TTIC 31230, Fundamentals of Deep Learning

David McAllester, Winter 2020

Rate-Distortion Autoencoders (RDAs)

Noisy Channel RDAs

Gaussian Variational Autoencoders (Gaussian VAEs)

Rate-Distortion Autoencoders
(Image Compression)

We compress a continuous signal y to a bit string $\tilde{z}_\Phi(y)$.

We decompress $\tilde{z}_\Phi(y)$ to $y_\Phi(\tilde{z}_\Phi(y))$.

We can then define a rate-distortion loss.

$$\mathcal{L}(\Phi) = E_{y \sim \mathrm{Pop}}\ |\tilde{z}_\Phi(y)| + \lambda \mathrm{Dist}(y, y_\Phi(\tilde{z}_\Phi(y)))$$

Common Distortion Functions

$$\Phi^* = \underset{\Phi}{\text{argmin}} \;\; E_{y \sim \text{Pop}} \, |\tilde{z}_\Phi(y)| + \lambda \text{Dist}(y, y_\Phi(\tilde{z}_\Phi(y)))$$

It is common to take

$$\text{Dist}(y, \hat{y}) = ||y - \hat{y}||^2 \qquad\qquad (L_2)$$

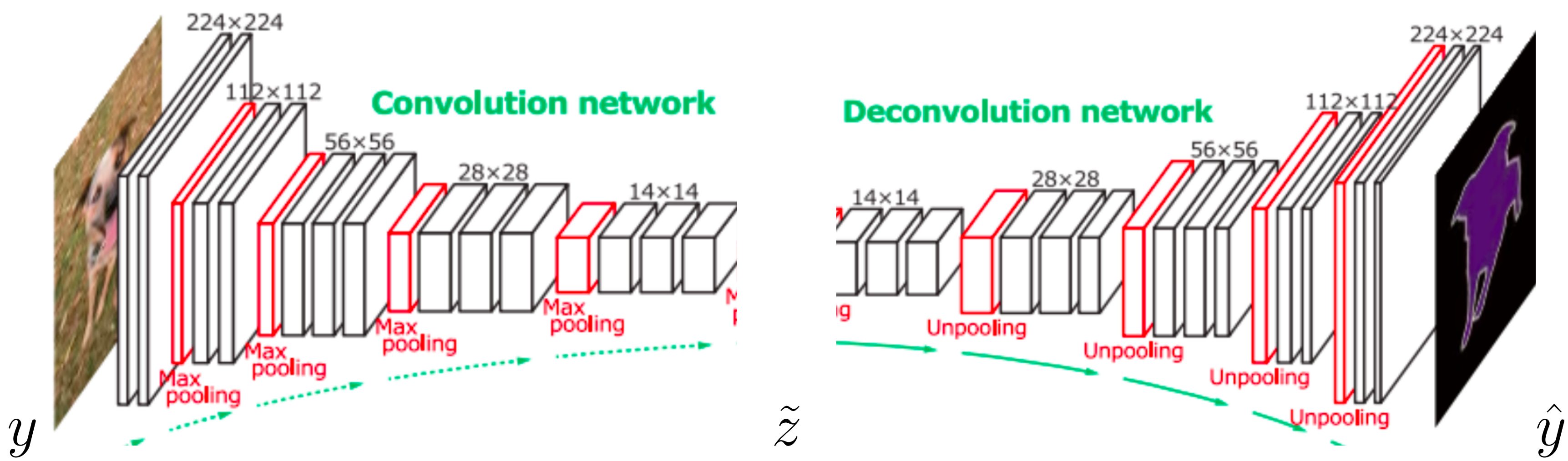
or

$$\text{Dist}(y, \hat{y}) = ||y - \hat{y}||_1 \qquad\qquad (L_1)$$

CNN-based Image Compression

These slides are loosely based on

End-to-End Optimized Image Compression, Balle, Laparra, Simoncelli, ICLR 2017.



