Clase 162 — Policy gradients

Parte: 2 — Deep Learning · Fuente: Géron, cap. 18 § Policy Gradients + Sutton & Barto, cap. 13. Duración estimada: 70 min.

Clase 162 — Policy gradients

> *Parte: 2 — Deep Learning · Fuente: Géron, cap. 18 § Policy Gradients + Sutton & Barto, cap. 13. Duración estimada: 70 min.*

Objetivo

Implementar policy gradient —REINFORCE (Williams 1992)—: parametrizar la policy con una red neuronal π_θ(a|s), optimizar directamente la expected return via gradiente. Es el método más simple de RL que usa redes y la base conceptual de PPO/A2C/A3C (clase 138).

Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Definir la policy como red state → softmax(actions).
- Calcular el gradiente REINFORCE: θ J = E[θ log π_θ(a|s) · G_t].
- Implementar el training loop: rollout → calcular returns → gradient ascent.
- Aplicar baseline (substraer V(s) de G_t) para reducir varianza.
- Reconocer la limitación: alta varianza, lento (motiva A2C/PPO).

Temas

- Expected return J(θ) = E_π[G].
- Policy gradient theorem: θ J = E[θ log π · Q].
- REINFORCE algorithm: rollout completo + apply gradient.
- Baseline para reducir varianza.
- Discounted returns con γ.

Definiciones y características

- Policy network: π_θ(a|s), típicamente MLP con softmax (discrete actions) o gaussiana (continuas).
- Log-prob: log π(a|s), lo que multiplicamos por G_t para el gradient.
- Discounted return G_t: Σ γ^k r_{t+k}.
- Baseline: cualquier función de s (e.g., V(s)) que reduce varianza sin sesgo.
- Advantage A = G - V(s): cuán mejor o peor fue la acción comparada con el valor esperado.

Dataset / recursos

- CartPole-v1.
- Librerías: gymnasium, tensorflow, keras.

Ejercicios

1. Policy network: Dense(32) → Dense(32) → Dense(2, softmax) para CartPole.
2. Rollout: ejecutar 1 episodio, guardar (s, a, r) por timestep.

3. Returns: calcular G_t para cada timestep con γ=0.99.
4. Gradient step: loss = -Σ log π(a_t|s_t) · G_t; backward; apply.
5. Con baseline: agregar V(s) head, restar de G antes del gradient.

Homework verificable

REINFORCE en CartPole:

1. Policy Dense(64) → Dense(64) → Dense(2, softmax).
2. Train 500 episodios.
3. Reportar return medio por época.
4. Comparar con/sin baseline.

Criterio de aceptación: episode return llega a ≥ 195 (resuelto) en ≤ 300 episodios; baseline acelera convergencia.

Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
Alta varianza en gradientes	Inherente a REINFORCE. Fix: baseline, batc
Policy collapse a una acción	LR alto o sin entropy bonus. Fix: bajar LR
Recompensas muy chicas → updates débiles	Normalizar G (mean=0, std=1) por episode.
Action probs nan	Inputs sin normalizar + softmax. Fix: clip
El env tiene rewards muy variables → entre	Probar con env más simple primero.

Preguntas frecuentes

¿REINFORCE en producción?

Casi no — pero es la base. PPO es el default industrial moderno (clase 138).

¿On-policy o off-policy?

REINFORCE es on-policy (entrena con datos generados por la policy actual). Off-policy (DQN, SAC) reutiliza datos viejos.

¿Cómo elijo γ?

0.99 default. Más alto = más visión a largo plazo, pero más varianza. Para tareas episódicas cortas, 0.95-0.99.

¿Entropy regularization?

Agregar -β · H(π) a la loss promueve exploración (policy no determinística temprano). Estándar.

¿Cómo veo si converge?

Plot del return promedio por epoch (smoothed). Sube → converge.

Referencias

- Géron, cap. 18 — Policy Gradients.

- Williams (1992), Simple Statistical Gradient-Following Algorithms (REINFORCE).
- Sutton & Barto (2018), cap. 13.

Siguiente clase

Clase 163 — Markov Decision Processes

Apéndice: notebook (primer bloque)

Primera celda ejecutable del notebook de la clase.

```
# Imports y configuración inicial
```

Archivos complementarios

- notebook.ipynb