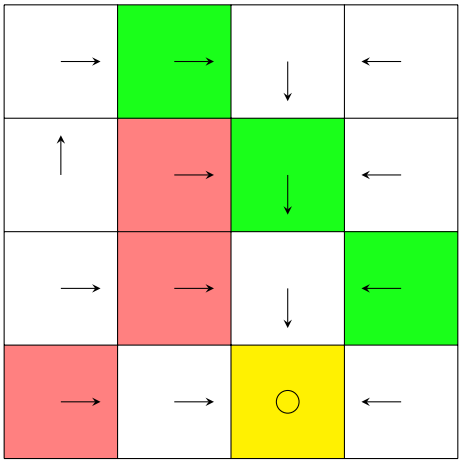
UAB
Universitat Autònoma
de Barcelona

Treball de Fi de Grau en Matemàtiques

Aprenentatge per Reforç



Autor: Biel Majó i Cornet

Tutor: Roberto Rubio i Nuñez

Juny de 2024

Índex

1 Introducció

L'aprenentatge per reforç és un tipus d'aprenentatge automàtic. Es diferencia d'altres tipus d'aprenentatge automàtic pel fet que utilitza recompenses per tal d'incentivar un agent a millorar. El concepte és molt semblant al d'entrenar un gos, on intentes modificar la seva conducta al relacionar l'acció desitjada amb una recompensa en forma de galeta. Això sí, en l'aprenentatge automàtic el que s'entrenen són distribucions de probabilitat en lloc de gossos i es donen puntuacions en comptes de galetes. Tot i que en aquest treball no aprofundirem ens els aspectes més pràctics, si no que ens centrarem en els més teòrics, és interessant saber quines aplicacions pot tenir.

L'aprenentatge per reforç és una eina molt útil a l'hora de resoldre problemes com crear programes que juguin a jocs complicats com els escacs, on les possibilitats són tant grans que és difícil definir quan una tirada ha estat ben executada. En aquest cas podríem dividir el problema en esdeveniments més senzills que haurem de valorar. Per exemple, perdre un peó tindria una recompensa negativa, però no tan negativa com perdre una torre. I fer escac i mat tindria una puntuació molt positiva, ja que ha de ser l'objectiu final de la partida d'escacs. Evidentment la utilitat de l'aprenentatge per reforç no es limita als jocs, qualsevol sistema difícil d'analitzar com actius financers o els fluxos de trànsit d'una zona poden ser millorats mitjançant l'aprenentatge per reforç.

L'objectiu d'aquest treball és estudiar des d'una perspectiva matemàtica els fonaments de l'aprenentatge per reforç. És una col·lecció de conceptes i idees extretes de diferents fonts, i en cap cas noves. La intenció és crear uns apunts que permetin entendre com funcionen alguns dels mètodes més significatius. Concretament, s'introdueixen mètodes de programació dinàmica, de Montecarlo i, per últim, mètodes basats en el gradient de funcions objectiu (gradient ascendent/descendent).

Així doncs, aquest treball només és una introducció a un món molt més extens i on podem trobar centenars de problemes més concrets però igualment complicats. Aquest és un tema d'interès en l'actualitat, ja que els mètodes d'aprenentatge automàtic estan començant a tenir efectes sobre el dia a dia de la societat i creix la desconfiança en aquests, principalment pel desconeixement. Conèixer el seu funcionament és una manera de saber les seves virtuts i les seves limitacions, i així, perdre la por que molta gent està adquirint a tot el que tingui a veure amb la intel·ligència artificial.

2 Conceptes Bàsics

Començarem introduint els conceptes bàsics que necessitarem per treballar en l'aprenentatge per reforç. Per tal de definir amb més facilitat els conceptes ens basarem tota l'estona en el mateix exemple.

2.1 Problema d'exemple

Imaginem un tauler en forma de graella. Cada casella pot ser de quatre tipus: camí, muntanya, arribada o neutre. El nostre objectiu serà trobar una ruta el màxim d'eficient des de qualsevol casella de sortida fins la casella d'arribada. Per millorar l'eficiència de la nostra ruta voldrem:

- Reduir la seva longitud el màxim possible.
- Evitar creuar muntanyes.
- Prioritzar passar per camins.

Vegem un exemple:

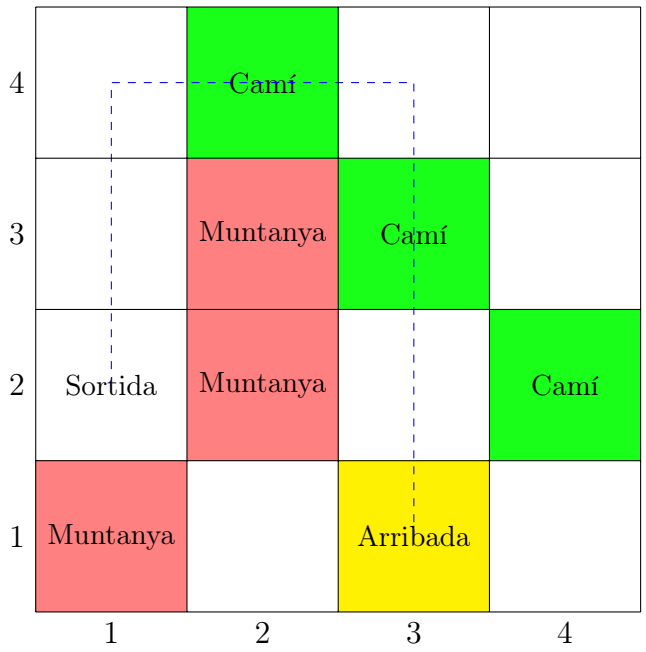


Figura 1: Exemple de jugada en el tauler.