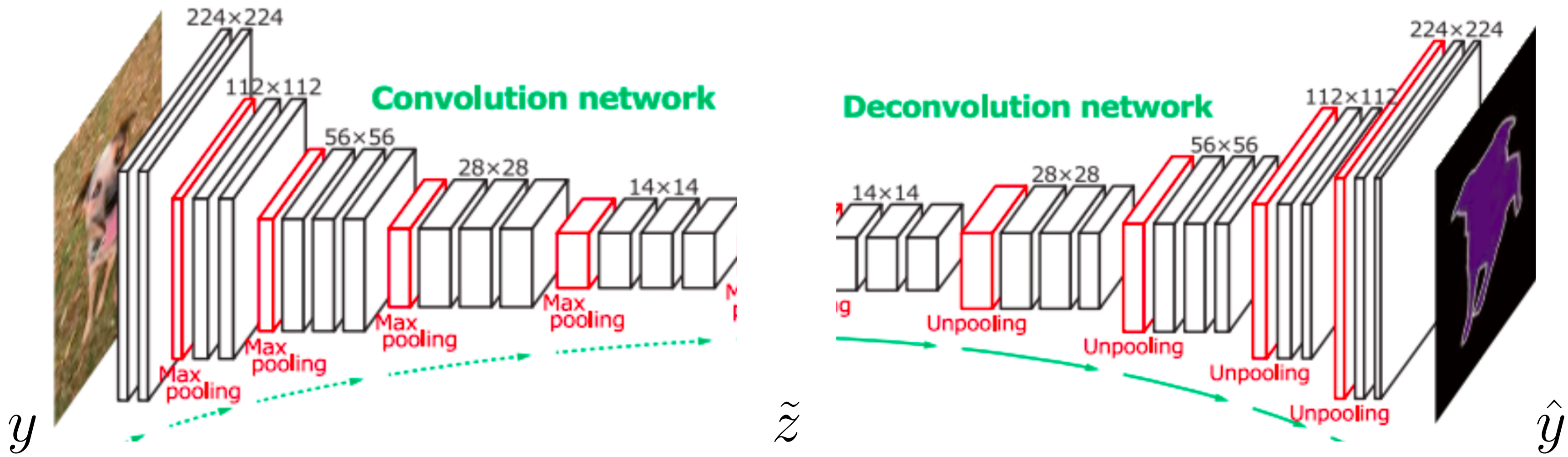
Rounding a Tensor

Take $z_\Phi(y)$ can be a layer in a CNN applied to image y. $z_\Phi(y)$ can have with both spatial and feature dimensions.

Take $\tilde{z}_\Phi(y)$ to be the result of rounding each component of the continuous tensor $z_\Phi(y)$ to the nearest integer.

$$\tilde{z}_\Phi(y)[x, y, i] = \lfloor z_\Phi(y)[x, y, i] + 1/2 \rfloor$$

Increasing Spatial Dimension in Decoding



Increasing Spatial Dimension in Decoding (Deconvolution)

To increase spatial dimension we use 4 times the desired output the features.

$$L'_{\ell+1}[x, y, i] = \sigma\left(W[\Delta X, \Delta Y, J, i]\ L'_{\ell}[x + \Delta X, y + \Delta Y, J]\right)$$

We then reshape $L'_{\ell+1}[X, Y, I]$ to $L'_{\ell+1}[2X, 2Y, I/4]$.

Rounding is not Differentiable

$$\Phi^* = \operatorname*{argmin}_{\Phi} \; E_{y \sim \mathrm{Pop}} \, |\tilde{z}_\Phi(y)| + \lambda \mathrm{Dist}(y, y_\Phi(\tilde{z}_\Phi(y)))$$

Because of rounding, $\tilde{z}_\Phi(y)$ is discrete and the gradients are zero.

We will train using a differentiable approximation.

Rate: Replacing Code Length with Differential Entropy

$$\mathcal{L}_{\text{rate}}(\Phi) = E_{y\sim\text{Pop}}\, |\tilde{z}_\Phi(y)|$$

Recall that $\tilde{z}_\Phi(y)$ is a rounding of a continuous encoding $z_\Phi(y)$.

We approximate the code length after rounding using a differentiable function of the value before rounding.

$$|\tilde{z}_\Phi(y)| \approx \sum_{x,y,i} \log_2(z_\Phi(y)[x,y,i])^+$$

This continuous value can be interpreted as a "differential entropy" with $p(z) \propto 1/z$ for some finite range of z.

