anvil

Normalising flows for lattice field theories

Joe Marsh Rossney, Michael Wilson

11th June 2020

Contents

1. Preliminaries

1.1 Notation

The notation used to identify lattice positions and directions will be:

- A subscript n, m labels lattice sites, the total number of which is V. A sum over lattice sites is written as $\sum_n \equiv \sum_{n=1}^{V}$.
- Greek letters $\mu, \nu \ldots$ label lattice dimensions, with $\hat{\mu}$ representing a unit vector (in units of a) in the dimension μ. A sum over lattice dimensions is written as $\sum_\mu \equiv \sum_{\mu=1}^{d}$ where d is always 2.
- We will not adopt any special notation to denote a vector quantity. However, a vector (or multiplet) σ, z, ... is distinguished from its components $(\sigma^1, \sigma^2, \ldots)$, z^i, ... by the absence of a superscript number or Latin letter i, j.
- We will generally parameterise the fields in terms of a set of coordinates labelled ϕ_n^i, where i runs from 1 to N_ϕ. Normally, ϕ refers to a $V \times N_\phi$-dimensional state vector which we refer to as a 'configuration'. However, occasionally (in particular when discussing coupling transformations) we will use ϕ to refer to a single scalar value.
- A subscript a, b labels field configurations in an ensemble, the total number of which is denoted N_{ens}.
- $\langle F \rangle$ denotes the ensemble average of F, which is defined by

$$\langle F \rangle \equiv \frac{1}{N_{\mathsf{ens}}} \sum_{a=1}^{N_{\mathsf{ens}}} F_a$$

1.2 Models

We consider discretised versions of two-dimensional Euclidean field theories. The continuum versions are defined as follows.

Scalar theory with quartic interaction

For a scalar field ϕ, a minimal $\mathbb{Z}_2$ symmetric action including interactions is given by the so-called "phi-four" action:

$$S = \int d^2x \left[\frac{1}{2} \big(\partial_\mu \phi(x)\big) \big(\partial_\mu \phi(x)\big) + \frac{1}{2} m^2 \phi^2 + \frac{1}{4!} \lambda \phi^4 \right] \tag{1}$$

If m^2 is negative, the $\mathbb{Z}_2$ symmetry is spontaneously broken and there are two distinct vacuum solutions.

$O(N)$ non-linear σ models

These are a class of two-dimensional theories of real, N-dimensional fields $\sigma = (\sigma^1, \sigma^2, \ldots, \sigma^N)$, defined by the action

$$S = \frac{1}{2g} \int d^2x \, \partial_\mu \sigma(x) \cdot \partial_\mu \sigma(x) \tag{2}$$

together with the constraint

$$\sigma(x) \cdot \sigma(x) = 1 \tag{3}$$

These models are analogous to classical spin models with pure nearest-neighbour interactions and no external fields, which spontaneously break their $O(N)$ symmetry at the critical temperature.

In particular, the $O(3)$ model is interest since it can sustain topological excitations – i.e. ones which cannot be created nor destroyed by local fluctuations.

CP^{N-1} non-linear σ models

These are a class of two-dimensional theories of complex N-component fields $z = (z^1, z^2, \ldots, z^N)$, defined by the action

$$S = \frac{1}{g} \int d^2x (D_\mu z)^* \cdot (D_\mu z) \tag{4}$$

together with the constraint

$$z^*(x) \cdot z(x) = 1 \tag{5}$$

$D_\mu = \partial_\mu + iA_\mu$ is the usual covariant derivative due to a local $U(1)$ symmetry.

The gauge field is

$$A_\mu = \frac{i}{2}\big(z^* \cdot \partial_\mu z - (\partial_\mu z^*) \cdot z\big) \tag{6}$$

briefly discuss motivation.

1.3 Lattice

We will work exclusively with a two-dimensional tetragonal lattice of dimensions (aL, aL), where a is the lattice spacing.

For convenience, we will universally define $a \equiv 1$, so that the lattice length L and volume V count, respectively, the number of lattice sites along a single dimension and in the entire lattice.

The discrete versions of derivatives, for $a = 1$, are:

$$\begin{aligned}
\delta_\mu f_n &= f_{n+\hat{\mu}} - f_n && \text{(forward difference operator)} \\
\delta^*_\mu f_n &= f_n - f_{n-\hat{\mu}} && \text{(backward difference operator)} \\
\bar{\delta}_\mu f_n &= \frac{1}{2}\big(f_{n+\hat{\mu}} - f_{n-\hat{\mu}}\big) && \text{(central difference operator)}
\end{aligned}$$

One possibility for the discrete Laplacian, which is Hermitian, is given by

$$\delta^2 f_n = \sum_\mu \delta^*_\mu \delta_\mu f_n = \sum_\mu \big(f_{n+\hat{\mu}} + f_{n-\hat{\mu}} - 2f_n\big) \tag{7}$$

As in the continuum, plane waves are eigenvectors of the Laplacian. However, the eigenvalues are different:

$$\delta^2 e^{ik\cdot x} = -\hat{k}^2 e^{ik\cdot x} \qquad\qquad \hat{k}_\mu = 2\sin\frac{k_\mu}{2} \tag{8}$$

For a function $F(x)$, the Fourier transform is

$$\tilde{F}(k) = \int \frac{d^d x}{(2\pi)^d} e^{ik\cdot x} F(x) \tag{9}$$

On a lattice, this becomes

$$\tilde{F}(k) = \frac{1}{V}\sum_x e^{ik\cdot x} F(x) \tag{10}$$

1.4 Sampling

Once a batch of field configurations have been generated, the Metropolis-Hastings algorithm constructs a Markov chain whose stationary distribution is the target distribution.

Given an (un-normalised) target density $p(\phi)$ and a set of proposals $\{\phi\}$ drawn from $\tilde{p}_f(\phi)$, the Metropolis-Hastings probability for accepting a transition is

$$P(\phi \to \phi') \sim \frac{\tilde{p}_f(\phi)}{\tilde{p}_f(\phi')} \frac{p(\phi')}{p(\phi)} \tag{11}$$

where $\sim x$ is short-hand for $\min(1, x)$.

In practice, we work with the logarithms of un-normalised probability densities, and the acceptance condition is evaluated as

$$\begin{aligned} \log R(\phi) &= \log \tilde{p}_f(\phi) - \log p(\phi) \\ P(\phi \to \phi') &\sim \exp\big(\log R(\phi) - \log R(\phi')\big) \end{aligned} \tag{12}$$

1.5 Observables

We consider the *connected* two point correlation function $G(x)$ as the basic building block for computing observables.

We will be dealing with lattice field theories which possess translational and rotational invariance along the lattice dimensions.

Translation invariance means that observables should be independent of lattice position n in limit of infinite ensemble size. We may as well, therefore, choose to take the volume average of observables which have a dependence on n, such as the two point correlator: $G(x) = \frac{1}{V} \sum_n G(n, x)$.

In the same vein, rotational invariance means that, where an observable depends on the lattice sites on one particular dimension, we may as well average over all lattice dimensions. In particular, we do not designate a particular dimension as the time dimension, so any observable defined for a time slice will be averaged over each dimension.

Uncertainties are calculated using the boostrap method. See appendix for details.

2. Lattice actions

2.1 ϕ^4 action

The lattice periodicity ensures that total derivatives in the Lagrangian vanish, letting us write the kinetic term as a Laplacian.

$$S = \sum_n \left[\frac{1}{2}\phi_n(-\delta^2 + m^2)\phi_n + \frac{1}{4!}\lambda\phi^4 \right] \tag{13}$$

By expanding the Laplacian defined by eq. 7 and taking advantage of the volume sum to replace the negative shift with a second positive shift, we obtain

$$S = \sum_n \left[(2 + \frac{1}{2}m^2)\phi_n^2 - \sum_\mu \phi_{n+\hat{\mu}}\,\phi_n + \frac{1}{4!}\lambda\phi_n^4 \right] \tag{14}$$

Currently, in the code, the factor of $\frac{1}{4!}$ is simply absorbed into the definition of λ.

2.2 $O(N)$ action

The action can be discretised using the forward derivative, $\partial_\mu\sigma(x) \to \delta_\mu\sigma_n = \sigma_{n+\hat{\mu}} - \sigma_n$

$$\begin{aligned} S &= \frac{1}{2g}\sum_n\sum_\mu \delta_\mu\sigma_n \cdot \delta_\mu\sigma_n \\ &= -\frac{1}{g}\sum_n\sum_\mu \left(\sigma_{n+\hat{\mu}} \cdot \sigma_n - 1\right) \end{aligned} \tag{15}$$

This has the same form as the Hamiltonian for a lattice of classical N-dimensional spin vectors with pure nearest-neighbour ferromagnetic interactions.

$$H_{\text{spin}} = -\sum_n\sum_\mu \sigma_{n+\hat{\mu}} \cdot \sigma_n \tag{16}$$

To relate them, we relabel the coupling $\frac{1}{g} \equiv \beta$, with β labelling, as is convention, the inverse temperature of the spin system. Then,

$$S = \beta H_{\text{spin}} + 2\beta V \tag{17}$$

The global shift of $2\beta V$ does not affect the dynamics of the system.