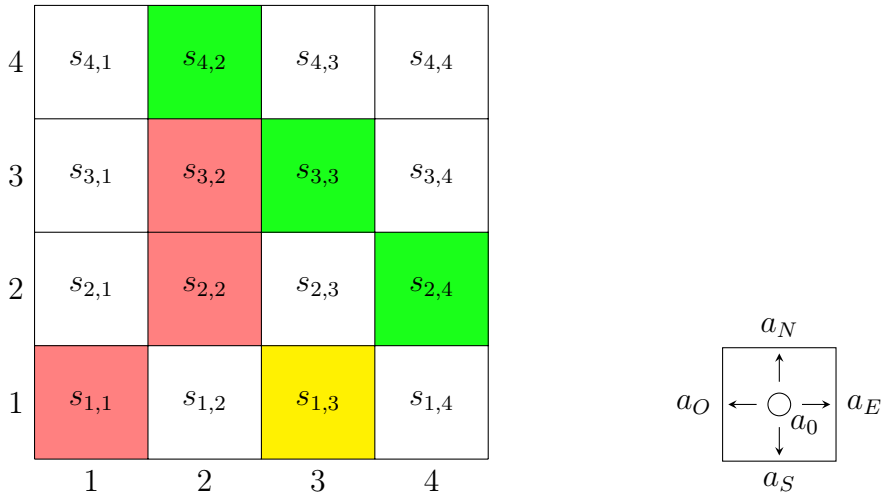
D'ara en endavant només farem servir els colors per indicar el tipus de casella. En aquest cas hem definit una ruta que ens porta d'una sortida concreta a l'arribada; però el que nosaltres voldríem és trobar una ruta per a qualsevol casella de sortida. A més de voler trobar una ruta, ens interessa saber si n'hi ha de millors, i en aquest cas, trobar-les.

2.2 Estats i accions

El primer concepte que definim és el d'estat. Serveix per descriure unes circumstàncies concretes del subjecte en relació a l'entorn. En el nostre exemple els estats són les caselles del tauler.

Ens referirem a l'espai d'estats com $\mathcal{S} = \{s_1, \cdots, s_n\}$, el conjunt de tots estats s_i possibles.

Per a cada estat considerem un conjunt d'accions possibles $\mathcal{A}_s$. Aquestes accions poden canviar l'estat del subjecte. En el nostre cas, en general, les accions possibles serien: anar al nord, al sud, a l'est o a l'oest i quedar-se quiet. Notem que si ens trobem a una vora no podrem executar totes les accions anteriors, per això definim un conjunt d'accions per a cada estat.



(a) Estats (b) Accions

Figura 2: Espais d'estats i accions del nostre exemple.

No necessàriament considerarem el resultat de les accions com a deterministes. Així doncs una acció ens dona una distribució de probabilitat sobre l'espai d'estats. Farem servir la notació $p(s'|s, a)$ per referir-nos a la probabilitat d'assolir l'estat s' executant l'acció a trobant-nos a l'estat s. Per exemple, podríem considerar que l'acció a_E de l'estat $s_{2,1}$; com que travessa una muntanya, no sempre ens serà possible avançar; aleshores $p(s_{2,2}|s_{2,1}, a_E) = 0.6$ i $p(s_{2,1}|s_{2,1}, a_E) = 0.4$.

2.3 Política

Una política descriu quines accions es prendran en funció de quin estat ens trobem. Per a cada parella estat-acció (s, a) tindrem que π és una distribució de probabilitat, on $\pi(a|s)$ és la probabilitat d'executar l'acció a si ens trobem a l'estat s. En el nostre model, suposant les accions deterministes ho indicarem amb fletxes per a fer-ho més intuïtiu, en altres casos, quan no sigui possible, simplement escriurem la distribució de probabilitat sobre les accions a cada estat:

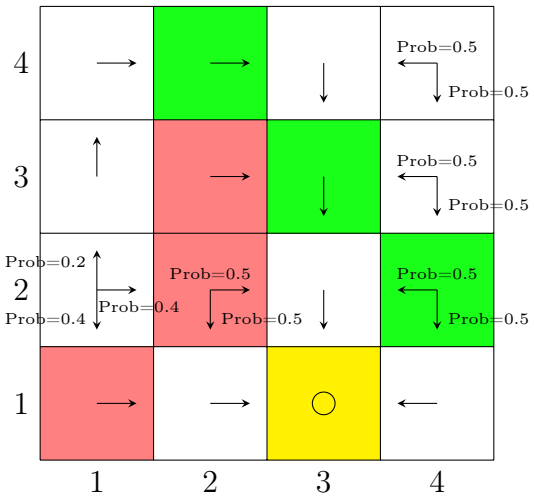


Figura 3: Representació d'una política si les accions són deterministes.

2.4 Recompensa

Per tal de valorar com de bona és una política utilitzarem les recompenses. Aquest és un dels conceptes més característics de l'aprenentatge per reforç. Una recompensa és un valor que assignem a una parella estat acció (pot ser a través d'una distribució de probabilitat). És l'eina que farem servir per valorar el comportament de l'agent, assignarem recompenses positives a les accions que considerem apropiades i recompenses negatives a les que vulguem desencoratjar. Podem definir un espai de recompenses $\mathcal{R}_{s,a}$ per cada parella estat-acció (s, a). Aquest espai de recompenses, format per nombre reals, pot definir una distribució per cada parella estat-acció.

