IP Scoring Rules: Application

Jason Konek[a] Ben Levinstein[b]

19 August 2022

[a]Department of Philosophy
Bristol

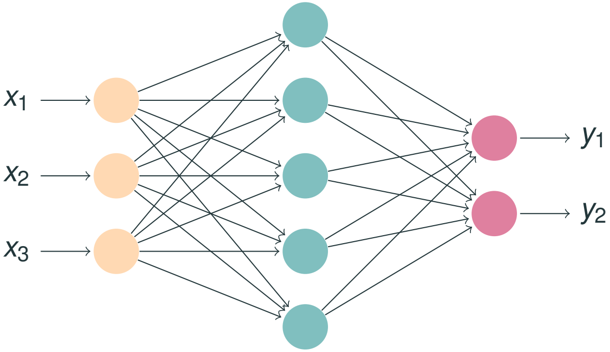
[b]Department of Philosophy
UIUC

Introduction

Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

Fashion-MNIST dataset

60,000 28x28 grayscale images

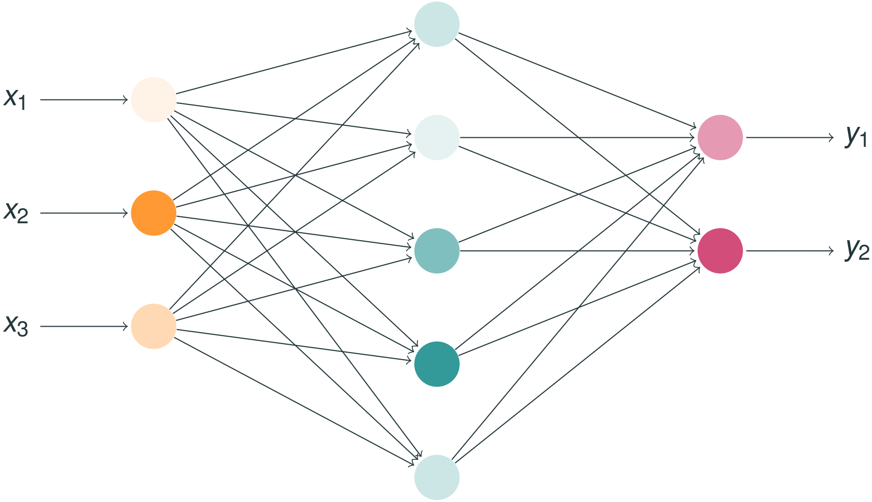


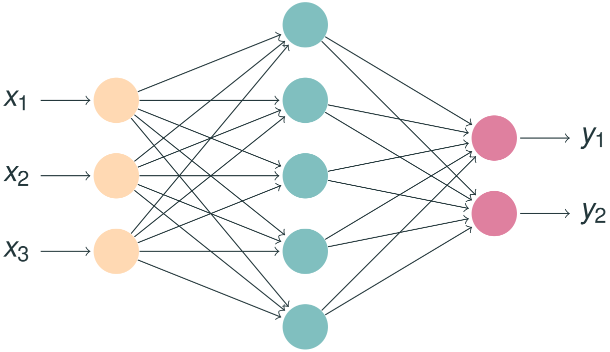
Input Layer:

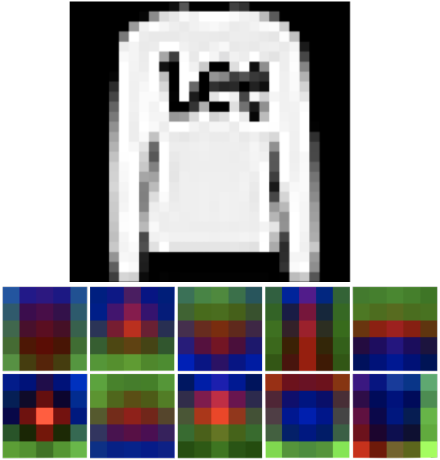
- 28x28=784 input nodes
- Take values between 0 (black) and 1 (white)

