The Survival Filter: Joint Survival Analysis with a Latent Time Series

Rajesh Ranganath
Computer Science Dept.
Princeton University
Princeton, NJ 08540

Adler Perotte
Biomedical Informatics Dept.
Columbia University
New York, NY 10032

Noémie Elhadad
Biomedical Informatics Dept.
Columbia University
New York, NY 10032

David M. Blei
Computer Science Dept.
Statistics Dept.
Columbia University
New York, NY 10027

Abstract

Survival analysis is a core task in applied statistics, which models time-to-failure or time-to-event data. In the clinical domain, for example, meaningful events are defined as the onset of different diseases for a given patient. Survival analysis is limited, however, for analyzing modern electronic health records. Patients often have a wide range of diseases, and there are complex interactions among the relative risks of different events. To this end, we develop the survival filter model, a time-series model for joint survival analysis that models multiple patients and multiple diseases. We develop a scalable variational inference algorithm and apply our method to a large data set of longitudinal patient records. The survival filter gives good predictive performance when compared to two baselines and identifies clinically meaningful patterns of disease interaction.

1 INTRODUCTION

Electronic health records enable unprecedented opportunity to understand and form predictions about disease [Jensen et al., 2012, Hripcsak and Albers, 2013]. With historical data about the trajectories of millions of patients, we can learn patterns of disease risk and exploit these patterns to provide better care to future patients.

The classical statistics tool for analyzing the progression of a disease is survival analysis, a method that estimates each patients hazard or risk for a disease in question [Cox, 1972]. Survival analysis is widely used in medical science to characterize and understand the progression of individual diseases [Shepherd et al., 1995, Stupp et al., 2005].

Classical survival analysis, however, cannot accommodate the complex health data that we now have collected; it is only formulated to analyze one disease at a time. In modern electronic health record data, patients often have several diseases (called "comorbidities") with complex interactions among them. Specifically, the occurrence of one disease often affects the progression of others. We need new tools to account for this complexity.

Consider the data in Figure 1, where time is in the x-axis. The top panel shows the setting that classical survival analysis requires. All patients begin at the same time, and we measure one disease outcome. (In this case, it is whether the patient is diagnosed with diabetes.) The bottom panel illustrates the real-world setting of electronic health records. Patients begin at different times and we simultaneously measure many different diseases. These data can potentially reveal interactions between patterns of progression, but classical survival analysis cannot provide such inferences. For example, Patient 1 in the bottom panel illustrates that diabetes and hypertension are significant risk factors for developing a myocardial infarction. A traditional survival analysis may be constructed to capture this specific interaction, but cannot simultaneously capture the relationship between risk factors for diabetes (e.g., obesity) and the onset of diabetes.

We build on survival analysis to develop the survival filter, a new probabilistic model for estimating multivariate risk patterns from large-scale electronic health records. The survival filter is a latent-variable time series model of diagnostic codes. Each patient is represented as a sequence of latent variables. At each time point, a patient's hazards for each disease relates to his or her latent representation.

The survival filter can be thought of as a joint survival analysis model where each patient's sequence of latent representations loosely represents his or her state of health. Given a large data set of patients over time—data of the form of the bottom panel of Figure 1—survival filter inference characterizes each patient and captures complex global patterns of interactions for a large set of diseases.

Using the survival filter, we study 13,000 patients from NewYork-Presbyterian Hospital; these data span over 20 years and contain 8,800 types of diagnoses. It scales well to this size of data and uncovers meaningful relationships be-

tween diseases that would otherwise be difficult to identify.

2 SURVIVAL ANALYSIS

In this section we review survival analysis and hazards.

Survival Analysis. Survival analysis studies the time duration until the occurrence of an event. Example events include failure of a machine, heart attack, and retirement.

Observations in survival analysis have two types. The first type of observation indicates the event has occurred (called "failure") at a specific time (failure time). The second type indicates the event has not occurred before the observed time. These observations are called censored observations because the true failure time is censored in the observed data. Formally, the observations in survival analysis can be represented as pairs (t, c) where t is a time, and c is a binary value that indicates whether the observation is censored.

The simplest model for survival analysis assumes that the failure times are drawn from some unknown distribution F over positive values. The setup assumes that all observations are synchronized at their starts. Given this modeling choice, a nonparametric estimate of the CDF of F, also denoted by F, is the Kaplan-Meier estimator [Kaplan and Meier, 1958]. The Kaplan-Meier estimator is the nonparametric maximum likelihood estimator of $1 - F(t)$, also called the survival function, in the presence of censored data.

The time measurements in survival analysis can be treated as continuous or discrete (e.g., months or years). In this article we will focus discrete survival times.

Hazards. An alternative view of survival analysis is through hazard functions. Hazard functions represent the instantaneous chance of failing at time t given survival up to time t. In the discrete time setting, the hazard is the conditional probability of failing at time t given that the failure occurs at time t or later,

$$h(t) = P(T = t | T \geq t). \tag{1}$$

The Nelson-Aalen [Nelson, 1972] estimator forms the nonparametric maximum likelihood estimator of the sum of hazard function over time (cumulative hazard). The CDF F of the underlying distribution implied by the hazards is

$$F(t) = 1 - \exp\left(\sum_{s=0}^{t} h(s)\right).$$

Unlike the cumulative distribution function and survival function, we can specify the hazard function locally in each discrete time block by a number between zero and one. We will use this property when we develop the survival filter.

3 THE SURVIVAL FILTER FOR ELECTRONIC HEALTH RECORDS

In this section we first describe electronic health records and corresponding survival problems. We then describe our model, the survival filter.

Electronic Health Records. The Electronic Health Record (EHR) comprises all documentation entered for a patient throughout their interactions with a healthcare institution. It contains a wide range of observations through time, ranging from free-text notes authored by clinicians, medication orders, laboratory test results, procedures, demographic information, and diagnosis codes.

