Clase 111 — Gradient clipping

Parte: 2 — Deep Learning · Fuente: Géron, cap. 11 § Gradient Clipping + Pascanu, Mikolov & Bengio (2013). Duración estimada: 45 min.

Clase 111 — Gradient clipping

> *Parte: 2 — Deep Learning · Fuente: Géron, cap. 11 § Gradient Clipping + Pascanu, Mikolov & Bengio (2013). Duración estimada: 45 min.*

Objetivo

Aplicar gradient clipping —limitar la norma o el valor de los gradientes antes de actualizar pesos— como protección contra exploding gradients, especialmente crítico en RNN/LSTM (clase 120) y en entrenamiento de LLMs. Diferenciar clipnorm (preserva dirección) de clipvalue (clipea por elemento).

Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Configurar clipping en cualquier optimizer Keras: Adam(clipnorm=1.0) o Adam(clipvalue=0.5).
- Saber cuándo clipnorm es preferible (default moderno): preserva dirección del gradiente.
- Implementar clipping manual en custom training loop con tf.clip_by_global_norm.
- Detectar exploding monitoreando la norma del gradiente.
- Reconocer que en Transformers de LLM, clipnorm=1.0 es estándar.

Temas

- Exploding revisitado: ¿qué pasa cuando ||grad|| crece exponencialmente?
- clipnorm: si ||g|| > c, escalar g ← g · c/||g||. Preserva dirección.
- clipvalue: g_i ← clip(g_i, -c, +c) por elemento. Cambia dirección.
- Global norm vs per-variable: clip_by_global_norm mira el norm del tensor concatenado de todos los pesos.

Definiciones y características

- Gradient clipping: limitar el tamaño del gradiente antes de aplicarlo.
- clipnorm: norma euclídea L2 máxima permitida. Si excede, se reescala manteniendo dirección.
- clipvalue: máximo valor absoluto por elemento.
- Global norm: ||g|| calculado sobre todos los gradientes concatenados como un solo vector.
- tf.clip_by_global_norm(grads, clip_norm=1.0): API moderna para clipping en custom loops.

Dataset / recursos

- Fashion-MNIST + un MLP propenso a exploding (LR alto + sin BN).
- Librerías: tensorflow, keras.

Ejercicios

1. Forzar exploding: entrenar MLP con Adam(lr=10.0) sobre Fashion-MNIST. loss = nan rápido.
2. Clipping al rescate: repetir con Adam(lr=10.0, clipnorm=1.0). Verificar que no explota (aunque sigue

malo el LR — clipping no es solución a LR mal calibrado, solo a explosión).

3. clipnorm vs clipvalue: comparar las dos con LR razonable. Para problemas estables son ~equivalentes; diferencias aparecen en patrones específicos.
4. Custom loop: implementar el paso con gradients = tape.gradient(loss, model.trainable_variables); gradients, _ = tf.clip_by_global_norm(gradients, 1.0); optimizer.apply_gradients(...).
5. Monitoreo: graficar ||grad|| por step. Verificar que no excede el clipnorm configurado.

Homework verificable

Reentrenar el modelo del ejercicio anterior con LR razonable + clipnorm=1.0 como práctica estándar:

1. MLP [300, 100] Fashion-MNIST.
2. Adam(learning_rate=1e-3, clipnorm=1.0).
3. Graficar la norma del gradiente en cada step (custom loop o callback).
4. Verificar que la curva está acotada a 1.0 cuando el modelo aún no convergió.

Criterio de aceptación: el modelo entrena estable (val_accuracy ≥ 0.87) y la norma del gradiente está acotada como esperado.

Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
clipnorm=0.01 y modelo no aprende	Clipping muy agresivo enmascara gradientes
Configuro clipvalue y los gradientes peque	Si el valor es chico y los gradientes son
Custom loop sin clipping → loss=nan ocasio	Sin clipping RNN explota. Fix: tf.clip_by_
Cliping pasa pero el problema es otro (LR	Clipping no arregla LR mal calibrado. Diag
clipnorm y clipvalue simultáneamente	Keras los aplica en cascada — comportamien

Preguntas frecuentes

¿clipnorm=1 o clipnorm=5?

1.0 para Transformers/LLMs (estándar). 5.0 para RNN/LSTM clásicos. Para MLPs/CNNs con BN, en general no hace falta clipping; un default de 1.0 no hace daño.

¿Clipping deteriora el modelo final?

Si el clipping nunca se activa (rara vez sobrepasás), no hace nada. Si se activa todo el tiempo, es una banda-aid sobre un problema más serio. Ideal: clipnorm por si las moscas pero no debería gatillarse seguido.

¿Cuándo monitorear la norma del gradiente?

Siempre que entrenes algo nuevo / no probado. Es el indicador más rápido de exploding/vanishing.

¿GradientTape global vs por variable?

Global norm (clip_by_global_norm) es lo correcto: trata el conjunto de pesos como un vector único. Per-variable distorsiona la dirección.

¿Por qué en LLMs es tan importante?

Transformers profundos + sequences largas → gradientes pueden ser muy heterogéneos. clipnorm=1.0 y

Adam(beta1=0.9, beta2=0.95) son la receta default de modelos como GPT-3 entrenados desde cero.

Referencias

- Géron, cap. 11 — Gradient Clipping.
- Pascanu, Mikolov & Bengio (2013), On the difficulty of training recurrent neural networks, ICML.
- tf.clip_by_global_norm.
- Keras Optimizer base — clipnorm/clipvalue/global_clipnorm.

Siguiente clase

Clase 112 — Transfer learning, unsupervised pretraining

Apéndice: notebook (primer bloque)

Primera celda ejecutable del notebook de la clase.

```
# Imports y configuración inicial
```

Archivos complementarios

- notebook.ipynb