Análisis Matemático para Inteligencia Artificial

Martín Errázquin (merrazquin@fi.uba.ar)

Especialización en Inteligencia Artificial

Diferenciación automática

Regla de la Cadena en forma matricial

@ escalar TODO → $\mathbb{R} \to \mathbb{R}$

$y = f(x)$ $z = g(y)$ $\underbrace{\frac{\partial z}{\partial x}}_{\mathbb{R}} = \underbrace{\frac{\partial z}{\partial y}}_{\mathbb{R}} \cdot \underbrace{\frac{\partial y}{\partial x}}_{\mathbb{R}}$

Sea $f(x_1(s,t), x_2(s,t))$

$\begin{pmatrix} s \\ t \end{pmatrix} \to \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to f$

$\underbrace{J_{\vec{x}}(s,t)}_{2\times 2}$ $\underbrace{\nabla_f(x_1,x_2)}_{1\times 2}$

$$\frac{\partial f}{\partial s} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial s} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial s}$$

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial t}$$

$\vec{F}: \mathbb{R}^n \to \mathbb{R}^m$ $\frac{dF}{d\vec{x}} \in \mathbb{R}^{n\times m}$

$J_{\vec{x}}(s,t)$

Y luego

$$\nabla_f(s,t) = \frac{df}{d(s,t)} = \frac{df}{dx}\frac{dx}{d(s,t)} = \underbrace{\overbrace{\begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix}}^{\nabla_f(x_1,x_2)}}_{1\times 2} \underbrace{\overbrace{\begin{bmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial s} & \frac{\partial x_2}{\partial t} \end{bmatrix}}^{J_{\vec{x}}(s,t)}}_{2\times 2} \in \mathbb{R}^{1\times 2}$$

Nota: Observar que si bien $\frac{df}{d(s,t)} \in \mathbb{R}^2$, hay que pasar por $\frac{dx}{d(s,t)} \in \mathbb{R}^{2\times 2}$. Podría interesarnos *compilar* el cálculo analítico para ser más eficientes en cómputo.

$! \; f: \mathbb{R}^{m\times n} \to \mathbb{R}^{p\times q} \Rightarrow \frac{\partial f}{\partial X} \in \mathbb{R}^{m\times n\times p\times q}$

kernel fusing

Derivadas y código

Situación: parte de mi código involucra, para un cierto valor de entrada x, calcular una función `f(x)` y su derivada `df(x)` en ese punto.

$f : \mathbb{R}^n \to \mathbb{R}$

```
def f(x):
    return x**2
```

Si cambia f(x) tambien debe cambiar df(x)

```
def df(x):
    return 2*x
```

- Cálculo 100 % analítico
  - (+) máxima eficiencia de cómputo
  - (−) máxima inflexibilidad de código

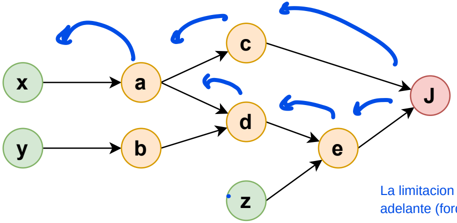
```
def df(x, h=1e-3):
    return (f(x+h) - f(x-h))/2h
```

- Cálculo 100 % numérico
  - (−) máxima ineficiencia de cómputo
  - (+) máxima flexibilidad de código

Se aproxima la derivada mediante un calculo numerico (si cambia f(x) no cambia (df(x))), pero es ineficiente computacionalmente

Grafo de cómputo

- Uso específico: composición de funciones de diversas entradas y me importa la derivada de la salida "final" respecto de cada entrada.
- Punto medio entre eficiencia y flexibilidad a través de *bloques diferenciables*.
- Cada bloque debe poder calcular la derivada de su salida respecto de todas sus entradas (en gral. analítica).
- Un "orquestador" se ocupa de ir aplicando regla de la cadena.
- ¡Cada bloque podría ser (adentro) una función compuesta!



$$\frac{\partial J}{\partial a} = \frac{\partial J}{\partial c} \cdot \frac{\partial c}{\partial a} + \frac{\partial J}{\partial e} \frac{\partial e}{\partial d} \cdot \frac{\partial d}{\partial a}$$

$$\frac{\partial J}{\partial x} = \frac{\partial J}{\partial a} \cdot \frac{\partial a}{\partial x}$$

La limitacion del Grafo de computo es que si o si tiene que hacer una pasada hacia adelante (fordward) y luego hacer la pasada hacia atras (backward)

Ejemplo (simplificado) de código

```
class Product(BaseOperation):
  def f(self, x, y):
      return x * y
```

$f(x,y) = x \cdot y$

```
  def df(self, x, y):
      return (y, x)
```

$\nabla_f(x,y) = (y, x)$

```
class Cosine(BaseOperation):
  def f(self, x):
      return np.cos(x)
```

$f(x) = \cos(x)$

```
  def df(self, x):
      return -np.sin(x)
```

$\nabla f(x) = (-\mathrm{sen}(x))$

$(*) = 7{,}4 \cdot 2 \cdot 2 + 29{,}6 \cdot 1 = 29{,}6 + 29{,}6 = 59{,}2$

Sea $z(x, y) = x^2 \cdot e^{x+\cos(y)}$, quiero $\nabla_z(2, \frac{\pi}{2})$.

x, y

$a = x^2$

$b = \cos(y)$

$c = x + b$

$d = \exp(c)$

$z = a \cdot d$

$\frac{\partial a}{\partial x} = 2x$

$\frac{\partial b}{\partial y} = -\sin(y)$

$\nabla_c(x, b) = (1, 1)$

$\frac{\partial d}{\partial c} = \exp(c)$

$\nabla_z(a, d) = (d, a)$

$x = 2, \; y = \pi/2$

$a = 2^2 = 4$

$b = \cos(\pi/2) = 0$

$c = 2 + 0 = 2$

$d = e^2 \approx 7{,}4$

$z = 4 \cdot 7{,}4 \approx 29{,}6$

$dx = da \cdot \frac{\partial a}{\partial x} + dc \cdot \frac{\partial c}{\partial x} = (*)$

$dy = db \cdot \frac{\partial b}{\partial y} = 29{,}6 \cdot -\sin(\frac{\pi}{2}) = -29{,}6$

$da = dz \cdot \frac{\partial z}{\partial a} = 1 \cdot 7{,}4 = 7{,}4$

$db = dc \cdot \frac{\partial c}{\partial b} = 29{,}6 \cdot 1 = 29{,}6$

$dc = dd \cdot \frac{\partial d}{\partial c} = 4 \cdot 7{,}4 = 29{,}6$

$dd = dz \cdot \frac{\partial z}{\partial d} = 1 \cdot 4 = 4$

$dz = 1$

$\frac{\partial z}{\partial m} \doteq dm$

$z(2, \frac{\pi}{2}) = 29{,}6$

$\nabla_z(2, \frac{\pi}{2}) = (59{,}2 \;, -29{,}6)$