# 1 Einops

Reference: https://github.com/arogozhnikov/einops

## 1.1 Features

- user exposed functions: rearrange, reduce, repeat
- self-documenting notation (via strings) for layouts of input and output arrays
- binding patterns to tensor/array dimensions
- optionally provide parameters to named dimensions
- wildcard ... for omission of zero or more range of axes/dimensions
- anonymous axis by specifying its size, but can't be matched across tensors
- allow new output indices by copy/repeat operation
- reduction via presence of matching indices in inputs or via omission of indices in output notation
- low number of backend functions to implement
- focus on data rearrangements and simple transformations (axes reordering, decomposition, composition, reduction, repeats)
- focus on 1 tensor/array transformations
- supported notations: named axis, anonymous axis, unitary axis, ellipsis, (de)compose parenthesis
- (de)composition operator, eg: $(a\ b)$
- supports a list of arrays as input with implied additional outer dimension corresponding to the list
- inferrable dimension sizes, given partial info as parameters
- hide backend framework inconsistency of notations for common array rearrangement operations
- each backend needs to implement some necessary tensor operations; proxy classes for these backends
- inverse transformations are easy to read off by switching patterns for input and output

## 1.2 Optimizations

- caching of tensor type map to backend type for performance
- caching of patterns, axes, and input shape

  compute unknown axis sizes and shape verification on first time, otherwise reuse sequence of commands previously generated

  only need shape verification, unknown axes size when other parameters are already cached

## 1.3 Approaches

- evidence based for API design, via real world use cases
- explicit separation of a few functions over 1 function, for better error messages
- consideration of adoption friction and ease of use
- under the hood, uses reshape-transpose-reshape sequence to achieve transforms

### 1.4 Known Issues

- does not enforce axes alignment between operations
- no means of integrated analysis/tracking of shapes

## 2 Tensor Indexing

abstract index notation:

- index is a label for slot
- index position matters
- coordinate independent, useful to represent tensor transformations

(einstein) index notation:

- index represent incrementing range just as in programming
- indexing refers to components
- basis dependent

index notation (for a binary operation):
$*(s_1, s_2, s_3)$
where
$s_1$: input index set
$s_2$: input index set
$s_3$: output index set

sum/contraction over all entries that are addressed by shared indices:
eg: $C \leftarrow op(s_1, s_2, s_3)(A, B)$ shorthand for:
$C[s_3] \leftarrow \sum_{(s_1 \cup s_2) \setminus s_3} op(A[s_1], B[s_2])$

### 2.1 Notation with Contravariant and Covariant Index

bound index: index occurs both in subscript and superscript

free index: index occurs only once in subscript or superscript; number of free indices is equal to the order of the tensor

$(x, y)$-tensor has $x$ contravariant (upper) free indices, $y$ covariant (lower) free indices; with a total order of $x + y$

applying unary function to object keeps the indices:
$exp(A^i_j) = exp(A)^i_j$

contractions occur for bound indices, eg: $A^i_j B^j_x = C^i_x$

special symbols:

- $\delta^i_j = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}$ (linear identity map)
- $\delta_{ii}$, $\delta^{ii}$ (transposition operation)
- $0^i_j$ (linear map to 0), $0^i$ (0 vector), etc.
- $A^j_i \delta^i_j = A^j_j = \sum_j a^j_j = trace(A)$
- $det(A^j_i)$

### 2.2 Properties

- associative

  let
  $s_3 \subseteq s_1 \cup s_2$
  $s_4 \cap (s_1 \cup s_2) = \emptyset$
  then,
  $*(s_3 s_4, s_4, s_3)(*(s_1, s_2 s_4, s_3 s_4)(A, B), C)$
  $= *(s_1, s_2, s_3)(A, *(s_2 s_4, s_4, s_2)(B, C))$

  order of evaluations:
  $(s_1 \to s_2 s_4) \to s_4 \to s_3$
  vs
  $s_1 \to (s_2 s_4 \to s_4) \to s_3$

