Stochastic Video Prediction with Conditional Density Estimation

Rui Shu[1]
ruishu@stanford.edu
James Brofos[2]
jbrofos@mitre.org
Frank Zhang[2]
fzhang@mitre.org
Hung Hai Bui[3]
hubui@adobe.com
Mohammad Ghavamzadeh[3]
ghavamza@adobe.com
Mykel Kochenderfer[4]
mykel@stanford.edu

[1] Biomedical Informatics, Stanford University
[2] The MITRE Corporation
[3] Adobe Research
[4] Department of Aerospace and Aeronautics, Stanford University

Abstract

Frame-to-frame stochasticity is a major challenge in video prediction. The use of standard feedforward and recurrent networks for video prediction leads to averaging of future states, which can in part be attributed to the networks' limited ability to model stochasticity. We propose the use of conditional variational autoencoders (CVAE) for video prediction. To model multi-modal densities in frame-to-frame transitions, we extend the CVAE framework by modeling the latent variable with a mixture of Gaussians in the generative network. We tested our proposed Gaussian mixture CVAE (GM-CVAE) on a simple video-prediction task involving a stochastically moving object. Our architecture demonstrates improved performance, achieving noticeably lower rates of blurring/averaging compared to a feedforward network and a Gaussian CVAE. We also describe how the CVAE framework can be applied to improve existing deterministic video prediction models.[1]

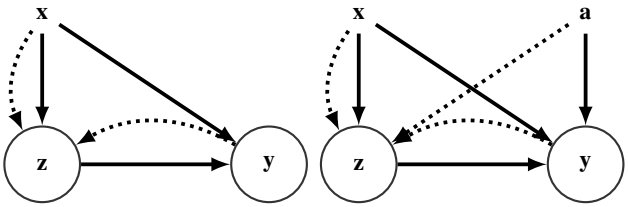
1 Introduction

Modeling videos is a challenging problem involving high-dimensional data with complex dynamics. As a result, most video prediction studies focus on video data where frame-to-frame transitions are strongly deterministic [6]. Recent developments in deep learning approaches for video prediction have sought to overcome this limitation. To address the issue of blurry predictions, Mathieu et al. [5] proposed replacing the mean squared error loss function with a custom loss function learned by a discriminator with adversarial training. A separate line of study by Oh et al. [6] tackled video prediction tasks where the next frame depends not only on the history of previous frames, but also on the action taken by an agent. However, these solutions ultimately rely on a deterministic network for next-frame prediction. As such, their proposed frameworks do not model the inherently stochastic frame-to-frame dynamics present in many video prediction tasks.

The limitations of a deterministic network is apparent when Oh et al. [6] applied their video prediction framework to the game of Ms. Pacman. While the network was able to predict the movement of the agent, it could not successfully handle the stochastically-moving ghosts, leading to the "disappearing ghosts phenomenon" (see https://youtu.be/cy96rtUdBuE). In cases where the video prediction task is inherently stochastic, it is natural to consider a probabilistic model of the next-frame distribution when conditioned on previous frames. In other words, we wish to solve the task of high-dimensional conditional density estimation.

2 Approach

We propose to perform conditional density estimation using a conditional variational autoencoder framework [8]. We further extend the conditional variational autoencoder model by introducing a Gaussian mixture distribution to tackle the issue of multi-modality in video prediction. We provide preliminary results where the use of a stochastic network capable of modeling multi-modality proves vital. Finally, we show that our proposed probabilistic model can be combined with the action-conditional model by Oh et al. [6], thus demonstrating a simple probabilistic extension to existing deterministic models. We intend to use this probabilistic model to address the "missing ghosts phenomenon," which remains an open problem.



(a) Frame prediction (b) Action-conditional frame prediction

Figure 1: The directed graphical model associated with the conditional variational autoencoder architecture. In 1(a), solid lines denote the generative model $p_\theta(\mathbf{z} \mid \mathbf{x})p_\theta(\mathbf{y} \mid \mathbf{z},\mathbf{x})$. Dashed lines denote the variational approximation $q_\phi(\mathbf{z} \mid \mathbf{x},\mathbf{y})$ to the intractable posterior $p_\theta(\mathbf{z} \mid \mathbf{x},\mathbf{y})$. The variational ϕ and generative θ parameters are learned jointly. In 1(b), an additional node is introduced to reflect the conditioning on the action.

