VI. Energy-based Model

Young-geun Kim

Department of Statistics and Probability

STT 997 (SS 2025)

Outline

1. Restricted Boltzmann Machine

2. Temporal Variants of Restricted Boltzmann Machine

Recap of Previous Section

- In the previous section, we reviewed linear methods, including ICA, and auto-regressive models.
- ICA enjoys fast data generation and dimension reduction capabilities but may lack the ability to express complex data generation processes.
- Auto-regressive models may be better at modelling complex structures, but generation may require substantial time. Dimension reduction is not provided, or there may not be a single latent vector that can express the information of all input features.
- In this lecture, we will review another generative model, energy-based models. Under some conditions, their model class can approximate arbitrary distributions and offers faster generation than auto-regressive models.

Energy-based Model

- Energy-based models are generative models that utilize energy functions, or *Boltzmann distributions* with unit temperature and unit Boltzmann constant:

$$p_\theta(\vec{x}) := \exp(-E_\theta(\vec{x}))/C(\theta) \tag{1}$$

where $E_\theta(\vec{x})$ indicates the energy, usually in an interpretable form, and $C(\theta) := \int \exp(-E_\theta(\vec{x}))d\vec{x}$ is the normalization constant.

- For example, the Gaussian distribution with parameters $(\vec{\mu}, \Sigma)$,

$$p_\theta(\vec{x}) = (2\pi)^{-p/2}|\Sigma|^{-1/2}\exp\Big(-(\vec{x}-\vec{\mu})^T\Sigma^{-1}(\vec{x}-\vec{\mu})/2\Big), \tag{2}$$

has an energy function $E_\theta(\vec{x}) = (\vec{x}-\vec{\mu})^T\Sigma^{-1}(\vec{x}-\vec{\mu})/2$, representing the squared Mahalanobis distance between $\vec{x}$ and the Gaussian distribution. Its normalization constant $C(\theta)$ is $(2\pi)^{p/2}|\Sigma|^{1/2}$.[1]

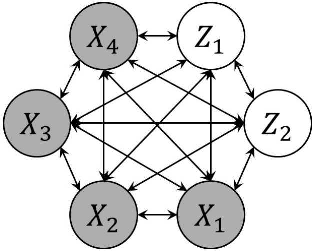
[1] In general, an exponential family distribution, represented by $p_\theta(\vec{x}) = h(\vec{x})\exp(\eta(\theta)^T T(\vec{x}) - A(\theta))$, can be interpreted as a special case of Equation (1). Here, $E_\theta(\vec{x}) = -\eta(\theta)^T T(\vec{x}) - \log h(\vec{x})$ and $C(\theta) = \exp(A(\theta))$.

Boltzmann Machine

- Boltzmann machines (Ackley et al., 1985) and their variants are notable examples of energy-based models. They commonly introduce latent variables, and define the energy density function over the joint distribution of latent variables and observations.
- Energies in Boltzmann models can be expressed as $E_\theta(\vec{z},\vec{x}) := -\Big((\vec{z}^T,\vec{x}^T)W(\vec{z}^T,\vec{x}^T)^T/2 + (\vec{\alpha}^T,\vec{\beta}^T)(\vec{z}^T,\vec{x}^T)^T\Big)$ where $\theta := (W,\vec{\alpha},\vec{\beta})^T$, W is symmetric, and the diagonal elements of W are all zero, indicating that there is no self-loops. Then, corresponding densities can be expressed as:

$$p_\theta(\vec{z},\vec{x}) \propto \exp\left(\frac{1}{2}\begin{pmatrix}\vec{z}\\ \vec{x}\end{pmatrix}^T \begin{pmatrix} W_{ZZ} & W_{ZX} \\ W_{ZX}^T & W_{XX}\end{pmatrix}\begin{pmatrix}\vec{z}\\ \vec{x}\end{pmatrix} + (\vec{\alpha}^T,\vec{\beta}^T)\begin{pmatrix}\vec{z}\\ \vec{x}\end{pmatrix}\right). \tag{3}$$

Boltzmann Machine



- For example, when we have $\vec{x} \in \mathbb{R}^{p=4}$ and $\vec{z} \in \mathbb{R}^{d=2}$, the $p_\theta(\vec{z}, \vec{x})$ is proportional to:

$$\exp\Big(W_{ZZ,1,2} z_1 z_2 + (W_{ZX,1,1} z_1 x_1 + \cdots + W_{ZX,2,4} z_2 x_4) + (W_{XX,1,2} x_1 x_2 + \cdots W_{XX,3,4} x_3 x_4) + (\alpha_1 z_1 + \alpha_2 z_2) + (\beta_1 x_1 + \cdots \beta_4 x_4) \Big). \tag{4}$$

Q: Don't we need bias terms in $p_\theta(\vec{z}, \vec{x})$?

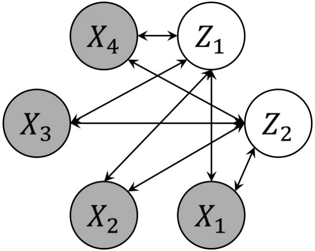
Boltzmann Machine

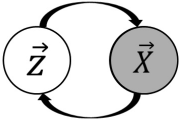
- Boltzmann machines assume binary latent variables and observed features.
- The tractable energies provide interpretation for logit functions. For a given $\vec{x}_{-j} := (x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_p)^T$, let $\vec{x}_{j,\text{on}} := (x_1, \ldots, x_j = 1, \ldots, x_p)^T$ and $\vec{x}_{j,\text{off}} := (x_1, \ldots, x_j = 0, \ldots, x_p)^T$. We have,

$$\begin{aligned} \log\left(\frac{p_\theta(\vec{z}, \vec{x}_{j,\text{on}})}{p_\theta(\vec{z}, \vec{x}_{j,\text{off}})}\right) &= -E_\theta(\vec{z}, \vec{x}_{j,\text{on}}) + E_\theta(\vec{z}, \vec{x}_{j,\text{off}}) \\ &= (W_{ZX}^T \vec{z} + W_{XX}\vec{x} + \vec{\beta})_j \end{aligned} \tag{5}$$

- However, due to the connections within $\vec{z}$ and $\vec{x}$, represented respectively by W_{ZZ} and W_{XX}, it may be difficult to evaluate $p_\theta(\vec{z}|\vec{x})$, $p_\theta(\vec{x}|\vec{z})$, and corrsponding logits.