Diagnosis codes (also called billing codes) are structured codes from a standard taxonomy, namely the ICD9 hierarchy (International Classification of Diseases, 9th revision). ICD9 codes are used in all healthcare institutions. While the ICD9 hierarchy contains approximately 16,000 codes, in practice about 9,000 of them are commonly documented. After each visit, a clinician assigns each patient a set of ICD9 codes to reflect the diseases or concerns that were taken care of during the visit.

For instance, after a visit to their primary provider, a patient is assigned ICD9 codes for "Essential Hypertension," and "Diabetes". At their next visit to their ophthalmologist, the patient can have the ICD9 code for "Nonproliferative Diabetic Retinopathy." In this example, while the patient has diabetes at both visits, the ICD9 code for diabetes is only observed at the first visit. Furthermore, note that the ICD9 codes are correlated. Retinopathy, an eye condition, is a common complication for patients with diabetes, and hypertension and diabetes are known comorbidities.

EHR data are longitudinal records, represented as a collection of per-patient time series ICD9 codes. Note that an extremely sparse set of ICD9 codes will be used for any given patient.[1] Our goal is to use these data to run a joint survival analysis of every ICD9 code. Specifically, we seek a method that:

- estimates the per-patient risk for all ICD9 codes at any given visit time;
- handles multiple survival problems that are not aligned in time across patients;
- scales to 9,000 ICD9 codes;
- captures interaction between survival problems (e.g., retinopathy, diabetes, and hypertension).

This differs significantly from the setup of traditional survival analysis, where patients are forced to start at the same

[1] In our representation, we use "visit time," not clock time, as the time unit. Visit time better represents time when studying the temporal course of diseases [Hripcsak et al., 2015].

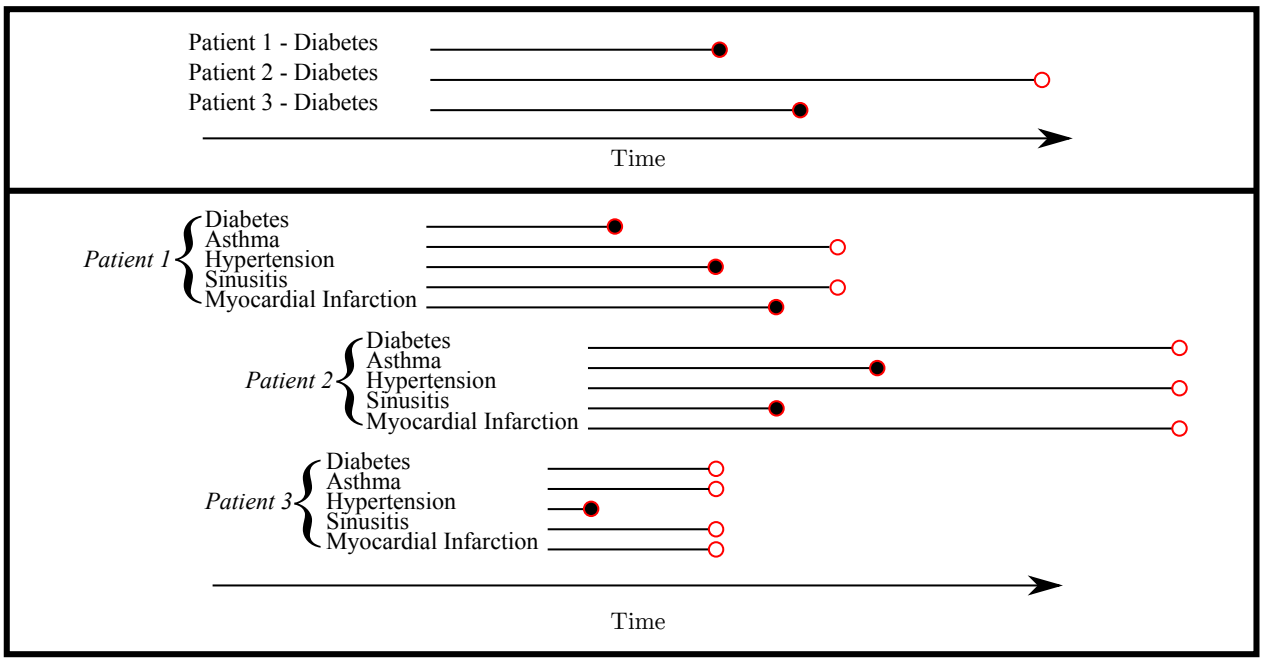

Figure 1: A comparison of standard survival analysis (top frame) and the survival filter (bottom frame). A filled circle represents an observed event, while an empty circle represents a censored one. In the case of standard survival analysis, patients in a cohort are aligned by an event. In the survival filter, patients are not aligned and unlike standard survival analysis, many conditions are considered simultaneously.

time and where we can only analyze a single disease. We now describe the survival filter, a model that addresses these goals to perform large-scale joint survival analysis of EHR.

Survival Filter. In the discrete time setting, let the observation pair (t, c) of time and censoring indicator be represented as a binary vector indexed by the clock with length equal to the observation time. This binary vector has a one in the last entry in the case of failure and is all zero in the case of censoring. We adopt this view of the observations for the survival filter.

Let P be the number of patients. Let n_p be the number of time intervals for patient p, and C be the number of codes. Then the observation $x_{p,t,c}$ is one if the code c is marked in the tth time interval for patient p and zero if code c has not yet occurred for patient p. To generate this data we propose a latent time series model where each patient has a K dimensional latent path which along with a $C \times K$ weight matrix W produces the hazard for each of the C codes. Let D be a distribution over the positive reals. The generative process for this model is

$$
\begin{aligned}
W &\sim D \\
z_{p,1} &\sim \text{Normal}(z_\mu, \sigma^2_{z_0}) \\
z_{p,t} &\sim \text{Normal}(z_{p,t-1}, \sigma^2_z) \\
x_{p,t,c} &\sim \text{Bernoulli}(1 - \exp(-W_c^\top \exp(z_{p,t}))).
\end{aligned}
$$

The hazard of a code c for patient p at a time t is $1 - \exp(-W_c^\top \exp(z_{p,t}))$. The positivity of W and $\exp(z_{p,t})$ guarantee that $1 - \exp(-W_c^\top \exp(z_{p,t}))$ is a valid hazard (probability) between zero and one. Larger $W_{c,k}$ indicate larger hazards for code c when factor k is active.

