Continuous Interpolation and Sampling of High-Dimensional Probability Distributions

Hongli Zhao

January 19, 2022

Why Tensor-Networks

- Tensor-Network offers a representation of quantum many-body states:

$$|\Psi\rangle = \sum_{i_1 \cdots i_N} C_{i_1 \cdots i_N} |i_1\rangle \otimes \cdots \otimes |i_N\rangle$$

  an N-particle, p-state system has p^N coefficients.
- Premise: particles have local interactions; the system can be well-approximated with fewer indices.
  - (simplified Ising model) $\exp\left(-\frac{1}{T}\sum_{i,j} J_{ij}\sigma_i\sigma_j\right)$
- Tensor-Train / Matrix Product States is an example of a *linear tensor-network*
  - represents a product measure exactly
  - can show denseness in Hilbert space
- First construed in 1992 [Fannes, Nachtergaele, Werner][1] and 1993 [Klümper, Schadschneider, Zittartz][2]
  - Rediscovered in 2011 by Ivan Oseledets[3]

[1] (1992) Finitely correlated pure states. and their symmetries

[2] (1993) Matrix Product Ground States for One-Dimensional Spin-1 Quantum Antiferromagnets

[3] (2011) Tensor-Train Decomposition

Graphical Representation of a Tensor

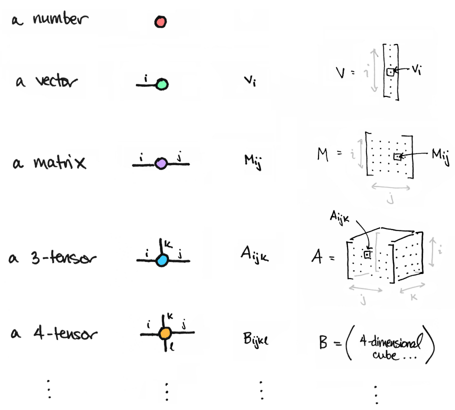

Figure: Tensors as nodes and edges

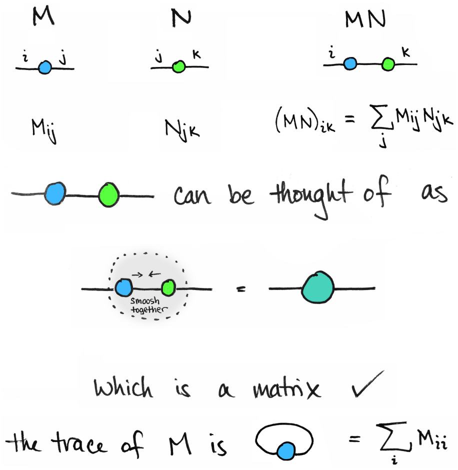

Figure: Tensor contractions as connecting edges

Review of Tensor-Train Decomposition

And Notations

- A tensor of size $n_1 \times n_2 \times \cdots \times n_d$ requires $O(n^d)$ storage

$$\mathbf{A}(i_1, i_2, \cdots, i_d) \approx \mathbf{C}_1(i_1) \cdot \mathbf{C}_2(i_2) \cdot \cdots \cdot \mathbf{C}_d(i_d)$$

$$= \sum_{\alpha_0, \alpha_2, \cdots, \alpha_{d-1}, \alpha_d}^{r_0, r_1, \cdots, r_d} \mathbf{C}_1(\alpha_0, i_1, \alpha_1) \cdot \mathbf{C}_2(\alpha_1, i_2, \alpha_2) \cdots \mathbf{C}_d(\alpha_{d-1}, i_d, \alpha_d)$$

here we have *open* boundary conditions $r_0 = r_d = 1$. If $\alpha_d = \alpha_1$, it is called a *tensor-ring*.