En el nostre exemple podríem definir unes recompenses del tipus:

- El subjecte entra a una muntanya: $r_M = -3$.
- El subjecte entra a una casella neutre: $r_N = -1$.
- El subjecte entra a un camí: $r_C = 0$.
- El subjecte entra a la casella d'arribada: $r_F = 5$.

D'aquesta manera, estaríem dissuadint el subjecte a entrar a la muntanya i encoratjant-lo a anar pels camins. En aquest cas les recompenses serien deterministes, en general no és necessari; aleshores utilitzarem la notació $p(r|s, a)$ per parlar de la probabilitat d'obtenir una recompensa r executant l'acció a des de l'estat s.

Observem l'exemple anterior amb sortida a $2, 1$:

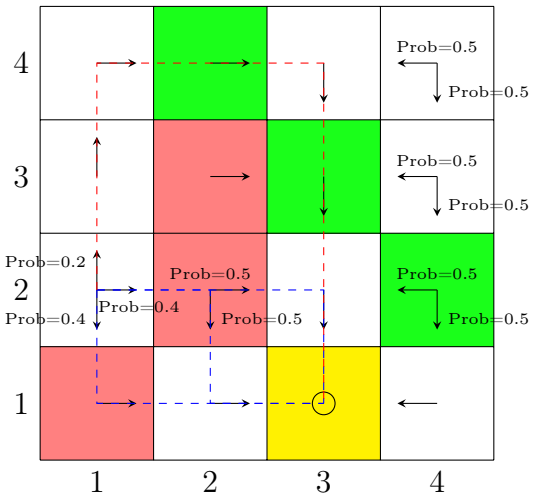


Figura 4: Rutes des de 2,1.

En aquest cas les trajectòries blaves tenen una recompensa immediata més petita que la vermella, però al final tenen més recompensa:

- Recompensa trajectòries blaves: -3-1+5=1.
- Recompensa trajectòria vermella: -1-1-1+0-1+0-1+5=0.

Tindríem un subjecte que només mira a curt termini.

Un altre problema que podem trobar, és que ens interessi que el subjecte faci el camí més curt possible. Per exemple, no ens interessa que es quedi a una casella de camí indefinidament, encara que no el penalitzi. Per evitar aquesta situació, definim uns altres conceptes.

2.4.1 Trajectòria i retorn

Una trajectòria és una cadena d'estats i accions amb les seves corresponents recompenses. A la suma de les recompenses d'una trajectòria l'anomenem retorn de la trajectòria. El retorn en diu com de bona és una política segons les trajectòries que se'n deriven.

2.4.2 Taxa de descompte

Definim la taxa de descompte com $\gamma \in (0, 1)$. A partir d'ara l'acció n-éssima enlloc d'una recompensa r passarà a tenir una recompensa $\gamma^n r$. Això ens soluciona dos problemes. El subjecte ara serà encoratjat a agafar les trajectòries més curtes quan en tingui una altra amb el mateix resultat però més llarga, però també ens obre la porta a definir trajectòries infinites sense preocupar-nos per la convergència del retorn.

Com a apunt d'interès, l'estructura d'aquest model és una cadena de Màrkov, sempre i quan l'espai d'estat sigui finit o numerable [1][2]. Així doncs, com que compleix la propietat

de Màrkov, (és a dir, que S_{t+1} només depén de S_t i no de S_{t-1}, S_{t-2},...), podem fer servir tots els resultat d'aquests tipus de processos estocàstics. En aquest treball no és massa rellevant, ja que ens basarem en coses més estructurals, sense entrar en el seu ús; però a l'hora d'aplicar els coneixement de forma més pràctica és molt útil aquest resultat.

3 Valor d'estat i equació de Bellman

Com que el nostre objectiu és trobar la millor política possible; per aconseguir-ho, primer hem de saber quantificar com de bona és una política. Aprofitant les recompenses, definim nous conceptes a partir dels retorns que ens ajudaran a veure quan una política és millor que una altra.

3.1 Valor d'estat

El primer que definim és el valor d'estat ($v_\pi(s)$). El valor d'estat és el retorn total esperat al sortir d'aquest. És a dir, l'esperança de la suma de les futures recompenses aplicant una política determinada, si ens trobem a un estat determinat. Veiem que depèn de la política, ja que afecta les trajectòries. Per definir-lo més rigorosament considerem $t = 0, 1, 2, ...$ un conjunt d'índexs per les accions i els estats. És a dir, en l'estat S_t, amb la política π, prendrem l'acció A_t i arribarem a l'estat S_{t+1}, el que resultarà en una recompensa R_{t+1}. Podem expressar una trajectòria a partir del moment t com:

$$S_t \xrightarrow{A_t} S_{t+1}, R_{t+1} \xrightarrow{A_{t+1}} S_{t+2}, R_{t+2} \xrightarrow{A_{t+2}} S_{t+3}, R_{t+3} \cdots$$

Així doncs, aplicant el descompte a cada pas; podem definir el retorn com a:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3}...$$