- commutative

  $(s_1, s_2, s_3)(A, B) = *(s_2, s_1, s_3)(B, A)$

- distributive

  $*(s_1, s_2, s_3)(A, B) + *(s_1, s_2, s_3)(A, C)$
  $= *(s_1, s_2, s_3)(A, B + C)$
  where $s_3 \subseteq s_1 \cup s_2$

## 3 Derivative Definition

$f : \mathbb{R}^{n_1 \times .. \times .. n_k} \to \mathbb{R}^{m_1 \times .. \times m_l}$
$D \in \mathbb{R}^{m_1 \times .. \times m_l \times n_1 \times .. \times n_k} =$
$\lim_{h \to 0} \frac{\|f(x+h) - f(x) - D \circ h\|}{\|h\|} = 0$
$\iff$ $D$ is a derivative of $f$ at $x$

where inner tensor product is:
$D \circ h = *(s_1 s_2, s_2, s_1)(D, h)$

## 4 Pushforward and Pullback

let $\bar{v} = \frac{\partial y}{\partial v}$ (pullback of node $v$)

let $\dot{v} = \frac{\partial v}{\partial x}$ (pushforward of node $v$)

then, for mixed mode:

$$\begin{aligned} \frac{\partial y}{\partial x} &= \sum_v \frac{\partial y}{\partial v} \frac{\partial v}{\partial x} \\ &= \sum_v \bar{v} \dot{v} \\ &= \sum_{v \in S} *(s_1 s_v, s_v s_2, s_1 s_2)(\bar{v}, \dot{v}) \end{aligned}$$

where:
$s_1 \equiv$ index set of output function y
$s_2 \equiv$ index set of input x
$s_v \equiv$ index set of node $v$
$S \equiv$ set of nodes in any cut of the DAG

computation savings from reordering but hard in general to find best solution:

- compressing order of tensors when possible
- multiply terms in order of tensor-order: vectors first, matrices next, ...

# 5 Forward Mode

$\sum_i \frac{\partial v_i}{\partial x_j} \frac{\partial f}{\partial x_j} = \frac{\partial f}{\partial x_j}$
where $x_j$ are leaf input variables and
where pushforwards of predecessor nodes $v_i$ are computed and cached by the time $\frac{\partial f}{\partial x_j}$ is computed

notation: let $\dot{v} = \frac{\partial v}{\partial x_j}$ be the pushforward of $v$

Generalized cases of local node connections:

- general unary function
- elementwise unary function
- binary addition
- binary multiplication

We seek to compute pushforwards for the above types of ops.

## 5.1 Pushforward of General Unary Function

let:
$f$ be a unary function with:

- domain index set $s_1$
- range index set $s_2$

$x$ be an input variable with index set $s_3$
$C = f(A)$, where $A$ and $C$ are nodes in expression DAG,
then, the pushforward $\dot{C}$ is:
$\dot{C} = *(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A), \dot{A})$
where $f'$ is the derivative of $f$

### 5.1.1 Proof

$f'$ is drivative of $f \iff \lim_{\tilde{h}\to 0} \frac{\|f(A+\tilde{h})-f(A)-f'(A)\circ\tilde{h}\|}{\|\tilde{h}\|} = 0$

let $\tilde{h} = A(x+h) - A(x)$
$\tilde{h} \to 0$ as $h \to 0$

$\lim_{\tilde{h}\to 0} \frac{\|f(A+\tilde{h})-f(A)-f'(A)\circ\tilde{h}\|}{\|\tilde{h}\|} = 0$
$\lim_{h\to 0} \frac{\|f(A+A(x+h)-A(x))-f(A)-f'(A)\circ(A(x+h)-A(x))\|}{\|A(x+h)-A(x)\|} = 0$
$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-f'(A)\circ(A(x+h)-A(x))\|}{\|A(x+h)-A(x)\|} = 0$

let $\dot{A}$ be derivative of $A \iff \lim_{h\to 0} \frac{\|A(x+h)-A(x)-\dot{A}\circ h\|}{\|h\|} = 0$
triangular inequality:
$\lim_{h\to 0} \frac{\|A(x+h)-A(x)\|}{\|h\|} - \frac{\|-\dot{A}\circ h\|}{\|h\|}$
$\leq \lim_{h\to 0} \frac{\|A(x+h)-A(x)-\dot{A}\circ h\|}{\|h\|} = 0$

