Differentiating Vector- and Matrix-Valued Functions

Szymon Rusinkiewicz

COS 302, Fall 2020

PRINCETON UNIVERSITY

Generalizing Functions…

Functions of scalars, vectors, matrices … *returning* scalars, vectors, matrices

- Function of a scalar, returning a scalar: $\mathbb{R} \longrightarrow \mathbb{R}$
  - Example: $f(x) = ax + b$
- Function of a scalar, returning a vector: $\mathbb{R} \longrightarrow \mathbb{R}^n$
  - Example: $\boldsymbol{f}(x) = x\boldsymbol{v}$
- Function of a vector, returning a scalar: $\mathbb{R}^n \longrightarrow \mathbb{R}$
  - Example: $f(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$
- Function of a vector, returning a vector: $\mathbb{R}^n \longrightarrow \mathbb{R}^n$
  - Example: $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$
- Function of a vector, returning a matrix: $\mathbb{R}^n \longrightarrow \mathbb{R}^{n \times n}$
  - Example: $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{x}^\top$
- Many other possibilities: function of a matrix, etc.

Generalizing Functions…and Taking Their Derivatives

- Function of a scalar, returning a scalar: $\mathbb{R} \longrightarrow \mathbb{R}$
  - Example: $f(x) = ax + b$ $\longrightarrow$ Ordinary derivative $\frac{df}{dx} : \mathbb{R} \to \mathbb{R}$
- Function of a scalar, returning a vector: $\mathbb{R} \longrightarrow \mathbb{R}^n$
  - Example: $\boldsymbol{f}(x) = x\boldsymbol{v}$ $\longrightarrow$ Vector-valued derivative $\frac{d\boldsymbol{f}}{dx} : \mathbb{R} \to \mathbb{R}^n$
- Function of a vector, returning a scalar: $\mathbb{R}^n \longrightarrow \mathbb{R}$
  - Example: $f(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ $\longrightarrow$ Gradient $\nabla f(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^{1\times n}$
- Function of a vector, returning a vector: $\mathbb{R}^n \longrightarrow \mathbb{R}^n$
  - Example: $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$ $\longrightarrow$ Jacobian $\nabla \boldsymbol{f}(\boldsymbol{x})$ or $\mathrm{J}\big(\boldsymbol{f}(\boldsymbol{x})\big) : \mathbb{R}^n \to \mathbb{R}^{n\times n}$
- Function of a vector, returning a matrix: $\mathbb{R}^n \longrightarrow \mathbb{R}^{n\times n}$
  - Example: $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{x}^\top$ $\longrightarrow$ Generalized Jacobian $\nabla \boldsymbol{F}(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^{n\times n\times n}$

Generalizing Functions…and Taking Their Derivatives

In general, if f is a function

$$f : \mathbb{R}^{(\textit{input shape})} \longrightarrow \mathbb{R}^{(\textit{output shape})}$$

then its generalized derivative will be a function

$$\nabla f : \mathbb{R}^{(\textit{input shape})} \longrightarrow \mathbb{R}^{(\textit{output shape})\times(\textit{input shape})}$$

where the extra dimensions on output correspond to taking partial derivatives with respect to all the input dimensions.

Tensors

So what if we end up with e.g. an $n \times n \times n$ object?

- *Tensors* are multidimensional generalizations of scalars, vectors, matrices.
- For our purposes, represented as multidimensional arrays of numbers.
- The number of indices in the shape can be called the *order*, *degree*, (confusingly) *dimension*, or (even more confusingly) *rank* of the tensor.
  - Example: Take a function of a vector, returning a matrix, and differentiate it. The resulting $n \times n \times n$ beastie is a degree-3 or 3^{rd}-order tensor.

Tensors in Python

- NumPy arrays can represent tensors
  - `A = np.zeros((5, 6, 7))` → `A.shape == (5, 6, 7)`
- Transpose can take a permutation of dimensions
  - `B = np.transpose(A, (2, 0, 1))` → `B.shape == (7, 5, 6)`
- Careful if using `np.matmul` or `np.dot` for tensor multiplication — `np.tensordot` lets you explicitly specify axes to sum over
  - `C = np.tensordot(A, B, (2, 0))` → `C.shape == (5, 6, 5, 6)`
  - `D = np.tensordot(A, B, ([2,1], [0,2]))` → `D.shape == (5, 5)`

Tensors and Differentiation

$$\nabla f : \mathbb{R}^{(\textit{input shape})} \longrightarrow \mathbb{R}^{(\textit{output shape})\times(\textit{input shape})}$$

