Lecture 4: Variational timeseries analysis and structured variational inference

In the previous lectures I introduced the general theory of stochastic variational inference. I will now discuss how to apply this theory to structured timeseries problems, where the latent variables are organized in time. These models are extremely important in many fields of science and engineering as they describe how the degrees of freedom of a system of interest evolve in time. For example, the Newton law

$$m\dot{\boldsymbol{v}}(t) = F(\boldsymbol{x}(t), \boldsymbol{v}(t))$$

defines the (deterministic) dynamics of a physical system. Stochasticity in physical systems arise when not all degrees of freedom are included in the model. In this case, the degrees of freedom that are not explicitly modeled are accounted using a stochastic perturbation term. The most common example of this phenomenon is given by the motion of a asteroid, where erratic random behavior arise from the collisions with a very large number of air particles. I will start this chapter with a simple everyday example where the need of timeseries models arise naturally. I will then introduce a general framework for variational inference in timeseries models.

1 Getting your bearings

Imagine to be lost in a city that you never visited. Since you cannot recognize any of the buildings around you, you need to rely on your Google map app on your phone. Unfortunately, the connection is unreliable and the localization is therefore very erratic. We denote your location at time t as x_t and the GPS localization given by the app as y_t. Both these quantities should be two-dimensional vectors since the location is expressed by two coordinates, however, in the sake of simplicity we will assume them to be one-dimensional. We can model the noisy measurements using a Gaussian model with your true location as mean:

$$\rho(y_t \mid x_t) = \mathcal{N}(y_t; x_t, \sigma_{lk}^2) \ . \tag{1}$$

How can you use these measurements to infer your position? The simplest choice is to use a maximum likelihood estimation at each time point:

$$\hat{x}_t = \text{argmax}_{x_t} \log \rho(y_t \mid x_t) \tag{2}$$

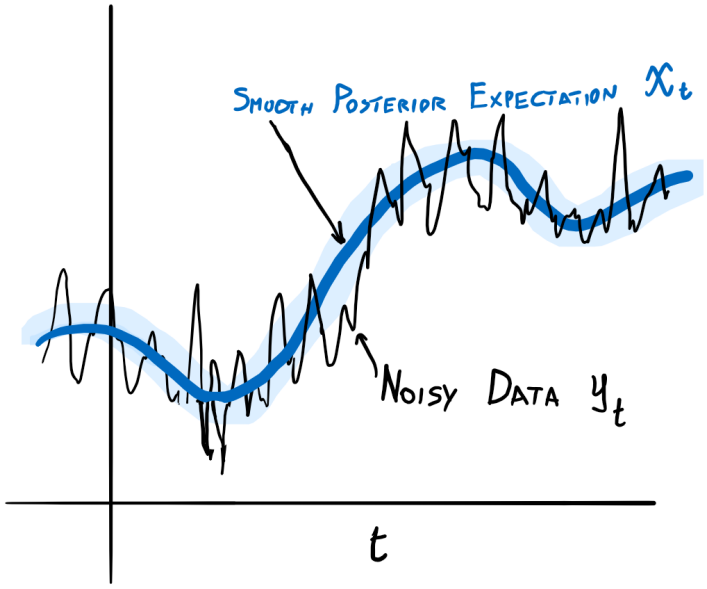

Figure 1: Bayesian smoothing.

it is easy to see that this estimate is simple equal to the measurement y_t. While this is a very intuitive solution, it is not able to denoise the measurements and it is therefore rather unreliable when σ_{lk}^2 is large. Using a Bayesian approach, you can reduce the effect of measurement noise in the estimate of your position by using an appropriate prior. Since the position changes as you move around, the prior should be expressed as the probability of your current location given your past location:

$$p(x_t \mid x_{t-1}). \tag{3}$$

In theory, this prior transition model could be immensely complicated involving your brain processing in deciding where to go, the translation of these brain signals in your muscles and so on. However, you can achieve a very satisfactory level of denoising without solving the fundamental problems of neuroscience by using a very simplified models that still capture the basic features of your motion. The most basic feature of human motion is its continuity. Your location changes gradually to nearby locations without sudden teletransportations. We can model this continuity using a Gaussian random walk model:

$$p(x_t \mid x_{t-1}) = \mathcal{N}(x_t; x_{t-1}, \mathrm{d}x^2) \ . \tag{4}$$

where $\mathrm{d}x$ is the average spatial increment. This model implies that you move at random locations with average velocity $v = \mathrm{d}x/\mathrm{d}t$, where $\mathrm{d}t$ is the time increment. This model is extremely simplistic as it assumes that you change direction at time point in a completely random direction. However, the assumption of continuity

encoded in the model is enough for providing a satisfactory level of denoising in most situations. Now assume that you know that you were in a known location x_0 (let say the train station) at time t. Since we assumed that each transition is conditionally independent from each other, we can write down the probability density of both positions and measurements as follows:

$$\begin{aligned} p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}) &= \prod_{t=1}^{T} \mathcal{N}(x_t; x_{t-1}, \mathrm{d}x^2)\mathcal{N}(y_t; x_t, \sigma_{lk}^2) \ , \\ &\propto \prod_{t=1}^{T} e^{-(x_t - x_{t-1})^2/2\mathrm{d}x^2 - (x_t - y_t)^2/2\sigma_{\mathrm{lk}}^2} \ . \end{aligned} \tag{5}$$