$\lim_{h\to 0} \frac{\|A(x+h)-A(x)\|}{\|h\|} = \frac{\|-\dot{A}\circ h\|}{\|h\|}$

substitute:
$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-f'(A)\circ(A(x+h)-A(x))\|}{\|A(x+h)-A(x)\|} = 0$
$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-f'(A)\circ(\dot{A}\circ h)\|}{\|A(x+h)-A(x)\|} = 0$

using definition of tensor inner product:
$\dot{A} \circ h = *(s_1 s_3, s_3, s_1)(\dot{A}, h)$
$f'(A) \circ (\dot{A} \circ h) = f'(A) \circ (*(s_1 s_3, s_3, s_1)(\dot{A}, h))$
$f'(A) \circ (\dot{A} \circ h) = *(s_2 s_1, s_1, s_2)(f'(A), *(s_1 s_3, s_3, s_1)(\dot{A}, h))$

using associativity:
$f'(A)\circ(\dot{A}\circ h) = *(s_2 s_3, s_3, s_2)(*(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A), \dot{A}), h)$
$f'(A) \circ (\dot{A} \circ h) = *(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A), \dot{A}) \circ h$

$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-*(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A),\dot{A})\circ h\|}{\|A(x+h)-A(x)\|} = 0$

$\tilde{h} \to 0$ as $h \to 0$
$(\exists k)\|A(x+h) - A(x)\| \leq \frac{1}{k}\|h\|$
$(\exists k)\frac{k}{\|h\|} \leq \frac{1}{\|A(x+h)-A(x)\|}$
then
$\lim_{h\to 0} \frac{k\|f(A(x+h))-f(A)-*(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A),\dot{A})\circ h\|}{\|h\|} \leq$
$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-*(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A),\dot{A})\circ h\|}{\|A(x+h)-A(x)\|} = 0$

$\lim_{h\to 0} \frac{\|f(A(x+h))-f(A)-*(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A),\dot{A})\circ h\|}{\|h\|} = 0$
$\iff *(s_2 s_1, s_1 s_3, s_2 s_3)(f'(A), \dot{A}) = \dot{C}$
is derivative of $f(A(x))$ wrt. $x$, then $\dot{C}$ is a pushforward of $C$

### 5.2 Pushforward of Elementwise Unary Function

let:
$x$ be an input variable with index set $s_2$
$f$ is an elementwise function
$A$ is a node in the expression DAG with index set $s_1$
$C = f(A)$
then:
pushforward of node $C$, $\dot{C}$, is:
$\dot{C} = *(s_1, s_1s_2, s_1s_2)(f'(A), \dot{A})$
where $f'$ is derivative of $f$

