بسم الله الرحمن الرحيم

انهارده هنقفل كلام علي الباك بروب وهنخش في الديب ليرننج .. لما بنتكلم علي الدييب ليرننج هنتكلم علي حاجتين CNNs and RNNs واللي محتاجين نعرفو هو ازاي ال 2 دول بيعملولهم تمرين .. فعلياً هو الباك بروب يعني .. المهم الدكتور مسميهم هم الاتنين decision functions ... المهم HW4 هيخلص انهارده ان شاء الله .. ربنا يعين ويوفق ... الدكتور مش بيطلع الامتحانات برا .. فبينزلو paper versions بس ... في سؤال الدكتور اتسألو ...

Q&A

Q: Do I need to know Matrix Calculus to derive the backprop algorithms used in this class?

A: No. We've carefully constructed our assignments so that you do **not** need to know Matrix Calculus.

That said, it's kind of handy.

الدكتور بيقلك ان ده يعني بيجي في الإيدين كسكيلل يعني . بس اشطا ..

Matrix Calculus

Let $y, x \in \mathbb{R}$ be scalars, $\mathbf{y} \in \mathbb{R}^M$ and $\mathbf{x} \in \mathbb{R}^P$ be vectors, and $\mathbf{Y} \in \mathbb{R}^{M \times N}$ and $\mathbf{X} \in \mathbb{R}^{P \times Q}$ be matrices

Numerator

Types of Derivatives	scalar	vector	matrix
scalar	$\frac{\partial y}{\partial x}$	$\frac{\partial \mathbf{y}}{\partial x}$	$\frac{\partial \mathbf{Y}}{\partial x}$
gradient vector	$\frac{\partial y}{\partial \mathbf{x}}$	$\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$	$\frac{\partial \mathbf{Y}}{\partial \mathbf{x}}$
matrix	$\frac{\partial y}{\partial \mathbf{X}}$	$\frac{\partial \mathbf{y}}{\partial \mathbf{X}}$	$\frac{\partial \mathbf{Y}}{\partial \mathbf{X}}$

Denominator

Matrix Calculus

Types of Derivatives	scalar
scalar	$\frac{\partial y}{\partial x} = \left[\frac{\partial y}{\partial x}\right]$ 1x1 matrix
vector	$\frac{\partial y}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial y}{\partial x_1} \\ \frac{\partial y}{\partial x_2} \\ \vdots \\ \frac{\partial y}{\partial x_P} \end{bmatrix}$ List of deriv
matrix	$\frac{\partial y}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial y}{\partial X_{11}} & \frac{\partial y}{\partial X_{12}} & \cdots & \frac{\partial y}{\partial X_{1Q}} \\ \frac{\partial y}{\partial X_{21}} & \frac{\partial y}{\partial X_{22}} & \cdots & \frac{\partial y}{\partial X_{2Q}} \\ \vdots & & & \vdots \\ \frac{\partial y}{\partial X_{P1}} & \frac{\partial y}{\partial X_{P2}} & \cdots & \frac{\partial y}{\partial X_{PQ}} \end{bmatrix}$

5

Matrix Calculus

Types of Derivatives	**scalar**	**vector**
scalar	$\frac{\partial y}{\partial x} = \left[\frac{\partial y}{\partial x}\right]$	with respect to scalar $\boxed{\frac{\partial \mathbf{y}}{\partial x}} = \left[\frac{\partial y_1}{\partial x} \quad \frac{\partial y_2}{\partial x} \quad \cdots \quad \frac{\partial y_N}{\partial x}\right]$
vector	$\frac{\partial y}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial y}{\partial x_1} \\ \frac{\partial y}{\partial x_2} \\ \vdots \\ \frac{\partial y}{\partial x_P} \end{bmatrix}$	N entries $\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} & \cdots & \frac{\partial y_N}{\partial x_1} \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_N}{\partial x_2} \\ \vdots & & & \\ \frac{\partial y_1}{\partial x_P} & \frac{\partial y_2}{\partial x_P} & \cdots & \frac{\partial y_N}{\partial x_P} \end{bmatrix}$ P entries