We you can see, the prior model links the measurements together since the variable x_t is influenced by the variable x_{t-1} which is measured as well. This allows the effect of each measurement to "spread" through the timeseries, thereby increasing the amount of data available at each time point and therefore denoising the estimation. If we are currently at time T, the Bayesian estimate of your current and past locations is captured by the posterior

$$p(\boldsymbol{x}_{1:T} \mid \boldsymbol{y}_{1:T}) \propto p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}) \propto \prod_{t=1}^{T} e^{-(x_t - x_{t-1})^2/2\mathrm{d}x^2 - (x_t - y_t)^2/2\sigma_{\mathrm{lk}}^2} \ . \tag{6}$$

This is also known as the *smoothing distribution*. The terminology comes from the fact that the latent timeseries is a smoothed version of the sequence of noisy data $\boldsymbol{y}_{1:T}$. The smoothing behavior follows from the statistical dependencies in the prior dynamical model, which "spreads" the effect of a data-point y_t to the surrounding timepoints, thereby smoothing the final estimate. This is visualized in Fig. 1. In this particular case, this is a multivariate Gaussian model which can be solved in closed form. However, we will now see how to obtain an approximate solution using stochastic variational inference.

1.1 Estimating the ELBO

The first step is to define a parameterized variational approximation over the latent variables. The simplest approach is to model the location at each time point using an independent Gaussian variable. This results in the Gaussian mean field approximation:

$$q(\boldsymbol{x}_{1:T}; \boldsymbol{\mu}_{1:T}, \boldsymbol{\sigma}_{1:T}) = \prod_{t=1}^{T} \mathcal{N}(x_t; \mu_t, \sigma_t^2) \ , \tag{7}$$

where $\boldsymbol{\mu}_{1:T}$ and $\boldsymbol{\sigma}_{1:T}$ are arrays containing the variational parameters at all time points. We now need to train those parameters to fit the true posterior distribution. As you surely know at this point, we can do this by using the

ELBO as loss function:

$$\mathrm{ELBO}(\boldsymbol{\mu}_{1:T}, \boldsymbol{\sigma}_{1:T}) = \mathbb{E}_{\boldsymbol{x}_{1:T}\sim q}\left[\log \frac{p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T})}{q(\boldsymbol{x}_{1:T}; \boldsymbol{\mu}_{1:T}, \boldsymbol{\sigma}_{1:T})}\right] , \tag{8}$$

$$\propto \mathbb{E}_{\boldsymbol{x}_{1:T}\sim q}\left[\sum_{t=1}^{T}\left(\underbrace{-\frac{(x_t - x_{t-1})^2}{2\mathrm{d}x^2}}_{\text{log prior}}\underbrace{-\frac{(x_t - y_t)^2}{2\sigma_{\mathrm{lk}}^2}}_{\text{log likelihood}}+\underbrace{\frac{(x_t - \mu_t)^2}{2\sigma_t^2} + \log \sigma_t^2}_{\text{variational density}}\right)\right] .$$

We now need to obtain a Monte Carlo estimator of the gradient of this loss. Since everything is differentiable and the Gaussian distribution can be re-parameterized, we can use the re-parameterization trick. For each time point, we express x_t as a function of its variational parameters and a fixed random variable:

$$x_t = \mu_t + \sigma_t \epsilon_t , \tag{9}$$

where as usual ϵ_t follows a a standard normal distribution. We can now re-express the ELBO as an average over the variables ϵ_t:

$$\mathbb{E}_{\boldsymbol{\epsilon}_{1:T}}\Bigg[\sum_{t=1}^{T}\Bigg(-\frac{((\mu_t - \mu_{t-1}) + (\sigma_t\epsilon_t - \sigma_{t-1}\epsilon_{t-1}))^2}{2\mathrm{d}x^2} - \frac{(\mu_t - y_t + \sigma_t\epsilon_t)^2}{2\sigma_{\mathrm{lk}}^2} + \frac{\epsilon_t^2}{2}\Bigg)\Bigg] + \frac{1}{2}\log\sigma_t^2 . \tag{10}$$

Note that the $\epsilon_t^2/2$ terms can be neglected since it does not depend on the variational parameters. We can now obtain a stochastic estimate of the gradient by sampling N values of each ϵ_t variable from standard normal distributions, replace the expectation with the finite average over the samples and compute the derivatives of this finite expression.

