Introduction to Deep Learning

IUT AI Chapter

Course Outline

- **What is Deep Learning?**
- **Improving Our Models**
- **Deep Learning for Images**
- **Deep Learning for Texts**
- **Advanced Topics**

This is a Semi-Advanced Course.

- So we assume the basic knowledge of:
  - Machine learning
  - Probability theory
  - Linear algebra and calculus
  - Python programming

Week 1

- **Linear Regression and Linear Classification**
- **Artificial Neural Networks**
- **Cost Functions**
- **Gradient descent**
- **Activation Functions**
- **BackPropagation**
- **Overfit and Underfit**

Supervised Learning

x_i — example

y_i — target value

$x_i = (x_{i1}, \dots, x_{id})$ — features

$X = \big((x_1, y_1), (x_2, y_2), \dots, (x_\ell, y_\ell)\big)$ — training set

$a(x)$ — model, hypothesis

$$x \longrightarrow a(x) \longrightarrow y^{pred}$$

Regression and Classification

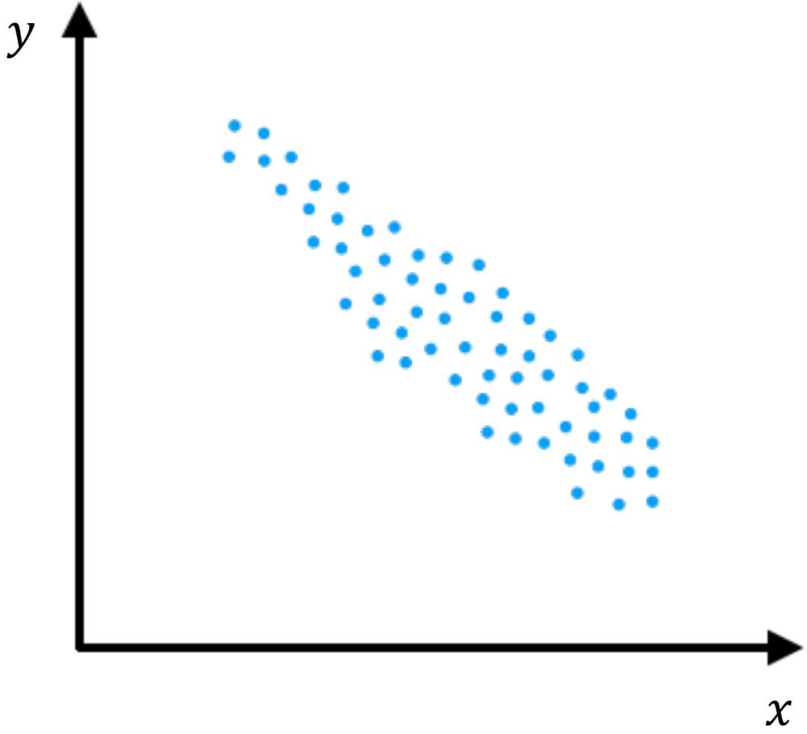
$y_i \in \mathbb{R}$ — regression task

- Salary prediction
- Movie rating prediction

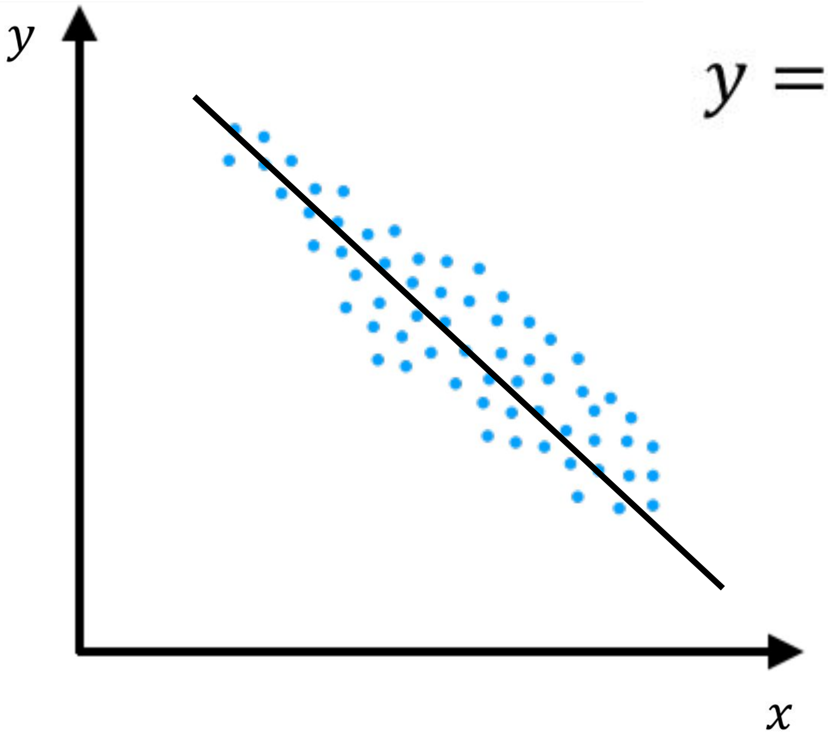
y_i belongs to a finite set — classification task

- Object recognition
- Topic classification

Linear Model for Regression example



Linear Model for Regression example



$$y = w_1 x + w_0$$

Construct Our Linear Model

$$a(x) = b + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$$

- **w1,w2,...,wd -> Coefficients (weights)**
- **b -> bias**
- **How many Parameters?**

How to Measure Our Model Quality?

$$L(w) = \frac{1}{\ell} ||Xw - y||^2 \to \min_{w,}$$

Exact Solution

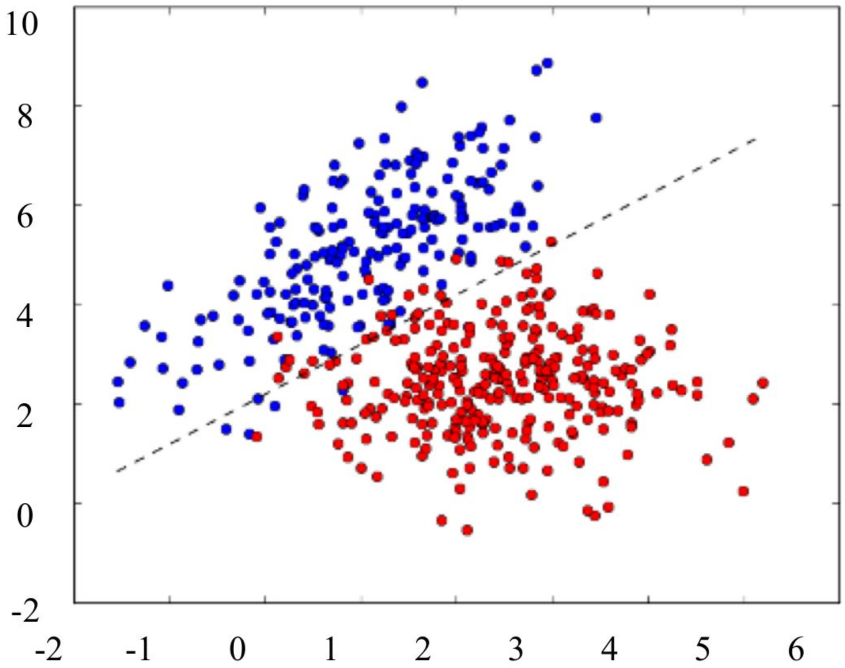
$$w = (X^T X)^{-1} X^T y$$

Why Don't We Just Use That Equation?

- **What is the time complexity of Matrix Inversion?**

O(n^3) Not very reasonable for High Dimensional Data.

Linear Model for Classification example



Construct Our Linear Model

Multi-class classification ($y \in \{1, \dots, K\}$):

$$a(x) = \arg \max_{k \in \{1,\dots,K\}} (w_k^T x)$$

Number of parameters: K*d ($w_k \in \mathbb{R}^d$)

Classification Loss

Classification accuracy:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}[a(x_i) = y_i]$$

- Not differentiable
- Doesn't assess model confidence

[P] — Iverson bracket:

$$[P] = \begin{cases} 1, & P \text{ is true} \\ 0, & P \text{ is false} \end{cases}$$

Classification Score

Class scores (**logits**) from a linear model:

$$z = (w_1^T x, \dots, w_K^T x)$$

$$\downarrow$$

$$(e^{z_1}, \dots, e^{z_K})$$

$$\downarrow$$

$$\sigma(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{K} e^{z_k}}, \dots, \frac{e^{z_K}}{\sum_{k=1}^{K} e^{z_k}} \right)$$

(softmax transform)

Find The Similarity

Predicted class probabilities (model output):

$$\sigma(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{K} e^{z_k}}, \dots, \frac{e^{z_K}}{\sum_{k=1}^{K} e^{z_k}} \right)$$

Target values for class probabilities:

$$p = ([y = 1], \dots, [y = K])$$

Similarity between z and p can be measured by the cross-entropy:

$$-\sum_{k=1}^{K} [y = k] \log \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}} = -\log \frac{e^{z_y}}{\sum_{j=1}^{K} e^{z_j}}$$

Cross-Entropy for Classification

Cross-entropy is differentiable and can be used as a loss function:

$$L(w,b) = -\sum_{i=1}^{\ell}\sum_{k=1}^{K}[y_i = k]\log\frac{e^{w_k^T x_i}}{\sum_{j=1}^{K} e^{w_j^T x_i}}$$

$$= -\sum_{i=1}^{\ell}\log\frac{e^{w_{y_i}^T x_i}}{\sum_{j=1}^{K} e^{w_j^T x_i}} \to \min_w$$

Gradient Descent

Optimization problem:

$$L(w) = \sum_{i=1}^{\ell} L(w; x_i, y_i) \to \min_w$$

w^0 — initialization

while True:

$$w^t = w^{t-1} - \eta_t \nabla L(w^{t-1})$$

if $\|w^t - w^{t-1}\| < \epsilon$ then break

Gradient Descent

Mean squared error:

$$\nabla L(w) = \frac{1}{\ell} \sum_{i=1}^{\ell} \nabla (w^T x_i - y_i)^2$$

- ℓ gradients should be computed on each step
- If the dataset doesn't fit in memory, it should be read from the disk on every GD step

Stochastic Gradient Descent

Optimization problem:

$$L(w) = \sum_{i=1}^{\ell} L(w; x_i, y_i) \to \min_w$$

w^0 — initialization

while True:

i = random index between 1 and ℓ

$w^t = w^{t-1} - \eta_t \nabla L(w^{t-1}; x_i; y_i)$

if $\|w^t - w^{t-1}\| < \epsilon$ then break

Mini-Batch Gradient Descent

Optimization problem:

$$L(w) = \sum_{i=1}^{\ell} L(w; x_i, y_i) \to \min_w$$

w^0 — initialization

while True:

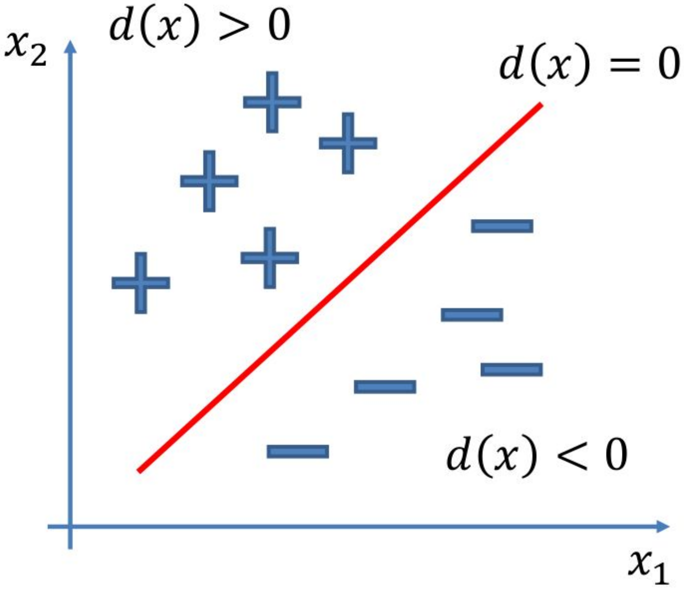
$i_1, \dots, i_m$ = random indices between 1 and ℓ

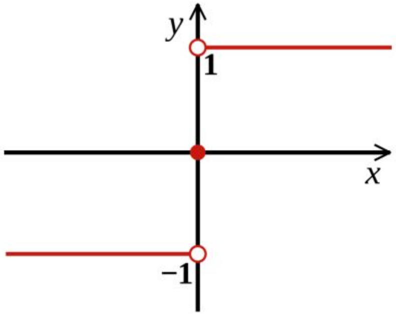
$w^t = w^{t-1} - \eta_t \frac{1}{m} \sum_{j=1}^{m} \nabla L\left(w^{t-1}; x_{i_j}; y_{i_j}\right)$

if $\|w^t - w^{t-1}\| < \epsilon$ then break

MLP (Multi-Layer Perceptron)

- Features: $x = (x_1, x_2)$
- Target: $y \in \{+1, -1\}$
- Decision function: $d(x) = \boldsymbol{w_0} + \boldsymbol{w_1}x_1 + \boldsymbol{w_2}x_2$
- Algorithm: $a(x) = \text{sign}(d(x))$

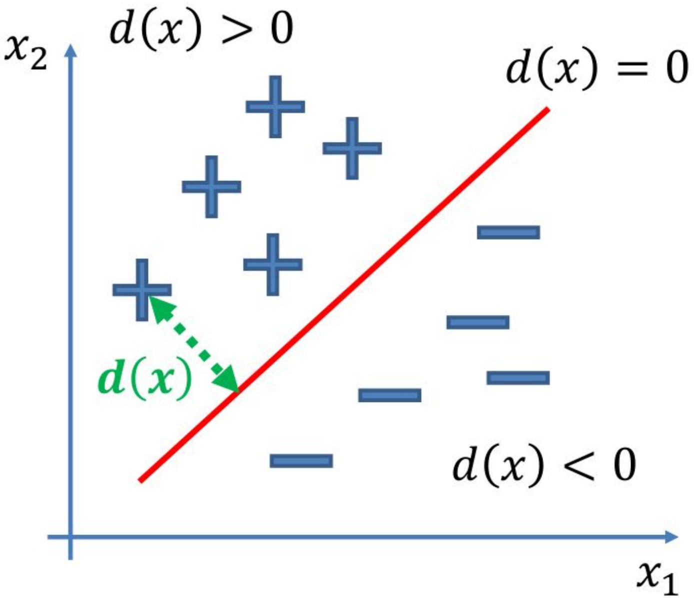


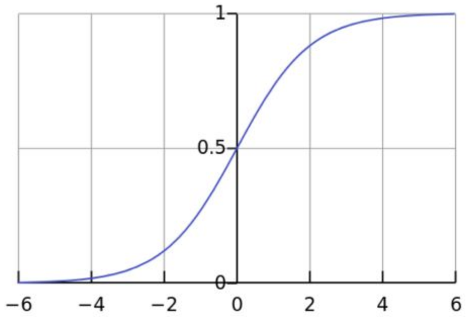


$\text{sign}(x)$

Logistic Regression

- Predicts probability of the positive class (+1)
- Decision function: $d(x) = \boldsymbol{w_0} + \boldsymbol{w_1} x_1 + \boldsymbol{w_2} x_2$
- Algorithm: $a(x) = \sigma\big(d(x)\big)$

