Dynamical Variational Autoencoders : discrete-time and continuous-time models. Links to stochastic calculus and stochastic differential equations

Benjamin Deporte : benjamin.deporte@ens-paris-saclay.fr

August 2025 - DRAFT

Summary

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t\in\mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t \in \mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t \in \mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t \in \mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t\in\mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t \in \mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Abstract

- **Data sequences** : we consider data sequences $(X_t)_{t \in \mathbb{T}} \in \mathbb{R}^D$, where $\mathbb{T}$ is a set of times, either discrete or continuous. ie : time-series, videos, motion captures, patient data...
- **Dynamical Variational Auto Encoders** are a class of VAE models in which some structure is given to the latent variables to express the time dependency of the X_t.
- **Discrete-time DVAEs** are a large set of models, from the well-known Kalman filter up to the Variational RNN. We review the Deep Kalman filter and the VRNN models.
- **Continuous-time DVAEs** use a continuous prior over the latent variables, which allows to deal with irregularly sampled data, or data with missing components. We review the Gaussian Process VAE.
- **Stochastic calculus and stochastic differential equations** provides an elegant mathematical framework for DVAEs. We give a survival kit on stochastic calculus and SDEs.
- **The solution of a linear SDE is a Gaussian process**. We can use known filtering and smoothing Kalman algorithms to compute the GP regression (ie posterior distribution) in GP-VAE with a linear cost.
- **Beyond GP : Latent SDE model** - Not all Gaussian Processes are the solution to a linear SDE. Also, if the solution of a general SDE is a Markov process, it is not necessarily a Gaussian process. This leads to considering Latent SDE model, where the latent prior is a general SDE.

Dynamical Variational Auto Encoders : what is it ?

- **Dynamical Variational Auto Encoders** are a class of VAEs in which some structure is given to the latent variables to encode the time dependency.
- DVAEs can be discrete-time or continuous models, can require regularly-sampled data, or can manage irregularly sampled data.
- For example, a Kalman filter is the simplest DVAE :
  - first order Markov chain for latent variables
  - linear Gaussian observation model.
- As in vanilla VAEs, inference is performed by evidence lower bound maximization.
- Notations

  - the data is a sequence of T points noted $x_{1:T} = \{(x_t)_{t=1,\dots,T}\} \in \mathbb{R}^F$.
  - the sequence of the associated T latent variables is $z_{1:T} = \{(z_t)_{t=1,\dots,T}\} \in \mathbb{R}^L$
  - optionally, there may be a sequence of -usually deterministic- T inputs $u_{1:T} = \{(u_t)_{t=1,\dots,T}\} \in \mathbb{R}^U$

Dynamical Variational Auto Encoders : what is it ?

- **Dynamical Variational Auto Encoders** are a class of VAEs in which some structure is given to the latent variables to encode the time dependency.
- DVAEs can be discrete-time or continuous models, can require regularly-sampled data, or can manage irregularly sampled data.
- For example, a Kalman filter is the simplest DVAE :
  - first order Markov chain for latent variables
  - linear Gaussian observation model.
- As in vanilla VAEs, inference is performed by evidence lower bound maximization.
- Notations
  - the data is a sequence of T points noted $x_{1:T} = \{(x_t)_{t=1,\dots,T}\} \in \mathbb{R}^F$.
  - the sequence of the associated T latent variables is $z_{1:T} = \{(z_t)_{t=1,\dots,T}\} \in \mathbb{R}^L$
  - optionally, there may be a sequence of -usually deterministic- T inputs $u_{1:T} = \{(u_t)_{t=1,\dots,T}\} \in \mathbb{R}^U$

Dynamical Variational Auto Encoders : what is it ?

- **Dynamical Variational Auto Encoders** are a class of VAEs in which some structure is given to the latent variables to encode the time dependency.
- DVAEs can be discrete-time or continuous models, can require regularly-sampled data, or can manage irregularly sampled data.
- For example, a Kalman filter is the simplest DVAE :
  - first order Markov chain for latent variables
  - linear Gaussian observation model.
- As in vanilla VAEs, inference is performed by evidence lower bound maximization.
- Notations
  - the data is a sequence of T points noted $x_{1:T} = \{(x_t)_{t=1,\dots,T}\} \in \mathbb{R}^F$.
  - the sequence of the associated T latent variables is $z_{1:T} = \{(z_t)_{t=1,\dots,T}\} \in \mathbb{R}^L$
  - optionally, there may be a sequence of -usually deterministic- T inputs $u_{1:T} = \{(u_t)_{t=1,\dots,T}\} \in \mathbb{R}^U$

Dynamical Variational Auto Encoders : what is it ?

