Deep Kalman Filters

Paper Review: Team Live and Learn

Introduction

Recently, increased compilation of Electronic Health Records (EHR) has opened an avenue for modeling patient's state over time. Usage of machine learning in this domain is allowing us to ask and answer various medical questions which were not possible even few decades ago. EHR provides temporal information on how patient's condition evolved over time and how different medical actions by the doctors influenced patient's state. In this paper Krishnan et al. introduces a technique for learning causal generative temporal models from noisy high-dimensional data using deep neural nets as building blocks for learning a broad range of Kalman filters.

Background and Related Work

There are three key pieces to the classical Kalman filter models: a sequence of unobserved hidden variables $z_1,..., z_T$, corresponding observed variables $x_1,..., x_T$ and corresponding actions $u_1,..., u_T$. In the medical context, unobserved variables can be thought as patient's true state, observed variables can be thought as diagnosis and lab test results and finally and actions will correspond to medical intervention or medications. In classical Kalman filter models, the latent space is assumed to evolve linearly and the relationship between latent space, observed space and actions are expressed in the form of linear dynamical system.
In this paper, the authors replace linear transformations with non-linear transformations which are parameterized by neural networks. While the non-linearity makes the computation of posterior distribution $p(z_1,...,z_T|x_1,...,x_T,u_1,...,u_T)$ intractable, it allows to address broader set of problems.

The authors use variational autoencoders and introduce a recognition neural network to approximate the intractable posterior distribution. Having learned the approximate posterior distribution of generative temporal model, the authors use it to perform counterfactual inference.

Model

The authors change all the linear functions the Kalman filter modeling to be neural networks. Suppose that we have a sequence of unobserved variables $z_1, z_2, \ldots, z_T \in R^s$. For each unobserved variable z_t, we have a corresponding observation $x_t \in R^d$ and an action $u_t \in R^c$. The following table compares how deep kalman filters replace linear functions with non-linear ones.

Classical Kalman filters	**Deep Kalman filters**
$z_1 \sim N(\mu_0; \Sigma_0)$	$z_1 \sim N(\mu_0; \Sigma_0)$
$z_t = G_t z_{t-1} + B_t u_{t-1} + \varepsilon_t \quad [\varepsilon_t \sim N(0, \Sigma_t)]$	$z_t \sim N(G_\alpha(z_{t-1}, u_{t-1}, \Delta_t), S_\beta(z_{t-1}, u_{t-1}, \Delta_t))$

$x_t = F_t z_t + \eta_t \quad [\eta_t \sim N(0, \Sigma_t)]$	$x_t \sim \Pi(F_\kappa(z_t))$

Note that the functions G_α, S_β, F_κ are parameterized by neural networks, Δ_t is the time difference between time $t-1$ and time t, and Π is a distribution (e.g. a Bernoulli distribution if the data is binary)

Training with Stochastic Backpropagation

To maximize the log-likelihood $\log p_\theta(x)$ for a generative model, they maximize the lower bound $L(x;(\theta,\phi))$ in the following equation using stochastic backpropagation:

$$\log p_\theta(x) \geq E_{q_\theta(z|x)}[\log p_\theta(x|z)] - KL(q_\theta(z|x)||p_0(z)) := L(x;(\theta,\phi))$$

The main learning algorithm is Algorithm 1 on the right, where q_ϕ is an easier-to-sample prior that factorizes according to equation (3) in the paper. The authors explore several variants of this function (e.g, one that is parameterized by an RNN, and another with a bi-directional RNN). To perform counterfactual inference, the authors use *do*-calculus. That is, slightly change the structure of the factor graph and forc action u_t to be a certain action $\tilde{u}_t$, and forward sample from that time and observe the differences this action induces.

Algorithm 1 Learning Deep Kalman Filters

while $notConverged()$ **do**
  $\vec{x} \leftarrow sampleMiniBatch()$
  Perform inference and estimate likelihood:
  1. $\hat{z} \sim q_\phi(\vec{z}|\vec{x},\vec{u})$
  2. $\hat{x} \sim p_\theta(\vec{x}|\hat{z})$
  3. Compute $\nabla_\theta \mathcal{L}$ and $\nabla_\phi \mathcal{L}$ (Differentiating (5))
  4. Update θ, ϕ using ADAM
end while

Experiments and Results

The authors perform two sets of experiments to illustrate the ability of the model to learn causality between actions and observations over time.