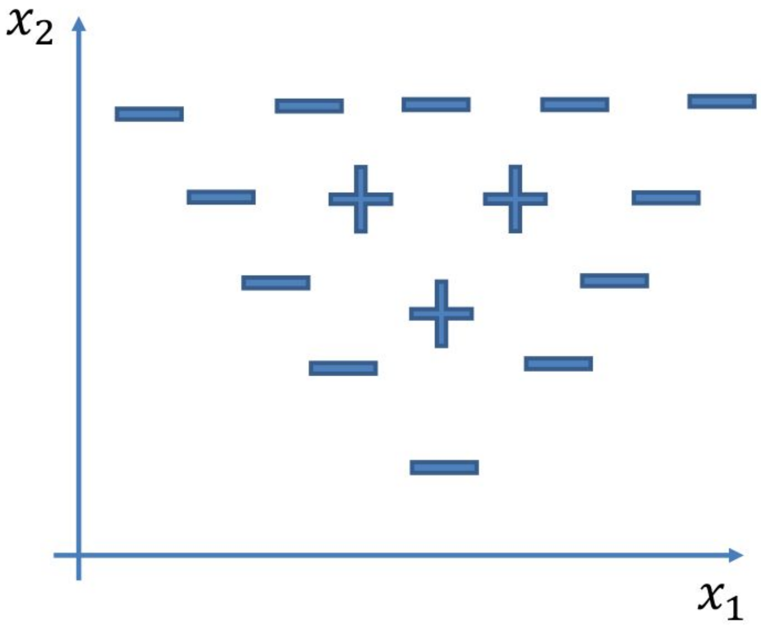




$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

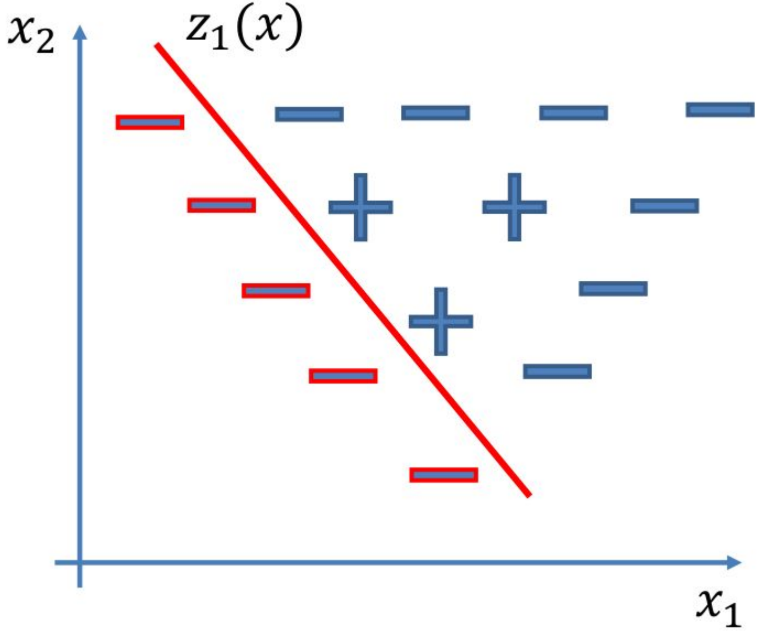
Triangle Problem

- Features: $x = (x_1, x_2)$
- Target: $y \in \{+1, -1\}$



Triangle Problem

- Features: $x = (x_1, x_2)$
- Target: $y \in \{+1, -1\}$



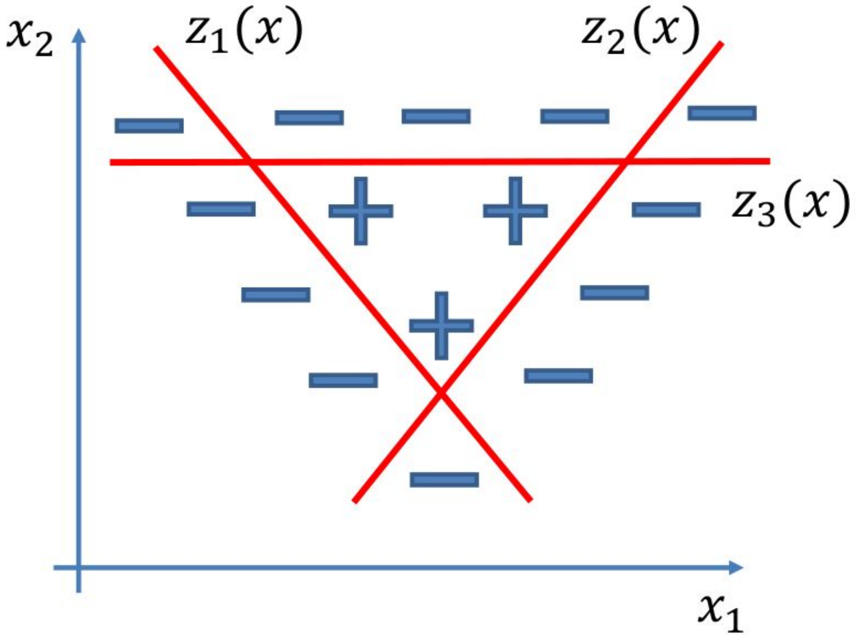
$$z_1 = \sigma(\boldsymbol{w_{0,1}} + \ \boldsymbol{w_{1,1}} x_1 + \boldsymbol{w_{2,1}} x_2)$$

Triangle Problem

- Features: $x = (x_1, x_2)$
- Target: $y \in \{+1, -1\}$

One Logistic Regression Per line.

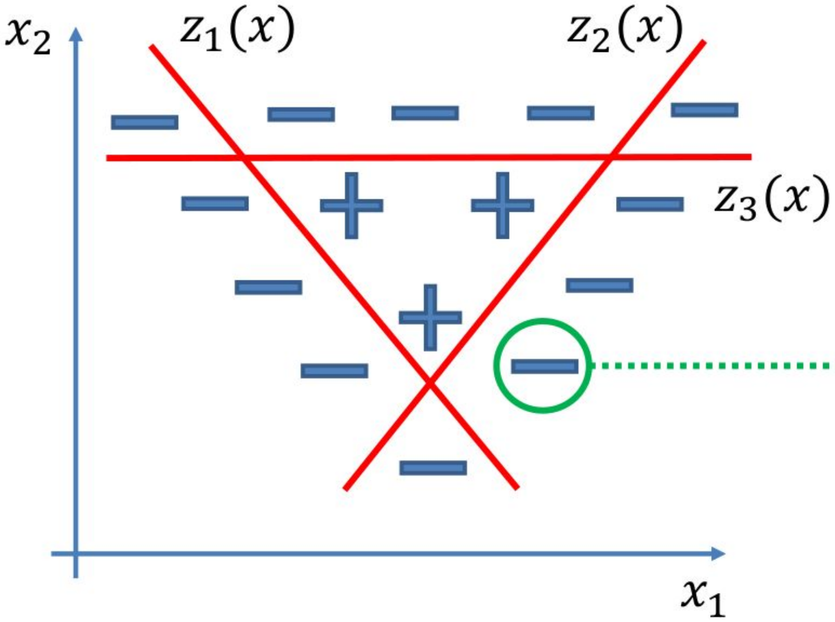
Assume that somehow we found those 3 line.



$$z_i = \sigma(\mathbf{w_{0,i}} + \mathbf{w_{1,i}} x_1 + \mathbf{w_{2,i}} x_2)$$

New Features

- Features: $x = (x_1, x_2)$
- Target: $y \in \{+1, -1\}$



New features:

$z_1(x)$	$z_2(x)$	$z_3(x)$	y
0.6	0.3	0.8	-1
0.7	0.7	0.7	+1

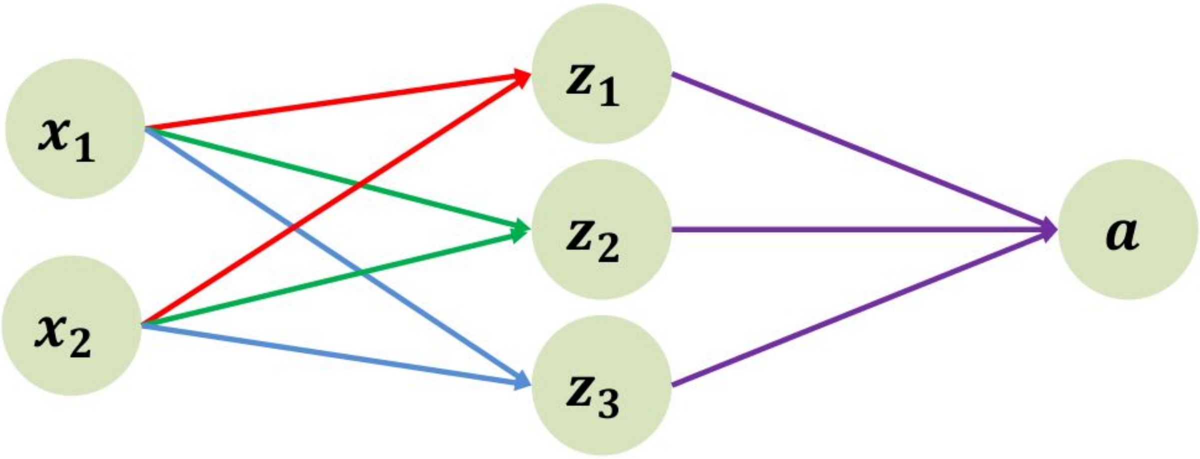
$(x_1, x_2) \rightarrow (z_1, z_2, z_3)$

What to do next?

$$z_i = \sigma(\boldsymbol{w_{0,i}} + \boldsymbol{w_{1,i}} x_1 + \boldsymbol{w_{2,i}} x_2)$$

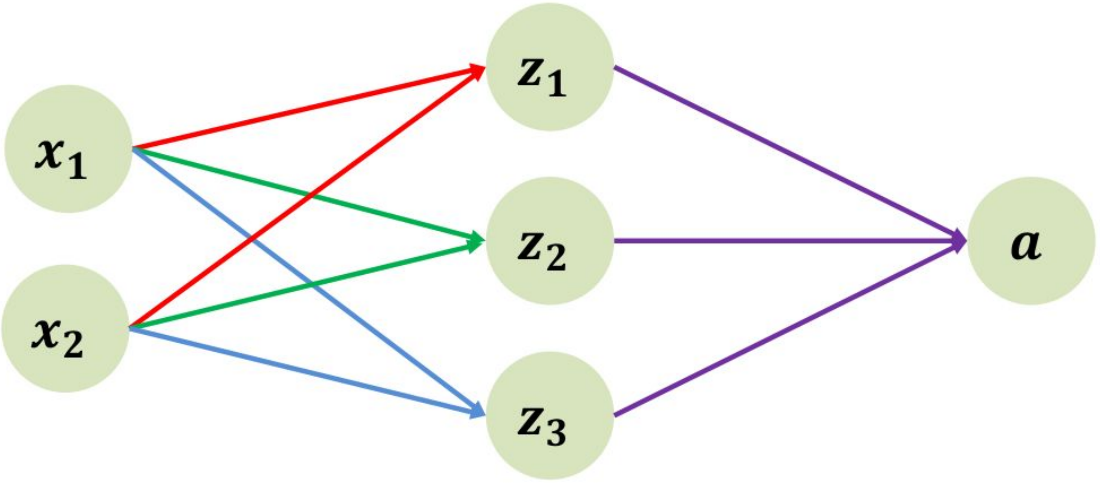
Computation Graph

- Let's rewrite our algorithm in terms of a **computation graph**:



Nodes: computed variables ($x_1, x_2, z_1, z_2, z_3, a$)
Edges: dependencies (we need x_1 and x_2 to compute z_1)

Multi-Layer Perceptron



Input layer | **Hidden layer** | **Output layer**

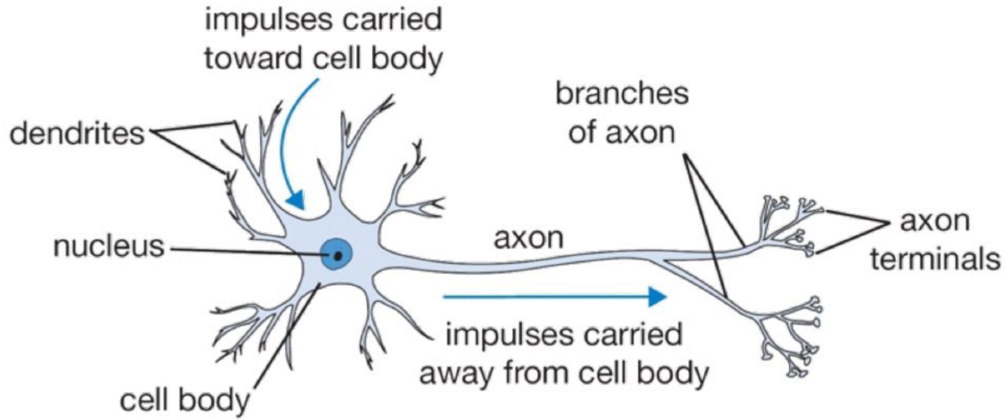
Features

Here each node is a **neuron**:

1. Take a linear combination of inputs
2. Apply **activation** function (e.g. $\sigma(x)$)

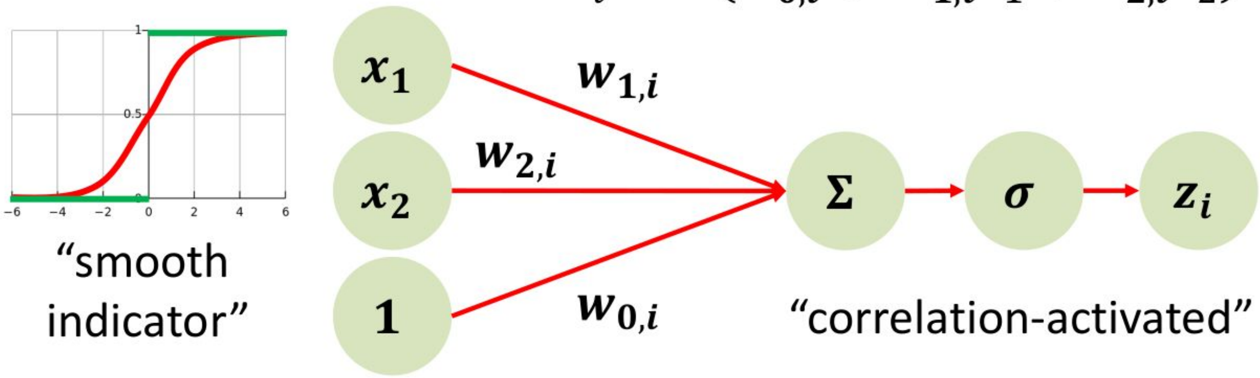
Why Neuron?