Fixem-nos que estem treballant amb variables aleatòries, sempre que la política no sigui determinista. Per tant, el valor d'estat $v_\pi(s)$ serà:

$$v_\pi(s) = \mathbb{E}[G_t | S_t = s].$$

3.2 Equació de Bellman

Ara ens podem preguntar com ho fem per a calcular fàcilment els valors d'estat. El primer que ens plantegem és fer servir la seva definició, on podrem notar que:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3}... = R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3}...) = R_{t+1} + \gamma G_{t+1}$$

així doncs, podem definir el valor d'estat a partir dels valors dels estats següents:

$$v_\pi(s) = \mathbb{E}[G_t | S_t = s] = \mathbb{E}[R_{t+1} + \gamma G_{t+1} | S_t = s] = \mathbb{E}[R_{t+1} | S_t = s] + \gamma \mathbb{E}[G_{t+1} | S_t = s]$$

El primer terme és l'esperança de la recompensa immediata des de l'estat actual, per tant, ho podem expressar com:

$$\mathbb{E}[R_{t+1}|S_t = s] = \sum_a \pi(a|s)\mathbb{E}[R_{t+1}|S_t = s, A_t = a] = \sum_a \left[\pi(a|s) \sum_r p(r|s,a)r\right]$$

on simplement multipliquem el valor esperat de la recompensa de cada acció per la seva probabilitat de ser escollida. La segona part la podem expressar com:

$$\begin{aligned}
\mathbb{E}[G_{t+1}|S_t = s] &= \sum_{s'} \mathbb{E}\left[G_{t+1}|S_t = s, S_{t+1} = s'\right] p(s'|s) \\
&= \sum_{s'} \mathbb{E}[G_{t+1}|S_{t+1} = s']p(s'|s) \\
&= \sum_{s'} v_\pi(s')p(s'|s) \\
&= \sum_{s'} v_\pi(s') \sum_a p(s'|s,a)\pi(a|s).
\end{aligned}$$

És a dir, en funció dels valors d'estat dels possibles estats següents, per la probabilitat d'arribar-hi. Així doncs, el valor d'estat de s amb la política π el podem expressar com:

Equació de Bellman:

$$\begin{aligned}
v_\pi(s) &= \sum_a \pi(a|s) \sum_r p(r|s,a)r + \gamma \sum_{s'} v_\pi(s') \sum_{s'} p(s'|s,a)\pi(a|s) \\
&= \sum_a \pi(a|s) \left[\sum_r p(r|s,a)r + \gamma \sum_{s'} v_\pi(s')p(s'|s,a)\right]
\end{aligned}$$

És important notar que el nom "equació"de Bellman pot ser confús. En realitat no tenim una sola equació de Bellman en un problema com el que estem plantejant, si no que tindrem una equació de Bellman per a cada estat. Així doncs, veurem que tenim un sistema lineal amb tantes equacions i incògnites com el cardinal del nostre conjunt d'estats. Concretament, tots els estats tenen una equació amb dependència dels estats als quals pots accedir des d'aquest estat.

Alternativament podrem trobar la següent forma equivalent per l'equació de Bellman [5]:

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma v(s')\right] \tag{1}$$

que farem servir més endavant.

3.3 Solucions de l'equació de Bellman

A trobar els valors d'estat d'una política n'hi diem avaluar aquesta política. És un pas important per treballar amb aprenentatge per reforç, ja que ens permet posar una quantitat a un concepte molt abstracte com és com de bona és aquesta política, com més alts els valors d'estat millor serà la política, donat que ens retorna recompenses més positives. Veure si una política és prou bona ho farem a 4.1. Com que cada valor d'estat depèn d'altres valors d'estat, podrem plantejar un sistema d'equacions amb tantes equacions com estats tingui l'espai d'estats $\mathcal{S}$. Totes juntes formaran un sistema que podrem expressar de forma vectorial, aquesta forma ens permetrà treballar de manera més fàcil amb el sistema d'equacions.

3.3.1 Forma vectorial de l'equació de Bellman

Per definir la forma vectorial de l'equació de Bellman reescriurem l'equació com:

$$v_\pi(s) = r_\pi(s) + \gamma \sum_{s_0} p_\pi(s_0|s) v_\pi(s_0)$$

on

$$r_\pi(s) = \sum_a \pi(a|s) \sum_r p(r|s,a) r, \quad p_\pi(s_0|s) = \sum_a \pi(a|s) \sum_{s_0} p(s_0|s,a).$$

Aquí, $r_\pi(s)$ és la mitjana de les recompenses immediates que es poden obtenir a partir de s sota la política π, i $p_\pi(s_0|s)$ és la probabilitat de passar de s a s_0 sota la política π.

Per poder expressar-ho en forma vectorial, indexem els estats com s_i $(i = 1, \ldots, n)$. Per a l'estat s_i, l'equació de Bellman és:

$$v_\pi(s_i) = r_\pi(s_i) + \gamma \sum_j p_\pi(s_j|s_i) v_\pi(s_j).$$

Definim els vectors columna i matrius següents:

$$\begin{aligned} v_\pi &= [v_\pi(s_1), \ldots, v_\pi(s_n)]^T \in \mathbb{R}^n \\ r_\pi &= [r_\pi(s_1), \ldots, r_\pi(s_n)]^T \in \mathbb{R}^n \\ P_\pi &\in \mathbb{R}^{n \times n} \text{ on } [P_\pi]_{ij} = p_\pi(s_j|s_i). \end{aligned}$$

És a dir, vector dels valors d'estat, un altre vector per la recompensa esperada des d'aquest estat i una matriu on cada elemennet $p_{i,j}$ és la probabilitat de passar de s_i a s_j en un pas.

