Modeling "Presentness" of Electronic Health Record Data to Improve Patient State Estimation

Jacob Fauber JFAUB001@UCR.EDU
Department of Computer Science & Engineering
University of California, Riverside

Christian R. Shelton CSHELTON@CS.UCR.EDU
Department of Computer Science & Engineering
University of California, Riverside

Abstract

Medical data are not missing at random. The problem is more acute when the observations are over an extended period of time; any particular variable is observed at relatively few time points. We take missing values to be the norm, and treat "presentness" (the times of observations) as additional features to augment the values observed. A joint model over both avoids the missing-at-random (MAR) assumption. We use piecewise-constant conditional intensity models (PCIMs) to build a generative model of observation times and values. We demonstrate its effectiveness in reconstruction of monitor readings of patient vitals from sparse EHR data.

1. Introduction

While initially developed as a databank for health care and billing records, Electronic Health Records (EHRs) are increasingly acknowledged as a rich potential data source for the improvement of patient care and advancement of personalized medicine. They are frequently used in disease prediction tasks, including sepsis (Desautels et al., 2016), diabetes (Bo and Sminchisescu, 2010), and arthritis (Carroll et al., 2011), and have been the subject of comparative studies (Wu et al., 2010) and general overviews (Jensen et al., 2012).

Many difficulties in using EHR data in other large-scale studies have been previously reported (Hersh et al., 2013). Most importantly, such data are potentially incomplete or inaccurate. EHR data are often collected for billing and other non-diagnosis purposes, and thus capture values related but not equivalent to those of the greatest interest for medical prediction tasks. Additionally, complete EHR data often do not appear with enough frequency to provide an accurate sampling of patient state. Given the expansive databank of EHRs for patient episodes which have no higher frequency data available, a joint generative model that can correlate EHRs to data at a finer timescale would increase the usefulness of EHR data in diagnoses and comparative studies. Such a model offers further advantages, including inference over missing data and improved prediction of future patient state. If such a model can also provide a coherent interpretation of this correlation, future EHR collection can be amended to better suit these tasks.

We suggest that EHR data are not missing at random. For instance, an extended gap in the EHR record may suggest the patient is stable; if EHR measurements occur with greater

frequency, it may suggest an unstable condition, with high variation in patient vitals across a short time. We desire to model a process which includes the frequency of EHR events in its estimation to allow encorporation of EHR event presence (or absence) into the posterior estimate.

We propose the use of piecewise-constant conditional intensity models (PCIMs) for this task. After introducing the data, PCIM, and comparison models, we measure its effectiveness in a representative inference problem.

Technical Significance We model missing-not-at-random (MNAR) event data in the medical context using a point process model. The existing piecewise-constant conditional intensity model (PCIM) is extended to include estimates over not only rate of event occurrence, but also a over event value. An new MCMC inference method is developed around this augmented PCIM which allows the probabilistic inference of continuous monitor values from irregularly sampled EHR data.

Clinical Relevance Most prior machine learning approaches to medical data assumed the data were present at random. Little to no work has previously reported on a joint generative model correlating low granularity EHR readings to higher granularity reports of patient vitals. We build a statistical model that relates the high frequency monitor data to the EHR values *and* EHR record times. This explicitly acknowledges and models that the timings alone of EHR entries are indicative of patient state. In so doing, we are able to improve reconstruction of patient state over models which assume the data are missing (or present) at random. This provides a method for using data that is missing-not-at-random, and provides some insight into the process by which clinicians choose when to chart vitals.

The majority of previous applications of machine learning to EHR data concerned disease prediction tasks. Most often these predictions assumed the data were present at random. Little to no work has previously reported on a joint generative model correlating low granularity EHR readings to higher granularity reports of patient vitals. By building a statistical model that relates the high frequency monitor data to the EHR values *and* times, acknowledging that the timings of EHR entries are indicative of patient state on their own, we are able to improve reconstruction of patient state over assuming the data are missing (or present) at random.

2. Prior Work

Bayesian networks (BNs) (Pearl, 2014) are commonly used to represent dependencies between variables, but have no intuitive way to incorporate the temporal dependencies found in medical data. Dynamic Bayesian networks (Dean and Kanazawa, 1989) model temporal dependencies over discrete time intervals. In the case of medical trajectories, it is not clear how event times should be discretized, because their occurrence is irregular.