This model handles the criteria described above. First, it provides a simultaneous analysis of all ICD9 codes. Second, it handles misaligned patients by defining the hazards to be a function of the a shared latent space rather than a function of a fixed shared clock $h(t)$ (as in the classical setting). Third, through the matrix W it captures the relationship between different survival problems. Specifically, when $W_{c,k}$ and $W_{d,k}$ are large, the events c and d are more likely to co-occur. Finally, as we will show in Section 4, we can perform efficient computation for the survival filter.

We consider two different priors on W: the log-normal and the gamma. The gamma distribution is sparsifying when its shape is less than one (this can be seen from the PDF), while the log-normal distribution has a heavier tail. Recent results [Mimno et al., 2014] have shown that log-normal distributions achieve better predictive results and diversity among the weights in a Poisson network model. We derive inference and compare both the log-normal and gamma prior in the experimental section.

Related Work. Survival analysis methods have been generalized in many ways beyond the Kaplan-Meier estimator.

One example of such an extension is recurrent event models [Clayton, 1994] which allow for multiple events rather than single events. Cox proportional hazards [Cox, 1972] introduces fixed covariates to scale the patient hazards based on their covariates. Time varying Cox proportional hazard models [Fisher and Lin, 1999] are like Cox proportional hazards, but have a set of covariates that change with time for each patient. Cox-proportional hazard methods are similar to ours in that they define different clocks through the covariates, but differ in that they require covariates and do not capture the relationship between different survival problems. The model with the most similar goal is MEPSUM [Dean et al., 2014]. MEPSUM is a mixture model for multiple kinds of events happening simultaneously. The relationship between events is captured via the latent class label as each latent class contains a nonparametric hazard function for every type of event. Unlike our model, their model assumes that patients are synchronized in time when there are no covariates. Additionally the formulation of their model means that it scales with the number of codes rather than the number of failures, which makes it impractical in large sparse datasets such as those found in electronic health records.

4 INFERENCE

The main computational problem in working with the survival filter is computing the posterior distribution of the weights and latent patient trajectory. Computing the posterior of the survival filter analytically is intractable since our likelihood cannot be integrated out. Thus posterior computations require approximations. In this section, we develop a scalable mean field variational inference [Jordan et al., 1999] algorithm to approximate the posterior distribution of the survival filter.

Variational inference transforms the posterior inference problem into an optimization problem. The optimization problem defined by variational inference seeks to find a distribution q in an approximating family that is close in KL divergence to the posterior distribution. This is equivalent to maximizing the following [Bishop, 2006]:

$$\mathcal{L}(q) := \mathrm{E}_q[\log p(x, z, W) - \log q(z, W)].$$

This function is called the evidence lower bound (ELBO) as it forms a lower bound on $\log p(x)$.

Variational Approximation. Recall for the survival filter the latent variables are (1) z_{pt} for all latent states associated with patient p at time index t and (2) W, the matrix shared across observations. The joint distribution can be written as

$$\begin{aligned} p(x,z,W) = &\prod_{k=1}^{K}\prod_{c=1}^{C} p(W_{c,k}) \\ &\prod_{p=1}^{P} p(z_{p,1,k}) \prod_{c:a_{p,1}} p(x_{p,1,c} \mid z_{p,1}, W) \\ &\prod_{t=2}^{n_p}\prod_{k=1}^{K} p(z_{p,t,k} \mid z_{p,t-1,k}) \prod_{c:a_{p,t}} p(x_{p,t,c} \mid z_{p,t}, W). \end{aligned}$$

The mean field family posits a variational distribution where the latent variables are independent of each other. Each factorized q belongs to the same family as in the generative process. Formally, the approximating distribution is

$$q(z,W) = \prod_{k=1}^{K}\prod_{c=1}^{C} q(W_{c,k} \mid {\lambda_{c,k}}^0, {\lambda_{c,k}}^1)$$

$$\prod_{p=1}^{P}\prod_{t=1}^{n_p}\prod_{k=1}^{K} p(z_{p,t,k} \mid \mu_{p,t,k}, \sigma^2_{p,t,k}),$$

where $\lambda_{c,k0}$, $\lambda_{c,k1}$, $\mu_{p,t,k}$, and $\sigma_{p,t,k}$ are variational parameters. These variational parameters are then set via an optimization procedure to maximize the ELBO.

Classical Optimization. Typical optimization methods for mean field variational inference iteratively optimize the variational parameters associated with each latent variable by holding the others fixed. These update are easy to derive when the model's log complete conditional (the log of the distribution of each latent variable conditioned on the rest) has analytic expectation with respect to the variational approximation. This analytic property most commonly occurs in conditionally conjugate exponential families models where the complete conditional is in the exponential family [Ghahramani and Beal, 2001]. Unfortunately, none of the latent variables in our model fall into this class. Instead, we derive a variational algorithm based on sampling from the variational approximation [Salimans and Knowles, 2013, Kingma and Welling, 2014, Rezende et al., 2014, Ranganath et al., 2014, Titsias and Lázaro-Gredilla, 2014].