Figure: Tensor-Train (left); Tensor-Ring (right)

- Advantages:
  - Storage depends linearly on d, but cubically on r
    - ★ Important to seek low-rank decompositions
  - Cost of linear algebra operations [1] depends linearly on d:

Operation	Cost
scalar add/mult.	$O(dnr^3)$
contraction	$O(dnr + dr^3)$
Hadamard product, dot product[2]	$O(dnr^2 + dr^4)$
matrix-vector multiply (TT format)	$O(dn^2r^4)$

- Other relevant algorithms:
  - TT-Round: Given TT $\mathbf{A}$, compress $\mathbf{B}$ such that $\frac{\|\mathbf{A}-\mathbf{B}\|_F}{\|\mathbf{A}\|_F} \leq \epsilon$ for some pre-specified ϵ or rank.
  - TT-Cross (AMEn-Cross, DMRG-Cross): Given a procedure to compute tensor elements, construct a low-parametric approximation to the tensor using a small number of evaluations.

[1] Implementations available in MATLAB, Python, C++, Julia (in progress)

[2] Can be obtained from computing a Hadamard product, then contracting with a tensor of all 1's.

Problem Statement

- We are interested in sampling from a target distribution of the Boltzmann-Gibbs form:

$$\pi(\mathbf{x}) = \frac{1}{Z_\beta} \exp(-\beta V(\mathbf{x}))$$

  where $V : \mathbb{R}^d \to \mathbb{R}$ is some energy potential, $Z_\beta = \int_\Omega \exp(-\beta V) d\mathbf{x}$ is the partition function that is often unknown.
- Issues with metastability: transition between metastable regions is a rare event
- For non-Gaussian distributions, typically use a variant of Metropolis-Hastings MCMC
  - requires multiple evaluations to generate independent samples
- General purpose sampler for un-normalized high-dimensional and multi-modal distributions?

Conditional Distribution Sampling

Decompose:

$$\pi(x_1, x_2, \cdots, x_d) = \pi_1(x_1) \cdot \pi_2(x_2|x_1) \cdots \pi_d(x_d|x_1, x_2, \cdots, x_{d-1})$$

where:

$$\pi_k(x_k|x_1, x_2, \cdots, x_{k-1}) = \frac{\int \pi(x_1, \cdots, x_{k-1}, x_k, x_{k+1}, \cdots, x_d) dx_{k+1} \cdots dx_d}{\int \pi(x_1, \cdots, x_{k-1}, x_k, \cdots, x_d) dx_k \cdots dx_d}$$

```
for i = 1, 2, ..., d do
   sample x_i ~ π_i
end
```

- Evaluation of high-dimensional integrals is costly
- However, a *surrogate model* can help us → tensor-train approximation
  - *[Dolgov 2020] Approximation and sampling of multivariate probability distributions in the tensor train decomposition*

Aside: Evaluating High-Dimensional Integrals in TT Format

Let $f : \mathbb{R}^d \to \mathbb{R}$, and quadrature be given by index set $I_1 \times I_2 \times \cdots \times I_d$ (assume discretization level N), with appropriate weights $\mathbf{w}$ for each dimension.

- (Recall 1d) Discretize $\mathbf{f} = \begin{pmatrix} f^1 \\ f^2 \\ \vdots \\ f^N \end{pmatrix}$, with $\mathbf{w} = \begin{pmatrix} w^1 \\ w^2 \\ \vdots \\ w^N \end{pmatrix}$, then:

$$\int f(x)dx \approx \sum_{k=1}^{N} \mathbf{w}_k \mathbf{f}_k = \mathbf{w}^T \mathbf{f}$$

- (General, formal)

$$\int f(\mathbf{x})dx_1 dx_2 \cdots dx_d \approx \sum_{i_1 i_2 \cdots i_d} \mathbf{f}_{i_1 i_2 \cdots i_d} \mathbf{w}_{i_1} \mathbf{w}_{i_2} \cdots \mathbf{w}_{i_d}$$

