STA 4103/5107 Computational Methods in Statistics II

Department of Statistics
Florida State University

Class 14
February 23, 2017

Midterm

- **Midterm Project: (Out: Thursday, 2/23)**
- **Project Report: (Due: Friday, 3/10, by noon)**
- **Project Presentation: (Tuesday 3/7 and Thursday 3/9)**
  - Required only for PhD students in Statistics.
  - Presentation Style: Slide presentation (PPT, PDF, etc).
  - Time: 5-7 minutes.

Midterm Presentation Schedule (Final)

Tuesday (03/07)	Thursday (03/09)
1. Qi, Kai	1. Al Amer, Fahad
2. Rene, Lexi	2. Chen, Yang
3. Shamp, Wright	3. Griffith, Marie
4. Shen, Jiahui	4. Hu, Guanyu
5. Steppi, Albert	5. Lee, Hwiyoung
6. Tang, Shao	6. Lee, In Koo
7. Um, Seungha	7. Li, Donghang
8. Wang, Xianbin	8. Lim, Jaehui
9. Wang, Yunfan	9. Liu, Sida
10. Xu, Zhixing	

Special Topic 1

Laplace Approximation and Point Process Filter

S1.1 Laplace Approximation

Basic Idea

- The idea behind the Laplace approximation is simple.
- We assume that the probability density $P(x) = P^*(x)/Z_P$, where the normalizing constant
$$Z_P = \int P^*(x)\,dx.$$
- We also assume that the unnormalized density $P^*(x)$ has a peak at a point x_0.
- We Taylor-expand the logarithm of $P^*(x)$ around this peak:
$$\log P^*(x) \approx \log P^*(x_0) - \frac{c}{2}(x - x_0)^2 + \dots,$$
where
$$c = -\frac{\partial^2}{\partial x^2} \log P^*(x)\Big|_{x=x_0}.$$

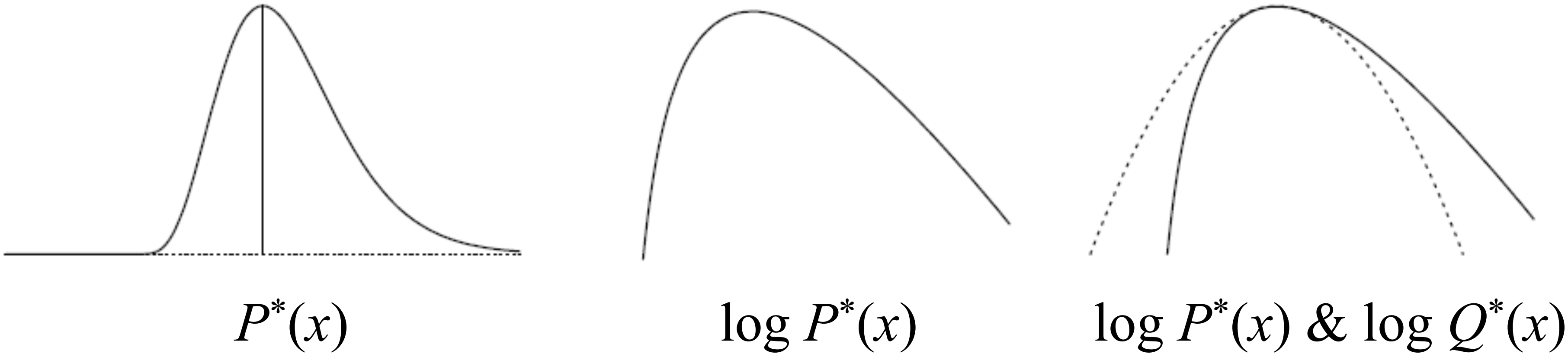
Basic Idea

- We then approximate $P^*(x)$ by an unnormalized Gaussian,

$$Q^*(x) \approx P^*(x_0)\exp[-\frac{c}{2}(x-x_0)^2],$$

and we approximate the normalizing constant Z_P by the normalizing constant of this Gaussian,

$$Z_Q \approx \int P^*(x_0)\exp[-\frac{c}{2}(x-x_0)^2]dx = P^*(x_0)\sqrt{\frac{2\pi}{c}}$$