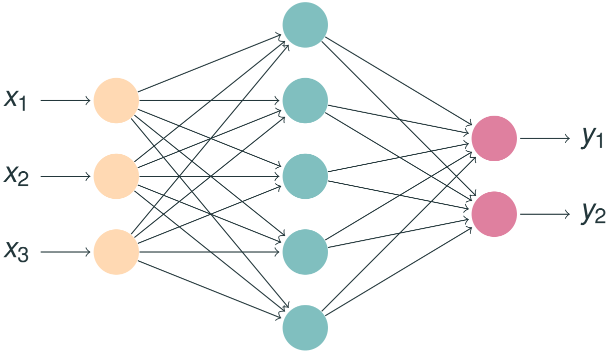
Output Layer:

- 10 output nodes
- Probabilities for different categories

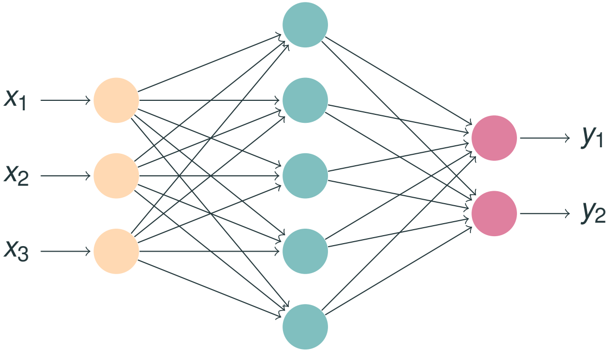

x_1
x_2
x_3
y_1
y_2


x_1
x_2
x_3
y_1
y_2

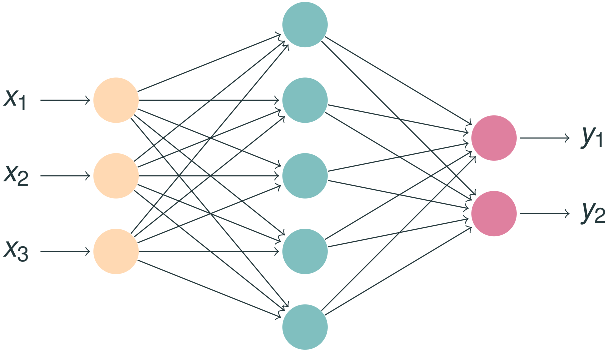




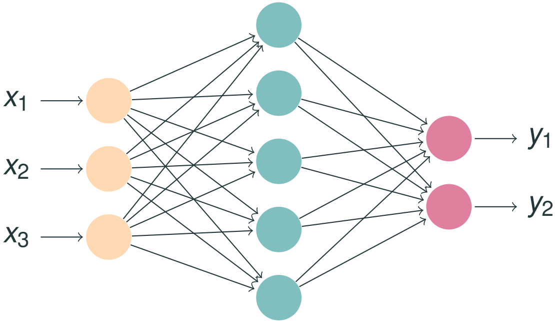
$$\begin{pmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \\ w_{4,1} & w_{4,2} & w_{4,3} \\ w_{5,1} & w_{5,2} & w_{5,3} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix}$$



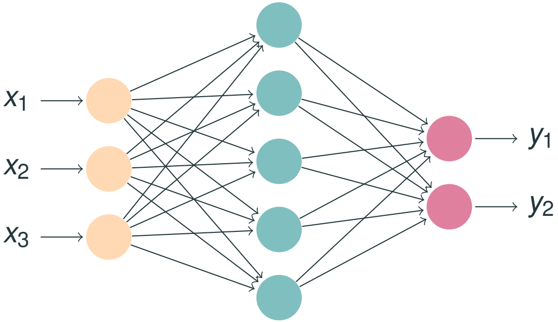
$$\begin{pmatrix} {\color{red}w_{1,1}} & {\color{red}w_{1,2}} & {\color{red}w_{1,3}} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \\ w_{4,1} & w_{4,2} & w_{4,3} \\ w_{5,1} & w_{5,2} & w_{5,3} \end{pmatrix} \begin{pmatrix} {\color{red}x_1} \\ {\color{red}x_2} \\ {\color{red}x_3} \end{pmatrix} = \begin{pmatrix} {\color{red}h_1} \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix}$$