- (General, TT) Approximate:

$$\mathbf{f}_{i_1 i_2 \cdots i_d} \approx \sum_{\alpha_0, \cdots, \alpha_{d-1}, \alpha_d} \mathbf{C}_1(\alpha_0, i_1, \alpha_1) \cdot \mathbf{C}_2(\alpha_1, i_2, \alpha_2) \cdots \mathbf{C}_d(\alpha_{d-1}, i_d, \alpha_d)$$

then:

$$\begin{aligned} &\int f(\mathbf{x}) dx_1 dx_2 \cdots dx_d \\ &\approx \sum_{i_1 i_2 \cdots i_d} \sum_{\alpha_0, \cdots, \alpha_d} \mathbf{C}_1(\alpha_0, i_1, \alpha_1) \cdot \cdots \cdot \mathbf{C}_d(\alpha_{d-1}, i_d, \alpha_d) \mathbf{w}_{i_1} \cdots \mathbf{w}_{i_d} \\ &= \sum_{\alpha_0, \cdots, \alpha_d} \left(\sum_{i_1} \mathbf{C}_1(\alpha_0, i_1, \alpha_1) \mathbf{w}_{i_1} \right) \cdot \left(\sum_{i_2} \mathbf{C}_2(\alpha_1, i_2, \alpha_2) \mathbf{w}_{i_2} \right) \\ &\quad \cdots \left(\sum_{i_d} \mathbf{C}_d(\alpha_{d-1}, i_d, \alpha_d) \right) \\ &= \mathbf{f}_{TT} \cdot \left\{ \bigotimes_{i=1}^{d} \mathbf{w} \right\} \end{aligned}$$

- The above can be computed sequentially as we loop over the cores $i = 1, 2, \cdots, d$.

Summary of Algorithm

- Input: Cores $\{\mathbf{C}_i\}_{i=1}^d$
- Output: Samples $\{\tilde{\mathbf{x}_n}\}_{n=1}^N$ distributed according to $\tilde{\pi} \approx \pi$
- Loop over each dimension $k = 1, 2, \cdots, d$
- Compute marginal PDF $p_k(x_k)$ vector:
  - If $k = 1$, contract all $k = 2, 3, \cdots, d$ dimensions

$$p_k(x_k) = \mathbf{f}_{TT} \times_2 \mathbf{w} \times_3 \cdots \times_d \mathbf{w}$$

  - If $k > 1$, update core k by multiplying fixed marginal densities $p(\tilde{x_1}), p(\tilde{x_2}), \cdots, p(\tilde{x}_{k-1})$ of sampled entries
- Enforce non-negativity by $p_k \leftarrow |p_k(x_k)|$
- Sample p_k via Inverse Rosenblatt:

$$\tilde{x}_k \leftarrow F_k^{-1}(q_k)$$

where:

$$F_k(z) \propto \int_{-\infty}^{z} p_k(y)dy, q_k \sim U(0, 1)$$

Comments

- Although target π is non-negative, TT-Cross may introduce approximation errors that yield negative values
- Uses piecewise polynomial interpolation to construct continuous TT surrogate: (Linear case)

$$\mathbf{C}_k(:, x_k, :) \leftarrow \frac{x_k - x_k^{i_k}}{x_k^{i_{k+1}} - x_k^{i_k}} \cdot \mathbf{C}_k(:, i_k + 1, :) + \frac{x_k^{i_k+1} - x_k}{x_k^{i_{k+1}} - x_k^{i_k}} \cdot \mathbf{C}_k(:, i_k, :)$$

- Inverse Rosenblatt may be replaced by a "smeared" discrete distribution, i.e.

$$\tilde{x}_k \sim \{c_1, c_2, \cdots, c_I\}$$

$$\tilde{x}_k \leftarrow \tilde{x}_k + \epsilon, \epsilon \sim \mathcal{N}(0, \frac{1}{2}\Delta_k)$$

  where Δ_k is grid size
- Only has likelihood of sampled points $\{\tilde{\mathbf{x}_n}\}$, not easy to evaluate arbitrary points

