Title: Learning a phase diagram from dynamics

Date: Apr 23, 2018 02:00 PM

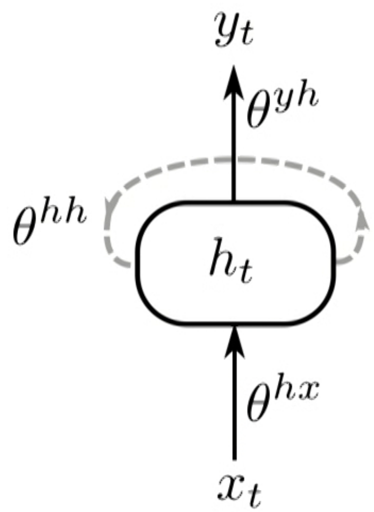
URL: http://pirsa.org/18040132

Abstract: <p>Time series data contains useful information on the phase of a system. Here we propose the use of recurrent neural networks (LSTM) to learn and extract such information in order to classify phases and locate phase boundaries. We demonstrate this on a many-body localized model, and attempt to interpret the learned behavior by looking at individual LSTM cells. We also discuss the validity of the learned model and investigate its limits.</p>

Learning a phase diagram from dynamics

Evert van Nieuwenburg, Eyal Bairey & Gil Refael

Caltech



https://arxiv.org/abs/1712.00450

https://physicsml.github.io

Machine Learning Journal Club – #MLJCETH

Machine Learning

Giving computers the ability to “learn” from data,
without being explicitly programmed.

(progressively improve performance on a specific task)

Machine Learning

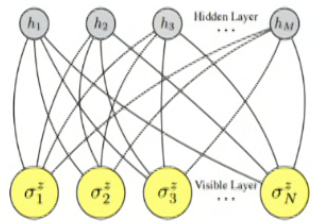
Giving computers the ability to “learn” from data,
without being explicitly programmed.

(progressively improve performance on a specific task)

(Not the same as AI!)

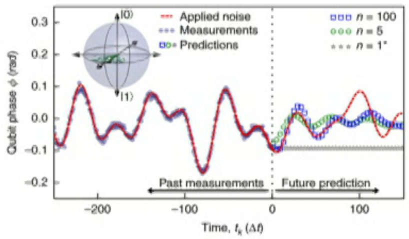
Context

States and tomography



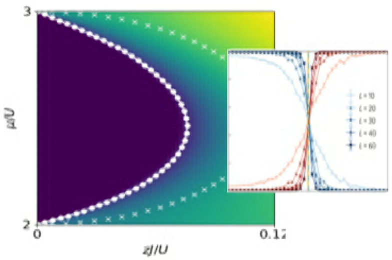
Carleo & Troyer, Science 10.1126

Qubits and control



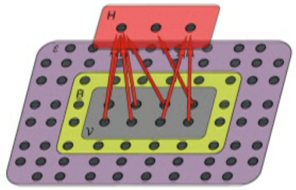
Gupta & Biercuk, arXiv 1712.01291

Phases and transitions



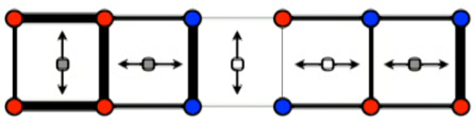
Carrasquilla & Melko, nphys4035
EvN, Liu, Huber, nphys4037

Networks and RG



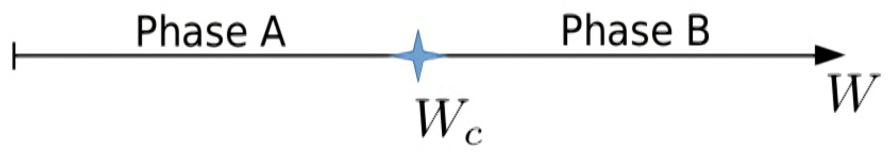
Koch-Janusz & Ringel, nphys 10.1038

ML-aided numerics

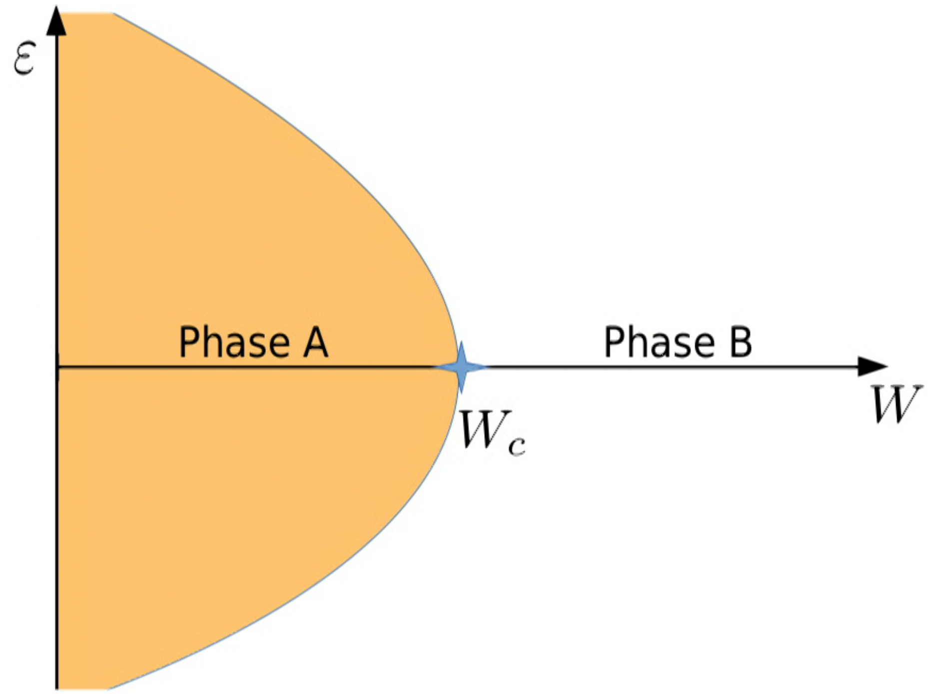


Wang, PRE 96, 051301

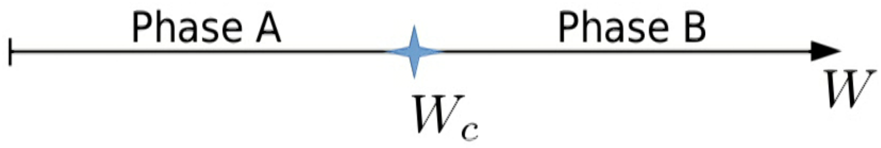
Phases and transitions



Phases and transitions



Phases and transitions



Do we have partial knowledge?

yes → Blanking

no → PCA, ..., “Confusion”

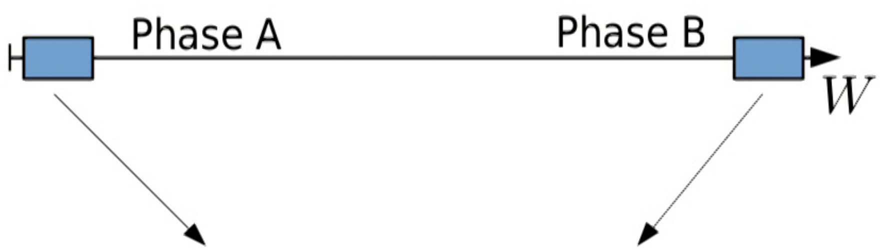
EvN, Y. Liu, S.D. Huber, nphys 4037

The answer still depends on the *data* itself (which measurements)