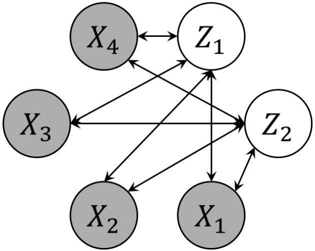
Restricted Boltzmann Machine: Density Model

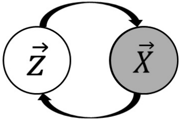




- Restricted Boltzmann Machines (RBMs) simplify Boltzmann machines by restricting connections between latent components and between observed features, defining a bipartite graph.
- They were originally proposed by Smolensky et al. (1986) under the name 'Harmonium' and became popular after Hinton (2002) proposed efficient algorithms to train the model.

Restricted Boltzmann Machine: Density Model





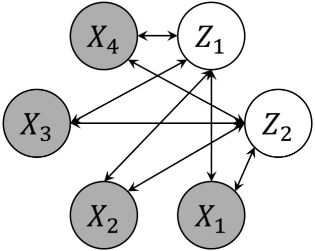
- Energies in RBMs can be expressed as:

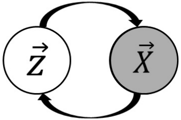
$$E_\theta(\vec{z},\vec{x}) := -\left(\vec{z}^T W\vec{x} + \vec{\alpha}^T\vec{z} + \vec{\beta}^T\vec{x}\right) \tag{6}$$

which corresponds to Equation (3), equipped with $W_{ZZ} = O$, $W_{XX} = 0$, and $W_{ZX} = W$.[2]

[2] RBMs assume binary observations and latent variables, but they can be extended to exponential family distributions (Welling et al., 2004). For simplicity, we handle binary cases in this lecture.

Restricted Boltzmann Machine: Density Model





- For example, when we have $\vec{x} \in \mathbb{R}^{p=4}$ and $\vec{z} \in \mathbb{R}^{d=2}$, the $p_\theta(\vec{z}, \vec{x})$ is proportional to:

$$\exp\Big((W_{1,1}z_1x_1 + \cdots + W_{2,4}z_2x_4) + (\alpha_1 z_1 + \alpha_2 z_2) + (\beta_1 x_1 + \cdots \beta_4 x_4)\Big). \tag{7}$$

Restricted Boltzmann Machine: Inference

- This joint density $p_\theta(\vec{z}, \vec{x})$, specified by the energy in Equation (6), exhibits conditional independency; That is, $p_\theta(\vec{x}|\vec{z}) = \prod_{j=1}^{p} p_\theta(x_j|\vec{z})$ and $p_\theta(\vec{z}|\vec{x}) = \prod_{j=1}^{d} p_\theta(z_j|\vec{x})$. Furthermore, RBMs benefit from interpretable inference:

$$p_\theta(z_j = 1|\vec{x}) = \text{Sigmoid}\Big((W\vec{x} + \vec{\alpha})_j\Big). \tag{8}$$

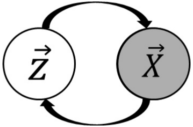
Hint: $p_\theta(\vec{z}|\vec{x}) = p_\theta(\vec{z}, \vec{x})/p_\theta(\vec{x}) \propto \exp((W\vec{x} + \vec{\alpha})^T\vec{z})$.
Similarly,

$$p_\theta(x_j = 1|\vec{z}) = \text{Sigmoid}\Big((W^T\vec{z} + \vec{\beta})_j\Big). \tag{9}$$

- That is, p_θ has a closed-form expression on the probability of $\vec{x}$ being allocated to each of 2^d clusters depending on whether each latent factor is activated or not.
- The restriction on the connection yields linear (conditional) logit functions:

$$\log \frac{p_\theta(z_1, \ldots, z_j = 1, \ldots, z_d|\vec{x})}{p_\theta(z_1, \ldots, z_j = 0, \ldots, z_d|\vec{x})} = (W\vec{x} + \vec{\alpha})_j. \tag{10}$$

Restricted Boltzmann Machine: Generation



- RBMs synthesize data using the following M-step Gibbs-sampling process:
    1. Sample $\tilde{\vec{X}}_1$ from a pre-specified distribution $p(\tilde{\vec{x}})$,[3] and use $\tilde{\vec{X}}_1$ to generate $\tilde{\vec{Z}}_1$ from $p_\theta(\vec{z}|\vec{x} = \tilde{\vec{X}}_1)$ (Equation (8)).
    2. For $m = 1, \ldots, M - 1$, use $\tilde{\vec{Z}}_m$ to generate $\tilde{\vec{X}}_{m+1}$ from $p_\theta(\vec{x}|\vec{z} = \tilde{\vec{Z}}_m)$ (Equation (8)), and use $\tilde{\vec{X}}_{m+1}$ to generate $\tilde{\vec{Z}}_{m+1}$ from $p_\theta(\vec{z}|\vec{x} = \tilde{\vec{X}}_{m+1})$ (Equation (9)).
    3. The $(\tilde{\vec{Z}}_M, \tilde{\vec{X}}_M)$ is the M-step Gibbs sample. Under some conditions for the stationary, its density converges to $p_\theta(\vec{z}, \vec{x})$ as $M \to \infty$.