Amb aquests objectes, obtenim la forma matriu-vector

$$v_\pi = r_\pi + \gamma P_\pi v_\pi$$

on v_π és el vector incògnita a resoldre.

3.3.2 Solució tancada de l'equació

Resolent $v_\pi = r_\pi + \gamma P_\pi v_\pi$ per v_π obtindrem:

$$v_\pi = (I - \gamma P_\pi)^{-1} r_\pi$$

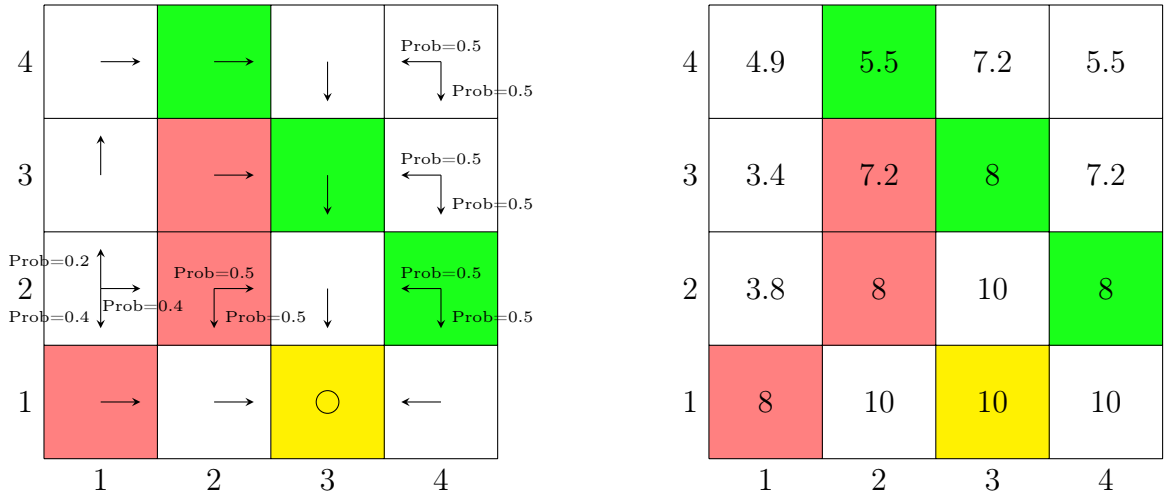
Aquesta solució pot ser molt complicada de computar ja que precisa d'una inversa de matriu. Tot i això és útil per un anàlisis teòric del problema. Pel que fa a les aplicacions per l'aprenentatge per reforç, es pot aproximar la solució amb mètodes numèrics, com veurem més endavant.

3.3.3 Exemple de càlcul dels valor d'estat

Agafant les recompenses com:

- El subjecte entra a una muntanya: $r_M = -3$.
- El subjecte entra a una casella neutre: $r_N = -1$.
- El subjecte entra a un camí: $r_C = 0$.
- El subjecte entra a la casella d'arribada: $r_F = 5$.

Podem calcular els valors d'estat de l'exemple, el resultat dels quals és:



(a) Política. (b) Valors d'estat.

Figura 5: Exemple de valors d'estat

3.4 Valors d'acció

En molts casos volem saber el valor de prendre una acció enlloc del de trobar-se a un estat, ja que la política es defineix per les accions i no pels estats. Així doncs, els valors d'estat són

molt útils per saber si la política és prou bona, però per millorar-la farem servir més els valors d'acció. Definirem el valor d'acció com:

$$q_\pi(s,a) = \mathbb{E}(G_t|S_t = s, A_t = a).$$

Aquí el valor de l'acció a des de l'estat s amb la política π. És important notar que el valor d'acció depèn tant de l'acció com de l'estat on ens trobem.
La suma ponderada dels valors d'acció des d'un estat és el valor d'estat:

$$v_\pi(s) = \mathbb{E}(G_t|S_t = s) = \sum_a \mathbb{E}(G_t|S_t = s, A_t = a)\pi(a|s) = \sum_a q_\pi(s,a)\pi(a|s),$$

així doncs tenim una relació entre els valors d'estat i d'acció. Això ens porta a veure que com hem vist anteriorment $v_\pi(s) = \sum_a \pi(a|s)\left[\sum_r p(r|s,a)r + \gamma \sum_{s'} v_\pi(s')p(s'|s,a)\right]$ i per tant,

$$q_\pi(s,a) = \sum_r p(r|s,a)r + \gamma \sum_{s'} v_\pi(s')p(s'|s,a)$$

Podem veure que per calcular el valor d'acció ens serà útil calcular els valor d'estat de certs estats, això ens indica què són conceptes molt relacionats, de fet, un defineix l'altre i viceversa. També és interessant notar que el valor d'acció està format per dos termes, el primer és el retorn esperat de l'acció immediata, el segon és el valor esperat a partir del següent estat.

4 Valor d'estat òptim i equació d'optimitat de Bellman

El nostre objectiu per l'aprenentatge per reforç és trobar la millor política possible. En aquest cas, és fàcil preguntar-nos si realment existeix una política òptima; considerem com seria:

Una política π^* serà òptima si per cada $s \in S$ i tota política π possible, $v_{\pi^*}(s) \geq v_\pi(s)$. Aquesta definició ens deixa amb quatre preguntes:

- Existència: Sempre existeix una política òptima?
- Unicitat: En cas d'existir, és única?
- Estocasticitat: És possible que sigui no determinista?
- Accessibilitat: En cas d'existir, és sempre possible trobar-la algorísmicament?

