Supplementary material: Learning Dynamics of Linear Denoising Autoencoders

Arnu Pretorius   Steve Kroon   Herman Kamper

Supplementary material

The following section provides detail omitted in the paper regarding the derivation of certain equations as well as additional comments.

A. Expected loss for linear DAEs

We derive the expected reconstruction loss over the noise distribution as presented in (1) in the paper. The expected loss can be written as

$$\mathbb{E}_{\epsilon}[\mathcal{L}] = \frac{1}{2N}\sum_{i=1}^{N}\mathbb{E}_{\epsilon}\left[||\mathbf{x}_i - W_2W_1\tilde{\mathbf{x}}_i||^2\right].$$

where $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \epsilon_i$, with ϵ sampled from an isotropic noise distribution with component variance s^2. Let $SE(\tilde{\mathbf{x}}_i) = ||\mathbf{x}_i - W_2W_1\tilde{\mathbf{x}}_i||^2$ and $M = W_2W_1$. Then

$$\begin{aligned}\mathbb{E}_{\epsilon}\left[SE(\tilde{\mathbf{x}}_i)\right] &= \mathbb{E}_{\epsilon}\left[||(I-M)\mathbf{x}_i + M(\mathbf{x}_i - \tilde{\mathbf{x}}_i)||^2\right] \\ &= SE(\mathbf{x}_i) + \mathbb{E}_{\epsilon}\left[||M(\mathbf{x}_i - \tilde{\mathbf{x}}_i)||^2\right]\end{aligned}$$

because the cross product terms vanish, since $\mathbb{E}_{\tilde{\mathbf{x}}}\left[\tilde{\mathbf{x}}_i\right] = \mathbf{x}_i$:

$$\begin{aligned}0 &= \mathbb{E}_{\epsilon}\left[\mathbf{x}_i^T(I-M)^TM(\mathbf{x}_i - \tilde{\mathbf{x}}_i)\right] \\ &= \mathbb{E}_{\epsilon}\left[(\mathbf{x}_i - \tilde{\mathbf{x}}_i)^TM^T(I-M)\mathbf{x}_i\right].\end{aligned}$$

We also have that

$$\begin{aligned}||M(\mathbf{x}_i - \tilde{\mathbf{x}}_i)||^2 &= (\mathbf{x}_i - \tilde{\mathbf{x}}_i)^TM^TM(\mathbf{x}_i - \tilde{\mathbf{x}}_i) \\ &= \text{tr}\left[(\mathbf{x}_i - \tilde{\mathbf{x}}_i)^TM^TM(\mathbf{x}_i - \tilde{\mathbf{x}}_i)\right] \\ &= \text{tr}\left[M(\mathbf{x}_i - \tilde{\mathbf{x}}_i)(\mathbf{x}_i - \tilde{\mathbf{x}}_i)^TM^T\right] \\ &= \text{tr}\left[M\epsilon_i\epsilon_i^TM^T\right]\end{aligned}$$

due to the invariance of the trace under cycle permutation of products. Therefore, in expectation over the noise we have

$$\mathbb{E}_{\epsilon}\left[||M(\mathbf{x}_i - \tilde{\mathbf{x}}_i)||^2\right] = \text{tr}\left[M(s^2I)M^T\right],$$

where $\text{Var}(\epsilon_i)$ represents the diagonal matrix containing the variances of each the components of the vector ϵ_i. However, since the noise is sampled i.i.d. for each component of ϵ_i, for all $i = 1, ..., N$, we get the expected reconstruction loss over the noise as

$$\begin{aligned}\mathbb{E}_{\epsilon}\left[\mathcal{L}\right] = \frac{1}{2N}\sum_{i=1}^{N}&||\mathbf{x}_i - W_2W_1\mathbf{x}_i||^2 \\ &+ Ns^2\text{tr}\left(W_2W_1W_1^TW_2^T\right).\end{aligned}$$