- **Dynamical Variational Auto Encoders** are a class of VAEs in which some structure is given to the latent variables to encode the time dependency.
- DVAEs can be discrete-time or continuous models, can require regularly-sampled data, or can manage irregularly sampled data.
- For example, a Kalman filter is the simplest DVAE :
  - first order Markov chain for latent variables
  - linear Gaussian observation model.
- As in vanilla VAEs, inference is performed by evidence lower bound maximization.
- Notations
  - the data is a sequence of T points noted $x_{1:T} = \{(x_t)_{t=1,\dots,T}\} \in \mathbb{R}^F$.
  - the sequence of the associated T latent variables is $z_{1:T} = \{(z_t)_{t=1,\dots,T}\} \in \mathbb{R}^L$
  - optionally, there may be a sequence of -usually deterministic- T inputs $u_{1:T} = \{(u_t)_{t=1,\dots,T}\} \in \mathbb{R}^U$

Dynamical Variational Auto Encoders : what is it ?

- **Dynamical Variational Auto Encoders** are a class of VAEs in which some structure is given to the latent variables to encode the time dependency.
- DVAEs can be discrete-time or continuous models, can require regularly-sampled data, or can manage irregularly sampled data.
- For example, a Kalman filter is the simplest DVAE :
  - first order Markov chain for latent variables
  - linear Gaussian observation model.
- As in vanilla VAEs, inference is performed by evidence lower bound maximization.
- Notations

> - the data is a sequence of T points noted $x_{1:T} = \{(x_t)_{t=1,\dots,T}\} \in \mathbb{R}^F$.
> - the sequence of the associated T latent variables is $z_{1:T} = \{(z_t)_{t=1,\dots,T}\} \in \mathbb{R}^L$
> - optionally, there may be a sequence of -usually deterministic- T inputs $u_{1:T} = \{(u_t)_{t=1,\dots,T}\} \in \mathbb{R}^U$

General formulation of DVAE

Generative model

$$\begin{aligned}
p(x_{1:T}, z_{1:T} | u_{1:T}) &= \prod_{t=1}^{T} p(x_t, z_t | x_{1:t-1}, z_{1:t-1}, u_{1:T}) \\
&= \prod_{t=1}^{T} p(x_t | x_{1:t-1}, z_{1:t}, u_{1:T}) p(z_t | x_{1:t-1}, z_{1:t-1}, u_{1:T}) \\
&= \prod_{t=1}^{T} p(x_t | x_{1:t-1}, z_{1:t}, u_{1:t}) p(z_t | x_{1:t-1}, z_{1:t-1}, u_{1:t})
\end{aligned}$$

The only assumption that is made is a a causal dependency of the x_t, z_t on the inputs $u_{1:t}$, thus allowing to change the conditioning $|u_{1:T}$ into $|u_{1:t}$.
In the rest of the presentation, we will consider systems with no input, and drop the conditioning on $u_{1:t}$ to simplify notations. However, the reasoning remains the same with inputs.

Posteriors

- The true posterior $p(z_{1:T}|x_{1:T})$ is usually untractable, but can be developed:

$$p(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} p(z_t|z_{1:t-1}, x_{1:T})$$

- As in vanilla Variational Auto Encoders (VAEs), the inference model is the approximation of the true posterior by an parametric encoder $q_\phi(z_{1:T}|x_{1:T})$, where ϕ is the set of parameters:

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T})$$

- Depending on the chosen graphical models and the corresponding D-separation results, the observation model $p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}, u_{1:t})$ (with θ_x the set of parameters of the observation model) and approximate posterior $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ will simplify.
- It is also considered a good practice to copy the expression of $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ from the expression of the true posterior resulting from the D-separation analysis (see next chapters for examples).