4.1 Equació d'optimitat de Bellman

Primer de tot hem de decidir quan una política és millor que una altra. Direm que una política π és millor o igual que π' si $v_\pi(s) \geq v_{\pi'}(s), \forall s \in \mathcal{S}$. Això ens defineix un conjunt parcialment ordenat de polítiques. En aquest cas $\pi \geq \pi'$ si i només si π és millor o igual que π'.

Així doncs, una política π serà òptima si i només si $\pi \geq \pi', \forall \pi'$. Una definició equivalent és que una política π serà òptima si i només si $v_\pi(s) = \max_{a \in \mathcal{A}(s)} q(s,a), \forall s \in \mathcal{S}$, és a dir, sempre agafa l'acció amb el valor d'acció més alt.

Ara definim, l'equació d'optimitat de Bellman [3][5]:

$$v_\pi(s) = \max_\pi \sum_a \pi(a|s) q(s,a), \forall s \in \mathcal{S}$$

on, recordem, $q(s,a)$ és el valor d'acció, és a dir, el retorn esperat si prenem l'acció a des de s.

Aquesta equació ens permetrà estudiar polítiques òptimes. Ens podem preguntar d'on surt aquesta equació; simplement és l'equació que per cada estat s ens dona el seu valor d'estat màxim. Així doncs, si tenim que l'equació es compleix per a tots els estats, la política serà màxima.

4.2 Maximització de l'equació d'optimitat de Bellman

Per resoldre l'equació d'optimitat de Bellman hem de trobar dues incògnites, $\pi(a|s)$ i $v(s)$ per a tots els estats. Això pot semblar impossible pel fet de tenir dues incògnites per cada estat, i només una sola equació, però ara veurem que en realitat les podem maximitzar per separat.

Fixem l'estat s, aleshores, $\sum_a \pi(a|s) = 1$, ja que π és una distribució sobre l'espai d'accions. Tenim que:

$$\sum_a \pi(a|s)q(s,a) \leq \sum_a \pi(a|s) \max_a q(s,a) = \max_a q(s,a).$$

Veiem que la igualtat només es complirà si $\pi(a^*|s) = 1$ quan $a^* = \arg\max_a q(s,a)$. Tot i que a^* no és necessàriament única, sabem que haurem de triar una de les polítiques que maximitza $q(s,a)$ per cada estat amb probabilitat 1. Així doncs, és separar $q(s,a)$ de $\pi(a|s)$, maximitzant per separat les dues.

4.3 Forma matricial de l'equació d'optimitat de Bellman

Igual que l'equació de Bellman, l'equació d'optimitat és en realitat un sistema d'equacions, consistint en una equació per a cada estat. Així doncs, de manera anàloga a l'anterior i per tal de poder analitzar més eficientment aquesta equació, definirem la seva forma matricial:

$$v = \max_\pi (r_\pi + P_\pi v)$$

on r_π és el vector de la mitjana de recompenses que és pot obtenir indexades per cada estat i P_π és la matriu de la probabilitat de passar d'un estat a un altre, és a dir, la mateixa definició que s'ha fet servir anteriorment. En aquest cas pensem en la funció max aplicada element a element.

4.4 Propietat contractiva de l'equació d'optimitat de Bellman

Per resoldre l'equació optimitat de Bellman ho farem amb un mètode iteratiu. Primer hem de veure que és una funció contractiva, és a dir, volem veure que la funció $v = \max_\pi(r_\pi + P_\pi v) = f(v)$ compleix el teorema del punt fix de Banach; i per tant, $x, f(x), f(f(x)), ...$ convergeix a v amb $v = \max_\pi(r_\pi + P_\pi v)$.

El teorema del punt fix en dius que si tenim una funció que és contractiva en un espai mètric, aquesta només té un punt fix. Aleshores podem definir la sèrie $x, f(x), f(f(x)), ...$ per aproximar la solució.

Comprovem que $v = \max_\pi(r_\pi + P_\pi v)$ és contractiva:
Considerem dos vectors $v_1, v_2 \in \mathbb{R}^{|S|}$, suposem $\pi_1^* = \arg\max_\pi(r_\pi + \gamma P_\pi v_1)$ i $\pi_2^* = \arg\max_\pi(r_\pi + \gamma P_\pi v_2)$
Aleshores,

$$f(v_1) = \max_\pi(r_\pi + \gamma P_\pi v_1) = r_{\pi_1^*} + \gamma P_{\pi_1^*} v_1 \geq r_{\pi_2^*} + \gamma P_{\pi_2^*} v_1$$

$$f(v_2) = \max_\pi(r_\pi + \gamma P_\pi v_2) = r_{\pi_2^*} + \gamma P_{\pi_2^*} v_2 \geq r_{\pi_1^*} + \gamma P_{\pi_1^*} v_2$$

on $\geq$ ho considerem element a element.
Per tant,

$$\begin{aligned} f(v_1) - f(v_2) &= r_{\pi_1^*} + \gamma P_{\pi_1^*} v_1 - (r_{\pi_2^*} + \gamma P_{\pi_2^*} v_2) \\ &\leq r_{\pi_1^*} + \gamma P_{\pi_1^*} v_1 - (r_{\pi_1^*} + \gamma P_{\pi_1^*} v_2) \\ &= \gamma P_{\pi_1^*}(v_1 - v_2). \end{aligned}$$