General Case

- We can generalize this integral to approximate Z_P for a density $P^*(x)$ over a K-dimensional space x.
- If the matrix of second derivatives of - log $P^*(x)$ at the maximum x_0 is A, defined by:

$$A_{ij} = -\frac{\partial^2}{\partial x_i \partial x_j} \log P^*(x)\Big|_{x=x_0}.$$

- Therefore,

$$\log P^*(x) \approx \log P^*(x_0) - \frac{1}{2}(x-x_0)^T A(x-x_0) + \ldots,$$

- We let

$$Q^*(x) \approx P^*(x_0) \exp[-\frac{1}{2}(x-x_0)^T A(x-x_0)],$$

General Case

- Then, the normalizing constant is

$$Z_Q = \int P^*(x_0)\exp[-\frac{1}{2}(x-x_0)^T A(x-x_0)]dx$$

$$= P^*(x_0)\int \exp[-\frac{1}{2}(x-x_0)^T A(x-x_0)]dx$$

- Note that for *k*-dimensional random vector $x \sim N(\mu, \Sigma)$

$$f(x) = \frac{1}{(2\pi)^{k/2}(\det\Sigma)^{1/2}}\exp[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)]$$

- Therefore,

$$Z_Q = P^*(x_0)\sqrt{\frac{(2\pi)^k}{\det A}}.$$

Property

- The Laplace approximation is a Gaussian based method.
- That is, the logarithm of a Gaussian function is a quadratic function, and this is the basic idea for the approximation.
- This method works well when the logarithm of the object function is concave.

S1.2 Point Process Filter

Filtering Estimation

- Let $x_1, x_2, \ldots, x_t, x_{t+1}, \ldots,$ form a process of interest, and instead of measuring x_ts directly, one observes $y_1, y_2, \ldots, y_t, y_{t+1}, \ldots$
- The goal is to use the observations, and joint probability models of x and y to estimate the unknown x_ts.
- Let $f(x_t \mid y_1, y_2, \ldots, y_t)$ be the posterior density function of x_t given a set of observations $(y_1, y_2, \ldots, y_t)$.
- We are interested in the filtering estimation (posterior mean):

$$\hat{x}_t = E_f(x_t \mid y_1, y_2, \ldots, y_t).$$

Nonlinear Filtering Problem

- Let the *state vector* $x_t \in \mathbf{R}^d$ and the observation vector $y_t \in \mathbf{R}^C$.
- **State equation:** $x_{t+1} = \mathrm{F}(x_t) + w_t$

  Observation equation: $y_t = \mathrm{G}(x_t) + q_t$
- Estimation:

  update equation

$$f(x_t \mid y_1, y_2, \ldots, y_{t-1}) = \int_{x_{t-1}} f(x_t \mid x_{t-1}) f(x_{t-1} \mid y_1, y_2, \ldots, y_{t-1})\, dx_{t-1}$$

  prediction equation

$$f(x_t \mid y_1, y_2, \ldots, y_t) = \frac{f(y_t \mid x_t) f(x_t \mid y_1, y_2, \ldots, y_{t-1})}{f(y_t \mid y_1, y_2, \ldots, y_{t-1})}$$

Point Process Observation

- Assume the state vector $x_k \in \mathbf{R}^d$
- Assume the observation is a **point process** in time interval [0, T].
- The time interval is discretized to time bins $t_1, t_2, \ldots, t_M$.
- For $c = 1, \ldots, C$, let the c-th observation component

$$y_{k,c} = \text{number of events in the } k\text{-th time bin.}$$

- This number is either 0 or 1 if the bin size is sufficiently small.
- Therefore, the observation at the k-th time bin is

$$y_k = \{y_{k,c}\}_c \in \mathbf{R}^C$$

State-Space Model

- Assume x_k follow a simple linear Gaussian transition. That is,

$$x_k = A x_{k-1} + w_k, \qquad w_k \in N(0, W)$$

where the A and W can be fitted in closed-form using **Maximum likelihood Estimation (MLE).**