Beach, Golubeva & Melko PRB **97**, 045207 (2018)

Blanking

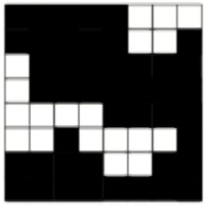
We (assume we) know the physics in parts of the phase space

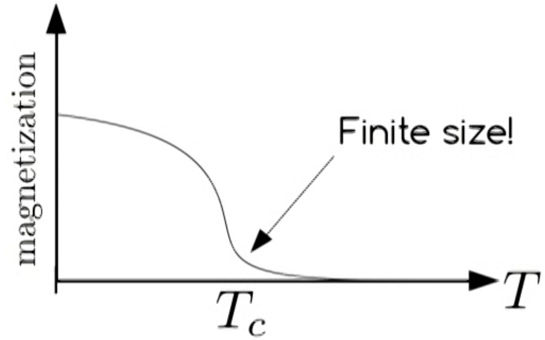


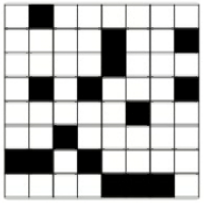
Use *data* from these regions to machine learn (read: fit) a model

Blanking on the Ising model

$$H = -\sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z$$



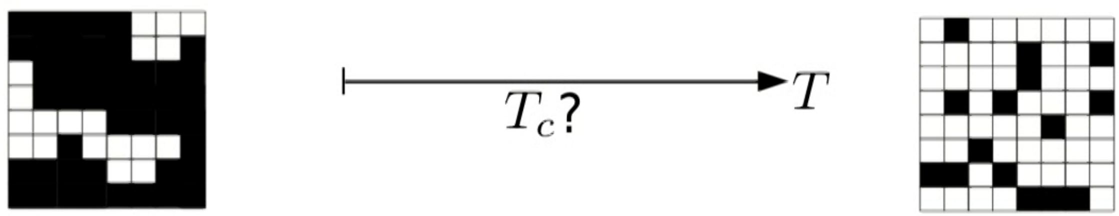




Monte Carlo generated ‘snapshots’ of the spin configuration

Blanking on the Ising model

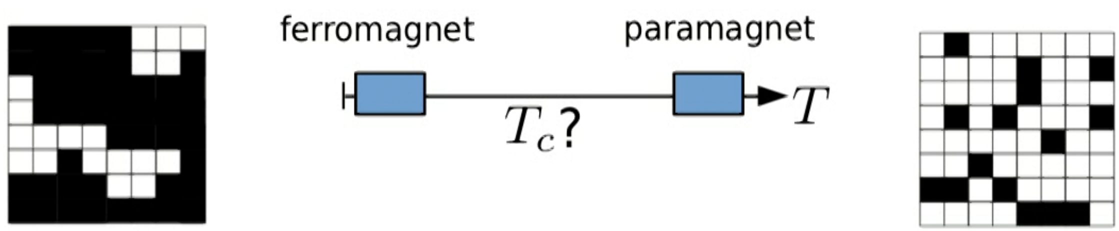
$$H = -\sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z$$



Monte Carlo generated ‘snapshots’ of the spin configuration

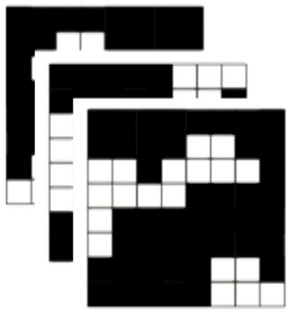
Blanking on the Ising model

$$H = -\sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z$$

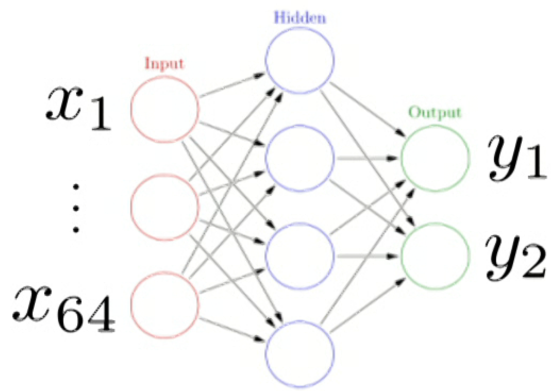


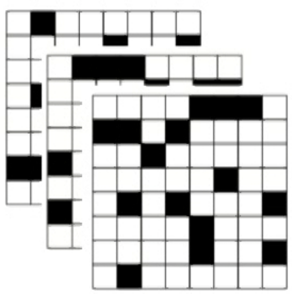
Monte Carlo generated 'snapshots' of the spin configuration

Blanking on the Ising model



$$\tilde{\mathbf{y}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

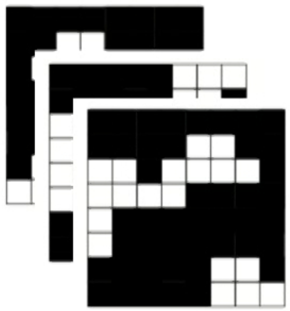




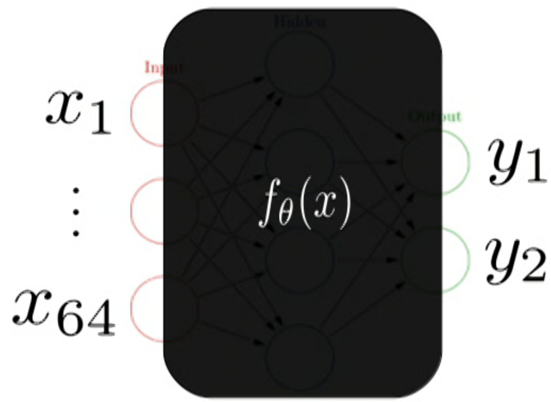
$$\tilde{\mathbf{y}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

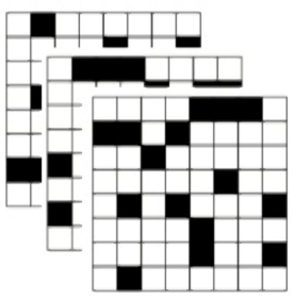
$$C_\theta(\mathbf{x}, \tilde{\mathbf{y}}) \sim \sum_{\text{examples}} (\tilde{\mathbf{y}} - f(\mathbf{x}))^2$$