- Neuron in a human brain:

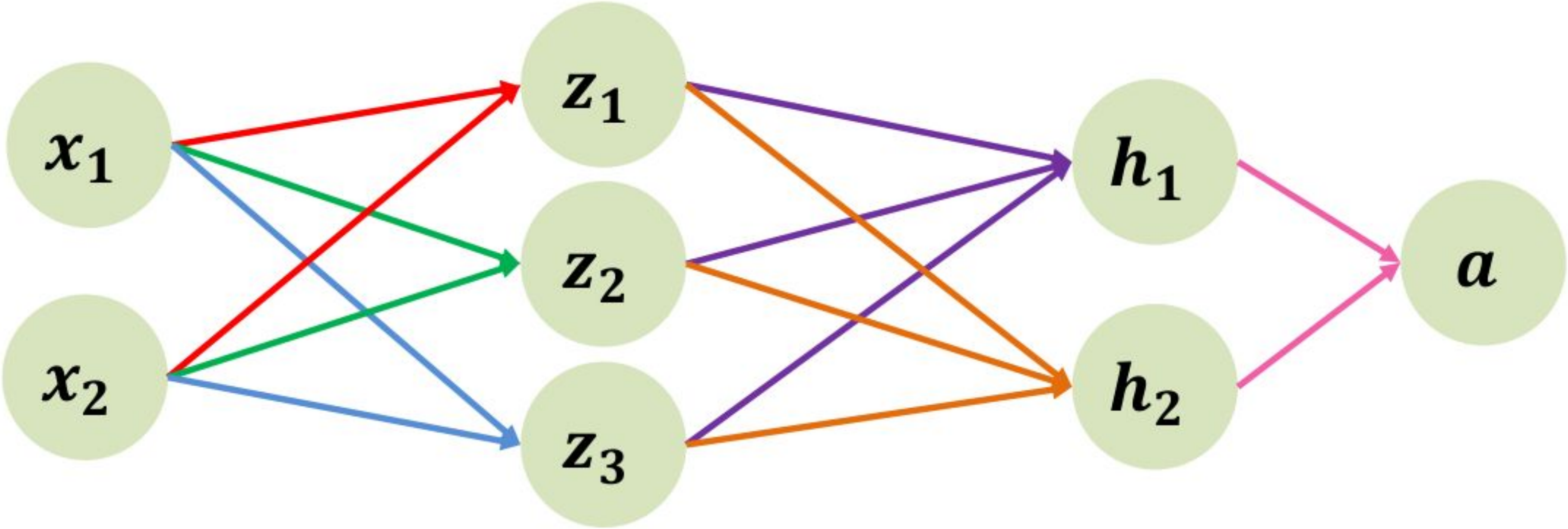


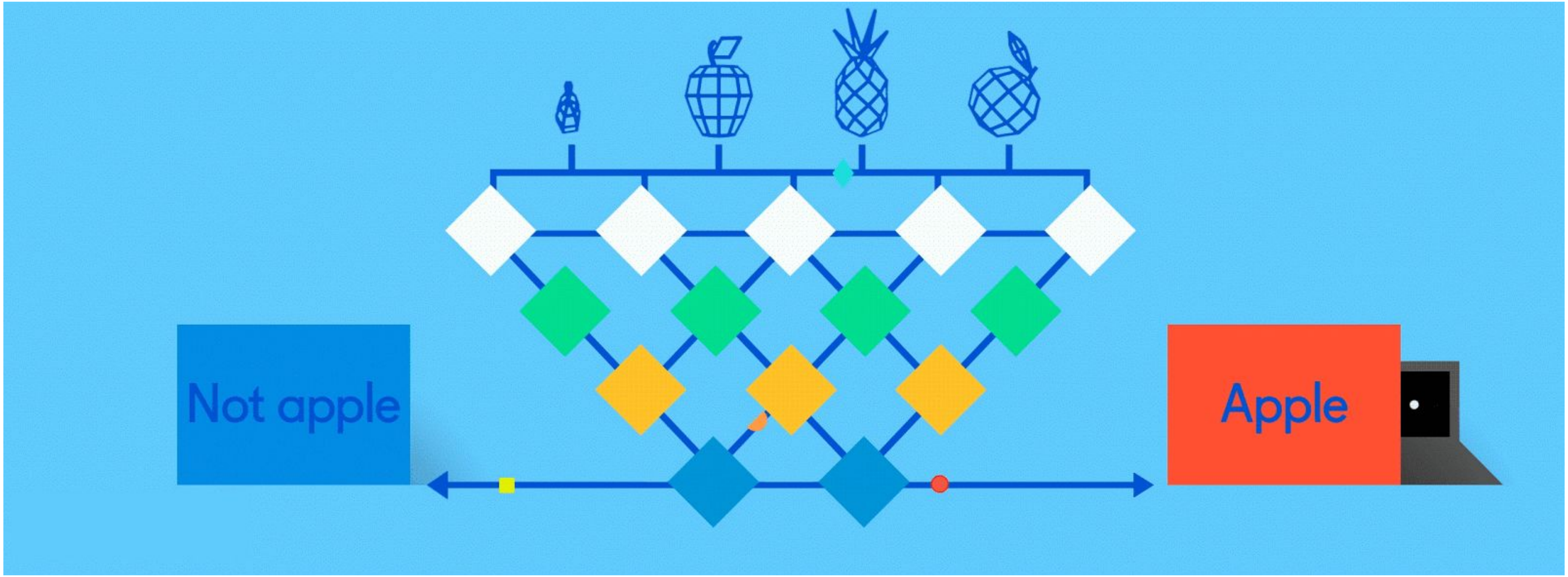
- Artificial neuron:

$$z_i = \sigma(\boldsymbol{w_{0,i}} + \boldsymbol{w_{1,i}} x_1 + \boldsymbol{w_{2,i}} x_2)$$



What is The activation Function? How many layers? How many neurons per layer?




Not apple
Apple

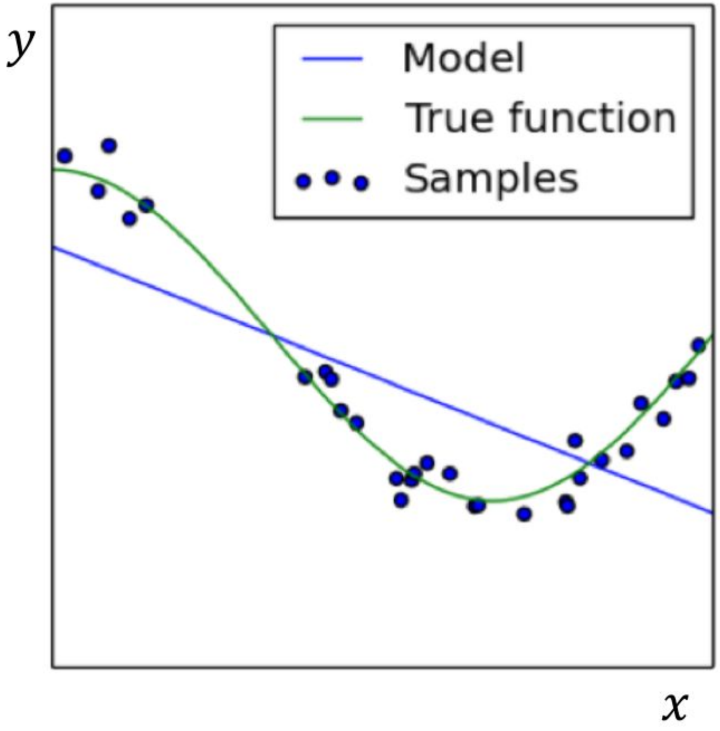
Generalization

- Consider a model with accuracy 80% on training set
- How will it perform on new data?
- In other words, does our model generalize well?

Underfitting

Training set: $X \subset \mathbb{R}$

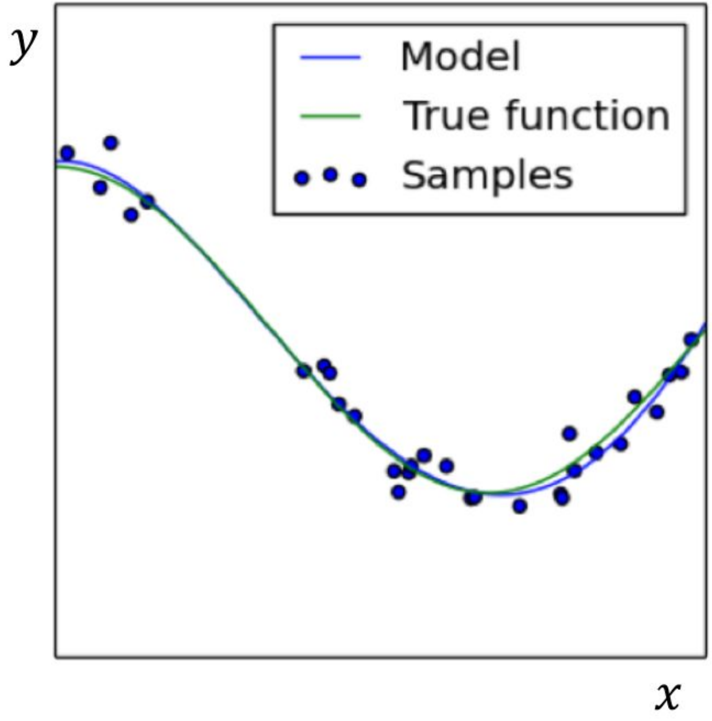
Model: $a(x) = b + w_1 x$



Appropriate Fitted Model

Training set: $X \subset \mathbb{R}$

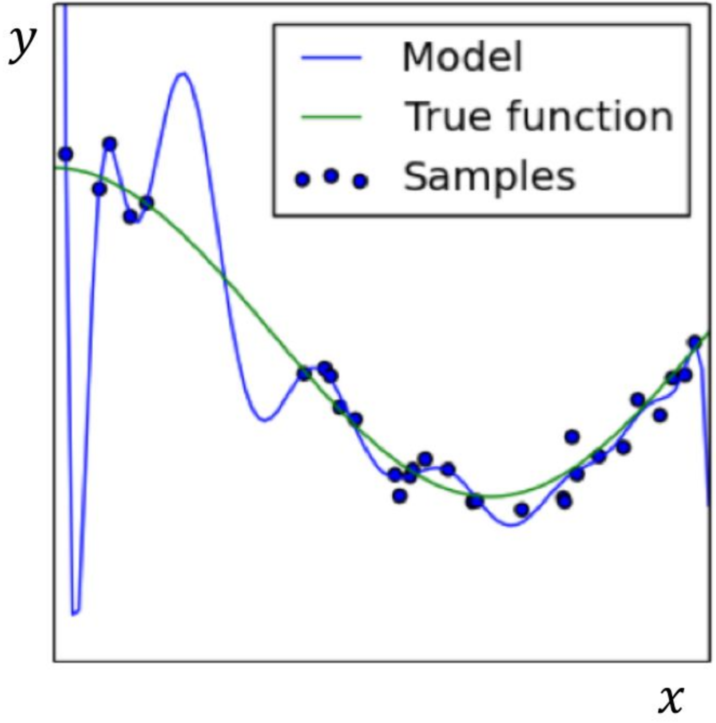
Model: $a(x) = b + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4$



Overfitting

Training set: $X \subset \mathbb{R}$

Model: $a(x) = b + w_1 x + w_2 x^2 + \cdots + w_{15} x^{15}$



Chain rule

- We know derivatives for simple functions:

$$\frac{dx^2}{dx} = 2x \qquad \frac{de^x}{dx} = e^x \qquad \frac{d\ln(x)}{dx} = \frac{1}{x}$$

- Let's take a composite function:

$$z_1 = z_1(x_1, x_2)$$

$$z_2 = z_2(x_1, x_2) \qquad \text{where } z_1, z_2, p \text{ are differentiable}$$

$$p = p(z_1, z_2)$$

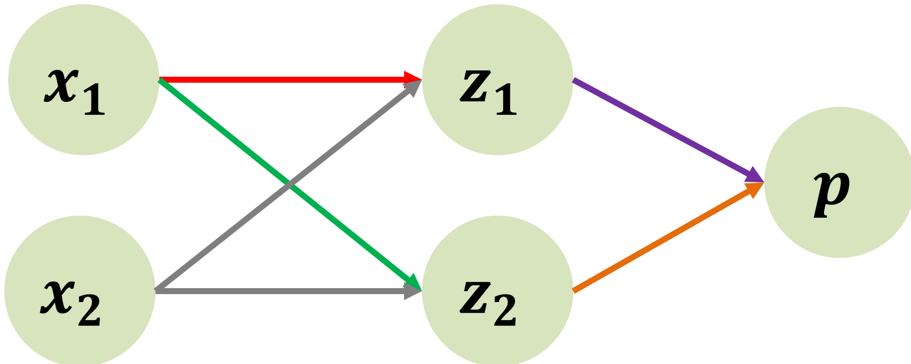
Chain rule: $$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

Example for $h(x) = f(x)g(x)$:

$$\frac{\partial h}{\partial x} = \frac{\partial h}{\partial f}\frac{\partial f}{\partial x} + \frac{\partial h}{\partial g}\frac{\partial g}{\partial x} = g\frac{\partial f}{\partial x} + f\frac{\partial g}{\partial x}$$

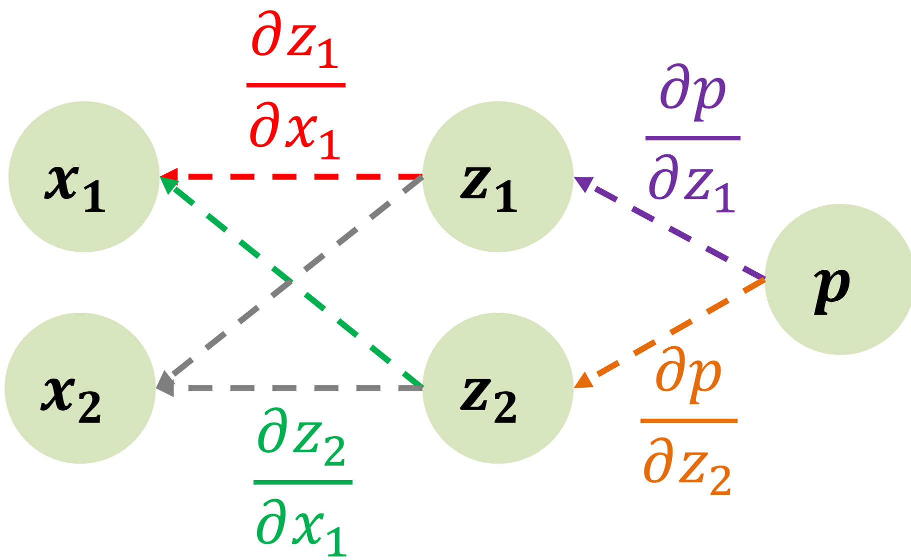
Derivatives computation graph

- Let's take our simple computation graph:



$$z_1 = z_1(x_1, x_2)$$
$$z_2 = z_2(x_1, x_2)$$
$$p = p(z_1, z_2)$$

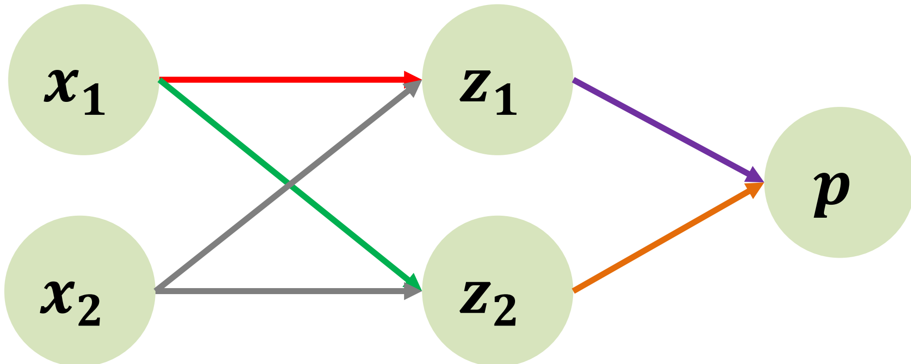
- And construct a new graph of derivatives:



Each edge is assigned to derivative of origin w.r.t. destination

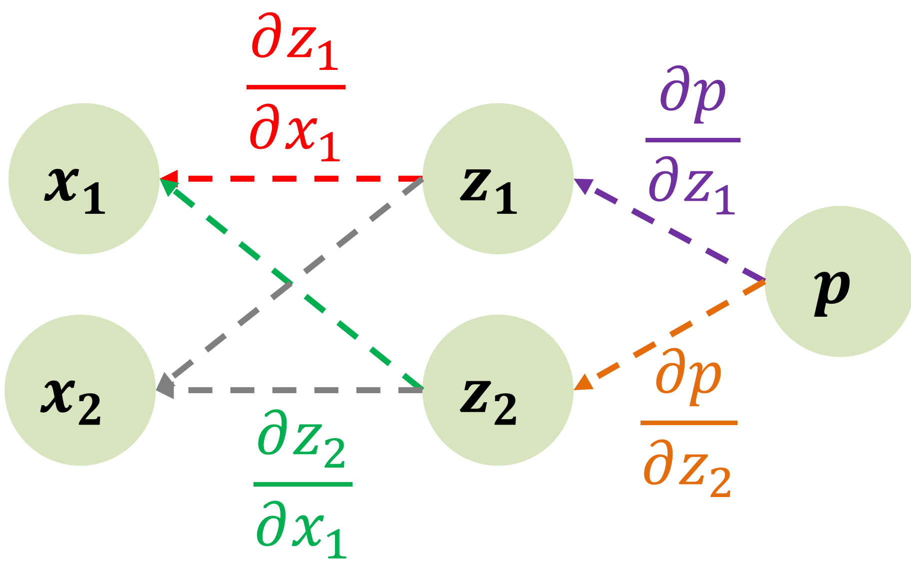
Derivatives computation graph

- Let's take our simple computation graph:



$$z_1 = z_1(x_1, x_2)$$
$$z_2 = z_2(x_1, x_2)$$
$$p = p(z_1, z_2)$$

- And construct a new graph of derivatives:

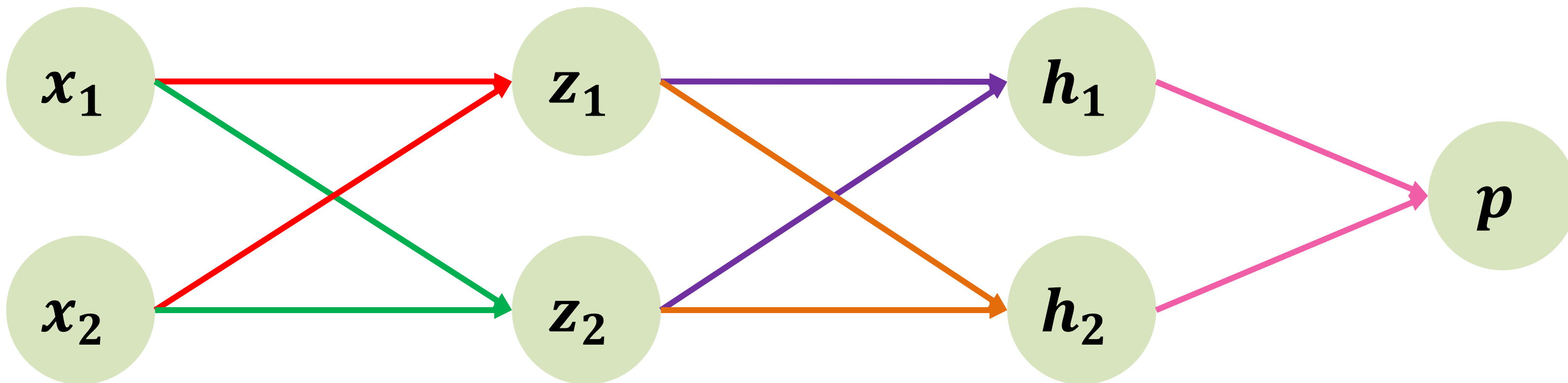


$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

You can see how
a **chain rule** works

Let’s go deeper

- A little bit more composite function:



$$z_1 = z_1(x_1, x_2) \qquad h_1 = h_1(z_1, z_2)$$

$$z_2 = z_2(x_1, x_2) \qquad h_2 = h_2(z_1, z_2)$$

$$p = p(h_1, h_2)$$

Let's go deeper

Chain rule: $$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial x_1}$$

Let's go deeper

Chain rule: $$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\boxed{\frac{\partial h_1}{\partial x_1}} + \frac{\partial p}{\partial h_2}\boxed{\frac{\partial h_2}{\partial x_1}}$$

$$\boxed{\frac{\partial h_1}{\partial x_1} = \frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1}}$$

$$\boxed{\frac{\partial h_2}{\partial x_1} = \frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}}$$

Let's go deeper

Chain rule:

$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\boxed{\frac{\partial h_1}{\partial x_1}} + \frac{\partial p}{\partial h_2}\boxed{\frac{\partial h_2}{\partial x_1}}$$

$$\boxed{\frac{\partial h_1}{\partial x_1} = \frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1}}$$

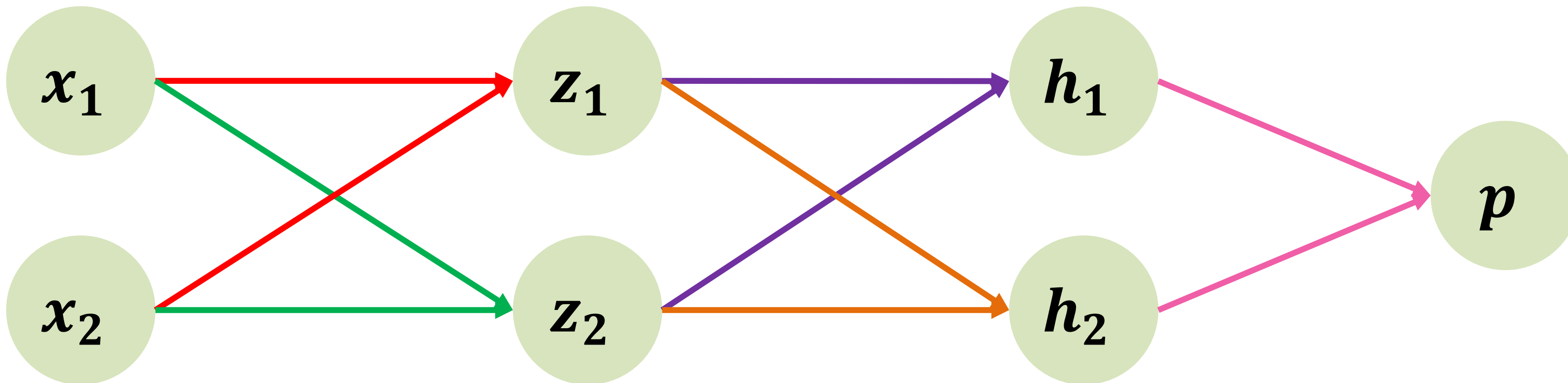
$$\boxed{\frac{\partial h_2}{\partial x_1} = \frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}}$$

$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\boxed{\left(\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1}\right)} + \frac{\partial p}{\partial h_2}\boxed{\left(\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}\right)}$$

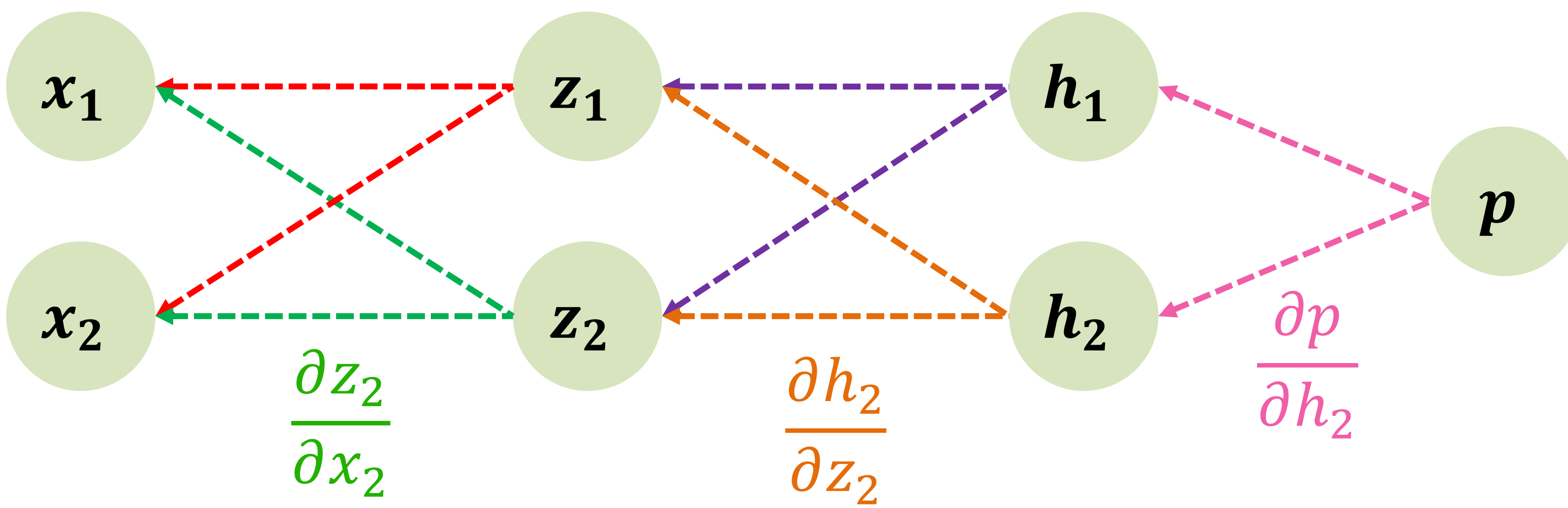
$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

Let's check out the derivatives graph!

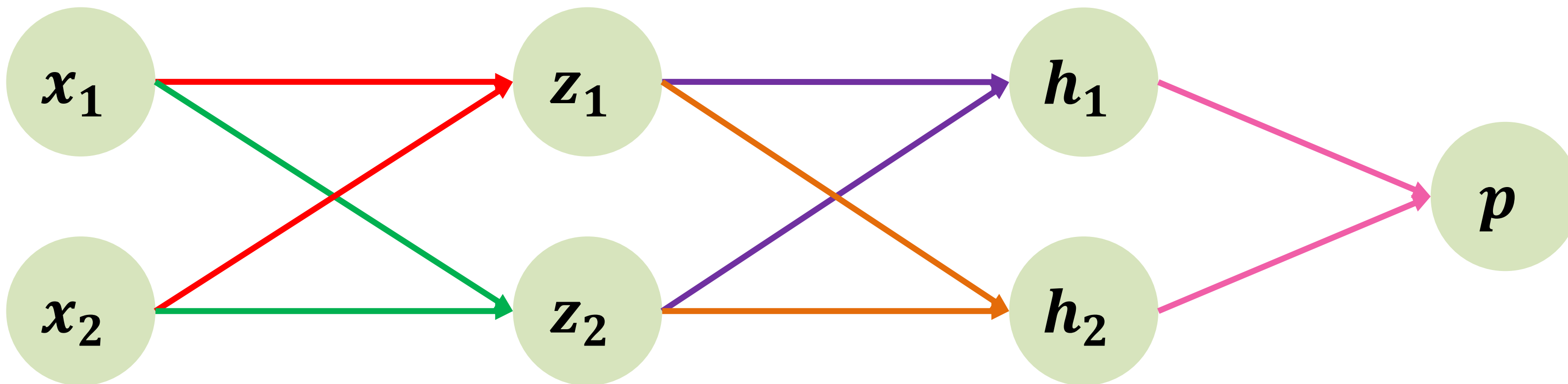
Let's go deeper



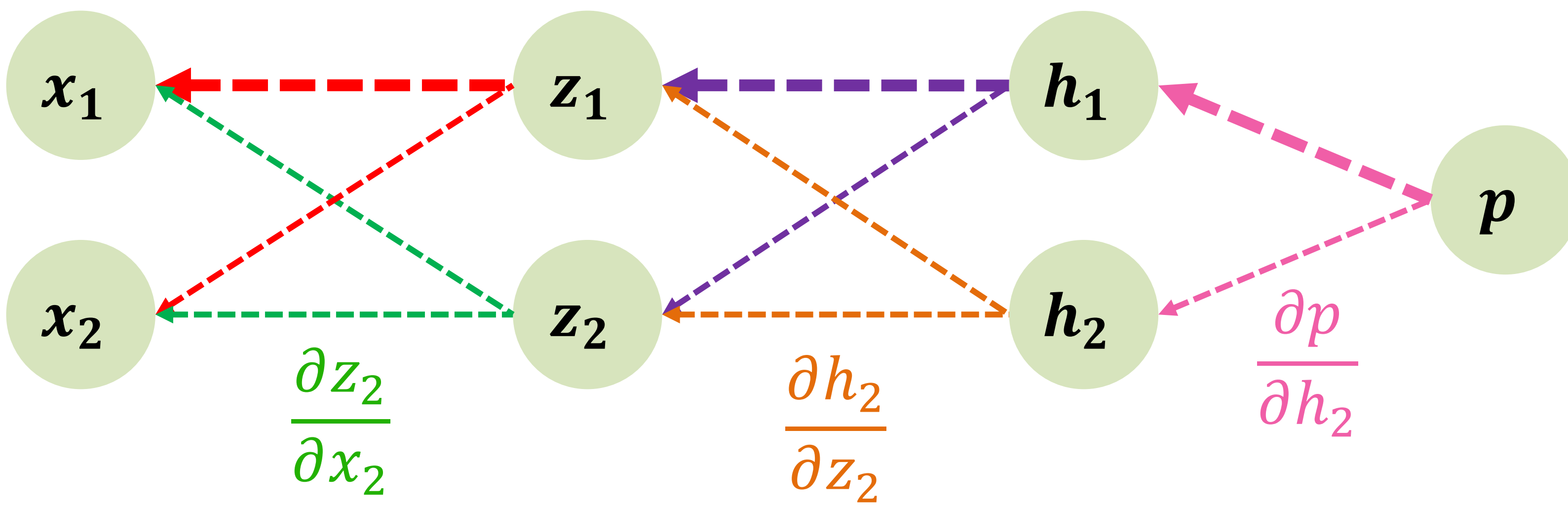
$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$



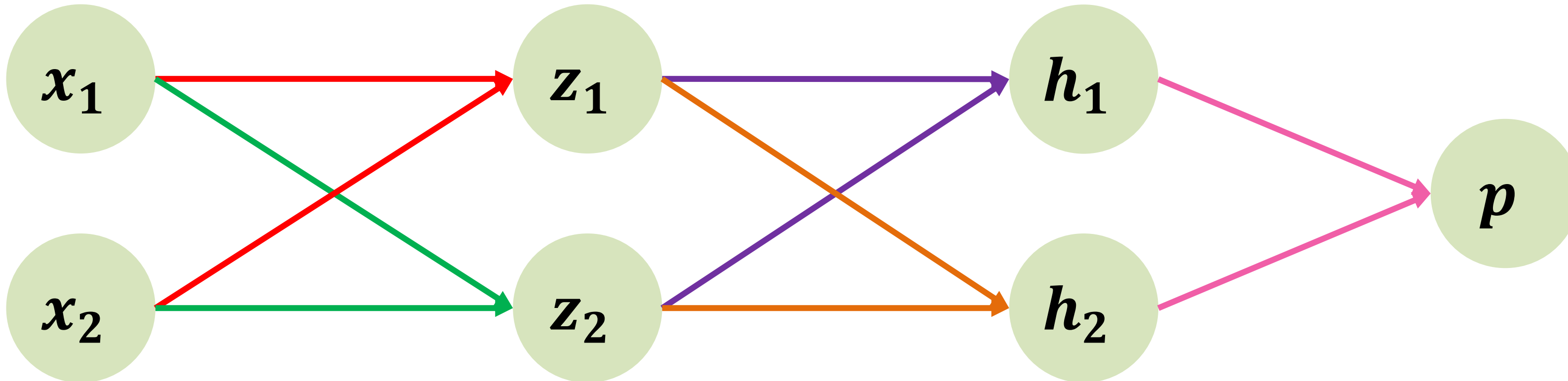
Let's go deeper



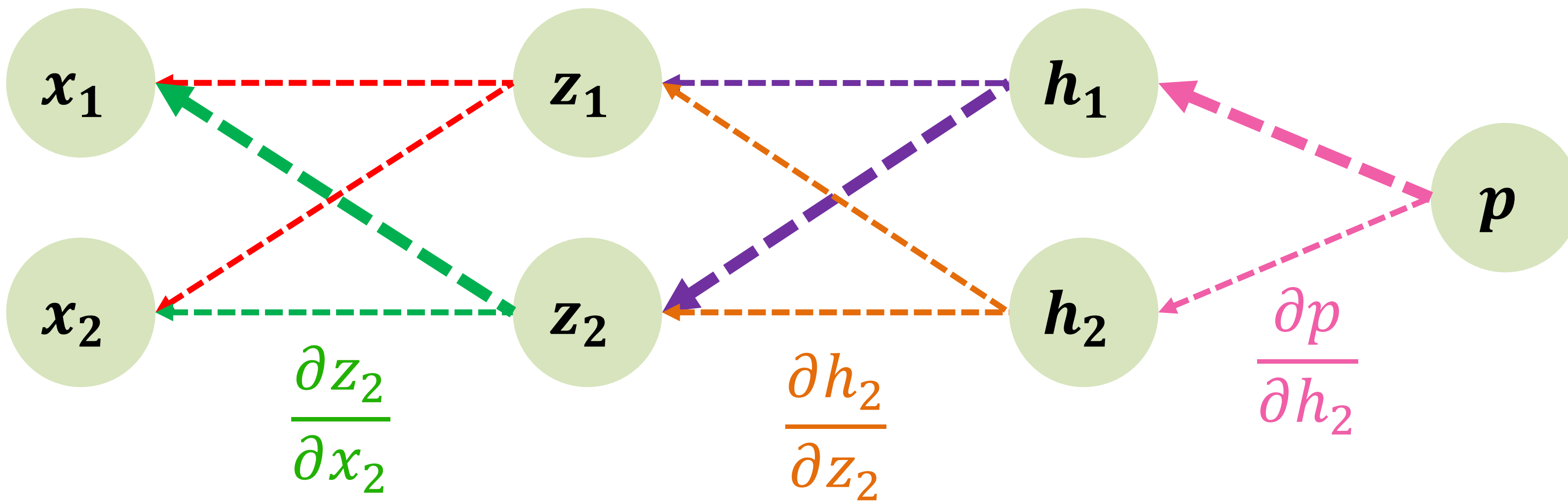
$$\frac{\partial p}{\partial x_1} = \boxed{\frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1}} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$