6

Matrix Calculus

Common Vector Derivatives

Let $\frac{\partial f(\vec{x})}{\partial \vec{x}} = \nabla_x f(\vec{x})$ be the vector derivative of f, $B \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^m$

Scalar Derivative

$f(x) \rightarrow \frac{\partial f}{\partial x}$

$bx \rightarrow b$

$xb \rightarrow b$

$x^2 \rightarrow 2x$

$bx^2 \rightarrow 2bx$

Vector Derivative

$f(x) \rightarrow \frac{\partial f}{\partial \vec{x}}$

$\cancel{x^T B \rightarrow B}$

$x^T b \rightarrow b$

$x^T x \rightarrow 2x$ vector x

$x^T B x \rightarrow 2Bx$

B symmetric

7

Vector Chain Rule

$\vec{x} \in \mathbb{R}^P \quad y \in \mathbb{R} \quad u \in \mathbb{R}^N$

$$\underbrace{\frac{\partial y}{\partial \vec{x}}}_{P\times 1} = \underbrace{\left(\underbrace{\left(\underbrace{\frac{\partial y}{\partial \vec{u}}}_{N\times 1}\right)^T}_{1\times N} \underbrace{\left(\underbrace{\frac{\partial \vec{u}}{\partial \vec{x}}}_{P\times N}\right)^T}_{N\times P}\right)^T}_{P\times 1}$$

$$\left[\frac{\partial y}{\partial \vec{x}}\right]_p \qquad \left[\frac{\partial y}{\partial \vec{u}}\right]_n \qquad \left[\frac{\partial \vec{u}}{\partial \vec{x}}\right]_{pn}$$

طيب هنا احنا بنبص علي الصوره بتاعت الباك بروب ألجورذم في الجينيرال فيرجن .. عندك كمان الفورود بروب .. بتقول ان عندك فانكشن f .. اللي انت عاوز تحسباه فانت عاوز تكتب الجورذم يحسبها .. الألجورذم هيعرف directed acyclic graph .. ده اللي هو ال computational graph .. ال forward computational هيقلك visit each node in topological order اللي هو معناها visit all the children of a node before visiting the node itself .. فلفاريبل اسمو ui .. عندو انبوتس اسمها v1 ... vN هتحسب ال ui كإنها intermediate function و تخزن النواتج في النوود اللي عندك . عشان هتحتاجها في الباك بروب ... في الباك بروب انت بت initialize dy/dy ل 1 .. و بعدين تبدأ ت visit each node in reverse topological order ...

الدكتور رسم الصوره وقال ان كل ال Us فانكشن في كل ال Vs ... انت هنا في الباك بروب دايماً هت visit the parent before the children ...

Training Backpropagation

Automatic Differentiation – Reverse Mode (aka. Backpropagation)

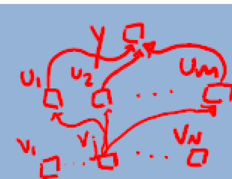
Forward Computation

1. Write an **algorithm** for evaluating the function $y = f(\mathbf{x})$. The algorithm defines a **directed acyclic graph,** where each variable is a node (i.e. the "**computation graph**")
2. Visit each node in **topological order.**
   For variable u_i with inputs $v_1,\ldots, v_N$
   a. Compute $u_i = g_i(v_1,\ldots, v_N)$
   b. Store the result at the node

Backward Computation (Version A)

1. **Initialize** $dy/dy = 1$.
2. Visit each node v_j in **reverse topological order.**
   Let $u_1,\ldots, u_M$ denote all the nodes with v_j as an input
   Assuming that $y = h(\mathbf{u}) = h(u_1,\ldots, u_M)$
   and $\mathbf{u} = g(\mathbf{v})$ or equivalently $u_i = g_i(v_1,\ldots, v_j,\ldots, v_N)$ for all i
   a. We already know dy/du_i for all i
   b. Compute dy/dv_j as below (Choice of algorithm ensures computing (du_i/dv_j) is easy)

$$\frac{dy}{dv_j} = \sum_{i=1}^{M} \frac{dy}{du_i}\frac{du_i}{dv_j}$$



Return partial derivatives dy/du_i for all variables

في فيرجن مختلف في الباك بروب .. التعديل بس هيبقا في انك هتعامل ال intermediate derivatives كإنهم intermediate variables موجودين عندك من البدايه ... احنا هن initialize them to 0 .. بس الطريقه اللي هنحسب بيها هتبقا مختلفه ... ال computational graph لسه متغيرش ... لما هتخش علي ال ui انت عارف كل ال children بتوعو اللي هو كل ال Vs اللي عندك ..