It would be easier to use a model that is already adapted to the continuous nature of the data. A continuous-time Bayesian network (CTBN) is an extension of a BN to continuous-time processes (Nodelman et al., 2002). Rather than modeling event trajectories directly, the CTBN models a homogeneous Markov process of the joint trajectories over a discrete number of variables, which can then generate events appropriately.

BNs have seen a wide range of applications in the medical field. Lin and Haug (2008) have used BNs to demonstrate the value of incorporating knowledge of data absences in a disease prediction task. In such a case, when the BN is learned, the structure may not appear sensible to a domain expert.

Extensive work has also been developed on the use of EHR data in health care, specifically for disease classification and survival analysis. Zhao and Weng (2011) have used BNs with EHR data in combination with PubMed text for pancreatic cancer prediction. Singh et al. (2015) found the prediction of kidney failure improved with the incorporation of temporal information from EHRs, using a task-learning approach.

Less work has been done concerning the application of PCIMs to EHR and other medical data. Weiss and Page (2013) have used a forest model based upon PCIMs to improve EHR forecasting results considerably over a generic Poisson process. Weiss (2017) also extended the PCIM model to allow parametric functions over the rate, focusing on the prediction of diabetic ketoacidosis using a log logistic distribution.

As far as we are aware, no previous work has directly sought to build a model correlating EHR data to more temporally fine grained observations.

3. Cohort

Our cohort consists of 1,340 anonymized patient episodes from a general pediatric intensive care unit (PICU). Each patient episode has measurements of patient vitals given in two forms: a monitor trajectory x_M and an EHR trajectory x_R. We take x_M as the fine-scale measurements and x_R as the coarse measurements. Each trajectory contains three event types: heart rate, respiratory rate, and pulse oximetry. 430 of the available episodes also include measurements of diastolic and systolic blood pressure. We consider each event type separately, building a joint model over the monitor and EHR records. This is a simplification and further gains could probably be achieved by building a joint model over all event types in both monitor and EHR modalities.

Episodes vary in length from less than an hour to more than two weeks. EHR events usually occur every thirty minutes, while monitor events usually occur every thirty seconds, although there are deviations from this pattern. We expect that such deviations are related to patient state; for instance, a patient with a rapidly fluctuating heart rate may have more frequent EHR measurements than one with a steady heart rate.

The training set consists of 50% of the available episodes selected at random. The testing set is the remaining 50% of the data.

4. Problem Statement

We would like a model that represents the joint distribution over both measurement times and values for both x_M and x_R. To test the effectiveness of this model, we consider its use on problems over the conditional distribution

$$P(x_M \mid x_R)$$

judging the model's ability to reconstruct the fine-scale measurements. The available data x_R and possible predictions of x_M can be seen in Figure 1.

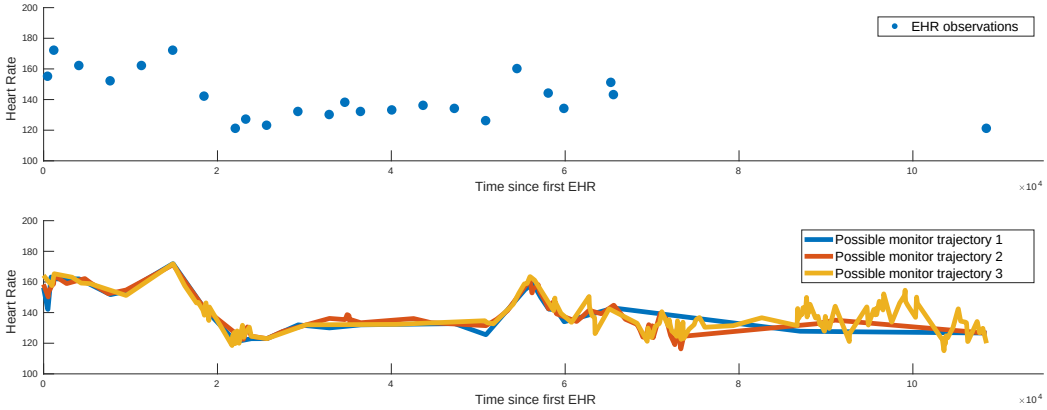

Figure 1: The first plot contains the observed variables from the EHR trajectory for a sample patient episode. The second plot contains examples of what must be estimated, the monitory trajectory. Any of the three trajectories could possibly accompany the observed variables. Note that the values of the EHR and monitor trajectories may not agree, even at identical timestamps.

