

Building a working autoencoder – I Network tricks - A

D.A. Forsyth

University of Illinois at Urbana Champaign

Housekeeping

- There is a great deal of housekeeping:
 - Build your own from scratch, and suffer (but learn)
 - and it won't work all that well (missing tricks)
- Use an API (list in notes)
- Build from someone else's code (Github, Huggingface)

Missing tricks:

- Scales and color
- Blurred outputs and skip connections
- Bad gradients and residual connections
- Optimization tricks

Scale and color

- Input scales:
 - Helpful to scale, offset pixel values so that mean is zero and standard deviation is one
 -
- Output scales:
 - Real pixels are in range $[0, 1]$
 - How do we force output to be like this?
 - Options:
 - map value to range
 - penalize over/under
 - bit of both

Value to range

- Terrible idea:

You might think that using

$$f(x) = \text{ReLU}(x) - \text{ReLU}(x - 1)$$

would be a good idea (because it maps any x to the range $[0, 1]$). In fact, it is a terrible idea, because once the output is outside that range, there is no gradient to push it back into the range (Section 16.3.1; **exercises**).

- **NO GRADIENT when you need it!**

Value to range

- Sigmoid:

$$\text{sigmoid}(x) = \frac{e^x}{1 + e^x}$$

- but this creates gradient problems when output close to zero or one

- Tanh:

- $x \rightarrow [-1, 1]$ and easy to get to $[0, 1]$
- similar gradient problems

Penalties

A second strategy is to accept the output of the convolutional layer, but penalize values that appear outside the range you want with a loss term. For example, apply the loss

$$L_{ud}(x) = x^2 \mathbb{I}_{[x < 0]} + (x - 1)^2 \mathbb{I}_{[x > 1]}.$$

Combination

It is often useful to mix these strategies. For example, a final layer that applies

$$f(x) = a(\tanh(x) + b)$$

will map x to the range $(a(b - 1), a(b + 1))$. If you choose $b < 1$ and $a > 1/(b + 1)$, then the output can be below zero (but not much) and above one (but not much). Further, the gradients won't be too small at zero or one. You can then push the outputs to be in that range with a penalty term.