#### 5.2.1 Proof

using definition of derivative:
$\lim_{h\to 0} \frac{1}{\|h\|} \|A(x+h) - A(x) - \dot{A} \circ h\| = 0$
$\iff$ $\dot{A}$ is derivative of $A$
per scalar tensor entry indexed by multi-indices $s$:
$\lim_{h\to 0} \frac{1}{\|h\|} |A(x+h)_s - A(x)_s - (\dot{A} \circ h)_s| = 0$
using chain rule:
let $g(x) = A(x)_s$
$f(g(x)) = f(A(x)_s)$
$(f(g(x)))' = f'(g(x))g'(x)$
$g'(x) = \dot{A}_s$
$(f(g(x)))' = f'(A(x)_s)\dot{A}_s$
using defintion of derivative:
$(f(g(x)))'$ is derivative of $f(g(x))$
$\iff \lim_{h\to 0} \frac{1}{\|h\|} |f(g(x+h)) - f(g(x)) - (f(g(x)))' \circ h_s| = 0$
$\lim_{h\to 0} \frac{1}{\|h\|} |f(A(x+h)_s) - f(A(x)_s) - f'(A(x)_s)\dot{A}_s \circ h_s| = 0$
$\lim_{h\to 0} \frac{1}{\|h\|} |f(A(x+h)_s) - f(A(x)_s) - f'(A(x)_s)(\dot{A} \circ h)_s| = 0$
above holds for all multi-indices $s \implies$
$\lim_{h\to 0} \frac{\|f(A(x+h)) - f(A(x)) - *(s_1,s_1,s_1)(f'(A(x)), \dot{A}\circ h)\|}{\|h\|} = 0$
using inner tensor product definition:
$*(s_1, s_1, s_1)(f'(A(x)), \dot{A}\circ h) = *(s_1, s_1, s_1)(f'(A(x)), *(s_1s_2, s_2, s_1)(\dot{A}, h))$
using associativity rule:
$*(s_1, s_1, s_1)(f'(A(x)), \dot{A}\circ h) = *(s_1s_2, s_2, s_1)(*(s_1, s_1s_2, s_1s_2)(f'(A(x)), \dot{A}), h)$
$*(s_1, s_1, s_1)(f'(A(x)), \dot{A}\circ h) = *(s_1, s_1s_2, s_1s_2)(f'(A(x)), \dot{A})\circ h$
substitute:
$\lim_{h\to 0} \frac{\|f(A(x+h)) - f(A(x)) - *(s_1,s_1s_2,s_1s_2)(f'(A(x)), \dot{A})\circ h\|}{\|h\|} = 0$
$\iff$ $*(s_1, s_1s_2, s_1s_2)(f'(A(x)), \dot{A})$ is derivative of $f(A(x))$ wrt. $x$
$\iff$ $\dot{C}$, pushforward of $C$, is $*(s_1, s_1s_2, s_1s_2)(f'(A(x)), \dot{A})$

### 5.3 Pushforward of Binary Addition

let $C = f(A, B)$ where $f$ is addition
then, $\dot{C} = \dot{A} + \dot{B}$ (sum of pushforwards of summands)

### 5.4 Pushforward of Binary Multiplication

let $C = *(s_1, s_2, s_3)(A, B)$, then
$\dot{C} = *(s_1 s_4, s_2, s_3 s_4)(\dot{A}, B) + *(s_1, s_2 s_4, s_3 s_4)(A, \dot{B})$

#### 5.4.1 Proof

use definition of derivative: $\dot{C} = \frac{\partial C}{\partial A}\dot{A} + \frac{\partial C}{\partial B}\dot{B}$
consider the case of $\frac{\partial C}{\partial B}\dot{B}$ (contribution from node $B$):

$\frac{\partial C}{\partial B}\dot{B} = C(x+h) - C(x) - \dot{C} \circ h$
$= *(s_1, s_2, s_3)(A, B(x+h))$
$- *(s_1, s_2, s_3)(A, B(x))$
$- *(s_1, s_2 s_4, s_3 s_4)(A, \dot{B}) \circ h$

where we let $\dot{C} = *(s_1, s_2 s_4, s_3 s_4)(A, \dot{B})$
where $s_4$ is the index set of input $x$ to $B$

using inner tensor difinition:
$x \circ y = *(s_1 s_2, s_2, s_1)(x, y)$

$C(x+h) - C(x) - \dot{C} \circ h =$
$*\,(s_1, s_2, s_3)(A, B(x+h))$
$- *(s_1, s_2, s_3)(A, B(x))$
$- *(s_3 s_4, s_4, s_3)(*(s_1, s_2 s_4, s_3 s_4)(A, \dot{B}), h)$

using associativity:
$*(s_3 s_4, s_4, s_3)(*(s_1, s_2 s_4, s_3 s_4)(A, \dot{B}), h)$
$= *(s_1, s_2, s_3)(A, *(s_2 s_4, s_4, s_2)(\dot{B}, h))$

