 1) \), we can rewrite the ELBO in closed form:

\[
2\mathcal{L} = -2N \log \sigma^2 - \sigma^2 (\|y\|^2_2 - 2y^T A \mu_q + \|A \mu_q\|^2_2) + \|L_q \|_2^2 - \log |K_{ZZ}| + 2 \log |L_q| + M
\]

\[
- \text{tr} (K_{ZZ} L_q L_q^T) - \mu_q^T K_{ZZ}^{-1} \mu_q \tag{7}
\]

where \( 1 \) is a column vector of ones. We can then compute the partial derivatives of \( \mathcal{L} \) w.r.t. the parameters of the proposal posterior \( q(u|X,Z) \) (i.e., \( \{ \mu_q, L_q \} \)), yielding:

\[
\frac{\partial \mathcal{L}}{\partial \mu_q} = \frac{1}{\sigma^2} (-A^T y + A^T A \mu_q) + K_{ZZ}^{-1} \mu_q = 0 \tag{8}
\]

\[
\frac{\partial \mathcal{L}}{\partial L_q} = \frac{1}{\sigma^2} A^T A L_q 11^T + (L_q^T - K_{ZZ}^{-1} L_q) = 0 \tag{9}
\]

Solving the above equations gives:

\[
\mu_q = \sigma^2 K_{ZZ} B K_{XX}^T y \tag{10}
\]

\[
L_q (1 + 11^T) = K_{ZZ} B K_{XX} \tag{11}
\]

with \( B = (K_{XX} + \sigma^2 K_{XX}^T K_{XX})^{-1} \). To solve the triangular matrix \( L_q \) from (11), we first compute the Cholesky decomposition of \( 1 + 11^T = CC^T \) and \( K_{ZZ} B K_{ZZ} = UU^T \). To simplify both side of (11) to \( L_q C = U \). \( L_q \) can then be solved by exploiting the triangular structure on both side with

\[
L_{i,i-k} = U_{i,i-k} - \sum_{j=0}^{k-1} L_{i,j} C_{i-j,i-k} \tag{12}
\]

where \( k = 0, \ldots, i-1 \), \( L_{i,j} \) is a shorthand for \( [L_q]_{i,j} \). We separate the model parameters into two groups, i.e., parameters w.r.t. the proposal posterior \( \{ \mu_q, L_q \} \) and the remaining parameters \( \Theta \), and use an EM-like algorithm to update both groups alternatively. The L-DKGPR algorithm is listed in Algorithm 1.

**Prediction.** Given the covariate matrix \( X \) for the test data, the predictive distribution is given by:

\[
p(f|x, x, y, Z) \approx \mathcal{N}(K_{X} x K_{ZZ}^{-1} \mu_q, K_{X} x - K_{X} x Z K_{ZZ}^{-1} K_{X} x) \tag{13}
\]

**Complexity.** The time complexity and space complexity of both inference and prediction are \( \mathcal{O}(NM^2) \) and \( \mathcal{O}(NM) \) respectively, where \( N \) is the number of measured outcomes, and \( M \) the number of inducing points.

---

**Experiments**

We compare L-DKGPR to several state-of-the-art LDA and GP methods on simulated as well as real-world benchmark data. The experiments are designed to answer research questions about accuracy, scalability, and interpretability of L-DKGPR: (RQ1) How does the performance of L-DKGPR compare with the state-of-the-art methods on standard longitudinal regression tasks? (RQ2) How does the scalability of L-DKGPR compare with that of the state-of-the-art longitudinal regression models? (RQ3) Can L-DKGPR reliably recover the rich correlation structure from the data? (RQ4) How do the different components of L-DKGPR contribute to its overall performance? (RQ5) What is the advantage of solving the exact ELBO in (7) compared to solving its original form in (6) using Monte Carlo sampling (Wilson et al. 2016b)?