5. Model

The piecewise-constant conditional intensity model (PCIM) (Gunawardana et al., 2011) is a model developed specifically for representing temporal dependencies between event streams.

It is a marked point process, specifying for each event type, for all time, the conditional rate (or intensity) function $\lambda_l(h|t) = \lim_{\delta \to 0} p(\text{event of type } l \in (t, t+\delta))/\delta$. The rate at any given point in time t is dependent on the event stream history up to time t: $h(t) = \{(t_i, v_i, l_i) | (t_i, v_i, l_i) \in x, t_i < t\}$, where event (t, v, l) signifies that at time t, value v was recorded in record $l \in \{M, R\}$. The rate function is represented by a tree whose internal nodes are binary questions about the history and whose leaves are constants, specifying the rate. This results in λ being a piecewise function of t, for any x. Figure 2 illustrates two PCIMs built for a marked point process.

We augment this model by including in all leaf nodes a distribution over the value associated with the event (the original PCIM only considers events without associated values). We take this to be a normal distribution. The mean is a linear function of a "context": a few summary statistics of the recent history. For this study we take them to be the most recent EHR value, the most recent monitor value, and the variance and mean of the monitor values over the last five minutes. When unavailable, these values are taken to be zero. The variance is a constant (per leaf). Thus, each leaf has six values: the rate of events, the four parameters of the linear dependence of the mean on the context, and the variance of the event value. The rate dictates the distribution over when an event will occur, and the other parameters specify the value associated with the event (e.g., a recording of the heart rate) if an event does occur.

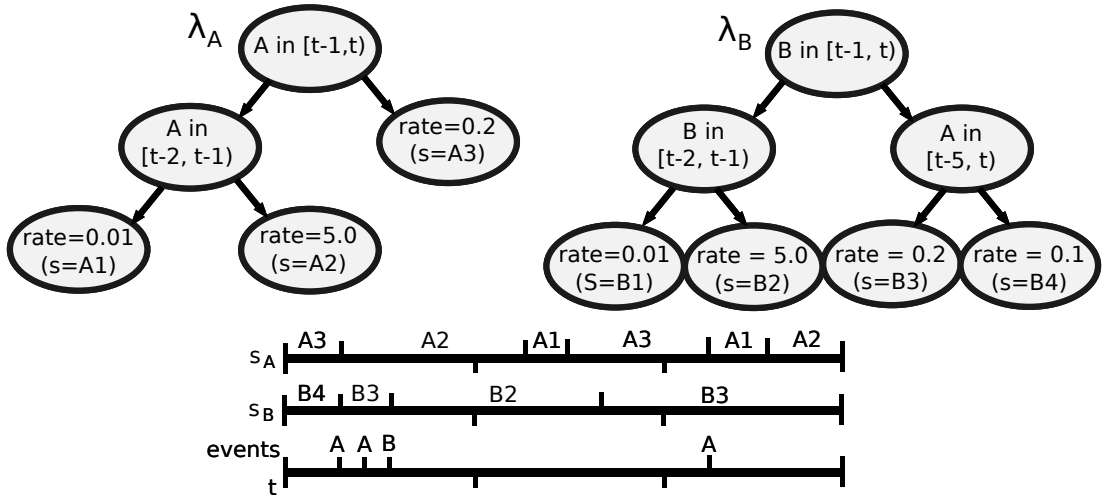

Figure 2: A sample PCIM for a single event type. Decision nodes may be much more complex, depending on any statistic in the history. Note that the rates of event B (right tree) depend on the history of events of label A. The time line at the bottom depicts the leaf (and thereby the rate) for each event type, given the sequence of events at the bottom.

Let $\mathcal{S}$ be the set of leaves of the tree. Let λ_s, w_s, and σ_s^2 be the rate, linear weights for the mean, and variance associated with leaf $s \in \mathcal{S}$. Let $E_s(x)$ be the subset of the events of x for which s is the active leaf. Let $z_x(t)$ be the context of x at time t.