[3] A typical choice is $p(\vec{x})$ by using real observations $\vec{X}$.

Restricted Boltzmann Machine: Training

- (Restricted) Boltzmann machines aim to find maximum likelihood estimators. However, the log-density,

$$\log p_\theta(\vec{x}) = -\log C(\theta) + \log \sum_{\vec{z}} \exp(-E_\theta(\vec{z}, \vec{x})) \tag{11}$$

  includes the normalization constant $C(\theta)$, posing computational challenges.

- Contrastive divergence (CD) (Hinton, 2002) is an algorithm tailored to train energy-based models. The CD algorithm utilizes the following expression of $\partial \log p_\theta(\vec{x})/\partial\theta$:

$$\partial \log p_\theta(\vec{x})/\partial\theta = -\sum_{\vec{z}} \Big(\partial E_\theta(\vec{z}, \vec{x})/\partial\theta\Big) p_\theta(\vec{z}|\vec{x}) + \sum_{\vec{z},\vec{x}} \Big(\partial E_\theta(\vec{z}, \vec{x})/\partial\theta\Big) p_\theta(\vec{z}, \vec{x}) \tag{12}$$

  Hint: Use $C(\theta) := \sum_{\vec{z},\vec{x}} \exp(-E_\theta(\vec{z}, \vec{x}))$ to derive
  $\partial \log C(\theta)/\partial\theta = -\sum_{\vec{z},\vec{x}} (\partial E_\theta(\vec{z}, \vec{x})/\partial\theta) p_\theta(\vec{z}, \vec{x})$.

Restricted Boltzmann Machine: Training

- In the first term, the $\sum_{\vec{z}} \Big(\partial E_\theta(\vec{z},\vec{x})/\partial\theta\Big) p_\theta(\vec{z}|\vec{x})$ represents the population mean of $\partial E_\theta(\vec{z},\vec{x})/\partial\theta$ over $p_\theta(\vec{z}|\vec{x})$. This is called *positive gradient* (or the gradient computed with *positive data*).
- In the second term, the $\sum_{\vec{z},\vec{x}} \Big(\partial E_\theta(\vec{z},\vec{x})/\partial\theta\Big) p_\theta(\vec{z},\vec{x})$ represents the population mean of $\partial E_\theta(\vec{z},\vec{x})/\partial\theta$ over $p_\theta(\vec{z},\vec{x})$. This is called *negative gradient* (or the gradient computed with *negative data*).
- Therefore, the score function, $\partial \log p_\theta(\vec{x})/\partial\theta$ is the difference between two gradients from positive and negative data, which is the origin of the name *contrastive divergence*.
- Note that $E_\theta(\vec{z},\vec{x})$ is tractable, e.g., in RBMs, $E_\theta(\vec{z},\vec{x}) = -\Big(\vec{z}^T W\vec{x} + \vec{\alpha}^T\vec{z} + \vec{\beta}^T\vec{x}\Big)$, which alleviates the computational issue.

Restricted Boltzmann Machine: Training

- Motivated by this, contrastive divergence algorithm applies Gibbs sampling to approximate the first and second terms.
- For a given training dataset $(x_i)_{i=1}^{n}$[4], the process of CD algorithm can be summarized as follows:
  1. (Sampling positive data) For a given observation $\vec{x}_i$, sample $\tilde{\vec{z}}_i$ from $p_\theta(\vec{z}|\vec{x} = \vec{x}_i)$. The pair $(\tilde{\vec{z}}_i, \vec{x}_i)$ is the i-th positive sample.
  2. (Sampling negative data) Sample the i-th negative datum, $(\tilde{\vec{z}}_{i,M}, \tilde{\vec{x}}_{i,M})$ using the M-step Gibbs sampling.
  3. Compute positive and negative gradients using sampled data from previous steps.
  4. Update θ by ascending the log-likelihood.

[4] We can use $\{x_i\}_{i=1}^{n}$ for generalized RBMs for continuous data.

Restricted Boltzmann Machine: Generation Result

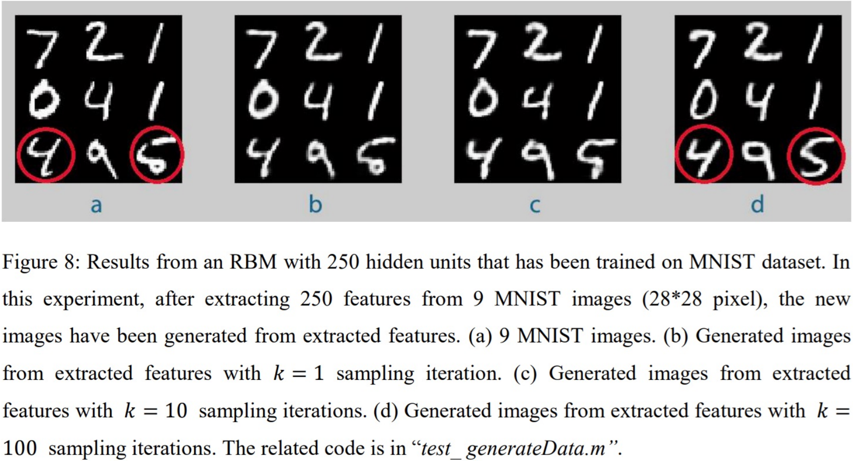


Figure 8: Results from an RBM with 250 hidden units that has been trained on MNIST dataset. In this experiment, after extracting 250 features from 9 MNIST images (28*28 pixel), the new images have been generated from extracted features. (a) 9 MNIST images. (b) Generated images from extracted features with $k = 1$ sampling iteration. (c) Generated images from extracted features with $k = 10$ sampling iterations. (d) Generated images from extracted features with $k = 100$ sampling iterations. The related code is in "*test_ generateData.m*".