**Data**

We used one simulated data set and three real-world longitudinal data sets in our experiments:

**Simulated Data.** We construct simulated longitudinal data that exhibit i.e., longitudinal correlation (LC) and multilevel correlation (MC) as follows: The outcome is generated using \( y = f(X) + \epsilon \) where \( f(X) \) is a non-linear transformation based on the observed covariate matrix \( X \) and the residual \( \epsilon \sim N(0, \Sigma) \). To simulate longitudinal correlation, we simply set \( \Sigma \) to a block diagonal matrix with non-zero entries for within-individual observations. To simulate multilevel correlation, we first split the individuals into \( C \) clusters and assign non-zero entries for the data points in the same cluster. Following (Cheng et al. 2019; Timonen et al. 2019), we simulate 40 individuals, 20 observations, and 30 covariates for each individual. We vary the number of clusters \( C \) from \( 2, 5 \).

**Study of Women’s Health Across the Nation (SWAN).** (Sutton-Tyrrell et al. 2005). SWAN is a multi-site longitudinal study designed to examine the health of women during the midlife years. We consider the task of predicting the CESD

---

**Algorithm 1: L-DKGPR**

**Input:** Training set \( S = \{X, y\} \), latent dimension \( D_v, D_i \), number of inducing points \( M \), gradient-based optimizer and its related hyper-parameters (i.e., learning rate, weight decay, mini-batch size), alternating frequency \( T \).

1. Initialize the parameters \( \Theta = \{ \sigma^2, Z, \alpha^{(v)}, \alpha^{(i)}, \gamma, \phi \} \)
2. while Not converged do
   3. Update proposal posterior \( q(u|X,Z) \) according to (10) and (12)
   4. \( t = 0 \)
   5. for \( t < T \) do
      6. Update \( \Theta \) using the input optimizer
      7. \( t = t + 1 \)

---

Details of generation of simulated data and of pre-processing of real-world data are provided in the Appendix.
score, which is used for screening for depression. Similar to (Liang et al. 2020), we define the adjusted CESD score by \( y = \text{CESD} - 15 \), thus \( y \geq 0 \) indicates depression. The variables of interest include aspects of physical and mental health, and demographic factors such as race and income. The resulting data set has 3,300 individuals, 137 variables and 28,405 records.

**General Social Survey (GSS).** (Smith et al. 2015). The GSS data were gathered over 30 years on contemporary American society collected with the goal of understanding and explaining trends and constants in attitudes, behaviors, and attributes. In our experiment, we consider the task of predicting the self-reported general happiness of 4,510 individuals using 1,553 features and 59,599 records. We follow the experimental setup in (Liang et al. 2020), with \( y = 1 \) indicates happy and \( y = -1 \) indicates the opposite.

**The Alzheimer’s Disease Prediction (TADPOLE).** ( Marinescu et al. 2018). The TADPOLE challenge involves predicting the symptoms related to Alzheimer’s Disease (AD) within 1-5 years of a group of high-risk subjects. In our experiment, we focus on predicting the ADAS-Cog13 score using the demographic features and MRI measures (Hippocampus, Fusiform, WholeBrain, Entorhinal, and MidTemp). The resulting data set has 1,681 individuals, 24 variables and 8,771 records.

**Experimental Setup**

To answer RQ1, we use both simulated data and real-world data. To evaluate the regression performance, similar to (Liang et al. 2020), we compute the mean and standard deviation of \( R^2 \) between the actual and predicted outcomes of each method on each data set across 10 independent runs. We use 50\%, 20\%, 30\% of data for training, validation, and testing respectively.

To answer RQ2, we take data from a subset consisting of 50 individuals with the largest number of observations from each real-world data. We record the run time per iteration of each method on both the 50-individual subset and full data set. Because not all baseline methods take advantage of GPU acceleration, we compare the run times of all the methods without GPU acceleration. We report execution failure if a method fails to converge within 48 hours or generates an execution error (Liang et al. 2020).

To answer RQ3, we rely mainly on simulated data since the actual correlation structures underlying the real-world data sets are not known. We evaluate the performance of each method by visualizing the learned correlation matrix and compare it to the ground truth correlation matrix on simulated data. Additionally, we illustrate how the correlation matrix learned by L-DKGPR can provide gain useful insights using a case study with the SWAN data. Results of the case study are included in the Appendix.