The sufficient statistics are then

$$c_s(x) = \sum_{(t,v)\in E_s(x)} 1 \qquad \rho_s(x) = \sum_{(t,v)\in E_s(x)} v$$

$$\zeta_s(x) = \sum_{(t,v)\in E_s(x)} z_x(t) \qquad \beta_s(x) = \sum_{(t,v)\in E_s(x)} z_x(t) z_x(t)^\top$$

and $d_s(x)$, the duration of the trajectory x during which s is the active leaf. $c_s(x)$ can also be understood as a count function. The total log-likelihood is then

$$\sum_{s\in\mathcal{S}} \left[c_s(x) \ln \lambda_s - d_s(x)\lambda_s - c_s(x)\frac{\ln(2\pi\sigma_s^2)}{2} - \frac{1}{2\sigma_s^2}\left(\rho_s(x) - 2\zeta_s(x)^\top w_s + w_s^\top \beta_s(x) w_x^\top\right)\right] .$$

Estimation of a PCIM from data is detailed by Gunawardana et al. (2011). A similar log-likelihood for the instance when the context is over only the current and previous event is provided in Casse (2014).

The PCIM requires the specification of a base set of binary questions $\mathcal{B}$ over h, which serve for potential nodes in the tree. Notably, the tree structure provided by a PCIM is interpretable, and the specification of base questions allows the incorporation of specific domain knowledge.

The tree-building algorithm is greedy, optimizing a Bayesian structure score: an exponential prior is placed over tree sizes and independent conjugate priors are placed over the parameters in the leaves. At each step, a leaf is replaced with the split test from the provided set $\mathcal{B}$ that maximizes the probability of the tree structure given the data (marginalizing out the leaf parameters). Because the Bayesian score marginalizes out the parameters, there is a general preference for smaller trees. The algorithm stops when any additional growth would cause the score to decrease.

5.1. Comparative Model: Gaussian Process

A natural comparison for our PCIM model would be a Gaussian process (GP). Unlike the PCIM model, a GP assumes the EHR data to be present at random, and therefore does not incorporate any missingness dependencies in its estimation. A GP is a continuous generalization of a multivariate Gaussian distribution. When the number of possible random variables is finite, the covariance can be described in a matrix; in a GP the covariance is described by a positive-definite kernel function, with a covariance matrix built by applying that function to the observed times.

Our GP needs multi-dimensional output (one for each event type: monitor and EHR). There are a number of more complex recent solutions for this problem, which often take the kernel parameters to be composed by a combination or transformation of independent underlying channels (Boyle and Frean, 2005; Seeger et al., 2005). Our solution is derived from the early joint prediction method known as co-kriging (Cressie, 1993).

In the multi-dimensional output case, the kernel function therefore maps a pair of times (t, t') *and event types* (l, l') to a covariance, allowing for cross-correlation between event types. We use a squared-exponential kernel which, with multi-dimensional output, takes the form

$$k_{SE}(t, t', l, l') = \sigma_{l,l'}^2 \exp \frac{(t - t')^2}{2\omega_{l,l'}^2}$$

where $\sigma_{l,l'}$ and $\omega_{l,l'}$ are the vertical and horizontal scale parameters with the constraint that $\sigma_{l,l'} = \sigma_{l',l}$ and $\omega_{l,l'} = \omega_{l',l}$ to assure symmetry.

Given a set of event times and types, if we order the events by type, the resulting covariance matrix is divided into submatrices correlating x_R to x_R, x_M to x_M, and x_R to x_M:

$$\left[\begin{array}{c:c} K_{M,M} & K_{M,R} \\ \hdashline K_{R,M} & K_{R,R} \end{array}\right]$$

This composition of positive-definite kernel functions is not itself guaranteed to be positive-definite, but an appropriate line search can optimize all scale parameters within the constraints of positive-definite matrices produced by the new kernel.