Blanking on the Ising model



$$\tilde{\mathbf{y}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$





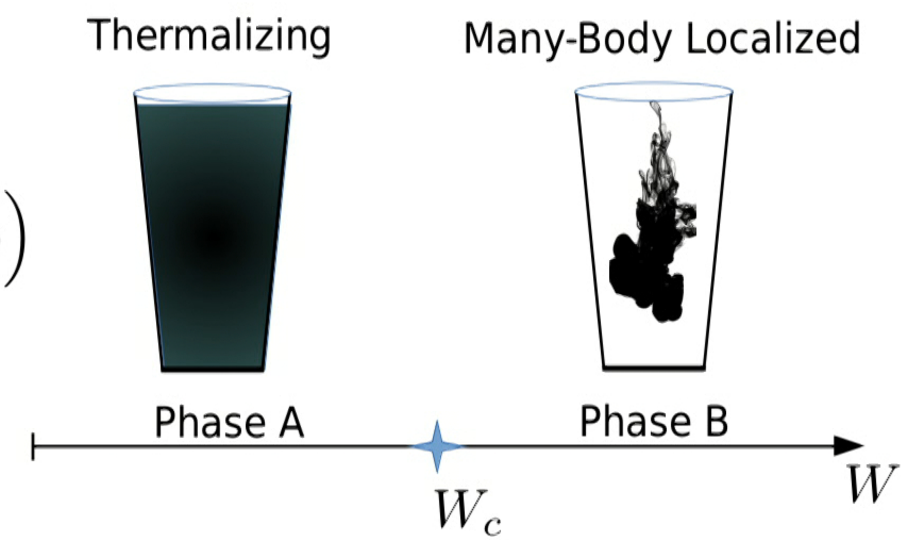
$$\tilde{\mathbf{y}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$C_\theta(\mathbf{x}, \tilde{\mathbf{y}}) \sim \sum_{\text{examples}} (\tilde{\mathbf{y}} - f(\mathbf{x}))^2$$

Many-Body Localization

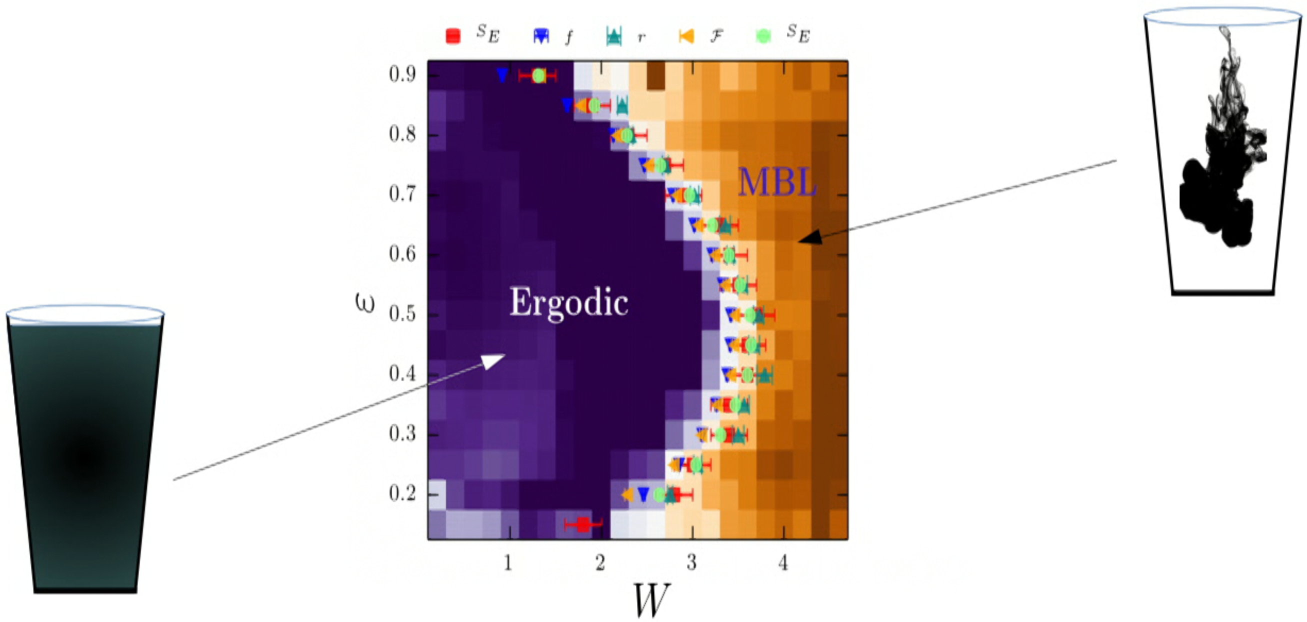
$$H = \sum \boldsymbol{\sigma}_i \cdot \boldsymbol{\sigma}_{i+1} + h_i \sigma_i^z \qquad h_i \in [-W, W]$$

$$\left(t \to \infty\right)$$



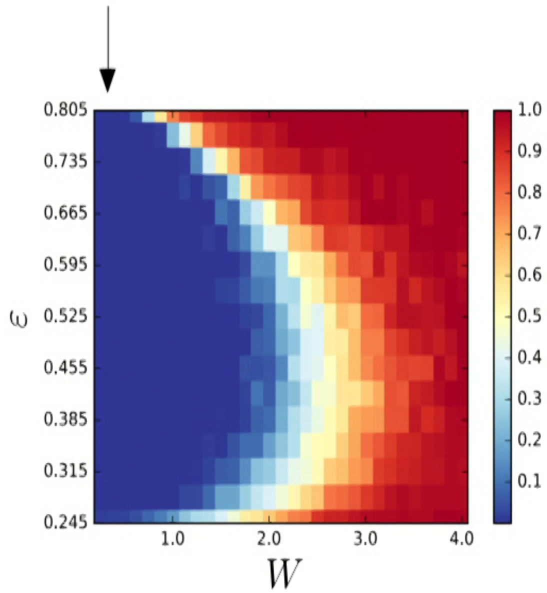
Many-Body Localization

$$H = \sum \boldsymbol{\sigma}_i \cdot \boldsymbol{\sigma}_{i+1} + h_i \sigma_i^z \qquad h_i \in [-W, W]$$



Luitz et. al., PRB **91**, 081103 (2015)

Machine learning an MBL transition



Schindler et. al., PRB **95**, 245134 (2017)

Trained on entanglement spectra

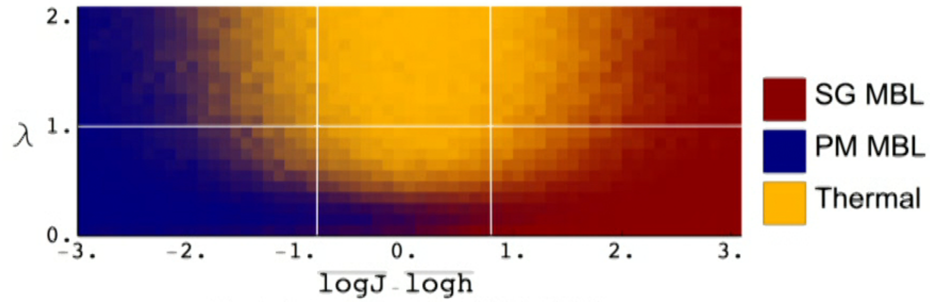
1) $|\psi\rangle \rightarrow \rho = |\psi\rangle\langle\psi|$

2) $\rho \rightarrow \rho_A = \mathrm{Tr}_B\, \rho$

3) $\{\lambda_i\} = \mathrm{diag}\, \rho_A$

Machine learning an MBL transition

Numerical evidence for an MBL→MBL transition



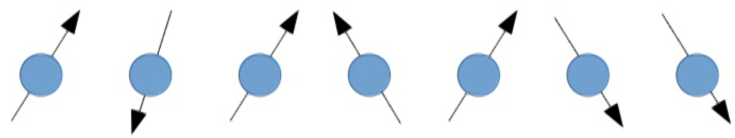
Vanderley et. al., arXiv:1711.00020

Also trained on entanglement spectra

Dynamics

Experimentally accessible data?

$$H = \sum \boldsymbol{\sigma}_i \cdot \boldsymbol{\sigma}_{i+1} + h_i \sigma_i^z$$