B. Learning dynamics for linear DAEs

We derive the expression for the learning dynamics of a linear DAE as presented in (5) in the paper. As departure point, we start by examining the expected scalar update equations over the noise model for a small learning rate α, which can be written as

$$\begin{aligned}\tau\frac{d}{dt}w_1 &= w_2(\lambda - w_2w_1\lambda) - \varepsilon w_2^2w_1 \\ \tau\frac{d}{dt}w_2 &= w_1(\lambda - w_2w_1\lambda) - \varepsilon w_2w_1^2.\end{aligned}$$

where $\tau = \frac{N}{\alpha}$, with N representing the number of training samples. Define $w = w_2w_1$ and using the product rule the update for w then becomes

$$\begin{aligned}\tau\frac{d}{dt}w &= \tau[w_1\frac{d}{dt}w_2 + w_2\frac{d}{dt}w_1] \\ &= w_1^2(\lambda - w_2w_1(\lambda+\varepsilon)) + w_2^2(\lambda - w_2w_1(\lambda+\varepsilon)) \\ &= (\lambda - w(\lambda+\varepsilon))(w_1^2 + w_2^2). \qquad (1)\end{aligned}$$

Next we make the following hyperbolic change of coordinates

$$w_1 = \sqrt{c_0}\sinh\left(\frac{\theta}{2}\right), w_2 = \sqrt{c_0}\cosh\left(\frac{\theta}{2}\right), \text{ for } w_1^2 < w_2^2$$

$$w_1 = \sqrt{c_0}\cosh\left(\frac{\theta}{2}\right), w_2 = \sqrt{c_0}\sinh\left(\frac{\theta}{2}\right), \text{ for } w_1^2 > w_2^2,$$

where θ parameterises points along the dynamics trajectory represented by $w_2^2 - w_1^2 = \pm c_0$ (Saxe et al., 2013). Note that with this change of coordinates we obtain

$$\begin{aligned}w &= c_0\cosh\left(\frac{\theta}{2}\right)\sinh\left(\frac{\theta}{2}\right) \\ &= c_0\left(\frac{e^{\frac{\theta}{2}} + e^{-\frac{\theta}{2}}}{2}\right)\left(\frac{e^{\frac{\theta}{2}} - e^{-\frac{\theta}{2}}}{2}\right) \\ &= \frac{c_0}{2}\left(\frac{e^{\theta} - e^{-\theta}}{2}\right) \\ &= \frac{c_0}{2}\sinh(\theta),\end{aligned}$$

so that

$$dw = \frac{c_0}{2}\cosh(\theta)d\theta.$$

Similarly,

$$\begin{aligned}
w_2^2 + w_1^2 &= c_0\cosh^2\left(\frac{\theta}{2}\right) + c_0\sinh^2\left(\frac{\theta}{2}\right) \\
&= c_0\left(\frac{e^{\frac{\theta}{2}} + e^{-\frac{\theta}{2}}}{2}\right)^2 + c_0\left(\frac{e^{\frac{\theta}{2}} - e^{-\frac{\theta}{2}}}{2}\right)^2 \\
&= \frac{c_0}{4}\left(e^{\theta} + 2 + e^{-\theta} + e^{\theta} - 2 + e^{-\theta}\right) \\
&= c_0\left(\frac{e^{\theta} + e^{-\theta}}{2}\right) \\
&= c_0\cosh(\theta)
\end{aligned}$$