6. Methods

Gaussian processes have well established inference methods that allow us to calculate $p(x_M \mid x_R)$. While work has been done in direct inference over PCIMs (Qin and Shelton, 2015), it relies on a forward pass which is difficult to determine in processes with a large number of complex non-Markovian dependencies, and it has not been extended to

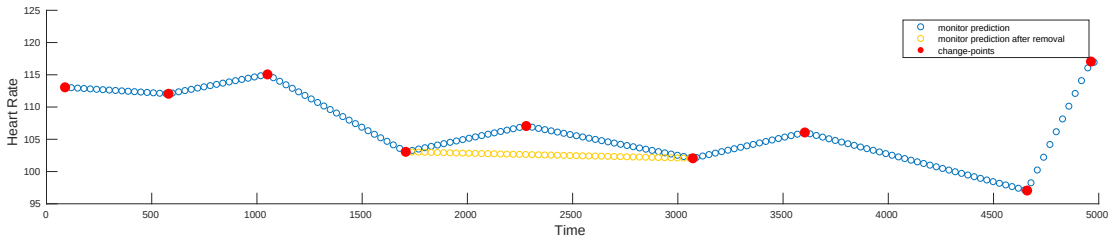

Figure 3: The orange line above represents the new trajectory after a node is removed.

PCIMs with values associated with the events. Instead, we will employ sampling to estimate $p(x_M \mid x_R)$.

In the experiments that follow, we desire use the PCIM with an EHR trajectory to establish a more granular monitor trajectory. Actual monitor trajectories may contain temporal irregularities just as EHR trajectories, but we ignore this, assuming all times of occurrence at thirty second intervals. Our estimation problem is over the values of these monitor observations, which occur at fixed times. A single-site MCMC algorithm, such as Metropolis-Hastings (Hastings, 1970) mixes too slowly, due to the dependencies in the values of x_M between time points. To overcome this obstacle, we modify the standard Metropolis-Hastings algorithm using Reversible Jump MCMC (RJ-MCMC) (Hoeting et al., 1999) to allow for faster mixing.

6.1. Reversible Jump MCMC

Rather than considering all values of x_M independently, we represent x_M as a piece-wise linear interpolation of a series of n change points, or nodes. The nodes, N, are time-value pairs and are initialized to the given EHR trajectory, x_R. For any step during the sampling process, three possible changes may occur.

1. Add a node: Between any two adjacent nodes, $(t_m, v_m), (t_{m+1}, v_{m+1})$ where $m < n$, a node is added at location $(t_m + t_{m+1})/2$ with a value of $(t_m + t_{m+1})/2 + \eta$, where η is a value drawn from a normal distribution.

2. Remove a node: Any node other than (t_1, v_1) or (t_n, v_n) is selected at random for removal.

3. Shift a node: The value of a node in set N selected at random is shifted by a value drawn from a normal distribution.

Figure 3 illustrates the impact to a linear interpolation in the case of a node removal. Due to this modification, the sample space of the Markov chain is no longer consistent in its dimension; this necessitates the use of an RJ-MCMC.

The RJ-MCMC enforces a "dimension-matching" constraint on the Markov chain, to ensure that the condition of detailed balance needed in a standard MCMC algorithm is consistent across different dimensions. In the Metropolis-Hastings algorithm, this means the multiplication of the acceptance ratio by a Jacobian term. However, in this instance,

because the addition and removal mechanisms have a natural symmetry, the Jacobian term reduces to one.

7. Results

We are estimating $P(x_M \mid x_R)$ using a revised Metropolis-Hastings algorithm on a learned PCIM. After a PCIM has been trained on half the data (670 patient episodes), the monitor trajectories M are estimated using only the PCIM and the EHR observations. We then compare this average to a simple linear interpolation of the EHR observations, and a GP estimate trained in an analagous way.

Experimental testing of the RJ-MCMC method involved a burn-in period of 25,000 iterations. Nodes were then collected for the subsequent 50,000 iterations. The three potential node edits (addition, removal, and value shifting) were given equal probability. Further testing showed that for addition and value shifting, using 5 as the standard deviation of this normal distribution provided for satisfactory mixing times. In the case of GP optimization, a coordinate gradient descent was used, with a stopping condition in the instance that a non-positive-definite matrix was produced.

7.1. Experimental Setup

To train our model, we divide the available patient episodes into a training and testing set of 670 each. We restate that for systolic and diastolic blood pressure, only about a third of the collected patient episodes were usable.

While all event types have separate models built for them, each model receives the same set of questions over history h by which to build its PCIM. This demonstrates the relative flexibility of a broad set of questions; however, model generation can be sped up by incorporating knowledge specific to event types to remove tests. For instance, pulse oximetry value fluctuates between 90 and 100, so tests on value outside of this range are useless, and could be eliminated.