1.2 Gradients and closed form solution

In practice, the re-parameterized expression of the ELBO in Eq. 10 is everything we need to compute since the gradients can be computed using automatic differentiation packages. However, in this case the gradients are analytically tractable and we can gain further insight into the nature of our variation mean field approximation by computing them explicitly. We start by computing the gradients with respect to the variance parameters. First of all, we notice that

$$\begin{aligned}\mathbb{E}\left[(\sigma_t\epsilon_t - \sigma_{t-1}\epsilon_{t-1})^2\right] &= \mathbb{E}\left[\sigma_t^2\epsilon_t^2 - 2\sigma_t\sigma_{t-1}\epsilon_t\epsilon_{t-1} + \sigma_{t-1}^2\epsilon_{t-1}^2\right] \\ &= \sigma_t^2\underbrace{\mathbb{E}\left[\epsilon_t^2\right]}_{=1} - 2\sigma_t\sigma_{t-1}\underbrace{\mathbb{E}[\epsilon_t\epsilon_{t-1}]}_{=0} + \sigma_{t-1}^2\underbrace{\mathbb{E}\left[\epsilon_{t-1}^2\right]}_{=1} \\ &= \sigma_t^2 + \sigma_{t-1}^2\end{aligned} \tag{11}$$

since ϵ_t and ϵ_{t-1} are uncorrelated. Therefore, the ELBO decouples into a sum of terms each depending only to the variance at a specific time point. Using this

result, we can easily compute the exact derivative (try it!), which results in the following expression

$$\frac{\partial \text{ELBO}}{\partial \sigma_t^2} = -\frac{1}{\mathrm{d}x^2} - \frac{1}{2\sigma_{\text{lk}}^2} + \frac{1}{2\sigma_t^2} \ . \tag{12}$$

We can now obtain a closed form solution for the variational variances by setting the gradient equal to zero. This result in the formula

$$\sigma_t^2 = \frac{1}{(\mathrm{d}x/\sqrt{2})^{-2} + \sigma_{lk}^{-2}} \ . \tag{13}$$

This expression is formally identical to the exact posterior variance of a univariate Gaussian inference with prior variance $\mathrm{d}x/2$ and likelihood variance σ_{lk}^2. The factor of two in the prior term follows from the fact that prior information comes from the two neighboring time points $t-1$ and $t+1$. From this result, we can already see that the posterior variance is underestimated since the expression assumes that the variance of each side of the prior is $\mathrm{d}x$. Remember that $\mathrm{d}x$ is the average change in position during one time step. Therefore, the variational mean field formula for the variance implicitly assumes the two positions x_{t+1} and x_{t-1} to be known exactly since the only source of variability in the prior is the assumed to be the average movement from those locations. In reality, these two locations are themselves estimated from the data and the resulting prior variance should be substantially higher when the measurements are unreliable. Note that, at the limit σ_{lk}^2, the approximate posterior variance converges to the relatively small value $(\mathrm{d}x/\sqrt{2})^2$ which is usually much lower than the exact posterior variance. For this reason, the mean field approach is rarely appropriate in timeseries analysis.
Computing the gradient with respect to the mean leads to the equation

$$\mu_t = \frac{(\mathrm{d}x/\sqrt{2})^{-2}}{(\mathrm{d}x/\sqrt{2})^{-2} + \sigma_{lk}^{-2}} \frac{1}{2}(\mu_{t-1} + \mu_{t+1}) + \frac{\sigma_{lk}^{-2}}{(\mathrm{d}x/\sqrt{2})^{-2} + \sigma_{lk}^{-2}} y_t \ . \tag{14}$$

This expression is again analogous to the close form solution with the prior mean being the average of the variational means at $t-1$ and $t+1$. Note that this is not a closed form solution since each mean parameter (except for the ones at 0 and T) appears in three different equations. However. the resulting linear system of equation can be solved using standard methods.

2 Hidden Markov timeseries models

The random walk model we discussed in the previous section is a special case of hidden Markov model (HMM). In a HMM, the probability distribution of latent variables at each time points are solely a function of the value at the previous time point and each latent variable is measured independently through an observation model. HMMs are widely used in statistics, timeseries analysis and machine learning. Consequently, several specialized algorithms have been

developed to perform both exact and approximate inference on these models. In this chapter, I will initially ignore this rich literature and simply deploy our variational inference techniques to this class of problems. At the end of the chapter, I will introduce the general theory of Bayesian filtering and smoothing which is at the foundation of most HMM algorithms. Finally, I will show how to integrate this theory to the variational approach in order to obtain specialized variational methods.

