Automatic Differentiation

Exact Derivatives to Machine Precision

John Martin Jr.

December 11, 2015

Introduction

2015 Survey

Automatic differentiation in machine learning: a survey[2]

What It's Not

Let $f(x) = \log(x) + \alpha x$.

Manual Differentiation

$$\frac{d}{dx} f(x) = \frac{1}{x} + \alpha$$

Numerical Differentiation

$$\frac{d}{dx} f(x) = \frac{f(x_k + \Delta x) - f(x_k)}{\Delta x}$$

Symbolic Differentiation

$$\frac{d}{dx} f(x) = \frac{d}{dx}(\log(x)) \left(\frac{d}{dx} x\right) + \left(\frac{d}{dx} \alpha\right) x + \alpha \left(\frac{d}{dx} x\right)$$

Big Idea

All numerical derivatives are compositions of a finite set of elementary operations for which derivatives are known.

- Apply chain rule
- Constant factor of overhead
- Handles control flow statements that can't be symbolically expressed

Example

Let $f(x) = \log(x) + \alpha x$. We know $f'(x) = \frac{1}{x} + \alpha$.

Evaluation trace

$$\begin{aligned} v_{-1} &\leftarrow x_1 = 5 \\ v_0 &\leftarrow x_2 = \alpha = 1 \\ v_1 &\leftarrow \log(v_{-1}) = 1.609 \\ v_2 &\leftarrow v_{-1} \cdot v_0 = 5 \\ v_3 &\leftarrow v_1 + v_2 = 6.609 \\ f(x) &\leftarrow v_3 = 6.609 \end{aligned}$$

Derivative trace

$$\begin{aligned} v'_{-1} &\leftarrow x'_1 = 1 \\ v'_0 &\leftarrow x'_2 = 0 \\ v'_1 &\leftarrow (1/v_{-1})v'_{-1} = 0.2 \\ v'_2 &\leftarrow v'_{-1}v_0 + v_{-1}v'_0 = 1 \\ v'_3 &\leftarrow v'_1 + v'_2 = 1.2 \\ f'(x) &\leftarrow v'_3 = 1.2 \end{aligned}$$

Finite Difference Example

Let $f(x) = \log(x) + \alpha x$, and choose $\Delta x = 1.$. We know $f'(x) = \frac{1}{x} + \alpha$. However, in this case

$$f'(x) = \frac{f(x + \Delta x) - f(x)}{\Delta x}.$$

Evaluation trace 1

$$\begin{aligned} v_{-1} &\leftarrow x_1 = 6 \\ v_0 &\leftarrow x_2 = \alpha = 1 \\ v_1 &\leftarrow \log(v_{-1}) = 1.792 \\ v_2 &\leftarrow v_{-1} \cdot v_0 = 6 \\ v_3 &\leftarrow v_1 + v_2 = 7.792 \\ f(x + \Delta x) &\leftarrow v_3 \end{aligned}$$

Evaluation trace 2

$$\begin{aligned} v_{-1} &\leftarrow x_1 = 5 \\ v_0 &\leftarrow x_2 = \alpha = 1 \\ v_1 &\leftarrow \log(v_{-1}) = 1.609 \\ v_2 &\leftarrow v_{-1} \cdot v_0 = 5 \\ v_3 &\leftarrow v_1 + v_2 = 6.609 \\ f(x) &\leftarrow v_3 \end{aligned}$$

$$f'(x) \approx 1.183 \neq 1.2$$

AD Is Also Fast

Consider an $m \times n$ Jacobian $\mathbf{J}$ from $f : \mathbb{R}^n \to \mathbb{R}^m$.

$$\mathbf{J} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_n} \end{pmatrix}$$

- Initialize with $\mathbf{x}' \leftarrow \hat{\mathbf{e}}_i$ and $\mathbf{x} \leftarrow \boldsymbol{\alpha}$.
- One trace computes one column
- Forward accumulation computed Jacobian with n traces

Jacobian Vector Products

Initialize the algorithm with $\mathbf{x}' \leftarrow \mathbf{q}$, where $\mathbf{q}$ is a constant vector.

$$\mathbf{Jq} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_n} \end{pmatrix} \begin{pmatrix} q_1 \\ \vdots \\ q_n \end{pmatrix} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} q_1 & + \cdots + & \frac{\partial y_1}{\partial x_n} q_n \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & + \cdots + & \frac{\partial y_m}{\partial x_n} \end{pmatrix}$$

We reduce the problem to one column.

