Atodiff in General

Idea: Functions in code are compositions of primitives: addition, multiplication, .., e^x, $\cos x$, ...

To compute a (partial) derivative of the composite expression we deploy chain rule.

The Jacobian of a function $f: \mathbb{R}^n \to \mathbb{R}^m$ is $m \times n$ matrix with entries

$$(Jac)_{ij} = \frac{\partial f^i}{\partial x^j}.$$

For example for a function $f: \mathbb{R}^3 \to \mathbb{R}^3$, $f(x^1, x^2, x^3) = (y^1, y^2, y^3)$

The Jacobian is

$$Jac(f) = \begin{pmatrix} \frac{\partial y^1}{\partial x^1} & \frac{\partial y^1}{\partial x^2} & \frac{\partial y^1}{\partial x^3} \\ \frac{\partial y^2}{\partial x^1} & \frac{\partial y^2}{\partial x^2} & \frac{\partial y^2}{\partial x^3} \\ \frac{\partial y^3}{\partial x^1} & \frac{\partial y^3}{\partial x^2} & \frac{\partial y^3}{\partial x^3} \end{pmatrix}$$

If we multiply the Jacobian with a vector $\vec{u} \in \mathbb{R}^n$ we get the directional derivative

$$\nabla_{\vec{u}} f = Jac(f)\,\vec{u} = \left(\frac{\partial f^i}{\partial x^j} u^j \right)$$

The Jacobian of scalar function $f: \mathbb{R}^n \to \mathbb{R}$ is its gradient ∇f.

Autodiff can be implemented in forward mode or in reverse mode

To understand the details let's look at the case of the function

$$f: \mathbb{R}^2 \to \mathbb{R}^2$$

given by

$$f(x_1, x_2) = \left(x_1 x_2 + \sin x_1,\ e^{x_1 - x_2} \right)$$

The Jacobian is:

$$Jac(f)(x_1, x_2) = \begin{pmatrix} x_2 + \cos x_1 & x_1 \\ e^{x_1 - x_2} & -e^{x_1 - x_2} \end{pmatrix}$$

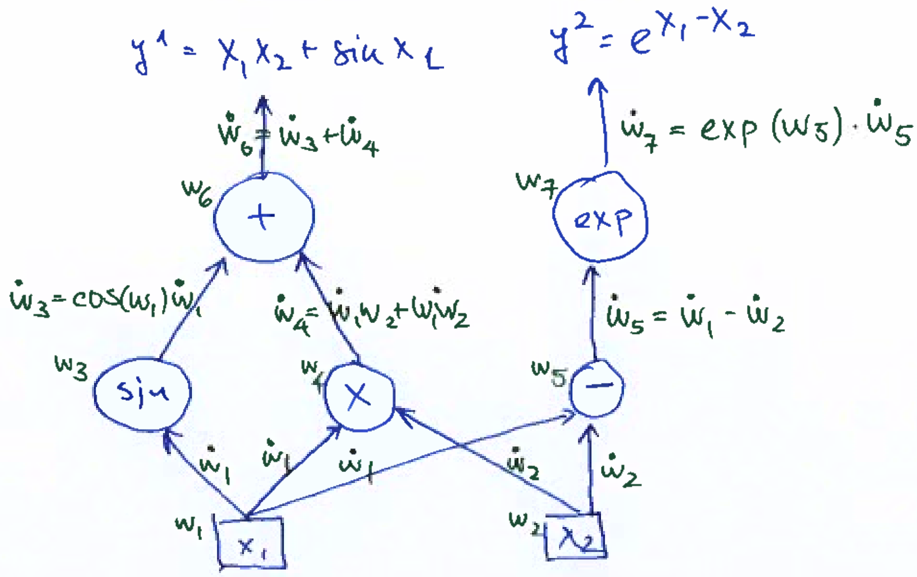
Autodiff in forward mode

In forward mode Autodiff traverses the chain rule from inside to outside. Let w_i be the output of the i'th node of the computational graph and let $\dot{w}_i = \partial w_i / \partial x$ where x is any component of the input (independent) variables. Then

$$\dot{w}_j = \sum_{j \in \{\text{predecessors of } i\}} \frac{\partial w_i}{\partial w_j} \cdot \dot{w}_j$$

where the sum is over the predecessors of i in the computational graph

ex: $f(x_1, x_2) = \left(x_1 x_2 + \sin x_1,\ e^{x_1 - x_2} \right)$



Autodiff compute the partial derivatives numerically, not symbolically. But on paper we can keep the values of x_1, x_2 as parameters. However we most specify the 'seed' derivatives $\dot{w}_1$ and $\dot{w}_2$.

With the choice $\dot{w}_1 = 1$, $\dot{w}_2 = 0$ we will compute $\partial y^1/\partial x^1$, $\partial y^2/\partial x^1$:

$$\dot{w}_6 = \dot{w}_3 + \dot{w}_4 = \cos(w_1)\,\dot{w}_1 + \dot{w}_1 w_2 + w_1 \dot{w}_2 = \cos x_1 + x_2 = \partial y^1/\partial x^1$$

$$\dot{w}_7 = \exp(w_5)\,\dot{w}_5 = \exp(w_1 - w_2)\left(\dot{w}_1 - \dot{w}_2\right) = \exp(x_1 - x_2) = \partial y^2/\partial x^1$$

With the choice $\dot{w}_1 = 0$, $\dot{w}_2 = 1$ we will compute $\partial y^1/\partial x^2$, $\partial y^2/\partial x^2$:

$$\dot{w}_6 = \dot{w}_3 + \dot{w}_4 = \cos(w_1)\,\dot{w}_1 + \dot{w}_1 w_2 + w_1 \dot{w}_2 = x_1 = \partial y^2/\partial x^1$$