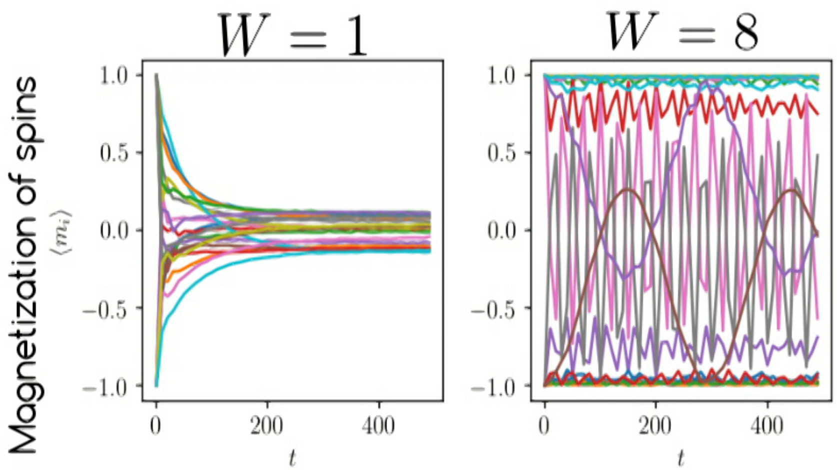
$i = 1$ $\qquad$ $i = 7$

$$|\psi\rangle(t) = e^{-iHt}|\psi\rangle(0)$$

Can we determine the phase from a “movie”?

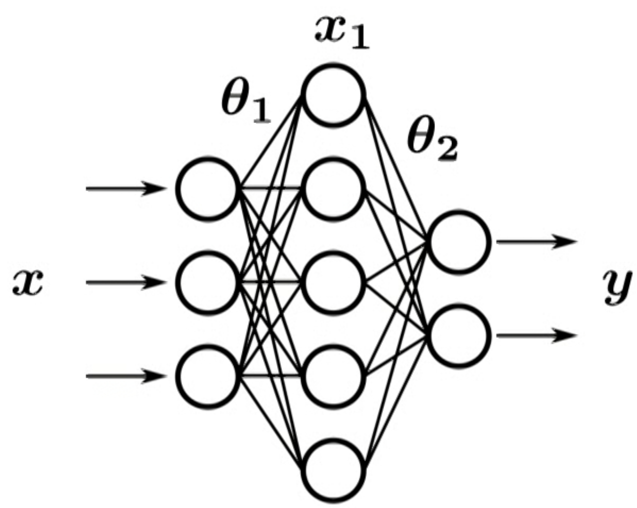
Dynamics

Experimentally accessible data?



Can we determine the phase from a “movie”?

(Feedforward) Neural Networks



Non-linear activation function

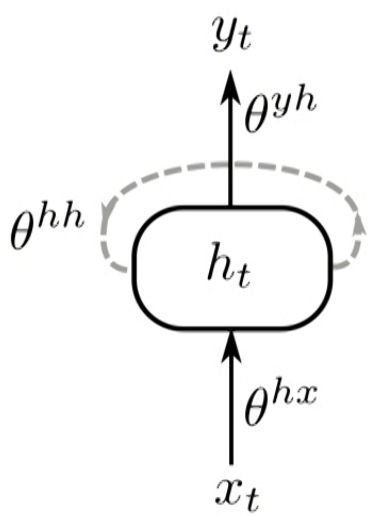
$$\boldsymbol{x_1} = \tanh\left(\boldsymbol{\theta_1} \cdot \boldsymbol{x}\right)$$

$$\boldsymbol{y} = \tanh\left(\boldsymbol{\theta_2} \cdot \boldsymbol{x_1}\right)$$

$$C_\theta(\mathbf{x}, \tilde{\mathbf{y}}) \sim \sum_{\text{examples}} (\tilde{\mathbf{y}} - \mathbf{y})^2$$

$$\boldsymbol{\theta} \to \boldsymbol{\theta} + \alpha \, \nabla_{\boldsymbol{\theta}} C$$

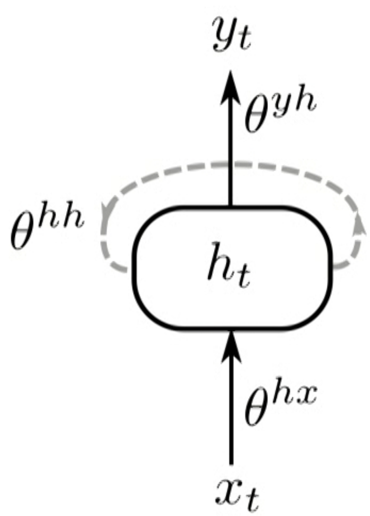
Recurrent Neural Networks



$$\boldsymbol{h}_t = \tanh\left(\boldsymbol{\theta}^{hh}\boldsymbol{h}_{t-1} + \boldsymbol{\theta}^{hx}\boldsymbol{x}_t\right)$$

$$\boldsymbol{y}_t = \boldsymbol{\theta}^{yh}\boldsymbol{h}_t$$

Recurrent Neural Networks



$$\boldsymbol{h}_t = \tanh\left(\boldsymbol{\theta}^{hh}\boldsymbol{h}_{t-1} + \boldsymbol{\theta}^{hx}\boldsymbol{x}_t\right)$$
$$\boldsymbol{y}_t = \boldsymbol{\theta}^{yh}\boldsymbol{h}_t$$

Actually, if

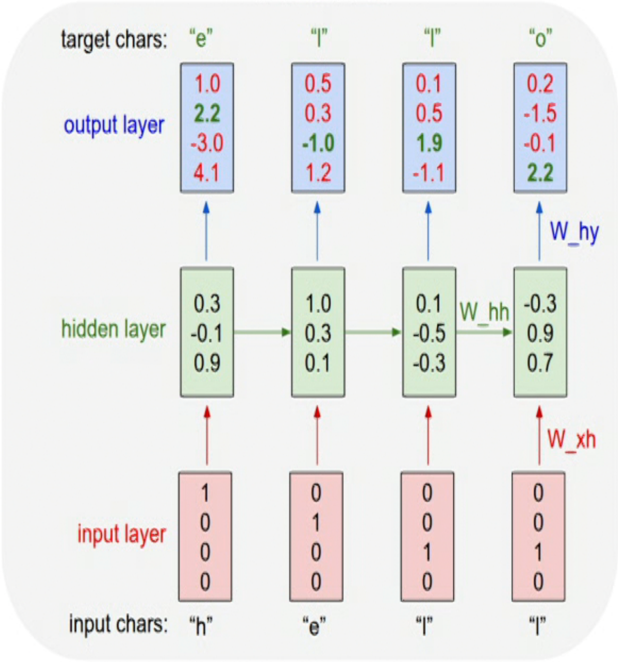
$$\boldsymbol{h}_t = \tanh(\boldsymbol{\theta}(\boldsymbol{h}_{t-1} \otimes \boldsymbol{x}_t))$$

this can be trained to learn an iMPS

(Christian Mendl @ Dresden)

Character level RNN

Hello



$$\boldsymbol{h}_t = \tanh\left(\boldsymbol{\theta}^{hh}\boldsymbol{h}_{t-1} + \boldsymbol{\theta}^{hx}\boldsymbol{x}_t\right)$$
$$\boldsymbol{y}_t = \boldsymbol{\theta}^{yh}\boldsymbol{h}_t$$

The Unreasonable Effectiveness of Recurrent Neural Networks
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Character level RNN

Proof. Omitted. □

Lemma 0.1. *Let $\mathcal{C}$ be a set of the construction.*

Let $\mathcal{C}$ be a gerber covering. Let $\mathcal{F}$ be a quasi-coherent sheaves of $\mathcal{O}$-modules. We have to show that

$$\mathcal{O}_{\mathcal{O}_X} = \mathcal{O}_X(\mathcal{L})$$

.