- If you take more advanced math, you'll learn that the "dimensions" of tensors behave in two different ways: *covariant* and *contravariant*.
- We won't go into that here, except to note that the dimensions arising from differentiation always behave "transposed".
  - For example, the gradient of a scalar function of a vector is a *row* vector.
  - Intimate connection to *directional derivatives*: multiplying a gradient by a *direction* $\boldsymbol{d}$ (an object of the input shape) gives you derivative of the output in that direction:

$$D_{\boldsymbol{d}} f(\boldsymbol{x}) \text{ or } \nabla_{\boldsymbol{d}} f(\boldsymbol{x}) = \nabla f(\boldsymbol{x})\, \boldsymbol{d}$$

    where the last dimension(s) of ∇f are dotted against $\boldsymbol{d}$.

Preliminaries

Before we get into specific examples of these generalized derivatives, let's review which rules from single-variable calculus still work:

- Derivative of a *constant* of any shape is 0
- Derivative of the variable with respect to which we're differentiating is 1, or the *identity* of the appropriate shape
- Derivative of a *sum* is the sum of derivatives
- Derivative of a *scalar multiple* is the constant times the derivative
- *Chain rule* works, but order might matter: $\frac{d}{dx} f\big(g(x)\big) = f'\big(g(x)\big)\, g'(x)$
- *Product rule* requires care about dimensions and transposes (stay tuned!)

Function of a Scalar, Returning a Vector

Simple…

$$f(x) = \begin{bmatrix} 3x + 42 \\ \sin x \end{bmatrix}$$

$$\frac{df}{dx} = \begin{bmatrix} 3 \\ \cos x \end{bmatrix}$$

Can also consider functions written as scalar/vector products:

$$f(x) = x\boldsymbol{v}$$

$$\frac{df}{dx} = \begin{bmatrix} \frac{d}{dx}(x\, v_1) \\ \frac{d}{dx}(x\, v_2) \\ \vdots \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \end{bmatrix} = \boldsymbol{v}$$

Function of a Vector, Returning a Scalar

This is the ordinary *gradient*, which is a *row* vector of partial derivatives:

$$f\left(\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}\right) = x_1^3 + x_1 x_2 + 42x_3$$

$$\begin{aligned} \nabla f &= \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \frac{\partial f}{\partial x_3} \end{bmatrix} \\ &= \begin{bmatrix} 3x_1^2 + x_2 & x_1 & 42 \end{bmatrix} \end{aligned}$$

Directional Derivative

$$\nabla f = \begin{bmatrix} 3x_1^2 + x_2 & x_1 & 42 \end{bmatrix}$$

How does f change with an infinitesimal step in direction $\boldsymbol{d} = \begin{bmatrix} 0.6 \\ 0 \\ 0.8 \end{bmatrix}$?

$$\nabla_{\boldsymbol{d}} f = \begin{bmatrix} 3x_1^2 + x_2 & x_1 & 42 \end{bmatrix} \begin{bmatrix} 0.6 \\ 0 \\ 0.8 \end{bmatrix} = 1.8x_1^2 + 0.6x_2 + 33.6$$

This is a *scalar*—the same shape as the output of f.

Directional Derivative

- What if we had done this in the previous case, where we had a function of a *scalar*, returning a *vector*?

$$f(x) = x\boldsymbol{v}$$
$$\frac{df}{dx} = \boldsymbol{v}$$

- Multiplying by a (scalar) infinitesimal step in x in "direction" 1, we get just $\boldsymbol{v}$.
- This is a (column) vector — the same shape as the output of f.

Function of a Vector, Returning a Scalar

Let's try a dot product!

$$
\begin{aligned}
f(\boldsymbol{x}) = \boldsymbol{v} \cdot \boldsymbol{x} &= \sum_i v_i x_i \\
\nabla f &= \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots \end{bmatrix} \\
&= \begin{bmatrix} v_1 & v_2 & \cdots \end{bmatrix} \\
&= \boldsymbol{v}^\top
\end{aligned}
$$

But $\boldsymbol{v} \cdot \boldsymbol{x} = \boldsymbol{v}^\top \boldsymbol{x} = \boldsymbol{x}^\top \boldsymbol{v}$, so we have the following:

$$\nabla(\boldsymbol{v}^\top \boldsymbol{x}) = \boldsymbol{v}^\top \quad \text{and} \quad \nabla(\boldsymbol{x}^\top \boldsymbol{v}) = \boldsymbol{v}^\top$$

Function of a Vector, Returning a Scalar

Next interesting case:

$$f(\boldsymbol{x}) = \boldsymbol{x} \cdot \boldsymbol{x} = \boldsymbol{x}^\top \boldsymbol{x} = \|\boldsymbol{x}\|^2 = \sum_i x_i^2$$

$$\begin{aligned} \nabla f &= \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots \end{bmatrix} \\ &= \begin{bmatrix} 2x_1 & 2x_2 & \cdots \end{bmatrix} \\ &= 2\boldsymbol{x}^\top \end{aligned}$$

Note the analogy to $\frac{d}{dx}x^2 = 2x$, but we need the transpose to get the output shape right.