Healing MNIST

In the first set of experiments, they show the performance of models with different recognition networks on a synthetic dataset called "Healing MNIST". To create this dataset, they used images from the original MNIST dataset and subjected them to step by step transformations (rotations), as well as bit flipping (with some probability), to obtain a sequence of transformations of the same image. They also superimposed an image over the top left corner of three consecutive transformations. Although this dataset consists of images, the authors created it with the intention of mirroring patterns in medical data; each "transformation" can be thought of as the observations of the patient at that timestep, and the superimposed images can be thought of as independent ailments/conditions that are observed in the history.

The authors created two Healing MNIST datasets; a "small" dataset consisting only of sequences of transformations of a single image each for the digits 1 and 5, and a "large" dataset consisting of transformations of 100 different images each of the digits 1 and 5. There were 40000 and 140000 sequences respectively in each dataset.

With these experiments, the authors aim to address two goals: First, they compare the log-likelihood of unseen test sequences under models trained with different recognition networks, and determine that bidirectional RNNs and RNNs provide the best results. Second, they provide some evidence of the model's ability to perform simple counterfactual inferences; specifically, the model is able to infer a latent state with high log-likelihood, for an unseen patient, and predict the observation.

Generative models of medical data

In the second set of experiments, the authors apply these deep learning based Kalman filters to the problem of making counterfactual inferences from medical data. Specifically, they're interested in measuring the effects of anti-diabetic drugs on a patient's A1c and glucose levels, and to do so, they attempt to use the model to infer the effects of *not* taking anti-diabetic drugs. They find that, as expected, patients who do not take anti-diabetic medication would be (as predicted by the model) more prone to higher glucose and A1c levels on average.

Critical Discussion

The paper has shows convincing results, suggesting that deep-learning nonlinear Kalman filters can be useful for counterfactual analysis. One natural extension of this work is that of **patient representations**: Although the model is designed to perform counterfactual inference, to approximate the latent state of unseen data with the parametric posterior, channeling sample input for actions that did not occur and observing their consequent effect, the model can also be used to naturally encode patient entities into latent vector space. Therefore, it should be then possible to compare different patients in this vectorspace, perform transitions or optimization based on these fundamental properties represented as different dimensions in the vector space.
One major shortcoming is that while the authors were able to show that such Kalman filters can learn the general directionality of the effects of certain actions, it is unclear whether the actual **values or observations** (or change in observed values) predicted by these models are reliable or accurate to some degree. Specifically, there is no investigation into the reliability of the model's inferences when it predicts, for example, that a given patient's glucose levels would be X units higher had they not taken the prescribed anti-diabetes medication. This calls into question the practical applicability of the model, especially in settings where the actual degree of change could be incredibly important. For example, we might want to infer how much a given medication will affect a patient's glucose levels when the effects of the medication are unknown (unlike in the dataset in this paper, where the provided medication is a known anti-diabetes drug). Although it is impossible to gather counterfactual data (for example, showing glucose levels for a given patient under the condition that they did take an anti-diabetes medicine and under the condition that they didn't), future works should make a better attempt to investigate this issue (by comparing effects on patients with very similar latent representations, for example).
Another potential extension to the work in this paper is that of **prediction** (as opposed to counterfactual inference). While the work in this paper shows that Kalman filters can be parameterized nonlinearly by neural networks to some success, the use of a bidirectional RNN as the recognition network (when training generative models on the medical data) greatly restricts potential applications of the model. Most importantly, such a model cannot be used to make predictions of the future, given past history, because the underlying bidirectional RNN requires the full set of observations (past and future) in order to make any kinds of inferences. Thus, while this paper shows that the model learns strong, valid statistical properties (because the counterfactual inferences made are valid), a more realistic and practical future exploration could involve exploring the performance of unidirectional sequential models that more easily lend themselves to making predictions about the future.

Finally, the encoded vector space information could be used to identify existing disease trajectories or patient characteristics, but it is unclear how this specific auto encoded representation will be more or similarly useful for this purpose than a latent space computed while maximizing this kind of disease or patient identification.