- Requires only one trace!
- No matrices needed

Dual Numbers

Define a dual number as $v + v'\epsilon$ with $\epsilon^2 = 0$ and augment our normal arithmetic so that

$$(v + v'\epsilon) + (u + u'\epsilon) = (v + u) + (v' + u')\epsilon,$$
$$(v + v'\epsilon)(u + u'\epsilon) = vu + (v'u + vu')\epsilon.$$

These mirror differentiation rules.

$$f(v + v'\epsilon) = f(v) + f'(v)v'\epsilon$$

$$f(g(v + v'\epsilon)) = f(g(v)) + f'(g(v))g'(v)v'\epsilon$$

Notice that $f(g(v + v'\epsilon))' = f'(g(v))g'(v)v'$.

Dual Numbers

$f'(x)$ is a result of augmented arithmetic operations.

$$f(g(v + v'\epsilon)) = f(g(v)) + f'(g(v))g'(v)v'\epsilon$$

- Dual arithmetic libraries exist
- Overloaded operators
- Code generators

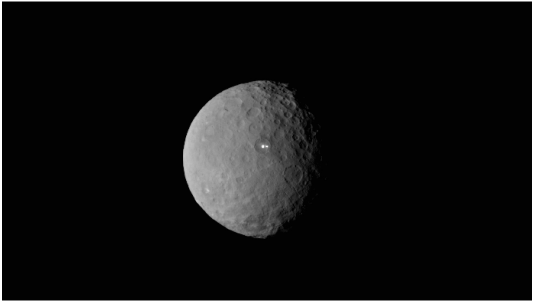
Dual Numbers Example

Consider $f(x) = \log(x) + \alpha x$. We know $f'(x) = \frac{1}{x} + \alpha$.

Evaluation trace

$$
\begin{aligned}
v_{-1} + v'_{-1}\epsilon &\leftarrow x_1 + x'_1\epsilon = 5 + \epsilon \\
v_0 + v'_0\epsilon &\leftarrow x_2 + x'_2\epsilon = \alpha + 0\epsilon \\
v_1 + v'_1\epsilon &\leftarrow \log(v_{-1} + v'_{-1}) = \log(v_{-1}) + (1/v_{-1})v'_{-1}\epsilon \\
v_2 + v'_2\epsilon &\leftarrow v_{-1}v_0 + (v'_{-1}v_0 + v_{-1}v'_0)\epsilon \\
v_3 + v'_3\epsilon &\leftarrow v_1 + v_2 \\
&= [\log(v_{-1}) + v_{-1}v_0] + \left[(1/v_{-1})v'_{-1} + v'_{-1}v_0 + v_{-1}v'_0\right]\epsilon \\
f'(x) &\leftarrow (1/v_{-1})v'_{-1} + v'_{-1}v_0 + v_{-1}v'_0 = 1.2
\end{aligned}
$$

Google's Ceres Solver



- Commemorates Gauss' least-squares model of Ceres' orbit
- Opensource C++ library[1]
- `git clone https://ceres-solver.googlesource.com/ceres-solver`

References I

[1] Sameer Agarwal, Keir Mierle, and Others.
Ceres solver.
http://ceres-solver.org.

[2] Atilim Gunes Baydin, Barak A. Pearlmutter, and Alexey Andreyevich Radul.
Automatic differentiation in machine learning: a survey.
CoRR, abs/1502.05767, 2015.