(Supplemental Material) Appendix:

Variational Inference for Gaussian Process Models for Survival Analysis

Minyoung Kim*
Dept. of Electronic Engineering
Seoul Nat'l Univ. of Science & Technology
Seoul, Korea

Vladimir Pavlovic
Dept. of Computer Science
Rutgers University
Piscataway, NJ 08854, USA

A Full Likelihood Derivation

We derive the full likelihood for the GPSAM model, specifically (9) in the main manuscript. For a censored example $(t, \mathbf{x})$, the tail probability $P(T \geq t|\mathbf{x})$ can be derived as follows:

$$\begin{aligned} P(T \geq t|\mathbf{x}) &= 1 - P(T < t|\mathbf{x}) && \text{(A.1)} \\ &= 1 - \int_0^t \lambda(s|\mathbf{x}) \exp\left(- \int_0^s \lambda(\tau|\mathbf{x}) \, d\tau \right) ds && \text{(A.2)} \\ &= 1 - \int_0^t -\Lambda'(s) \, ds && \text{(A.3)} \\ &= 1 - (\Lambda(0) - \Lambda(t)) && \text{(A.4)} \\ &= \exp\left(- \int_0^t \lambda(\tau|\mathbf{x}) \, d\tau \right) = \Lambda(t). && \text{(A.5)} \end{aligned}$$

Note that we let $\Lambda(t) = \exp\big(- \int_0^t \lambda(\tau|\mathbf{x}) \, d\tau \big)$. Combining this with the event examples, for which we use $\log P(t|\mathbf{x}) = \log \lambda(t|\mathbf{x}) - \int_0^t \lambda(\tau|\mathbf{x}) \, d\tau$ straightforwardly from (1), we have the full log-likelihood as in (9).

B More Experimental Results

In addition to the experimental results reported in the main manuscript, we also provide additional empirical results in this section.

First, to see the effect of the approximation model complexity of our variational inference methods (i.e., the number of pseudo inputs (M) in VI_{PI} and the number of random features (m) in VI_{RF}), we vary these parameters in model learning, and record the test performance measure, in this case the concordance index. Specifically, M is chosen from the set $\{10, 20, 40, 60, 80\}$ for VI_{PI} and m is from $\{5, 10, 20, 30, 50\}$. The results on three datasets are shown in Table 1.

The result implies that increasing the number of random features (m in VI_{RF}) clearly improves the prediction performance. This is because we have more accurate Monte Carlo estimates for the kernel functions. On the other hand, increasing the size of the pseudo inputs does not necessarily yield a better model (sometimes deteriorating the performance). As we infer the posterior GP latent function values at the test inputs based on the pseudo inputs, it is rather more important where these anchor points are positioned than how many they are. In our case, clustering the data by $M = 20$ center points performs consistently well across all datasets.

The second set of experiments is on the effect of the approximation model complexity on the inference time. For the five different model complexities similarly as before, we compare the running times

* Also affiliated with Rutgers University.

Table 1: The concordance index scores (%) when we vary the approximation model complexity. For VI_{PI}, we test with the number of pseudo inputs $M = 10, 20, 40, 60, 80$, and for VI_{RF} the number of random features $m = 5, 10, 20, 30, 50$.

(a) VLC dataset

M	$\mathbf{VI}_{PI}^{f}$	$\mathbf{VI}_{PI}^{p}$	m	$\mathbf{VI}_{RF}^{f}$	$\mathbf{VI}_{RF}^{p}$
10	57.34 ± 10.74	56.62 ± 9.17	5	72.68 ± 9.79	71.55 ± 8.83
20	71.99 ± 3.67	70.87 ± 4.75	10	75.57 ± 4.87	75.56 ± 4.81
40	72.26 ± 3.50	70.08 ± 4.93	20	73.91 ± 4.09	73.49 ± 3.94
60	64.47 ± 6.53	68.25 ± 4.71	30	75.46 ± 4.29	74.57 ± 4.35
80	71.17 ± 4.99	68.32 ± 4.01	50	76.79 ± 4.08	76.49 ± 4.51