Plugging these results into the update for w given in (1), yields

$$\frac{\tau c_0\cosh(\theta)}{2}\frac{d\theta}{dt} = \left(\lambda - \frac{c_0}{2}\sinh(\theta)(\lambda + \varepsilon)\right)c_0\cosh(\theta),$$

and as a result,

$$\tau\frac{d\theta}{dt} = \lambda\left(2 - \beta\sinh(\theta)\right),$$

where $\beta = c_0\left(1 + \frac{\varepsilon}{\lambda}\right)$. To solve for t, we write

$$t = \int_{\theta_0}^{\theta_f}\frac{\tau}{\lambda\left(2 - \beta\sinh(\theta)\right)}d\theta$$

and integrate:

$$t = \frac{\tau}{\zeta\lambda}\left[\ln\left(\frac{\zeta + \beta + 2\tanh(\frac{\theta}{2})}{\zeta - \beta - 2\tanh(\frac{\theta}{2})}\right)\right]_{\theta_0}^{\theta_f}$$

where $\zeta = \sqrt{\beta^2 + 4}$ and initial parameter value $\theta_0 = \sinh^{-1}(2w/c_0)$. Let $\delta_0 = \tanh\left(\frac{\theta_0}{2}\right)$ and $\delta_f = \tanh\left(\frac{\theta_f}{2}\right)$, then

$$t = \frac{\tau}{\lambda\zeta}\ln\frac{\left(\zeta + \beta + 2\delta_f\right)\left(\zeta - \beta - 2\delta_0\right)}{\left(\zeta - \beta - 2\delta_f\right)\left(\zeta + \beta + 2\delta_0\right)},$$

so that

$$e^{\lambda\zeta t/\tau} = \frac{\left(\zeta + \beta + 2\delta_f\right)\left(\zeta - \beta - 2\delta_0\right)}{\left(\zeta - \beta - 2\delta_f\right)\left(\zeta + \beta + 2\delta_0\right)}.$$

Multiplying by the denominator, expanding, and defining $E = e^{\lambda\zeta t/\tau}$, we obtain

$$\begin{aligned}
&-2E\delta_f\left(\zeta + \beta + 2\delta_0\right) \\
&+ E\left(\zeta^2 + 2\zeta\delta_0 - \beta^2 - 2\beta\delta_0\right) \\
&= 2\delta_f\left(\zeta - \beta - 2\delta_0\right) \\
&+ \left(\zeta^2 - 2\zeta\delta_0 - \beta^2 - 2\beta\delta_0\right),
\end{aligned}$$

which yields

$$\begin{aligned}
&\delta_f\left((1 - E)\left(2\beta + 4\delta_0\right) - 2(E + 1)\zeta\right) \\
&= (1 - E)\left(\zeta^2 - \beta^2 - 2\beta\delta_0\right) - 2(1 + E)\zeta\delta_0.
\end{aligned}$$

Solving for $\theta_f(t)$, we obtain the hyperbolic parameter equation

$$\theta_f(t) = 2\tanh^{-1}\left[\frac{(1 - E)\left(\zeta^2 - \beta^2 - 2\beta\delta\right) - 2(1 + E)\zeta\delta}{(1 - E)\left(2\beta + 4\delta\right) - 2(1 + E)\zeta}\right]$$

where $\delta = \tanh\left(\frac{\theta_0}{2}\right)$. Using

$$w(t) = \frac{c_0}{2}\sinh\left(\theta_t\right),$$

(where $\theta_t = \theta_f(t)$) to track the weight trajectory gives equation (5) in the paper.

C. Learning dynamics for linear WDAEs

We derive the expression for the learning dynamics of a linear WDAE as presented in (7) in the paper. Reconstruction loss with weight decay gives the scalar loss associated with an eigenvalue λ as

$$\ell_\gamma = \frac{\lambda}{2\tau}(1 - w_2w_1)^2 + \frac{N\gamma}{2\tau}(w_1^2 + w_2^2),$$

where γ is the penalty parameter that controls the amount of regularisation that is being applied. The update equations for the weights then follow as

$$\begin{aligned}
\tau\frac{d}{dt}w_1 &= w_2(\lambda - w_2w_1\lambda) - N\gamma w_1 \\
\tau\frac{d}{dt}w_2 &= w_1(\lambda - w_2w_1\lambda) - N\gamma w_2,
\end{aligned}$$

assuming the initial $w_2 = w_1$ (which holds approximately for small initial values), we have for $w = w_2w_1$ that

$$\begin{aligned}
\tau\frac{d}{dt}w &= 2w(\lambda - w\lambda) - 2N\gamma w \\
&= 2w(\lambda - N\gamma - w\lambda).
\end{aligned}$$

