Inicialización de Pesos y Funciones de Activación

Estabilidad, flujo de gradientes y neuronas muertas

Cesar Garcia

2025

Introducción

- Entender por qué **no cualquier inicialización funciona**
- Explicar **gradientes que desaparecen o explotan**
- Comparar funciones de activación comunes
- Introducir inicializaciones Xavier y He
- Relacionar activación, inicialización y estabilidad

¿Por qué dos redes con la misma arquitectura pueden comportarse de forma radicalmente distinta?

Recordatorio: forward y backward

Propagación doble

En una red profunda:

- el **forward** propaga activaciones
- el **backward** propaga gradientes

Ambos dependen de:

- pesos
- activaciones

¿Qué ocurre si las activaciones crecen o se atenúan capa tras capa?

El problema de los gradientes

Vanishing y exploding gradients

En redes profundas:

- gradientes pueden **hacerse muy pequeños**
- o **crecer sin control**

Consecuencias:

- aprendizaje muy lento
- inestabilidad numérica

¿Por qué este problema empeora al aumentar la profundidad?

Intuición matemática

Producto de derivadas

El gradiente total es un producto de derivadas locales:

$$\frac{\partial L}{\partial W} = \prod_k \frac{\partial z_{k+1}}{\partial z_k}$$

Si cada factor es:

- $< 1 \rightarrow$ desaparece
- $1 \rightarrow$ *explota*

¿Qué papel juegan aquí las funciones de activación?

Funciones de activación

Por qué no basta con capas lineales

Sin activaciones:

- la red colapsa a una sola transformación lineal

Activaciones introducen:

- no linealidad
- capacidad expresiva

¿Qué perderíamos si quitamos todas las activaciones?

Sigmoid y Tanh

Activaciones clásicas

Propiedades:

- salida acotada
- derivadas pequeñas en saturación

Problemas:

- gradientes que desaparecen
- entrenamiento lento

¿Por qué estas funciones fueron problemáticas en redes profundas?

ReLU

Activación moderna

ReLU:

$$f(x) = \max(0, x)$$

Ventajas:

- gradiente constante en región activa
- entrenamiento más estable

Problemas:

- neuronas muertas

¿Qué significa que una neurona esté "muerta"?

Variantes de ReLU

Mitigar neuronas muertas

Ejemplos:

- Leaky ReLU
- ELU (conceptual)

Idea:

- permitir gradiente negativo pequeño

¿Qué compromiso introduce permitir valores negativos?

Inicialización de pesos

El punto de partida importa

Pesos mal inicializados pueden:

- amplificar señales
- atenuarlas demasiado

La inicialización busca:

- mantener varianza estable

¿Por qué no inicializar todos los pesos en cero?

Xavier (Glorot)

Equilibrio forward/backward

Xavier busca:

- preservar varianza de activaciones
- preservar varianza de gradientes

Funciona bien con:

- Tanh
- Sigmoid

¿Qué asume Xavier sobre la activación?

He initialization

Diseñada para ReLU

He initialization:

- ajusta la varianza para ReLU
- compensa activaciones que anulan valores negativos

Resultado:

- flujo de gradientes más estable

¿Por qué ReLU necesita una inicialización distinta?

Activación + inicialización

Paquetes coherentes

Buenas combinaciones:

- ReLU + He
- Tanh + Xavier

Malas combinaciones:

- ReLU + Xavier (profundo)
- Sigmoid + inicialización arbitraria

¿Qué problema esperas ver con una mala combinación?

Señales prácticas

Diagnóstico temprano

Síntomas de problemas:

- pérdida no cambia
- gradientes ~ 0
- activaciones saturadas

Mirar:

- curvas
- estadísticas internas

¿Qué mirarías primero si una red no aprende?

Puente al notebook

Qué vamos a observar

En el notebook:

- compararemos inicializaciones
- compararemos activaciones
- mediremos gradientes y activaciones

Todo con el **mismo modelo y dataset**.

¿Qué esperas que ocurra al cambiar solo la inicialización?

Idea clave de la sesión

Estabilidad antes que profundidad

Antes de redes profundas:

- asegúrate de que el gradiente fluye

Activación e inicialización:

definen si aprender es posible.

¿Qué combinación te parece más robusta ahora?