2.3 $O(2)$ action

Since the N-dimensional σ fields take values on the unit $(N-1)$-sphere, it can be convenient to use generalised spherical coordinates to parameterise them in terms of $N-1$ angles.

The $O(2)$ fields live on a circle and a parameterised by a single 'polar' angle θ.

$$\sigma_{O(2)} = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \qquad \theta \in [0, 2\pi) \tag{18}$$

$$\begin{aligned} S_{O(2)} &= -\beta \sum_n \sum_\mu \Big(\cos\theta_{n+\hat{\mu}} \cos\theta_n + \sin\theta_{n+\hat{\mu}} \sin\theta_n - 1 \Big) \\ &= -\beta \sum_n \sum_\mu \Big(\cos(\theta_{n+\hat{\mu}} - \theta_n) - 1 \Big) \end{aligned} \tag{19}$$

2.4 $O(3)$ action

For $O(3)$ we have the standard parameterisation of a unit sphere, in terms of polar angle θ and 'azimuthal' angle ϕ.

$$\sigma_{O(3)} = \begin{pmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{pmatrix} \qquad \theta \in [0, \pi],\ \phi \in [0, 2\pi) \tag{20}$$

$$S_{O(3)} = -\beta \sum_n \sum_\mu \Big(\cos\theta_{n+\hat{\mu}} \cos\theta_n + \sin\theta_{n+\hat{\mu}} \sin\theta_n \cos(\phi_{n+\hat{\mu}} - \phi_n) - 1 \Big) \tag{21}$$

2.5 CP^{N-1}: implicit gauge field

There are several possible CP^{N-1} lattice actions. The simplest choice is

$$S = -\frac{N}{g} \sum_n \sum_\mu \Big(\left| z^*_{n+\hat{\mu}} \cdot z_n \right|^2 - 1 \Big) \tag{22}$$

which clearly bears much resemblance to the $O(N)$ action.

In fact, for $N = 2$ this action is equivalent to the $O(3)$ lattice action given by eq. 15, up to a factor of $1/2$ in the coupling. We will sometimes denote the coupling by $\frac{N}{g} \equiv \beta$ to

make contact with classical spin systems.

2.6 CP^1: implicit gauge field

A simple parameterisation for CP^1 is

$$z = \begin{pmatrix} \cos\frac{\theta}{2} e^{i\phi/2} \\ \sin\frac{\theta}{2} e^{-i\phi/2} \end{pmatrix} \tag{23}$$

with $\theta \in [0, \pi]$ and $\phi \in [0, 2\pi]$, yields a simple form for the action in terms of the polar angles:

$$S = -\frac{2}{g} \sum_n \sum_\mu \left(\cos\theta_n \cos\theta_{n+\mu} + \sin\theta_n \sin\theta_{n+\mu} \cos(\phi_n - \phi_{n+\mu}) - 1 \right) \tag{24}$$

which is the same as the $O(3)$ action in eq. 21 with $\beta = \frac{2}{g}$.

3. Normalising flows

3.1 The general picture

A normalising flow is a map $f : \phi \mapsto x$ between random scalar variables $\phi \sim \tilde{p}_f$ and $x \sim r$, where $\tilde{p}_f(\phi)$ is the density of the actual distribution resulting from the normalising flow, $p(\phi)$ is the target density and $r(x)$ is a simple base distribution. This map must be differentiable, invertible and have a differentiable inverse.

For a given map f and a sample $\{\phi \,|\, f(\phi) = x \sim r\}$, a probability density can be obtained using the change of variables formula:

$$\tilde{p}_f(\phi) = r\big(f(\phi)\big) \left| \det \frac{\partial f(\phi)}{\partial \phi} \right| \tag{25}$$

The idea is to embed neural networks within the flow, which can be trained to minimise some loss function that quantifies how well $\tilde{p}_f$ approximates the target distribution.

In more general terms, the aim is to find a representation of the target distribution which factorises into a product of simple distributions, and uses neural networks to learn an invertible map between the two.

3.2 Loss function

The loss function used in `anvil` is the shifted Kullbach-Leibler (KL) divergence between the distribution resulting from the flow model and the target distribution defined by the density

$$p(\phi) = \left| \det J^{\mathsf{param}}(\phi) \right| \frac{e^{-S(\phi)}}{\mathcal{Z}} \tag{26}$$

which includes a possible Jacobian determinant from a non-trivial parameterisation of the fields.

The loss function will generally look like

$$\frac{1}{N_{\mathsf{batch}}} \sum_{a=1}^{N_{\mathsf{batch}}} \left\{ \log r\big(f(\phi_a)\big) + \log \left| \det \frac{\partial f(\phi_a)}{\partial \phi_a} \right| - \log \left| \det J^{\mathsf{param}}(\phi_a) \right| + S(\phi_a) \right\} \tag{27}$$

For more details see appendix 9.2.

3.3 Coupling layers

`anvil` uses the coupling layer construction to implement normalising flow models. This approach results in flows which are easy to invert and for which the density $\tilde{p}_f(\phi)$ can be calculated very efficiently.

A general coupling layer has the following structure:

$$g_l \; : \begin{cases} \phi^R \mapsto x^R = \phi^R \\ \phi^B \mapsto x^B = C_l\big(\phi^B; \{\mathbf{N}(\phi^R)\}\big) \end{cases} \tag{28}$$

The non-trivial coupling transformation C_l, which acts on just one of the partitions, has parameters $\{\mathbf{N}(\phi^R)\}$ which are themselves bijective functions of the other partition, and which are generally dense neural networks – i.e. connecting every element in ϕ^R to every element in ϕ^B.

Provided C_l is invertible, this construction allows the forward and inverse map to be evaluated from a single pass of the neural networks.

The Jacobian for a coupling transformation is (in block notation)

$$J_l = \begin{pmatrix} \mathbb{I} & \mathbf{0} \\ \frac{\partial C_l}{\partial \phi^R} & \operatorname{diag} \frac{\partial C_l}{\partial \phi^B} \end{pmatrix} \tag{29}$$

and the determinant is simply the product of terms on the leading diagonal

$$\big|\det J_l\big| = \prod_{\phi_n^i \in \phi^B} \left| \frac{\partial C_l}{\partial \phi_n^i} \right| \tag{30}$$

Through this construction the ϕ at different lattice sites are coupled, at no additional expense for the computation of the Jacobian determinant![1]

This can also be written in terms of the inverse transformation

$$\big|\det J_l\big| = \prod_{x_n^i \in x^B} \left| \frac{\partial C_l^{-1}}{\partial x_n^i} \right|^{-1} \tag{31}$$

Coupling layers can be combined through function composition: $f = g_1 \circ g_2 \circ \ldots \circ g_{N_{\text{lay}}}$, alternating between transformations of the R and B partitions. Since $\det AB =$

[1] I.e. the Jacobian determinant is the same as it would be if we replaced $\{\mathbf{N}(\phi^R)\}$ with parameters that had no dependence on ϕ^R.

$\det A \det B$, the logarithm of the Jacobian determinant, which is all that's needed for the loss function, is just

$$\log \left| \det \frac{\partial f(\phi)}{\partial \phi} \right| = \sum_{l=1}^{N_{\text{lay}}} \log \left| \det J_l \right| \tag{32}$$

From hereon, we will drop the R and B subscripts and assume we are always talking about a transforming a vector ϕ that is one or other of the partitions. Furthermore, we will drop the explicit dependence of the neural networks on the passive partition.

3.4 Convex combinations

There are situations where composition multiple transformations does not sufficiently increase the expressivity of the flow. In particular, some coupling transformations form a group under function composition, i.e.

$$g_i \circ g_j = g_k \qquad\qquad g_i, g_j, g_k \in G \tag{33}$$

which means the composition of any number of transformations could just as well be achieved by a single transformation.[2]

One way of generating more complex probability densities is by forming a mixture of simpler ones

$$\tilde{p}_f(\phi) = \sum_{i=1}^{N_{\text{mix}}} \rho_i \, \tilde{p}_{g_i}(\phi) \qquad\qquad \sum_{i=1}^{N_{\text{mix}}} \rho_i = 1 \tag{34}$$

with the mixture weights being appropriately normalised so that $\tilde{p}_f(\phi)$ is a valid probability measure.

In geometry, this is known as a convex combination.

3.5 Normalising flows as diffeomorphisms

It has been stated that the map f must be differentiable, invertible and have a differentiable inverse. These are exactly the conditions a map must satisfy to be a diffeomorphism.

Diffeomorphisms exist only between manifolds of the same topological class. Hence, there

[2]However, even if this is the case (as with affine transformations), there is still benefit to composing because it can improve the ability of the gradient descent to find a minimum.

is no "one size fits all" for normalising flow models; a model must be selected whose coupling layers preserve the topological properties of the target manifold. In addition, the base distribution needs to be defined on the target manifold.