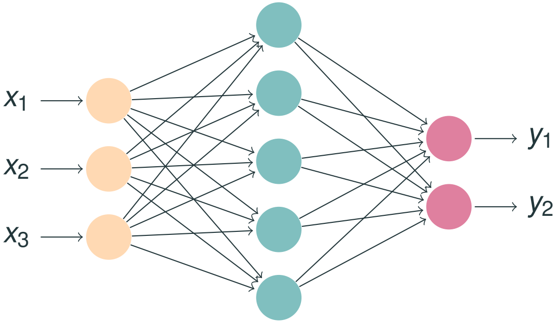
$$\begin{pmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \\ w_{4,1} & w_{4,2} & w_{4,3} \\ w_{5,1} & w_{5,2} & w_{5,3} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix}$$



$$\begin{pmatrix} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} \\ v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

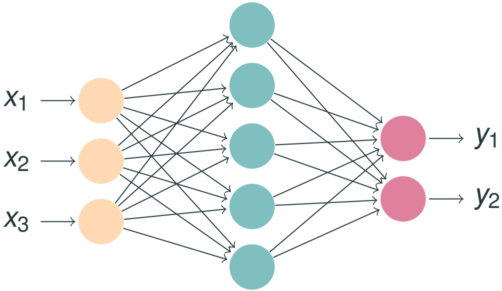


$$\begin{pmatrix} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} \\ v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$



$$\begin{pmatrix} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} \\ v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

Training



Output Layer ⇒ Probabilities:

- Softmax:
  $p_1 = \frac{e^{y_1}}{e^{y_1}+e^{y_2}}$, $p_2 = \frac{e^{y_2}}{e^{y_1}+e^{y_2}}$

Per-sample loss:

- Cross Entropy: $-Log(p_i)$, where i is the label of the sample

Loss:

Fix a batch of inputs/labels $\{\langle \mathrm{input}_1, \mathrm{label}_1 \rangle, \ldots, \langle \mathrm{input}_n, \mathrm{label}_n, \rangle\}$

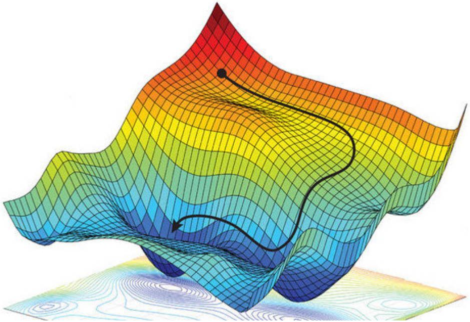
$$\mathcal{L}(w_{1,1}, \ldots, w_{5,3}, v_{1,1}, \ldots, v_{2,5}) = \frac{1}{n} \sum_{i \le n} -Log(p_{\mathrm{label}_i})$$

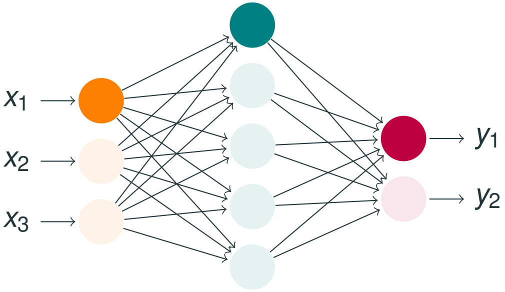
How to lower loss? Look at $\nabla \mathcal{L}$ to figure out how to nudge weights to improve as quickly as possible.

