Modeling Clinical Time Series Using Gaussian Process Sequences

Zitao Liu Lei Wu Milos Hauskrecht

Department of Computer Science, University of Pittsburgh

Motivation

Development of accurate models of complex clinical time series data is critical for **understanding the disease**, its dynamics, and subsequently **patient management** and **clinical decision making**.


Disease understanding


Patient management


Making decision

Goal

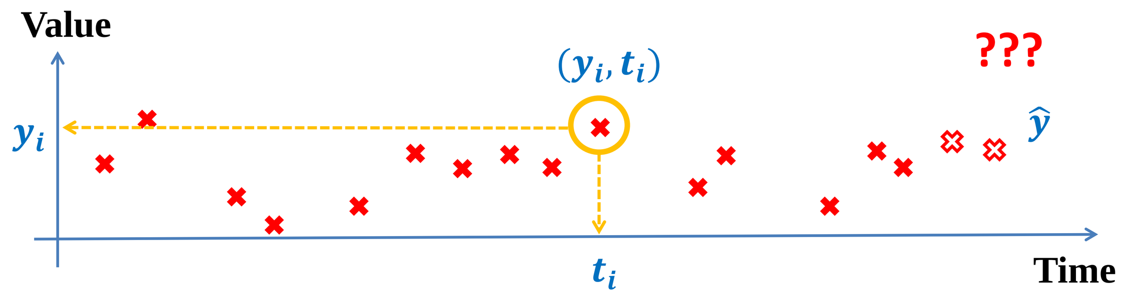
"Develop accurate models of complex clinical time series!"

Specifically, a prediction model that can:

1. Handle missing values
2. Deal with irregular time sampling intervals
3. Make accurate long term predictions

Problem Statement

We define the time series prediction/regression function for clinical time series as: $g : \mathbf{Y}_{\text{obs}} \times t \rightarrow \mathbf{y}$ where $\mathbf{Y}_{\text{obs}}$ is a sequence of past observation-time pairs $\mathbf{Y}_{\text{obs}} = (\mathbf{y}_i, t_i)_{i=1}^{n}$ such that, $0 < t_i < t_{i+1}$, $\mathbf{y}_i$ is a p-dimensional observation vector made at time (t_i), and n is the number of past observations; and $t > t_n$ is the time at which we would like to predict the observation $\mathbf{y}$. Irregularly sampled, $t_{i+1} - t_i \neq t_i - t_{i-1}$.



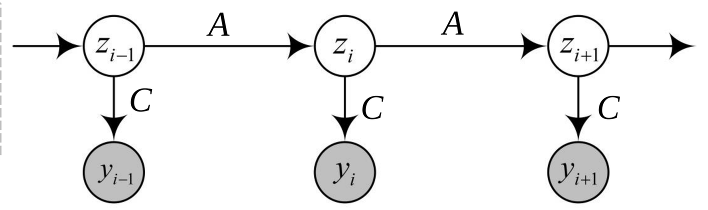
Background

- **Linear Dynamical System (LDS)**

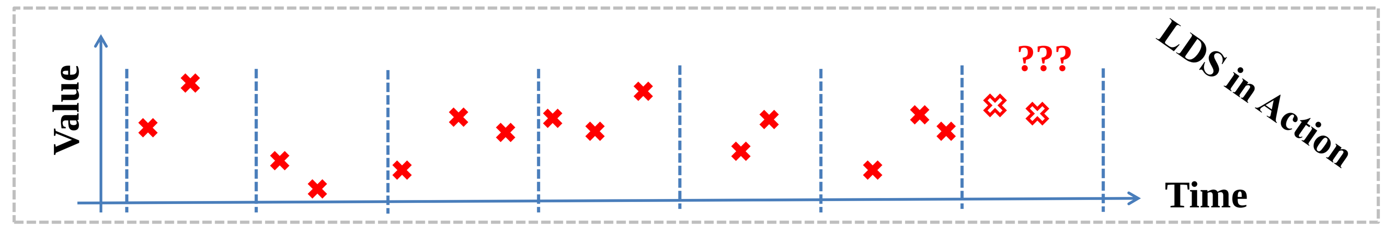
$$\mathbf{z}_{t+1} = A\mathbf{z}_t + \mathbf{w}_t \qquad \mathbf{y}_t = C\mathbf{z}_t + \mathbf{v}_t$$

$$\mathbf{z}_1 \sim \mathcal{N}(\pi_1, V_1),\ \mathbf{w}_t \sim \mathcal{N}(0, Q),\ \mathbf{v}_t \sim \mathcal{N}(0, R)$$

$$p(\mathbf{z}_{t+1} \mid \mathbf{z}_t) = \mathcal{N}(A\mathbf{z}_t, Q), \quad p(\mathbf{y}_t \mid \mathbf{z}_t) = \mathcal{N}(C\mathbf{z}_t, R)$$



Y – time series of observations;
Z – hidden states driving the dynamics.