- For y_k, we assume a generalized linear model (GLM) with an inhomogeneous Poisson process conditioned on x_k. That is,

$$y_{k,c} \sim Poisson(\lambda_{k,c})$$

where

$$\lambda_{k,c} = \exp(\mu_c + \alpha_c^T x_k)$$

- $\{\mu_c, \alpha_c\}$ can also be identified using **MLE.**

System Identification

- For each $c = 1, \ldots, C$, assume the observations are $\{x_k, y_{k,c}\}$.
- We maximize the likelihood

$$L = p(\{y_{k,c}\} \mid \{x_k\}) = \prod_{k=1}^{M} p(y_{k,c} \mid x_k) = \prod_{k=1}^{M} \frac{e^{-\lambda_{k,c}} (\lambda_{k,c})^{y_{k,c}}}{y_{k,c}!}$$

- The log-likelihood is:

$$\begin{aligned} LL &= \sum_{k=1}^{M} y_{k,c} \log(\lambda_{k,c}) - \lambda_{k,c} + const \\ &= \sum_{k=1}^{M} y_{k,c} (\theta_c^T X_k) - \exp(\theta_c^T X_k) + const \end{aligned}$$

where

$$\theta_c = (\mu_c, \alpha_c^T)^T, X_k = (1, x_k^T)^T$$

System Identification

- We use a Newton-Raphson method.

$$\frac{\partial LL}{\partial \theta_c} = \sum_{k=1}^{M} y_{k,c} X_k - \exp(\theta_c^T X_k) X_k$$

$$\frac{\partial^2 LL}{\partial \theta_c \partial \theta_c^T} = -\sum_{k=1}^{M} \exp(\theta_c^T X_k) X_k X_k^T$$

- Recursive update:

$$(\theta_c)_{i+1} = (\theta_c)_i - \left(\frac{\partial^2 LL}{\partial \theta_c \partial \theta_c^T} \right)_i^{-1} \left(\frac{\partial LL}{\partial \theta_c} \right)_i$$

Point Process Filter

- To estimate the posterior $f(x_k \mid y_1, y_2, \ldots, y_k)$, we can use a sequential Monte Carlo (SMC) method.
- However, the method depends on number of sample points at each time step. (Note: these sample points are also called “particles” and the SMC is also called “particle filtering”).
- A large number of particles often leads to inefficient computation.
- Here we introduce an efficient, deterministic estimation method, called **point process filter**.
- This method is based on Laplace approximation by approximate the posterior at each time using a Gaussian distribution.

Estimation Process

- We use the recursive formula

$$f(x_k \mid y_1, y_2, \ldots, y_k) \propto f(y_k \mid x_k) f(x_k \mid y_1, y_2, \ldots, y_{k-1})$$

- Assume that conditioned on x_k, all components in y_k are independent. Therefore

$$f(y_k \mid x_k) = \prod_{c=1}^{C} f(y_{k,c} \mid x_k)$$

$$= \prod_{c=1}^{C} \exp(-\lambda_{k,c})) \frac{(\lambda_{k,c})^{y_{k,c}}}{y_{k,c}!}$$

- We use the following notation to simplify the sub-index

$$\boldsymbol{a}_{1:n} = \boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n.$$

Time Update

- We approximate the posterior using a Gaussian distribution at each time $k-1$. That is, let

$$x_{k-1|k-1} = E(x_{k-1} \mid y_{1:k-1}) \qquad W_{k-1|k-1} = Var(x_{k-1} \mid y_{1:k-1})$$

- Then

$$f(x_k \mid y_{1:k-1}) = \int f(x_k \mid x_{k-1}) f(x_{k-1} \mid y_{1:k-1}) dx_{k-1}$$

is also normally distributed.

- The mean is computed as:

$$\begin{aligned} x_{k|k-1} &= E(x_k \mid y_{1:k-1}) = E(Ax_{k-1} + \underline{w_k} \mid y_{1:k-1}) \\ &= AE(x_{k-1} \mid y_{1:k-1}) \\ &= Ax_{k-1|k-1} \end{aligned}$$

Time Update

- The covariance is computed as:

$$W_{k|k-1} = Var(x_k \mid y_{1:k-1}) = Var(Ax_{k-1} + w_k \mid y_{1:k-1})$$
$$= Var(Ax_{k-1} \mid y_{1:k-1}) + Var(w_k)$$
$$= AW_{k-1|k-1}A^T + W$$