Let $w = \langle w_{1,1}, \ldots, w_{5,3}, v_{1,1}, \ldots, v_{2,5} \rangle$

Training:

- Gradient of $\mathcal{L}$ points in the direction steepest ascent
  - $\nabla \mathcal{L} = \left\langle \frac{\partial \mathcal{L}}{\partial w_{1,1}}, \ldots, \frac{\partial \mathcal{L}}{\partial v_{2,5}} \right\rangle$
- Nudge weights in opposite direction
  - $w - \epsilon \nabla \mathcal{L}(w)$
- Repeat
- Doing this will move some weights around much more than others (the ones that have a bigger impact on loss)





Output Layer ⇒ Probabilities:

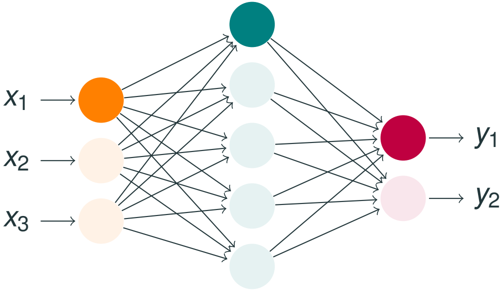
- Softmax: $p_1 = \frac{e^{y_1}}{e^{y_1}+e^{y_2}}$, $p_2 = \frac{e^{y_2}}{e^{y_1}+e^{y_2}}$

Per-sample loss:

- Cross Entropy: $-Log(p_i)$, where i is the label of the sample

Choose $\langle \text{input}, \text{label} \rangle$ with $\text{label}=1$

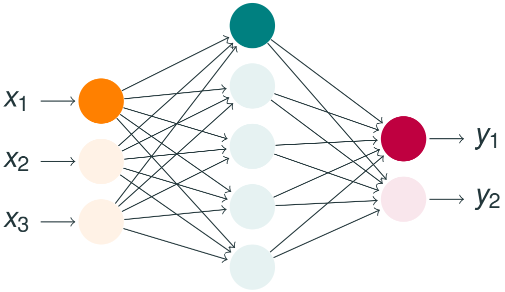
$$\mathcal{L}(y_1, y_2) = -Log(p_1)$$
$$= -Log(\frac{e^{y_1}}{e^{y_1} + e^{y_2}})$$
$$\frac{\partial \mathcal{L}}{\partial y_1} = -\frac{e^{y_2}}{e^{y_1} + e^{y_2}}$$



$$\begin{pmatrix} v_{1,1} & v_{1,2} & v_{1,3} & v_{1,4} & v_{1,5} \\ v_{2,1} & v_{2,2} & v_{2,3} & v_{2,4} & v_{2,5} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

$$y_1(v_{1,1}, v_{1,2}, v_{1,3}, v_{1,4}, v_{1,5}, h_1, h_2, h_3, h_4, h_5) = v_{1,1}h_1 + v_{1,2}h_2 + v_{1,3}h_3 + v_{1,4}h_4 + v_{1,5}h_5$$

$$\frac{\partial y_1}{\partial h_1} = v_{1,1}$$



$$\begin{pmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \\ w_{4,1} & w_{4,2} & w_{4,3} \\ w_{5,1} & w_{5,2} & w_{5,3} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{pmatrix}$$

$$h_1(w_{1,1}, w_{1,2}, w_{1,3}, x_1, x_2, x_3) = w_{1,1}x_1 + w_{1,2}x_2 + w_{1,3}x_3$$

$$\frac{\partial h_1}{\partial w_{1,1}} = x_1$$

$$\frac{\partial \mathcal{L}}{\partial w_{1,1}} = \frac{\partial \mathcal{L}}{\partial y_1}\frac{\partial y_1}{\partial h_1}\frac{\partial h_1}{\partial w_{1,1}} = \left(-\frac{e^{y_2}}{e^{y_1} + e^{y_2}}\right)(v_{1,1})(x_1)$$

Exercise 1

- output: $\langle p_1, p_2, p_3 \rangle$
- target: $\langle v_1, v_2, v_3 \rangle = \langle 0, 1, 0 \rangle$
- **Spherical Loss**:

$$\frac{1}{3} \sum_{i \leq 3} \left(1 - \frac{|1 - p_i - v_i|}{\sqrt{p_i^2 + (1 - p_i)^2}} \right)$$