(b) MLC dataset

M	$\mathbf{VI}_{PI}^{f}$	$\mathbf{VI}_{PI}^{p}$	m	$\mathbf{VI}_{RF}^{f}$	$\mathbf{VI}_{RF}^{p}$
10	62.40 ± 9.94	61.58 ± 10.25	5	66.94 ± 4.65	67.61 ± 5.01
20	72.68 ± 3.00	69.44 ± 6.48	10	67.16 ± 5.09	67.21 ± 4.69
40	71.19 ± 5.56	70.99 ± 7.07	20	67.76 ± 5.16	67.46 ± 4.76
60	71.37 ± 6.63	70.93 ± 7.34	30	66.71 ± 5.10	66.94 ± 4.31
80	68.44 ± 7.68	69.09 ± 6.42	50	68.08 ± 4.98	67.00 ± 5.35

(c) Divorce dataset

M	$\mathbf{VI}_{PI}^{f}$	$\mathbf{VI}_{PI}^{p}$	m	$\mathbf{VI}_{RF}^{f}$	$\mathbf{VI}_{RF}^{p}$
10	62.83 ± 2.89	63.03 ± 3.40	5	63.47 ± 2.34	63.02 ± 2.43
20	63.60 ± 3.17	62.89 ± 3.92	10	64.64 ± 3.22	65.41 ± 2.17
40	61.21 ± 2.61	62.27 ± 5.02	20	64.46 ± 2.34	64.32 ± 2.41
60	59.94 ± 5.69	59.70 ± 7.98	30	63.71 ± 3.87	65.30 ± 2.12
80	62.11 ± 4.43	63.32 ± 4.51	50	63.73 ± 2.67	64.56 ± 3.47

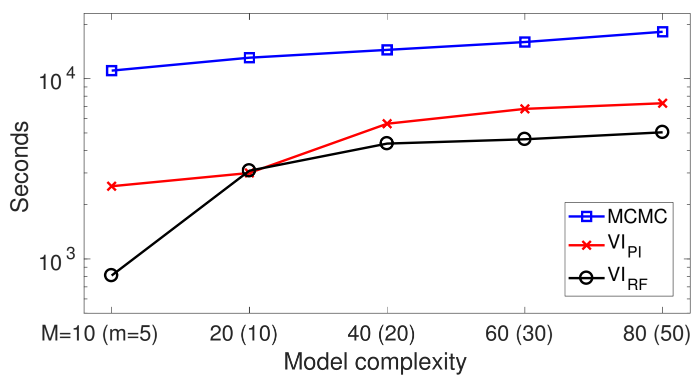

Figure 1: The inference running times for MCMC, VI_{PI}^{f}, and VI_{RF}^{f} on the MLC dataset. We vary the model complexity: $M \in \{10, 20, 40, 60, 80\}$ for VI_{PI}^{f}, and $m \in \{5, 10, 20, 30, 50\}$ for VI_{RF}^{f} and the MCMC. All methods are implemented in MATLAB run on 2.4GHz Intel Xeon CPU.

of VI_{PI}^{f}, VI_{RF}^{f}, and also the MCMC method on the MLC dataset in Fig. 1. For the other datasets, they are summarized in Table 2. All methods are implemented in MATLAB run on 2.4GHz Intel Xeon CPU. As shown, our variational inference approaches are an order of magnitude faster than the MCMC, while achieving comparable or often superior prediction performance.

Table 2: (VLC and Divorce datasets) The inference running times (in seconds) with varying model complexity. For VI^f_{PI}, we vary the number of pseudo inputs $M = 10, 20, 40, 60, 80$. For VI^f_{RF}, we vary the number of random features $m = 5, 10, 20, 30, 50$. The MCMC with $m = 50$ takes 1.6337×10^4 seconds on the VLC dataset, and 3.2357×10^4 seconds on the Divorce dataset. All methods are implemented in MATLAB run on 2.4GHz Intel Xeon CPU.

(a) VLC dataset

M	VI^f_{PI}	m	VI^f_{RF}
10	1.8313×10^3	5	9.5802×10^2
20	2.2438×10^3	10	2.1313×10^3
40	4.1936×10^3	20	3.1971×10^3
60	4.9995×10^3	30	3.7319×10^3
80	4.5279×10^3	50	4.6245×10^3

(b) Divorce dataset

M	VI^f_{PI}	m	VI^f_{RF}
10	6.3978×10^3	5	3.8486×10^3
20	9.8063×10^3	10	3.9375×10^3
40	1.2724×10^4	20	6.0096×10^3
60	4.5859×10^3	30	8.6824×10^3
80	9.2486×10^3	50	1.2714×10^4