4. Flows on $\mathbb{R}^D$

4.1 Real NVP

Real NVP is a flow model which combines affine coupling transformations via function composition.

An affine coupling layer takes the form

$$C_l\big(\phi; s, t\big) = e^{s_l} \odot \phi + t_l \tag{35}$$

where s_l and t_l are bijective functions that are learned by neural networks, and $\odot$ is an element-wise product.

For density estimation we use the inverse transformation,

$$C_l^{-1}\big(x; s, t\big) = \big(x - t_l\big) \odot e^{-s_l} \tag{36}$$

which shows good numerical stability if we pass s_l through a ReLU.

For the change of variables formula, we need the determinant of the Jacobian of C_l. The lower-right block of the Jacobian matrix in 29 is

$$\operatorname{diag} \frac{\partial C_l}{\partial \phi} = \operatorname{diag} e^{s_l} \tag{37}$$

which leads to the following Jacobian determinant for a coupling layer

$$\big|\det J_l\big| = \prod_{n(l)=1}^{V/2} \prod_{i=1}^{N_\phi} \exp s_l(\phi^i_{l,n(l)}) \tag{38}$$

The index $n(l)$ selects lattice sites from the partition which is being transformed by the l^{th} coupling layer.

The logarithm is extremely efficient to evaluate.

$$\log\big|\det J_l\big| = \sum_{n(l)=1}^{V/2} \sum_{i=1}^{N_\phi} s_l(\phi^i_{l,n(l)}) \tag{39}$$

4.2 Spline flows

Splines are functions defined piecewise by polynomials. The idea here is to define a coupling transformation that a closed interval to itself and is given by a spline.

To achieve this, the interval is divided into

- K segments (bins): $\{\mathcal{B}^k\}$, $k = 1, 2, \ldots, K$
- $K + 1$ 'knot points' (bin boundaries): $\{(x^k, \phi^k)\}$, $k = 0, 1, \ldots, K$.

A unique polynomial is defined in each segment using the outputs of a neural network as boundary conditions. With only two coupling layers (one for each partition), spline flows can be made arbitrarily expressive by increasing K.

We will make use of the following definitions for the width of bin $\mathcal{B}^k$ and the fractional position in a bin:

$$w_k = x^k - x^{k-1} \tag{40}$$

$$\alpha(x) = \frac{x - x^{k-1}}{w_k} \tag{41}$$

4.3 Linear spline

This construction, which maps $[0, 1]$ to itself, is the simplest and cheapest to implement.

A neural network outputs K values $\{\hat{p}_k\}$ which are immediately normalised using the softmax function

$$p_k = \frac{e^{\hat{p}_k}}{\sum_{k'=1}^{K} e^{\hat{p}_{k'}}} \tag{42}$$

so that $\sum_{k=1}^{K} p_k = 1$. The p_k are interpreted as probability masses for fixed bins of width $w = K^{-1}$.

Given $x \in \mathcal{B}^k$, the inverse coupling transform is defined as the cumulative distribution function associated with the p_k

$$C_l^{-1}(x; \{p_k\}) = \sum_{k'=1}^{k-1} p_{k'} + \int_0^{\alpha} d\alpha'\, p_k = \sum_{k'=1}^{k-1} p_{k'} + \alpha(x) p_k \tag{43}$$

The gradient is just the probability density p_k/w, so the contribution to the loss function

from one coupling layer is

$$\log\left|\det J_l\right| = \text{const.} - \sum_{n(l)}^{V/2}\sum_{i=1}^{N_\phi} \log p_{l,k(n,i)} \tag{44}$$

Note that the appropriate value $p_{l,k(n,i)}$ must be selected based on which bin $x^i_{l,n}$ resides in, which can be determined using bisection search.

One weakness of the linear spline map is that it is very tricky to include optimisation of the bin widths – i.e. the x^k – in the gradient descent. [REF Muller et al]. However, it has the advantage that the entire batch of x are sorted into the same set of bins.

4.4 Quadratic spline

This time, a neural network outputs K un-normalised widths $\{\hat{w}_k\}$ for each bin, and $K+1$ un-normalised 'height' values $\{\hat{h}_k\}$ for the bin boundaries.

The widths are normalised using a softmax, but the heights are normalised according to

$$h_k = \frac{e^{\hat{h}_k}}{\sum_{k'=1}^{K} \frac{w_{k'}}{2}\left(e^{\hat{h}_{k'}} + e^{\hat{h}_{k'+1}}\right)} \tag{45}$$

so that linear interpolation between the h_k yields a valid probability density function defined piecewise by

$$p_k(\alpha) = \left[h_k + \alpha(x)(h_{k+1} - h_k)\right] w_k \tag{46}$$

The coupling transformation is once again defined as the cumulative distribution function, which is, for $x \in \mathcal{B}^k$,

$$\begin{aligned} C_l^{-1}(x;\{h_k,w_k\}) &= \sum_{k'=1}^{k-1}\int_0^1 d\alpha'\, p_{k'}(\alpha') + \int_0^\alpha d\alpha'\, p_k(\alpha') \\ &= \sum_{k'=1}^{k-1}\frac{h_{k'}+h_{k'+1}}{2}\, w_{k'} + \alpha(x) h_k\, w_k + \frac{\alpha(x)^2}{2}(h_{k+1}-h_k)\, w_k \end{aligned} \tag{47}$$

The gradient is, again, p_k/w_k, so the contribution to the loss function reads

$$\log\left|\det J_l\right| = -\sum_{n(l)}^{V/2}\sum_{i=1}^{N_\phi} \log\left[h_k + \alpha(x)(h_{k+1}-h_k)\right] \tag{48}$$

4.5 Rational quadratic spline

Rational quadratic splines are defined by the quotient of two quadratic functions.

Since there are twice as many degrees of freedom as the simple quadratic case, the derivatives at the internal knot points also need to be fully specified by the neural network outputs, which must yield $3K - 1$ parameters:

- K widths $\{w_k\}$ for the bins, having been normalised using a softmax and multiplied by $2B$.
- K heights $\{h_k\}$ located at the bin centers, also having been normalised using a softmax and multiplied by $2B$.
- $K - 1$ gradients $\{d_k\}$ at the internal knot points, which are made positive using a softplus

  $$d_k = \log(1 + e^{\hat{d}_k}) \tag{49}$$

  so that the transformation is strictly monotonic.

The boundary conditions $(\phi^0, x^0) = (-B, -B)$, $(\phi^K, x^K) = (B, B)$, $d_0 = D_0$, $d_K = D_K$ constrain the remaining degrees of freedom. The authors set $D_0 = D_K = 1$ and define the coupling transformation as the identity outside the interval $[-B, B]$, thus allowing the map to take unconstrained inputs.

For convenience, define the slopes of the straight lines connecting knot points as

$$\frac{\phi^k - \phi^{k-1}}{x^k - x^{k-1}} = \frac{h_k}{w_k} \equiv s_k \tag{50}$$

A coupling transformation can then by defined piecewise by the unique rational quadratic

$$C_l^{-1}(x; \{h_k, s_k, d_k\}) = \sum_{k'=1}^{k-1} h_{k'} + \frac{h_k\big[s_k\alpha^2 + d_k\alpha(1-\alpha)\big]}{s_k + (d_{k+1} + d_k - 2s_k)\alpha(1-\alpha)} \tag{51}$$

for $x \in \mathcal{B}^k$.

The gradient is

$$\frac{1}{w_k}\frac{dC_l^{-1}}{d\alpha} = \frac{s_k^2\Big[d_{k+1}\alpha^2 + 2s_k\alpha(1-\alpha) + d_k(1-\alpha)^2\Big]}{\Big[s_k + (d_{k+1} + d_k - 2s_k)\alpha(1-\alpha)\Big]^2} \tag{52}$$

5. Flows on $\mathbb{S}^1$

5.1 $\mathbb{S}^1 \to \mathbb{S}^1$ diffeomorphisms

Diffeomorphisms are differentiable, invertible maps between smooth manifolds whose inverse is also differentiable. According to [1], the following four conditions are sufficient to guarantee that f is a diffeomorphism from $\mathbb{S}^1$ to itself:

$$f(0) = 0 \tag{53}$$
$$f(2\pi) = 2\pi \tag{54}$$
$$\nabla f(\phi) > 0 \tag{55}$$
$$\nabla f(\phi)|_{\phi=0} = \nabla f(\phi)|_{\phi=2\pi} \tag{56}$$

Note that

- (53) and (54) ensure 0 and 2π are mapped to the same point.
- (55) means f is a strictly monotonic function and is hence invertible.
- (56) implies that the Jacobians in the change of variables formula agree, so that the resulting probability density is continuous.

These conditions restrict 0 and 2π to being fixed points, but the authors of this paper propose an easy remedy which is the addition of a learnable phase shift.

5.2 Circular spline

The rational quadratic spline transformations discussed in section 4.5 have two fixed points where the gradients must also be specified.

Therefore, the $\mathbb{S}^1$ topology can be trivially built into the transformation by setting $C_l^{-1}(0) = 0$, $C_l^{-1}(2\pi) = 2\pi$, and making $d_0 = d_K$ part of the neural network output.