Posteriors

- The true posterior $p(z_{1:T}|x_{1:T})$ is usually untractable, but can be developed:

$$p(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} p(z_t|z_{1:t-1}, x_{1:T})$$

- As in vanilla VAEs, the inference model is the approximation of the true posterior by an parametric encoder $q_\phi(z_{1:T}|x_{1:T})$, where ϕ is the set of parameters:

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T})$$

- Depending on the chosen graphical models and the corresponding D-separation results, the observation model $p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}, u_{1:t})$ (with θ_x the set of parameters of the observation model) and approximate posterior $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ will simplify.
- It is also considered a good practice to copy the expression of $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ from the expression of the true posterior resulting from the D-separation analysis (see next chapters for examples).

Posteriors

- The true posterior $p(z_{1:T}|x_{1:T})$ is usually untractable, but can be developed:

$$p(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} p(z_t|z_{1:t-1}, x_{1:T})$$

- As in vanilla VAEs, the inference model is the approximation of the true posterior by an parametric encoder $q_\phi(z_{1:T}|x_{1:T})$, where ϕ is the set of parameters:

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T})$$

- Depending on the chosen graphical models and the corresponding D-separation results, the observation model $p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}, u_{1:t})$ (with θ_x the set of parameters of the observation model) and approximate posterior $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ will simplify.
- It is also considered a good practice to copy the expression of $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ from the expression of the true posterior resulting from the D-separation analysis (see next chapters for examples).

Posteriors

- The true posterior $p(z_{1:T}|x_{1:T})$ is usually untractable, but can be developed:

$$p(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} p(z_t|z_{1:t-1}, x_{1:T})$$

- As in vanilla VAEs, the inference model is the approximation of the true posterior by an parametric encoder $q_\phi(z_{1:T}|x_{1:T})$, where ϕ is the set of parameters:

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T})$$

- Depending on the chosen graphical models and the corresponding D-separation results, the observation model $p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}, u_{1:t})$ (with θ_x the set of parameters of the observation model) and approximate posterior $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ will simplify.
- It is also considered a good practice to copy the expression of $q_\phi(z_t|z_{1:t-1}, x_{1:T})$ from the expression of the true posterior resulting from the D-separation analysis (see next chapters for examples).

Likelihood

- Observation model and encoder:

$$p_\theta(x_{1:T}, z_{1:T}) = \prod_{t=1}^{T} p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t-1}) \tag{1}$$

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T}) \tag{2}$$

- Log likelihood

$$\log p(x_{1:T}) = \log \frac{p(x_{1:T}, z_{1:T})}{p(z_{1:T}|x_{1:T})} \tag{3}$$

$$= \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} \frac{q_\phi(z_{1:T}|x_{1:T})}{p(z_{1:T}|x_{1:T})} \tag{4}$$

$$= \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} + \mathbb{KL}\left(q_\phi(z_{1:T}|x_{1:T}) || p(z_{1:T}|x_{1:T})\right) \tag{5}$$

$$\geq \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} = \mathcal{L}(\theta, \phi, X) \tag{6}$$

Likelihood

- Observation model and encoder:

$$p_\theta(x_{1:T}, z_{1:T}) = \prod_{t=1}^{T} p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t-1}) \tag{1}$$

$$q_\phi(z_{1:T}|x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t|z_{1:t-1}, x_{1:T}) \tag{2}$$

- Log likelihood

$$\log p(x_{1:T}) = \log \frac{p(x_{1:T}, z_{1:T})}{p(z_{1:T}|x_{1:T})} \tag{3}$$

$$= \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} \frac{q_\phi(z_{1:T}|x_{1:T})}{p(z_{1:T}|x_{1:T})} \tag{4}$$

$$= \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} + \mathbb{KL}\left(q_\phi(z_{1:T}|x_{1:T}) || p(z_{1:T}|x_{1:T})\right) \tag{5}$$

$$\geq \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log \frac{p(x_{1:T}, z_{1:T})}{q_\phi(z_{1:T}|x_{1:T})} = \mathcal{L}(\theta, \phi, X) \tag{6}$$

Variational Lower Bound

- Lower bound:

$$\mathcal{L}(\theta,\phi,X) = \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})} \log\left(\frac{\prod_{t=1}^T p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t-1})}{\prod_{t=1}^T q_\phi(z_t|z_{1:t-1}, x_{1:T})}\right) \tag{7}$$

$$= \mathbb{E}_{q_\phi(z_{1:T}|x_{1:T})}\left(\sum_{t=1}^T \log p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) - \sum_{t=1}^T \log\frac{q_\phi(z_t|z_{1:t-1}, x_{1:T})}{p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t-1})}\right) \tag{8}$$

$$= \sum_{t=1}^T \mathbb{E}_{q_\phi(z_{1:t}|x_{1:T})} \log p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) - \tag{9}$$

$$\sum_{t=1}^T \mathbb{E}_{q_\phi(z_{1:t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t|z_{1:t-1}, x_{1:T}) || p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t-1})\right) \tag{10}$$

