REDES NEURONALES 2024

Clase 19 parte 1

Lunes 28 de octubre 2024

FAMAF, Universidad Nacional de Córdoba

Instituto de Física Enrique Gaviola (UNC-CONICET)

J02/11/23 c20p2

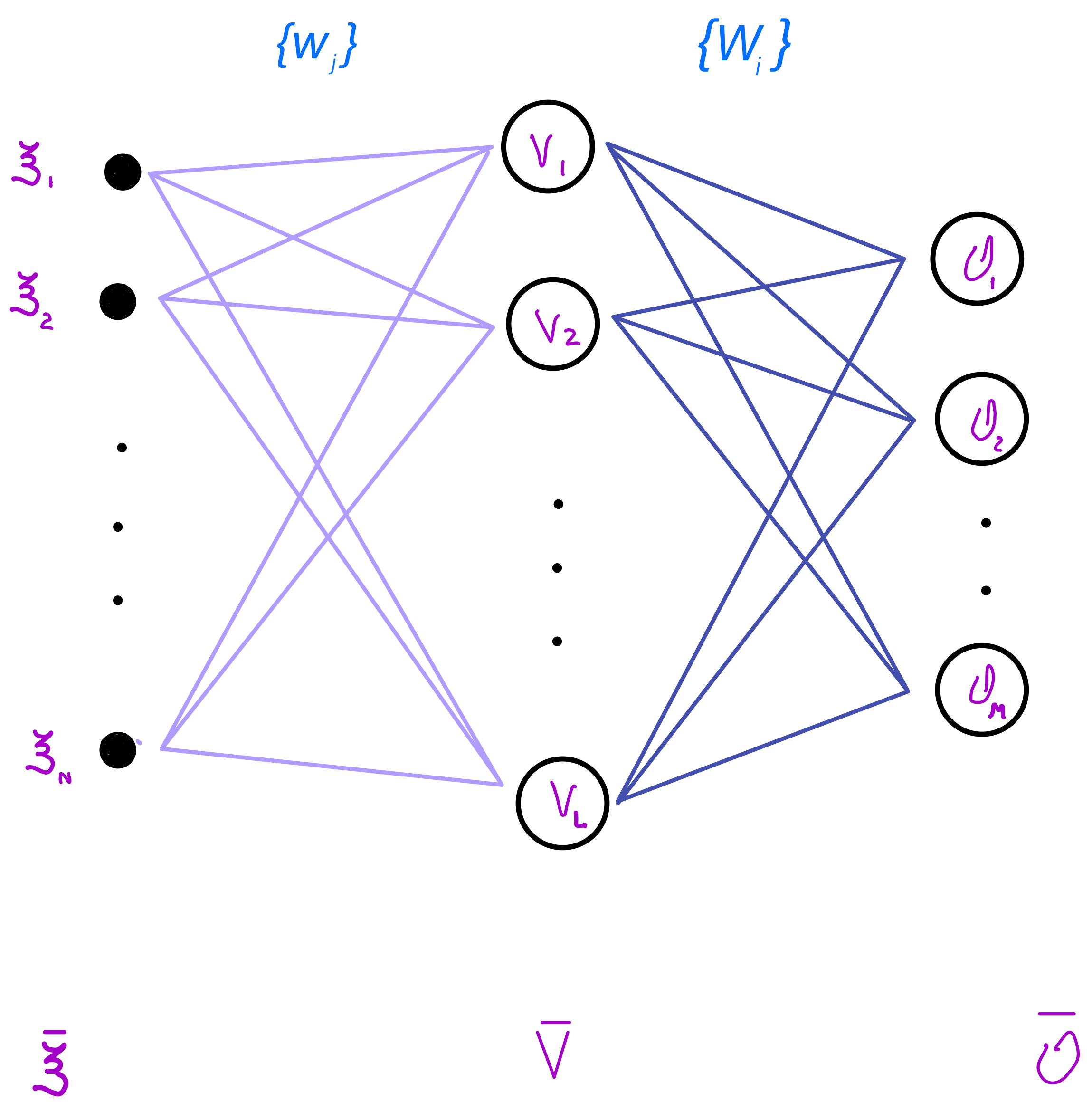
REPASO

BACK-PROPAGATION EN RED FEED-FORWARD CON UNA CAPA OCULTA

Buscamos hacer inferencias sobre cierta función desconocida *f*

$$f : \mathbb{R}^N \longrightarrow \mathbb{R}^m$$

dado un conjunto de datos etiquetados $\mathcal{C} = \{(\bar{\xi}^\mu, \bar{\zeta}^\mu)\} \quad \mu = 1, 2, \dots, P$