Thus,

$$\begin{aligned}
t &= \int_{w_0}^{w_f}\frac{\tau}{2w(\lambda - N\gamma - w\lambda)}dw \\
&= \frac{\tau}{2}\left[\frac{\ln(w) - \ln(\lambda - N\gamma - w\lambda)}{\lambda - N\gamma}\right]_{w_0}^{w_f} \\
&= \frac{\tau}{2(\lambda - N\gamma)}\ln\left(\frac{w_f(\lambda - N\gamma - w_0\lambda)}{w_0(\lambda - N\gamma - w_f\lambda)}\right).
\end{aligned}$$

Then solving for w_f gives

$$w_f(t) = \frac{\xi E_\gamma}{E_\gamma - 1 + \xi/w_0},$$

where $E_\gamma = e^{2\xi t/\tau}$ and $\xi = (1 - N\gamma/\lambda)$.

D. Optimal learning rates

We derive expressions for the optimal learning rates for linear DAEs and WDAEs as presented in (8) in the paper. First, consider the expected scalar DAE loss

$$\ell_\varepsilon = \frac{\lambda}{2\tau}(1 - w_2 w_1)^2 + \frac{\varepsilon}{2\tau}(w_2 w_1)^2.$$

The Hessian of ℓ_ε is given by

$$H = \begin{bmatrix} \frac{\partial^2 \ell_\varepsilon}{\partial w_1^2} & \frac{\partial^2 \ell_\varepsilon}{\partial w_1 w_2} \\ \frac{\partial^2 \ell_\varepsilon}{\partial w_2 w_1} & \frac{\partial^2 \ell_\varepsilon}{\partial w_2^2} \end{bmatrix},$$

where

$$\frac{\partial^2 \ell_\varepsilon}{\partial w_1^2} = \frac{w_2^2}{\tau}(\lambda + \varepsilon),$$
$$\frac{\partial^2 \ell_\varepsilon}{\partial w_2^2} = \frac{w_1^2}{\tau}(\lambda + \varepsilon),$$
$$\frac{\partial^2 \ell_\varepsilon}{\partial w_1 w_2} = \frac{\partial^2 \ell_\varepsilon}{\partial w_2 w_1} = \frac{2 w_2 w_1}{\tau}(\lambda + \varepsilon) - \frac{\lambda}{\tau}.$$

Now, if we assume $w_2 = w_1$, and let $a = \frac{\partial^2 \ell_\varepsilon}{\partial w_1^2} = \frac{\partial^2 \ell_\varepsilon}{\partial w_2^2}$ and $b = \frac{\partial^2 \ell_\varepsilon}{\partial w_2 w_1}$, the eigenvalues for the Hessian be can computed as

$$\begin{aligned} \det(H - \lambda_H I) &= \left| \begin{bmatrix} a - \lambda_H & b \\ b & a - \lambda_H \end{bmatrix} \right| \\ &= (a - \lambda_H)^2 - b^2, \end{aligned}$$

which gives $\lambda_H = a - b$ or $\lambda_H = a + b$. The second order update for a single weight w at time t is then given by

$$w^{t+1} = w^t - \left(\frac{\partial \ell_\varepsilon}{\partial w^t} \right) / \lambda_H,$$

where the maximum λ_H, is when $w_2 = w_1 = 1$, such that

$$\begin{aligned} \lambda_H &= \frac{1}{\tau}(\lambda + \varepsilon) + \frac{2}{\tau}(\lambda + \varepsilon) - \frac{\lambda}{\tau} \\ &= \frac{2\lambda + 3\varepsilon}{\tau}. \end{aligned}$$