Training | Backpropagation

Automatic Differentiation – Reverse Mode (aka. Backpropagation)

Forward Computation

1. Write an **algorithm** for evaluating the function $y = f(\mathbf{x})$. The algorithm defines a **directed acyclic graph**, where each variable is a node (i.e. the **"computation graph"**)
2. Visit each node in **topological order**.
   For variable u_i with inputs $v_1, \ldots, v_N$
   a. Compute $u_i = g_i(v_1, \ldots, v_N)$
   b. Store the result at the node

Backward Computation (Version B)

1. **Initialize** all partial derivatives dy/du_j to 0 and $dy/dy = 1$.
2. Visit each node in **reverse topological order**.
   For variable $u_i = g_i(v_1, \ldots, v_N)$
   a. We already know dy/du_i
   b. Increment dy/dv_j by $(dy/du_i)(du_i/dv_j)$
   (Choice of algorithm ensures computing (du_i/dv_j) is easy)

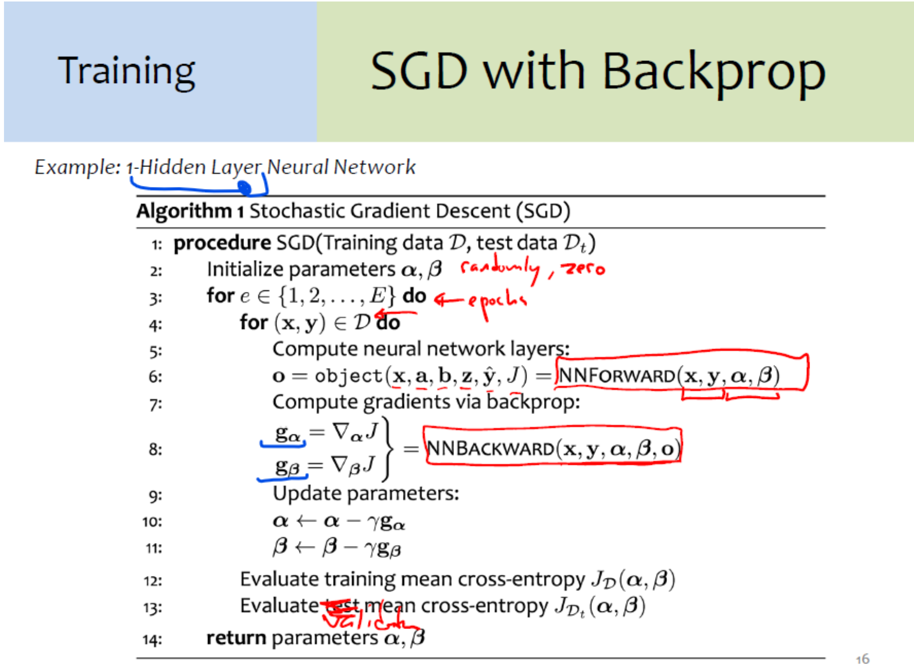
Return partial derivatives dy/du_i for all variables

Training | Backpropagation

Why is the backpropagation algorithm efficient?

1. Reuses **computation from the forward pass** in the backward pass
2. Reuses **partial derivatives** throughout the backward pass (*but only if the algorithm reuses shared computation in the forward pass*)

   (Key idea: partial derivatives in the backward pass should be thought of as variables stored for reuse)



المهم الدكتور حط مثال كدا وقال ان احنا فعلا ممكن نستخدم ال finite difference method عشان نتشيك الحل بتاعنا في الباكبروب .. انا بحب الدكتور ده جداً ..... يلا نخش في الديب ليرننج

اول حاجه هنتكلم علي ال CNN بس الأول هنسال ليه الاهتمام ده ... عشان فيه فلوس كتيره الناس بتستثمر فيها .. بقيت المحاضره جري كتير .. لو مهتم اقراها تاني