A partir de la expresión del ECM en función de todas las componentes de los vectores $\overline{W}_i$ y $\overline{w}_j$, vamos a aplicar el descenso por el gradiente de una forma muy específica.

$$E[\{\overline{w}\}] = \frac{1}{2} \sum_{\mu=1}^{P} \sum_{i=1}^{M} \left[\zeta_i^\mu - O_i^\mu \right]^2$$

$$= \frac{1}{2} \sum_{\mu=1}^{P} \sum_{i=1}^{M} \left[\zeta_i^\mu - g_2\left(\sum_{j=1}^{L} W_{ij}\, g_1\left(\sum_{k=1}^{N} w_{jk}\, \xi_k^\mu \right)\right)\right]^2$$

Primero corregimos los pesos sinápticos $\overline{W}_i$ (mayúscula) entre la capa oculta y la capa de salida sin importarnos por los $\overline{w}_j$ (minúscula).

$$W_{ij}^{\text{nuevo}} = W_{ij}^{\text{anterior}} + \Delta W_{ij}$$

donde:

$$\Delta W_{ij} = -\eta \frac{\partial E}{\partial W_{ij}}$$

$$= \eta \sum_{\mu=1}^{P} \cancel{\frac{2}{2}} \left[\zeta_i^\mu - O_i^\mu \right] g_2'(h_i^\mu)\, V_j^\mu$$

$$= \eta \sum_{\mu=1}^{P} \delta_i^\mu\, V_j^\mu$$

con

$$\delta_i^\mu = g_2'(h_i^\mu) \left[\zeta_i^\mu - O_i^\mu \right]$$

Una vez que hemos actualizado todos los parámetros sinápticos de la última capa, pasamos a hacer lo mismo con los parámetros que van de la entrada a la capa oculta, o sea, los $\overline{w}_j$ minúsculos. Para eso calculamos estas componentes del gradiente:

$$w_{jk}^{\text{nuevo}} = w_{jk}^{\text{viejo}} + \Delta w_{jk}$$

donde

$$\Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}}$$

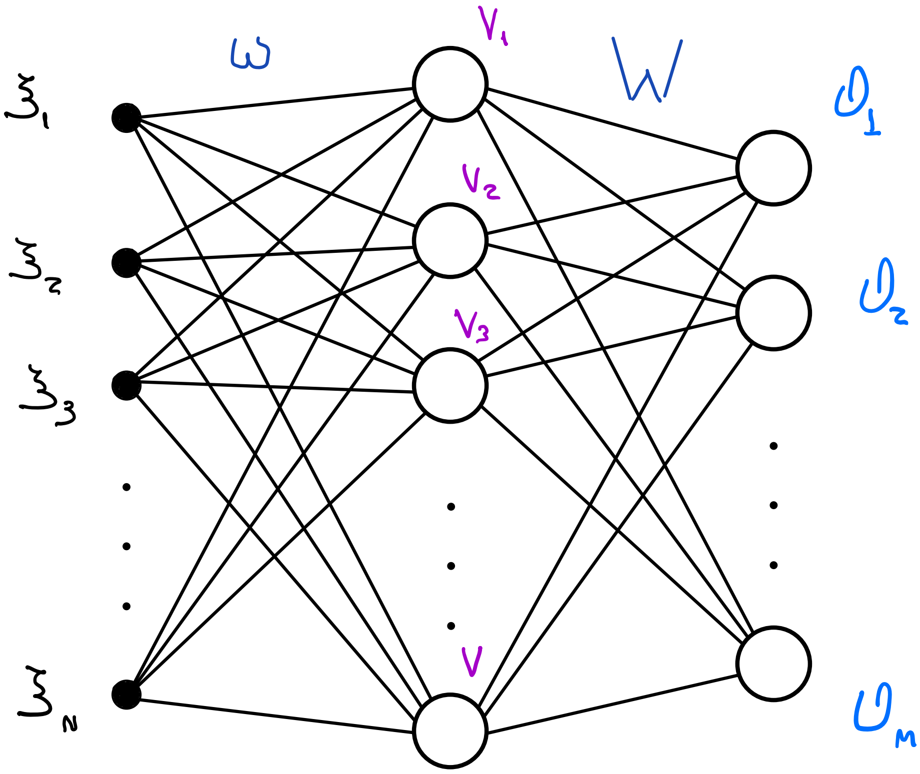
$$= \eta \sum_{\mu=1}^{p} \left[\zeta_i^\mu - O_i^\mu\right] g_2'(h_i^\mu)\, W_{ij}\, g_1'(h_j^\mu)\, \xi_k^\mu$$