Since the spline transformation is already strictly monotonic, conditions (53 - 56) are all satisfied.

5.3 Mobius transformation

Not yet implemented

5.4 Non-compact projection

Take a point P on the unit circle, which can be parameterised by coordinates $(y^1, y^2) = (\sin\theta, \cos\theta)$, $(y^1)^2 + (y^2)^2 = 1$ in the embedding space, with $\theta \in [-\pi, \pi)$.

This point can be mapped to a point Y on the line tangent to the point $(0, -1)$, by considering the intersection of this tangent line with the line $\vec{QP}$, where $Q = (0, 1)$,

The result is the stereographic projection map

$$\begin{aligned} s \ &: (-\pi, \pi) \to \mathbb{R} \\ &\theta \to Y \end{aligned}$$

$$Y = \frac{y^1}{1 - y^2} = \frac{\sin\theta}{1 - \cos\theta} = \frac{2\sin\frac{\theta}{2}\cos\frac{\theta}{2}}{1 - \cos^2\frac{\theta}{2} + \sin^2\frac{\theta}{2}} = \cot\frac{\theta}{2} \tag{57}$$

which is defined for each point on the circle *except for* Q (we can denote this set $\mathbb{S}^1 - \{Q\}$.

Of course, this is not a diffeomorphism of the $\mathbb{S}^1$ class. However, it might be 'good enough' in the sense that, if we *define* $s(0) = 0$ and $s(2\pi) = 2\pi$, then the four conditions for a circle diffeomorphism are satisfied.

For convenience, we will define $\phi \equiv \theta + \pi \in [0, 2\pi)$ and $R \equiv Y^{-1}$, so that

$$R = \tan\left(\frac{\phi - \pi}{2}\right) \tag{58}$$

The map s has gradient

$$\frac{dR}{d\phi} = \frac{1}{2}\sec^2\left(\frac{\phi - \pi}{2}\right) \tag{59}$$

Clearly the inverse is

$$\phi = 2\tan^{-1} R + \pi \tag{60}$$

with gradient

$$\frac{d\phi}{dR} = \frac{2}{1 + R^2} \tag{61}$$

With this invertible projection map at our disposal, we can take a normalising flow on the real line, f, (e.g. Real NVP) and extend it to a map from $\mathbb{S}^1 - \{Q\}$ to itself.

$$\begin{aligned} s^{-1} \circ f \circ s \ &: (0, 2\pi) \to (0, 2\pi) \\ &\phi \mapsto x \end{aligned}$$

$$(s^{-1} \circ f \circ s)(\phi) = 2 \tan^{-1} \left[A \tan \left(\frac{\phi - \pi}{2} \right) + b \right] + \pi \tag{62}$$

with inverse

$$(s \circ f^{-1} \circ s^{-1})(x) = 2 \tan^{-1} \left[A^{-1} \tan \left(\frac{x - \pi}{2} \right) - A^{-1} b \right] + \pi \tag{63}$$

The determinant from the change of variables is

$$\begin{aligned} \left| \det \frac{\partial (s^{-1} \circ f \circ s)(\phi)}{\partial \phi} \right| &= \left| 2 \prod_{n=1}^{V} \frac{\partial \tan^{-1}(f_n)}{\partial f_n} \right| \left| \det \frac{\partial f(R)}{\partial R} \right| \left| \prod_{n=1}^{V} \frac{\partial R_n(\phi_n)}{\partial \phi_n} \right| \\ &= \left[\prod_{n=1}^{V} \frac{1}{1 + f_n^2} \right] \left| \det \frac{\partial f(R)}{\partial R} \right| \left[\prod_{n=1}^{V} \sec^2 \left(\frac{\phi_n - \pi}{2} \right) \right] \end{aligned} \tag{64}$$

where a factor of 2 from the first part and $1/2$ from the last have cancelled, and once again these contributions are always positive.

The contribution to the loss function is hence

$$\begin{aligned} \log \left| \det \frac{\partial (s^{-1} \circ f \circ s)(\phi)}{\partial \phi} \right| &= - \sum_{n=1}^{V} \log(1 + f_n^2) \\ &\quad + \log \left| \det \frac{\partial f(R)}{\partial R} \right| - 2 \sum_{n=1}^{V} \log \cos \left(\frac{\phi_n - \pi}{2} \right) \end{aligned} \tag{65}$$

6. Flows on $\mathbb{S}^2$

6.1 Non-compact projection

In section 5.4 the stereographic projection map was used to construct a flow on $\mathbb{S}^1$. However the stereographic projection generalises to a class of invertible maps from $\mathbb{S}^{D-1} - \{Q\} \to \mathbb{R}^{D-1}$ for any dimension $D \geq 2$.

Embedding the sphere using coordinates $\vec{y} \in \mathbb{R}^D$, the stereographic projection map is defined by

$$Y^i = \frac{y^i}{1 - y^D} \qquad i = 1, 2, \ldots, D-1 \tag{66}$$

with inverse

$$y^i = \frac{2Y^i}{S^2 + 1} \;\; ; \quad y^D = \frac{S^2 - 1}{S^2 + 1} \tag{67}$$

where $S^2 = \sum_{j=1}^{N-1} (Y^j)^2$.

In the case of $\mathbb{S}^2$, we define angles and axes such that

$$\begin{aligned} y^1 &= \sin\theta \, \cos\phi \\ y^2 &= \sin\theta \, \sin\phi \\ y^3 &= \cos\theta \end{aligned}$$

with $\theta \in [0, \pi]$ and $\phi \in [-\pi, \pi)$.[3]

If choose to invert the radial coordinate (so that $\cot \to \tan$), we have

$$Y^1 = \tan\frac{\theta}{2} \, \cos\phi \tag{68}$$

$$Y^2 = \tan\frac{\theta}{2} \, \sin\phi \tag{69}$$

$(R, \Theta) \equiv (\tan\frac{\theta}{2}, \phi)$ are the usual polar coordinates on $\mathbb{R}^2$. Thus, R is always positive, unlike in the one-dimensional case.

The inverse is

$$\theta = 2\tan^{-1}\left[\sqrt{(Y^1)^2 + (Y^2)^2}\right] \tag{70}$$

$$\phi = \arctan 2(Y^2, Y^1) \tag{71}$$

[3] Note the different domain for the azimuthal angle, which is necessary to match up with the definition of $\arctan 2$ used in the inverse projection map.

where the variant of the inverse tangent function known as $\arctan 2$, which maps input to $(-\pi, \pi)$, is used for the azimuthal angle.

The Jacobian determinant from the forward transformation is

$$\begin{aligned} \left|\det J_s\right| &= \left| \det \begin{pmatrix} \frac{1}{2}\sec^2\frac{\theta}{2}\cos\phi & \frac{1}{2}\sec^2\frac{\theta}{2}\sin\phi \\ -\tan\frac{\theta}{2}\sin\phi & \tan\frac{\theta}{2}\cos\phi \end{pmatrix} \right| \\ &= \frac{1}{2}\sec^2\frac{\theta}{2}\tan\frac{\theta}{2} \end{aligned} \tag{72}$$

which is always positive regardless of the modulus sign. The loss function will pick up a contribution from the logarithm of this Jacobian,

$$\log\left|\det J_s\right| = \log\sin\frac{\theta}{2} - 3\log\cos\frac{\theta}{2} + \text{const.} \tag{73}$$

For the inverse transformation, making use of $R^2 = (Y^1)^2 + (Y^2)^2$, we have

$$\begin{aligned} \left|\det J_{s^{-1}}\right| &= \left| \det \begin{pmatrix} \frac{2Y^1}{R(1+R^2)} & \frac{2Y^2}{R(1+R^2)} \\ \frac{-Y^2}{R^2} & \frac{Y^1}{R^2} \end{pmatrix} \right| \\ &= \frac{2}{R(1+R^2)} \end{aligned} \tag{74}$$

$$\log\left|\det J_{s^{-1}}\right| = -\log\big(R(1+R^2)\big) + \text{const.} \tag{75}$$

As a sanity check, we can see that

$$\frac{2}{R(1+R^2)} = \left[\frac{1}{2}\tan\frac{\theta}{2}\big(1+\tan^2\frac{\theta}{2}\big)\right]^{-1} = \left[\frac{1}{2}\tan\frac{\theta}{2}\sec^2\frac{\theta}{2}\right]^{-1} \tag{76}$$

This implies that, should no transformation of the Y^i occur between projection and inverse, their contributions to the loss function cancel exactly, as they should.

With this invertible projection map at our disposal, we can take a normalising flow on $\mathbb{R}^2$, f, (we use Real NVP) and construct a map $s^{-1} \circ f \circ s$ from $\mathbb{S}^2 - \{Q\}$ to itself.

The projection layers do not couple different lattice sites, so the Jacobian determinant for the transformation of an entire configuration will just be the product of Jacobians for each lattice site.