Images are from Keyvanrad and Homayounpour (2014).

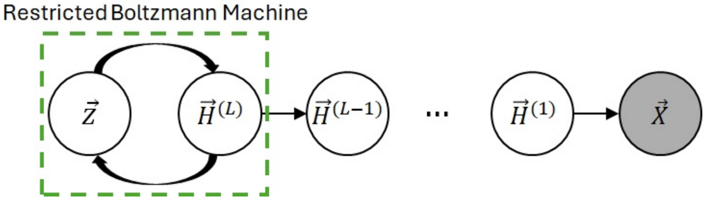
Restricted Boltzmann Machine: Flexibility vs. Interpretability

- Compared with Boltzmann machines, RBMs reduce the flexibility of density models by removing connections within $\vec{z}$ and within $\vec{x}$ while enhancing the interpretability.
- RBMs are (distributional) universal approximators, meaning that the reduced model class is flexible enough to approximate arbitrary distribution:
  Theorem 2.4. in Le Roux and Bengio (2008): For any $\epsilon > 0$ and any distribution $\mathbb{P}(\vec{x})$ defined on $\{0, 1\}^p$, there exists a parameter θ for an RBM such that $\mathrm{KL}(\mathbb{P}(\vec{x}) || \mathbb{P}_\theta(\vec{x})) < \epsilon$.

Deep Energy-based Model

- Motivated by RBMs, several of the earliest deep generative models have been proposed, including deep belief networks (DBNs) (Bengio et al., 2006) and deep Boltzmann machines (Salakhutdinov and Hinton, 2009).
- Recently, from another perspective in the score-matching literature, deep energy estimator networks (Saremi et al., 2018) have been proposed. These integrate neural networks into modeling energy functions and minimize Fisher divergence rather than KL divergence in principles of likelihood maximization.
- In the remainder of this subsection, we review DBNs.

Deep Belief Network



- Deep belief networks (Bengio et al., 2006) extend RBMs by stacking hidden layers.

$$p_\theta(\vec{z}, \vec{h}^{(L)}, \vec{h}^{(L-1)}, \ldots, \vec{h}^{(1)}, \vec{x}) := p_\theta(\vec{z}, \vec{h}^{(L)}) p_\theta(\vec{h}^{(L-1)} | \vec{h}^{(L)}) p_\theta(\vec{h}^{(1)} | \vec{h}^{(2)}) p_\theta(\vec{x} | \vec{h}^{(1)}). \quad (13)$$

Here, $p_\theta(\vec{h}^{(l-1)} | \vec{h}^{(l)}) = \prod_i p_\theta(\vec{h}_i^{(l-1)} | \vec{h}^{(l)})$ and

$$\vec{H}_i^{(l-1)} | \vec{H}^{(l)} \sim \mathrm{Ber}\Big(\sigma(\vec{b}^{(l)} + W^{(l)} \vec{H}^{(l)})\Big). \quad (14)$$

Deep Belief Network: Reconstruction Result

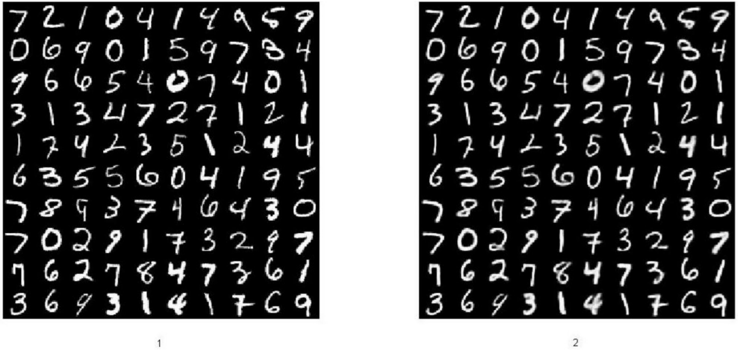


Figure 7: Plotting 100 samples with *plotData* function in *DataStore* class. The first image is 100 samples from MNIST dataset and the second one is reconstructed samples with a DBN model. The related code is in “*test plotData.m”* file.

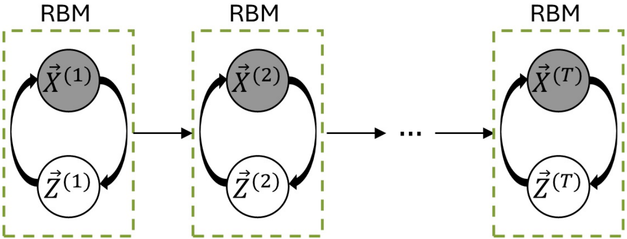
Images are from Keyvanrad and Homayounpour (2014).

Outline

1. Restricted Boltzmann Machine

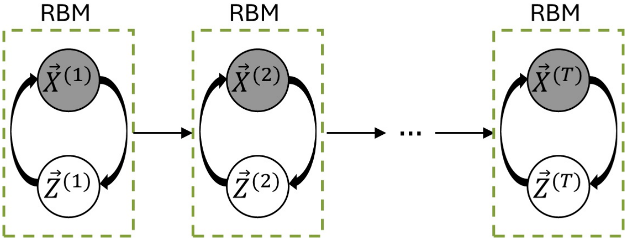
2. Temporal Variants of Restricted Boltzmann Machine

Temporal Restricted Boltzmann Machine: Density Model



- RBMs face challenges in handling temporal data because they do not allow within connections between observations from different time steps.
- Temporal RBMs (TRBMs) (Taylor et al., 2006) are a temporal extension of RBMs that allows connections from past time variables to present ones.