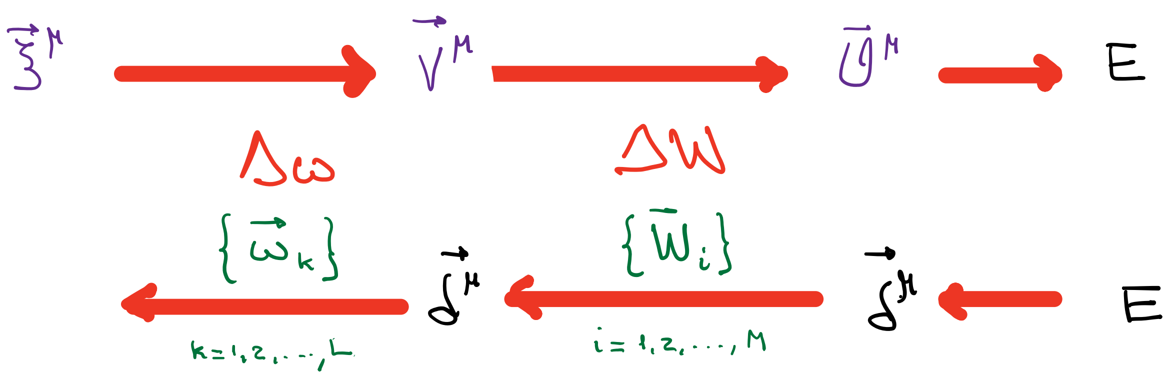
$$= \eta \sum_{\mu=1}^{p} \delta_i^\mu\, W_{ij}\, g_1'(h_j^\mu)\, \xi_k^\mu$$

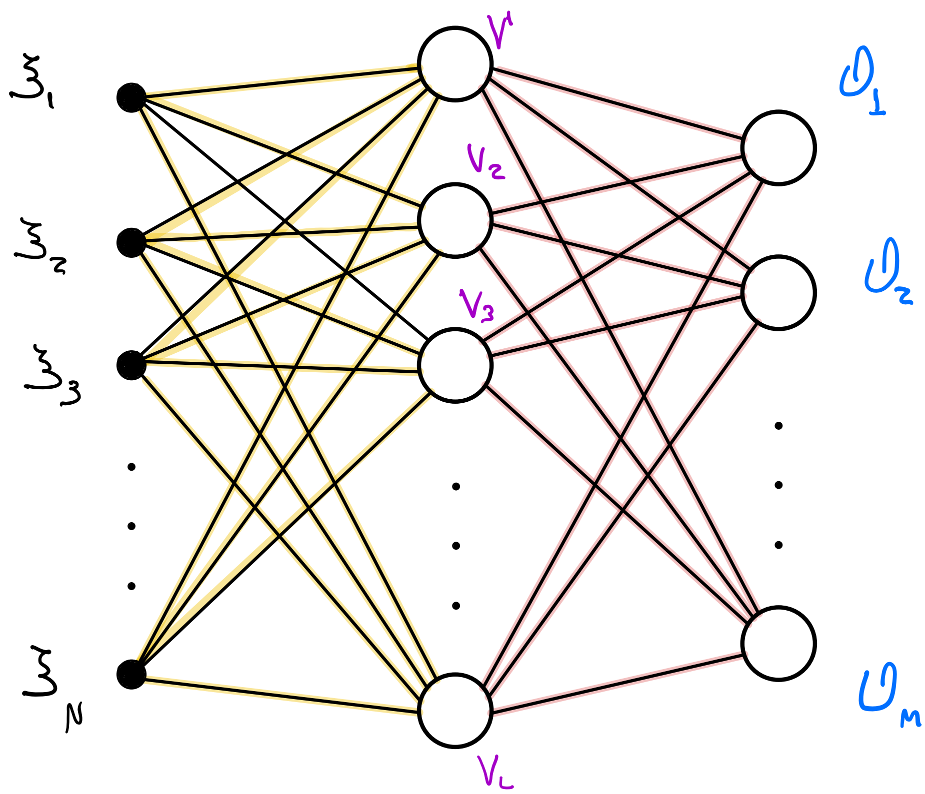
$$= \eta \sum_{\mu=1}^{p} \delta_j^\mu\, \xi_k^\mu$$