Background (con't)

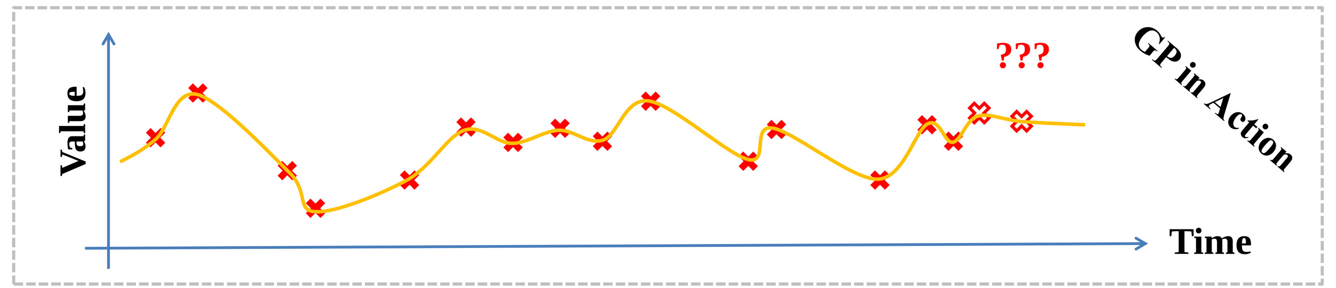
- **Gaussian Process (GP)**

GP is an extension of a multivariate Gaussian to **<u>distributions over functions</u>**. Defined by two components: $\mathcal{GP}(m(x), k(x, x'))$.

- Mean function: $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$
- Covariance function: $K(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]$

GP regression equations:

- Estimated Mean ($\bar{f}_*$): $K(x_*, \mathbf{x})\left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} \mathbf{y}$
- Estimated Covariance ($Cov(f_*)$): $K(x_*, x_*) - K(x_*, \mathbf{x})\left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} K(x_*, \mathbf{x})$

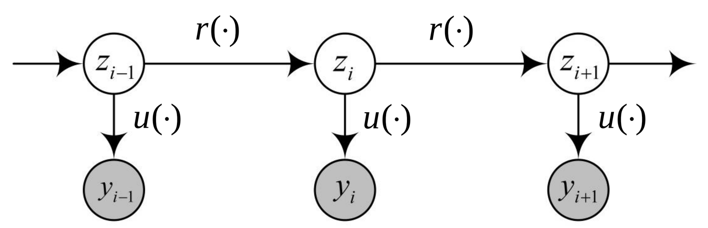


- **Discrete non-linear model (GPIL)**

$$\mathbf{z}_{t+1} = r(\mathbf{z}_t) + \mathbf{w}_t \qquad \mathbf{y}_t = u(\mathbf{z}_t) + \mathbf{v}_t$$

$$\mathbf{z}_1 \sim \mathcal{N}(\pi_1, V_1),\ \mathbf{w}_t \sim \mathcal{N}(0, Q),\ \mathbf{v}_t \sim \mathcal{N}(0, R)$$