In this instance, we selected 357 questions concerning the mean, variance, count, time interval, and last value of h:

1. Whether an EHR observation occurred in the past 5, 10, ..., 60 minutes.
2. Whether the last EHR value was below 2, 4, 6, ..., 200.
3. Whether the last monitor value was 2, 4, 6, ..., 200.
4. Whether the clock time of the current event is within 5, 10, ..., 55 minutes of the next new hour.
5. Whether the clock time can be evenly divided by two hours, an hour, half an hour, fifteen minutes, ten minutes, and five minutes.
6. Whether an EHR event has occurred in the last 25, 50, ..., 3600 seconds.
7. Whether the variance of monitor events over the past 1000 seconds is below 5, 10, ..., 50

	Linear	GP	PCIM-	PCIM
Heart Rate	861.4	870.2	750.6	671.9
Respiratory Rate	34.9	31.4	33.9	30.4
Pulse Oximetry	6.2	7.3	7.4	7.5*
Systolic B.P	66.4	62.7	67.7	63.8
Diastolic B.P.	44.8	40.7	43.3	39.7

Figure 4: The mean squared error over all testing monitor observations, for each event type and method. *: can be improved to 5.0 using method described in the text.

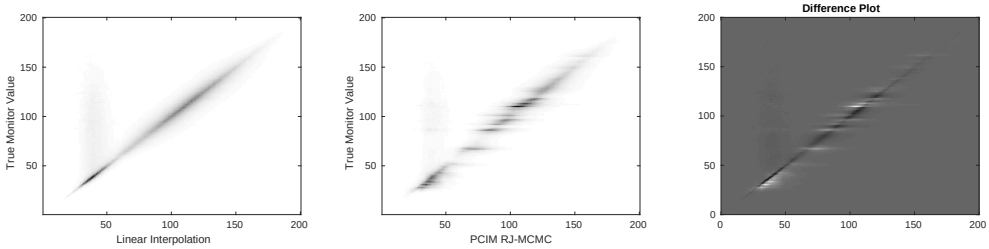

Figure 5: Heatmaps of the true test monitor values as a function of linearly interpolated and RJ-MCMC monitor trajectories for heart rate. To see the improvement, a difference plot between the two is to the right. The difference plot illustrates that, while the PCIM creates some minor striations in its prediction, it also corrects a significant number of underpredicted values from linear interpolation. Similar though much less frequent and severe interpolation errors are corrected in other event types as well.

Tests of type 4 and 5 are especially useful for EHR trajectories, in which usually occur at the top of the hour.

Several other kernels were tested with the GP method, including the Rational Quadratic kernel, which performed similarly, and a periodic kernel, which performed worse. In addition to a GP with multi-dimensional outputs, we present two other comparison methods. The first is a simple linear interpolation of x_M given trajectory x_R. The second is another PCIM, which was learned with all of the tests concerning values in the monitor trajectory history removed from the base set (called "PCIM-"). That is, the rate at which an event occurs cannot depend on the monitor values. This forces the model to treat the EHR values to be missing at random.

We report the mean squared reconstruction error in Figure 4. In general, EHRs are expected to have a very high correlation with the monitor readouts, so linear interpolation is expected to do well in all cases. The most notable improvement is clearly in heart rate. Heart rate has a bimodal distribution, and the heatmaps indicate that for both the PCIM and linear interpolation methods, there is a tendency to report low monitor readings as high. However, the difference plot (Figure 5) indicates that some of this behavior has been accounted for by the PCIM method.

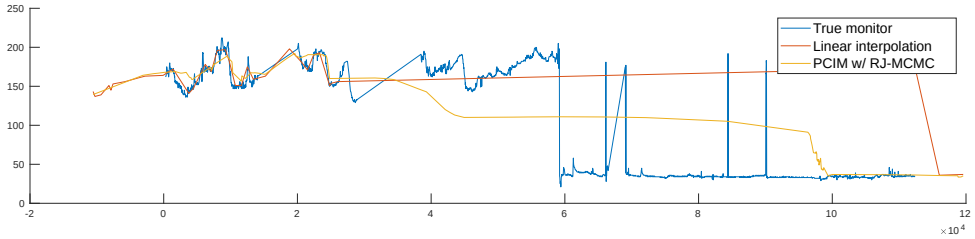

Figure 6: This patient episode represents of a strong improvement when using the PCIM. There is a long stretch in which EHR data is absent; the PCIM often provides considerable improvement when the observed data is lacking.