- Therefore,

$$f(x_k \mid y_{1:k}) \propto f(y_k \mid x_k) f(x_k \mid y_{1:k-1})$$
$$= \left(\prod_{c=1}^{C} \exp(-\lambda_{k,c}) \frac{(\lambda_{k,c})^{y_{k,c}}}{y_{k,c}!} \right) \cdot \exp(-\frac{1}{2}(x_k - x_{k|k-1})^T W_{k|k-1}^{-1}(x_k - x_{k|k-1}))$$

Measurement Update

- Then, the logarithm of the posterior is

$$\log f(x_k \mid y_{1:k})$$
$$= \left(\sum_{c=1}^{C} y_{k,c} \log \lambda_{k,c} - \lambda_{k,c} \right) - \frac{1}{2}(x_k - x_{k|k-1})^T W_{k|k-1}^{-1}(x_k - x_{k|k-1}) + const$$

- We approximate this posterior by a Gaussian distribution

$$\log f(x_k \mid y_{1:k}) = -\frac{1}{2}(x_k - x_{k|k})^T W_{k|k}^{-1}(x_k - x_{k|k}) + const$$

- Then,

$$\frac{1}{2}(x_k - x_{k|k})^T W_{k|k}^{-1}(x_k - x_{k|k})$$
$$= \frac{1}{2}(x_k - x_{k|k-1})^T W_{k|k-1}^{-1}(x_k - x_{k|k-1}) - \left(\sum_{c=1}^{C} y_{k,c} \log \lambda_{k,c} - \lambda_{k,c} \right) + const$$

Measurement Update

- Differentiate w.r.t. to x_k, we have

$$
\begin{aligned}
& W_{k|k}^{-1}(x_k - x_{k|k}) \\
&= W_{k|k-1}^{-1}(x_k - x_{k|k-1}) - \sum_{c=1}^{C}\left[y_{k,c}\frac{\partial \log \lambda_{k,c}}{\partial x_k} - \frac{\partial \lambda_{k,c}}{\partial x_k}\right] \\
&= W_{k|k-1}^{-1}(x_k - x_{k|k-1}) - \sum_{c=1}^{C}\left[y_{k,c}\alpha_c - \lambda_{k,c}\alpha_c\right]
\end{aligned}
$$

- Differentiate again,

$$
W_{k|k}^{-1} = W_{k|k-1}^{-1} + \sum_{c=1}^{C}\frac{\partial \lambda_{k,c}}{\partial x_k}\alpha_c = W_{k|k-1}^{-1} + \sum_{c=1}^{C}\alpha_c \lambda_{k,c}\alpha_c^T
$$

Measurement Update

- Let $x_k = x_{k|k\text{-}1}$ after the second differentiation, we have

$$W_{k|k} = \left(W_{k|k-1}^{-1} + \sum_{c=1}^{C} \alpha_c \exp(\mu_c + \alpha_c^T x_{k|k-1}) \alpha_c^T \right)^{-1}$$

- Let $x_k = x_{k|k\text{-}1}$ after the first differentiation, we have

$$W_{k|k}^{-1}(x_{k|k-1} - x_{k|k}) = -\sum_{c=1}^{C} [y_{k,c}\alpha_c - \exp(\mu + \alpha_c^T x_{k|k-1})\alpha_c]$$

- Therefore,

$$x_{k|k} = x_{k|k-1} + W_{k|k} \sum_{c=1}^{C} [y_{k,c} - \exp(\mu_c + \alpha_c^T x_{k|k-1})]\alpha_c$$

Algorithm

- **Point Process Filter Algorithm:**

  Update from time *k*-1 to *k*:

$$x_{k-1} \mid y_{1:k-1} \sim N(x_{k-1|k-1}, W_{k-1|k-1}) \rightarrow x_k \mid y_{1:k} \sim N(x_{k|k}, W_{k|k})$$

  Time update: $W_{k|k-1} = AW_{k-1|k-1}A^T + W$

$$x_{k|k-1} = Ax_{k-1|k-1}$$

  Measurement update:

$$W_{k|k} = \left(W_{k|k-1}^{-1} + \sum_{c=1}^{C} \alpha_c \exp(\mu_c + \alpha_c^T x_{k|k-1}) \alpha_c^T \right)^{-1}$$

$$x_{k|k} = x_{k|k-1} + W_{k|k} \sum_{c=1}^{C} [y_{k,c} - \exp(\mu_c + \alpha_c^T x_{k|k-1})] \alpha_c$$