con

$$\delta_j^\mu = g_1'(h_j^\mu) \sum_i W_{ij}\, \delta_i^\mu$$


ξ1
ξ2
ξ3
ξN
ω
V1
V2
V3
V
W
O1
O2
OM


ξ^μ
V^μ
O^μ
E
Δω
ΔW
{ω_k}
δ^μ
{W_i}
δ^μ
E
k=1,2,...,L
i=1,2,...,M


ξ1
ξ2
ξ3
ξN
V1
V2
V3
VL
O1
O2
OM

¿Cuántos parámetros tenemos?

$$(N \times L) + (L \times M) = L \times (N + M)$$

Este método se llama *back-propagation* o retro-propagación. Noten que le mostramos el elemento μ del conjunto de entrenamiento a la red, la información viaja hacia adelante, en dirección a la salida. Con los resultados de las M salidas calculamos el error. Con el error primero calculamos las correcciones a los pesos sinápticos $\overline{\boldsymbol{W}}_i$ entre la capa oculta y la capa de salida. Una vez actualizados estos parámetros sinápticos pasamos a corregir los pesos sinápticos $\overline{w}_j$ entre la entrada y la capa oculta.

REPASO DE BACK PROPAGATION

Pseudo-código

- Presentamos $\bar{\xi}^{\mu}$, calculamos primero $\bar{V}^{\mu}$ y luego $\bar{O}^{\mu}$.
- Con $\bar{O}^{\mu}$ y $\bar{\xi}^{\mu}$ calculamos $E(\{\bar{\mathbf{W}}\})$.
- Con $E(\{\bar{\mathbf{W}}\})$ calculamos los valores de las correcciones δ_i^{μ} $(i=1,2,\ldots,M)$ y con ellos $E(\{\bar{\mathbf{W}}\})$ los pesos sinápticos $\bar{\mathbf{W}}$ que van de la capa oculta a la capa de salida

$$W_{ij}^{nuevo} = W_{ij}^{viejo} + \Delta W_{ij}$$

- Con los δ_i^{μ} $(i=1,2,\ldots,N)$ calculamos los δ_j^{μ} $(i=1,2,\ldots,M$ y $j=1,2,\ldots,L)$ y con ellos actualizamos los pesos sinápticos $\bar{\mathbf{w}}_j$, que van de la entrada a la capa oculta.

$$w_{ij}^{nuevo} = w_{ij}^{viejo} + \Delta w_{ij}$$

El algoritmo **ÉPOCA**

A. Sea $\mu = 1$

B. Mientras $(\mu \leq P)$ repetimos

1. Con $\bar{\xi}^{\mu}$ calculamos $\bar{\boldsymbol{V}}^{\mu}$

2. Con $\bar{\boldsymbol{V}}^{\mu}$ calculamos $\bar{\boldsymbol{O}}^{\mu}$

3. Con $\bar{\boldsymbol{O}}^{\mu}$ y $\bar{\xi}^{\mu}$ calculamos E

4. Con E calculamos $\bar{\delta}_i^{\mu}$

5. Con $\bar{\delta}_i^{\mu}$ calculamos los $\Delta\, \bar{\boldsymbol{W}}$ y los acumulamos

$$\Delta \overline{W} = \Delta \overline{W} + \Delta \overline{W}^{\mu}$$

6. Con $\bar{\delta}_i^{\mu}$ calculamos los $\bar{\delta}_j$

7. Con $\bar{\delta}_j$ actualizamos los $\Delta\, \bar{\boldsymbol{w}}$ y los acumulamos

$$\Delta \overline{w} = \Delta \overline{w} + \Delta \overline{w}^{\mu}$$

8. Pasamos al próximo ejemplo $\mu = \mu + 1$

9. Si $\mu = P$ actualizamos los $\bar{\boldsymbol{W}}_i$ y $\bar{\boldsymbol{w}}_j$ y volvemos a **B**

Leemos los datos:

- Conjunto de entrenamiento $(\overline{\xi}^{\mu}, \overline{\zeta}^{\mu})$
- Tolerancia *tol*
- Razón de aprendizaje η

Sea $t=1$

Llamamos a la rutina ***ÉPOCA*** que devuelve E_{trrain}, $\overline{\mathbf{W}}$ y $\overline{\mathbf{w}}$

Con $\overline{\mathbf{W}}$ y $\overline{\mathbf{w}}$ calculamos E_{test}

Si $E_{test} < tol$ paramos si no hacemos $t = t+1$

Hasta acá supusimos que actualizamos los acoplamientos sinápticos $\overline{\boldsymbol{W}}$ y $\overline{\boldsymbol{w}}$ después de mostrarle a la red todos los ejemplos del conjunto de entrenamiento, o sea después de una época. Esta forma de actualizar se conoce como *actualización en lote (batch)*.

$$\Delta W_{ij} = \underbrace{\Delta W_{ij}^{(1)} + \Delta W_{ij}^{(2)} + \cdots + \Delta W_{ij}^{(p)}}_{\mu}$$

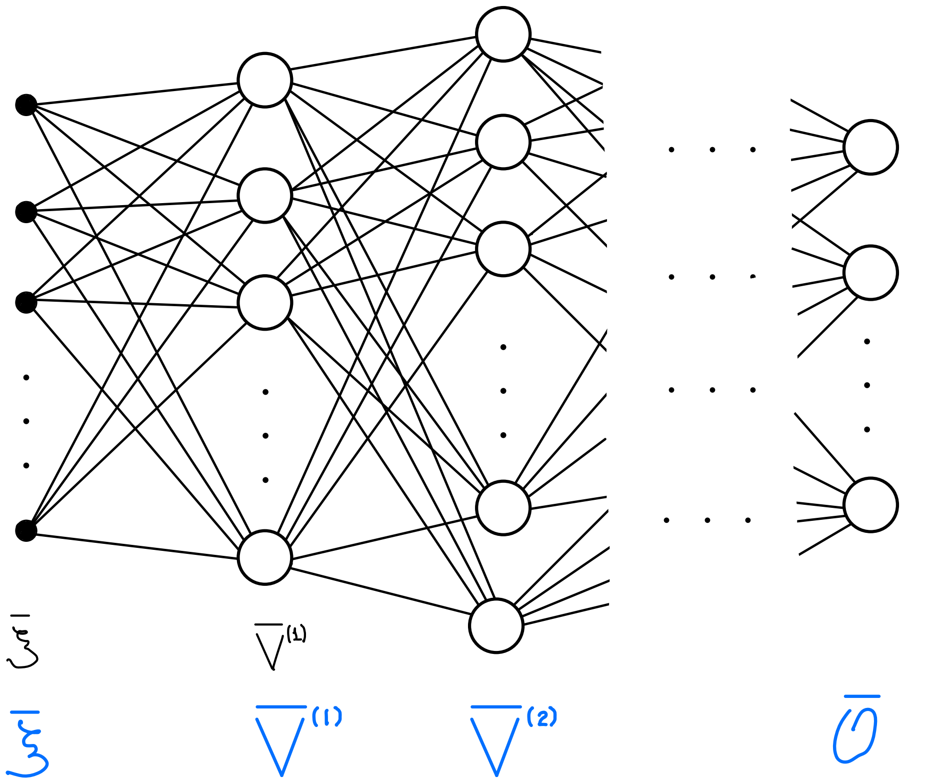
$$\Delta w_{jk} = \underbrace{\Delta w_{jk}^{(1)} + \Delta w_{jk}^{(2)} + \cdots + \Delta w_{jk}^{(p)}}_{\mu}$$

Otra posibilidad que no hemos analizado es actualizar todos los pesos sinápticos $\overline{\boldsymbol{W}}_i$ y $\overline{\boldsymbol{w}}_j$ después de presentarle cada ejemplo del conjunto de entrenamiento a la red. Este método se denomina *actualización en línea (on line)*. Es una actualización más fina y precisa pero requiere mucho más cálculo numérico.