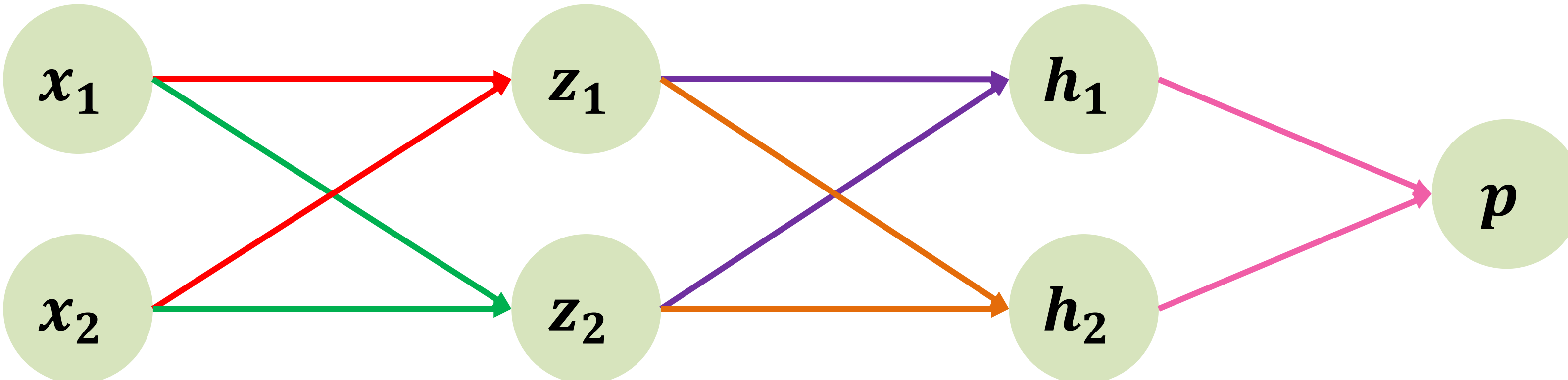
Let’s go deeper



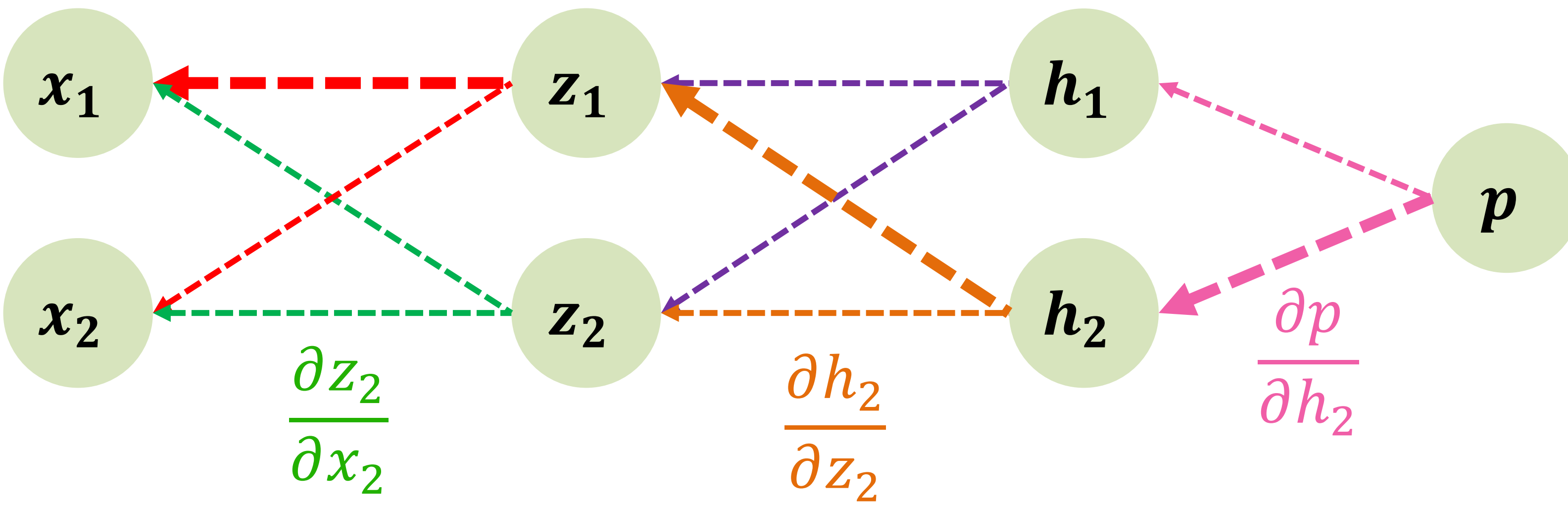
$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \boxed{\frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1}} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$



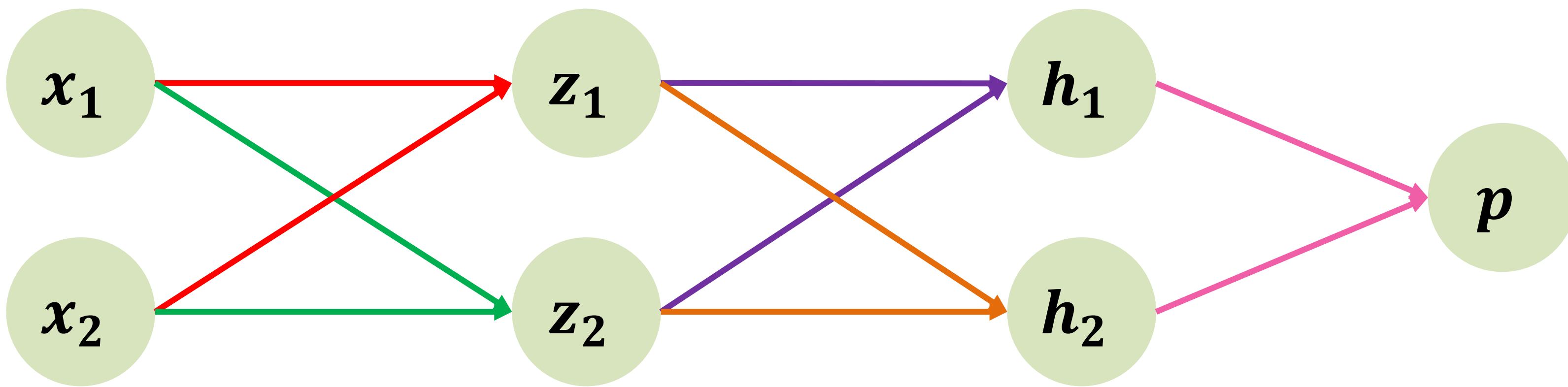
Let's go deeper



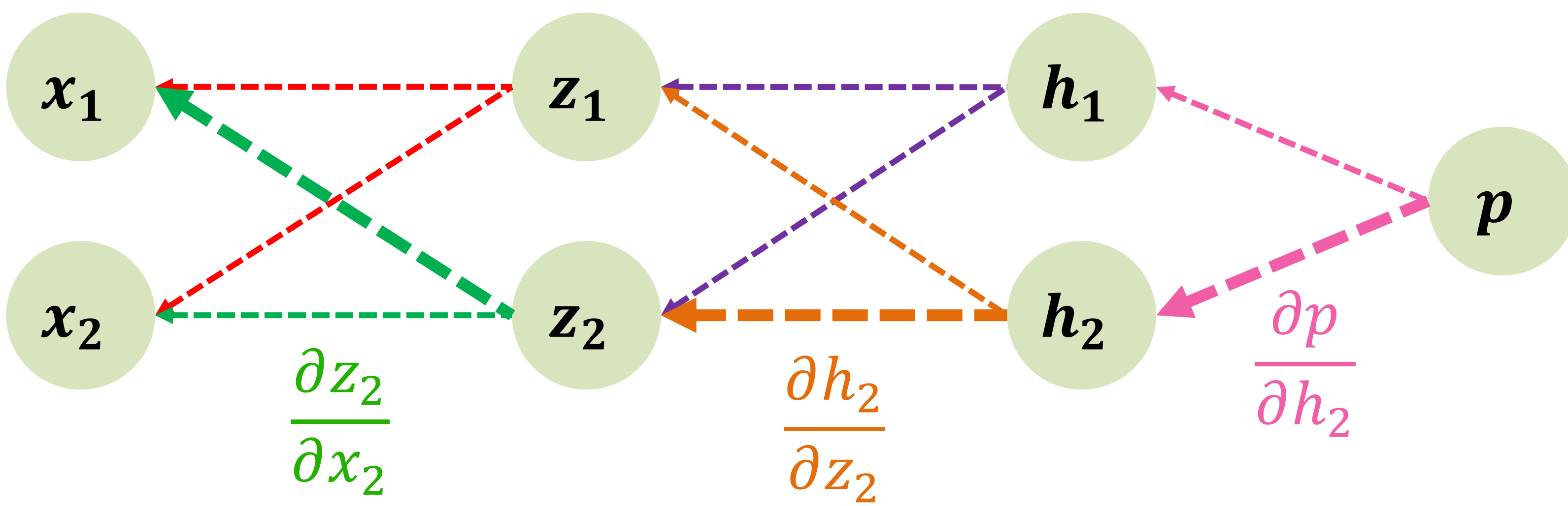
$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \boxed{\frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1}} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$



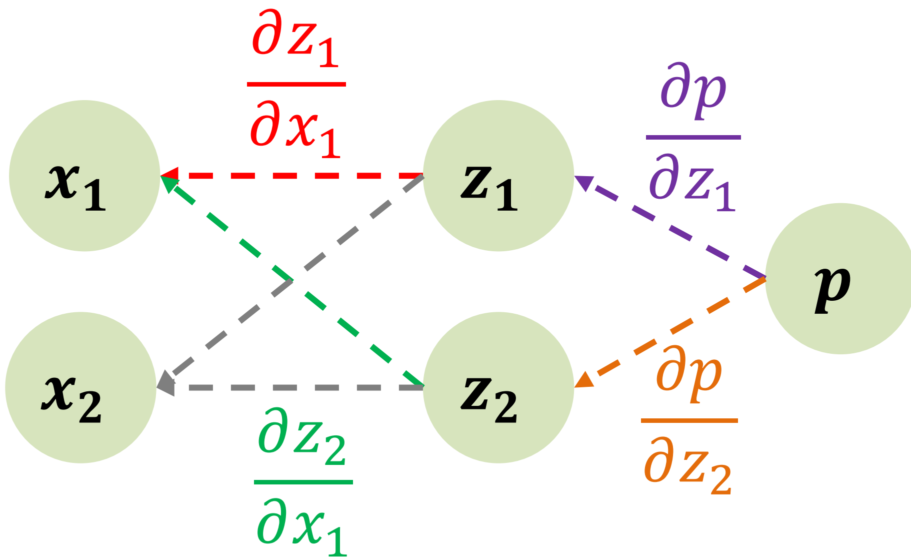
Let's go deeper



$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \boxed{\frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}}$$



How this graph of derivatives helps

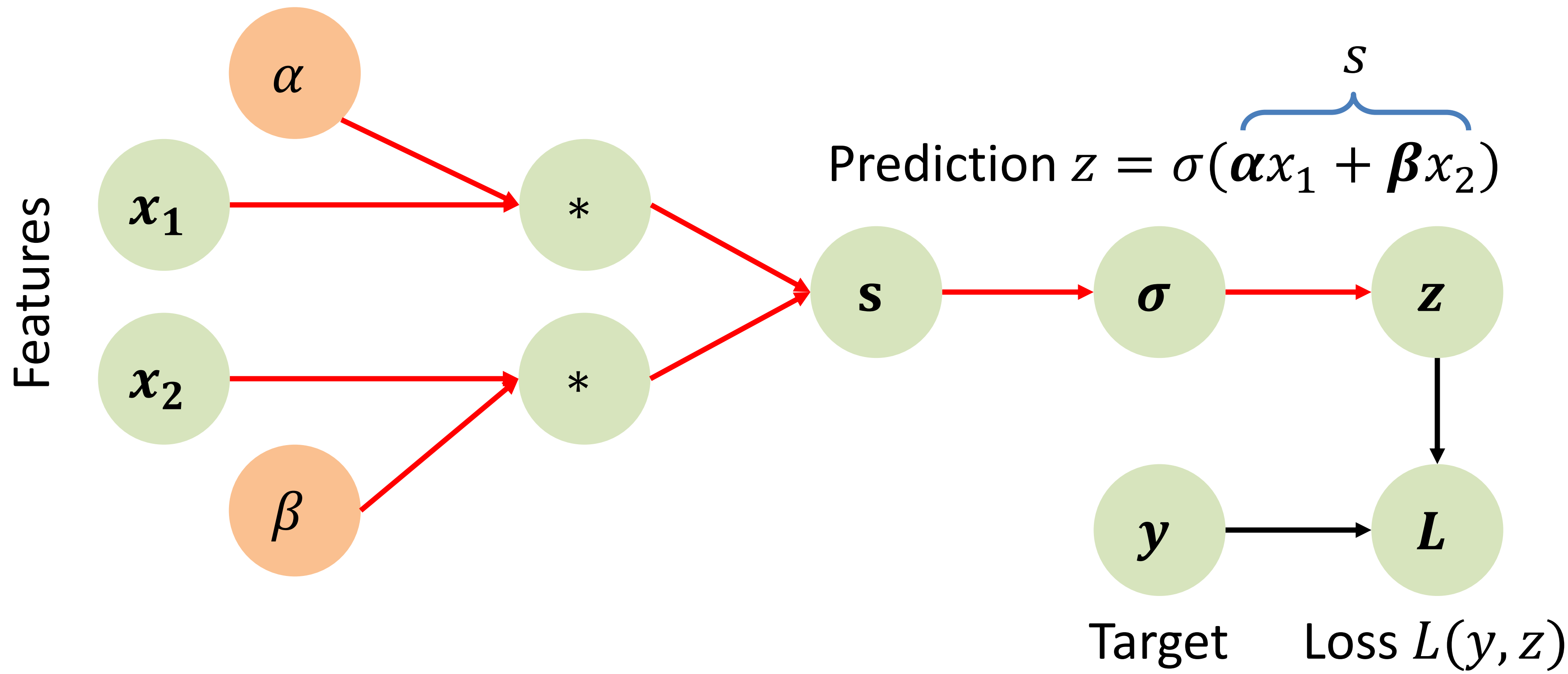


$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

How to calculate a derivative of node a w.r.t. node b:

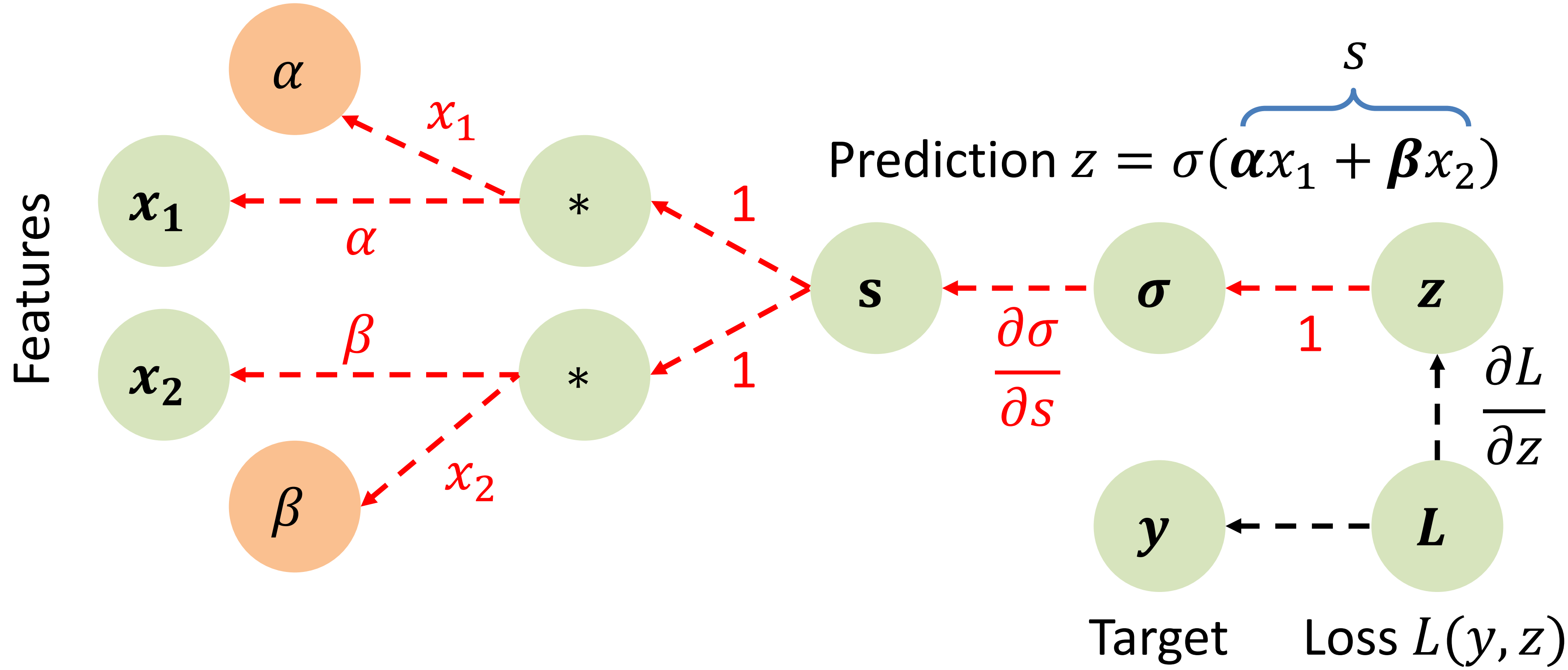
- Find an unvisited path from a to b
- Multiply all edge values along this path
- Add to the resulting derivative

How chain rule helps to train a neuron



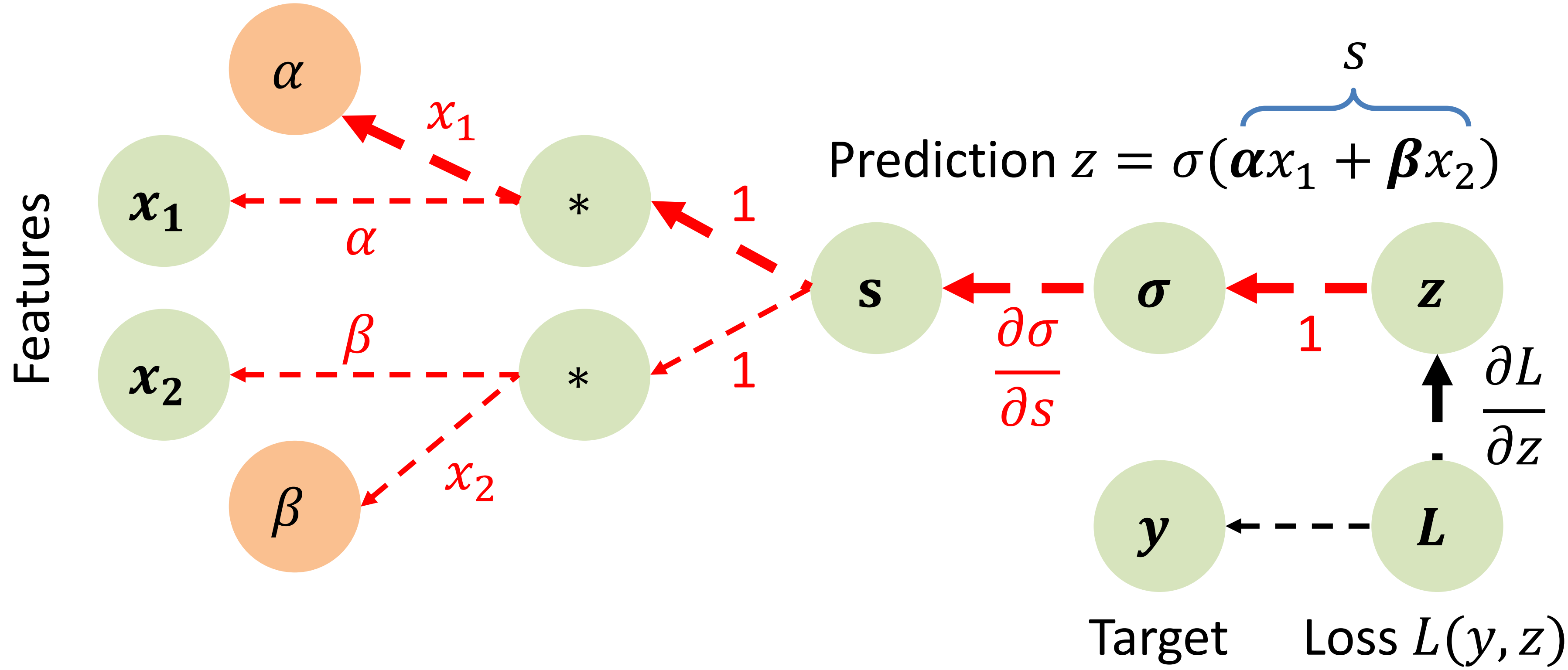
For SGD to work we need $\dfrac{\partial L}{\partial \alpha}, \dfrac{\partial L}{\partial \beta}$

Derivatives computation graph



For SGD to work we need $\frac{\partial L}{\partial \alpha}, \frac{\partial L}{\partial \beta}$

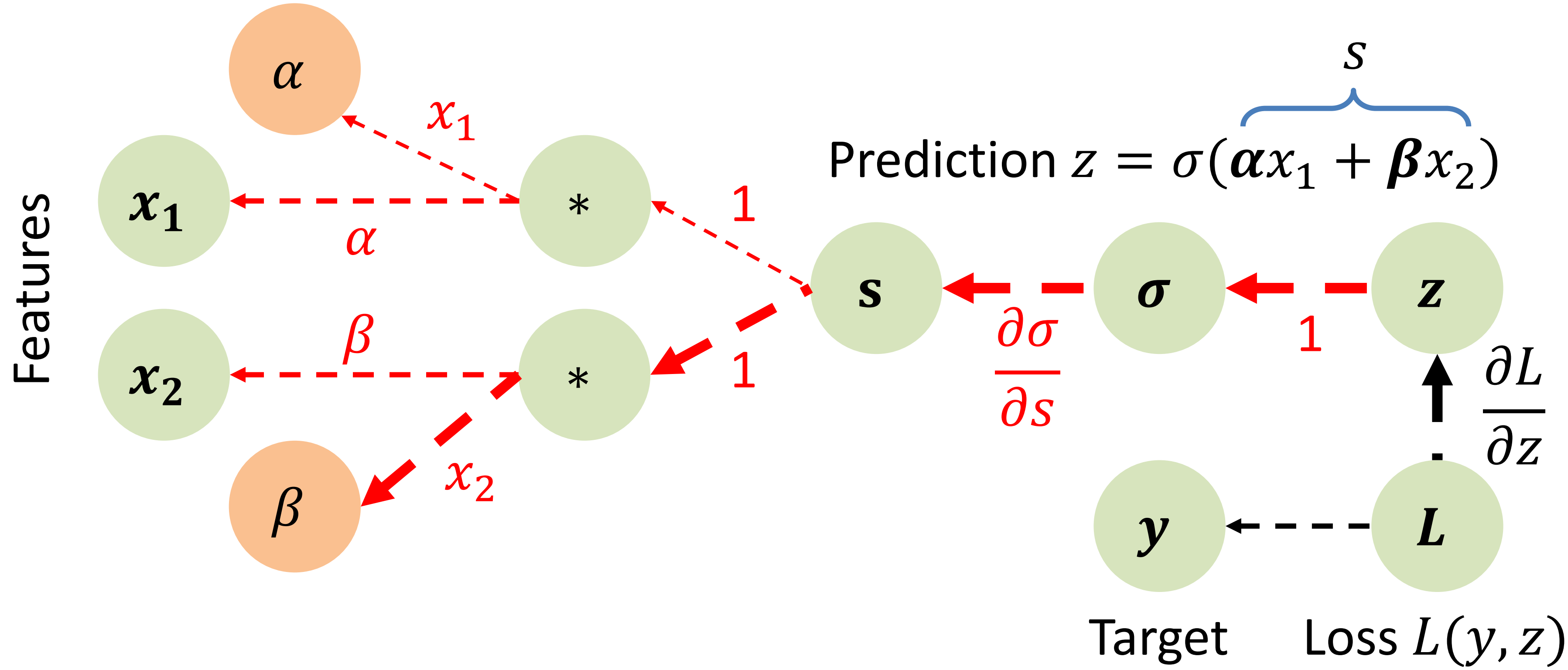
Derivatives computation graph



For SGD to work we need $\frac{\partial L}{\partial \alpha}, \frac{\partial L}{\partial \beta}$

$$\frac{\partial L}{\partial \alpha} = \frac{\partial L}{\partial z} \frac{\partial \sigma}{\partial s} x_1$$

Derivatives computation graph



For SGD to work we need $\frac{\partial L}{\partial \alpha}, \frac{\partial L}{\partial \beta}$

$$\frac{\partial L}{\partial \alpha} = \frac{\partial L}{\partial z} \frac{\partial \sigma}{\partial s} x_1 \qquad \frac{\partial L}{\partial \beta} = \frac{\partial L}{\partial z} \frac{\partial \sigma}{\partial s} x_2$$

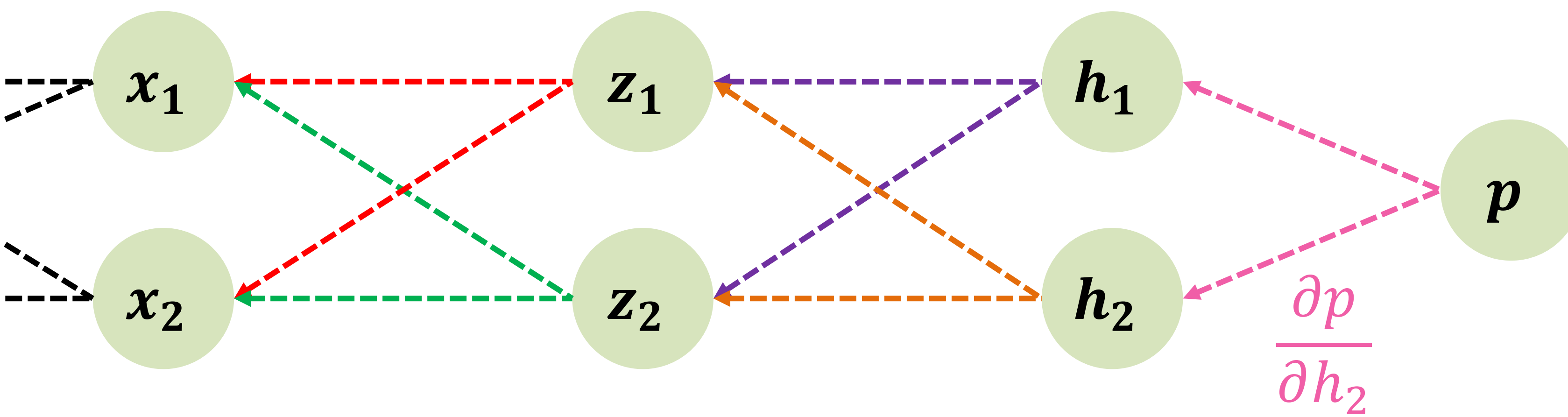
Let's look at MLP with 3 hidden layers

Let's drill down to an actual parameter of MLP:

$$h_2 = \sigma(\boldsymbol{w_0} + \boldsymbol{w_1} z_1 + \boldsymbol{w_2} z_2)$$

Gradient Descent:

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial p}\frac{\partial p}{\partial w_1} = \frac{\partial L}{\partial p}\frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial w_1}$$



We need to do this efficiently!