Temporal Restricted Boltzmann Machine: Density Model

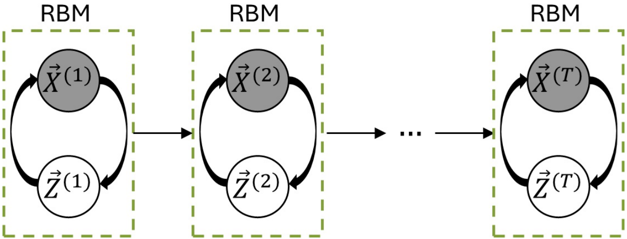


- TRBMs first build an RBM for the initial time step:

$$p_\theta(\vec{z}^{(1)}, \vec{x}^{(1)}) \propto \exp\Big(-E_\theta^{(1)}(\vec{z}^{(1)}, \vec{x}^{(1)})\Big) \tag{15}$$

where $E_\theta^{(1)}(\vec{z}^{(1)}, \vec{x}^{(1)}) := -((\vec{z}^{(1)})^T W \vec{x}^{(1)} + \vec{\alpha}^T \vec{z}^{(1)} + \vec{\beta}^T \vec{x}^{(1)})$.

Temporal Restricted Boltzmann Machine: Density Model

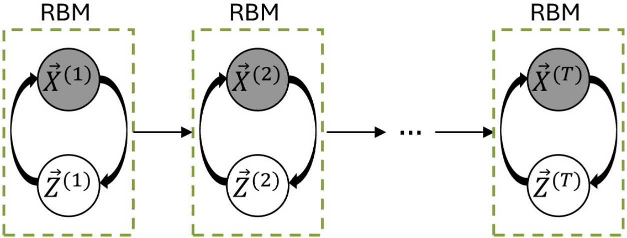


- Next, the distribution of $(\vec{Z}^{(2)}, \vec{X}^{(2)})$ given the initial time step variables can be expressed as:

$$p_\theta(\vec{z}^{(2)}, \vec{x}^{(2)} | \vec{z}^{(1)}, \vec{x}^{(1)}) \propto \exp(-E_\theta^{(2)}(\vec{z}^{(2)}, \vec{x}^{(2)}; \vec{z}^{(1)}, \vec{x}^{(1)})) \qquad (16)$$

where $E_\theta^{(2)}(\vec{z}^{(2)}, \vec{x}^{(2)}; \vec{z}^{(1)}, \vec{x}^{(1)}) := -((\vec{z}^{(2)})^T W \vec{x}^{(2)} + \left(\vec{\alpha} + U\vec{z}^{(1)}\right)^T \vec{z}^{(2)} + \vec{\beta}^T \vec{x}^{(2)})$.

Temporal Restricted Boltzmann Machine: Density Model

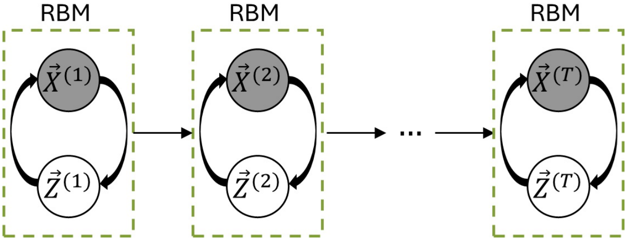


- Finally, TRBMs assume time homogeneity and sequentially stack RBMs for next time steps:

$$p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}) \propto \exp(-E_\theta^{(t)}(\vec{z}^{(t)}, \vec{x}^{(t)}; \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))})) \qquad (17)$$

where the energy is defined as $-((\vec{z}^{(t)})^T W \vec{x}^{(t)} + \left(\vec{\alpha} + U\vec{z}^{(t-1)}\right)^T \vec{z}^{(t)} + \vec{\beta}^T \vec{x}^{(t)})$.

Temporal Restricted Boltzmann Machine: Density Model



- Note that $E_\theta^{(t)}(\vec{z}^{(t)}, \vec{x}^{(t)}; \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))})$ is a function of $(\vec{z}^{(t-1)}, \vec{z}^{(t)}, \vec{x}^{(t)})$. Consequently, $p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}) = p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(t-1)})$.
- That is, when inferring (the distribution of) $(\vec{Z}^{(t)}, \vec{X}^{(t)})$, TRBMs assume that $\vec{Z}^{(t-1)}$ contains all relevant information from $(\vec{Z}^{(1:(t-2))}, \vec{X}^{(1:(t-1))})$.

Temporal Restricted Boltzmann Machine: Inference

- Since $p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))})$ is an RBM with $E_\theta^{(t)}$, it enjoys conditional independency, i.e., $p_\theta(\vec{x}^{(t)} | \vec{z}^{(1:t)}, \vec{x}^{(1:(t-1))}) = \prod_{j=1}^{p} p_\theta(x_j^{(t)} | \vec{z}^{(1:t)}, \vec{x}^{(1:(t-1))})$ and $p_\theta(\vec{z}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:t)}) = \prod_{j=1}^{d} p_\theta(z_j^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:t)})$.
- TRBMs also have an interpretable inference:

$$p_\theta(z_j^{(t)} = 1 | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:t)}) = \text{Sigmoid}\Big((W\vec{x}^{(t)} + \vec{\alpha} + U\vec{z}^{(t-1)})_j \Big). \tag{18}$$

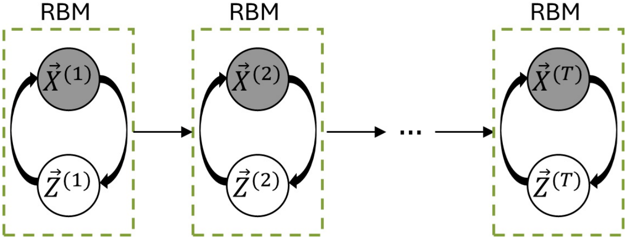
  Similarly,

$$p_\theta(x_j^{(t)} = 1 | \vec{z}^{(1:t)}, \vec{x}^{(1:(t-1))}) = \text{Sigmoid}\Big((W^T\vec{z}^{(t)} + \vec{\beta})_j \Big). \tag{19}$$