- The first term is the usual **reconstruction error**.
- The second term is a regularization term, summing over the time steps the average divergence between the approximate posterior distribution of the latent variable at time t, and its real distribution.
- As in vanilla VAE, the sampling over q_ϕ requires the use of the "re parametrization trick" (see [?]), for $\mathcal{L}(\theta,\phi,X)$ to be differentiable w.r.t. θ,ϕ.

Summary DVAE

General Dynamical VAEs : generative and inference models; variational lower bound

$$p(x_{1:T}, z_{1:T}) = \prod_{t=1}^{T} p_{\theta_x}(x_t | x_{1:t-1}, z_{1:t}) p_{\theta_z}(z_t | z_{1:t-1}, x_{1:t-1}) \tag{11}$$

$$q_\phi(z_{1:T} | x_{1:T}) = \prod_{t=1}^{T} q_\phi(z_t | z_{1:t-1}, x_{1:T}) \tag{12}$$

$$\begin{aligned} \mathcal{L}(\theta, \phi, X) &= \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{1:t}|x_{1:T})} \log p_{\theta_x}(x_t | x_{1:t-1}, z_{1:t}) \\ &- \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{1:t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t | z_{1:t-1}, x_{1:T}) || p_{\theta_z}(z_t | z_{1:t-1}, x_{1:t-1})\right) \end{aligned} \tag{13}$$

Deep Kalman Filter

Deep Kalman Filter Directed Acyclic Graph (DAG):

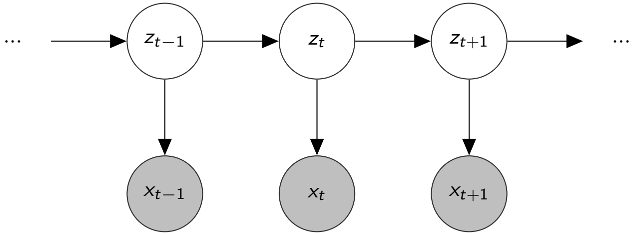


Figure: Probabilistic model of a Deep Kalman Filter

Deep Kalman Filter - generative model

Using **D-separation** on the DAG to simplify the general Dynamical Variational Auto Encoder (DVAE) expressions 11 and 12. Conditioning on z_t and z_{t-1} drives:

$$p_{\theta_x}(x_t|x_{1:t-1}, z_{1:t}) = p_{\theta_x}(x_t|z_t) \tag{14}$$

$$p_{\theta_z}(z_t|z_{1:t-1}, x_{1:t}) = p_{\theta_z}(z_t|z_{t-1}) \tag{15}$$

$$q_{\phi}(z_t|z_{1:t-1}, x_{1:T}) = q_{\phi}(z_t|z_{t-1}, x_{t:T}) \tag{16}$$

Deep Kalman Filter - generative model - 2

We then choose Gaussian distributions for $p_{\theta_x}, p_{\theta_z}$ and q_ϕ, with mean and diagonal covariance, learnt by neural networks.

$$p_{\theta_x}(x_t|z_t) = \mathcal{N}(x_t|\mu_{\theta_x}(z_t), \operatorname{diag}\sigma^2_{\theta_x}(z_t)) \tag{17}$$

$$p_{\theta_z}(z_t|z_{t-1}) = \mathcal{N}(z_t|\mu_{\theta_z}(z_{t-1}), \operatorname{diag}\sigma^2_{\theta_z}(z_{t-1})) \tag{18}$$

$$q_\phi(z_t|z_{t-1}, x_{t:T}) = \mathcal{N}(z_t|\mu_\phi(z_{t-1}, x_{t:T}), \operatorname{diag}\sigma^2_{\theta_z}(z_{t-1}, x_{t:T})) \tag{19}$$

Some other formulations of the approximate posterior (encoder) are possible. For example:

$$q_\phi(z_t|z_{t-1}, x_t)$$
$$q_\phi(z_t|z_{1:t}, x_{1:t})$$
$$q_\phi(z_t|z_{1:T}, x_{1:T})$$

We have chosen 16 for the implementation, as it has the same formulation as the true posterior and respects the corresponding dependencies.