To answer RQ4, we compare the performance of L-DKGPR with L-RBF-GPR, a variant that replaces the learned deep kernel with a simple RBF kernel; and L-DKGPR-\( c \), a variant of L-DKGPR without the time-invariant effects.

To answer RQ5, we compare the regression performance and hyper-parameter choices of L-DKGPR solved using Algorithm 1 with the version of L-DKGPR solved using Monte Carlo sampling (Wilson et al. 2016b) on SWAN and GSS data sets.

**Baseline Methods.** We compare L-DKGPR with the following baseline methods: (i) Conventional longitudinal regression models, i.e., GLMM (Bates et al. 2015) and GEE (Inan and Wang 2017); (ii) State-of-the-art longitudinal regression models, i.e., LMLFM (Liang et al. 2020) and LGPR (Timonen et al. 2019); (iii) State-of-the-art Gaussian Process models for general regression, i.e., KISSGP with deep kernel (Wilson et al. 2016b) (we use the same deep structure and in our time-varying kernel) and ODVGP (Salimbeni et al. 2018). Implementation details and hyper-parameter settings of L-DKGPR as well as the baseline approaches are provided in the Appendix.

**Results**

We report the results of our experiments designed to answer the research questions RQ1-RQ4.

**L-DKGPR vs. Baseline Longitudinal Regression Methods.** The results are reported in Table 1 and Table 2 for simulated and real-world data sets respectively. In the case of simulated data, we find that KISSGP, ODVGP, GEE and GLMM fail in the presence of MC with the mean \( R^2 \) being negative (indicative of models containing variables that are not predictive of the response variable). This can be explained by the fact that GEE is designed only to handle pure LC, thus fails to account for CC or MC. While GLMM is capable of handling CC, it requires practitioners to specify the cluster structure responsible for CC prior to model fitting. However, in our experiments, the cluster structure is unknown a priori. Hence it is not surprising that GLMM performs poorly. Though both KISSGP and ODVGP are conceptually able to handle data with complex correlation, they both experience dramatic performance drop when the data exhibit cluster correlation (or time-invariant effects). Moreover, we find that although LMLFM outperforms GLMM and GEE in the presence of MC, its \( R^2 \) is still quite low. This is because LMLFM accounts for only a special case of MC, namely, for CC among individuals observed at the same time points, and not all of the CC present in the data. We find that LGPR performs rather poorly on both simulated and real-world data. This might due to the fact that LGPR obtains the contributions of each variable to the kernel independently before calculating their weighted sum. Though it is possible to incorporate higher-order interactions between variables into LGPR, doing so requires estimating large numbers of interaction parameters, with its attendant challenges, especially when working with small populations. In contrast to the baseline methods, L-DKGPR consistently and significantly outperforms the baselines by a large margin. On the real-world data sets, L-DKGPR outperforms the longitudinal baselines in most of the cases.

**Scalability of L-DKGPR vs. Baseline Methods.** We see from Table 2 that most longitudinal baselines, i.e., LGPR, GLMM, and GEE, fail to process real-world data sets with very large numbers of covariates. Indeed, the computational

\(^3\)Data and codes used in this paper are publicly available at https://github.com/junjie.liang672/L-DKGPR.
Table 1: Regression accuracy $R^2$ (%) comparison on simulated data with different correlation structures.