Proof. This is an algebraic space with the composition of sheaves $\mathcal{F}$ on $X_{\acute{e}tale}$ we have

$$\mathcal{O}_X(\mathcal{F}) = \{morph_1 \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{F}$ of $\mathcal{O}$-modules. □

Lemma 0.2. *This is an integer $\mathcal{Z}$ is injective.*

Proof. See Spaces, Lemma ??. □

Lemma 0.3. *Let S be a scheme. Let X be a scheme and X is an affine open covering. Let $\mathcal{U} \subset \mathcal{X}$ be a canonical and locally of finite type. Let X be a scheme. Let X be a scheme which is equal to the formal complex.*

The following to the construction of the lemma follows.

Let X be a scheme. Let X be a scheme covering. Let

$$b : X \to Y' \to Y \to Y \to Y' \times_X Y \to X.$$

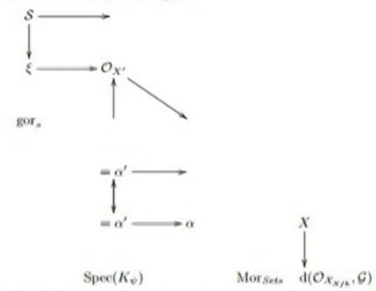
be a morphism of algebraic spaces over S and Y.

Proof. Let X be a nonzero scheme of X. Let X be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

1. $\mathcal{F}$ is an algebraic space over S.
2. If X is an affine open covering.

Consider a common structure on X and X the functor $\mathcal{O}_X(U)$ which is locally of finite type. □

This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram



is a limit. Then $\mathcal{G}$ is a finite type and assume S is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type f_*. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_{X'}$ is a sheaf of rings.

□

Proof. We have see that $X = \mathrm{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of X is an open neighbourhood of U. □

Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.
A *reduced above* we conclude that U is an open covering of $\mathcal{C}$. The functor $\mathcal{F}$ is a "field

$$\mathcal{O}_{X,x} \longrightarrow \mathcal{F}_{\bar{x}} \ \text{-}1(\mathcal{O}_{X_{\acute{e}tale}}) \longrightarrow \mathcal{O}_{X_\ell}^{-1}\mathcal{O}_{X_\lambda}(\mathcal{O}_{X_\eta}^{\overline{v}})$$

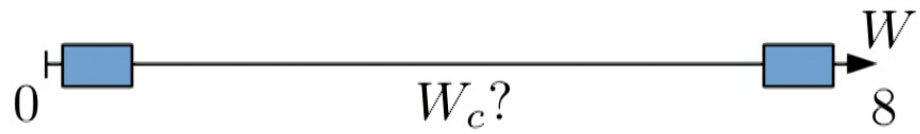
is an isomorphism of covering of $\mathcal{O}_{X_i}$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that X is an isomorphism.
The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over S.
If $\mathcal{F}$ is a scheme theoretic image points. □

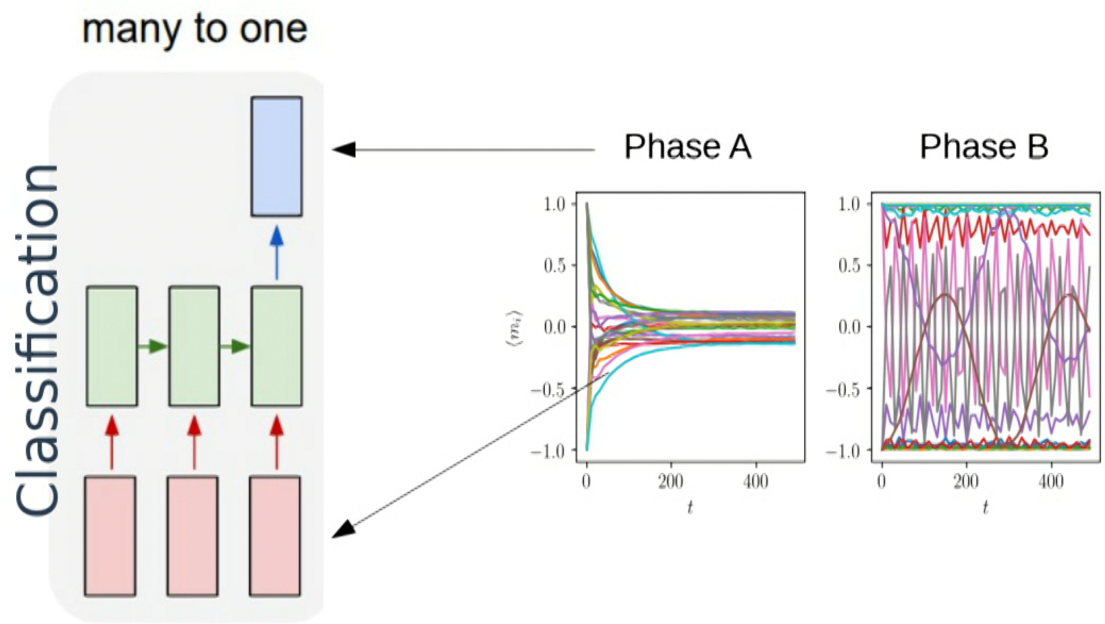
If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X_\lambda}$ is a closed immersion, see Lemma ??. This is a sequence of $\mathcal{F}$ is a similar morphism.

The Unreasonable Effectiveness of Recurrent Neural Networks
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