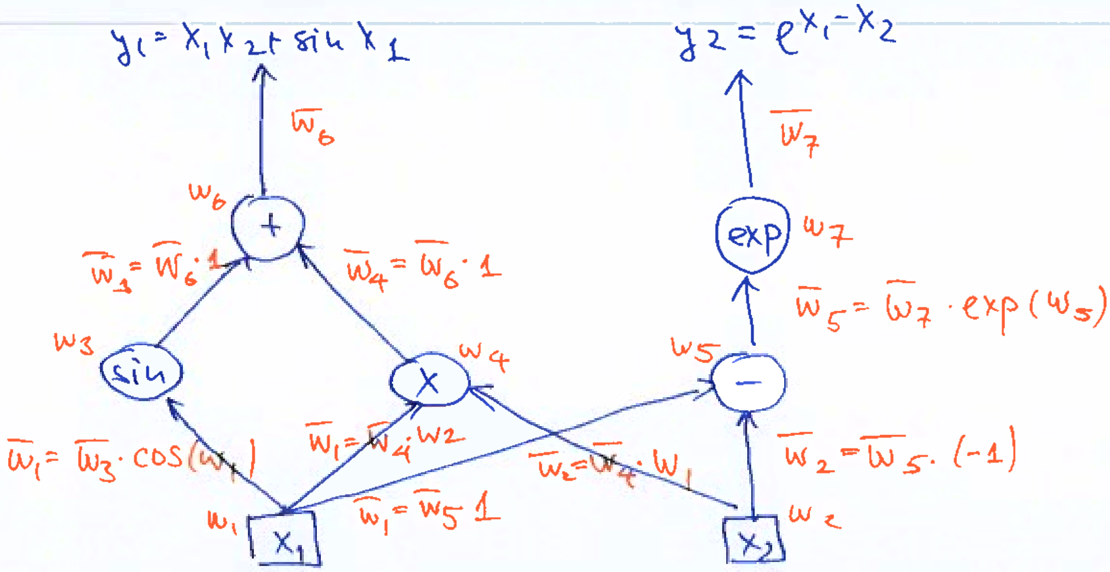
$$\dot{w}_7 = \exp(w_5)\,\dot{w}_5 = \exp(w_1 - w_2)(\dot{w}_1 - \dot{w}_2) = -\exp(x_1 - x_2) = \partial y^2/\partial x^2$$

Autodiff in reverse mode

In reverse mode the chain rule is traversed from outside to inside.
Let $\bar{w}_i = \partial y / \partial w_i$ where y is any component of the output (dependent) variables. Then

$$\bar{w}_i = \sum_{j \in \{\text{successors of } i\}} \bar{w}_j \cdot \frac{\partial w_j}{\partial w_i}$$

ex: $f(x_1, x_2) = (x_1 x_2 + \sin x_1, e^{x_1 - x_2})$



Again we will keep the values of x_1, x_2 as parameters. However we must specify the 'seed' derivatives $\bar{w}_6$ and $\bar{w}_7$.

With the choice $\bar{w}_6 = 1$ and $\bar{w}_7 = 0$ we will compute $\partial y^1/\partial x^1$, $\partial y/\partial x^2$

$$\bar{w}_1 = \bar{w}_3 \cdot \cos(w_1) + \bar{w}_4 \cdot w_2 + \bar{w}_5 \cdot 1 = \bar{w}_6 \cos(w_1) + \bar{w}_6 \cdot w_2 + \bar{w}_7 \cdot \exp(w_5)$$
$$= \cos(x_1) + x_2 = \partial y^1/\partial x^1$$
$$\bar{w}_2 = \bar{w}_4 \cdot w_1 + \bar{w}_5(-1) = \bar{w}_6 \cdot 1 + \bar{w}_7 \cdot \exp(w_5)(-1) = x_1 = \partial y/\partial x^2$$

With the choice $\bar{w}_6 = 0$ and $\bar{w}_7 = 1$ we will compute $\partial y^2/\partial x^1$, $\partial y^2/\partial x^2$

$$\bar{w}_1 = \bar{w}_3 \,\mathrm{co}(w_1) + \bar{w}_4 \cdot w_2 + \bar{w}_5 \cdot 1 = \bar{w}_6 \cos(w_1) + \bar{w}_6 w_2 + \bar{w}_7 \cdot \exp(w_5)$$
$$= \exp(x_1 - x_2) = \partial y^2/\partial x^1$$
$$\bar{w}_2 = \bar{w}_4 \cdot w_1 + \bar{w}_5(-1) = \bar{w}_6 \cdot 1 + \bar{w}_7 \exp(w_5)(-1) = -e^{x_1 - x_2} = \partial y/\partial x^2$$

Autodiff Homework

For a function $f: \mathbb{R}^n \to \mathbb{R}^m$ need:

- n passes to compute the Jacobian in forward mode
- m passes to compute the Jacobian in reverse mode.

In ML the Loss Function is $f: \mathbb{R}^n \to \mathbb{R}^1$ where n is a large number. Reverse-mode Autodiff = BackProp is used.

HW: Compute the Jacobian of $f: \mathbb{R}^3 \to \mathbb{R}^2$ given by:

$$f(x_1, x_2, x_3) = (x_1 x_2 + x_1 x_3,\ \ln(x_1 + x_2))$$

in both forward and reverse mode autodiff.