2.1 Latent Markov timeseries models

The latent dynamics of a HHM is specified by a Markov timeseries model. I will occasionally refer to these models as *stochastic processes* or just *processes*. The density of a Markov timeseries models can be expressed as a chain of transition probabilities $\tau_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$:

$$p(\boldsymbol{x}_{1:T}; \boldsymbol{x}_0) = \prod_{t=1}^{T} \tau_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) \ . \tag{15}$$

where for notational convenience we assumed that the initial condition x_0 is fixed. For example, a univariate Brownian motion is defined by Gaussian transition densities with their mean given by the value of the process at the previous time point:

$$p_{\mathrm{bm}}(\boldsymbol{x}_{1:T}; x_0) = \prod_{t=1}^{T} \mathcal{N}(x_t; x_{t-1}, \sigma^2_{\mathrm{bm}}) \ . \tag{16}$$

In other words, each increment $\Delta x_t = x_{t+1} - x_t$ of a Brownian motion follows an independent normal distribution with mean zero and standard deviation σ_{bm}. Brownian motions are random walks, meaning that the best possible prediction we can make for its future values given the current value is the current value itself. More formally,

$$\mathbb{E}[x_{t+\tau} \mid x_t] = x_t \tag{17}$$

for any $\tau > 0$. This is easy to see since

$$\mathbb{E}[x_{t+\tau} \mid x_t] = \mathbb{E}\left[x_t + \sum_{n=0}^{\tau-1} \Delta x_n\right] = x_t + \sum_{n=0}^{\tau-1} \mathbb{E}[\Delta x_n] = x_t \ . \tag{18}$$

Intuitively, this means that the process wonders around without a specific direction. Brownian motions are a special case of linear autoregressive models, where the probability distribution of the value at the next time point is a Gaussian whose mean is a linear function of the value at the previous time point:

$$p_{\mathrm{ar}}(\boldsymbol{x}_{1:T}; x_0) = \prod_{t=1}^{T} \mathcal{N}(x_{t-1}; \alpha(\beta - x_t); , \sigma^2_{\mathrm{ar}}) \ . \tag{19}$$

When $\alpha > 0$, the process tend to revert its values toward the equilibrium point β. On the other hand, when $\alpha < 0$ the process "explodes" exponentially and its values tend to $\pm\infty$ for T tending to infinity. So far, we assumed the variance of the transition density to be a fixed parameter. However, very often the variance of a process should be a probabilistic timeseries model itself. For example, the standard deviation of the returns of a stock (volatility in financial jargon) changes with time in a partially unpredictable way. This behavior can be captured with a stochastic volatility model:

$$p_{\text{sv}}(\boldsymbol{x}_{1:T}, \boldsymbol{\sigma}_{1:T}; x_0, \sigma_0) = \prod_{t=1}^{T} \mathcal{N}(\sigma_t; \alpha(\beta - \sigma_{t-1}), \sigma_{t-1}^2 \nu^2) \times \mathcal{N}(x_t; x_{t-1}, \sigma_{t-1}^2) \ . \tag{20}$$

here, the stock price is modeled as a Brownian motion and the volatility process is autoregressive with $\alpha > 0$. This ensures that the volatility fluctuates around the equilibrium point β. This reflects the observed behavior of stock prices, where periods of extremely high or low volatility are often followed to a return to the normal range. Note that in this model the variance of the volatility process is scaled by the previous volatility value. This ensures that the process stay positive-valued since the amount of stochastic perturbations becomes infinitesimally small as the process approaches zero.
We can generalize all these models by making the parameters of the Gaussian transition density an arbitrary function of the previous value:

$$p_{\text{nar}}(\boldsymbol{x}_{1:T}; \boldsymbol{x}_0) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t; f_\mu(\boldsymbol{x}_{t-1}), f_\sigma^2(\boldsymbol{x}_{t-1})) \ . \tag{21}$$

where f_μ and f_σ are arbitrary functions. This is an extremely flexible approach that can be used to model complex non-linear dynamics developed in physics, chemistry, engineering and other sciences. It also allows us to use state-of-the-art deep learning networks for f_μ and f_σ to learn the dynamics from observed data.

2.2 Observation models

In many situations, the timeseries $\boldsymbol{x}_{1:T}$ cannot be measured directly. Instead, its values can be inferred from a series of noisy and incomplete observations. Given the latent timeseries, we assume that noisy observations are sampled independently at each time point:

$$p(\boldsymbol{y}_{1:T} \mid \boldsymbol{x}_{1:T}) = \prod_{t=1}^{T} \rho_t(\boldsymbol{y}_t \mid \boldsymbol{x}_t) \ . \tag{22}$$

The densities ρ_t are often called *observation models* or *emission models*. These observation models define the likelihood of a timeseries model. The form of the observation model reflects the measurement device used to probe the latent

process. Very often, the observation models are chosen to have the latent process as mean since the reading of the measurement device is often calibrated to be an unbiased estimate of the latent variable. For example, it is common to use Gaussian observation models such as

$$\rho(\boldsymbol{y}_t \mid \boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{x}_t, \sigma_{lk}^2 I) \ . \tag{23}$$