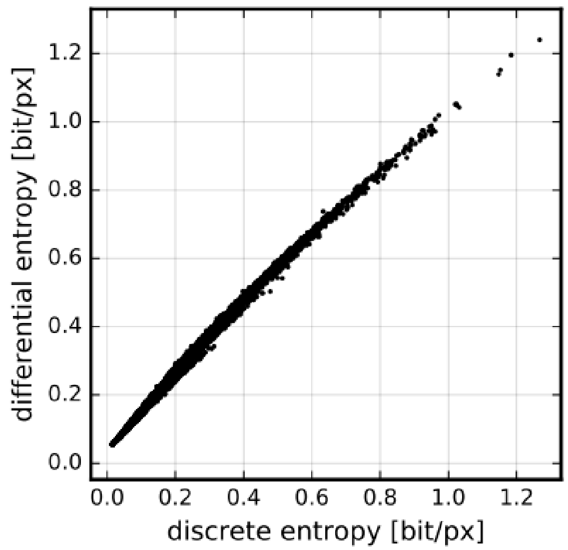
Distortion: Replacing Rounding with Noise

We can make distortion differentiable by modeling rounding as the addition of noise.

$$\begin{aligned}\mathcal{L}_{\text{dist}}(\Phi) &= E_{y\sim\text{Pop}}\,\text{Dist}(y, y_\Phi(\tilde{z}_\Phi(y)))\\ &\approx E_{y,\epsilon}\,\text{Dist}(y,\; y_\Phi(z_\Phi(y)+\epsilon))\end{aligned}$$

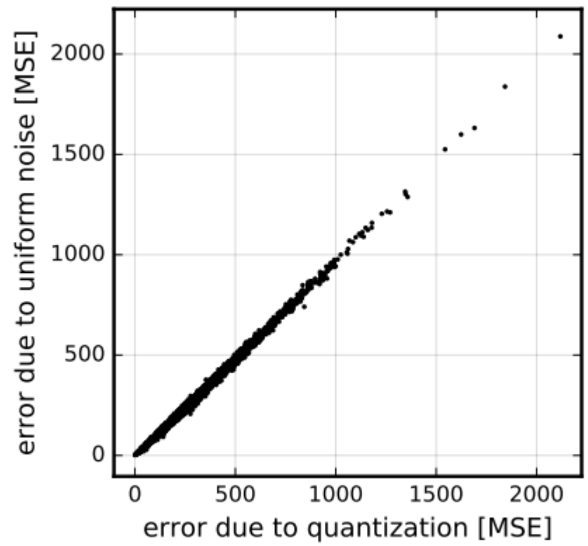
Here ϵ is a noise vector each component of which is drawn uniformly from $(-1/2, 1/2)$.

Rate: Differential Entropy vs. Discrete Entropy



Each point is a rate for an image measured in both differential entropy and discrete entropy. The size of the rate changes as we change the weight λ.

Distortion: Noise vs. Rounding



Each point is a distortion for an image measured in both a rounding model and a noise model. The size of the distortion changes as we change the weight λ.

JPEG at 4283 bytes or .121 bits per pixel

JPEG, 4283 bytes (0.121 bit/px), PSNR: 24.85 dB/29.23 dB, MS-SSIM: 0.8079

JPEG 2000 at 4004 bytes or .113 bits per pixel

JPEG 2000, 4004 bytes (0.113 bit/px), PSNR: 26.61 dB/33.88 dB, MS-SSIM: 0.8860

Deep Autoencoder at 3986 bytes or .113 bits per pixel

Proposed method, 3986 bytes (0.113 bit/px), PSNR: 27.01 dB/34.16 dB, MS-SSIM: 0.9039

Noisy-Channel RDAs

The image compression case study training was based on a differentiable loss which can be written in the form

$$\Phi^* = \underset{\Phi}{\mathrm{argmin}} \left(E_y \; - \ln p_\Phi(z_\Phi(y))\right) + \lambda E_{y,\epsilon} \; \mathrm{Dist}(y, y_\Phi(z_\Phi(y) + \epsilon))$$

In a rate-distortion auto-encoder we will measure rate directly on continuous variables.

The problem is that the first term — the cross entropy term — should be viewed as being infinite — there are infinitely many bits in a real number.

Mutual Information Replaces Cross Entropy

We replace

$$\Phi^* = \underset{\Phi}{\mathrm{argmin}} \left(E_y \; - \ln p_\Phi(z_\Phi(y))\right) + \lambda E_{y,\epsilon} \; \mathrm{Dist}(y, y_\Phi(z_\Phi(y)+\epsilon))$$

by

$$\tilde{z} \; = \; z_\Phi(y) + \epsilon \;\; (\epsilon \text{ is random noise — typically Gaussian})$$

$$\Phi^* \; = \; \underset{\Phi}{\mathrm{argmin}} \;\; I(y, \tilde{z}) + \lambda E_{y,\epsilon} \; \mathrm{Dist}(y, y_\Phi(\tilde{z}))$$

By the channel capacity theorem $I(y, \tilde{z})$ is the **rate** of information transfer from y to $\tilde{z}$.

