Random Number Characterization via Quantum Inspired Machine Learning

Tommaso Faorlin, Alessandro Marcomini, Samuele Piccinelli
tommaso.faorlin@studenti.unipd.it, a.marcomini@fz-juelich.de, s.piccinelli@fz-juelich.de

Università degli Studi di Padova
JÜLICH Forschungszentrum
Stockholm University

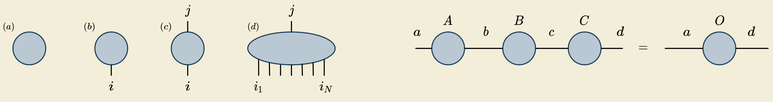
1. Introduction

The generation of good random numbers impacts research and applications beyond pure academic interests, in countless fields such as cryptography and simulations. For most of them, it is of outmost importance to **understand if a set of numbers is truly random** or contains some residual correlations. An innovative approach exploits **Tensor Networks** (TNs), powerful data structures that spring from **quantum many-body physics** and are now increasingly applied to machine learning (ML) applications.

In this work we **review** these architectures and investigate their power to study strings of random numbers. Results show how the **linear properties of TNs** allow to **detect long-range correlations** ($\mathcal{O}(2^{21})$) in short pseudo-random sequences and provide a strong **interpretability** to the outputs. On the other hand, they fail to discriminate quantum randomness from state-of-the-art pseudo random samples, certifying the quality of the latter.

2. Tensor Networks

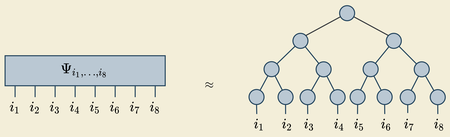
An N-rank tensor is a mathematical object with N indices, $T_{\alpha_1 \dots \alpha_N}$. We can use the **tensor diagram formalism** to represent them: each node corresponds to a tensor and each link to a running index.



As a result, a node without links is a scalar (rank 0), with one link is a vector (rank 1), with two links is a matrix, etc. **Tensor contractions** are represented by **closed links**. In formulae, it holds Einstein's notation:

$$\sum_{b,c} A_{a,b} B_{b,c} C_{c,d} \equiv A_{a,b} B_{b,c} C_{c,d} = O_{a,d}$$

A large tensor is **efficiently represented** by approximating it via factorization to a contracted product of lower-order tensors. Physically, this means treating a full many-body quantum system by **limiting its components' interactions**. Suitable architectures for both open and closed boundary conditions are **binary tree tensor networks** (BTTNs):



The dimension of closed links (**bond dimension** χ) can be tuned to find a threshold between complexity and correct full state representation. It allows to reduce to the most informative subspace of the whole Hilbert space. To approximate an N-rank tensor with local dimension d as a BTTN implies: $\mathcal{O}(d^N) \longrightarrow \mathcal{O}(Nd\chi^2)$ (exponential reduction in memory). Other benefits are the **exponential speed-up** of computations as well as theoretical insight and **interpretability**.

3. Tensor Network Machine Learning

TTNs turn out to be also a very natural way to parametrize ML models. We solved **binary classification** problems where each piece of data $\mathbf{x}$ is mapped to each label $\ell \in \{0, 1\}$ through a linear decision function

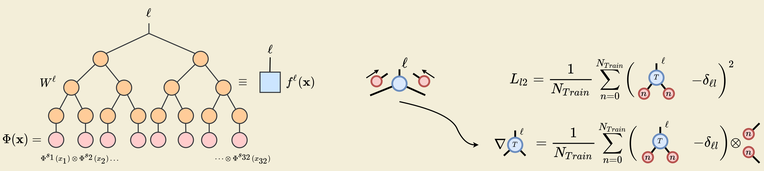
$$f^\ell(\mathbf{x}) = W^\ell \cdot \phi(\mathbf{x}) = \langle W^\ell | \phi(\mathbf{x}) \rangle$$

where W is the **optimized weight tensor** and $\phi(\cdot)$ is a multi-dimensional **feature map** that maps each piece of input data $\mathbf{x}$ to a **unentangled wave-function** of a N-body quantum system.
When input values are numbers in $[0, 1]$, the most common choice is the following **spin-map**:

$$\phi^{s_j}(x_j) = \left[\cos\left(\frac{\pi}{2} x_j\right), \sin\left(\frac{\pi}{2} x_j\right) \right].$$

Upon contraction between W^ℓ and $\phi(\mathbf{x})$ the result $f^\ell(\mathbf{x})$ is vector of **unitary norm**: its components are the **amplitude probabilities** for $\phi(\mathbf{x})$ to belong to each class ℓ. We assign the sample $\mathbf{x}$ to the class with the highest value of $|f^\ell(\mathbf{x})|$.