Continuous TT Expansion

Goal: Want a surrogate TT distribution that enforces non-negativity and cheap to evaluate to arbitrary precision

- Motivating example: Let $f \in L^2(\mathbb{R})$, and an orthonormal basis $\{\phi_i\}$, then:

$$f = \sum_{i=1}^{\infty} \langle f, \phi_i \rangle \cdot \phi_i$$

- **Definition**: (Tensor product of Hilbert spaces) Let $\mathcal{H}_1, \mathcal{H}_2$ be two Hilbert spaces; for each $\phi_1 \in \mathcal{H}_1, \phi_2 \in \mathcal{H}_2$, let $\phi_1 \otimes \phi_2$ denote the conjugate bilinear form acting on $\mathcal{H}_1 \otimes \mathcal{H}_2$ by:

$$(\phi_1 \otimes \phi_2)(\psi_1, \phi_1) = \langle \phi_1, \psi_1 \rangle \cdot \langle \psi_2, \phi_2 \rangle$$

a natural inner product on bilinear forms is defined by:

$$\langle \eta \otimes \mu, \phi \otimes \psi \rangle = \langle \eta, \phi \rangle \cdot \langle \mu, \psi \rangle$$

we then define $\mathcal{H}_1 \otimes \mathcal{H}_2$ as the completion of the set containing all linear combinations of the bilinear forms.

- **(Theorem)**
  1. $\mathcal{H}_1 \otimes \mathcal{H}_2$ is a Hilbert space
  2. Let $\{\phi_n\}, \{\psi_m\}$ be bases for $\mathcal{H}_1, \mathcal{H}_2$, $\{\phi_n \otimes \psi_m\}$ is a basis for $\mathcal{H}_1 \otimes \mathcal{H}_2$.
  3. Let $L^2(\Omega_1, \mu_1), L^2(\Omega_2, \mu_2)$ be two separable Hilbert spaces with bases $\{\phi_n\}, \{\psi_m\}$,
  $$L^2(\Omega_1 \times \Omega_2, \mu_1 \otimes \mu_2)$$
  is isomorphic to
  $$L^2(\Omega_1, \mu_1) \otimes L^2(\Omega_2, \mu_2)$$
- Recall for orthonormal bases:

$$\int_\Omega \phi_i^2 = 1, \int_\Omega \phi_i \phi_j = 0, (i \neq j)$$

Let square-integrable $f : \Omega \to \mathbb{R}$ ($\Omega \subset \mathbb{R}^d$), let $\{\phi_i\}$ be an orthonormal basis for $L^2(\Omega)$ (e.g. Legendre polynomials). Then f has the unique decomposition:

$$f(x_1, x_2, \cdots, x_d) = \sum_{i_1 i_2 \cdots i_d}^{\infty} \mathbf{A}_{i_1 i_2 \cdots i_d} \phi_{i_1}(x_1) \phi_{i_2}(x_2) \cdots \phi_{i_d}(x_d)$$

- However, $\mathbf{A}_{i_1 \cdots i_d}$ has exponential dependence on dimensions
- Seek:

$$\mathbf{A}_{i_1 \cdots i_d} \approx \sum_{\alpha_0, \cdots, \alpha_d} \mathcal{C}_1(\alpha_0, i_1, \alpha_1) \cdots \mathcal{C}_d(\alpha_{d-1}, i_d, \alpha_d)$$

- Questions:
  1. How to obtain **A**?
  2. How to enforce non-negativity?
  3. Given **A**, how to sample efficiently from the surrogate distribution?