Sampling based Variational Inference. We briefly review stochastic optimization before discussing sampling based variational inference. In the following, we use λ as an example parameter. Let $\mathcal{L}(\lambda)$ be a function to be maximized and let $\hat{\nabla}_\lambda \mathcal{L}(\lambda)$ be a draw from a random variable whose expectation is the true gradient $\nabla_\lambda \mathcal{L}(\lambda)$. Let ρ_t be the learning rate, then stochastic optimization updates to the current parameter λ_t can be made with

$$\lambda_{t+1} = \lambda_t + \rho_t \hat{\nabla}_\lambda \mathcal{L}(\lambda).$$

Algorithm 1 Stochastic Variational Inference for the Survival Filter

Input: data X
Initialize λ randomly, $t = 1$.
repeat
  Sample a batch of datapoint $x_{1...B}$
  for b = 1...b in parallel **do**
    Use stochastic optimization with Eq. 4 to find the optimal $\mu_{b,v}$ and $\sigma_{b,v}$
  end for
  Compute the noisy global gradient for λ (Example: Eq. 8)
  Update λ with RMSProp
until change in validation metric is small

This update converges to a local maximum when the learning rate satisfies the Robbins Monro conditions

$$\sum_{t=1}^{\infty} \rho_t = \infty$$
$$\sum_{t=1}^{\infty} \rho_t^2 < \infty.$$

Stochastic optimization has become a widely used tool in variational inference.

Returning to variational inference, the variational objective is an expectation with respect to the variational approximation. Sampling based variational inference works by writing the gradient of the ELBO as an expectation followed by a stochastic optimization driven by Monte Carlo estimates of the gradient written as an expectation. The gradient as an expectation step comes in two main flavors: (1) those based on transformations and (2) those based on the score function (gradient of the log probability) of the variational approximation. We use both of these techniques to derive an inference algorithm for the survival filter (See the appendix for derivation details).

Algorithm. Algorithm 1 summarizes the parallelized stochastic variational inference algorithm we use to approximate posteriors of the variational approximation.

Scalability To scale to a large number of censored codes, we need to be able to efficiently compute the likelihood for a patient $p(x_p)$. Let $a_{p,t}$ be the set of codes that have not occurred before time t for patient p. Define $R_{p,t}$ as the relative log likelihood of the failed codes minus the previously failed codes:

$$R_{p,t} = \sum_{c:x_{p,t,c}=1} \log p(x_{p,t,c}=1) - \log p(x_{p,t,c}=0) - \sum_{c:[C]\setminus a_{p,t}} \log p(x_{p,t,c}=0).$$

Note that $R_{p,t}$ can be computed on the order of the number of failures for patient p

By the generative process, we have that the likelihood is

$$\begin{aligned}
\log p(x_p) &= \sum_{t=1}^{n_p} \sum_{c \in a_{p,t}} \log p(x_{p,t,c}) \\
&= \sum_{t=1}^{n_p} \sum_{c=1}^{C} \log p(x_{p,t,c}=0) + R_{p,t} \\
&= \sum_{t=1}^{n_p} \sum_{c=1}^{C} \log(1-(1-\exp(-W_c^\top \exp(z_{p,t})))) + R_{p,t} \\
&= \sum_{t=1}^{n_p} \sum_{c=1}^{C} -W_c \exp(z_{p,t}) + R_{p,t} \\
&= -\left(\sum_{c=1}^{C} W_c\right)\left(\sum_{t=1}^{n_p} \exp(z_{p,t})\right) + \sum_{t=1}^{n_p} R_{p,t}. \qquad (2)
\end{aligned}$$

This means that the likelihood for a patient can be computed in time $O((C+n_p)K + n_p s_p K)$ where s_p is the number of failures for patient p instead of of $O(Cn_pK)$. Thus the runtime scales with the number of uncensored codes rather than by the total number of codes. This efficiency in computing the likelihood will allow for the construction of efficient inference algorithms that scale with the number of failures in the data.

5 EMPIRICAL STUDY

In this section, we describe our experimental setup and results.

Datasets. Our dataset comprises the longitudinal records of 13,180 patients from a large, metropolitan healthcare institution, NewYorkPresbyterian Hospital. IRB approval was obtained for these experiments. The patient records contain documentation pertaining to all visit types, including outpatient visits, emergency department visits, as well as hospital admissions and intensive care stays (thus with varying ICD9 codes through time for a given patient). The only criteria to include patients in the dataset was at least 5 visits overall to the institutions and among them at least 3 to a primary provider care clinic. We truncated the longitudinal records of patients to 50 visits at most, and thus the mode of the visits was 50. Note that even though the time unit for our analysis is visit, the patient records actually have a wide range of durations (mean 14.5 years; std dev 8 years; median 15.5 years).

For the 13,180 patients, there were overall 8,722 unique ICD9 codes present in at least one visit. On average, each visit had 3.61 ICD9 codes assigned (std dev 2.28; median 3.05), and patients had an average of 189 (std dev 178; median 138) ICD9 codes in their longitudinal records, corre-

sponding to 57 unique codes on average (std dev 36; median 49).

Thus, our dataset represents a large set of patients with a wide range of conditions, as reflected by the large number of ICD9 codes in the dataset.

For our experiments, we held out 100 patient records for validation and parameter tuning and 1,000 patient records for testing purposes.

Evaluation Metrics. Standard evaluations in traditional survival analysis rely on concordance; essentially how well can the model rank patients according to the order in which the outcome is observed. This assumes a common clock, or t_0 for all patients, an assumption not held for the survival filter model. Instead, we propose the following three metrics: predictive log likelihood on held out data to assess model fitness, and two metrics well defined in the case of multiple, simultaneous survival analyses. All three metrics are computed by looking forward in the patient time series. We keep the approximate posterior of the shared weights from the training cohort fixed throughout testing.

