What is `grad_outputs` in the `torch.autograd.grad` function?

Casey Barkan

July 28, 2024

It is surprisingly difficult to find a precise definition of the kwarg `grad_outputs` in PyTorch's `autograd.grad` function. I've found many questions and discussions about `grad_outputs` on various online forums, but none have a clear mathematical explanation of what `grad_outputs` is and how it's used by `autograd.grad`. The purpose of these notes is to provide a clear mathematical explanation.

Consider the following tensors:

1. Tensor X with components X_{bq}, where $b \in \{0, ..., B-1\}$ and $q \in \{0, ..., Q-1\}$.
2. Tensor $Y = Y(X)$ with components $Y_{b'n}$, where $b' \in \{0, ..., B'-1\}$ and $n \in \{0, ..., N-1\}$. Often $B = B'$ (hence the the same letter B), which occurs when the B dimension is the batch dimension.
3. Tensor V with the same shape as Y, which will be the value of `grad_outputs`. Often, `V=torch.ones_like(Y)`.

Let `res` be the tensor defined by the code:

```
res = torch.autograd.grad(Y,X,grad_outputs=v)[0]
```

The components of `res` are given by

$$\texttt{res}_{bq} = \sum_{b'=0}^{B'-1} \sum_{n=0}^{N-1} V_{b'n} \frac{\partial Y_{b'n}}{\partial X_{bq}} \tag{1}$$

When dealing with batches, $B = B'$ and $Y_{bn}(X) = f_n(X_{b:})$. For PINNs, f_n may be the model (which, for each batch b, takes the input $X_{b:}$), and Y is the tensor containing the model's output for every batch. In this case,

$$\frac{\partial Y_{b'n}}{\partial X_{bq}} = \delta_{b'b} \frac{\partial}{\partial X_{bq}} f_n(X_{b:}) \tag{2}$$

Therefore,

$$\texttt{res}_{bq} = \sum_{n=0}^{N-1} V_{bn} \frac{\partial}{\partial X_{bq}} f_n(X_{b:}) \tag{3}$$

For a typical PINN, f is a scalar-valued function (i.e. $N = 1$) and `v=torch.ones_like(Y)`. Hence, in this case,

$$\texttt{res}_{bq} = \frac{\partial}{\partial X_{bq}} f(X_{b:}) \tag{4}$$

which is what is needed for the ODE ($Q = 1$) or PDE ($Q > 1$) loss.