Obtaining coefficient tensor

- (1d example) Choose collocation points $\{x^{(j)}\}_{j=1}^{N}$ along with quadrature weights $\mathbf{w}$, a finite number of bases $\{\phi_i\}_{i=1}^{M}$. Let:

$$f \approx \sum_{i=1}^{M} a_i \phi_i$$

enforce equality on collocation points:

$$\underbrace{\begin{pmatrix} f(x^{(1)}) \\ f(x^{(2)}) \\ \vdots \\ f(x^{(N)}) \end{pmatrix}}_{\mathbf{f}} = \underbrace{\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1M} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2M} \\ \vdots & \cdots & \ddots & \vdots \\ \phi_{N1} & \phi_{N2} & \cdots & \phi_{NM} \end{pmatrix}}_{\text{feature matrix}, \boldsymbol{\Phi}} \cdot \underbrace{\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{pmatrix}}_{\text{coefficient tensor}, \mathbf{a}}$$

then:

$$\mathbf{a} = \boldsymbol{\Phi}^{\dagger} \mathbf{f}$$

- Comments:
  - Usually take $N = p + 1$
  - Pseudoinverse may be ill-conditioned

- (Alternative)

$$f \approx \sum_{i=1}^{M} a_i \phi_i$$

then for $j = 1, 2, \cdots, M$:

$$\int_\Omega \left(\sum_{i=1}^{M} a_i \phi_i \right) \phi_j = \sum_i \underbrace{\int_\Omega \phi_i \phi_j}_{=\delta_{i=j}} = a_j = \int_\Omega f \phi_j \approx \sum_{k=1}^{N} w_k f(x^{(k)}) \phi_j(x^{(k)})$$

- (In vector form)

$$\mathbf{a} = \tilde{\mathbf{\Phi}}^T \cdot \mathbf{f}$$

where:

$$\tilde{\mathbf{\Phi}}(:, k) \leftarrow \mathbf{w} \circ \mathbf{\Phi}(:, k)$$

Obtaining coefficient tensor: Generalization

- For each dimension, the coefficients can be solved via:

$$\mathbf{a} = \mathbf{D} \cdot \mathbf{f}$$

  where $\mathbf{D}$ is some form of data matrix.
- Let $\mathbf{F}$ be a tensor, then we have the following generalization:

$$A_{i_1 \cdots i_d} = \sum_{j_1 \cdots j_d} D_{i_1 j_1} \cdots D_{i_d j_d} F_{j_1 \cdots j_d}$$

- Approximate:

$$F_{j_1 \cdots j_d} \approx \sum_{\beta_0, \cdots, \beta_d} \mathcal{C}(\beta_0, j_1, \beta_1) \cdots \mathcal{C}_d(\beta_{d-1}, j_d, \beta_d)$$

  consequently:

$$A_{i_1 \cdots i_d} \approx$$

$$\sum_{j_1 \cdots j_d} D_{i_1 j_1} \cdots D_{i_d j_d} \left(\sum_{\beta_0, \cdots, \beta_d} \mathcal{C}(\beta_0, j_1, \beta_1) \cdots \mathcal{C}_d(\beta_{d-1}, j_d, \beta_d) \right)$$

$$=$$

$$\sum_{\beta_0, \cdots, \beta_d} \left(\sum_{j_1} \mathcal{C}_1(\beta_0, j_1, \beta_1) \cdot D^T_{j_1 i_1} \right) \cdots \left(\sum_{j_d} \mathcal{C}_d(\beta_{d-1}, j_d, \beta_d) \cdot D^T_{j_d i_d} \right)$$

Non-negativity on interpolated points

- Given target probability distribution $\pi(\mathbf{x})$, TT-cross $p(\mathbf{x}) = \sqrt{\pi(\mathbf{x})}$ instead
- $p(\tilde{\mathbf{x}})$ can then be evaluated in $O(dnr + dr^3)$ via tensor contraction $\Rightarrow$ May recover $\pi(\mathbf{x}) = p^2(\mathbf{x})$
  - Here $\tilde{\mathbf{x}}$ can be arbitrary because we have analytic forms of the basis