Function of a Vector, Returning a Scalar

Even more interesting:

$$f(\boldsymbol{x}) = \|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^\top \boldsymbol{x}} = \sqrt{\sum x_i^2}$$

Applying the chain rule:

$$\begin{aligned}
\nabla f &= \frac{1}{2}\left(\boldsymbol{x}^\top \boldsymbol{x}\right)^{-\frac{1}{2}} \nabla\left(\boldsymbol{x}^\top \boldsymbol{x}\right) \\
&= \frac{2\boldsymbol{x}^\top}{2\sqrt{\boldsymbol{x}^\top \boldsymbol{x}}} \\
&= \frac{\boldsymbol{x}^\top}{\|\boldsymbol{x}\|}
\end{aligned}$$

Directional Derivative

$$\nabla \|\boldsymbol{x}\| = \frac{\boldsymbol{x}^\top}{\|\boldsymbol{x}\|}$$

As $\boldsymbol{x}$ changes by an infinitesimal step in direction $\boldsymbol{d}$,

$$\nabla_{\boldsymbol{d}} \|\boldsymbol{x}\| = \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \cdot \boldsymbol{d}$$

Intuitive: if $\boldsymbol{d}$ is in the direction of $\boldsymbol{x}$, change in $\|\boldsymbol{x}\|$ is 1 times the step size, etc.

Function of a Vector, Returning a Vector

Let's move on to a Jacobian:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}) &= \boldsymbol{M}\boldsymbol{x} \\ \nabla \boldsymbol{f} &= \begin{bmatrix} \frac{\partial \boldsymbol{f}}{\partial x_1} & \frac{\partial \boldsymbol{f}}{\partial x_2} & \cdots \end{bmatrix} \end{aligned}$$

The only terms in $\boldsymbol{M}\boldsymbol{x}$ involving x_i come from the i^{th} column of $\boldsymbol{M}$, so:

$$\frac{\partial \boldsymbol{f}}{\partial x_1} = \begin{bmatrix} M_{11} \\ M_{21} \\ \vdots \end{bmatrix}, \quad \frac{\partial \boldsymbol{f}}{\partial x_2} = \begin{bmatrix} M_{12} \\ M_{22} \\ \vdots \end{bmatrix}, \text{ etc.}$$

Function of a Vector, Returning a Vector

- Stitching everything together,

$$\begin{aligned}\nabla(\boldsymbol{M}\boldsymbol{x}) &= \left[\begin{pmatrix} M_{11} \\ M_{21} \\ \vdots \end{pmatrix} \quad \begin{pmatrix} M_{12} \\ M_{22} \\ \vdots \end{pmatrix} \quad \cdots \quad \right] \\ &= \boldsymbol{M}\end{aligned}$$

- Special case: $\nabla(\boldsymbol{x}) = \nabla(\boldsymbol{I}\boldsymbol{x}) = \boldsymbol{I}$
- This reinforces our intuition that differentiating any constant thing times $\boldsymbol{x}$ gives just that constant, whether it's a scalar, vector, matrix, etc.

Function of a Vector, Returning a Matrix

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{v}^\top = \begin{bmatrix} x_1 v_1 & x_1 v_2 & \cdots \\ x_2 v_1 & x_2 v_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Just apply the rules, and watch the tensor appear!

$$\begin{aligned} \nabla \boldsymbol{F}(\boldsymbol{x}) &= \begin{bmatrix} \frac{\partial F}{\partial x_1} & \frac{\partial F}{\partial x_2} & \cdots \end{bmatrix} \\ &= \begin{bmatrix} \begin{bmatrix} v_1 & v_2 & \cdots \\ 0 & 0 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} & \begin{bmatrix} 0 & 0 & \cdots \\ v_1 & v_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} & \cdots \end{bmatrix} \end{aligned}$$

Function of a Vector, Returning a Matrix

$$\nabla \boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \boldsymbol{v}^\top & \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix} \boldsymbol{v}^\top & \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \end{bmatrix} \boldsymbol{v}^\top & \cdots \end{bmatrix}$$

At this point, it might be tempting to factor out the $\boldsymbol{v}^\top$. But be careful!