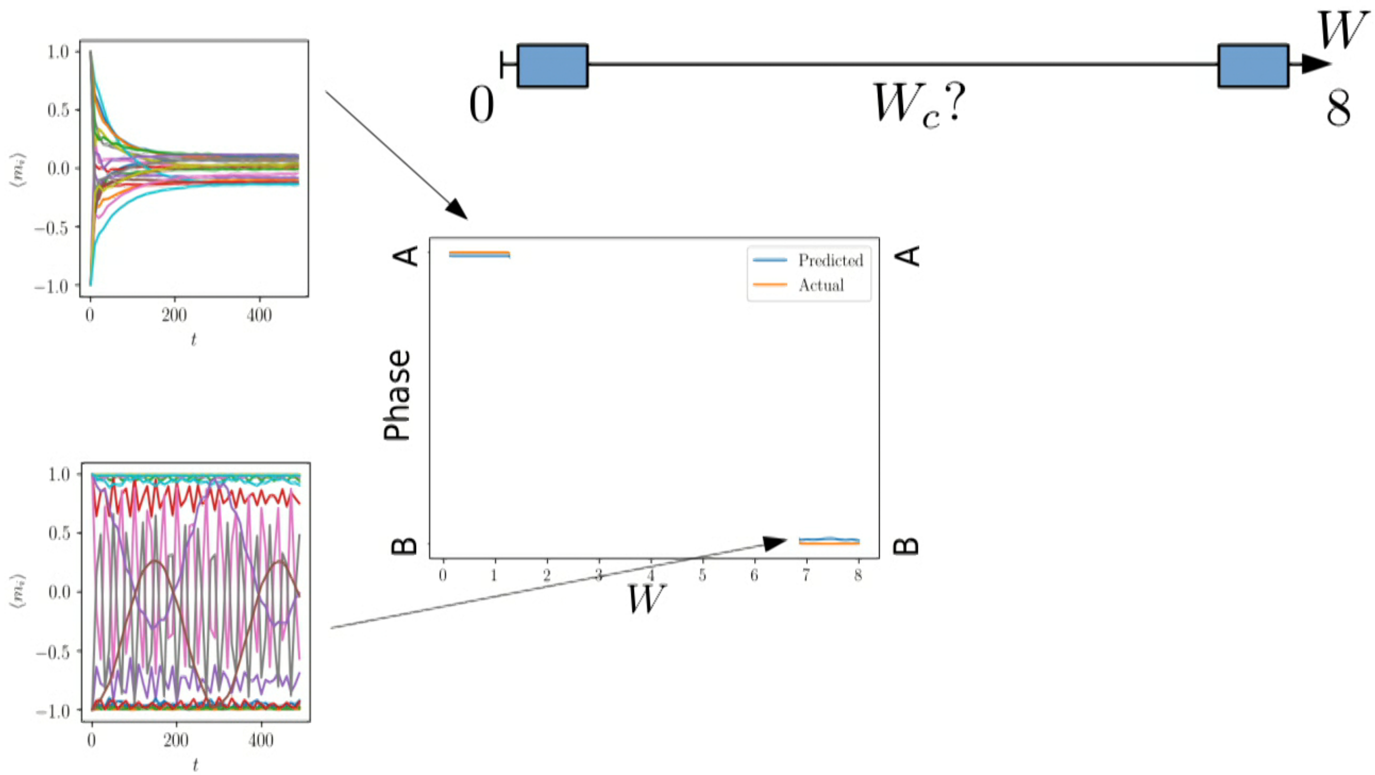
quantumfrontiers.com – Machine Learning the arXiv