The first metric computes the log likelihood of all ICD9 codes that have not yet occurred at each visit conditional on all the patient history prior to the visit. Thus, log likelihood is:

$$\log p(x_{c,p,t}) = - W_c^\top z_{p,t-1} \mathcal{I}(x_{c,p,t} = 0) + \log(1 - \exp(-W_c^\top z_{p,t-1})) \mathcal{I}(x_{c,p,t} = 1),$$

where I is the indicator function. For each patient in the test set, the predictive log likelihood is computed at each visit after the third visit. Procedurally, this means that we test at visit 4 conditioning on the first three visit, followed by testing at visit 5, conditioning on the first four, and so on.

The second metric computes the Mean Average Ranking of the codes that failed (i.e., first time observed) at visit t in the set of all ICD9 codes that have not yet failed (i.e., not yet observed) at that visit. The ICD9 ranks are computed by ordering the hazards the model assigns to each code at visit t based on the patient's latent state at visit $t-1$.

The third metric is Recall at D (in our experiments, D is set to 10). Recall at D computes how many failed codes (i.e., first time observed) are in the top D codes, as ordered by the hazards assigned by the model to each code.

Baselines We consider two baselines for this problem. The first baseline, which we call Mean Disease Risk, considers the frequency of ICD9 codes over the entire population. Given the training set of longitudinal records, a mean hazard is computed for each ICD9 code. Thus, this baseline outputs a fixed hazard prediction through time for any new patient visit.

The second baseline is patient specific, and is called Person Disease Risk. It computes a single hazard for all ICD9 codes based on the empirical frequency of failures at all previous visits. This baseline captures in essence the level of sickness of a patient, as sicker patients experience more code failures (i.e., observe more ICD9 codes).

Factor A	**Factor B**
Lumbargo	Depressive disorder
Osteoarthrosis	Anxiety state
Myalgia and myositis	Major depressive disorder, recurrent
Pain in joint	Major depressive disorder, single episode
Pain in limb	Dysthymic disorder
Backache	Adjustment disorder
Arthropathy	Unknown cause of morbidity or mortality
Pain in joint involving lower leg	Panic disorder without agoraphobia
Cervicalgia	Unspecified personality disorder
Pain in joint involving shoulder region	Palpitations
Factor C	**Factor D**
HIV counseling	Headache
Pregnant state, incidental	Dizziness and giddiness
Vaginitis	Migraine
Special gynecological examination	Disorder of optic nerve and visual pathways
Routine gynecological examination	Visual field defect
Counseling and advice on contraception	Unspecified endocrine disorder
Mother with single liveborn	Cushing's syndrome
Supervision of other normal pregnancy	Optic atrophy
Normal delivery	Neoplasm of endocrine glands
Leiomyoma of uterus	Visual discomfort

Figure 3: Example factors for the survival filter represented by the ICD9 codes with highest hazard for each factor.

Hyperparameters. We set the prior variance on the initial state of the latent trajectories to 10 and the prior mean -3. The large variance accounts for the fact that patient's records start at different points. We set the transition variance to .1. For log-normal weights we set the log scale to $\log(10^{-10})$, and the shape to 30. This distribution places a lot of mass near zero. To encourage sparsity of the gamma weights, we set the shape to .02 and the rate to 0.3.

RMSProp also contains a scaling parameter. For the local maximization step we use a decreasing schedule given by $(1+t)^{-.8}$ and for the global gradients we set the constant to .1.

We explore several different sizes for the latent space ranging from $K = 5$ to $K = 100$.

Results. Figure 2 plots the evaluation metrics as a function of K for the log-normal model. We find that the model with $K = 25$ does best with the best predictive log likelihood and a nearly best performance on Mean Average Ranking. All of the models outperform the plotted mean disease risk baseline on all metrics. We find that the gamma models performs worse than the log-normal model for all K with a best test log likelihood of -284874. Finally, all of the models outperform the person disease risk baseline on log likelihood (-359079).

Figure 3 displays four of the factors found by the log-normal survival filter with $K = 25$. These components represent clinically meaningful groups of conditions.

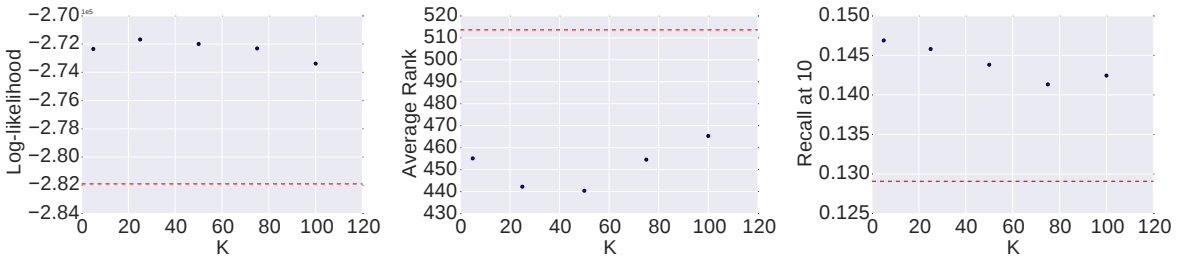

Figure 2: The survival filter (blue dots) outperforms the mean disease risk baseline (dashed red line) for all values of K on all metrics.

6 DISCUSSION

In this paper, we have developed the survival filter, a latent timeseries model for joint survival analysis. The main advantages of the survival include jointly modeling time-to-event data for a large set of events and without specifying alignment across individuals, and an efficient mean field variational inference algorithm that scales in the number of events. We demonstrated the use of the survival filter by measuring the predictive likelihood on a real-world clinical data set and demonstrate superior performance relative to baselines and interpretable latent factors.

The survival filter is a general joint survival analysis model and can be used to study survival problems beyond those in electronic health records. It can be used in any situation where there are multiple simultaneous surival problems that are not necessarily aligned by a true clock. For example, the survival filter can be used to make movie recommendations. In this setting the codes are movies, and the patients are the users. Here, failure of a particular code at time t means that a movie was watched at time t and the hazards capture the chance that a movie is watched by a user at time t.