- torch.mean($\langle x_1, x_2, x_3 \rangle$)=$\left\langle \frac{1}{3} \sum_{i \leq 3} x_i \right\rangle$
- torch.div($\langle x_1, x_2, x_3 \rangle$, $\langle y_1, y_2, y_3 \rangle$)=$\left\langle \frac{x_1}{y_1}, \frac{x_2}{y_2}, \frac{x_3}{y_3} \right\rangle$
- torch.abs($\langle x_1, x_2, x_3 \rangle$)=$\langle |x_1|, |x_2|, |x_3| \rangle$
- torch.sqrt($\langle x_1, x_2, x_3 \rangle$)=$\left\langle \sqrt{x_1}, \sqrt{x_2}, \sqrt{x_3} \right\rangle$
- torch.square($\langle x_1, x_2, x_3 \rangle$)=$\left\langle x_1^2, x_2^2, x_3^2 \right\rangle$

- torch.square(output) +torch.square(1-output)
- **Add the spherical loss function to `Spherical_with_exercises.ipynb`**

Neural Net Classification with IP Scoring Rules

Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

2-Parameter IP Scoring Rule

$$\mathcal{I}_i(\mathcal{D}) = \mathcal{F}_i(\mathcal{D}) + \mathcal{S}_i(\mathcal{D}) = \int_{(\mathcal{D}\setminus\mathcal{D}_i)\cup(\mathcal{D}_i\setminus\mathcal{D})} |\phi_i(x_1, x_2, x_3)|\, d\mu$$

$$\text{where } \phi_i(x_1, x_2, x_3) = \begin{cases} \lambda x_i & \text{if } x_i < 0 \\ \gamma x_i & \text{if } x_i \geq 0 \end{cases} \quad \text{for some } \lambda \geq \gamma > 0$$

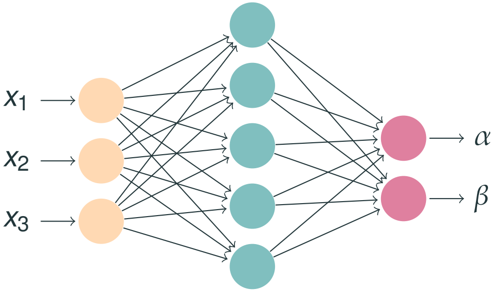
Admissible $\mathcal{D}_f$ relative to our 2-parameter IP scoring rule

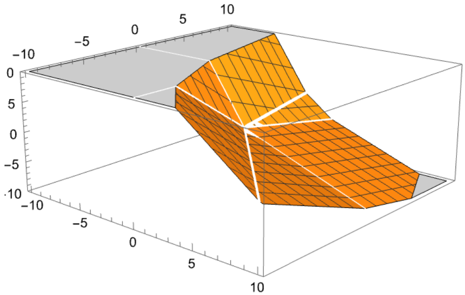
$$\mathcal{D} = \{\langle x_1, x_2, x_3\rangle \mid x_3 \geq f(x_1, x_2)\} \subseteq \mathbb{R}^3$$