Hence, the contribution to the loss function from the map will be

$$\log\left|\frac{\partial(s^{-1}\circ f\circ s)(\phi)}{\partial\phi}\right| = \left[\sum_n \log\left|\det J_{s^{-1}}\big(f(s(\phi))\big)\right|\right]\left|\log\frac{\partial f\big(s(\phi)\big)}{\partial s(\phi)}\right|\left[\sum_n \log\left|\det J_s(\phi)\right|\right] \tag{77}$$

In practice, we evaluate the terms from left to right, since we start with the base fields x and generate the ϕ using $(s^{-1}\circ f^{-1}\circ s)(x)$.

6.2 Recursive flow

7. Uncertainties

7.1 The effect of autocorrelations

Markov-chain Monte Carlo methods such as Metropolis-Hastings are built on the concept of transitions between states, with transition probability depending on the current and proposed state. This means the set of configurations resulting from an MCMC process are correlated.

For an observable $\mathcal{O}$, the (un-normalised) autocorrelation function is given by

$$\Gamma_{\mathcal{O}}(k,t) = \left\langle \left(\mathcal{O}_{k+t} - \langle \mathcal{O}_{k+t} \rangle_M\right)\left(\mathcal{O}_k - \langle \mathcal{O}_k \rangle_M\right) \right\rangle_M \tag{78}$$

where $\langle\,\rangle_M$ denotes an average over an 'ensemble' of Markov chains.

Thankfully, if the Markov chain is allowed to reach a stationary state – a.k.a 'thermalise' – then $\Gamma_{\mathcal{O}}$ depends only on 'separations' t, and the ensemble average can be estimated by taking an average over a single chain.

$$\Gamma_{\mathcal{O}}(t) = \frac{1}{N_{\mathsf{conf}} - t} \sum_{k=1}^{N_{\mathsf{conf}}-t} \left(\mathcal{O}_{k+t} - \bar{\mathcal{O}}\right)\left(\mathcal{O}_k - \bar{\mathcal{O}}\right) \tag{79}$$

where $\bar{\mathcal{O}} \equiv \frac{1}{N_{\mathsf{conf}}} \sum_{k=1}^{N_{\mathsf{conf}}} \mathcal{O}_k$.

The exponential autocorrelation time $\tau_{\mathcal{O}}$ governs the large-t exponential decay of the autocorrelation function: $\Gamma_{\mathcal{O}}(t) \sim e^{-t/\tau_{\mathcal{O}}}$.

However, a measure of autocorrelation that is more readily attainable is the integrated autocorrelation time

$$\tau_{\mathcal{O}}^{\mathsf{int}} = \frac{1}{2} \sum_{t=-\infty}^{\infty} \frac{\Gamma_{\mathcal{O}}(t)}{\Gamma_{\mathcal{O}}(0)} = \frac{1}{2} + \sum_{t=1}^{\infty} \frac{\Gamma_{\mathcal{O}}(t)}{\Gamma_{\mathcal{O}}(0)} \tag{80}$$

In the simple case where $\Gamma_{\mathcal{O}}(t)/\Gamma_{\mathcal{O}}(0) = e^{-t/\tau_{\mathcal{O}}}$, we find that $\tau_{\mathcal{O}}^{\mathsf{int}} = \tau_{\mathcal{O}} - \frac{1}{2}$.

The practical consequence of autocorrelations is that the true statistical variance for an ensemble average is greater than the measured variance $\Gamma_{\mathcal{O}}(0)$. It is possible to show that

$$\sigma^2_{\mathcal{O},\mathsf{true}} = \Gamma_{\mathcal{O}}(0) \frac{2\tau_{\mathcal{O}}^{\mathsf{int}}}{N} + O(N^{-2}) \tag{81}$$

7.2 Approach to dealing with autocorrelations

The strategy adopted here is to sample from the Markov chain at an interval of $2\tau_{\mathcal{O}}^{\text{int}}$. This means the statistical error on an ensemble average takes the Gaussian form for the standard error on the mean

$$\sigma_{\mathcal{O},2\tau} = \sqrt{\frac{\Gamma_{\mathcal{O}}(0)}{N}} \tag{82}$$

When $\tau_{\mathcal{O}}^{\text{int}}$ is estimated from an observable, the sum is truncated at a value W, referred to as the summation window. The choice of of summation window should minimise the sum of the two dominant contributions to the error:

1. A bias due to truncating the sum. Assuming a simple exponential form this is[4]

$$b_W = -\sum_{t=W+1}^{\infty} e^{-t/\tau_{\mathcal{O}}} = -\tau_{\mathcal{O}}\, e^{-W/\tau_{\mathcal{O}}} \tag{83}$$

2. A statistical error which dominates the tail of $\tau_{\mathcal{O}}^{\text{int}}$ where the signal/noise ratio is low. This is given approximately by [REF SOKAL]

$$\sigma_W \approx 2\tau_{\mathcal{O}}^{\text{int}} \sqrt{\frac{W}{N}} \tag{84}$$

Thus, we want the minimum of

$$\Delta(W) = \tau_{\mathcal{O}}^{\text{int}}\, e^{-W/\tau_{\mathcal{O}}^{\text{int}}} + 2\tau_{\mathcal{O}}^{\text{int}} \sqrt{\frac{W}{N}} \tag{85}$$

In practice, we take as the summation window the smallest W upon which

$$\frac{\partial \Delta(W)}{\partial W} = -e^{-W/\tau_{\mathcal{O}}^{\text{int}}} + \frac{\tau_{\mathcal{O}}^{\text{int}}}{\sqrt{WN}} \tag{86}$$

changes sign.

7.3 The bootstrap technique

The bootstrap method is an approach for estimating statistical errors on ensemble averages of the fields, and functions of these. If used correctly, it can produce accurate

[4] This result can be easily obtained using a geometric series, though it may be more intuitive to evaluate the equivalent integral expression.

uncertainties very efficiently.

The central idea is to create a 'bootstrap sample' of field ensembles, where each bootstrap sampling comprises N_{ens} field configurations chosen randomly *with replacement*.

Observables are then computed for each ensemble in the bootstrap sample, and the error is quoted as the standard deviation of the resulting distribution.

In practice, we can avoid excess computations by calculating the volume-averaged two point correlators for every configuration in the ensemble, and then create a bootstrap sample of these.

8. Observables

8.1 Two point correlation functions

8.1.1 Two point correlator: Scalar

The connected two point correlation function for a scalar field with is

$$G(n,x) = \langle \phi_{n+x}\, \phi_n \rangle - \langle \phi \rangle^2 \tag{87}$$

The $\langle \phi \rangle$ has no dependence on n due to translational invariance Making further use of the translational symmetry, we can improve the estimator by averaging over the volume

$$G(x) = \frac{1}{V} \sum_n G(n,x) \tag{88}$$

Let us also make use of translational invariance to define the volume-averaged *disconnected*[5] two-point correlation function, which is defined for a single field labelled by an index 'a' rather than for an ensemble

$$G_V(x\,;a)_{\text{disconn}} = \frac{1}{V} \sum_n \phi_{n+x,a}\, \phi_{n,a} \tag{89}$$

This is a useful quantity since it allows us to study autocorrelations between field configurations generated with an MCMC algorithm, at different length scales.

In the interest of eliminating duplicated calculations, we can use the fact that

$$G(x) = \langle G_V(x)_{\text{disconn}} \rangle - \langle \phi \rangle^2 \tag{90}$$

8.1.2 Two point correlator: $O(N)$

The two point correlation function is given by

$$G(n,x) = \langle \sigma_{n+x} \cdot \sigma_n \rangle - \langle \sigma \rangle^2 \tag{91}$$

Since translational symmetry holds, we can apply the volume-averaged definitions as in

[5] The volume-average of ϕ may be non-zero, so this is not a connected correlation function.

the scalar case.

8.1.3 Two point correlator: CP^{N-1}

CP^{N-1} observables are often defined in terms of the outer product

$$P_n \equiv z_n^* \otimes z_n \tag{92}$$

The disconnected correlation function is given by

$$G(n,x)_{\text{disconn}} = \langle \operatorname{Tr} P_{n+x} P_n \rangle = \langle |z_{n+x}^* \cdot z_n|^2 \rangle \tag{93}$$

where we have used the trace property $\operatorname{Tr} AB = \sum_i (AB)_{ii} = \sum_i \sum_j A_{ij} B_{ji}$.

The connected correlator is just $G(x)_{\text{disconn}} - 1$ since $\langle \operatorname{Tr} P_n \rangle = \langle z_n^* \cdot z_n \rangle = \langle 1 \rangle$.

Not 100% sure

8.1.4 Fourier space correlator

The Fourier transform of $G(x)$ on a periodic lattice is

$$\tilde{G}(k) = \frac{1}{V} \sum_x e^{ik \cdot x} G(x) \tag{94}$$

where $k = \frac{2\pi \mathrm{n}}{L}$ take discrete values in the first Brillouin zone, labelled by integers $\mathrm{n} = -\frac{L}{2}, -\frac{L}{2}+1, \ldots, \frac{L}{2}-1$.