Veremos pronto que la forma más adecuada es una actualización en *mini-lotes (mini-batch)*, o sea, algo intermedio entre los métodos en lote y en línea.

EL MÉTODO BACK-PROPAGATION PARA UN NÚMERO ARBITRARIO DE CAPAS OCULTAS

Ahora podemos generalizar el método muy fácilmente para un número arbitrario de capas ocultas

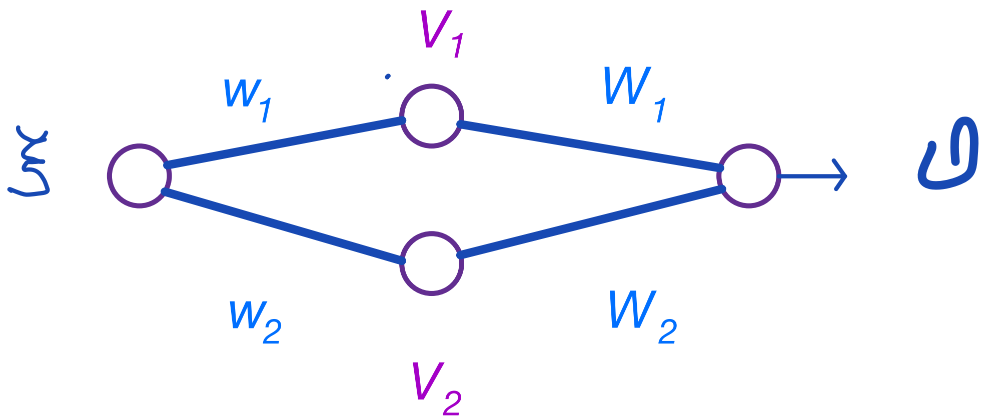


Si queremos actualizar los acoplamientos $\overline{w_{pq}}$ entre la capa q (anterior) y p (posterior)

$$\Delta \overline{w_{pq}}^{(\alpha)} = \eta \sum_{\mu=1}^{P} \delta_{p}^{\mu} \overline{V}_{q}^{\mu(\alpha)}$$

$$O_i = g\left(\sum w^{(4)} g\left(\sum w^{(3)} g\left(\sum w^{(2)} g\left(\sum w^{(1)} \xi\right)\right)\right)\right)$$

Caso1: cuatro neuronas en dos capas (una oculta)



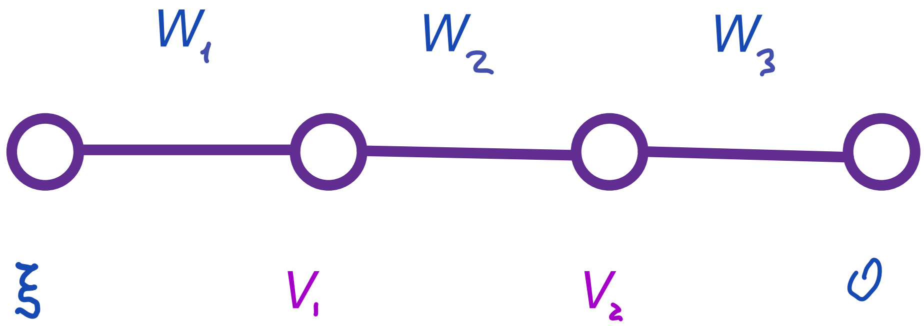
$$V_1 = g(w_1 \xi)$$

$$V_2 = g(w_2 \xi)$$

$$O = g_2(W_1 V_1 + W_2 V_2)$$

$$= g_2(W_1\, g_1(w_1 \xi) + W_2\, g_1(w_2 \xi))$$

Caso 2: cuatro neuronas en tres capas (dos ocultas)



$$O = g(w_3\, V_2) = g(w_3\, g(w_2\, V_1))$$

$$= g(w_3\, g(w_2\, g(w_1 \xi)))$$