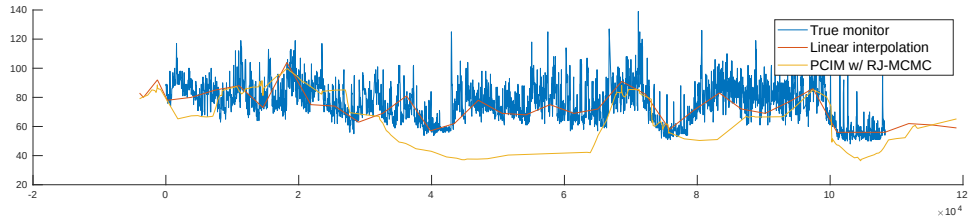

Figure 7: This patient episode represents of a rare failure when using the PCIM. It is possible for the PCIM to inappropriately ignore EHR data, choosing to follow a mean below the actual values.

Pulse Oximetry is largely stable around value 100. Furthermore, it is the only variable type with a hard cap, at 100. The GP does not account for this, which is why it performs poorly. While the PCIM also does not naturally account for this, a simplified PCIM with a constant function for the value of its leaves can be trained which will include this hard cap. If RJ-MCMC is run over this simplified PCIM, the MSE for Pulse Oximetry improves to 5.0.

The PCIM- model is outperformed by the PCIM in nearly every event type. In heart rate, the event type most improved by the PCIM, PCIM- still performs well; we speculate this is because a structured tree representation of event rate is naturally a better fit for bimodal distributions than linear interpolation or GP, even when EHR data is treated as missing at random. In all other event types, the PCIM- model performs poorly. A probable explanation for this can be seen in Figure 7; when the PCIM- cannot correct data using temporal information, the natural striation pattern created by a piecewise assumption performs slightly worse than linear interpolation.

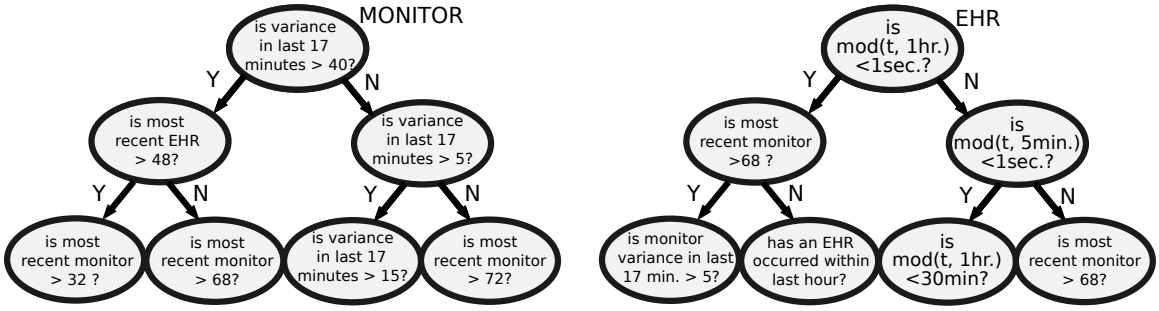

Figure 8: The first few children of the root node in the PCIM constructed for heart rate.

7.2. Model Analysis

The PCIM model has the advantage of being interpretable. The PCIM tree is very deep (at least ten levels deep), but we can look at children near the root to get an idea of dependencies that the PCIM has encoded, shown in Figure 8.

Monitor trajectories check variance of the history immediately; when there is low variance it is usually safe to guess the previous observation value more exactly. EHR trajectories first check for whether observations occur on the hour, suggesting that the rare instances, when they do not, have a distinct rate and value to associate with them. Within the first two levels of the tree for the monitor trajectory, the EHR function splits based on the values. This allows the inference to adjust the monitor estimate based on the frequency of EHR measurements.

