# Machine learning Fisher Information Metric from bitstrings

First Last[1,*]
[1] *USC affiliation*
(Dated: August 2, 2022)

We present a machine-learning based method "Bitstring-ChiFc" which, given a dataset corresponding to a family of distributions of bitstrings parameterized by a manifold, can produce a rough approximation for the corresponding Fisher Information Metric. We observe that for multiple toy models there are often enough simple patterns in the data that this approach achieves satisfactory approximation even for dataset sizes small compared to the number of possible bitstrings.

## I. PRESENTATION

### A. Introduction

Let's talk about phase transitions. Consider the Hamiltonian on a $2 \times L$ lattice given by the following equations.

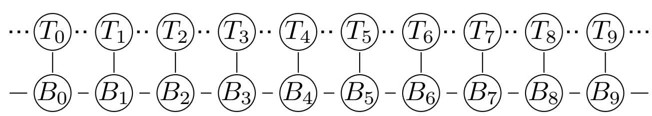
$$H(s, K, U) = (1 - s)H_0 + sH_1, \tag{1}$$

$$H_0 = -\sum_{i=0}^{L-1} (X_{T_i} + X_{B_i}), \tag{2}$$

$$H_1 = \sum_{i=0}^{L-1} \Big( K Z_{T_i} Z_{T_{i+1}} - K Z_{T_i} Z_{B_i} - K Z_{B_i} Z_{B_{i+1}} - K Z_{T_i} + \frac{U}{2} Z_{B_i} \Big). \tag{3}$$

Here qubits $T_L$ and $B_L$ are identified with $T_0$ and $B_0$ respectively.

It is called "Frustrated Ladder Hamiltonian" and is schematically represented by the following diagram:



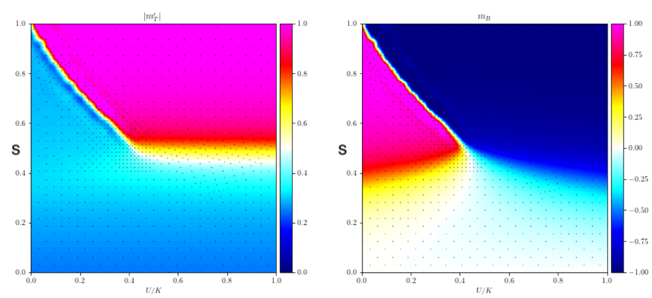

On this diagram of $L = 10$ Frustrated Ladder Hamiltonian the solid lines represent ferromagnetic couplings and dotted lines — antiferromagnetic couplings. For a fixed $L$ the Frustrated Ladder Hamiltonian depends on 3 parameters, $s, K, U$. We set $K = 1$ and consider the values $s \in [0, 1]$, $U \in [0, 1]$.

How would you find phase transitions of that Hamiltonian? For that particular Hamiltonian people already know a couple of order parameters given by the following equations.

$$|m'_T| = \left| \sum_i Z_{T_i} (-1)^i \right| \tag{4}$$

$$m_B = \sum_i Z_{B_i} \tag{5}$$

These are called "staggered magnetization of the top row" and "magnetization of the bottom row" respectively. For $L = 10$ we can compute these for various values of the parameters of the Hamiltonian and produce the following diagrams:




## II. TODO

1. Complete presentation section above.
2. Write down the proofs for the fidelity susceptibility claims below.
3. Describe the models and practical results for them.

## III. INTRODUCTION

TODO:

Such a family can arise, e.g., from measurements of a low-temperature Gibbs ensemble of Hamiltonians parametrized by a parameter $\lambda$.

## IV. CLASSICAL FIDELITY SUSCEPTIBILITY

Classical fidelity between 2 probability distributions $p$ and $q$ of bitstrings $z$ is defined as

$$F_c(p, q) = \sum_z \sqrt{p(z)q(z)}. \tag{6}$$

We are interested in the fidelity between bitstring distributions at different $s$ (e.g. $s = s_1$ and $s = s_2$), which we will denote as $F_c(s_1, s_2)$.

* email

[NE: This is an example of a commonly used in-line comment which is separated by color. I could say something like: "This sentence is awkward" or "Needs citation" or very meta "Please use enquote for real quotes and not literal quotes."]