8.2 Two point observables

8.2.1 Susceptibility

The susceptibility is given by the Fourier space correlator at zero momentum:

$$\chi = \tilde{G}(0) = \frac{1}{V} \sum_x G(x) \tag{95}$$

8.2.2 Ising energy density

The average of correlators at unit separation measures the energy density associated with a Hamiltonian that contains purely nearest-neighbour interactions. This is often called the 'Ising energy density'.

$$E_{\text{ising}} = \frac{1}{d}\sum_{\mu} G(\hat{\mu}) \tag{96}$$

8.2.3 Zero momentum correlator

If we interpret one of the lattice dimensions as a time dimension, we can define a Fourier transformed correlator in each 'time slice'.

$$\tilde{G}(\vec{k}, t) = \frac{1}{L^{d-1}}\sum_{\vec{x}} e^{i\vec{k}\cdot\vec{x}}\, G(\vec{x}, t) \tag{97}$$

where $\vec{x}$ denotes the $d-1$ spatial dimensions (although this is just a single dimension in the $d = 2$ case).

The zero momentum correlator refers to the $\vec{k} = \vec{0}$ case. We will take advantage of rotational symmetry and define this as a dimensional average.

$$\tilde{G}_{\vec{0}}(t) = \frac{1}{d}\big(\tilde{G}(\vec{0}, t) + \tilde{G}(0, \ldots, t, 0) + \ldots + \tilde{G}(0, t, \ldots, 0) + \tilde{G}(t, \vec{0})\big) \tag{98}$$

8.2.4 Effective pole mass

For a periodic lattice, an effective pole mass can be defined by

$$m_p(t) = \cosh^{-1}\left(\frac{\tilde{G}_{\vec{0}}(t-1) + \tilde{G}_{\vec{0}}(t+1)}{2\tilde{G}_{\vec{0}}(t)}\right) \tag{99}$$

The quantity $m_p(t)^{-1}$ is also an estimator for the exponential correlation length. See appendix for details.

8.2.5 Second-moment correlation length

The second-moment correlation length can be defined for lattice two point correlators as

$$\xi^2_{(2m)} = \frac{1}{2d}\frac{\sum_x x^2\, G(x)}{\sum_x G(x)} \tag{100}$$

See appendix for details.

8.2.6 Low-momentum approximations for the correlation length

These definitions should coincide with the second-moment correlation length in the limit $V \to \infty$.

$$\xi^2_{(a)} = \frac{1}{d}\sum_\mu \frac{L_\mu^2}{(2\pi)^2}\left(1 - \frac{\mathbb{R}\mathrm{e}\,\tilde{G}(2\pi/L_\mu)}{\tilde{G}(0)}\right) \tag{101}$$

$$\xi^2_{(b)} = \frac{1}{d}\sum_\mu \frac{1}{4\sin^2(\pi/L_\mu)}\left(\frac{\tilde{G}(0)}{\mathbb{R}\mathrm{e}\,\tilde{G}(2\pi/L_\mu)} - 1\right) \tag{102}$$

The notation $\tilde{G}(2\pi/L_\mu)$ means to imply that the μ^{th} component of the momentum vector is $2\pi/L_\mu$, and all other components are zero.

See appendix for details.

8.3 Classical spin observables

As previously mentioned, we can interpret σ as N-dimensional classical spin vectors. In the classical spin context, we can define observables as functions of the spin Hamiltonian and squared magnetisation, which are defined by

$$H_{\text{spin}} = -\sum_n \sum_\mu \sigma_{n+\hat{\mu}} \cdot \sigma_n \qquad\qquad M^2 = \sum_n \sum_m \sigma_m \cdot \sigma_n$$

8.3.1 Energy density

The energy density is defined as the expectation value of the Hamiltonian density:

$$\begin{aligned} E_{\text{spin}} &= \frac{1}{V}\langle H_{\text{spin}}\rangle \\ &= -\frac{1}{V}\frac{1}{N_{\text{ens}}}\sum_{a=1}^{N_{\text{ens}}}\sum_{n}\sum_{\mu}\sigma_{n+\hat{\mu},a}\cdot\sigma_{n,a} \\ &= -\sum_{\mu}G(\hat{\mu}) \end{aligned} \tag{103}$$

This should be equal to $-d$ times the Ising energy.

8.3.2 Magnetic susceptibility

The magnetic susceptibility is defined in an analogous way[6]

$$\begin{aligned} \chi_m &= \frac{1}{V}\langle M^2\rangle \\ &= \frac{1}{V}\frac{1}{N_{\text{ens}}}\sum_{a=1}^{N_{\text{ens}}}\sum_{n}\sum_{m}\sigma_{m,a}\cdot\sigma_{n,a} \\ &= \sum_{x}G(x) \\ &= \tilde{G}(0) \end{aligned} \tag{104}$$

This should be equal to the susceptibility defined previously.

8.3.3 Heat capacity

At the next order there is the specific heat capacity[7]

$$c = \frac{\beta^2}{V}\Big[\langle H_{\text{spin}}^2\rangle - \langle H_{\text{spin}}\rangle^2\Big] = \beta^2\frac{\langle H_{\text{spin}}^2\rangle}{V} - \beta^2 V E_{\text{spin}}^2 \tag{105}$$

[6] This is also equivalent to the definition in ref. [2], which is $\chi_m = \frac{1}{V}\sum_n\sum_m\langle\sigma_m\cdot\sigma_n\rangle$, by just swapping the order of sums.

[7] The factors of β in the definition stem from the fact that the heat capacity is conventionally defined as a derivative of with respect to temperature, the inverse of β. To be explicit, $c = \frac{1}{V}\frac{\partial E_{\text{spin}}}{\partial T} = \frac{\beta^2}{V}\frac{\partial^2}{\partial\beta^2}\log\mathcal{Z}$

8.4 Definitions involving the $O(N)$ action

The $O(N)$ lattice action is

$$S = \frac{\beta}{2} \sum_n \sum_\mu \delta_\mu \sigma_n \cdot \delta_\mu \sigma_n = -\beta \sum_n \sum_\mu \left(\sigma_{n+\hat{\mu}} \cdot \sigma_n - 1 \right)$$

8.4.1 Alternative definition of energy density

Another definition of energy density, found in e.g. ref. [2]

$$\begin{aligned} E_{\text{deriv}} &= \frac{1}{V} \sum_n \sum_\mu \langle \delta_\mu \sigma_n \cdot \delta_\mu \sigma_n \rangle \\ &= \frac{2}{\beta V} \langle S \rangle \\ &= 2E_{\text{spin}} + 4 \end{aligned} \tag{106}$$

8.4.2 Equivalent definition of heat capacity

Eq. 105 is simply the variance of the $O(N)$ action, divided by volume.

$$c = \frac{1}{V} \left[\langle S^2 \rangle - \langle S \rangle^2 \right] \tag{107}$$

8.5 Topological observables

8.5.1 Charge density: geometrical, $O(3)$

A geometrical definition of topological charge density q for the $O(3)$ non-linear σ model on the $d = 2$ lattice was provided by Berg and Luscher in [2]. The general intuition is that the topological charge for a field configuration should count the number of times the field 'wraps around' the unit 2-sphere.

The charge density is defined for sites n^* on the 'dual lattice' – a point on the dual lattice is the midpoint of a square whose corners are adjacent sites on the lattice – and is a function of the four σ's surrounding n^*. By imagining that the four spin vectors define points on the same unit 2-sphere, we can interpret the surface area enclosed by geodesic paths connecting the points as a measure of topological charge density.

Add diagram.

The definition consists of the following set of equations. Firstly,

$$q_{n^*} = \frac{1}{4\pi}\Big(\Delta(\sigma_n, \sigma_{n+\hat{\mu}_1}, \sigma_{n+\hat{\mu}_1+\hat{\mu}_2}) + \Delta(\sigma_n, \sigma_{n+\hat{\mu}_1+\hat{\mu}_2}, \sigma_{n+\hat{\mu}_2})\Big) \tag{108}$$

defines the topological charge density in terms of a function of three of the four spins surrounding n^*. The function $\Delta(\vec{a}, \vec{b}, \vec{c})$ is the *signed*[8] area of a (unit radius) spherical triangle with corners given by the vectors $\vec{a}, \vec{b}, \vec{c}$.

If we instead parameterise Δ by three angles on the surface of the 2-sphere,

$$\begin{aligned}\cos\alpha &= \vec{a}\cdot\vec{b}\\ \cos\beta &= \vec{b}\cdot\vec{c}\\ \cos\gamma &= \vec{c}\cdot\vec{a}\end{aligned}$$

then Δ is given very simply by

$$\Delta(\alpha, \beta, \gamma) = \pm(\alpha + \beta + \gamma - \pi) \tag{109}$$

with the sign given by the sign of $\vec{a}\cdot(\vec{b}\times\vec{c}) = \sin\beta$.