This object is an $n \times n \times n$ tensor, and if you factor it into an $n \times 1 \times n$ tensor and a $1 \times n$ vector, you have to remember which dimensions should be multiplied!

Generalizing Product Rule

- As we just saw, tensor multiplication can get confusing. This complicates cleanly stating a generalized product rule.
- But, let's derive a rule for vector-vector (dot) products:

$$\nabla(\boldsymbol{v} \cdot \boldsymbol{w}) = \nabla\left(\sum_i v_i w_i\right)$$

where $\boldsymbol{v}$ and $\boldsymbol{w}$ are both potentially functions of $\boldsymbol{x}$.

- Writing out the partial derivatives,

$$\nabla(\boldsymbol{v} \cdot \boldsymbol{w}) = \begin{bmatrix} \frac{\partial(\sum v_i w_i)}{\partial x_1} & \frac{\partial(\sum v_i w_i)}{\partial x_2} & \dots & \end{bmatrix}$$

Generalizing Product Rule

- Because v_i and w_i are just *scalars*, the product rule works normally:

$$\frac{\partial\left(\sum v_i w_i\right)}{\partial x_1} = \sum_i \left(v_i \frac{\partial w_i}{\partial x_1} + \frac{\partial v_i}{\partial x_1} w_i\right)$$

- Applying the distributive rule, we get

$$\nabla(\boldsymbol{v} \cdot \boldsymbol{w}) = \boldsymbol{v} \cdot (\nabla \boldsymbol{w}) + (\nabla \boldsymbol{v}) \cdot \boldsymbol{w}$$

- Or, in matrix notation,

$$\nabla(\boldsymbol{v}^\top \boldsymbol{w}) = \boldsymbol{v}^\top \nabla \boldsymbol{w} + \boldsymbol{w}^\top \nabla \boldsymbol{v}$$

Bilinear Form

Let's apply our newly-derived knowledge!

$$\begin{aligned} f(\boldsymbol{x}) &= \boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} \\ \nabla f &= \boldsymbol{x}^\top \nabla(\boldsymbol{M}\boldsymbol{x}) + (\boldsymbol{M}\boldsymbol{x})^\top \nabla \boldsymbol{x} \\ &= \boldsymbol{x}^\top \boldsymbol{M} + \boldsymbol{x}^\top \boldsymbol{M}^\top \boldsymbol{I} \\ &= \boldsymbol{x}^\top (\boldsymbol{M} + \boldsymbol{M}^\top) \end{aligned}$$

Note the similarity to $\frac{d}{dx}(ax^2) = 2ax$.

The Grand Finale: Least Squares

- We've mentioned before that our methods for solving overdetermined linear systems of the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ minimize a least-squares residual:

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$$

- Let's apply the methods we've learned to find the $\boldsymbol{x}$ that minimizes this, by taking the derivative (gradient) and setting it equal to 0.

The Grand Finale: Least Squares

- Applying the chain rule:

$$\nabla \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 = 2\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\| \, \nabla \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|$$

- And again (order matters!):

$$= 2\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\| \, \frac{(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top}{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|} \nabla(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})$$

- And computing a final Jacobian:

$$= 2(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top \boldsymbol{A}$$

The Grand Finale: Least Squares

- To check, let's derive this a different way:

$$
\begin{aligned}
\|\boldsymbol{Ax}-\boldsymbol{b}\|^2 &= (\boldsymbol{Ax}-\boldsymbol{b})^\top(\boldsymbol{Ax}-\boldsymbol{b}) \\
&= \boldsymbol{x}^\top A^\top \boldsymbol{Ax} - \boldsymbol{x}^\top A^\top \boldsymbol{b} - \boldsymbol{b}^\top \boldsymbol{Ax} + \boldsymbol{b}^\top \boldsymbol{b} \\
\nabla\|\boldsymbol{Ax}-\boldsymbol{b}\|^2 &= \boldsymbol{x}^\top\left(A^\top A + (A^\top A)^\top\right) - \boldsymbol{b}^\top A - \boldsymbol{b}^\top A + 0 \\
&= 2\boldsymbol{x}^\top A^\top A - 2\boldsymbol{b}^\top A \\
&= 2(\boldsymbol{x}^\top A^\top - \boldsymbol{b}^\top)A \\
&= 2(\boldsymbol{Ax}-\boldsymbol{b})^\top A
\end{aligned}
$$

The Grand Finale: Least Squares

- Setting the gradient equal to a row vector of zeros:

$$2(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^\top \boldsymbol{A} = \mathbf{0}^\top$$

- Transposing and dividing by 2:

$$\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) = \mathbf{0}$$

- And finally, rearranging:

$$\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^\top \boldsymbol{b}$$