- Their (conditional) logit functions can be expressed as:

$$\log \frac{p_\theta(z_1^{(t)}, \ldots, z_j^{(t)} = 1, \ldots, z_d^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:t)})}{p_\theta(z_1^{(t)}, \ldots, z_j^{(t)} = 0, \ldots, z_d^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:t)})} = (W\vec{x}^{(t)} + \vec{\alpha} + U\vec{z}^{(t-1)})_j. \tag{20}$$

Temporal Restricted Boltzmann Machine: Generation



- TRBMs can sequentially generate data using RBMs corresponding to each time step:
  1. Apply M-step Gibbs-sampling to generate $(\tilde{\vec{Z}}_M^{(1)}, \tilde{\vec{X}}_M^{(1)})$ whose limit distribution is $p_\theta(\vec{z}^{(1)}, \vec{x}^{(1)})$.
  2. For $t = 2, \ldots, T$, apply M-step Gibbs-sampling to generate $(\tilde{\vec{Z}}_M^{(t)}, \tilde{\vec{X}}_M^{(t)})$ whose limit distribution is $p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))} = \tilde{\vec{Z}}_M^{(1:(t-1))}, \vec{x}^{(1:(t-1))} = \tilde{\vec{X}}_M^{(1:(t-1))})$.
  3. The $(\tilde{\vec{Z}}_M^{(1:T)}, \tilde{\vec{X}}_M^{(1:T)})$ is the generated data.

Temporal Restricted Boltzmann Machine: Training

- The $p_\theta(\vec{z}^{(1:T)}, \vec{x}^{(1:T)})$ is proportional to $\exp(-\sum_{t=1}^{T}(E_\theta^{(t)} + \log C^{(t)}))$. This can be viewed as an RBM with latent variables $\vec{z}^{(1:T)}$, observations $\vec{x}^{(1:T)}$, and an energy function $\sum_{t=1}^{T}(E_\theta^{(t)}(\vec{z}^{(t)}, \vec{x}^{(t)}; \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}) + \log C^{(t)}(\theta; \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}))$.
- We can apply results from usual CD algorithms. For example, Equation (12) implies:

$$\begin{aligned}\partial \log p_\theta(\vec{x}^{(1:T)})/\partial\theta = &- \sum_{\vec{z}^{(1:T)}} \Big(\sum_{t=1}^{T}(\partial E_\theta^{(t)}/\partial\theta + \partial \log C^{(t)}/\partial\theta)\Big) p_\theta(\vec{z}^{(1:T)}|\vec{x}^{(1:T)}) \\ &+ \sum_{\vec{z}^{(1:T)}, \vec{x}^{(1:T)}} \Big(\sum_{t=1}^{T}(\partial E_\theta^{(t)}/\partial\theta + \partial \log C^{(t)}/\partial\theta)\Big) p_\theta(\vec{z}^{(1:T)}, \vec{x}^{(1:T)}).\end{aligned} \tag{21}$$

- Here, $\partial \log C^{(t)}/\partial\theta = -\sum_{\vec{z}^{(t)}, \vec{x}^{(t)}}(\partial E_\theta^{(t)}/\partial\theta) p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)}|\vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))})$.

Temporal Restricted Boltzmann Machine: Training

- In implementation, the coefficient for latent variables, $\vec{\alpha}$ is separately modeled for $t = 1$ and $t \geq 2$.
- Several computational techniques can be applied to generate positive and negative samples, and alternative approximations can be used for implementing the CD algorithm for TRBMs. See Taylor et al. (2006) for details.

Temporal Restricted Boltzmann Machine: Generation Result

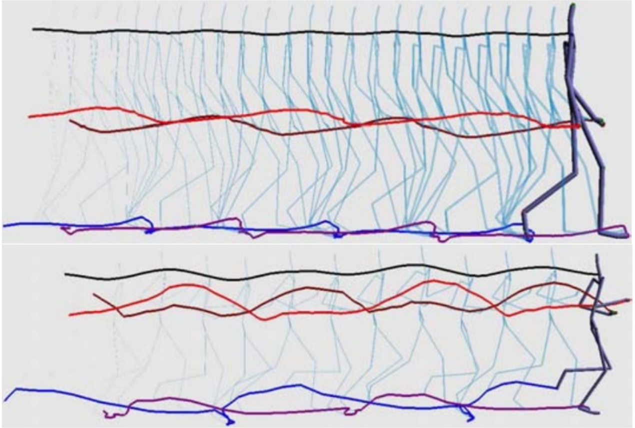
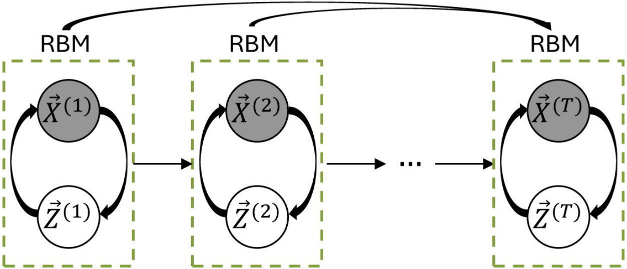


Figure 3: After training, the same model can generate walking (top) and running (bottom) motion (see videos on the website). Each skeleton is 4 frames apart.

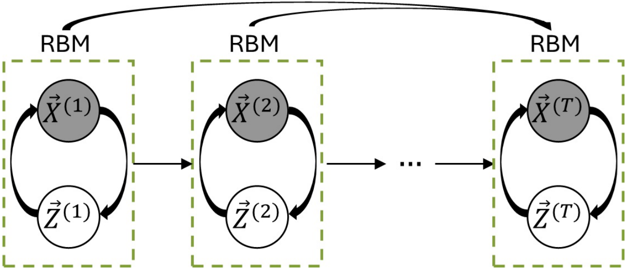
Images are from Taylor et al. (2006).