| Data sets | $N$  | $I$  | $P$ | L-DKGPR | KISSGP | ODVGP | LGPR | LMLFM | GLMM  | GEE    |
|-----------|------|------|-----|---------|--------|-------|------|-------|-------|--------|
| TADPOLE   | 595  | 50   | 24  | 44.0 ± 5.6 | 12.1 ± 10.1 | 9.0 ± 14.1 | -26.1 ± 9.0 | 8.7 ± 5.1 | 50.8 ± 5.5 | -11.4 ± 4.8 |
| SWAN      | 550  | 50   | 137 | 46.8 ± 4.9 | 42.4 ± 4.6 | 29.0 ± 3.1 | -16.6 ± 12.7 | 38.6 ± 4.2 | 40.1 ± 7.7 | 46.4 ± 8.0 |
| GSS       | 1,500| 50   | 1,553| 19.1 ± 3.7 | 12.5 ± 6.3 | -7.6 ± 3.3 | N/A | 15.3 ± 1.4 | N/A | -4.6 ± 3.5 |
| TADPOLE   | 8,771| 1,681| 24  | 64.9 ± 1.4 | 0.6 ± 3.9 | 21.1 ± 1.0 | N/A | 10.4 ± 0.6 | 61.9 ± 1.9 | 17.6 ± 0.7 |
| SWAN      | 28,405| 3,300| 137 | 52.5 ± 0.4 | 20.5 ± 7.6 | 24.9 ± 21.8 | N/A | 48.6 ± 2.0 | N/A | N/A |
| GSS       | 59,599| 4,510| 1,553| 56.9 ± 0.1 | 53.1 ± 0.9 | 15.4 ± 27.0 | N/A | 54.8 ± 2.2 | N/A | N/A |

Table 2: Regression accuracy $R^2$ (%) on real-world data sets. We use ‘N/A’ to denote execution error.

| Data sets  | L-DKGPR | L-DKGPR-v | L-DKGPR-i | L-RBF-GPR |
|------------|---------|-----------|-----------|-----------|
| TADPOLE    | 64.9 ± 1.4 | 13.2 ± 1.1 | 56.3 ± 1.3 | 55.5 ± 2.4 |
| SWAN       | 52.5 ± 0.4 | 29.0 ± 3.2 | 16.7 ± 2.4 | 5.4 ± 1.6 |
| GSS        | 56.9 ± 0.1 | 56.2 ± 0.1 | -0.2 ± 0.2 | -14.1 ± 0.4 |

Table 3: Effect on the regression accuracy $R^2$ (%) of different components of L-DKGPR.

- Ablation Study. Regression accuracy comparison on complete real-world data sets is shown in Table 3. Role of time-invariant component: We see a dramatic drop in regression performance when time-invariant effects are not modeled (L-DKGPR-v) as compared to when they are (L-DKGPR). This result underscores the importance of modeling the time-independent components of LC and CC for accurate modeling of longitudinal data. This task is simplified by the decomposition of the correlation structure into the time-varying and time-invariant components. The time-invariant component is analogous to estimating the mean correlation whereas the time-varying component contributes to the residual. Hence, the decomposition of the correlation structure
The result indicates that under the same hyper-parameter setting, our solver outperforms the sampling solver by a large margin. To ensure similar regression performance, we have to modify the hyper-parameters for the sampling solver by increasing the number of inducing points from 10 to 128 and using about 10 times more training iterations. The result indicates that coping with the variance of the noisy ELBO approximation increases the number of parameters and hence the number of iterations needed.

Effect of the Number of Inducing Points $M$. The number of points provide a trade-off between approximation accuracy and efficiency in sparse GP methods. In this experiment, we vary the number of inducing points $M$ from 5 to 100 on simulated data and record the $R^2$ as shown in Figure 3. We find that when the number of inducing points exceeds a certain threshold, i.e., 10 in all simulated settings, regression performance becomes quite stable, an observation that is supported by our experiments with real-world data as well (results omitted). A theoretical study (Burt, Rasmussen, and Van Der Wilk 2019) points out that when input data are normally distributed, the number of inducing points needed in theory is $M = \lceil \log(1600) \rceil$. However, our experiments show that in practice $M \approx \log N$ inducing points suffice to achieve satisfactory results. We conjecture that this is because instead of drawing the inducing points from a $k$-DPP process from the input data, we optimize representation of the inducing points jointly with the other model parameters, thereby identifying effective inducing points that adequately reflect the variance of the input data. Proving or refuting this conjecture would require a deeper theoretical analysis of L-DKGPR.