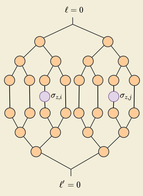
In our case, the optimized weights tensor will play the role of two wave functions, $|W^\ell\rangle \equiv |\psi^\ell_{\text{TTN}}\rangle$. We compute over them **observables** and entropies of interest.
For the **optimization** of the TTN, we exploit **gauge transformations** via QR decomposition. We define a cost function, e.g. the MSE loss L_{l2} and update the value of the tensor following the gradient.



4. Analytical tools

An index of pseudo-randomness is given by site-site correlations calculated over encoded inputs as follows:

$$C^{\ell\ell'}_{ij} = \frac{\langle \Psi^\ell_{\text{TTN}} | \sigma^z_i \sigma^z_j | \Psi^{\ell'}_{\text{TTN}} \rangle}{\sqrt{\langle \Psi^\ell_{\text{TTN}} | \Psi^\ell_{\text{TTN}} \rangle \langle \Psi^{\ell'}_{\text{TTN}} | \Psi^{\ell'}_{\text{TTN}} \rangle}}$$



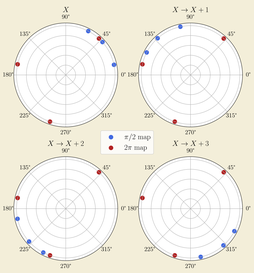
In truly random sequences no systematic correlations arise. For each layer we compute the **entanglement entropy** as an index of the **information distribution** among nodes. For the node X it holds:

$$\rho_X = \text{Tr}_{\bar{X}} |\Psi_{\text{TN}}\rangle \langle \Psi_{\text{TN}}| \ , \quad S(X) = -\text{Tr}(\rho_X \log \rho_X)$$

We trace out the complementary system by self-contraction. Finally, to better recognize periodic pattern, we propose a **spin encoding invariant under modulo operations**:

$$\phi^{s_j}(x_j) = [\cos(2\pi x_j), \sin(2\pi x_j)] \, .$$

An example for $\boldsymbol{x} = [0.1324, 0.4652, 0.6982]$:

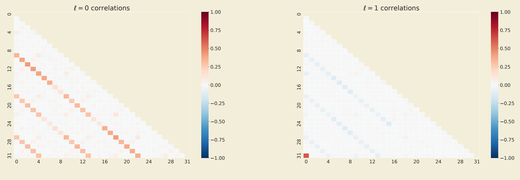


5. Computational resources

All our simulations have been carried out on a cluster of NVIDIA RTX GPU with 15 physical cores hosted by *CloudVeneto*. A standard bond dimension $\chi = 25$ has been used for most simulations. Over 160k train samples of size 32, the computational time per epoch is ~ 70 min.

6. Results examples

The model was tested against quantum random numbers for increasingly complex generated pseudo-random sequences. We ran some tests with **in-sequence repetition of $\mathcal{P}$ random values**, using also the new spin map proposal. Here an example of $\mathcal{P} = 9$ vs $\mathcal{P} = 31$: test accuracy is 100% and the correlation maps display interpretable results.

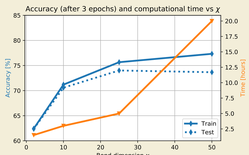


The network classifies $\ell = 1$ both by exploiting positive correlations (red squares) and the inverse information in $\ell = 0$. Swapping the labels the results are mirrored, meaning that the network is trained **symmetrically**.

After analysing different cases, we can state that TTNs show **very good results results in distinguishing quantum random sequences from periodic pseudo-random ones** up to periods way behind the input size ($\mathcal{P} \sim 2^{21}$). On the other hands, they **fail to discriminate QRNs from PRNs** generated via Marsenne-Twister (accuracy $\sim 50\%$). This further certificates the **quality** of such pseudo-random number generators.