Recurrent Temporal Restricted Boltzmann Machine



- Recurrent temporal RBMs (RTRBMs) (Sutskever et al., 2008) introduce a recurrent structure to model long-term dependency structures.
- As in TRBMs, $p_\theta(\vec{z}^{(1)}, \vec{x}^{(1)})$ is an RBM.

Recurrent Temporal Restricted Boltzmann Machine: Density Model



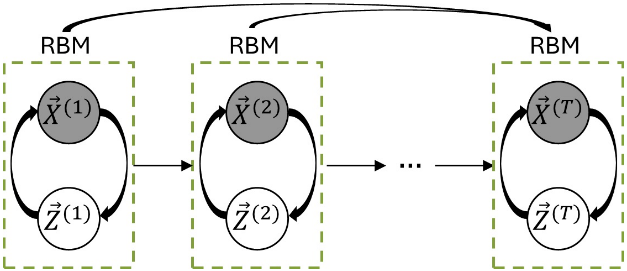
- For $t = 2$, the energy function is modified as follows:

$$p_\theta(\vec{z}^{(2)}, \vec{x}^{(2)} | \vec{z}^{(1)}, \vec{x}^{(1)}) \propto \exp\left((\vec{z}^{(2)})^T W \vec{x}^{(2)} + \left(\vec{\alpha} + U\vec{r}^{(1)}\right)^T \vec{z}^{(2)} + \vec{\beta}^T \vec{x}^{(2)}\right) \tag{22}$$

where $r_j^{(1)} := \mathbb{P}_\theta(Z_j^{(1)} = 1 | \vec{x}^{(1)}) = \sigma\left((W\vec{x}^{(1)} + \vec{\alpha})_j\right)$.

- The modification involves replacing $\vec{z}^{(1)}$ in the energy function with $\vec{r}^{(1)}$.

Recurrent Temporal Restricted Boltzmann Machine: Density Model



- For further time steps:

$$p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}) \propto \exp\left((\vec{z}^{(t)})^T W \vec{x}^{(t)} + \left(\vec{\alpha} + U \vec{r}^{(t-1)}\right)^T \vec{z}^{(t)} + \vec{\beta}^T \vec{x}^{(t)}\right) \tag{23}$$

where $r_j^{(t-1)} := \mathbb{P}_\theta(Z_j^{(t-1)} = 1 | \vec{x}^{(t-1)}, \vec{r}^{(t-2)}) = \sigma\left((W\vec{x}^{(t-1)} + \vec{\alpha} + U\vec{r}^{(t-2)})_j\right)$.

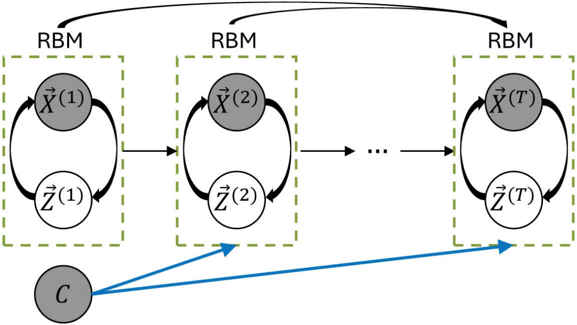
Recurrent Temporal Restricted Boltzmann Machine: Notes on the Recurrent Unit

- For a given parameter θ, the recurrent nature of $\vec{r}^{(t-1)}$ manifests itself as a (deterministic) function of $\vec{x}^{(1:(t-1))}$.
- Compared with TRBMs that employ $\vec{Z}^{(t-1)}$ in their temporal dependency model, RTRBMs provide greater stability and are more effective at incorporating information from observations at earlier time steps.
- Since $\vec{r}^{(t)}$ is a function of θ, training RTRBMs requires recursively computing $\partial \vec{r}^{(t)} / \partial \theta$. See Section 3.2. in Mittelman et al. (2014) for details.

(R)TRBMs are Universal Approximators

- Both TRBMs and RTRBMs are (distributional) universal approximators for arbitrary finite time series of binary random vectors that are time-homogeneous and hold Markov property:
  Theorem 1 in Odense and Edwards (2016): For any $\epsilon > 0$ and any distribution $\mathbb{P}(\vec{x}^{(1:T)})$ defined on $\{0,1\}^T$, if $\mathbb{P}(\vec{x}^{(1:T)})$ is time-homogeneous with finite time dependence, then there exists a parameter θ for a TRBM such that $\mathrm{KL}(\mathbb{P}(\vec{x}^{(1:T)})||\mathbb{P}_\theta(\vec{x}^{(1:T)})) < \epsilon$.
- Results for RTRBMs can be found at the reference.

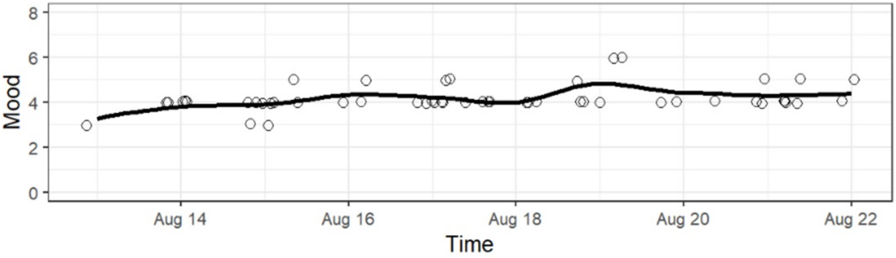
Heterogeneous-Dynamic Restricted Boltzmann Machine