In terms of the vectors, Δ may be given explicitly, including the sign, by

$$\exp\left(i\frac{\Delta(\vec{a}, \vec{b}, \vec{c})}{2}\right) = \frac{1 + \vec{a}\cdot\vec{b} + \vec{b}\cdot\vec{c} + \vec{c}\cdot\vec{a} + i\vec{a}\cdot(\vec{b}\times\vec{c})}{\left[2(1+\vec{a}\cdot\vec{b})(1+\vec{b}\cdot\vec{c})(1+\vec{c}\cdot\vec{a})\right]^{1/2}} \tag{110}$$

This leads to

$$\Delta(\vec{a}, \vec{b}, \vec{c}) = 2\tan^{-1}\left(\frac{\vec{a}\cdot(\vec{b}\times\vec{c})}{1 + \vec{a}\cdot\vec{b} + \vec{b}\cdot\vec{c} + \vec{c}\cdot\vec{a}}\right) \tag{111}$$

Since we always take the smaller of the two possible spherical triangles, we require that $\Delta(\vec{a}, \vec{b}, \vec{c}) \in (-2\pi, 2\pi)$, with extremal values corresponding to half of the surface area of the sphere being enclosed. However, note that eq. 111 runs into problems when two of the vectors lay on a great circle.

An important technical point is that we must use the variant of the $\tan^{-1}$ function, known as *arctan2*, which maps values to $(-\pi, \pi)$.

[8] This is just so that 'winds' in opposite directions cancel out rather than add. It helps to picture the case for $\mathbb{S}^1$.

8.5.2 Charge density: geometrical, CP^{N-1}

For CP^{N-1} the geometrical approach results in the following definition of lattice topological charge density [2]

$$q_{n^*} = \frac{1}{2\pi}\,\mathbb{I}\mathrm{m}\left[\log \mathrm{Tr}(P_{n+\hat{\mu}+\hat{\nu}}P_{n+\hat{\mu}}P_n) + \log \mathrm{Tr}(P_{n+\hat{\nu}}P_{n+\hat{\mu}+\hat{\nu}}P_n)\right] \tag{112}$$

8.5.3 Charge

Topological charge is defined in terms of the charge density by

$$Q = \sum_{n^*} q_{n^*} \tag{113}$$

8.5.4 Susceptibility

Hence, the topological susceptibility is

$$\chi_t = \frac{1}{V}\langle Q^2 \rangle \tag{114}$$

8.6 Deterministic approach for small lattices

To do

9. Appendices

9.1 Generalised spherical coordinates

Given that the non-linear σ models come with a unit norm constraint, parameterising the fields using generalised spherical coordinates [3] can be useful since the constraint is built in.

For the $O(N)$ models the parameterisation looks like:

$$\begin{aligned}
\sigma^1 &= \cos\phi^1 \\
\sigma^2 &= \sin\phi^1 \cos\phi^2 \\
\sigma^3 &= \sin\phi^1 \sin\phi^2 \cos\phi^3 \\
&\vdots \\
\sigma^{N-1} &= \sin\phi^1 \dots \sin\phi^{N-2} \cos\phi^{N-1} \\
\sigma^N &= \sin\phi^1 \dots \sin\phi^{N-1}
\end{aligned} \tag{115}$$

with the angles taking values in the range $\phi^1, \dots, \phi^{N-2} \in [0, \pi], \phi^{N-1} \in [0, 2\pi)$.

In the CP^{N-1} case things are slightly different. Let us fix the gauge so that the N^{th} component of the field is real:

$$z = \begin{pmatrix} x^1 + ix^{N+1} \\ x^2 + ix^{N+1} \\ \vdots \\ x^{N-1} + ix^{2N-1} \\ x^N \end{pmatrix} \tag{116}$$

The x can then be parameterised using generalised spherical coordinates:

$$\begin{aligned}
x^1 &= \cos\phi^1 \\
x^2 &= \sin\phi^1 \cos\phi^2 \\
x^3 &= \sin\phi^1 \sin\phi^2 \cos\phi^3 \\
&\vdots \\
x^{D-2} &= \sin\phi^1 \dots \sin\phi^{D-2} \cos\phi^{D-1} \\
x^{D-1} &= \sin\phi^1 \dots \sin\phi^{D-1}
\end{aligned} \tag{117}$$

although this time there are $D = 2N-1$ angles, taking values in the range $\phi_1, \ldots, \phi_{D-2} \in [0, \pi], \phi^{D-1} \in [0, 2\pi)$.

9.2 Kullbach-Leibler divergence

Representation learning requires a rigorous way of quantifying how well a generated probability distribution approximates the target.

The Kullbach-Leibler (KL) divergence is a measure of how one probability distribution differs from a reference probability distribution. For probability measures P and Q over a set $\mathcal{X}$, the KL divergence is defined by

$$D_{KL}(P \,||\, Q) = \int_{\mathcal{X}} dP \log\left(\frac{dP}{dQ}\right) \tag{118}$$

The two vertical bars in the argument are there to make explicit that Q is the reference distribution. This is important because, despite often being seen as a 'distance' between two distributions, the KL divergence is asymmetric and so the order of arguments matters.

To translate this into a more useful form, we introduce probability density functions $p \equiv \frac{dP}{d\mu}$ and $q \equiv \frac{dQ}{d\mu}$ for some measure μ on $\mathcal{X}$. In terms of the density functions, the KL divergence reads

$$\begin{aligned} D_{KL}(p \,||\, q) &= \int_{\mathcal{X}} d\mu\, p \log\left(\frac{p}{q}\right) \\ &= -\int_{\mathcal{X}} d\mu\, p \log q + \int_{\mathcal{X}} d\mu\, p \log p \end{aligned} \tag{119}$$

These two terms are, respectively, the cross entropy between P and Q, and the entropy of P. Thus, the KL divergence is always non-negative, and is zero if $P = Q$.

The loss function used in `anvil` is the shifted[9] Kullbach-Leibler (KL) divergence between the distribution resulting from the flow model, with probability measure $d\tilde{P}_f(\phi) = d\mu_0\, \tilde{p}_f(\phi)$ over the space of field configurations, and the 'target' (reference) distribution $dP(\phi) = d\mu_0\, p(\phi)$.

$$\begin{aligned} L(\tilde{P}_f) &\equiv D_{KL}(\tilde{P}_f \,||\, P) - \text{const.} \\ &= \int d\mu_0\, \tilde{p}_f(\phi) \big[\log \tilde{p}_f(\phi) - \log p(\phi) - \text{const.} \big] \end{aligned} \tag{120}$$

[9] The constant shift has no effect on the shape of the loss function, but lets us neglect explicit computation of additive constants. The benefit will become evident in the examples to follow.

For lattice field theories, $d\mu_0$ arises from discretisation of the functional integral, taking the form of a product over every component of every field on every lattice site. For a theory of a single field defined on a lattice with V sites, parameterised by N_ϕ coordinates, this is

$$d\mu_0 \equiv \prod_{n=1}^{V} \prod_{i=1}^{N_\phi} d\phi_n^i \tag{121}$$

Practically speaking, stochastic gradient descent is used to train neural networks within the flow model to minimise a stochastic estimate of this loss function. This amounts to exchanging the integral over the probability measure for an ensemble average (or, to use machine-learning terminology, a batch average).

$$\int d\mu_0 \, \tilde{p}_f(\phi) \xrightarrow[estimate]{stochastic} \frac{1}{N_{\text{batch}}} \sum_{a=1}^{N_{\text{batch}}}$$

It is important that the target density $p(\phi)$ is such that $d\mu_0 \, p(\phi)$ is a valid probability measure. This means any Jacobian factor arising from a non-trivial parameterisation of the fields must be absorbed into the definition of $p(\phi)$.

Scalar field theory

We start by writing down the probability measure for the target distribution

$$dP(\phi) = \prod_{n=1}^{V} d\phi_n \frac{e^{-S(\phi)}}{\mathcal{Z}} \tag{122}$$

Thus, it is clear that we must associate the density function $p(\phi)$ with $\frac{1}{\mathcal{Z}} e^{-S(\phi)}$.

We generally cannot compute $\mathcal{Z}$ directly, but since its logarithm appears as an additive constant in the loss function, we can absorb it into the definition of our loss function using the constant shift, giving

$$\begin{aligned} L_{\text{scalar}} &\equiv \int \prod_{n=1}^{V} d\phi_n \, \tilde{p}_f(\phi) \Big(\log \tilde{p}_f(\phi) - \log p(\phi) - \log \mathcal{Z} \Big) \\ &= \int \prod_{n=1}^{V} d\phi_n \, \tilde{p}_f(\phi) \Big(\log \tilde{p}_f(\phi) + S(\phi) \Big) \end{aligned} \tag{123}$$

The quantity we actually evaluate (the stochastic estimate) is, using eq. 25

$$\frac{1}{N_{\text{batch}}} \sum_{a=1}^{N_{\text{batch}}} \left[\log r\big(f(\phi_a)\big) + \log \left| \det \frac{\partial f(\phi_a)}{\partial \phi_a} \right| + S(\phi_a) \right] \tag{124}$$

$O(N)$ non-linear σ models

The probability measure for the lattice $O(N)$ non-linear σ models is

$$dP(\sigma) = \prod_{n=1}^{V} \delta(\sigma_n^2 - 1) \prod_{i=1}^{N} d\sigma_n^i \; \frac{e^{-S(\sigma)}}{\mathcal{Z}} \tag{125}$$

As discussed earlier in this report, for $N > 2$, there is a Jacobian factor due to parameterising the σ fields with a set of angles, which is the same as that which occurs when expressing the surface area element of an $N-1$ sphere in spherical coordinates, and is denoted $\Omega_{N-1}(\phi_n)$.

$$\begin{aligned} dP(\phi) &= \prod_{n=1}^{V} \Omega_{N-1}(\phi_n) \prod_{i=1}^{N-1} d\phi_n^i \; \frac{e^{-S(\phi)}}{\mathcal{Z}} \\ &= \left[\prod_{n=1}^{V} \prod_{i=1}^{N-1} d\phi_n^i \right] \left[\prod_{m=1}^{V} \prod_{j=1}^{N-1} (\sin \phi_m^j)^{N-1-j} \; \frac{e^{-S(\phi)}}{\mathcal{Z}} \right] \end{aligned} \tag{126}$$

This makes it clear that we should associate $p(\phi)$ with the second square bracket.