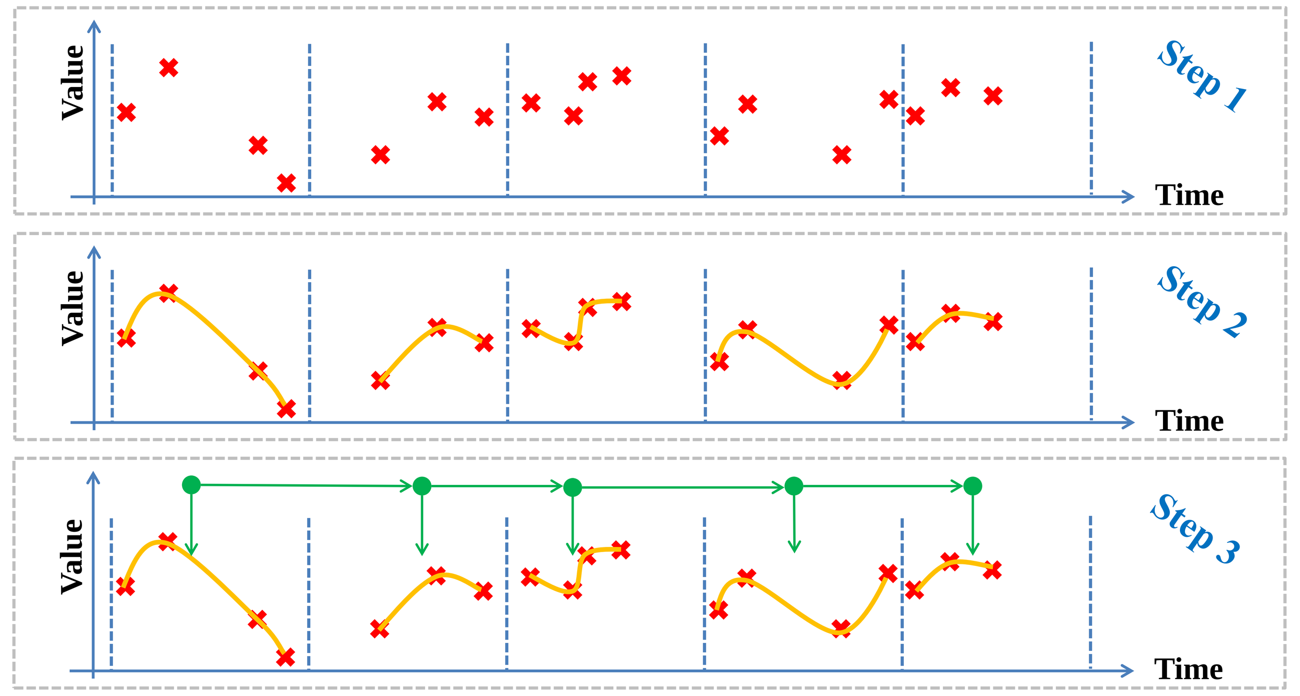
$r(\cdot)$ – unknown transition function;
$u(\cdot)$ – unknown measurement function.



Y – time series of observations;
Z – hidden states driving the dynamics.

State Space Gaussian Process

- **Idea Illustration**



State Space Gaussian Process (con't)

- **State Space Gaussian Process(SSGP) Model**

We consider the Gaussian process *q*(**t**) with the mean function formed by a combination of a fixed set of basis functions with coefficients, β:

$$q(\mathbf{t}) = f(\mathbf{t}) + \mathbf{h}(\mathbf{t})^T \boldsymbol{\beta}, \quad f(\mathbf{t}) \sim \mathcal{GP}_f(0, K(\mathbf{t}, \mathbf{t}'))$$

In this definition, *f*(**t**) is a zero mean GP , *h*(**t**) denotes a set of fixed basis functions, for example, $h(\mathbf{t}) = (1, t, t^2, \ldots)$, and β is a Gaussian prior, $\beta \sim \mathcal{N}(\mathbf{b}, I)$. Therefore, *q*(**t**) is another GP process, defined by:

$$q(\mathbf{t}) \sim \mathcal{GP}_q(\mathbf{h}(\mathbf{t})^{\mathrm{T}} \mathbf{b}, K(\mathbf{t}, \mathbf{t}') + \mathbf{h}(\mathbf{t})^T \mathbf{h}(\mathbf{t}'))$$

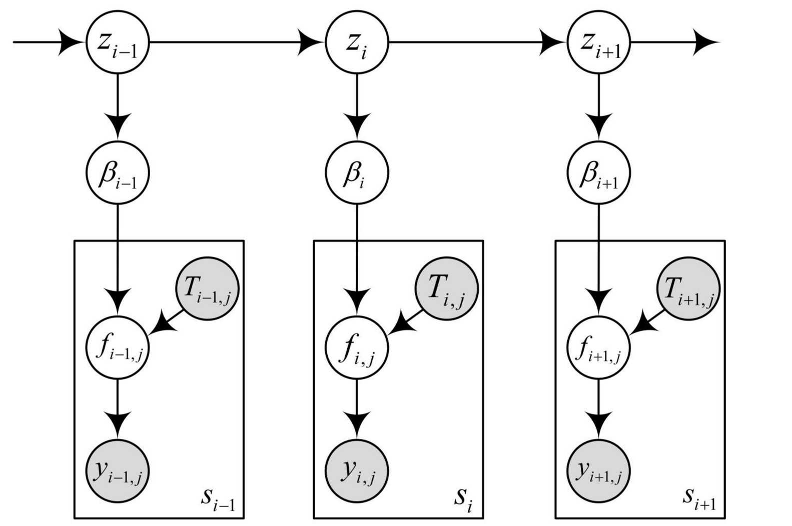


Figure 1. Graphical representation of the state-space Gaussian process model. Shaded nodes $y_{i,j}$ denote (irregular) observations and shaded nodes $T_{i,j}$ denote times associated with each observation. Each rectangle (plate) corresponds to a window, which is associated with its own local GP. S_i is the number of observations in each window. $f_{i,j}$ is Gaussian field.