**Conclusion**

We have presented L-DKGPR, a novel longitudinal deep kernel Gaussian process regression model that overcomes some of the key limitations of existing state-of-the-art GP regression methods for predictive modeling from longitudinal data. L-DKGPR fully automates the discovery of complex multilevel correlations from longitudinal data. It incorporates a deep kernel learning method that combines the expressive power of deep neural networks with the flexibility of non-parametric kernel methods, to capture the complex multilevel correlation structure from longitudinal data. L-DKGPR uses a novel additive kernel that simultaneously models both time-varying and the time-invariant effects. We have shown how L-DKGPR can be efficiently trained using latent space inducing points and the stochastic variational method. We report results of extensive experiments using both simulated and real-world benchmark longitudinal data sets that demonstrate the superior predictive accuracy as well as scalability of L-DKGPR over the state-of-the-art LDA and GP methods. A case study with a real-world data set illustrates the potential of L-DKGPR as a source of useful insights from complex longitudinal data.

### Table 4: Effect of solving L-DKGPR using Algorithm 1 vs. Monte Carlo sampling.

| Data sets | Solver   | $M$ | Iterations | $R^2$ (%) |
|-----------|----------|-----|------------|-----------|
| SWAN      | Alg. 1   | 10  | 300        | 52.5±0.4  |
|           | Sampling | 10  | 300        | 3.1±0.2   |
|           | Sampling | 128 | 3,000      | 51.4±0.4  |
| GSS       | Alg. 1   | 10  | 300        | 56.9±0.1  |
|           | Sampling | 10  | 300        | 4.5±0.1   |
|           | Sampling | 128 | 3,000      | 55.6±0.1  |

Table 4: Effect of solving L-DKGPR using Algorithm 1 vs. Monte Carlo sampling.
Acknowledgements

This work was funded in part by the NIH NCATS grant UL1 TR002014 and by NSF grants 2041759, 1636795, the Edward Frymoyer Endowed Professorship at Pennsylvania State University and the Sudha Murty Distinguished Visiting Chair in Neurocomputing and Data Science funded by the Pratiksha Trust at the Indian Institute of Science (both held by Vasant Honavar).

References

Aggarwal, C. C.; Hinneburg, A.; and Keim, D. A. 2001. On the surprising behavior of distance metrics in high-dimensional space. In International conference on database theory. 420–434. Springer.

Bates, D.; Mächler, M.; Bolker, B.; and Walker, S. 2015. Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, Articles 67(1): 1–48.

Burt, D.; Rasmussen, C. E.; and Van Der Wilk, M. 2019. Rates of Convergence for Sparse Variational Gaussian Process Regression. In International Conference on Machine Learning, 862–871.

Cheng, L.; Ramchandran, S.; Vatanen, T.; Lietzén, N.; Lahesmaa, R.; Vehtari, A.; and Lähdesmäki, H. 2019. An additive Gaussian process regression model for interpretable non-parametric analysis of longitudinal data. Nature communications 10(1): 1798.

Duvenaud, D.; Nickisch, H.; and Rasmussen, C. E. 2011. Additive Gaussian processes. In Advances in Neural Information Processing Systems, 226–234.

Fitzmaurice, G. M.; Laird, N. M.; and Ware, J. H. 2012. Applied longitudinal analysis, volume 998. John Wiley & Sons.

Gibbons, R. D.; and Hedeker, D. 1997. Random effects probit and logistic regression models for three-level data. Biometrics 52:1527–1537.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. Deep learning. MIT press.

Hedeker, D.; and Gibbons, R. D. 2006. Longitudinal data analysis, volume 451. John Wiley & Sons.

Hsieh, T.-Y.; Wang, S.; Sun, Y.; and Honavar, V. 2020. Explainable Multivariate Time Series Classification: A Deep Neural Network Which Learns To Attend To Important Variables As Well As Informative Time Intervals. arXiv preprint arXiv:2011.11631.

Inan, G.; and Wang, L. 2017. PGEE: An R Package for Analysis of Longitudinal Data with High-Dimensional Covariates. R Journal 9(1): 393–402.

Liang, J.; Xu, D.; Sun, Y.; and Honavar, V. 2020. LMLFM: Longitudinal Multi-Level Factorization Machine. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 4811–4818.