. We can now write the joint distribution of latent variables and observations:

$$p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}; \boldsymbol{x}_0) = \prod_{t=1}^{T} \tau_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) \rho_t(\boldsymbol{y}_t \mid \boldsymbol{x}_{t-1}) \ . \tag{24}$$

The smoothing posterior is then given by the formula:

$$p(\boldsymbol{x}_{1:T} \mid \boldsymbol{y}_{1:T}; \boldsymbol{x}_0) = \frac{p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}; \boldsymbol{x}_0)}{\int p(\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}; \boldsymbol{x}_0) \mathrm{d}\boldsymbol{x}_{1:T}} \tag{25}$$

2.2.1 Example: Stochastic orbital dynamics

Timeseries models arise naturally in physical applications since most laws of physics describe the temporal evolution of some variables of interest. For example, consider the dynamic of a small asteroid with mass m in orbit around the sun. We denote the location of the asteroid relative to the center of mass of the sun as the three-dimensional vector $\boldsymbol{x}(t)$. The strongest force acting on the asteroid is the gravitational pull of the sun. Furthermore, we account for the effect of random collisions with other asteroids using a stochastic term with variance σ_s^2. This results in the following stochastic equation of motion

$$m\frac{\mathrm{d}^2\boldsymbol{x}(t)}{\mathrm{d}t^2} = - \underbrace{mGM\frac{\boldsymbol{x}(t)}{\|\boldsymbol{x}(t)\|^3}}_{\text{Gravitational force}} + \underbrace{\sigma_s\zeta(t)}_{\text{Stochastic collisions}} \tag{26}$$

where G is the Newton's constant and M is the mass of the sun. The term $\zeta(t)$ is a standard Gaussian variable sampled independently for each time t which models randomness induced by collisions with other asteroids. The second order equation can be re-written as a pair of coupled first order equations:

$$m\frac{\mathrm{d}\boldsymbol{v}(t)}{\mathrm{d}t} = -mGM\frac{\boldsymbol{x}(t)}{\|\boldsymbol{x}(t)\|^3} + \sigma_s\boldsymbol{\zeta}(t) \tag{27}$$

$$\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} = \boldsymbol{v}(t) \ , \tag{28}$$

which in turn can be discretized using the Euler Maruyama rule:

$$\boldsymbol{v}_{t+\mathrm{d}t} = \boldsymbol{v}_t - \mathrm{d}tGM\frac{\boldsymbol{x}_t}{\|\boldsymbol{x}_t\|^3} + \sqrt{\mathrm{d}t}\sigma_s\boldsymbol{\zeta}_t \tag{29}$$

$$\boldsymbol{x}_{t+\mathrm{d}t} = \boldsymbol{x}_t + \boldsymbol{v}_t \ . \tag{30}$$

This results in a Markov timeseries model with the following Gaussian transition density:

$$p(\boldsymbol{v}_{t+\mathrm{d}t} \mid \boldsymbol{v}_t) = \mathcal{N}\left(\boldsymbol{v}_{t+\mathrm{d}t}; \boldsymbol{v}_t - \mathrm{d}tGM\frac{\boldsymbol{x}_t}{\|\boldsymbol{x}_t\|^3}, \mathrm{d}t\sigma_s^2/m^2 I\right) \ . \tag{31}$$

Note that the position variable is computed deterministically from its initial condition and the velocity using the discretized recursive equation:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + \mathrm{d}t\boldsymbol{v}_{t-1} \ . \tag{32}$$

Now assume that the position of the asteroid is measured at each time point using radar sensors from a set of artificial satellites. However, since the asteroid is very small these measurements are highly unreliable. We can model this noisy measurement process using a Gaussian observation model:

$$\rho(\boldsymbol{y}_t \mid \boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{x}_t, \sigma_{\mathrm{lk}}^2 I) \ . \tag{33}$$

3 Mean field variational smoothing in HMMs

In most cases, the filtering posterior of HMMs cannot be obtained in closed form. However, we can approximate the posterior using our usual stochastic variational inference algorithm. As usual, the first step is to define a parameterized variational approximation. The variational approximation is itself a (parameterized) probabilistic timeseries model, which can be expressed as a Markov model:

$$q(\boldsymbol{x}_{1:T}; \theta_{1:T}) = \prod_{t=1}^{T} q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}; \boldsymbol{\theta}_t) \ , \tag{34}$$

where the $q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}; \boldsymbol{\theta}_t)$ are variational transition probabilities. This is a so called *structured variational distribution* since it models statistical dependencies between the variables in the posterior. Using structured distributions is particularly important in timeseries models since the prior usually has a strong auto-correlation structure, which would result in a severe underestimation of the posterior variance if the mean field approach is used.
The loss function of the resulting variational optimization is the following evidence lower bound:

$$\mathbb{E}_{\boldsymbol{x}_{1:T}\sim q}\left[\log \frac{p(\boldsymbol{x}_{1:T})p(\boldsymbol{y}_{1:T} \mid \boldsymbol{x}_{1:T})}{q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}; \boldsymbol{\theta}_t)}\right] \ . \tag{35}$$