Joint distribution: $p(D) = p(\mathbf{z}, \boldsymbol{\beta}, \mathbf{Y}) = p(\mathbf{z}_1) \prod_{i=2}^{m} p(\mathbf{z}_i \mid \mathbf{z}_{i-1}) \prod_{i=1}^{m} (\boldsymbol{\beta}_i \mid \mathbf{z}_i) \prod_{i=1}^{m} \prod_{j=1}^{s_i} p(\mathbf{y}_{i,j} \mid \boldsymbol{\beta}_i)$

- **Learning**

Parameter Set: $\Omega = \{\Theta, \{\boldsymbol{\beta}_i\}, A, C, R, Q, \pi_1, V_1\}$ (Θ denotes covariance function parameters)

- Learn Θ: gradient based methods($\frac{\partial \log p(\mathbf{Y} \mid \Theta)}{\partial \Theta} = -\frac{1}{2} \mathrm{Tr}\left[K^{-1} \frac{\partial K}{\partial \Theta}\right] + \frac{1}{2} \mathbf{Y}^T K^{-1} \frac{\partial K}{\partial \Theta} K^{-1} \mathbf{Y}$)
- Learn $\Omega \backslash \Theta$: EM algorithm with $\mathcal{Q} = \mathbb{E}_{\boldsymbol{\beta}, \mathbf{z}}[\log p(\boldsymbol{\beta}, \mathbf{z}, \mathbf{Y})]$

- **Prediction**

To support the prediction inference, we need the following steps:

1. Split $\mathbf{Y}_{\text{obs}}$ and *t* into windows.
2. For windows that do not contain *t*, extract the last values in those windows as **β**s and feed them into *Kalman Filter* algorithms to infer the most recent hidden state $\mathbf{z}_k$ where *k* is the index of the last window that does not contain *t*.
3. Get $\boldsymbol{\beta}_{k+1} = CA\mathbf{z}_k$ from $\mathbf{z}_{k+1} = A\mathbf{z}_k$ and $\boldsymbol{\beta}_{k+1} = C\mathbf{z}_{k+1}$.
4. If *t* is in window *k+1*, use observations $(\mathbf{y}_{k+1}, t_{k+1})$ in window *k+1* and $\boldsymbol{\beta}_{k+1}$ to make the prediction, where $\mathbf{y} = \boldsymbol{\beta}_{k+1} + K(t, t_{k+1}) K^{-1}(t_{k+1}, t_{k+1})(\mathbf{y}_{k+1} - \boldsymbol{\beta}_{k+1})$; otherwise find out the window index *i* where *t* belongs to. The prediction at *t* is $\mathbf{y} = CA^{i-k}\mathbf{z}_k$.

Experiments

- **Data**

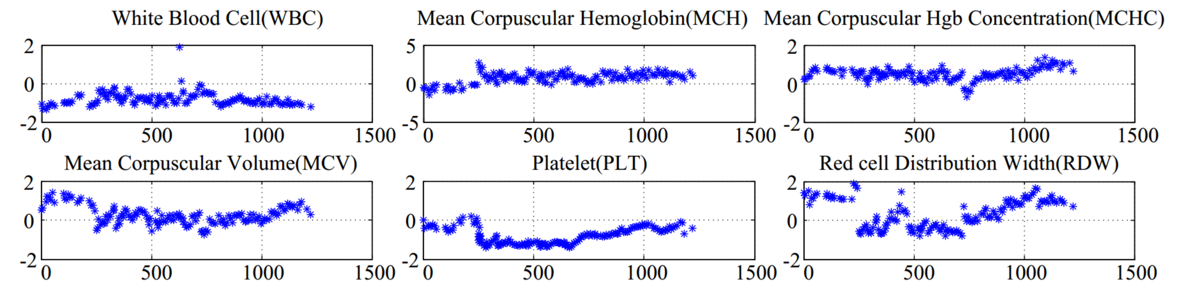


Figure 2. Time series for six tests from the Complete Blood Count(CBC) panel for one of the patients.