2.1 Frame prediction

Suppose $\mathbf{x}$ denotes the history of previous frames and $\mathbf{y}$ denotes the next frame. If we introduce a latent variable $\mathbf{z}$ that controls the frame transition dynamics, then we wish to learn the weight θ that parameterizes the distribution $p_\theta(\mathbf{z},\mathbf{y} \mid \mathbf{x}) = p_\theta(\mathbf{z} \mid \mathbf{x})p_\theta(\mathbf{y} \mid \mathbf{x},\mathbf{z})$. This model, shown in Figure 1(a), reflects our belief that, in a stochastic video prediction task with Markovian dynamics, the next frame is a function of the previous frame and injected noise. We further condition $\mathbf{z}$ on $\mathbf{x}$ so that our model takes into consideration the possibility that the previous frame can influence the level of noise injection. In this setup, the stochastic variable $\mathbf{z}$ *explains away* features of $\mathbf{y}$ that cannot be accounted for by $\mathbf{x}$, and the level to which $\mathbf{z}$ explains away the features of $\mathbf{y}$ is further modulated by $\mathbf{x}$.

To learn the weights ϕ and θ that parameterize the inference and generative networks respectively, we minimize (with respect to ϕ,θ) the following loss function based on the variational lower bound

$$\mathcal{L} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\ln q_\phi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) - \ln p_\theta(\mathbf{z} \mid \mathbf{x})p_\theta(\mathbf{y} \mid \mathbf{x},\mathbf{z})\right]. \tag{1}$$

Following Kingma et al. [3], we optimize the objective function using stochastic gradient variational Bayes. Once we have trained the network, $\mathbf{y}$ can be sampled from $p_\theta(\mathbf{y} \mid \mathbf{x})$ using a two-step process,

$$\mathbf{z} \sim p_\theta(\mathbf{z} \mid \mathbf{x}),\ \mathbf{y} \sim p_\theta(\mathbf{y} \mid \mathbf{x},\mathbf{z}), \tag{2}$$

and can be used for generating subsequent frames.

2.2 Action-conditional frame prediction

In action-conditional video prediction, the action of the agent influences the next frame. We introduce this into the model (Figure 1(b)) by including an action variable $\mathbf{a}$, such that $p_\theta(\mathbf{y},\mathbf{z} \mid \mathbf{x},\mathbf{a}) = p_\theta(\mathbf{z} \mid \mathbf{x})p_\theta(\mathbf{y} \mid \mathbf{x},\mathbf{a},\mathbf{z})$.

[1] This work was completed and submitted during the first author's internship at Adobe Research.

As part of our design choice, we choose not to incorporate the current action information $\mathbf{a}$ when sampling $\mathbf{z}$ in the generative network. This reflects a dynamical system where randomness in $\mathbf{y}$ is not directly caused by $\mathbf{a}$, but is instead a by-product of a deterministic action interacting with the inherent stochasticity of the system. We minimize the loss function

$$\mathcal{L} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{a},\mathbf{y})}\left[\ln q_\phi(\mathbf{z} \mid \mathbf{x},\mathbf{a},\mathbf{y}) - \ln p_\theta(\mathbf{z} \mid \mathbf{x}) p_\theta(\mathbf{y} \mid \mathbf{x},\mathbf{a},\mathbf{z})\right], \quad (3)$$

and learn a model that performs action-conditional next-frame generation.

2.3 Multi-modal densities

Standard VAEs are constrained by the use of factorized Gaussian distributions in both the generative and recognition networks. This is problematic if the density estimation task involves strong multi-modality—an issue ubiquitous in video prediction. For example, in video game prediction, starting from the same frame may result in divergent next-frame trajectories. Rather than making posterior inference approximation more flexible [4, 7], we propose to incorporate multi-modality directly into our generative network by making $p_\theta(\mathbf{z} \mid \mathbf{x})$ a Gaussian mixture density.

