Practical Appendix for *On the Anatomy of MCMC-Based Maximum Likelihood Learning of Energy-Based Models*

Erik Nijkamp, Mitch Hill, Tian Han, Song-Chun Zhu, and Ying Nian Wu
UCLA Department of Statistics
8117 Math Sciences Bldg.
Los Angeles, CA 90095-1554

A Energy Initialization and Scale of SGD Learning Rate for Convergent ML

In this section we discuss some details about initializing the energy function and scaling the SGD learning rate. Energy initialization is important for efficient convergent ML but not crucial for non-convergent ML. We find that convergent ML is most effective when r_t (see Section 3.2) has approximately the same order of magnitude throughout training. With noise $\varepsilon = 0.015$, we observe that r_t typically lies in the range [0.08, 0.15] for large t. However, when the initial weights θ_1 come from standard ConvNet initialization, we observe $r_1 \approx 10^{-6}$. To address this we use the scaled energy

$$U(x;\theta) = \frac{F(x;\theta)}{\varepsilon^2/2}, \tag{1A}$$

where F is a ConvNet. This is equivalent to using the Langevin update

$$X_{\ell+1} = X_\ell - \frac{\partial}{\partial x} F(X_\ell;\theta) + \varepsilon Z_\ell. \tag{2A}$$

When θ_1 is obtained from standard ConvNet initialization and the rescaled energy (1A) is used, we observe that

$$r_1 = \left[\frac{1}{L+1}\sum_{\ell=0}^{L}\left\|\frac{\partial}{\partial y}F(Y_1^{(\ell)};\theta_1)\right\|_2\right] \approx 0.01$$

which is within a reasonable magnitude of the approximate target range [0.08, 0.15]. Additional scaling is required when $r_1 \approx 0.01$ is either too low or high for the ideal noise ε and the target range of r_t but the same principles apply.

We note that the rescaling causes further complications, since the computational loss

$$d_{s_t}(\theta) = \frac{2}{\varepsilon^2}\left(E_q[F(X;\theta_t)] - E_{s_t}[F(X;\theta_t)]\right)$$

now depends on ε. To address this, we find that is helpful to use a scaled learning rate $\gamma = \frac{\varepsilon^2}{2}\gamma_0$ where $\gamma_0 \approx 0.0005$, to obtain the update gradient

$$\gamma\Delta\theta_t = \gamma_0\left[\frac{\partial}{\partial\theta}\left(\frac{1}{n}\sum_{i=1}^{n}F(X_i^+;\theta_t) - \frac{1}{m}\sum_{i=1}^{m}F(X_i^-;\theta_t)\right)\right] \tag{3A}$$

where $\Delta\theta_t$ is given by (8). When using the vanilla SGD update

$$\theta_{t+1} = \theta_t - \gamma\Delta\theta_t, \tag{4A}$$

the scale of the parameter change $\|\theta_{t+1} - \theta_t\|_2 = \|\gamma\Delta\theta_t\|_2$ depends only on the scale of $\|\frac{\partial}{\partial\theta}F(x;\theta_t)\|_2$ and the scale of γ_0 and not on the scale of ε. We find that this enables standardized weight initialization and LR tuning that is independent of ε. In practical training of convergent models we implement ML learning using (1A), (2A), (3A), and (4A).

B Annealing the Learning Rate

Annealing the learning rate can greatly reduce the number of weight updates needed for realistic convergent learning. Approximate MCMC convergence of short-run samples (i.e. $s_t \approx p_{\theta_t}$) only needs to occur at the end of training for convergent learning. Using a high SGD learning rate with non-convergent learning dynamics early in training helps the model learn realistic features before annealing the learning rate to correct the steady-state oversaturation. We use a high learning rate of 0.05 early in training and anneal to the target value of 0.0005 over about 100,000 updates.

C Sampling from the Data Distribution

For the 2D toy experiments presented in the paper, one can easily generate infinite samples from the true density q. Additional complications arise when training an EBM p_θ to model image data because typical image datasets only contain a finite number of samples so that the data distribution q is actually a Dirac-delta distribution over the training images. In this case the true target distribution q is actually degenerate over the continuous state space $\mathbb{R}^N$.

The discrepancy between the degenerate target distribution q and the fully-supported distribution p_θ can cause instabilities during training. Images with a solid color background such as MNIST digits can easily be assigned disproportionately low energy, because the energy function p_θ can learn to discriminate between positive and negative samples based on the behavior of a few consistent pixels in the training data. If p_θ is able to consistently assign lower energy to positive images based on features do not occur for the negative images then learning can collapse as $d_{s_t} \to -\infty$.