$C(x+h) - C(x) - \dot{C} \circ h =$
$*\,(s_1, s_2, s_3)(A, B(x+h))$
$- *(s_1, s_2, s_3)(A, B(x))$
$- *(s_1, s_2, s_3)(A, *(s_2 s_4, s_4, s_2)(\dot{B}, h))$

using distributivity:

$C(x+h) - C(x) - \dot{C} \circ h =$
$*\,(s_1, s_2, s_3)(A, B(x+h) - B(x) - *(s_2 s_4, s_4, s_2)(\dot{B}, h))$

using definition of derivative:

$\lim_{h \to 0} \frac{\|B(x+h) - B(x) - \dot{B} \circ h\|}{\|h\|} = 0 \iff$
$\dot{B}$ is a derivative of $B$ at $x$

$\lim_{h \to 0} \frac{\|C(x+h) - C(x) - \dot{C} \circ h\|}{\|h\|}$
$\leq \|A\| \lim_{h \to 0} \frac{\|B(x+h) - B(x) - \dot{B} \circ h\|}{\|h\|} = 0$
$\iff *(s_1, s_2 s_4, s_3 s_4)(A, \dot{B})$ is the pushfoward contribution to $\dot{C}$ from $B$

similar logic follows for contribution to $\dot{C}$ from $A$, then the proof is complete

## 6 Reverse Mode

node $v_i$ stores $\frac{\partial y_j}{\partial v_i} = \bar{v_i}$

computing for node $z$:
$\bar{z} = \frac{\partial y_j}{\partial z} = \sum_f \frac{\partial y_j}{\partial f} \frac{\partial f}{\partial z}$ where $f$ is successor/output node of $z$, eg: there is an outgoing edge from $z$ to $f$

$\bar{z} = \sum_f \bar{f} \frac{\partial f}{\partial z}$, where $\bar{f}$ is the pullback of $f$ and is computed and cached at node $f$ by the time $\bar{z}$ needs to be updated

$O(m)$ rounds for $m$ outputs $y_j$

Generalized cases of local node connections:

- general unary function
- elementwise unary function
- binary addition
- binary multiplication

### 6.1 Pullback of General Unary Function

let:
$Y$ be an output function with index set $s_3$
$f$ be a general unary function with domain index set $s_1$ and range index set $s_2$
$A, C$ be nodes in expression DAG
$C = f(A)$
then, contribution of $C$ to pullback of $A$, $\bar{A}$, is:
$*(s_3s_2, s_2s_1, s_3s_1)(\bar{f}, f'(A))$ where $f'$ is derivative of $f$

#### 6.1.1 Proof

$\bar{A} = \frac{\partial Y}{\partial A} = \frac{\partial Y}{\partial f}\frac{\partial f}{\partial A} = \bar{f}\frac{\partial f}{\partial A}$

$\bar{f}$ satisfies $\lim_{\tilde{h}\to 0}\frac{1}{\|\tilde{h}\|}\|Y(f+\tilde{h}) - Y(f) - \bar{f}\circ\tilde{h}\| = 0$

let: $\tilde{h} = f(A+h) - f(A)$, $f = f(A)$
$Y(f+\tilde{h}) - Y(f) - \bar{f}\circ\tilde{h} = Y(f(A+h)) - Y(f(A)) - \bar{f}\circ(f(A+h) - f(A))$