Liang, K.-Y.; and Zeger, S. L. 1986. Longitudinal data analysis using generalized linear models. Biometrika 73(1): 13–22.

Liu, H.; Ong, Y.-S.; Shen, X.; and Cai, J. 2020. When Gaussian process meets big data: A review of scalable GPs. IEEE Transactions on Neural Networks and Learning Systems.

Marinescu, R. V.; Oxtoby, N. P.; Young, A. L.; Bron, E. E.; Toga, A. W.; Weiner, M. W.; Barkhof, F.; Fox, N. C.; Klein, S.; Alexander, D. C.; et al. 2018. TADPOLE Challenge: Prediction of Longitudinal Evolution in Alzheimer’s Disease. arXiv preprint arXiv:1805.03909.

McCulloch, C. E. 1997. Maximum likelihood algorithms for generalized linear mixed models. Journal of the American statistical Association 92(437): 162–170.

Quintana, F. A.; Johnson, W. O.; Waetjen, L. E.; and B. Gold, E. 2016. Bayesian nonparametric longitudinal data analysis. Journal of the American Statistical Association 111(515): 1168–1181.

Rasmussen, C. E. 2003. Gaussian processes in machine learning. In Summer School on Machine Learning, 63–71. Springer.

Salimbeni, H.; Cheng, C.-A.; Boots, B.; and Deisenroth, M. 2018. Orthogonally decoupled variational gaussian processes. In Advances in neural information processing systems, 8711–8720.

Seeger, M.; Williams, C. K. I.; and Lawrence, N. D. 2003. Fast Forward Selection to Speed Up Sparse Gaussian Process Regression. In Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics.

Smith, T. W.; Marsden, P.; Hout, M.; and Kim, J. 2015. General Social Surveys, 1972–2014. National Opinion Research Center at the University of Chicago.

Sutton-Tyrrell, K.; Wildman, R. P.; Matthews, K. A.; Chae, C.; Lasley, B. L.; Brockwell, S.; Pasternak, R. C.; Lloyd-Jones, D.; Sowers, M. F.; Torrens, J. I.; et al. 2005. Sex hormone–binding globulin and the free androgen index are related to cardiovascular risk factors in multiethnic premenopausal and perimenopausal women enrolled in the Study of Women Across the Nation (SWAN). Circulation 111(10): 1242–1249.

Tang, X.; Yao, H.; Sun, Y.; Aggarwal, C. C.; Mitra, P.; and Wang, S. 2020. Joint Modeling of Local and Global Temporal Dynamics for Multivariate Time Series Forecasting with Missing Values. In Proceedings of the AAAI Conference on Artificial Intelligence, 5956–5963.

Timonen, J.; Mannerström, H.; Vehtari, A.; and Lähdesmäki, H. 2019. An interpretable probabilistic machine learning method for heterogeneous longitudinal studies. arXiv preprint arXiv:1912.03549.

Verbeke, G.; Fieuws, S.; Molenberghs, G.; and Davidian, M. 2014. The analysis of multivariate longitudinal data: A review. Statistical methods in medical research 23(1): 42–59.

Wang, K.; Pleiss, G.; Gardner, J.; Tyree, S.; Weinberger, K. Q.; and Wilson, A. G. 2019. Exact Gaussian processes on a million data points. In Advances in Neural Information Processing Systems, 14648–14659.
Wang, M. 2014. Generalized estimating equations in longitudinal data analysis: a review and recent developments. *Advances in Statistics* 2014.

Williams, C. K.; and Rasmussen, C. E. 2006. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Wilson, A. G.; Hu, Z.; Salakhutdinov, R.; and Xing, E. P. 2016a. Deep kernel learning. In *Artificial Intelligence and Statistics*, 370–378.

Wilson, A. G.; Hu, Z.; Salakhutdinov, R. R.; and Xing, E. P. 2016b. Stochastic variational deep kernel learning. In *Advances in Neural Information Processing Systems*, 2586–2594.

Xiong, Y.; Kim, H. J.; and Singh, V. 2019. Mixed Effects Neural Networks (MeNets) With Applications to Gaze Estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7743–7752.