Mutual Information Replaces Cross Entropy

$$I(y, \tilde{z}) = E_{y,\tilde{z}} \ln \frac{p(y, \tilde{z})}{p(\tilde{z})p(y)}$$

$$= E_{y,\tilde{z}} \ln \frac{p(\tilde{z} \mid z_\Phi(y))}{p(\tilde{z})}$$

A Variational Bound

$$p(\tilde{z}) = E_y\, p(\tilde{z} \mid z_\Phi(y))$$

We cannot compute $p(\tilde{z})$.

Instead we have a model $p_\Phi(\tilde{z})$.

The model corresponds to the "code" we are using to approximate the true distribution $p(\tilde{z})$.

A Variational Bound

$$
\begin{aligned}
I(y,\tilde{z}) &= E_{y,\tilde{z}}\ \ln \frac{p(\tilde{z} \mid z_\Phi(y))}{p(\tilde{z})} \\
&= E_{y,\tilde{z}}\ \ln \frac{p(\tilde{z} \mid z_\Phi(y))}{p_\Phi(\tilde{z})} + E_{\tilde{z}}\ \ln \frac{p_\Phi(\tilde{z})}{p(\tilde{z})} \\
&= E_{y,\tilde{z}}\ \ln \frac{p(\tilde{z} \mid z_\Phi(y))}{p_\Phi(\tilde{z})} - KL(p(\tilde{z}), p_\Phi(\tilde{z})) \\
&\leq E_{y,\tilde{z}}\ \ln \frac{p(\tilde{z} \mid z_\Phi(y))}{p_\Phi(\tilde{z})}
\end{aligned}
$$

Cross MI

$$I(y, \tilde{z}) \leq E_{y,\tilde{z}} \ln \frac{p(\tilde{z} \mid y)}{p_\Phi(\tilde{z})}$$

We might call the right hand side "cross MI" written $I(y, \tilde{z}, p_\Phi)$.

Cross MI, unlike true MI, is measurable.

A Fundamental Equation for the Continuous Case

$$\tilde{z} \;=\; z_\Phi(y) + \epsilon \;\; (\epsilon \text{ is random noise — typically Gaussian})$$

$$\Phi^* \;=\; \operatorname*{argmin}_\Phi E_{y,\tilde{z}} \;\; \ln \frac{p(\tilde{z}|z_\Phi(y))}{p_\Phi(\tilde{z})} + \lambda \mathrm{Dist}(y, y_\Phi(\tilde{z}))$$

Gaussian Noisy-Channel RDA

$$\Phi^* = \underset{\Phi}{\mathrm{argmin}}\ E_{y\sim \mathrm{Pop}} \left(\begin{array}{c} KL(p_\Phi(\tilde{z}|y), p_\Phi(\tilde{z})) \\ \\ +\lambda\ E_{\tilde{z}\sim p_\Phi(\tilde{z}|y)}\ \mathrm{Dist}(y,\ y_\Phi(\tilde{z})) \end{array} \right)$$

$$p_\Phi(\tilde{z}[i] \mid y) = \mathcal{N}(z_\Phi(y)[i], \sigma^\epsilon_\Phi(y)[i]))$$

$$p_\Phi(\tilde{z}[i]) = \mathcal{N}(\mu_\Phi[i], \sigma^z_\Phi[i])$$

$$\mathrm{Dist}(y, \hat{y}) = ||y - \hat{y}||^2$$

Closed Form KL-Divergence

$$KL(p_\Phi(\tilde{z}|y), p_\Phi(\tilde{z}))$$

$$= \sum_i \frac{\sigma^\epsilon_\Phi(y)[i]^2 + (z_\Phi(y)[i] - \mu_\Phi[i])^2}{2\sigma^z_\Phi[i]^2} + \ln \frac{\sigma^z_\Phi[i]}{\sigma^\epsilon_\Phi(y)[i]} - \frac{1}{2}$$