$f'$ satisfies $\lim_{h\to 0}\frac{1}{\|h\|}\|f(A+h) - f(A) - f'(A)\circ h\| = 0$

substitute:
$\lim_{\tilde{h}\to 0}\frac{1}{\|\tilde{h}\|}\|Y(f+\tilde{h}) - Y(f) - \bar{f}\circ\tilde{h}\| = 0$
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A)) - \bar{f}\circ(f(A+h) - f(A))\| = 0$
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A)) - \bar{f}\circ(f'(A)\circ h)\| = 0$

using tensor inner product:
$f'(A)\circ h = *(s_2s_1, s_1, s_2)(f'(A), h)$
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A)) - \bar{f}\circ(*(s_2s_1, s_1, s_2)(f'(A), h))\| = 0$

using tensor inner product:
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- *(s_3s_2, s_2, s_3)(\bar{f}, *(s_2s_1, s_1, s_2)(f'(A), h))\| = 0$

using associativity:
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- *(s_3s_1, s_1, s_3)(*(s_3s_2, s_2s_1, s_3s_1)(\bar{f}, f'(A)), h)\| = 0$

$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- *(s_3s_2, s_2s_1, s_3s_1)(\bar{f}, f'(A))\circ h\| = 0$
$\iff *(s_3s_2, s_2s_1, s_3s_1)(\bar{f}, f'(A))$
is the derivative of $Y(f(A))$ and is the contribution of node $C$ to pullback $\bar{A}$

### 6.2 Pullback of Elementwise Unary Function

let:
$Y$ be an output function with index set $s_2$
$f$ be an elementwise unary function
$A$ be a node in expression DAG with index set $s_1$
$C$ be a node in expression DAG with index set $s_1$
$C = f(A)$
then, contribution of $C$ to pullback $\bar{A}$ is:
$*(s_2s_1, s_1, s_2s_1)(\bar{f}, f'(A))$
where $f'$ is derivative of $f$

#### 6.2.1 Proof

$\bar{A}_{\text{via } f} = \frac{\partial Y}{\partial f}\frac{\partial f}{\partial A} = \bar{f}\frac{\partial f}{\partial A}$

$\bar{f}$ is derivative of $Y(f)$ satisfies:
$\lim_{\tilde{h}\to 0}\frac{1}{\|\tilde{h}\|}\|Y(f+\tilde{h}) - Y(f) - \bar{f}\circ\tilde{h}\| = 0$

let $\tilde{h} = f(A+h) - f(A)$, $f = f(A)$, then:
$Y(f+\tilde{h}) - Y(f) - \bar{f}\circ\tilde{h}$
$= Y(f(A+h)) - Y(f(A))$
$- \bar{f}\circ(f(A+h) - f(A))$

$f'$ is derivative of $f$ and entry-wise, then:
$\lim_{h\to 0}\frac{1}{\|h\|}\|f(A+h) - f(A)$
$- *(s_1, s_1, s_1)(f'(A), h)\| = 0$

$\tilde{h}\to 0$ as $h\to 0$, substitute:
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- \bar{f}\circ(f(A+h) - f(A))\| = 0$
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- \bar{f}\circ(*(s_1, s_1, s_1)(f'(A), h))\| = 0$

using tensor inner product definition:
$\bar{f}\circ(*(s_1, s_1, s_1)(f'(A), h))$
$= *(s_2s_1, s_1, s_2)(\bar{f}, *(s_1, s_1, s_1)(f'(A), h))$

using associativity property:
$\bar{f}\circ(*(s_1, s_1, s_1)(f'(A), h))$
$= *(s_2, s_1, s_1, s_2)(*(s_2s_1, s_1, s_2s_1)(\bar{f}, f'(A)), h)$

using tensor inner product definition:
$\bar{f}\circ(*(s_1, s_1, s_1)(f'(A), h))$
$= *(s_2s_1, s_1, s_2s_1)(\bar{f}, f'(A))\circ h$

substitute:
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- \bar{f}\circ(*(s_1, s_1, s_1)(f'(A), h))\| = 0$
$\lim_{h\to 0}\frac{1}{\|h\|}\|Y(f(A+h)) - Y(f(A))$
$- *(s_2s_1, s_1, s_2s_1)(\bar{f}, f'(A))\circ h\| = 0$
$\iff *(s_2s_1, s_1, s_2s_1)(\bar{f}, f'(A))$
is the derivative of $Y(f(A))$ wrt. $A$ and is the contribution of node $C$ to pullback $\bar{A}$