- **Choice of Covariance Functions**($K = K_1 + K_2$)
  - Mean Reverting Property: $K_1 = \sigma_1 \exp(\theta_1 |\mathbf{t} - \mathbf{t}'|)$
  - Periodicity: $K_2 = \sigma_2 \exp(\theta_2 \sin^2\left[\frac{\omega}{2\pi}(\mathbf{t} - \mathbf{t}')\right])$

- **Evaluation Metric**

Root Mean Square Error(RMSE):

$$RMSE = \left[n^{-1} \sum_{i=1}^{n} |y_i - y_i|^2\right]^{1/2}$$

- **Results**

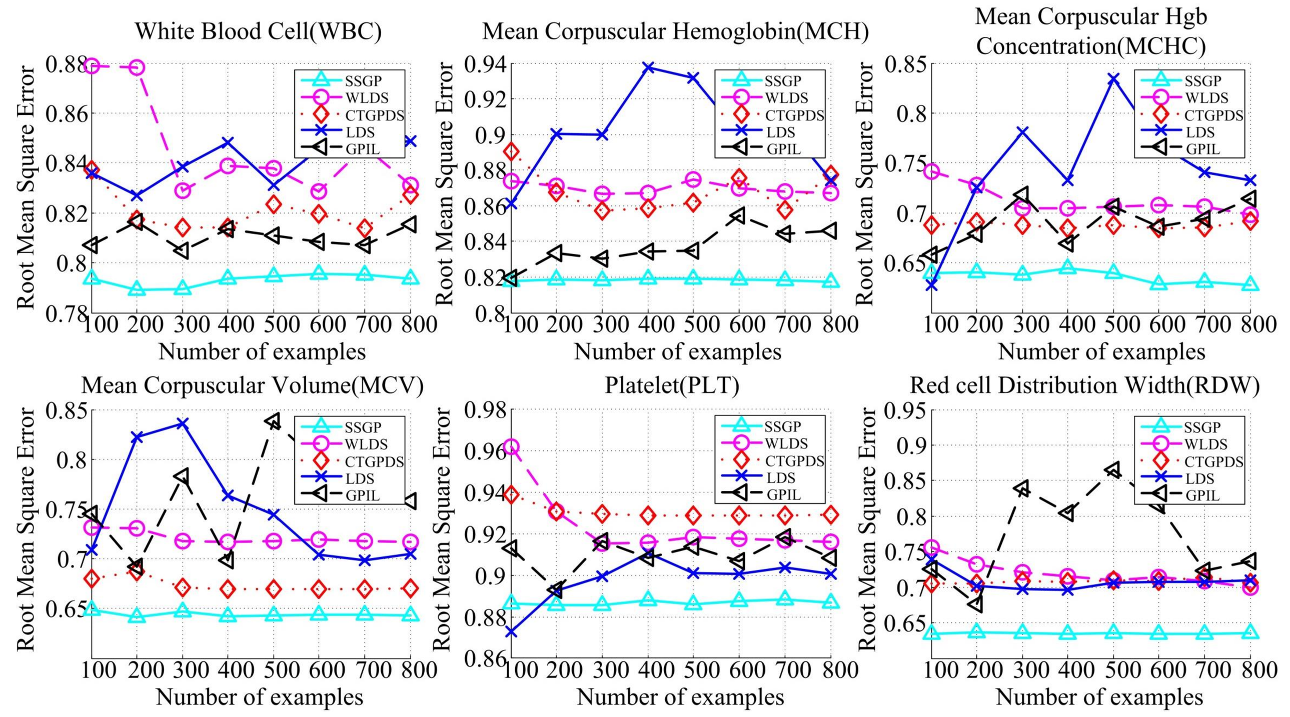


Figure 3. Root Mean Square Error(RMSE) on CBC test samples.

Future Work

- Study and model dependences among multiple time series
- Extend to switching-state and controlled dynamical systems

Acknowledgement

This research work was supported by grants R01LM010019 and R01GM088224 from the National Institutes of Health. Its content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

Reference

- M. Hauskrecht, M. Valko, I. Batal, G. Clermont, S. Visweswaran, and G.F. Cooper, Conditional outlier detection for clinical alerting, in AMIA Annual Symposium Proceedings, 2010, p. 286.
- Carl Edward Rasmussen and Christopher K. I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.
- R. Turner, M.P. Deisenroth, and C.E. Rasmussen, State-space inference and learning with Gaussian processes, in AISTATS, vol. 9, 2010, pp. 868-875.