However, using Gaussian mixtures makes the regularization component of the objective functions (1) and (3) intractable. In Equation (1) for example, the KL term, $\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\ln q_\phi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) - \ln p_\theta(\mathbf{z} \mid \mathbf{x})\right]$, no longer has a closed-form solution if $p_\theta(\mathbf{z} \mid \mathbf{x})$ is a Gaussian mixture. While it is possible to approximate the Kullback-Leibler divergence using Monte Carlo sampling, this increases the variance of the gradient estimator. Instead, we follow Hershey et al. [1] and use a closed-form approximation of the KL term. Supposing that f is a Gaussian distribution and g is a mixture of k Gaussians, $g = \sum_{i=1}^{k} \pi_i g_i$, where $g_{1:k}$ are the individual components with weights $\pi_{1:k}$, then $\mathrm{KL}(f\|g)$ can be approximated with,

$$\mathrm{KL}(f\|g) \approx \log \frac{1}{\sum_i \pi_i \exp(-\mathrm{KL}(f\|g_i))}. \quad (4)$$

This approximation reduces the variance of the estimated gradient. We call this model the "Gaussian mixture conditional variational autoencoder" (GM-CVAE).

3 Experiments

In this section, we present a toy stochastic video prediction task that demonstrates the importance of using a stochastic network that handles multi-modality. We also discuss how the CVAE framework can be incorporated with work by Oh et al. [6] to tackle the "missing ghosts phenomenon."

3.1 Stochastic sprite

The stochastic sprite dataset is a toy video prediction problem. As the name implies, the environment consists of a sprite that moves stochastically (Figure 2). We use a two-layer multilayer perceptron (MLP) that outputs $\mathbf{y}$ given $\mathbf{x}$. We then incorporate the CVAE framework by including a 1-dimensional latent variable $\mathbf{z}$. The GM-CVAE framework uses four component Gaussians in the mixture.

By observing the sampled trajectories from the trained networks (Figure 2), it is easy to see that the deterministic MLP is incapable of handling the stochastically-moving sprite. The trained MLP is susceptible to averaging over the possible future states, causing the appearance multiple ghosts (albeit with lower signal strength). Incorporating the CVAE framework noticeably reduces the undesired averaging behavior. Furthermore, the stochastic movement of the sprite is strongly multi-modal (for instance, the sprite can choose to make a sharp left or right turn); this multi-modality is better accounted for by the GM-CVAE, which explains why the GM-CVAE model appears to achieve lower rates of future-state averaging than CVAE.

3.2 Ms. Pacman

The stochastic sprite dataset shows the limitations of a deterministic network and sheds light on the "disappearing ghost phenomenon" in Oh et al. [6]. Because the ghosts in Ms. Pacman move stochastically, the deterministic network is not able to tell *a priori* how the ghosts will move. When

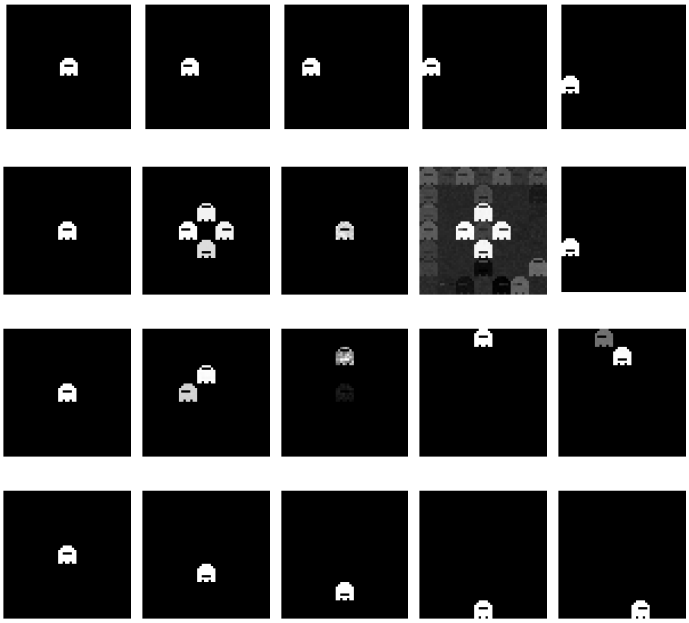


Figure 2: First row: A sample trajectory from our dataset consisting of a sprite that pivots stochastically at select locations. Remaining rows: sampled trajectories from a trained MLP, CVAE, and GM-CVAE respectively. See https://youtu.be/5fe30qSxW5s.

paired with an L_2 loss, the deterministic network chooses to average over all possible ghost movements. When performing multi-step prediction, the cascaded averaging causes the ghosts to decay into the background.