8. Discussion

We have proposed the use of an extended PCIM for modeling temporal dependencies between EHR data and higher granularity trajectories. We have provided a practical method by which to use the PCIM in inference problems, and demonstrated that such a method outperforms commonplace methods which do not take temporal dependencies into account. This performance boost is especially evident when the EHR data has routine and substantial deviations from the more detailed event trajectory.

In machine learning, the problem of 'missingness' in data is common, and can be incorporated to improve estimation, and among other tasks in the medical field, such as survival analysis and disease prediction. With EHR data, the problem is similar in structure but opposite in literal meaning; EHR data is *absent* almost everywhere. However, its presence is not at random, and this temporal information can be used to improve medical prediction.

Acknowledgments

This work was supported partially by the National Science Foundation (IIS 1510741). The authors sincerely thank Christopher Newth and Robinder Khemani from Children's Hospital Los Angeles for suggesting the studying and making the data available.

References

Liefeng Bo and Cristian Sminchisescu. Twin gaussian processes for structured prediction. *International Journal of Computer Vision*, 87(1):28–52, 2010.

Phillip Boyle and Marcus Frean. Multiple output gaussian process regression. 2005.

Robert J Carroll, Anne E Eyler, and Joshua C Denny. Naïve electronic health record phenotype identification for rheumatoid arthritis. In *AMIA annual symposium proceedings*, volume 2011, page 189. American Medical Informatics Association, 2011.

Juan Ignacio Casse. *Automatic Co-Clustering for Social Network and Medical Data*. PhD thesis, University of California at Riverside, December 2014.

Noel Cressie. *Statistics for spatial data*. John Wiley & Sons, 1993.

Thomas Dean and Keiji Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5(3):142–150, 1989.

Thomas Desautels, Jacob Calvert, Jana Hoffman, Melissa Jay, Yaniv Kerem, Lisa Shieh, David Shimabukuro, Uli Chettipally, Mitchell D Feldman, Chris Barton, et al. Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach. *JMIR medical informatics*, 4(3), 2016.

Asela Gunawardana, Christopher Meek, and Puyang Xu. A model for temporal dependencies in event streams. In *Advances in Neural Information Processing Systems*, pages 1962–1970, 2011.

W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

William R Hersh, Mark G Weiner, Peter J Embi, Judith R Logan, Philip RO Payne, Elmer V Bernstam, Harold P Lehmann, George Hripcsak, Timothy H Hartzog, James J Cimino, et al. Caveats for the use of operational electronic health record data in comparative effectiveness research. *Medical care*, 51(8 0 3):S30, 2013.

Jennifer A Hoeting, David Madigan, Adrian E Raftery, and Chris T Volinsky. Bayesian model averaging: a tutorial. *Statistical science*, pages 382–401, 1999.

Peter B Jensen, Lars J Jensen, and Søren Brunak. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395, 2012.

Jau-Huei Lin and Peter J Haug. Exploiting missing clinical data in Bayesian network modeling for predicting medical problems. *Journal of biomedical informatics*, 41(1):1–14, 2008.

Uri Nodelman, Christian R Shelton, and Daphne Koller. Continuous time bayesian networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 378–387. Morgan Kaufmann Publishers Inc., 2002.

Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier, 2014.

Zhen Qin and Christian R Shelton. Auxiliary gibbs sampling for inference in piecewise-constant conditional intensity models. In *UAI*, pages 722–731, 2015.

Matthias Seeger, Yee-Whye Teh, and Michael Jordan. Semiparametric latent factor models. Technical report, 2005.

Anima Singh, Girish Nadkarni, Omri Gottesman, Stephen B Ellis, Erwin P Bottinger, and John V Guttag. Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration. *Journal of biomedical informatics*, 53:220–228, 2015.

Jeremy C Weiss. Piecewise-constant parametric approximations for survival learning. In *Machine Learning for Healthcare Conference*, pages 1–12, 2017.

Jeremy C Weiss and David Page. Forest-based point process for event prediction from electronic health records. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 547–562. Springer, 2013.

Jionglin Wu, Jason Roy, and Walter F Stewart. Prediction modeling using EHR data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, 48(6):S106–S113, 2010.

Di Zhao and Chunhua Weng. Combining pubmed knowledge and EHR data to develop a weighted Bayesian network for pancreatic cancer prediction. *Journal of biomedical informatics*, 44(5):859–868, 2011.