Fidelity susceptibility is defined as the term $\chi_{F_c}(s)$ in the Taylor expansion

$$F_c(s, s+\delta s) = 1 - \frac{\delta s^2}{2}\chi_{F_c}(s) + O(\delta s^3). \tag{7}$$

For such Taylor expansion to exist it is sufficient that the probabilities have a Taylor expansion up to $O(\delta s^3)$. More generally, probability distribution can depend on a point $\lambda$ on a manifold $\Lambda$, in which case the Tailor expansion (7) would become

$$F_c(\lambda, \lambda+\delta\lambda) = 1 - \frac{\delta\lambda_j\delta\lambda_k}{2}\chi^{jk}_{F_c}(\lambda) + O(\delta\lambda^3). \tag{8}$$

### A. Classical and quantum fidelity susceptibility

Fact 1: For pure states $\mathbb{E}\chi_{F_c}(s) = \frac{1}{2}\chi_F(s)$ where the expectation is over all orthogonal bases to perform the measurement in.

TODO:proof

Fact 2: For computational basis measurement of a non-degenerate ground state of a real-valued Hamiltonian $H$, then $\chi_{F_c}(s) = \chi_F(s)$ almost everywhere.

TODO:proof

## V. PROBLEM SETUP

- In this work we consider a family of distributions of bitstrings $\{\mathcal{D}_\lambda\}_{\lambda\in\Lambda}$, each of length $n$.
- We are given a finite sample $\mathcal{D}_{\text{train}}$ of size $N$ of pairs $(\lambda, z)$ s.t. $P(z|\lambda) = P_{\mathcal{D}_\lambda}(z)$.
- We are also given (possibly implicitly via coordinate description of $\Lambda$) a naive metric $g^0$ on $\Lambda$.
- We are asked to estimate the Fisher information metric $g$ on $\Lambda$ corresponding to distributions $\mathcal{D}_\lambda$.
- Locations with high $g/g^0$ are then considered to be conjectured locations of possible phase transitions.

We focus on the task of identifying phase transitions in that family. Rigorously speaking, phase transitions are only defined in the limit $n \to \infty$, while we are dealing with finite size systems. A solution to that is to look at Fisher information metric: high distances according to Fisher information metric for points close according to naive metric likely correspond to phase transitions.

## VI. BITSTRING-CHIFC METHOD

In this work we propose the following method:

- Collect training dataset $\mathcal{D}_{\chi_{F_c}\text{-train}}$ of the form $(\lambda_0, \delta\lambda, z, y)$, where $z$ is sampled from $p(\bullet, \lambda = \lambda_z)$, $p_+ = p(\lambda_z = \lambda_0 + \delta\lambda/2 | \lambda_z = \lambda_0 \pm \delta\lambda/2)$, and $\mathbb{E}(y|\lambda_0, \delta\lambda, z) = p_+$. In practice $y \in \{0, 1\}$. Do it in the following way:
  - Consider $\mathcal{D}_{\text{train}}$ consisting of pairs $(z, \lambda)$.
  - Sample pairs $(z_{i+}, \lambda_{i+})$, $(z_{i-}, \lambda_{i-})$ from $\mathcal{D}_{\text{train}}$.
  - Compute $\lambda_i = (\lambda_{i+} + \lambda_{i-})/2$, $\delta\lambda_i = \lambda_{i+} - \lambda_{i-}$.
  - Add tuples $(\lambda_i, \delta\lambda_i, z_{i+}, 1)$ and $(\lambda_i, \delta\lambda_i, z_{i-}, 0)$ to the dataset $\mathcal{D}_{\chi_{F_c}\text{-train}}$.
- Train a model $M$, which given $(\lambda_0, \delta\lambda, z)$ will predict $l = M(\lambda_0, \delta\lambda, z)$ s.t. $p_+ = (1 + e^{-l\cdot\delta\lambda})^{-1}$. Do this by minimizing cross-entropy loss on the dataset $\mathcal{D}_{\chi_{F_c}\text{-train}}$.
- Estimate

$$\chi^{jk}_{F_c}(\lambda) = \text{smoothen}\left(\lambda_1 \mapsto \text{mean}_{(z,\lambda_1)\in\mathcal{D}_{\text{train}}} M(\lambda_1, 0, z)^j M(\lambda_1, 0, z)^k\right) \tag{9}$$

TODO: expand the explanation.

$$\chi_{F_c}(\lambda) = \lim_{\delta\lambda\to 0}\frac{2}{\delta\lambda^2}\left(1 - \mathbb{E}_{z\sim Q(\bullet)}\frac{\sqrt{P(z|\lambda-\delta\lambda/2)P(z|\lambda+\delta\lambda/2)}}{Q(z)}\right)$$
$$\simeq \lim_{\delta\lambda\to 0}\mathbb{E}_Q\frac{2}{\delta\lambda^2}\frac{2\sinh^2(l\delta\lambda/4)}{\cosh(l\delta\lambda/2)} \simeq \frac{1}{4}\mathbb{E}_{z|\lambda}M(\lambda, 0, z)^2.$$

TODO: models

TODO: experiments