The instability described above arises because certain linear dimensions of the training distribution have a much lower local standard deviation than synthesized samples, which have a standard deviation of at least ε in all directions from Langevin sampling. In fact, all dimensions of the training distribution have a local standard deviation of 0 since q is a Dirac-delta function. We can overcome this discrepancy by adding Gaussian noise when sampling images from the training set:

$$X_j^+ = x_{\varphi(j)}^+ + \varepsilon_{\text{data}}\, Q_j \tag{5A}$$

where $\varphi(j) \sim \text{Unif}(\{1, \ldots, N_{\text{data}}\})$, $x_{\varphi(j)}^+ \in \{x_i^+\}_{i=1}^{N_{\text{data}}}$ is a training image, $Q_j \sim \text{N}(0, I_N)$ and $\varepsilon_{\text{data}} \geq \varepsilon$. If (5A) is applied each time that an image is sampled from the training data, then q becomes a Gaussian mixture model with modes at the training data and isotropic covariance with standard deviation $\varepsilon_{\text{data}}$ around each mode. Therefore q is no longer degenerate over $\mathbb{R}^N$ and the minimum local standard deviation ε of the Langevin process is smaller than the minimum local standard deviation $\varepsilon_{\text{data}}$ of q as required for stable learning. We typically use $\varepsilon_{\text{data}} = 2\varepsilon$.

D A Note on Oscillation

As described in Section 3.1, we always observe that d_{s_t} is approximately symmetrically distributed around 0 for sufficiently large values of t for both convergent and non-convergent ML. We refer to this behavior as oscillation of energy differences between the positive and negative samples. We can further identify two types of oscillation: *weak oscillation* and *strong oscillation*.

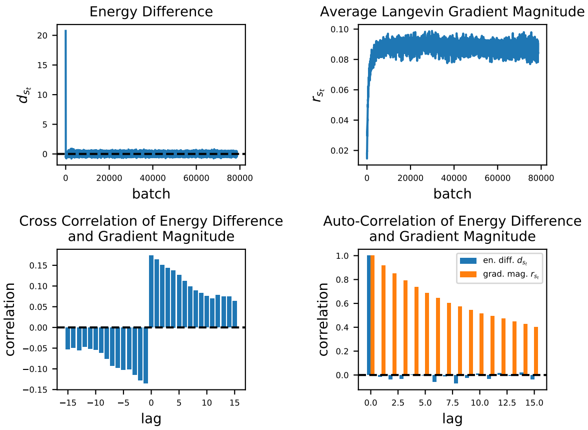

Figure 1: Diagnostic plots for ML learning with weak oscillation. The values of d_{s_t} are symmetrically distributed around 0 (*upper left*) and the Langevin gradient magnitude converges to a value that is balanced with ε (*upper right*) as expected. However, there is no observable trend in the autocorrelation of d_{s_t} (*lower right*). Although d_{s_t} does oscillate around 0, the oscillation is not dependent on the outcome of recent learning iterations.

Weak oscillation refers to learning outcomes where the sign of d_{s_t} is not influenced by the sign of $d_{s_{t_0}}$ for $t_0 < t$. The prototypical case of weak oscillation occurs for perfect modeling ($q = p_{\theta_t} = s_t$) with no learning (learning rate $\gamma = 0$). In this case $d_{s_t} = 0$ and the difference of the finite-sample expectation from the positive and negative samples will be symmetrically distributed around 0 independently of the finite-sample expectations from any previous learning iteration t_0. We observe weak oscillation for toy 2D distributions and for image datasets when a low SGD learning rate is used.

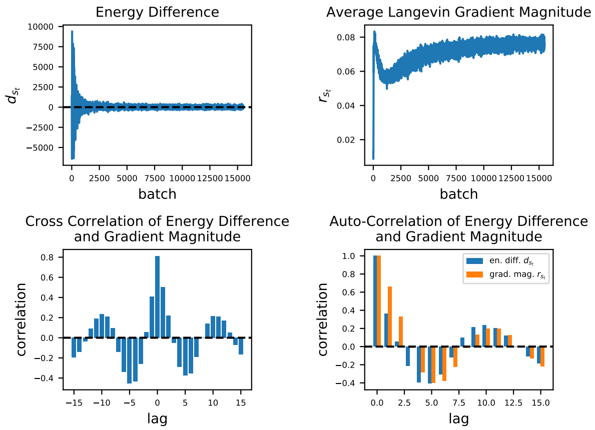

Figure 2: Diagnostic plots for ML learning with strong oscillation. The values of d_{s_t} are symmetrically distributed around 0 (*upper left*) and the Langevin gradient magnitude converges to a value that is balanced with ε (*upper right*) as expected. This time d_{s_t} has a strong negative autocorrelation at a short-range lag (*lower right*). Moreover, Langevin gradient magnitude shares this short-range negative auto-correlation (*lower right*) and the cross-correlation of d_{s_t} and r_t (*lower left*) follows the contraction/expansion relation described in Section 3.1. The oscillation of d_{s_t} around 0 is highly dependent on the outcome of recent updates.

Strong oscillation refers to learning outcomes where the sign of d_{s_t} in the current learning iteration tends to be the opposite of the sign of $d_{s_{t_0}}$ for recent learning iterations t_0. In other words, expansion updates tend to immediately follow contraction updates and vice-versa when the learning system experiences strong oscillation, as described in the Section 3.1. We observe strong oscillation when learning image datasets with Adam or with SGD and high learning rate. Strong oscillation appears to occur primarily for high-dimensional energy functions.

In general, it appears that weak oscillation tends to occur for convergent ML and strong oscillation tends to occur for non-convergent ML. The substantial dependence on previous learning iterations exhibited during strong oscillation could be an indicator that the model is changing too quickly to learn a realistic steady-state from the distribution of negative samples. In contrast, weak oscillation is the expected outcome in the case of perfect ML learning and a convergent model should also display this behavior.