Non-negativity of marginals

- Let I, J denote multi-index $\mathcal{I} = (i_1, i_2, \cdots, i_d), \mathcal{J} = (j_1, j_2, \cdots, j_d)$, and:

$$p(\mathbf{x}) = \sum_{\mathcal{I}} \mathbf{A}_{\mathcal{I}} \psi_{\mathcal{I}}(\mathbf{x})$$

where $\psi_{\mathcal{I}} = \phi_{i_1} \phi_{i_2} \cdots \phi_{i_d}$ then:

$$p(\mathbf{x})^2 = \sum_{\mathcal{I}, \mathcal{J}} \mathbf{A}_{\mathcal{I}} \mathbf{A}_{\mathcal{J}} \psi_{\mathcal{I}} \psi_{\mathcal{J}}$$

substituting in tensor-train:

$$\approx \sum_{i_1, \cdots, i_d, j_1, \cdots, j_d} \mathbf{A}_{i_1 \cdots i_d} \mathbf{A}_{j_1 \cdots j_d} (\phi_{i_1} \phi_{j_1}) \cdots (\phi_{i_d} \phi_{j_d})$$

- Then the marginal p_1 is obtained as:

$$p_1 = \int_{\Omega_2 \times \cdots \times \Omega_d} \pi(\mathbf{x}) dx_2 \cdots dx_d =$$

$$\int_{\Omega_2 \times \cdots \times \Omega_d} \sum_{i_1, \cdots, i_d, j_1, \cdots, j_d} \mathbf{A}_{i_1 \cdots i_d} \mathbf{A}_{j_1 \cdots j_d} (\phi_{i_1} \phi_{j_1}) \cdots (\phi_{i_d} \phi_{j_d}) dx_2 \cdots dx_d$$

by orthonormality:

$$= \sum_{\mathcal{I}, \mathcal{J}} \underbrace{\mathbf{A}_{i_1 i_2 \cdots i_d} \mathbf{A}_{j_1 i_2 \cdots i_d}}_{=: G_{i_1 j_1}} (\phi_{i_1} \phi_{j_1})$$

- Definition: Let $\mathbf{T}$ be a multi-dimensional array with size $(n_1, n_2, \cdots, n_d)$, the k-th *unfolding* refers to the matrix:

$T_{i_1 \cdots i_k, i_{k+1} \cdots i_d} =$ `reshape(T, prod(n1:nk-1), prod(nk:nd))`

- Let S denote the first unfolding of $\mathcal{A}$, then:

$$G = SS^T$$

is positive semidefinite by construction. Then we have:

$$p_1(z) = \phi(z)^T SS^T \phi(z) = [S^T \phi(z)]^T [S^T \phi(z)]$$

Valid Probability Distribution

- The above surrogate in fact defines a distribution even though partition function of the target is unknown, if we set:

$$\mathbf{A} \leftarrow \frac{\mathbf{A}}{\|\mathbf{A}\|_F}$$

- $$\int \tilde{\pi}(\mathbf{x}) d\mathbf{x} = \int p(\mathbf{x})^2 d\mathbf{x} = \int \sum_{\mathcal{I},\mathcal{J}} \mathbf{A}_{\mathcal{I}} \mathbf{A}_{\mathcal{J}} \psi_{\mathcal{I}} \psi_{\mathcal{J}} d\mathbf{x}$$

$$= \sum_{i_1,\cdots,i_d,j_1,\cdots,j_d} \mathcal{A}_{i_1\cdots i_d} \mathcal{A}_{j_1\cdots j_d} \Big(\int \phi_{i_1} \phi_{j_1} dx_1 \Big) \cdots \Big(\int \phi_{i_d} \phi_{j_d} dx_d \Big)$$

$$= \sum_{i_1,\cdots,i_d,j_1,\cdots,j_d} \mathbf{A}^2_{i_1\cdots i_d} = \|\mathbf{A}\|_F^2 = 1$$

- In addition, can put **A** in "left-right" QR form
  - For x_k, tensor contraction (integrating out variables $(x_1, \cdots, x_{k-1}, x_{k+1}, \cdots, x_d)$) is identity
  - Can essentially sample N points in $O(Nd)$

Questions?