Acknowledgements Rajesh Ranganath is supported by an NDSEG fellowship. David M. Blei is supported by NSF BIGDATA NSF IIS-1247664, ONR N00014-11-1-0651, and DARPA FA8750-14-2-0009. Noémie Elhadad and Adler Perotte are supported by NSF IIS-1344668.

7 APPENDIX

Sample-based gradients Transformation based approaches [Kingma and Welling, 2014, Rezende et al., 2014, Titsias and Lázaro-Gredilla, 2014] write the ELBO as an expectation with respect to the a standard distribution without variational parameters and moves the differential operator inside of the expectation. Formally let $r(y)$ be a standard distribution and let T be a transformation such that $T(y, \lambda)$ is distributed as q_λ, then the ELBO can be written as

$$\mathcal{L}(\lambda) = \mathrm{E}_r[\log p(x, T(y, \lambda)) - \log q(T(y, \lambda))].$$

When T and the model and approximation are differentiable the gradient of the ELBO is given by the ELBO as

$$\nabla_\lambda \mathcal{L}(\lambda) = \mathrm{E}_r[\nabla_z[\log p(x, T(y, \lambda)) - \log q(T(y, \lambda))]\nabla_\lambda T].$$

See Kingma and Welling [2014] for a complete derivation of this identity. In our use below, r will be the standard normal transformation and T will be $z = \sigma y + \mu$.

Score function based approaches [Ranganath et al., 2014, Mnih and Gregor, 2014] are based on the following identity

$$\nabla_\lambda \mathcal{L}(\lambda) = \mathrm{E}_q[\nabla_\lambda \log q(z|\lambda)(\log p(x, z) - \log q(z))]. \tag{3}$$

See Ranganath et al. [2014] for a derivation of this.

The convergence time of stochastic optimization is related to its noise, so in the sequel we derive analytically when possible.

Gradient for $q(z_{p,t})$. The variational approximation $q(z_{p,t,k}|\mu_{p,t,k}, \sigma^2_{p,t,k})$ is a normal distribution with mean $\mu_{p,t,k}$ and variance $\sigma^2_{p,t,k}$. The gradient for the variational parameters can be mostly computed analytically.

$$\begin{aligned}
\nabla_{\mu_{p,t,k}} \mathcal{L} &= -\exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k}) \sum_{c=1}^{C} \mathrm{E}[W_c] \\
&\quad - \frac{1}{\sigma_z^2}(2\mu_{p,t,k} - \mu_{p,t-1,k} - \mu_{p,t+1,k}) \\
&\quad + \nabla_{\mu_{p,t,k}} \mathrm{E}[R_{p,t}], \\
\nabla_{\sigma^2_{p,t,k}} \mathcal{L} &= -\frac{1}{2}\exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k}) \sum_{c=1}^{C} \mathrm{E}[W_c] \\
&\quad - \frac{1}{\sigma_z^2} - \nabla_{\sigma^2_{p,t,k}} \mathrm{E}[R_{p,t}] + \frac{1}{2\sigma^2_{p,t,k}}.
\end{aligned} \tag{4}$$

$\mathrm{E}[W_c]$ can be computed analytically for both the gamma and log-normal distribution. Recall the definition of $R_{p,t}$

$$\begin{aligned}
R_{p,t} = &\sum_{c:x_{p,t,c}=1} \log p(x_{p,t,c} = 1) - \log p(x_{p,t,c} = 0) \\
&- \sum_{c:[C]\setminus a_{p,t}} \log p(x_{p,t,c} = 0).
\end{aligned}$$

Its expected value is

$$\begin{aligned}\mathrm{E}_q[R_{p,t}] = &\sum_{c:x_{p,t,c}=1} \mathrm{E}[\log p(x_{p,t,c}=1)] \\ &+ \mathrm{E}[W_c]^\top \mathrm{E}[\exp(z_{p,t})] \\ &+ \sum_{c:[C]\setminus a_{p,t}} \mathrm{E}[W_c]^\top \mathrm{E}[\exp(z_{p,t})].\end{aligned}$$

Its derivative with respect to its mean is computed by the transformation approach using the identity $z = y\sqrt{\sigma^2_{p,t,k}} + \mu_{p,t,k}$ where y is drawn from a standard normal $\mathcal{N}$.

$$\begin{aligned}&\nabla_{\mu_{p,t,k}} \mathrm{E}_q[R_{p,t}] \\ &\quad= \sum_{c:x_{p,t,c}=1} \mathrm{E}_{q(W_c)} \mathrm{E}_{y\sim\mathcal{N}} \left[\frac{W_{c,k}\exp(z)}{\exp(-W_c^\top \exp(z)) - 1}\right] \\ &\quad+ \mathrm{E}[W_{c,k}] \exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k}) \\ &\quad+ \sum_{c:[C]\setminus a_{p,t}} \mathrm{E}[W_{c,k}] \exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k}).\end{aligned}$$

Similarly it's derivative with respect to the variance is given as

$$\begin{aligned}&\nabla_{\sigma^2_{p,t,k}} \mathrm{E}_q[R_{p,t}] \\ &\quad= \frac{1}{2}\Big(\sum_{c:x_{p,t,c}=1} \mathrm{E}_{q(W_c)} \mathrm{E}_{y\sim\mathcal{N}} \left[\frac{yW_{c,k}\exp(z)}{\exp(-W_c^\top \exp(z)) - 1}\right] \\ &\quad+ \mathrm{E}[W_{c,k}] \exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k}) \\ &\quad+ \sum_{c:[C]\setminus a_{p,t}} \mathrm{E}[W_{c,k}] \exp(\mu_{p,t,k} + \frac{1}{2}\sigma^2_{p,t,k})\Big).\end{aligned}$$

We can compute noisy, unbiased estimates of this gradient by sampling from the variational approximation for $W_{c,k}$ and sampling y from the standard normal.