- Heterogeneous-Dynamic RBMs (HDRBMs) (Kim et al., 2024) introduce an additional information C (e.g., comorbidity) to model heterogeneous group dynamics:

$$\begin{aligned} & p_\theta(\vec{z}^{(t)}, \vec{x}^{(t)} | \vec{z}^{(1:(t-1))}, \vec{x}^{(1:(t-1))}, c) \\ & \propto \exp\left((\vec{z}^{(t)})^T W \vec{x}^{(t)} + \left(\vec{\alpha} + (U + cV) \vec{r}^{(t-1)} \right)^T \vec{z}^{(t)} + \vec{\beta}^T \vec{x}^{(t)} \right). \end{aligned} \tag{24}$$

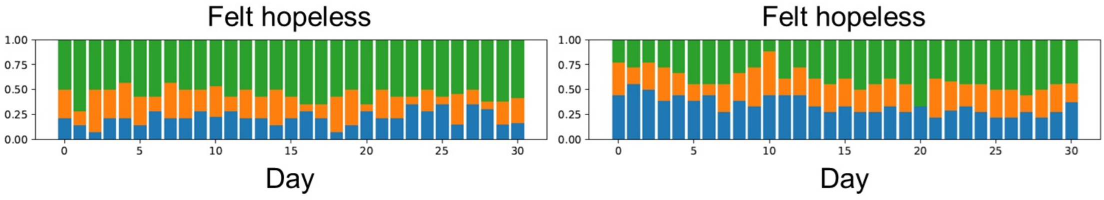
HDRBM: Ecological Momentary Assessment Data



- Ecological momentary assessment (EMA) data are mobile health data that evaluate participants' daily moods.
- EMA data may show distinct patterns in people with comorbidities.

Images are from Ruwaard et al. (2018).

HDRBM: Ecological Momentary Assessment Data



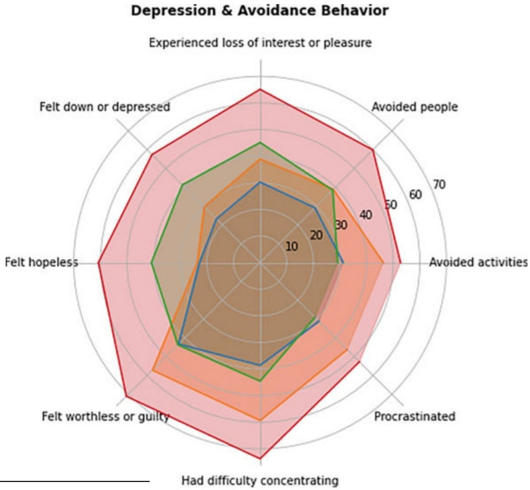
(a) Comorbid group

(b) Others

- We used 23 item responses related to mood as $\vec{X}^{(t)}$ and the comorbidity as C. The dimension of $\vec{Z}^{(t)}$ was two.
- There were total 32 participants: Generalized anxiety disorder $(n = 13)$; Major depressive disorder $(n = 5)$; Both $(n = 14)$.
- Mental symptoms were not static over time but rather volatile.

Images are from Kim et al. (2024).

HDRBM: Ecological Momentary Assessment Data



The image is from Kim et al. (2024).

ICA vs. Auto-regressive Model vs. Energy-based Model

- Independent Component Analysis: $\mathbb{P}_\theta(\vec{x}) = \int \delta_{\vec{x}}(W\vec{z})d\mathbb{P}(\vec{z})$
- Auto-regressive Model: $p_\theta(\vec{x}) = p_\theta(x_1)p_\theta(x_2|x_1)\cdots p_\theta(x_p|x_1,\ldots,x_{p-1})$
- Energy-based Model: $p_\theta(\vec{x}) = \int C(\theta)^{-1}\exp(-E_\theta(\vec{z},\vec{x}))d\vec{z}$.
  Q: What would be their strong and weak points?

References I

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2006). Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.

Keyvanrad, M. A. and Homayounpour, M. M. (2014). A brief survey on deep belief networks and introducing a new object oriented toolbox (deebnet). *arXiv preprint arXiv:1408.3264*.

Kim, S., Kim, Y.-g., and Wang, Y. (2024). Temporal generative models for learning heterogeneous group dynamics of ecological momentary assessment data. *Biometrics*, 80(4):ujae115.

References II

Le Roux, N. and Bengio, Y. (2008). Representational power of restricted boltzmann machines and deep belief networks. *Neural computation*, 20(6):1631–1649.

Mittelman, R., Kuipers, B., Savarese, S., and Lee, H. (2014). Structured recurrent temporal restricted boltzmann machines. In *International Conference on Machine Learning*, pages 1647–1655. PMLR.

Odense, S. and Edwards, R. (2016). Universal approximation results for the temporal restricted boltzmann machine and the recurrent temporal restricted boltzmann machine. *Journal of Machine Learning Research*, 17(158):1–21.

Ruwaard, J., Kooistra, L., and Thong, M. (2018). Ecological momentary assessment in mental health research: A practical introduction, with examples in r (–build 2018-11-26). *Amsterdam: APH Mental Health*.

Salakhutdinov, R. and Hinton, G. (2009). Deep boltzmann machines. In *Artificial intelligence and statistics*, pages 448–455. PMLR.

References III

Saremi, S., Mehrjou, A., Schölkopf, B., and Hyvärinen, A. (2018). Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*.

Smolensky, P. et al. (1986). Information processing in dynamical systems: Foundations of harmony theory.

Sutskever, I., Hinton, G. E., and Taylor, G. W. (2008). The recurrent temporal restricted boltzmann machine. *Advances in neural information processing systems*, 21.

Taylor, G. W., Hinton, G. E., and Roweis, S. (2006). Modeling human motion using binary latent variables. *Advances in neural information processing systems*, 19.

Welling, M., Rosen-Zvi, M., and Hinton, G. E. (2004). Exponential family harmoniums with an application to information retrieval. *Advances in neural information processing systems*, 17.