where

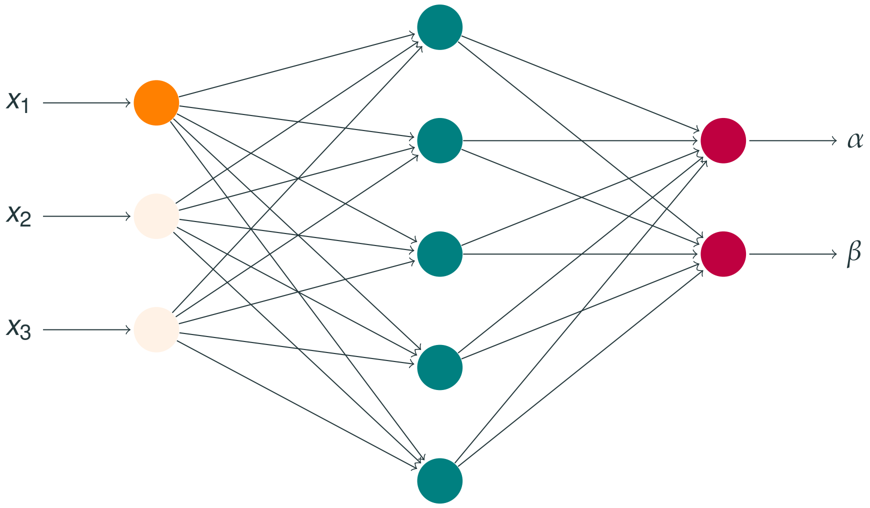
$$f(x_1, x_2) = \begin{cases} \frac{-\gamma(\alpha x_1 + \beta x_2)}{\lambda} & \text{if } x_1 \geq 0, x_2 \geq 0 \\ \frac{-\lambda(\alpha x_1 + \beta x_2)}{\gamma} & \text{if } x_1 < 0, x_2 < 0 \\ \frac{-(\alpha\lambda x_1 + \beta\gamma x_2)}{\gamma} & \text{if } x_1 < 0, x_2 \geq 0, \alpha\lambda x_1 + \beta\gamma x_2 < 0 \\ \frac{-(\alpha\lambda x_1 + \beta\gamma x_2)}{\lambda} & \text{if } x_1 < 0, x_2 \geq 0, \alpha\lambda x_1 + \beta\gamma x_2 \geq 0 \\ \frac{-(\alpha\gamma x_1 + \beta\lambda x_2)}{\gamma} & \text{if } x_1 \geq 0, x_2 < 0, \alpha\gamma x_1 + \beta\lambda x_2 < 0 \\ \frac{-(\alpha\gamma x_1 + \beta\lambda x_2)}{\lambda} & \text{otherwise} \end{cases}$$

for some $\alpha, \beta > 0$





$$\textbf{Loss} = \mathcal{L}(\alpha, \beta) = \mathcal{I}_i(\mathcal{D}_f)$$

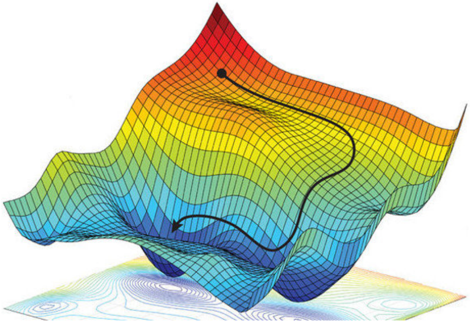

x_1
x_2
x_3
α
β

Training: Nudge weights in opposite direction of gradient of $\mathcal{L}$

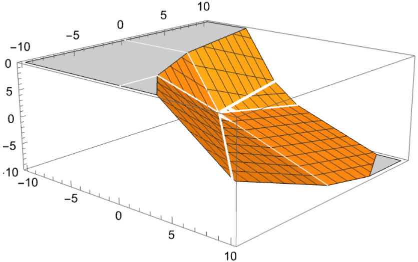
$$\nabla\mathcal{L} = \left\langle \frac{\partial\mathcal{L}}{\partial w_{1,1}}, \ldots, \frac{\partial\mathcal{L}}{\partial v_{2,5}} \right\rangle$$

Calculating $\frac{\partial\mathcal{L}}{\partial w_{1,1}}$ will involve, *e.g.*,

$$\frac{\partial\mathcal{L}}{\partial\alpha}\frac{\partial\alpha}{\partial h_1}\frac{\partial h_1}{\partial w_{1,1}}$$



Where's the classification?



Classification is a decision problem:

- Must specify gambles associated with classifying an image as: T-shirt/top, Trouser, or Pullover
  - T-shirt/top: $X = \langle x_1, x_2, x_3 \rangle$
  - Trouser: $Y = \langle y_1, y_2, y_3 \rangle$
  - Pullover: $Z = \langle z_1, z_2, z_3 \rangle$

- Choice set: $A = \{X, Y, Z\}$
- Reject U from A iff there is some V in A such that

$$V - U \in \mathcal{D}$$