The gradient can be computed in time $O((C + n_p)K + n_p s_p K)$ rather than $O(Cn_pK)$. The only portion of the gradient that we cannot compute analytically is the portion associated with failing codes. This portion requires sampling from the variational approximation. This means we can exploit the sparsity of the failures as we only have to sample small fractions of the W matrix rather than the entire W matrix. This results in an order of magnitude less samples from the underlying random generator for each noisy gradient and produces lower variance gradients than sampling entirely.[2]

The gradient of the variational parameters of the first and last point can be expressed similarly.

[2]Variance can be a problem in sampling based variational approximations [Kingma and Welling, 2014, Rezende et al., 2014, Ranganath et al., 2014].

Gradients for Log-Normal W. Similar to the time series $z_{p,t}$, the only component of the gradient of the variational parameters of W that is not analytically tractable is due to the failures. To symmetrize this update with the latent time series, we represent the log-normal distribution as an exponentiated normal. That is $W_{c,k} = \exp(\tilde{W}_{c,k})$, where $\tilde{W}_{c,k}$ is normally distributed with mean μ_w and variance σ^2_w. In this setup, the variational approximation for $\tilde{W}_{c,k}$ is normally distributed with variational parameters $\lambda^0_{c,k}$ (the mean) and $\lambda^1_{c,k}$ (the variance). The gradient for the mean of the variational approximation is given by

$$\nabla_{\lambda^0_{c,k}} = -\frac{1}{\sigma^2_w}(\mu_w - W_{c,k}) \tag{5}$$

$$- \exp(\lambda^0_{c,k} + \frac{1}{2}\lambda^1_{c,k}) \sum_{p=1}^{P}\sum_{v=1}^{n_p} \mathrm{E}[\exp(z_{p,t,k})] \tag{6}$$

$$+ \nabla_{\lambda^0_{c,k}} \mathrm{E}[R_{p,t}]. \tag{7}$$

and the gradient of the variance is

$$\begin{aligned}\nabla_{\lambda^0_{c,k}} = &-\frac{1}{2\sigma^2_w} \\ &- \exp(\lambda^0_{c,k} + \frac{1}{2}\lambda^1_{c,k}) \sum_{p=1}^{P}\sum_{v=1}^{n_p} \frac{1}{2}\mathrm{E}[\exp(z_{p,t,k})] \\ &+ \nabla_{\lambda^1_{c,k}} \mathrm{E}[R_{p,t}] + + \frac{1}{2\lambda^1_{c,k}}.\end{aligned}$$

The gradients of R are symmetric to the time series updates for both the mean and variance parameter.

Note that the gradient can be computed in time proportional to the number of failures ($O((C + n_p)K + \left(\sum_{p=1}^{P} n_p s_p\right) K)$) rather than the number of codes multiplied by the number of visits ($O(CK\sum_{p=1}^{P} n_p)$) as sum of $\mathrm{E}[\exp(z_{p,t,k})]$ across all patients and visits can be shared for each code.

Gradients for Gamma W Finally, we consider the gradient of the parameters of the variational approximation of W when W is drawn from a gamma distribution in the generative process. In this setup, the variational approximation for each entry in the weight matrix is gamma distributed with shape $\lambda^0_{c,k}$ and scale $\lambda^1_{c,k}$. Similarly the only part of the gradient for this approximation that cannot be computed analytically is due to the failures. The gradient with respect to the shape is

$$\begin{aligned}\nabla_{\lambda^0_{c,k}} = &-\beta_w \lambda^1_{c,k} + (\alpha_w - 1)\Psi^{(1)}(\lambda^0_{c,k}) + 1 \\ &+ (1 - \lambda^0_{c,k})\Psi^{(1)}(\lambda^0_{c,k}) \\ &- \lambda^1_{c,k} \sum_{p=1}^{P}\sum_{v=1}^{n_p} \frac{1}{2}\mathrm{E}[\exp(z_{p,t,k})] \\ &+ \nabla_{\lambda^0_{c,k}} \mathrm{E}[R_{p,t}].\end{aligned}$$

where Ψ is the digamma function and $\Psi^{(1)}$ is its derivative. The gradient with respect to the scale parameter is

$$\begin{aligned}\nabla_{\lambda^1_{c,k}} &= \beta_w \lambda^0_{c,k} + \frac{(\alpha_w - 1)}{\lambda^1_{c,k}} + \frac{1}{\lambda^0_{c,k}} \\ &- \lambda^0_{c,k} \sum_{p=1}^{P} \sum_{v=1}^{n_p} \frac{1}{2} \mathrm{E}[\exp(z_{p,t,k})] \\ &+ \nabla_{\lambda^1_{c,k}} \mathrm{E}[R_{p,t}].\end{aligned}$$

Similar to the log-normal case this gradient scales with number of failures, and the $\log p(x = 0)$ terms can be computed analytically.

Rather than using transformations to compute gradient of the expected value of $\log p(x = 1)$, we use score style gradients. We already know how to evaluate $\log p(x = 1)$, so the all we need to compute the gradient is the score function of the gamma distribution for both the shape α and the scale κ. The score function of the shape is

$$\nabla_{\lambda^0_{c,k}} \log q(w_{c,k}) = -\log(\lambda^1_{c,k}) - \Psi(\lambda^0_{c,k}) + \log(w_{c,k})$$

The score function of the scale is

$$\nabla_{\kappa} \log q(w_{c,k}) = -\frac{\lambda^0_{c,k}}{\lambda^1_{c,k}} + \frac{w_{c,k}}{{\lambda^1_{c,k}}^2}.$$

Plugging this into Eq. 3 and approximating the expectation by Monte Carlo yields a noisy gradient of the ELBO.