The loss function is then (again absorbing $\log \mathcal{Z}$ into the shift)

$$L_{O(N)} = \int \prod_{n=1}^{V} \prod_{i=1}^{N-1} d\phi_n^i \; \tilde{p}_f(\phi) \Big(\log \tilde{p}_f(\phi) - \sum_{n=1}^{V} \log \Omega_{N-1}(\phi_n) + S(\phi) \Big) \tag{127}$$

The stochastic estimate of the loss function hence becomes

$$\begin{aligned} \frac{1}{N_{\text{batch}}} \sum_{a=1}^{N_{\text{batch}}} \Bigg\{ & \log r\big(f(\phi_a)\big) + \log \left| \det \frac{\partial f(\phi_a)}{\partial \phi_a} \right| \\ & - \sum_{n=1}^{V} \log \Omega_{N-1}(\phi_{n,a}) + S(\phi_a) \Bigg\} \end{aligned} \tag{128}$$

The term due to the parameterisation can be evaluated reasonably cheaply using array

routines:

$$\log \Omega_{N-1}(\phi_n) = \begin{pmatrix} N-2 \\ N-3 \\ \vdots \\ 1 \end{pmatrix} \cdot \log \sin \begin{pmatrix} \phi_n^1 \\ \phi_n^2 \\ \vdots \\ \phi_n^{N-2} \end{pmatrix} \tag{129}$$

To convince ourselves that this loss function makes sense, let us take the case of the Heisenberg ($O(3)$) model in the infinite-temperature limit, where $S(\phi) \to 0$ and spins are distributed uniformly on the 2-sphere.

Here, the term arising from the parameterisation is $-\sum_n \log \sin \phi_n^1$. Thus, since the loss is being minimised, angles closer to the equator will be favoured compared to the angles near the poles, as they should be.

Furthermore, we can imagine that we generate initial configurations which themselves follow the uniform distribution, meaning the first term will be $+\sum_n \log \sin f(\phi_n^1)$. Since we started with fields which already followed the target distribution, we would hope that the model density will converge to $\tilde{p}_f \to r \implies f \to \mathbb{I}$, with $\mathbb{I}$ denoting the identity map. If it does so, the terms coming from the parameterisation and the base distribution exactly cancel, so these terms do not provide an avenue for further minimisation of the loss function, which is indeed what we would hope for.

9.3 Correlation lengths

The exponential correlation length ξ marks a transition between power law and exponential behaviour in the two point correlation function. For separations much larger than the correlation length,

$$G(x) \sim \frac{e^{-|x|/\xi}}{|x|^{\vartheta}} \qquad\qquad |x| \gg \xi \tag{130}$$

where ϑ is an exponent which depends on the system under study; for Gaussian models (at most quadratic in the fields and their derivatives), $\vartheta = \frac{d-1}{2}$.

The second-moment correlation length, defined by

$$\xi_{(2m)}^2 = C \frac{\int d^d x \, x^2 \, G(x)}{\int d^d x \, G(x)} \tag{131}$$

can be shown to coincide with ξ^2 for a suitable proportionality constant C, by switching to spherical coordinates and performing the one-dimensional integral over $|x|$ by parts. For one-dimensional Gaussian systems ($\vartheta = 0$), $C = \frac{1}{2}$.

We can note that $\xi^2_{(2m)}$ may equivalently be expressed as

$$\xi^2_{(2m)} = -\frac{C}{\tilde{G}(0)} \frac{\partial^2}{\partial k^2} \tilde{G}(k) \bigg|_{k_\mu = 0} \tag{132}$$

A related quantity is the 'physical' or 'pole' mass, resulting from the pole of the propagator at zero spatial momentum (the 'wall-wall' propagator).

$$\tilde{G}(\vec{0}, t) \sim e^{-mt} \qquad\qquad t \gg m^{-1} \tag{133}$$

The reciprocal of the mass can be seen as a correlation length in one dimension.

Lattice definitions

This is all well and good for infinite systems, but we would like to be able to define a correlation length on a finite lattice, which matches ξ in the $V \to \infty$ limit.

We will take the constant of proportionality to be $(2d)^{-1}$ so that, when a dimensional average is taken over the lattice, the result is consistent with a one-dimensional Gaussian model.

The most obvious thing to do is replace eq. 131 with its discrete, finite volume analogue, although we must remember to account for periodic boundary conditions since the separation x appears explicitly.

$$\xi^2_{(2m)} = \frac{1}{2d} \frac{\sum_x x^2 \, G(x)}{\sum_x G(x)} \tag{134}$$

Low momentum approximations to the correlation length

Another option is to take the discrete, finite volume analogue of eq. 132:

$$\begin{aligned} \xi^2_{(a)} &= -\frac{1}{2d\tilde{G}(0)} \sum_\mu \frac{\tilde{G}(2\pi/L_\mu) + \tilde{G}(-2\pi/L_\mu) - 2\tilde{G}(0)}{(2\pi/L_\mu)^2} \\ &= \frac{1}{d} \sum_\mu \frac{L_\mu^2}{(2\pi)^2} \left(1 - \frac{\mathbb{R}\mathrm{e}\, \tilde{G}(2\pi/L_\mu)}{\tilde{G}(0)} \right) \end{aligned} \tag{135}$$

The notation $\tilde{G}(2\pi/L_\mu)$ means to imply that the μ^{th} component of the momentum vector is $2\pi/L_\mu$, and all other components are zero. In the second line, we have used

the fact that $\tilde{G}(-k) = \tilde{G}(k)^*$ since $G(x)$ is real.

Using the form of the lattice propagator $\tilde{G}(k) \propto \left(\sum_\mu \hat{k}_\mu^2 + \xi^{-2}\right)^{-1}$ (where the proportionality hides renormalisation effects),

$$\frac{2\tilde{G}(0)}{\tilde{G}(2\pi/L_\mu) + \tilde{G}(-2\pi/L_\mu)} = \frac{\tilde{G}(0)}{\mathbb{R}\mathrm{e}\,\tilde{G}(2\pi/L_\mu)} = \xi^2\Big(4\sin^2(\pi/L_\mu) + \frac{1}{\xi^2}\Big) \tag{136}$$

Hence, we have a third expression for the correlation length:

$$\xi_{(b)}^2 = \frac{1}{d}\sum_\mu \frac{1}{4\sin^2(\pi/L_\mu)}\left(\frac{\tilde{G}(0)}{\mathbb{R}\mathrm{e}\,\tilde{G}(2\pi/L_\mu)} - 1\right) \tag{137}$$

Pole mass

On a periodic lattice with extent T in the time dimension, it can be shown that the zero-momentum correlator takes the following form (for a single particle, or in the long-t limit)

$$\tilde{G}(\vec{0}, t) \sim e^{-mt} + e^{-m(T-t)} = 2e^{-m\frac{T}{2}}\cosh(mt') \tag{138}$$

with $t' \equiv \frac{T}{2} - t$.

We can obtain m as follows:

$$\begin{aligned}\frac{\tilde{G}(\vec{0}, t+1) + \tilde{G}(\vec{0}, t-1)}{2\tilde{G}(\vec{0}, t)} &= \frac{\cosh\big(m(t'+1)\big) + \cosh\big(m(t'-1)\big)}{2\cosh(mt')} \\ &= \frac{2\cosh(mt')\cosh(m)}{2\cosh(mt')} \\ &= \cosh(m)\end{aligned} \tag{139}$$

The dimensional averaging has already been taken care of in the definition of the zero-momentum correlator.

References

[1] M S Albergo, G Kanwar, and P E Shanahan. Normalizing flows on tori and spheres. *arXiv preprint*, 2020.

[2] B Berg and M Luscher. Definition and statistical distributions of a topological number in the lattice o(3) sigma model. *Nuclear Physics B*, 190:412–424, 1981.

[3] L E Blumenson. A derivation of n-dimensional spherical coordinates. *The American Mathematical Monthly*, 67:63, 1960.