De manera similar, es pot demostrar que $f(v_2) - f(v_1) \leq \gamma P_{\pi_2^*}(v_2 - v_1)$, la qual cosa implica $f(v_1) - f(v_2) \geq \gamma P_{\pi_2^*}(v_1 - v_2)$. Aleshores,

$$\gamma P_{\pi_2^*}(v_1 - v_2) \leq f(v_1) - f(v_2) \leq \gamma P_{\pi_1^*}(v_1 - v_2).$$

Definim

$$z = \max\left\{|\gamma P_{\pi_2^*}(v_1 - v_2)|, |\gamma P_{\pi_1^*}(v_1 - v_2)|\right\} \in \mathbb{R}^{|S|}.$$

Per definició, $z \geq 0$. Aquí, $\max(\cdot)$, $|\cdot|$ i $\geq$ són tots element per element.

$$-z \leq \gamma P_{\pi_2^*}(v_1 - v_2) \leq f(v_1) - f(v_2) \leq \gamma P_{\pi_1^*}(v_1 - v_2) \leq z,$$

el que implica $|f(v_1) - f(v_2)| \leq z$. Com que $|f(v_1) - f(v_2)| \leq z$,

$$\|f(v_1) - f(v_2)\|_\infty \leq \|z\|_\infty, \tag{2}$$

on $\|\cdot\|_\infty$ és la norma, en aquest cas, el màxim dels valors absoluts dels elements del vector. Aquí, la desigualtat encara és vàlida si la norma es substitueix per altres normes com $\|\cdot\|_2$ o $\|\cdot\|_1$. D'altra banda, suposem que z_i és l'entrada i-èssima de z, i p_i^T i q_i^T són la i-èssima fila de $P_{\pi_1^*}$ i $P_{\pi_2^*}$, respectivament. Aleshores,

$$z_i = \max\{\gamma|p_i^T(v_1 - v_2)|, \gamma|q_i^T(v_1 - v_2)|\}.$$

Ja que p_i és un vector amb tots els elements no negatius i la suma dels elements és igual a 1, tindrem que

$$|p_i^T(v_1 - v_2)| \leq p_i^T|v_1 - v_2| \leq \|v_1 - v_2\|_\infty.$$

De manera similar, tenim $|q_i^T(v_1 - v_2)| \leq \|v_1 - v_2\|_\infty$. Per tant, $z_i \leq \gamma\|v_1 - v_2\|_\infty$ i, per tant,

$$\|z\|_\infty = \max_i |z_i| \leq \gamma\|v_1 - v_2\|_\infty.$$

Substituint aquesta desigualtat a l'equació 2 obtenim

$$\|f(v_1) - f(v_2)\|_\infty \leq \gamma\|v_1 - v_2\|_\infty,$$

El que ens indica que és contractiva, i complirà la hipòtesi del punt fix.

4.5 Resolució de l'equació d'optimitat de Bellman

Arribats a aquest punt podem disposar-nos a resoldre l'equació d'optimitat de Bellman. Recordem que l'equació que volem resoldre és:

$$v = \max_\pi (r_\pi + P_\pi v)$$

Suposant que v^* és solució de l'equació, aleshores v^* és un punt fix, ja que $v^* = f(v^*)$. Per tant, com que hem vist que $f(v)$ és contractiva, existirà sempre una solució.

A més, pel teorema del punt fix, si iterem el resultat, tenim un mètode numèric que hi convergeix. Més ben dit, considerant la successió següent:

$$v_{k+1} = f(v_k)$$

v_k convergeix a v^* quan $k \to \infty$.

Ara bé, fins ara només hem estat resolent l'equació que hem plantejat, però no hem comprovat que realment v^* a més de ser solució de l'equació d'optimitat de Bellman també sigui el valor d'estat màxim de cada estat. És a dir, que $v^* = v_{\pi^*} \geq v_\pi$ per tot π.

Sabem que

$$v_\pi = r_\pi + \gamma P_\pi v_\pi$$

$$v^* = \max(r_\pi + \gamma P_\pi v^*) = r_{\pi^*} + \gamma P_{\pi^*} v^* \geq r_\pi + \gamma P_\pi v^*$$

Aleshores, tenim que:

$$v^* - v_\pi \geq (r_{\pi^*} + \gamma P_{\pi^*} v^*) - (r_\pi + \gamma P_\pi v_\pi) = \gamma P_\pi (v^* - v_\pi)$$

A partir d'aquí podem crear la successió següent:

$$v^* - v_\pi \geq \gamma P_\pi(v^* - v_\pi) \geq \gamma^2 P_\pi^2(v^* - v_\pi) \geq \cdots \geq \gamma^n P_\pi^n(v^* - v_\pi)$$