3: $\frac{\partial p}{\partial h_1}$ $\frac{\partial p}{\partial h_2}$

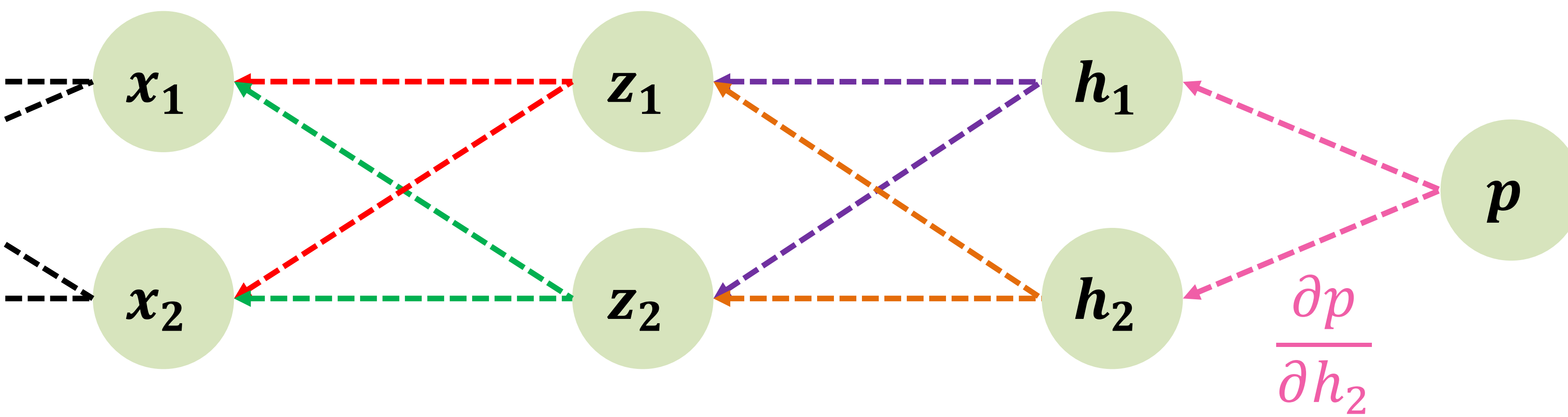
We will need these for GD

2: $\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$ $\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$

1:

$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\boxed{\frac{\partial p}{\partial h_1}}$ $\boxed{\frac{\partial p}{\partial h_2}}$

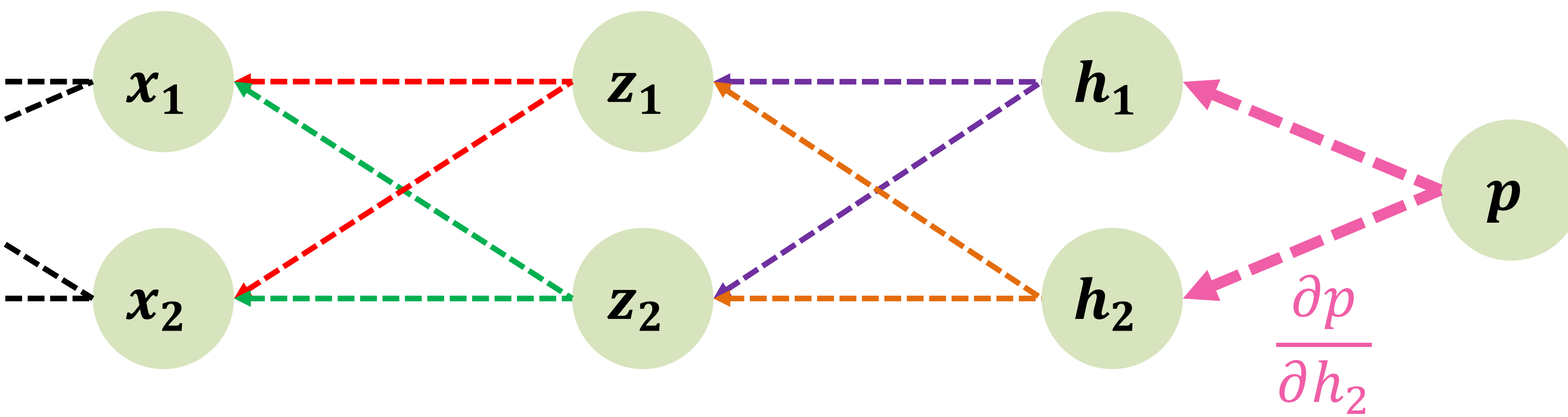
We will need these for GD

2: $$\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$$

$$\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$$

1: $$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\frac{\partial p}{\partial h_1}$ $\frac{\partial p}{\partial h_2}$

We will need these for GD

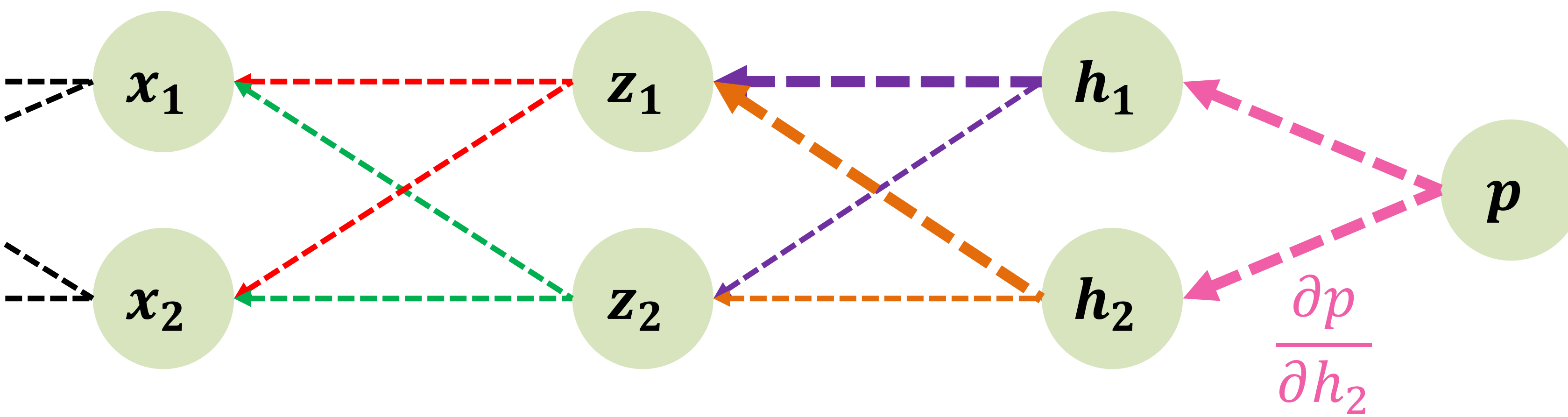
2: $$\boxed{\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}}$$

$$\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$$

1:

$$\frac{\partial p}{\partial x_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\frac{\partial p}{\partial h_1}$ $\quad$ $\frac{\partial p}{\partial h_2}$

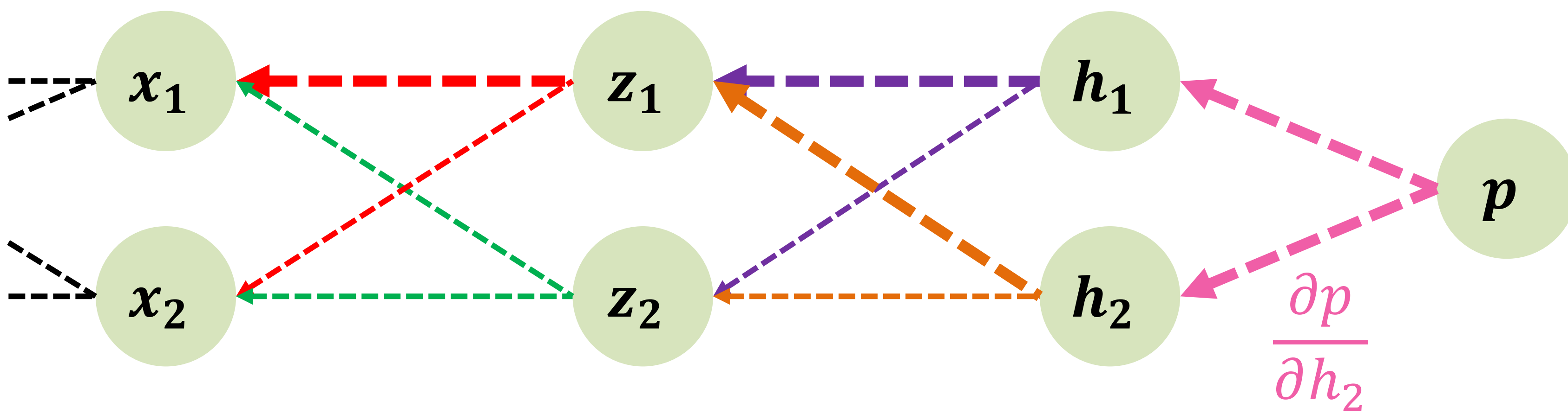
We will need these for GD

2: $\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$ $\qquad$ $\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$

1:

$$\frac{\partial p}{\partial x_1} = \boxed{\frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_1}} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\frac{\partial p}{\partial h_1}$ $\frac{\partial p}{\partial h_2}$

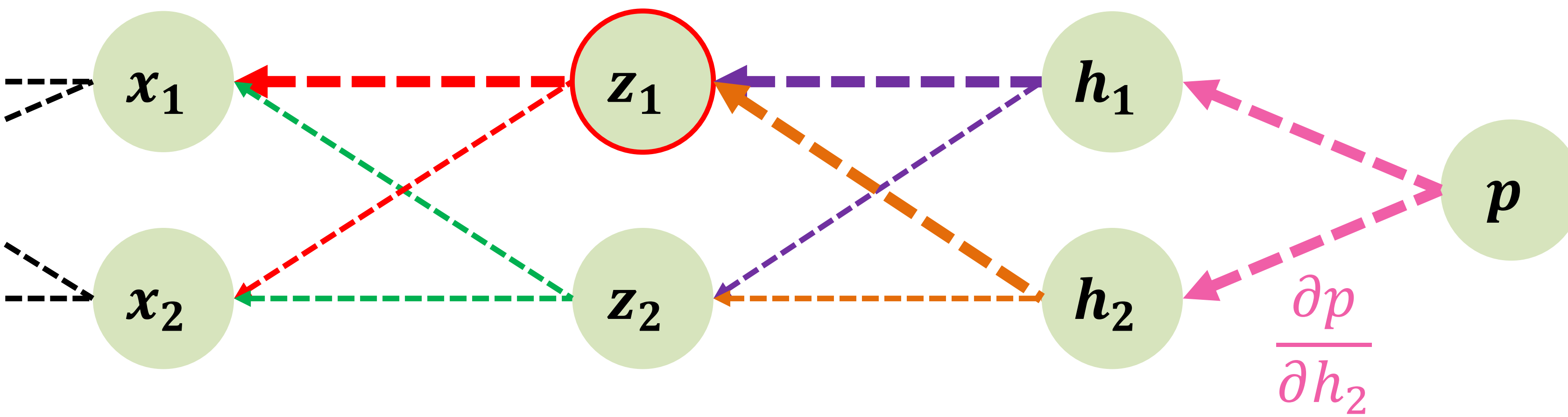
We will need these for GD

2: $\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$ $\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$

1:

$$\frac{\partial p}{\partial x_1} = \boxed{\left(\frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\right)\frac{\partial z_1}{\partial x_1}} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\frac{\partial p}{\partial h_1}$ $\frac{\partial p}{\partial h_2}$

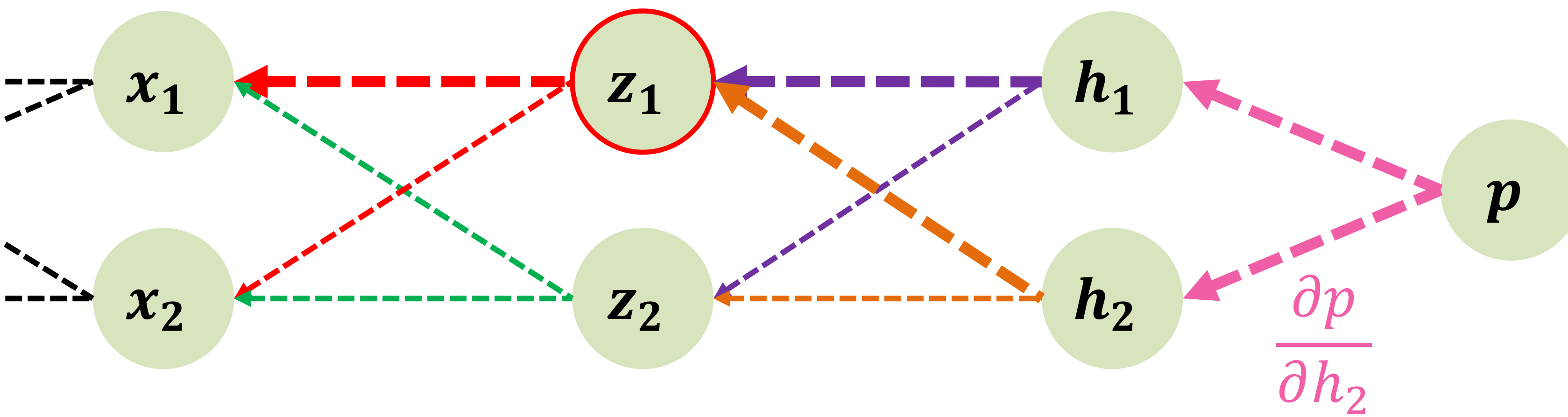
We will need these for GD

2: $\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$ $\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$

1:

$$\frac{\partial p}{\partial x_1} = \boxed{\left(\frac{\partial p}{\partial z_1}\right)\frac{\partial z_1}{\partial x_1}} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}\frac{\partial z_1}{\partial x_2} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



We can reuse previous computations

3: $\frac{\partial p}{\partial h_1}$ $\frac{\partial p}{\partial h_2}$

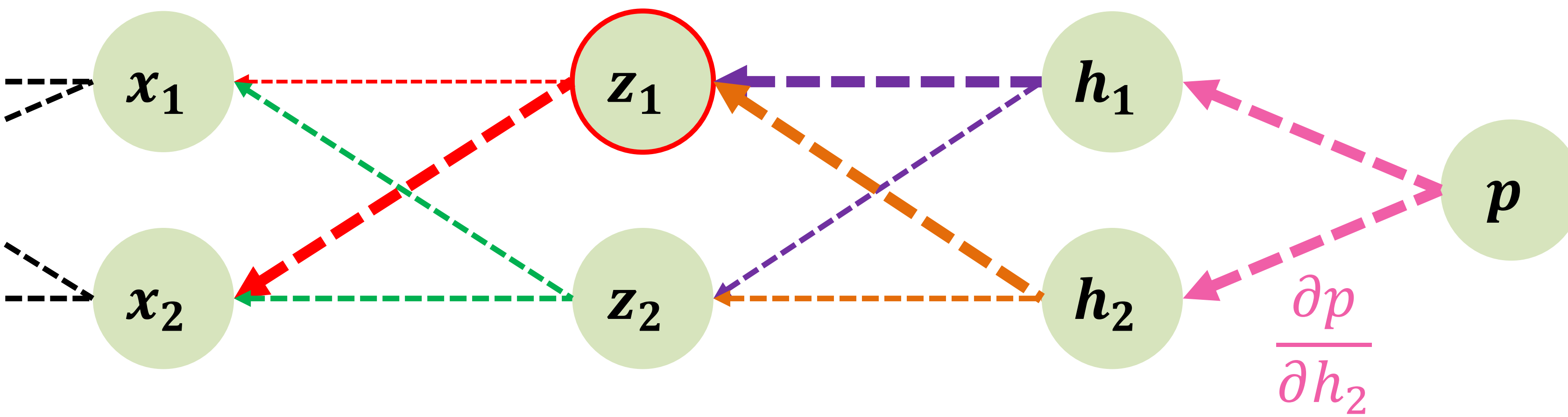
We will need these for GD

2: $\frac{\partial p}{\partial z_1} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_1}$ $\frac{\partial p}{\partial z_2} = \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}$

1:

$$\frac{\partial p}{\partial x_1} = \left(\frac{\partial p}{\partial z_1}\right)\frac{\partial z_1}{\partial x_1} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_1} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_1}$$

$$\frac{\partial p}{\partial x_2} = \boxed{\left(\frac{\partial p}{\partial z_1}\right)\frac{\partial z_1}{\partial x_2}} + \frac{\partial p}{\partial h_1}\frac{\partial h_1}{\partial z_2}\frac{\partial z_2}{\partial x_2} + \frac{\partial p}{\partial h_2}\frac{\partial h_2}{\partial z_2}\frac{\partial z_2}{\partial x_2}$$