Machine learning an MBL transition

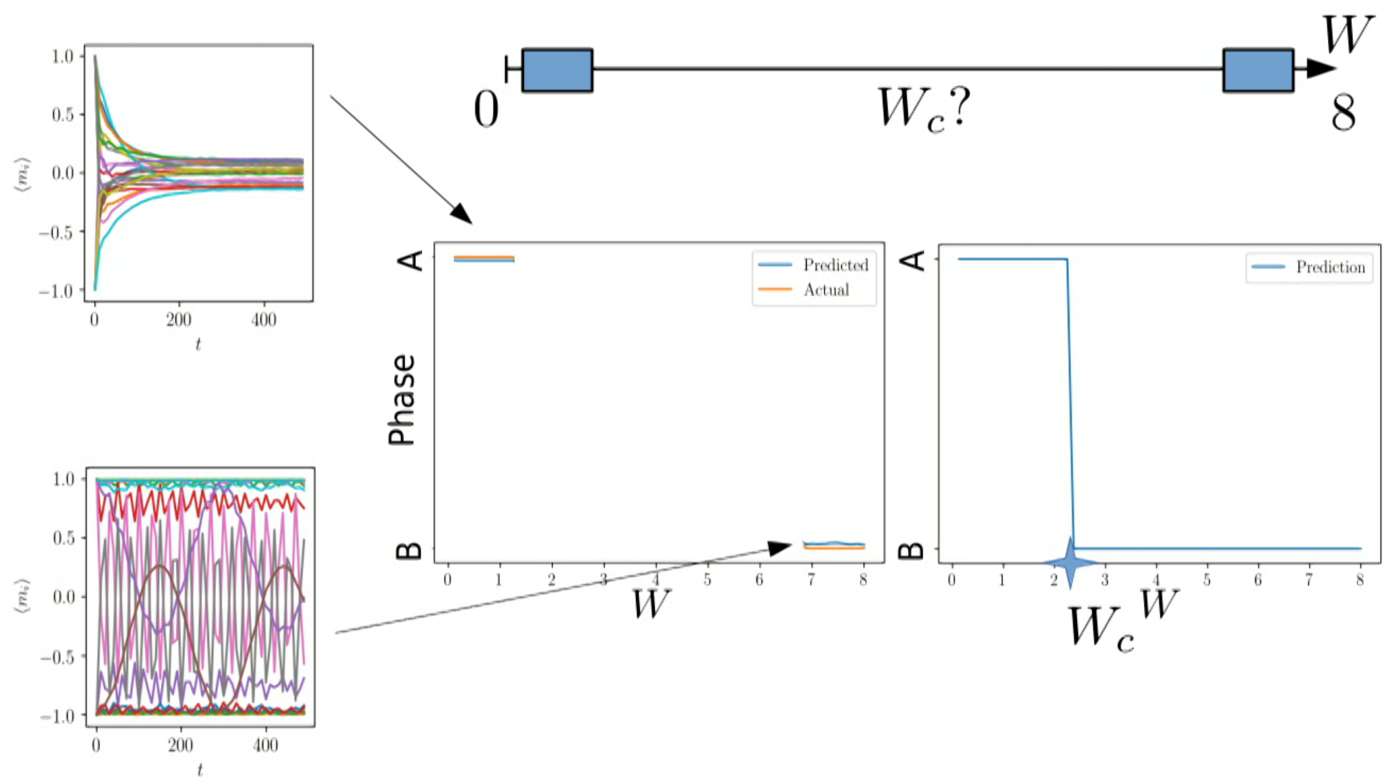




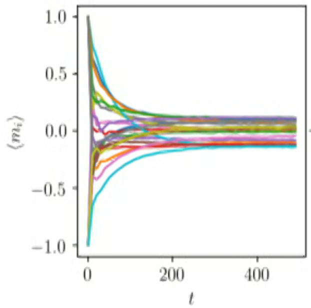
Machine learning an MBL transition



Machine learning an MBL transition

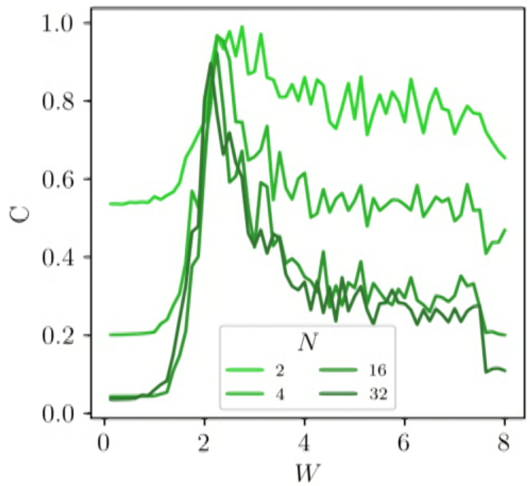


Machine learning an MBL transition

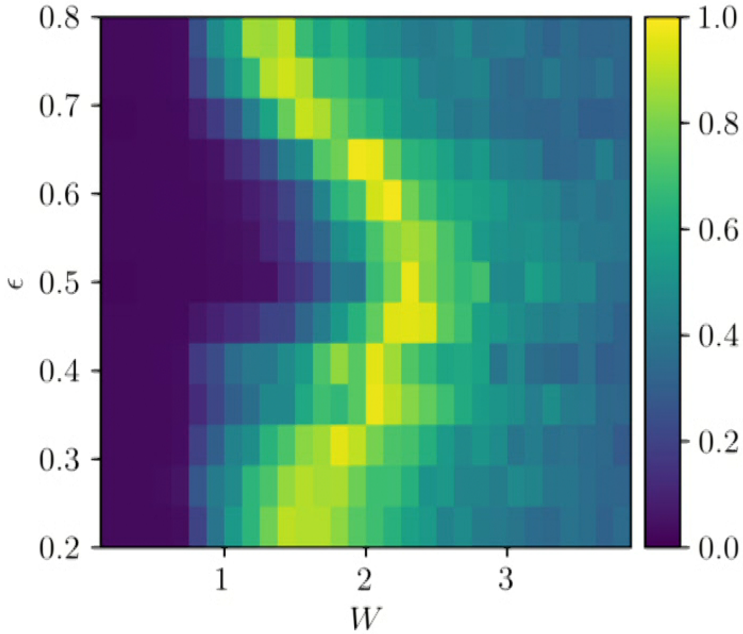


$\mathcal{N}$ → $\begin{pmatrix} p_A \\ p_B \end{pmatrix}$ → $C = -\ln \frac{p_A}{p_A + p_B}$

$p_A + p_B = 1$



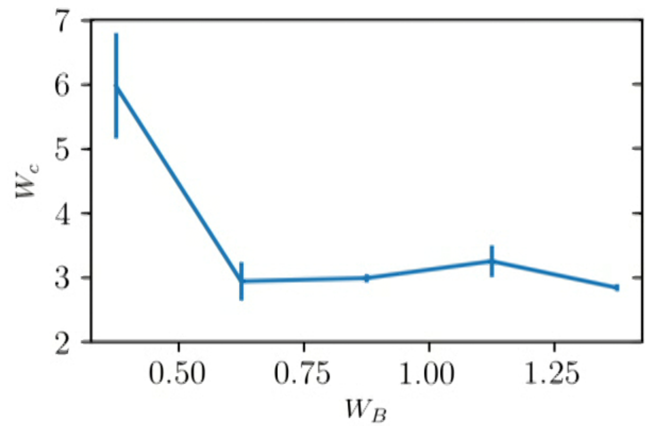
Machine learning an MBL transition



EvN, Eyal Bairey, Gil Refael, arXiv:1712.00450

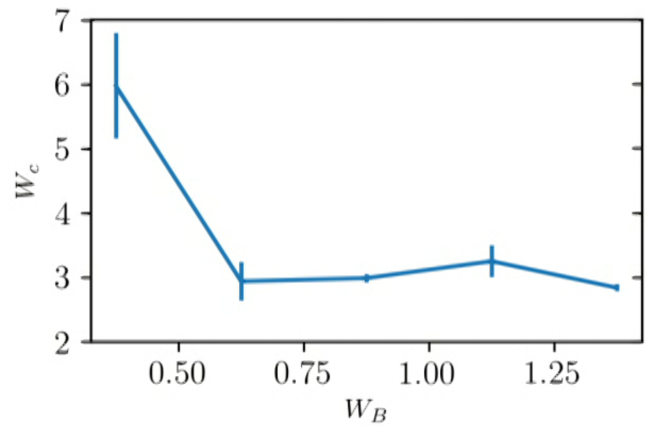
Machine learning an MBL transition

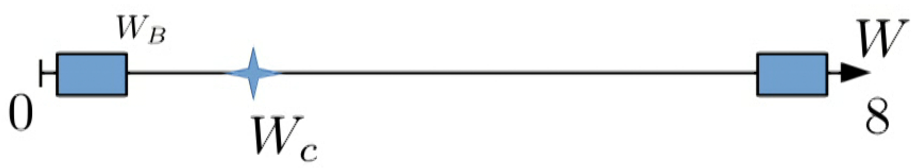
Check dependence on included data!



Machine learning an MBL transition

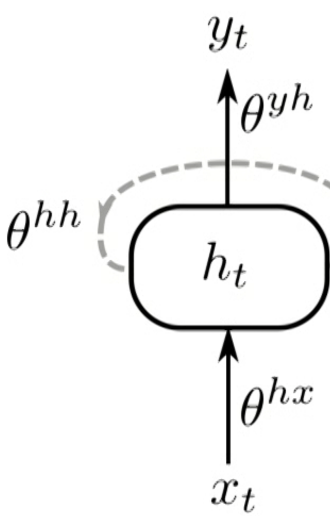
Check dependence on included data!

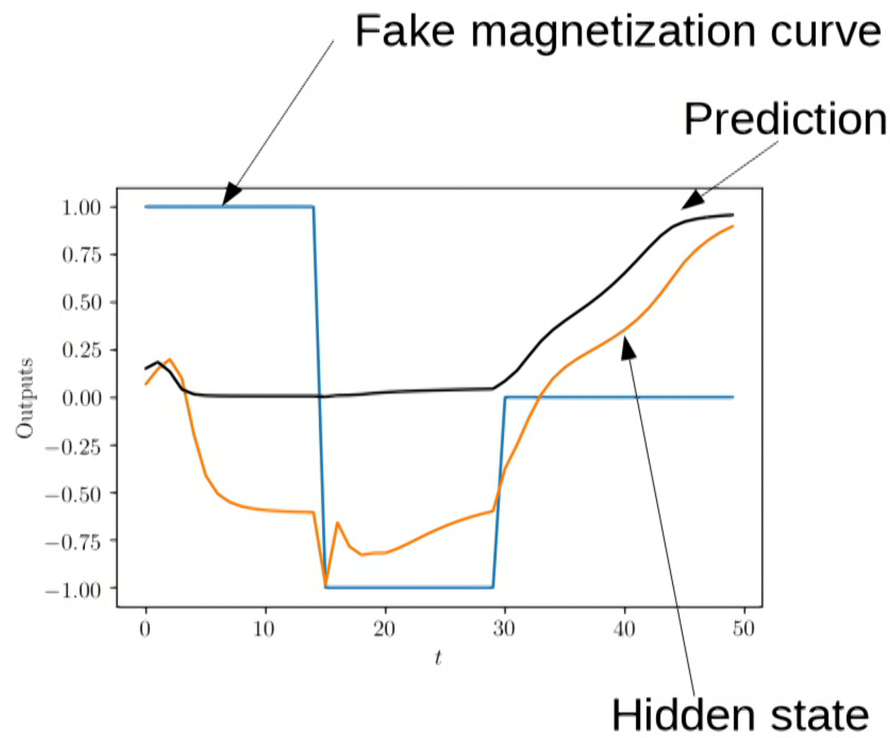




Learning single spin behavior

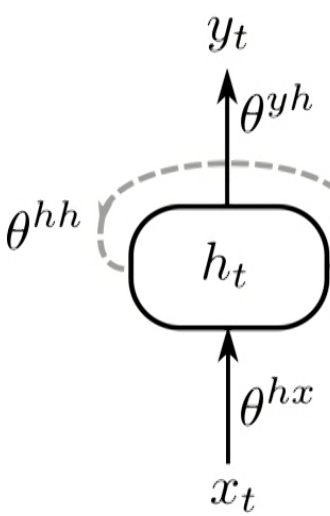
Single ‘unit’ (1D hidden state)