Therefore, the optimal learning rate is

$$\alpha_\varepsilon = 1/\lambda_H = \frac{\tau}{2\lambda + 3\varepsilon}.$$

For WDAEs with penalty parameter γ, a very similar derivation gives

$$\alpha_\gamma = \frac{\tau}{2\lambda + \gamma}.$$

Taking the ratio of the optimal DAE rate to that for the WDAE gives

$$R = \frac{\alpha_\varepsilon}{\alpha_\gamma} = \frac{2\lambda + \gamma}{2\lambda + 3\varepsilon}.$$

E. Equivalent scalar solutions

In Section 4 of the paper, the DAE fixed point solution is shown to be

$$w_\varepsilon^* = \frac{\lambda}{\lambda + \varepsilon}.$$

Now if $w = w_2 w_1$ and $w_2 = w_1$, then for WDAE we have that the scalar loss is given by

$$\ell_\gamma = \frac{\lambda}{2\tau}(1 - w)^2 + \frac{\gamma}{\tau} w,$$

and

$$\frac{\partial \ell_\gamma}{\partial w} = -\frac{\lambda}{\tau}(1 - w) + \frac{\gamma}{\tau}.$$

Setting the above equal to zero and solving gives

$$w_\gamma^* = 1 - \gamma/\lambda.$$

To obtain the value of γ for which the two fixed points are equal, we set $w_\gamma^* = w_\varepsilon^*$ and solve for γ to find

$$\gamma = \frac{\lambda\varepsilon}{\lambda + \varepsilon}.$$

F. Estimated dynamics for nonlinear networks

The dynamics for the nonlinear networks trained in Figure 6 in the paper were estimated using the following approach. First, compute

$$\Sigma_{xx} = \sum_{i=1}^{N} \mathbf{x}_i \mathbf{x}_i^T = V \Lambda V^T,$$

using an eigen-decomposition giving eigenvalues $\lambda_j, j = 1, ..., D$. Then at regular intervals compute

$$\hat{\Sigma}_{xx}(t) = \sum_{i=1}^{N} \mathbf{x}_i \hat{\mathbf{x}}_i(t)^T,$$

where $\hat{\mathbf{x}}(t)$ is the estimated reconstruction of input at time t generated by the autoencoder network. Finally, using the following rotation to obtain the diagonal matrix

$$\hat{\Lambda}(t) = V^T \hat{\Sigma}_{xx}(t) V,$$

where the diagonal contains the estimated eigenvalues $\hat{\lambda}_j(t)$, we can compute an estimate for the identity mapping associated with each eigenvalue as $\hat{\lambda}_j(t)/\lambda_j \in [0, 1]$.

G. Learning dynamics for tanh autoencoder networks

We investigated the dynamics of learning for nonlinear AEs, WDAEs and DAEs, using tanh activations.

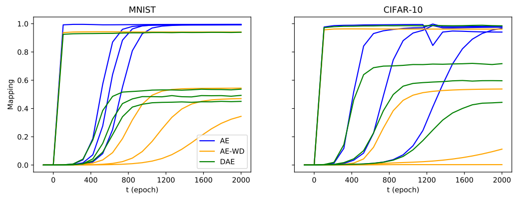

Figure 1. Learning dynamics for nonlinear networks using tanh activation. AE (blue), WDAE (orange) and DAE (green). **Left**: MNIST **Right**: CIFAR-10.

Figure 1 shows the dynamics for these networks trained on MNIST ($N = 50000$) and CIFAR-10 ($N = 30000$) with equal learning rates. For the DAE, the input was corrupted using sampled Gaussian noise with mean zero and $\sigma^2 = 2$. For the WDAE, the amount of weight decay was set to $\gamma = 0.0045$. During the course of training, the identity mapping associated with each eigenvalue was estimated using the approach described in Section F, at equally spaced intervals of size 100 epochs.

References

Saxe, A. M., McClelland, J. L., and Ganguli, S. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv:1312.6120*, 2013.