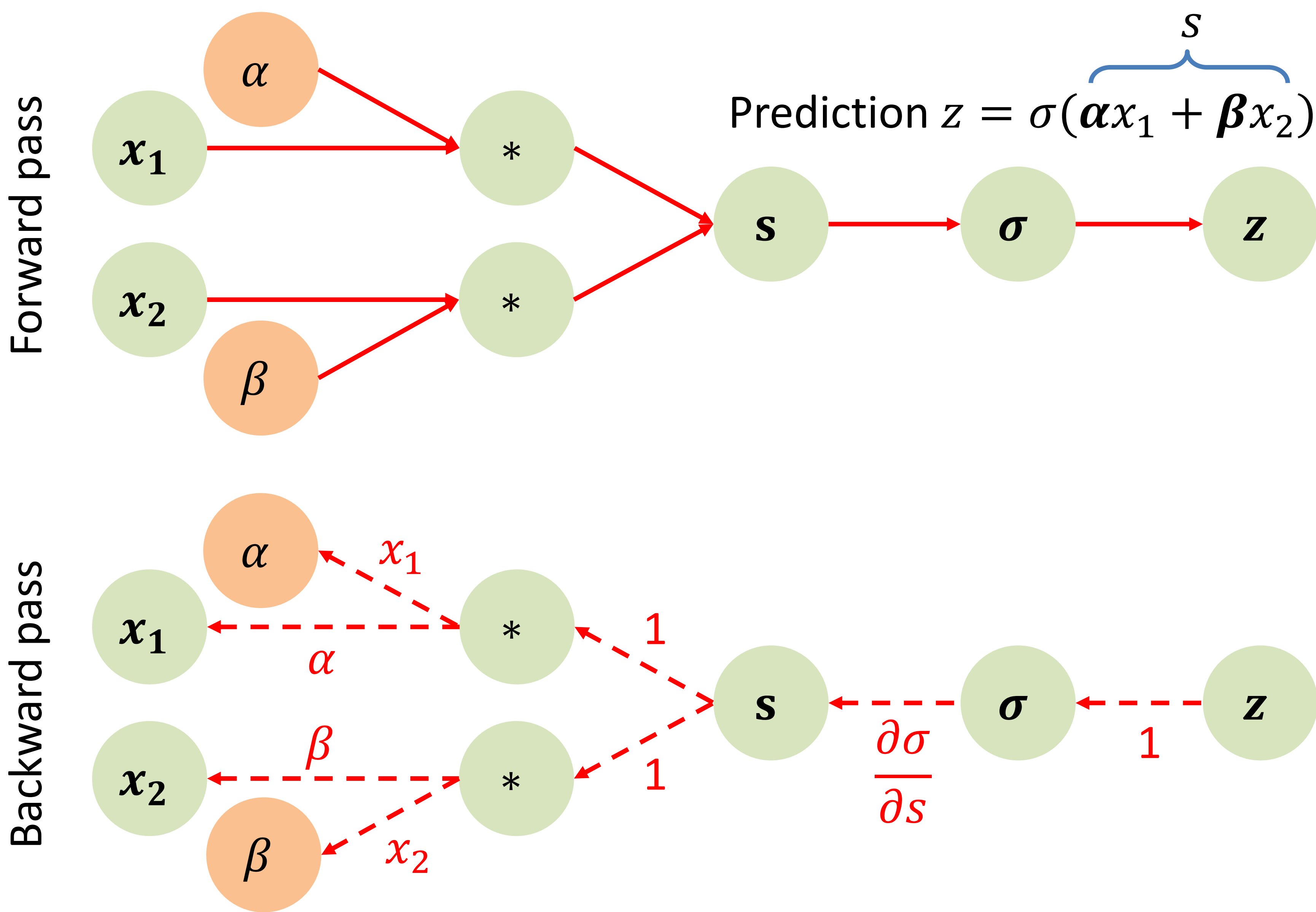
This is called reverse-mode differentiation

- In application to neural networks it has one more name: **back-propagation**.

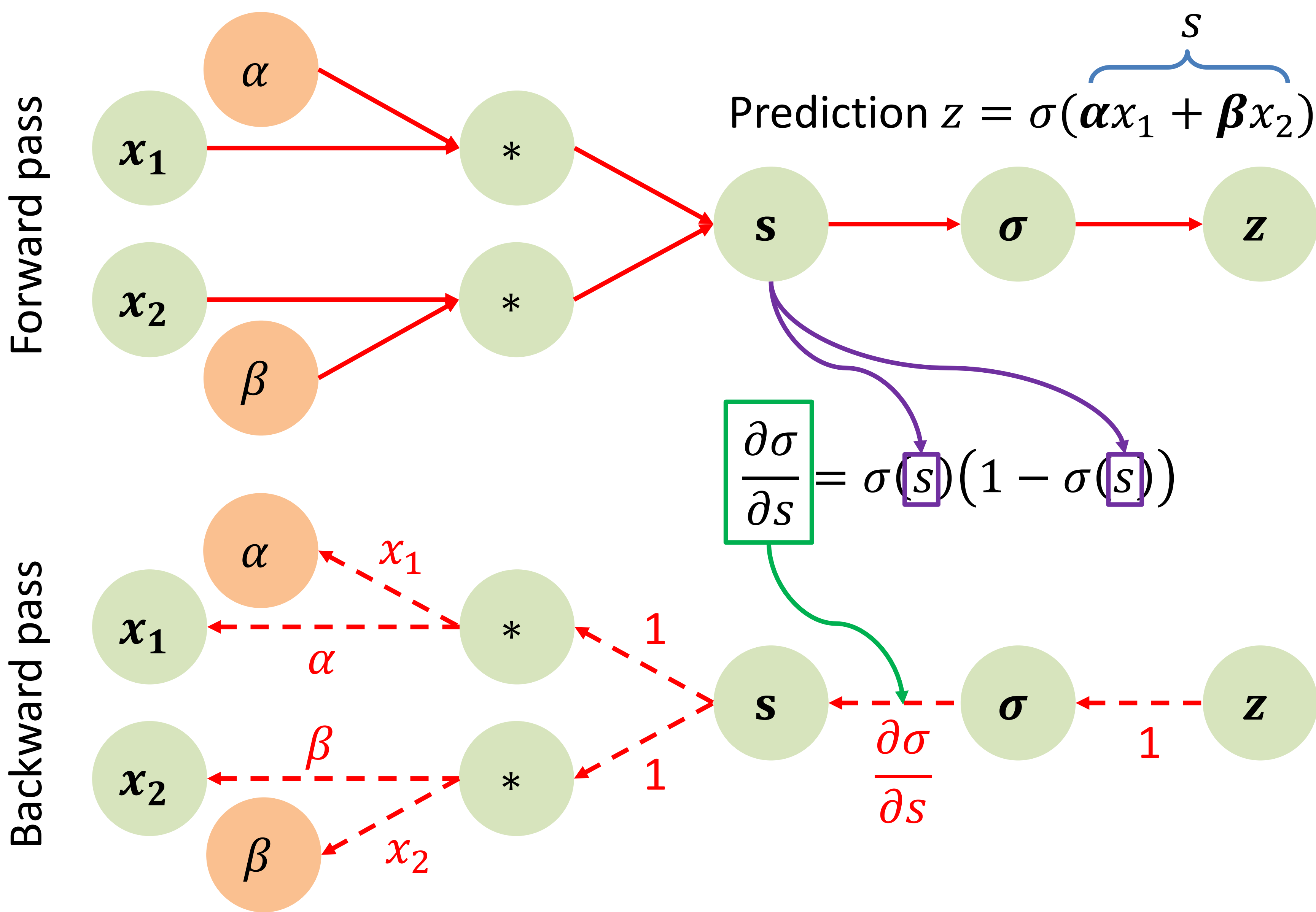
- It works **fast**, because we reuse computations from previous steps.

- In fact, for each edge we compute its value only once. And multiply by its value exactly once.

Back-propagation (Back-prop)

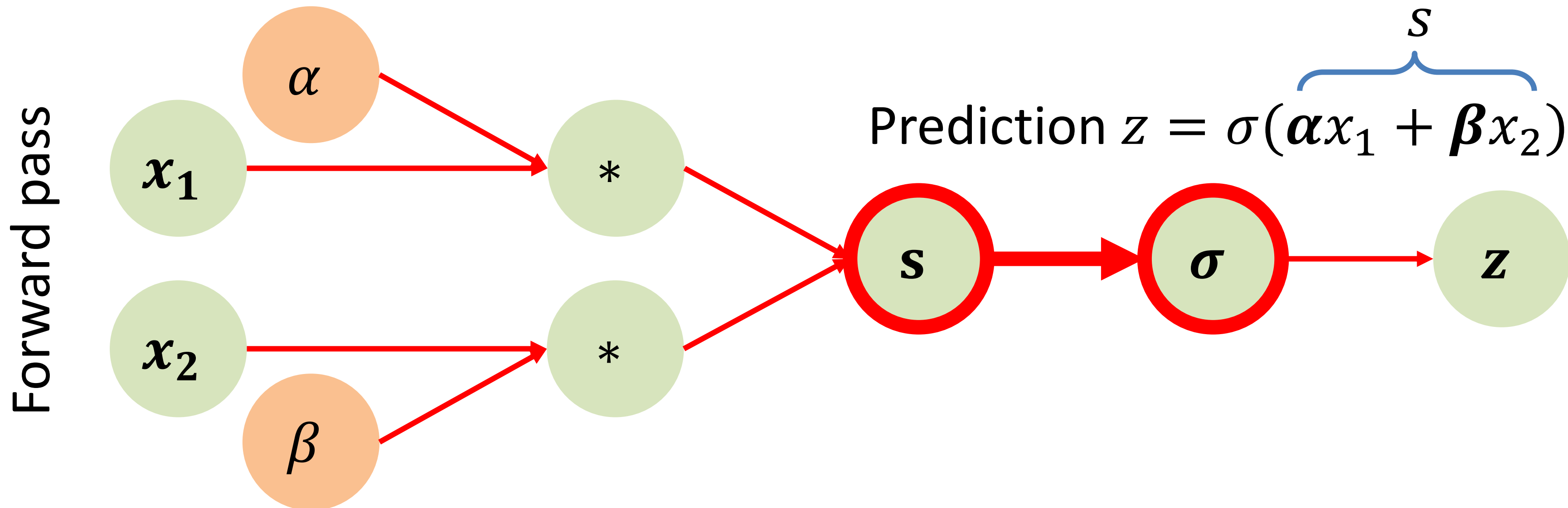


Back-propagation (Back-prop)



Forward pass interface

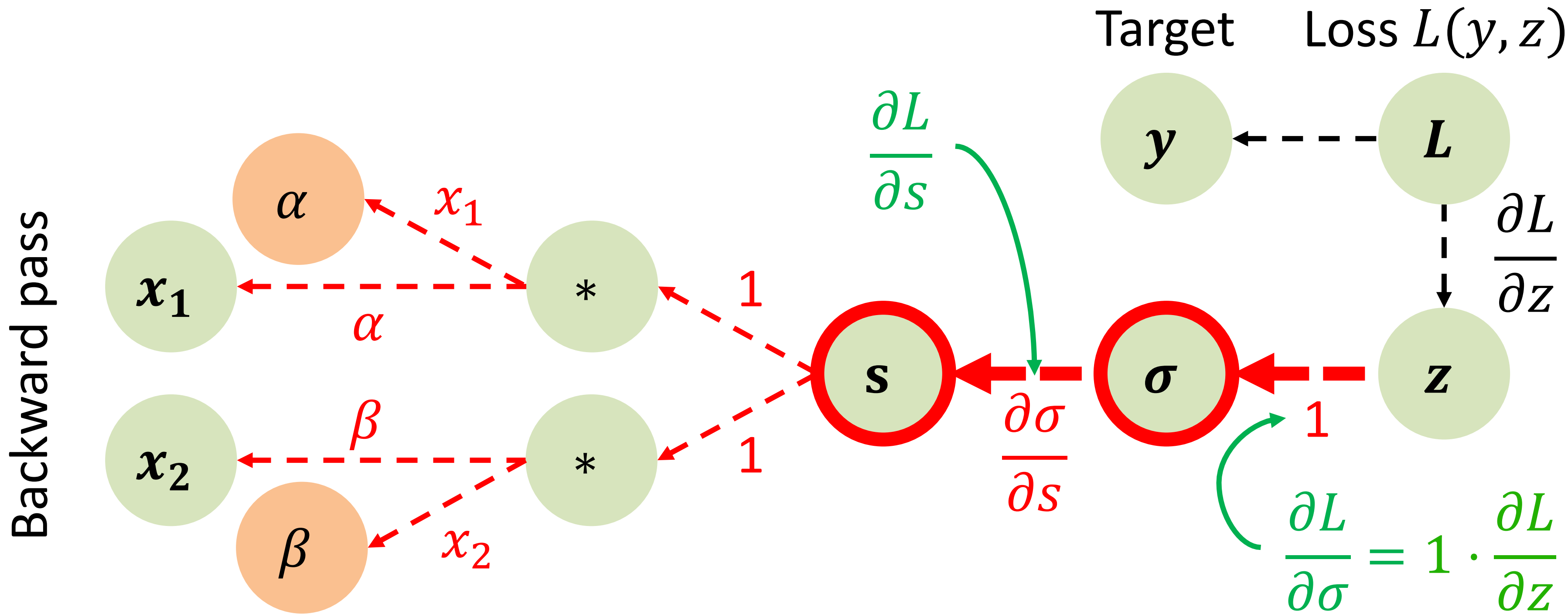
Let's implement a sigmoid activation node!



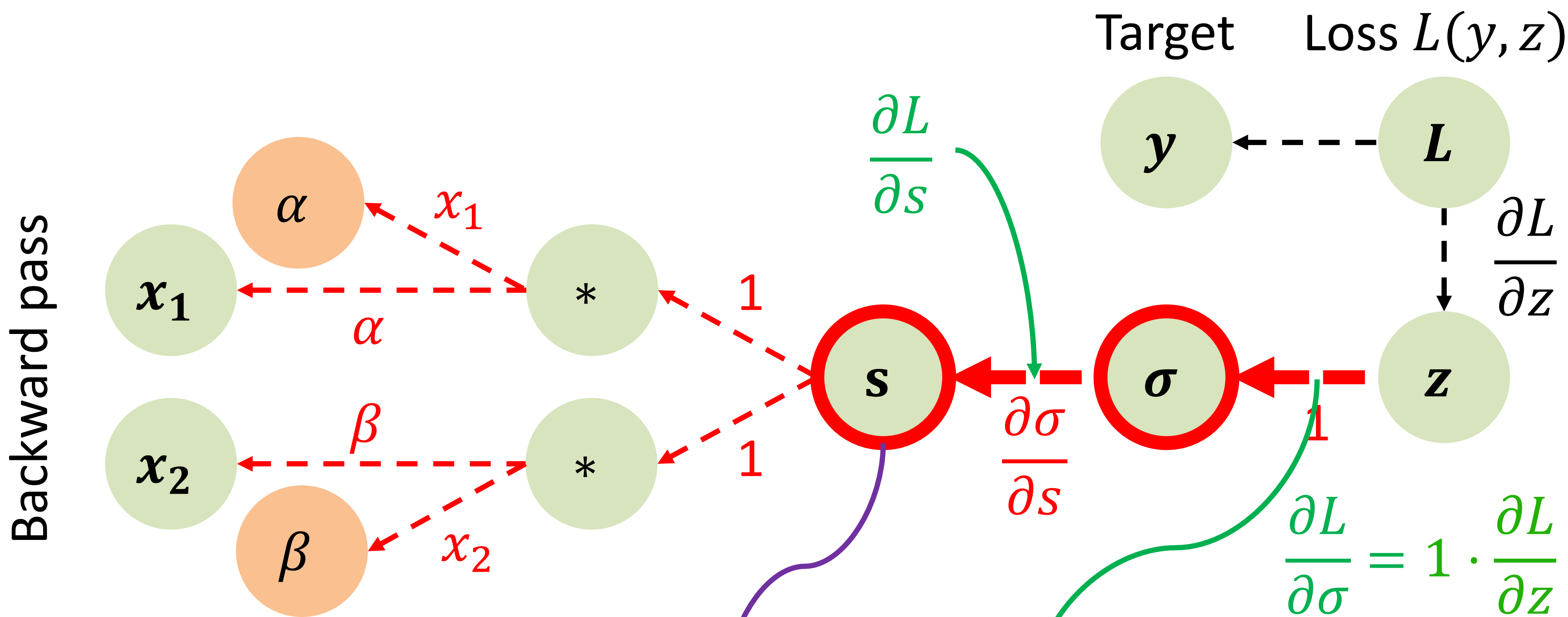
Prediction $z = \sigma(\underbrace{\boldsymbol{\alpha} x_1 + \boldsymbol{\beta} x_2}_{s})$

```
def forward_pass(inputs):
  return 1. / (1 + np.exp(-inputs))
```

Backward pass interface



Backward pass interface




```
def backward_pass(inputs, incoming_gradient):
    sigmoid = 1. / (1 + np.exp(-inputs))
    return sigmoid * (1 - sigmoid) * incoming_gradient
```

$$\frac{\partial L}{\partial s} = \frac{\partial \sigma}{\partial s} \cdot \frac{\partial L}{\partial \sigma}$$