The ELBO can be simplified by using the fact that the logarithm of a product is the sum of the logarithms:

$$\mathbb{E}_{\boldsymbol{x}_{1:T}\sim q}\left[\sum_{t=1}^{T} \log \frac{p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\rho_t(\boldsymbol{y}_t \mid \boldsymbol{x}_t)}{q(\boldsymbol{x}_t \mid q(\boldsymbol{x}_{t-1}; \boldsymbol{\theta}_t)}\right] \ . \tag{36}$$

The evidence lower bound can be optimized by stochastic gradient descent. Remember that the samples from q need to be parameterized in order to back-propagate through the variational sampler. The exact form of the re-parameterization formula depends on the choice of the variational distributions $q(\boldsymbol{x}_t\ q(\boldsymbol{x}_{t-1}; \boldsymbol{\theta}_t)$. Not that in this case the re-parameterization function should also depend on $\boldsymbol{x}_t$ since the variational density is conditioned over its value. Let us assume that we can reparameterize the function as

$$\boldsymbol{x}_t = \boldsymbol{r}(\boldsymbol{\epsilon}_t; \boldsymbol{x}_{t-1}, \boldsymbol{\theta}_t), \qquad \boldsymbol{\epsilon}_t \sim q_0(\boldsymbol{\epsilon}_t)\ , \tag{37}$$

where $\boldsymbol{r}(\cdot; \boldsymbol{\theta}_t)$ is a known differentiable function and $q_0(\boldsymbol{\epsilon}_t)$ is a known distribution independent from the variational parameters. We can now write the re-parameterized ELBO as

$$\mathbb{E}_{\boldsymbol{\epsilon}_{1:T} \sim q}\left[\sum_{t=1}^{T} \log \frac{p_t(\boldsymbol{r}(\boldsymbol{\epsilon}_t; \boldsymbol{x}_{t-1}, \boldsymbol{\theta}_t) \mid \boldsymbol{r}(\boldsymbol{\epsilon}_{t-1}; \boldsymbol{x}_{t-2}, \boldsymbol{\theta}_{t-1}))\rho_t(\boldsymbol{y}_t \mid \boldsymbol{r}(\boldsymbol{\epsilon}_t; \boldsymbol{x}_{t-1}, \boldsymbol{\theta}_t))}{q(\boldsymbol{r}(\boldsymbol{\epsilon}_t; \boldsymbol{x}_{t-1}, \boldsymbol{\theta}_t) \mid \boldsymbol{r}(\boldsymbol{\epsilon}_{t-1}; \boldsymbol{x}_{t-2}, \boldsymbol{\theta}_{t-1}); \boldsymbol{\theta}_t)}\right]\ , \tag{38}$$

which again can be estimated using a finite number of particles to obtain a stochastic gradient estimator.

3.1 Structured variational inference

In the example at the beginning of this chapter we saw that the mean field approximation can lead to a severe underestimation of the posterior variance. However, it is usually not straightforward to define an appropriate structured variational distribution. A possible solution is to use the multivariate normal approach we outlined in chapter TODO. However, this will likely fail to capture the potentially non-linear and non-Gaussian dynamics of the latent timeseries model and therefore result in poor approximations. In general, it would be ideal to use a variational timeseries that includes the prior timeseries model as special case for some values of the variational parameters. In fact, at least when the effect of the data is weak, most of the statistical dependencies in the posterior follow from the structure of the prior. Usually, the transition probabilities can be expressed as a fixed parameterized density $\pi(\boldsymbol{x}_t; \boldsymbol{\phi})$ where the value of the parameters $\boldsymbol{\phi}$ depends on the previous value $\boldsymbol{x}_{t-1}$:

$$\tau_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) = \pi(\boldsymbol{x}_t; \boldsymbol{\phi}(\boldsymbol{x}_{t-1}, t))\ , \tag{39}$$

where $\boldsymbol{\phi}(\boldsymbol{x}_{t-1}, t)$ is a known link function that determines the distribution of $\boldsymbol{x}_t$ conditioned on $\boldsymbol{x}_{t-1}$. For example, the transition probabilities in a Brownian motion are given by a Gaussian density with link

$$\boldsymbol{\phi}_{\mathrm{bm}}(x_{t-1}) = (\mu(x_{t-1}), \sigma^2(x_{t-1})) = (x_{t-1}, \sigma^2_{\mathrm{bm}})\ , \tag{40}$$

where the link maps the value x_{t-1} to the two parameters of the Gaussian transition. Note that in this case the variance parameter does not depend on