### 6.3 Pullback of Binary Addition

let:
$C = +(A, B)$
then,
$\bar{A} = \bar{C}$
$\bar{B} = \bar{C}$
where $\bar{C}$ is the pullback of $C$

### 6.4 Pullback of Binary Multiplication

let: $Y$ be an output node with index set $s_4$
$C = *(s_1, s_2, s_3)(A, B)$ be a multiplication node of expression DAG
then, contribution of node $C$ to pullback $\bar{B}$ is:
$*(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A)$
contribution of node $C$ to pullback $\bar{A}$ is:
$*(s_4 s_3, s_2, s_4 s_1)(\bar{C}, B)$

#### 6.4.1 Proof

focus on derivation of pullback contribution to $\bar{B}$
using definition of derivative:
$\lim_{\tilde{h} \to 0} \frac{1}{\|\tilde{h}\|} \|Y(C + \tilde{h}) - Y(C) - \frac{\partial Y}{\partial C} \circ \tilde{h}\| = 0$
$\iff \bar{C} = \frac{\partial Y}{\partial C}$ is derivative of $Y$ evaluated at $C$

let $\tilde{h} = *(s_1, s_2, s_3)(A, h)$, then
$Y(C + \tilde{h}) - Y(C) - \bar{C} \circ \tilde{h}$
$= Y(C + *(s_1, s_2, s_3)(A, h)) - Y(C) - \bar{C} \circ *(s_1, s_2, s_3)(A, h)$
$= Y(*(s_1, s_2, s_3)(A, B) + *(s_1, s_2, s_3)(A, h)) - Y(*(s_1, s_2, s_3)(A, B)) - \bar{C} \circ *(s_1, s_2, s_3)(A, h)$
$= Y(*(s_1, s_2, s_3)(A, B + h)) - Y(*(s_1, s_2, s_3)(A, B)) - \bar{C} \circ *(s_1, s_2, s_3)(A, h)$

using definition of tensor inner product:
$\bar{C} \circ *(s_1, s_2, s_3)(A, h) = *(s_4 s_3, s_3, s_4)(\bar{C}, *(s_1, s_2, s_3)(A, h))$

using associativity property:
$\bar{C} \circ *(s_1, s_2, s_3)(A, h) = *(s_4 s_2, s_2, s_4)(*(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A), h)$
$\bar{C} \circ *(s_1, s_2, s_3)(A, h) = *(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A) \circ h$

$Y(C + \tilde{h}) - Y(C) - \bar{C} \circ \tilde{h}$
$= Y(*(s_1, s_2, s_3)(A, B + h)) - Y(*(s_1, s_2, s_3)(A, B))$
$- *(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A) \circ h$

substitute:
$0 = \lim_{h \to 0} \frac{1}{\|*(s_1, s_2, s_3)(A, h)\|} \|Y(*(s_1, s_2, s_3)(A, B + h)) - Y(*(s_1, s_2, s_3)(A, B) - *(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A) \circ h\|$
$\iff *(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A)$ is derivative of
$Y(*(s_1, s_2, s_3)(A, B))$ wrt. $B$
then, contribution of $C$ to $\bar{B}$, eg $(\frac{\partial Y}{\partial B})_{\text{via } C}$ is:
$*(s_4 s_3, s_1, s_4 s_2)(\bar{C}, A)$

use similar logic for derivation of pulllback contribution to $\bar{A}$:
contribution of $C$ to $\bar{A}$ is: $*(s_4 s_3, s_2, s_4 s_1)(\bar{C}, B)$