7. Scaling with the bond dimension

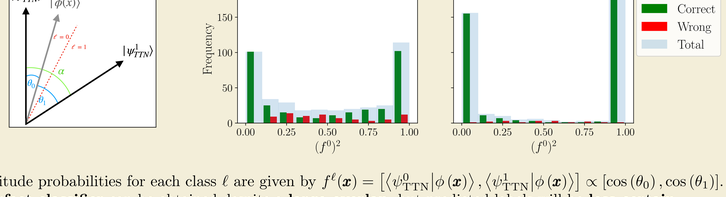
By **increasing the bond dimension the accuracy saturates**: for $\chi > 25$ the gain in accuracy does not justify the steep scaling of the computational efforts.



Hence, the network cannot grasp the **variability** of the data, independently of its representative power: this certifies the goodness of pseudo-random sequences (**impossible information compression**).

8. Overlap of tensor network states and discriminative power

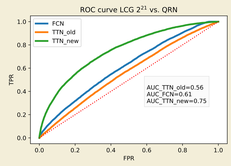
The contraction $\langle \Psi^0_{\text{TTN}} | \Psi^1_{\text{TTN}} \rangle$ defines the **overlap** between trained tensors over different labels, i.e. **how close** their **quantum representations** are. We expect a correlation between the **classification ability** of the TTN and this **overlap**: for $\langle \Psi^0_{\text{TTN}} | \Psi^1_{\text{TTN}} \rangle = \pm 1$ we should find a dumb classifier (labels 0 and 1 share the same features).



Amplitude probabilities for each class ℓ are given by $f^\ell(\boldsymbol{x}) = \left[\langle \psi^0_{\text{TTN}} | \phi(\boldsymbol{x}) \rangle, \langle \psi^1_{\text{TTN}} | \phi(\boldsymbol{x}) \rangle \right] \propto [\cos(\theta_0), \cos(\theta_1)]$.
A **perfect classifier** can be obtained despite a **large overlap**, but predicted labels will be **less certain** ($f^\ell(\boldsymbol{x}) \neq [100\%, 0\%]$ and $f^\ell(\boldsymbol{x}) \neq [0\%, 100\%]$). This can be seen in the plots to the right, enlarging χ for a study case.

9. Fully connected network

We compare TTN performances with a fully connected network composed of three layers, 32 input nodes and binary output. We perform a random grid search over **500 models** and architectures for each task.



A **TTN with our map** proposal achieves the **best performance** for long-range correlations detection. However, by changing the seed for pseudo-random number generation the pre-trained TTN accuracy drops to 50%, i.e. the network lacks of **generalization**.
We generate sequences of 32 PRNs via Linear Conguential Generator with $\mathcal{P} = 2^{20}$. Interestingly, the **TTN has noticeably better performances**: this is highly significant, since the TN never sees the same sequence twice. **Pre-processing** the data by spectral analysis via **discrete cosine transform** equates TTN and FCN scores.

10. Conclusion and outlook

We reviewed the **functioning and potential of TTNs**, exploring growing complexity levels and proposing both problems and solutions. We observed how TNs **can detect** correlations **beyond the dimension of input sequences**, with better performances with respect to NNs - even if the lack of generalization suggests that TNs might be learning actual numbers instead of relations.

Future work could introduce **higher dimensional feature maps**, test the impact of data pre-processing, try more **advanced classic machine learning** models (e.g. CNNs) and study the **predictability** of trained TTNs.

References and acknowledgements

[1] Edwin M. Stoudenmire and David J. Schwab: *Supervised learning with tensor networks*, Advances in Neural Information Processing Systems, 29 (2016)
[2] Pietro Silvi et al. *The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems*, SciPost Phys. Lect. Notes, page 8 (2019)
[3] Marco Trenti: *Tensor Networks Machine Learning algorithms for event recognition in High Energy Physics*. Master thesis, Università degli Studi di Padova (2020)

A special thanks to Prof. Simone Montangero and Dr. Giuseppe Magnifico (University of Padova) for their teachings and support, and to Marco Trenti (University of Ulm) for his code and insights.