Data subsampling. Given the noisy gradients just derived, we can use stochastic optimization to maximize the ELBO. This procedure is inefficient in that every single observation has to be iterated over in order to compute the gradient of the variational parameters for shared W. Another way this procedure is computationally wasteful is that at early iterations work done on all of the local parameters is based on the randomly initialized variational parameters for shared structure. These inefficiencies can be prohibitive when dealing with large datasets or large data instances.

Stochastic variational inference (SVI) [Hoffman et al., 2013] addresses this by using stochastic optimization. SVI works by first identifying local parameters, latent variables associated with a datapoint, and global parameters, shared latent variables. Next a datapoint is sampled, the optimal variational distribution for the local parameters is computed based on the current value of the global parameters, and a noisy gradient based on the sampled data point is computed. SVI generalizes this to drawing batches of datapoints rather than drawing a single datapoint at each update.

In the survival filter the global parameters are the weights and the local parameters are the latent trajectory. For a fixed variational approximation on W, we compute the optimal local variational parameters by running a stochastic optimization procedure with noisy gradient given by Eq. 4. An example global gradient for log-normal weights is given by

$$\begin{aligned}&-\frac{1}{\sigma_w}(\mu_w - W_{c,k}) \\ &\quad - \frac{P}{B}(\exp(\lambda^0_{c,k} + \frac{1}{2}\lambda^1_{c,k}) \sum_{b=1}^{B} \sum_{v=1}^{n_{p_b}} \mathrm{E}[\exp(z_{p_b,t,k})] \\ &\quad + \nabla_{\lambda^0_{c,k}} \mathrm{E}[R_{p_b,t}]). \end{aligned} \tag{8}$$

where we have reweighted the data term to maintain unbiasedness of the gradient.

Our approach differs from standard SVI in that we use stochastic optimization to compute the optimal local variational parameters. This approach allows differs from the double stochastic approaches in sampling based variational methods [Titsias and Lázaro-Gredilla, 2014, Ranganath et al., 2014] in that we do run a complete maximization procedure for each data point that is sampled rather than simply follow a noisy gradient. This maximization step can be time consuming, so we find the optimal local variational approximation in parallel.

Learning Rates. The standard robbins monro learning rate can be challenging to set in that it does not account for varying length scales or different amounts of noise in each gradient of the coordinate. We instead use RMSProp[3] which scales the gradient by the square root of a running average of the squared gradient. This handles varying length scales as multiplying the objective by a constant does not change the step. RMSProp controls for noise as the moving average of the squared gradient is larger when the variance of the gradient is larger.

References

C. Bishop. *Pattern Recognition and Machine Learning*. Springer New York., 2006.

D. Clayton. Some approaches to the analysis of recurrent event data. *Statistical Methods in Medical Research*, 3 (3):244–262, 1994.

D. R. Cox. Regression models and like-tables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 43(2):187–220, 1972.

D. O. Dean, D. J. Bauer, and M. J. Shanahan. A discrete-time multiple event process survival mixture (MEPSUM) model. *Psychological methods*, 19(2):251, 2014.

L. D. Fisher and D. Y. Lin. Time-dependent covariates in the Cox proportional-hazards regression model. *Annual review of public health*, 20(1):145–157, 1999.

Z. Ghahramani and M. Beal. Propagation algorithms for variational Bayesian learning. In *NIPS 13*, pages 507–513, 2001.

[3] www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

M. Hoffman, D. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(1303–1347), 2013.

G. Hripcsak and D. J. Albers. Next-generation phenotyping of electronic health records. *Journal of the American Medical Informatics Association*, 20(1):117–121, 2013. ISSN 1067-5027. doi: 10.1136/amiajnl-2012-001145.

G. Hripcsak, D. J. Albers, and A. Perotte. Parameterizing time in electronic health record studies. *Journal of the American Medical Informatics Association*, 2015.

P. B. Jensen, L. J. Jensen, and B. Søren. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews. Genetics*, 13(6):395–405, 2012. ISSN 1471-0056. doi: 10.1038/nrg3208.

M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. Introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

D. Kingma and M. Welling. Auto-encoding variational Bayes. *International Conference on Learning Representations*, 2014.

D. Mimno, G. Gopalan, and D Blei. Necessary evil or first choice? Non-conjugate priors and Poisson community models. In *NIPS Workshop on Variational Inference*, 2014.

A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. In *ICML*, 2014.

W. Nelson. Theory and applications of hazard plotting for censored failure data. *Technometrics*, 14(4):945–966, 1972.

R. Ranganath, S. Gerrish, and D. Blei. Black box variational inference. In *International Conference on Artifical Intelligence and Statistics*, 2014.

D. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *ArXiv e-prints*, January 2014.

T. Salimans and D. Knowles. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

J. Shepherd, S. M. Cobbe, I. Ford, C. G. Isles, A. R. Lorimer, P. W. Macfarlane, J. H. McKillop, and C. J. Packard. Prevention of coronary heart disease with pravastatin in men with hypercholesterolemia. *New England Journal of Medicine*, 333(20):1301–1308, 1995. doi: 10.1056/NEJM199511163332001. PMID: 7566020.

R. Stupp, W. P. Mason, M. J. van den Bent, M. Weller, B. Fisher, M. J. B. Taphoorn, K. Belanger, A. A. Brandes, C. Marosi, U. Bogdahn, J. Curschmann, R. C. Janzer, S. K. Ludwin, T. Gorlia, A. Allgeier, D. Lacombe, J. G. Cairncross, E. Eisenhauer, and R. O. Mirimanoff. Radiotherapy plus concomitant and adjuvant temozolomide for glioblastoma. *New England Journal of Medicine*, 352(10):987–996, 2005. doi: 10.1056/NEJMoa043330. PMID: 15758009.

M. Titsias and M. Lázaro-Gredilla. Doubly stochastic variational Bayes for non-conjugate inference. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1971–1979, 2014.