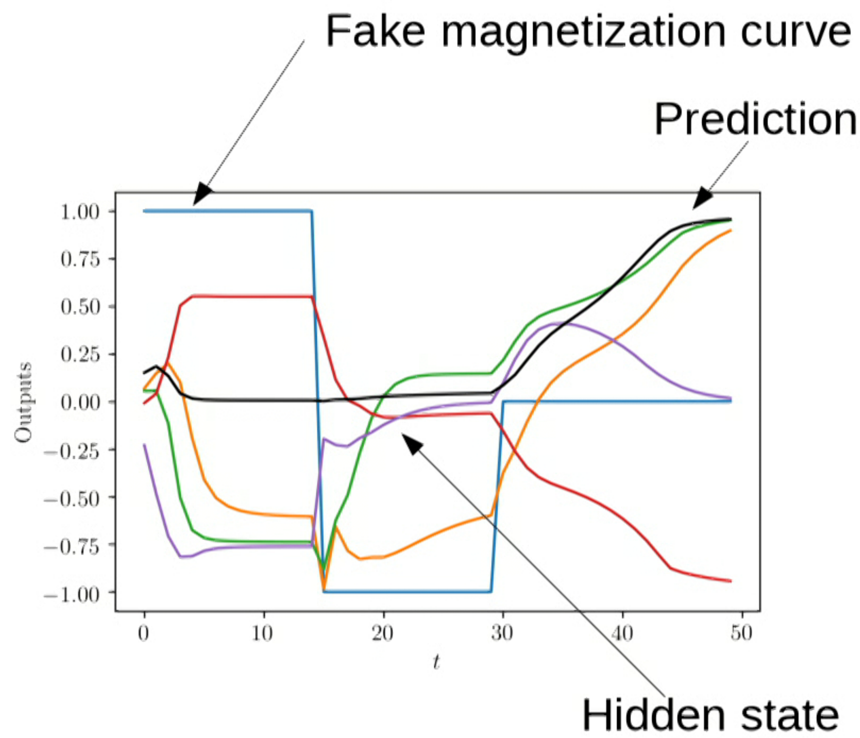




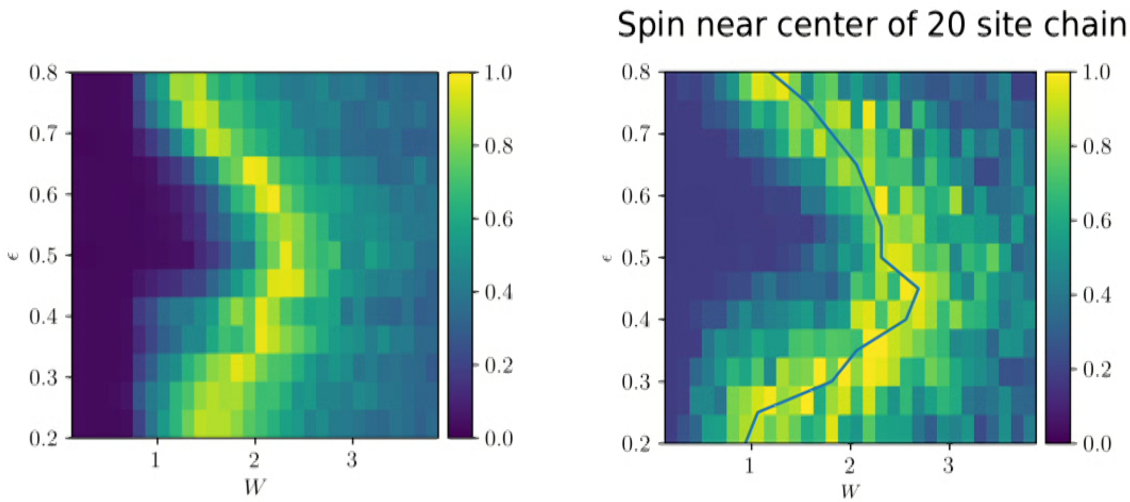
Learning single spin behavior

Four 'units' (4D hidden state)

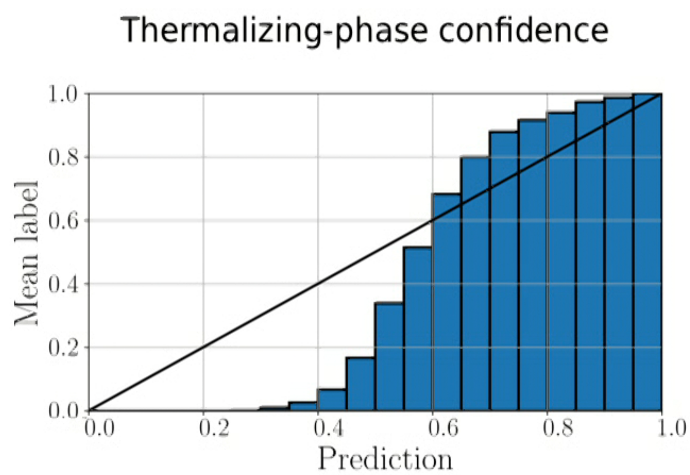




Learning single spin behavior



Network Calibration
(how well does confidence correspond to accuracy?)



Future?

RNNs with ‘attention’

can we overcome the fixed input size problem?

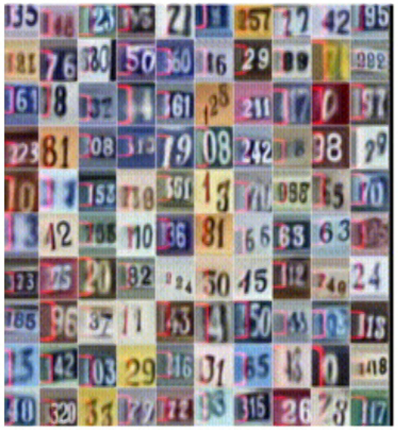
RNNs as wavefunction Ansatz

Is there a class of wavefunctions efficiently encoded?

RNNs for quantum control

Feedback circuits and/or forward prediction of qubits

Niu et al. arXiv:1803.01857, Foesel et al. arXiv:1802.05267



Future?

RNNs with ‘attention’

can we overcome the fixed input size problem?

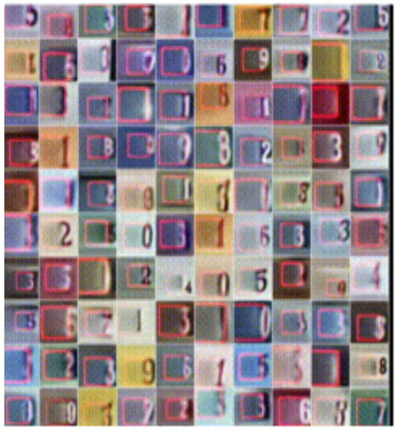
RNNs as wavefunction Ansatz

Is there a class of wavefunctions efficiently encoded?

RNNs for quantum control

Feedback circuits and/or forward prediction of qubits

Niu et al. arXiv:1803.01857, Foesel et al. arXiv:1802.05267



Future?

RNNs with ‘attention’

can we overcome the fixed input size problem?

RNNs as wavefunction Ansatz

Is there a class of wavefunctions efficiently encoded?

RNNs for quantum control

Feedback circuits and/or forward prediction of qubits

Niu et al. arXiv:1803.01857, Foesel et al. arXiv:1802.05267