It is thus imperative to have a model that successfully models the stochastic frame-to-frame dynamics of Ms. Pacman. Accurately modeling the high-dimensional frame dynamics of a game such as Ms. Pacman remains an open problem. While previous works have explored the use of VAEs for video prediction and control-based video prediction [2, 9], it is not clear if these models can be easily scaled to model Ms. Pacman. To this end, the architecture by Oh et al. [6] is impressive in that it is able to perform accurate frame prediction in all cases except when stochasticity is vital. We thus propose to follow Oh et al. closely and demonstrate how the CVAE framework can be layered on top of their existing architecture, as shown in Figure 3.

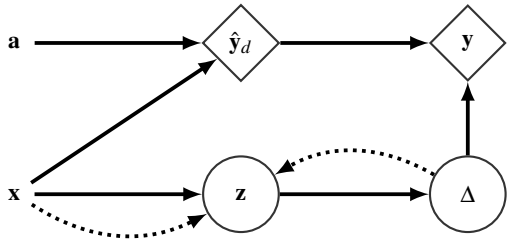


Figure 3: The probabilistic action-conditional model in Figure 1(b) can be implemented in a way that easily incorporates the trained network proposed by Oh et al. [6]. Note that we define $\mathbf{y} = \hat{\mathbf{y}}_d + \Delta$.

We note that the original architecture already makes a deterministic approximation $\hat{\mathbf{y}}_d$ of the next-frame $\mathbf{y}$. A simple method of incorporating the probabilistic framework (Figure 1(b)) is to auto-encode $\Delta = \mathbf{y} - \hat{\mathbf{y}}_d$ using the latent variable $\mathbf{z}$. Using the trained network from Oh et al. [6], the objective function (3) reduces to

$$\mathcal{L} = \mathbb{E}_{q_\phi(\mathbf{z}|\Delta,\mathbf{x})}\left[\ln q_\phi(\mathbf{z} \mid \Delta,\mathbf{x}) - \ln p_\theta(\Delta,\mathbf{z} \mid \mathbf{x})\right]. \quad (5)$$

Given the accuracy with which $\hat{\mathbf{y}}_d$ already approximates $\mathbf{y}$, Δ is only necessary for introducing minor changes that corrects $\hat{\mathbf{y}}_d$ when stochasticity is involved. In essence, the stochastic channel $\mathbf{x} \to \mathbf{z} \to \Delta$ learns to model the residual stochasticity that cannot be accounted for by $\hat{\mathbf{y}}_d$.

[1] J. R. Hershey and P. A. Olsen. Approximating the Kullback-Leibler divergence between gaussian mixture models. *International Conference on Acoustics Speech and Signal Processing*, 2007.

[2] M. J. Johnson, D. Duvenaud, A. B. Wiltschko, S. R. Datta, and R. P. Adams. Composing graphical models with neural networks for structured representations and fast inference. *ArXiv e-prints*, March 2016.

[3] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2013.

[4] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther. Auxiliary deep generative models. *International Conference of Machine Learning*, 2016.

[5] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. *International Conference on Learning Representations*, 2016.

[6] J. Oh, X. Guo, H. Lee, R. Lewis, and S. Singh. Action-conditional video prediction using deep networks in Atari games. *Neural Information Processing Systems*, 2015.

[7] D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. *International Conference of Machine Learning*, 2015.

[8] K. Sohn, X. Yan, and H. Lee. Learning structured output representation using deep conditional generative models. *Neural Information Processing Systems*, 2015.

[9] M. Watter, J. Springenberg, J. Boedecker, and M. Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. *Neural Information Processing Systems*, 2015.