the input. If we have a transition model of this form, we can construct an appropriate parameterized variational approximation as a trade-of between the parameters induced by the prior link and a learning term:

$$q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}; \boldsymbol{\alpha}_t, \lambda_t) = \pi(\boldsymbol{x}_t \mid \boldsymbol{\lambda} \odot \boldsymbol{\phi}(\boldsymbol{x}_{t-1}) + (1 - \boldsymbol{\lambda}) \odot \boldsymbol{\alpha}_t) \tag{41}$$

where $\boldsymbol{\lambda}_t \in (0, 1)$ is an array learnable trade-off parameters (one for each parameter of the density) and $\boldsymbol{\alpha}_t$ is another learnable array of "perturbation" parameters that accounts for the influence of the data. In this expression, the symbol $\odot$ denotes the element-wise product of the two arrays. This variational posterior reduces to the prior when $\lambda_t = 0$ and to a mean field approximation when $\lambda_t = 1$. Note that this distribution has the same form of the closed-form solution of a Gaussian inference.

3.1.1 Example: Structured inference for Brownian motions

We can now use Eq. 41 to construct a structured variational distribution for our original Brownian motion smoothing problem. As noted above, the transition models of a Brownian motion is Gaussian with link function

$$\boldsymbol{\phi}_{\mathrm{bm}}(x_{t-1}) = (\mu(x_{t-1}), \sigma^2(x_{t-1})) = (x_{t-1}, \sigma^2_{\mathrm{bm}}) \ . \tag{42}$$

We can then define a variational approximation as an update of these Gaussian transitions:

$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t; \lambda_{\mu_t} x_{t-1} + (1 - \lambda_{\mu_t})\alpha_{\mu_t}, \lambda_{\sigma t}\sigma^2_{\mathrm{lk}} + (1 - \lambda_{\sigma t})\alpha_{\sigma t}) \tag{43}$$

where the variational parameters λ_{μ_t} and $\lambda_{\sigma t}$ are the trade-off parameters for mean and variance respectively while α_{μ_t} and $\alpha_{\sigma t}$ are their respective perturbation parameters. As you can see, this variational models reduces to the prior model if we set all the trade-off parameters to be equal to zero. The ELBO of the resulting variational problem can be obtained from Eq. 38, using the re-parameterization functions

$$r(\epsilon_t; x_{t-1}, \lambda_{\mu_t}, \lambda_{\sigma t}, \alpha_{\mu_t}, \alpha_{\sigma t}) = \lambda_{\mu_t} x_{t-1} + (1 - \lambda_{\mu_t})\alpha_{\mu_t} + \sigma_t \epsilon_t \tag{44}$$

with $\sigma_t = \lambda_{\sigma t}\sigma^2_{\mathrm{lk}} + (1 - \lambda_{\sigma t})\alpha_{\sigma t}$. Computing the analytical re-parameterized expression for the ELBO and its gradient is rather cumbersome. However, this is not required in practice as we can compute them using an automatic differentiation framework.

4 Exact filtering and smoothing equations

As we just saw, variational inference offers a clean way to approximately solve Bayesian smoothing problems. However, an important sub-class of those problems admit a very useful and instructive exact solution based on the theory of recursive Bayesian estimation. This is the theory behind the famous Kalman filter (and smoother) and many other useful techniques such as particle filters.

The first equation of recursive Bayesian estimation is the so called predictive equation. This gives the probability of the next state x_{t+1} given the posterior filtering distribution $p(x_t \mid y_{1:T})$, which quantifies the Knowles of the state x_t given all the observations up to t.
Predictive equation:

$$p(x_{t+1} \mid y_{1:t}) = \int \tau_t(x_{t+1} \mid x_t) p(x_t \mid y_t) \mathrm{d}x_t \tag{45}$$

But how do we compute the filtering distribution? Simple. We can use Bayes theorem to update the predictive distribution (which in this case as the role of a prior distribution) given the new data-point y_t. This result in the
Update equation:

$$p(x_{t+1} \mid y_{1:t+1}) = \frac{1}{Z_t} \rho_t(y_{t+1} \mid x_{t+1}) p(x_{t+1} \mid y_{1:t}) \tag{46}$$

where $Z_t = \int \rho_t(y_{t+1} \mid x_{t+1}) p(x_{t+1} \mid y_{1:t}) \mathrm{d}x_{t+1}$.
We can now apply these equations recursively starting from a prior distribution $p_0(x_0)$.
These equations are fully general and can be applied to any dynamical model. Unfortunately, the integrals in the equations cannot usually be solved exactly. However, there is a very interesting special case where that admit an exact solution. In the sake of simplicity, we will see this in a univariate case. However, all the formula can be extended to the multivariate case. Consider a linear Gaussian dynamical model:

$$\tau_t(x_{t+1} \mid x_t) = \mathcal{N}(x_{t+1}; a x_t, \sigma^2) \tag{47}$$

paired with a linear Gaussian emission model:

$$\rho_t(y_t \mid x_t) = \mathcal{N}(y_t; x_t, \nu^2) \ . \tag{48}$$

Let's now solve the recursive Bayesian equations for this linear Gaussian model. We assume the filtering distribution $p(x_{t+1} \mid y_{1:t})$ to be Gaussian parameterized by μ_t and s_t^2. To solve the integral in the predictive equation (Eq. 45), we can simply use reparameterization and notice that:

$$\begin{aligned} x_{t+1} &= a x_t + \sigma \epsilon_t \\ &= a(\mu_t + s_t \xi_t) + \sigma \epsilon_t \\ &= a\mu_t + a s_t \xi_t + \sigma \epsilon_t \ , \end{aligned} \tag{49}$$

where ϵ_t and ϵ_t are two independent standard normal variables. We can therefore conclude that x_{t+1} follows the distribution:

$$p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(x_{t+1}; \tilde{\mu}_{t+1}, \tilde{s}_{t+1}^2) \tag{50}$$

with predictive mean

$$\tilde{\mu}_{t+1} = a\mu_t \ ,$$

and predictive variance

$$\tilde{s}_{t+1}^2 = a^2 s_t^2 + \sigma^2 \ .$$

The Bayesian update rule (Eq. 46) is now a standard Gaussian conjugate inference whth prior $\mathcal{N}(x_{t+1}; \tilde{\mu}_{t+1}, \tilde{s}_{t+1}^2)$ and likelihood $\mathcal{N}(y_{t+1}; x_{t+1}, \nu^2)$. The resulting update rules can be found in any Bayesian statistics book:

$$p(x_{t+1} \mid y_{1:t+1}) = \mathcal{N}(x_{t+1}; \mu_{t+1}, s_{t+1}^2) \tag{51}$$

where

$$\mu_{t+1} = g_{t+1}\tilde{\mu}_{t+1} + (1 - g_{t+1})y_{t+1} \tag{52}$$

$$g_{t+1} = s_{t+1}^2 / \tilde{s}_{t+1}^2 \tag{53}$$

$$s_{t+1}^2 = (\tilde{s}_{t+1}^{-2} + \nu^{-2})^{-2} \ . \tag{54}$$

Note that the mean is updated as a weighted average of the predicted value $\tilde{\mu}_{t+1}$ and the observed data y_t. The mixing coefficient g_t is referred to as the Kalman gain as it regulates the impact of incoming data. So far we developed the equation of a filtering problem, where the posterior distribution is only conditioned on the past and present data. Can we generalize this to a smoothing problem where we condition on the future as well? The answer is yes and it turns out that we can recycle all our previous calculations!
We can start by using Byes theorem to update the filtering posterior (now having the role of a prior) into the smoothing posterior:

$$p(x_t \mid y_{1:T}) \propto p(y_{t+1:T} \mid x_t) p(x_t \mid y_{1:t}) \ , \tag{55}$$

where we omitted the normalization constant. Here, the role of the likelihood is taken by $p(y_{t+1:T} \mid x_t)$: the probability of all future observations given the current state. We can compute this quantity using another recursive equation that run backward in time. The backward recursive equations follow from the Markov property of the model:

$$p(y_{t+1:T} \mid x_t) = \int p(y_{t+1:T} \mid x_{t+1}) \tau_t(x_{t+1} \mid x_t) \mathrm{d}x_{t+1} \tag{56}$$

$$= \int p(y_{t+2:T} \mid x_{t+1}) \rho_t(y_{t+1} \mid x_{t+1}) \tau_t(x_{t+1} \mid x_t) \mathrm{d}x_{t+1} \tag{57}$$

To simplify the notation, we can now introduce a new symbol for the "backward message":

$$\beta_t(x_t) = p(y_{t+1:T} \mid x_t)$$

This leads to the following backward recursive equations for the backward message:

$$\beta_t(x_t) = \int \beta_{t+1}(x_{t+1}) \rho_t(y_{t+1} \mid x_{t+1}) \tau_t(x_{t+1} \mid x_t) \mathrm{d}x_{t+1} \tag{58}$$

which should be initialized with $\beta_{T+1}(x_{T+1}) = 1$. We can now obtain the smoothing posterior by multiplying the filtering posterior with the backward message and re-normalize:

$$p(x_t \mid y_{1:T}) = \frac{1}{\tilde{Z}_t}\beta_t(x_t)p(x_t \mid y_{1:t}) \tag{59}$$

where $\tilde{Z}_t = \int \beta_t(x_t)p(x_t \mid y_{1:t})\mathrm{d}x_t$. This is the famous Forward-backward algorithm, which in turn is a special case of message passing in graphical models. Exact solutions can be again obtained in the linear Gaussian case. This rsults in a modification of the Kalman filter known as the Rauch–Tung–Striebel smoother.