Standardizing $p_\Phi(z)$

$$KL(p_\Phi(\tilde{z}|y), p_\Phi(\tilde{z}))$$

$$= \sum_i \frac{\sigma^\epsilon_\Phi(y)[i]^2 + (z_\Phi(y)[i] - \mu_\Phi[i])^2}{2\sigma^z_\Phi[i]^2} + \ln\frac{\sigma^z_\Phi[i]}{\sigma^\epsilon_\Phi(y)[i]} - \frac{1}{2}$$

$$KL(p_{\Phi'}(\tilde{z}|y), \mathcal{N}(0, I))$$

$$= \sum_i \frac{\sigma^\epsilon_{\Phi'}(y)[i]^2 + z_{\Phi'}(y)[i]^2}{2} + \ln\frac{1}{\sigma^\epsilon_{\Phi'}(y)[i]} - \frac{1}{2}$$

Standardizing $p_\Phi(z)$

$$KL_\Phi = \sum_i \frac{\sigma^\epsilon_\Phi(y)[i]^2 + (z_\Phi(y)[i] - \mu_\Phi[i])^2}{2\sigma^z_\Phi[i]^2} + \ln\frac{\sigma^z_\Phi[i]}{\sigma^\epsilon_\Phi(y)[i]} - \frac{1}{2}$$

$$KL_{\Phi'} = \sum_i \frac{\sigma^\epsilon_{\Phi'}(y)[i]^2 + z_{\Phi'}(y)[i]^2}{2} + \ln\frac{1}{\sigma^\epsilon_{\Phi'}(y)[i]} - \frac{1}{2}$$

Setting Φ' so that

$$\begin{aligned} z_{\Phi'}(y)[i] &= (z_\Phi(y)[i] - \mu_\Phi[i])/\sigma^z_\Phi[i] \\ \sigma^\epsilon_{\Phi'}(y)[i] &= \sigma^\epsilon_\Phi(y)[i]/\sigma^z_\Phi[i] \end{aligned}$$

gives $KL(p_\Phi(\tilde{z}|y), p_\Phi(\tilde{z})) = KL(p_{\Phi'}(\tilde{z}|y), \mathcal{N}(0, I))$.

Standardizing $p_\Phi(z)$

Without loss of generality the Gaussian noisy channel RDA becomes.

$$\Phi^* = \operatorname*{argmin}_\Phi \; E_{y\sim \mathrm{Pop}} \left(\begin{array}{c} KL(p_\Phi(\tilde{z}|y), \mathcal{N}(0, I)) \\ \\ +\lambda E_{\tilde{z}\sim p_\Phi(\tilde{z}|y)} \,\mathrm{Dist}(y, y_\Phi(\tilde{z})) \end{array} \right)$$

Reparameterization Trick for Optimizing Distortion

$$p_\Phi(\tilde{z}[i]|y) = \mathcal{N}(z_\Phi(y)[i], \sigma_\Phi[i])$$

$$E_{\tilde{z}\sim p_\Phi(\tilde{z}|y)}\ ||y - y_\Phi(\tilde{z})||^2$$

$$= \ E_{\epsilon\sim\mathcal{N}(0,I)}\ \tilde{z}[i] = z_\Phi(y)[i] + \sigma_\Phi(y)[i]\epsilon[i]; \quad ||y - y_\Phi(\tilde{z})||^2$$

Sampling

Sample $\tilde{z} \sim \mathcal{N}(0, I)$ and compute $y_\Phi(\tilde{z})$

[Alec Radford]

Summary: Rate-Distortion

RDA: y continuous, $\tilde{z}$ a bit string,

$$\Phi^* = \operatorname*{argmin}_{\Phi} E_{y\sim\mathrm{Pop}} \; |\tilde{z}_\Phi(y)| + \lambda \mathrm{Dist}(y, y_\Phi(\tilde{z}_\Phi(y)))$$

Gaussian RDA: $\tilde{z} = z_\Phi(y) + \sigma_\Phi(y) \odot \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I)$

$$\Phi^* = \operatorname*{argmin}_{\Phi} E_{y\sim\mathrm{Pop}} \left(\begin{array}{c} KL(p_\Phi(\tilde{z}|y), \mathcal{N}(0,I)) \\ \\ +\lambda E_{\tilde{z}\sim p_\Phi(\tilde{z}|y)} \; \mathrm{Dist}(y, y_\Phi(\tilde{z})) \end{array} \right)$$

Issue: Do we expect compression to yield useful features?

END