Deep Kalman Filter - ELBO

Using D-Separation, the Evidence Lower Bound (ELBO) 13 simplifies into:

$$\mathcal{L}(\theta, \phi, X) = \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{1:t}|x_{1:T})} \log p_{\theta_x}(x_t|z_t) - \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{1:t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t|z_{t-1}, x_{t:T}) || p_{\theta_z}(z_t|z_{t-1})\right) \quad (20)$$

$$= \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_t|x_{1:T})} \log p_{\theta_x}(x_t|z_t) - \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t|z_{t-1}, x_{t:T}) || p_{\theta_z}(z_t|z_{t-1})\right) \quad (21)$$

Deep Kalman Filter - summary

Deep Kalman Filter

- **generative model**

$$p_{\theta_x}(x_t|z_t) = \mathcal{N}(x_t|\mu_{\theta_x}(z_t), \operatorname{diag}\sigma^2_{\theta_x}(z_t)) \tag{22}$$

$$p_{\theta_z}(z_t|z_{t-1}) = \mathcal{N}(z_t|\mu_{\theta_z}(z_{t-1}), \operatorname{diag}\sigma^2_{\theta_z}(z_{t-1})) \tag{23}$$

- **inference model**

$$q_\phi(z_t|z_{t-1}, x_{t:T}) = \mathcal{N}(z_t|\mu_\phi(z_{t-1}, x_{t:T}), \operatorname{diag}\sigma^2_{\theta_z}(z_{t-1}, x_{t:T})) \tag{24}$$

- **Variational Lower Bound (VLB) for training**

$$\mathcal{L}(\theta, \phi, X) = \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_t|x_{1:T})} \log p_{\theta_x}(x_t|z_t) - \sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_{t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t|z_{t-1}, x_{t:T}) || p_{\theta_z}(z_t|z_{t-1})\right) \tag{25}$$

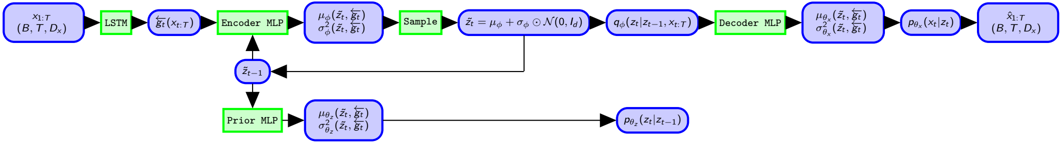
DKF - Torch

- The $\mathbb{KL}\left(q_\phi || p_{\theta_z}\right)$'s have a close form, as the two distributions are Gaussians (see **??**)
- Following [?], we use forward Long Short Term Memory (LSTM) to encode sequences such as $x_{1:t}$, and backward LSTM to encode sequences such as $x_{t:T}$, as inputs into the Multi Layer Perceptron (MLP) parametrizing the distributions.
- For example:

$$\overleftarrow{g_t} = \text{Backward LSTM}(\overleftarrow{g_{t+1}}, x_t) \text{ (encodes } x_{t:T})$$
$$q_\phi(z_t|z_{t-1}, x_{t:T}) = \mathcal{N}(z_t|\mu_\phi(z_{t-1}, \overleftarrow{g_t}), \operatorname{diag}\sigma_\phi^2(z_{t-1}, \overleftarrow{g_t}))$$

DKF - Torch - Schematic blocks

The PyTorch implementation is described below:



$$\mathcal{L}(\theta, \phi, X) = \sum_{t=1}^T \mathbb{E}_{q_\phi(z_t|x_{1:T})} \log p_{\theta_x}(x_t|z_t) - \sum_{t=1}^T \mathbb{E}_{q_\phi(z_{t-1}|x_{1:T})} \mathbb{KL}\left(q_\phi(z_t|z_{t-1}, x_{t:T}) || p_{\theta_z}(z_t|z_{t-1})\right)$$

DVAEs and SDEs

SDEs

Beyond linear SDEs and Gaussian Processes

Beyond

Outro

Conclusions

Annexes

Annexes