On smoothing and differentiation

October 8, 2017

The goal of this note is to derive and understand the formula

$$\nabla_\theta \mathbb{E}_{x\sim N(0,1)}\left[f(\theta+\sigma x)\right] = \mathbb{E}_{x\sim N(0,1)}\left[\frac{x}{\sigma} f(\theta+\sigma x)\right]. \tag{0.1}$$

The derivation involves change of variables, known in this context as the *reparameterization trick*, followed by the application of the *likelihood ratio trick* and a subsequent inverse change of variables.

1 Reparameterization trick

One can derive the reparameterization trick

$$\mathbb{E}_{x\sim N(0,1)}\left[f(\theta+\sigma x)\right] = \mathbb{E}_{y\sim N(\theta,\sigma^2)}\left[f(y)\right] \tag{1.1}$$

via the change of variables $y = \theta + \sigma x$ with $dy = \sigma dx$ as

$$\int f(\theta+\sigma x) N(x|0,1)\, dx = \int f(y) N\left(\frac{y-\theta}{\sigma}|0,1\right)\frac{dy}{\sigma} = \int f(y)\sigma N(y|\theta,\sigma^2)\frac{dy}{\sigma}.$$

One could also derive Formula (1.1) intuitively, since shifting a function is equivalent to shifting a Gaussian in the opposite direction.

2 Smoothing operator

We can compactly rephrase (1.1) using the smoothing operator S which acts on functions f by convolving with a Gaussian kernel as

$$S f_\theta = S_\theta f. \tag{2.1}$$

The dependence on the parameters θ can be shifted from the function to the kernel.

3 Derivative operator

If we want to compute a gradient of a smoothed function, we can push the gradient inside the expectation

$$\nabla_\theta \mathbb{E}_{x\sim N(0,1)}\left[f(\theta+\sigma x)\right] = \mathbb{E}_{x\sim N(0,1)}\left[\nabla_\theta f(\theta+\sigma x)\right].$$

A compact notation for this is

$$D\left[Sf_\theta\right] = S\left[Df_\theta\right], \tag{3.1}$$

where D is the derivative operator. As we see, *smoothing and differentiation commute.*

4 Derivative of smoothing

Formula (3.1) allows us to compute the derivative $D\left[Sf_\theta\right]$ of a smoothed function Sf_θ by smoothing the derivative Df_θ of the initial function f_θ. Somewhat surprisingly, shifting the dependence on the parameters from the function f to the smoothing operator S using (2.1), we can compute $D\left[Sf_\theta\right]$ even if we can't differentiate f_θ,

$$D\left[Sf_\theta\right] = D\left[S_\theta f\right].$$

We have to differentiate the smoothing operator instead. This is reminiscent of Heisenberg vs Schrödinger picture in quantum mechanics.

5 Likelihood ratio trick

The likelihood ratio trick

$$\nabla_\theta \mathbb{E}_{y\sim N(\theta,\sigma^2)}\left[f(y)\right] = \mathbb{E}_{y\sim N(\theta,\sigma^2)}\left[\nabla_\theta \ln N(y|\theta,\sigma^2) f(y)\right]$$

allows us to switch the order of differentiation and parameterized smoothing

$$D\left[S_\theta f\right] = S_\theta\left[D\ln N_\theta f\right]. \tag{5.1}$$

However, D and S_θ do not commute this time, since we are getting an extra factor $\ln N_\theta$ in front of f.

6 Differentiation vs multiplication

Shifting the dependence on θ in (5.1) back from S to f, we obtain

$$S_\theta\left[fD\ln N_\theta\right] = S\left[f_\theta \frac{x}{\sigma}\right].$$

If you followed the whole chain of reasoning, you now see the following remarkable identity

$$S\left[Df_\theta\right] = S\left[\frac{x}{\sigma} f_\theta\right]. \tag{6.1}$$

In words, it means that *differentiation is equivalent to multiplication by $\frac{x}{\sigma}$ when followed by smoothing.* The particular form of the multiplicative factor follows from using Gaussian noise. Other formulas in this note do not rely on any particular choice of the sampling distribution.

7 Conclusion

Finally, switching the order of smoothing and differentiation in the left-hand side of (6.1), which is allowed by (3.1), we obtain (0.1) advertised in the beginning,

$$D\left[Sf_\theta\right] = S\left[\frac{x}{\sigma}f_\theta\right]. \tag{7.1}$$

Differentiation of a smoothed function Sf_θ is equivalent to smoothing of a surrogate function $\frac{x}{\sigma}f_\theta$.

8 Future work

The most symmetric formulation seems to be (6.1). Differentiation can be replaced by multiplication inside smoothing. This sounds reminiscent of the Fourier transform. Is there any relation? Another question is how reliable this formula is in practice. On the left, one averages values of the derivative, whereas on the right, on averages function values with some weights. It sounds like finite-difference approximation. Is it true then that the left-hand side is more precise in practice, given a finite sample size?