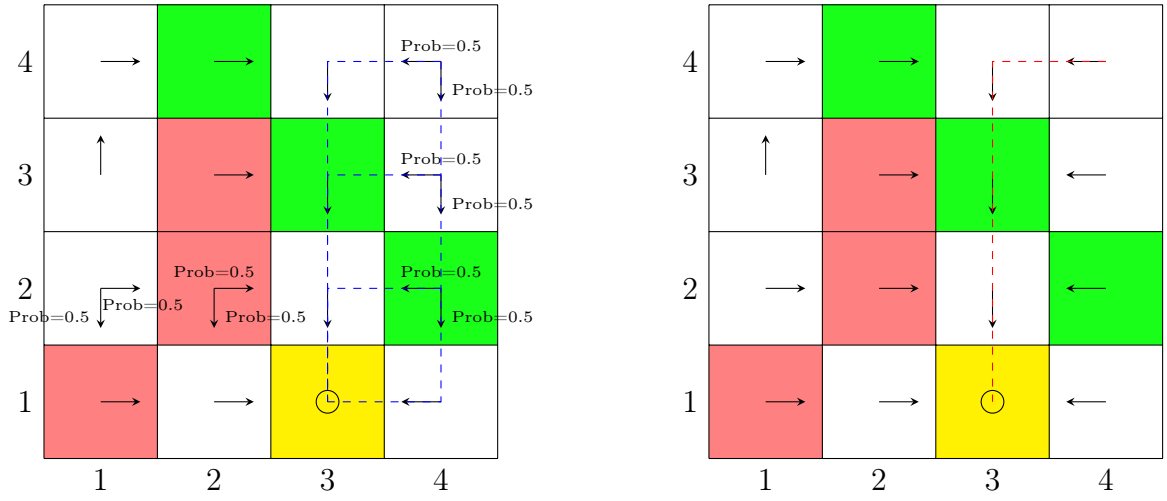
Així doncs:

$$v^* - v_\pi \geq \lim_{n\to\infty} \gamma^n P_\pi^n(v^* - v_\pi) = 0$$

ja que $\gamma < 1$ i P_π^n és no negativa i els seus elements són menors que 1, i per tant, és una funció on les files són distribucions.

Així doncs, queda demostrat que v^* és òptima. Com que ara tenim v^* podem definir π^*, com qualsevol política que compleixi que per tot estat el seu valor d'estat sigui el màxim. És a dir, com que existeix un màxim que podem agafar, $\pi^* = \arg\max_\pi(r + \gamma_\pi v^*)$ és sempre una solució de l'equació d'optimitat de Bellman. Notem que la política no és única, tot i que v^* sí que ho sigui per a cada estat.

Un fet interessant sobre la unicitat i la forma de les polítiques òptimes és el fet de que sempre tindrem una política òptima determinista, tot i que també en poden existir d'estocàstiques. Podem pensar en el següent exemple:



(a) Política òptima estocàstica

(b) Política òptima determinista

Figura 6: Exemple de polítiques òptimes

En aquest exemple veiem dues polítiques òptimes per al mateix problema, veient així que no tenim unicitat de solució per aquest problema, però també ens serveix per exemplificar que sempre que tinguem una política no determinista òptima, en podem trobar una de determinista simplement triant una de les trajectòries. Això és perquè per ser òptima una política, totes les opcions han de ser les que tenen el màxim retorn esperat, per tant, si és estocàstica, el retorn esperat de totes les opcions és el mateix, per tant, qualsevol combinació de les possibilitats té el mateix retorn esperat i podrà definir una política òptima determinista.

En el cas de l'exemple, com que tenim 5 estats on la política tria entre dues opcions, podríem trobar $2^5 = 32$ polítiques òptimes deterministes diferents només a partir d'aquesta política estocàstica.

5 Programació dinàmica

Fins ara hem estat veient la part més teòrica, aquí comencem a buscar diferents métodes per millorar les polítiques. Entenem com a programació dinàmica un conjunt d'algorismes que ens serveixen per a trobar polítiques òptimes sempre que tinguem un coneixement total sobre l'entorn. És a dir, necessitem conèixer tots els estats i les seves accions, entenent com accions, la distribució que defineix cada acció en l'espai d'estats. La programació dinàmica, doncs, té moltes limitacions, tot i això, és fonamental per la majoria d'algorismes més avançant que no necessiten un coneixement complet de l'entorn.

5.1 Avaluació de la política

Primer de tot, introduirem un algorisme per trobar els valors d'estat d'una política. Recordem que l'equació de Bellman era un sistema d'equacions lineal que ens resolia els valors d'estat de tots els estats de l'entorn, però el seu càlcul pot ser molt feixuc i en la majoria de casos és més pràctic aproximar-lo. Recordem que tenim l'equació definida a 1

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma v_\pi(s') \right].$$

Que vectorialment la podem expressar com:

$$v_\pi = r_\pi + \gamma P_\pi v_\pi.$$

Així doncs, com hem vist a la secció 3.4, aquesta equació és contractiva, per tant, podem aplicar el teorema del punt fix i obtenim un mètode per trobar amb una precisió arbitraria els valors d'estat d'una política qualsevol.

Aprofitant la naturalesa algorísmica d'aquests mètodes, per facilitat la comprensió d'aquests utilitzarem un pseudocodi. A continuació tenim el pseudocodi de l'avaluació de la política

def **avaluació de la política**(π, v_0, θ): $\rightarrow v_\pi$

:paràmetre π: política a avaluar

:paràmetre v_0: inicialització dels valors d'estat

:paràmetre θ: llindar més gran que 0 a partir del qual aturem les iteracions

:retorn v_π: valors d'estat de la política

Si $\Delta < \theta$ repeteix:

$\Delta \leftarrow 0$

$\forall s \in \mathcal{S}$ repeteix:

$v(s) \leftarrow v_0(s)$

$$v_0(s) \leftarrow \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma v(s')\right]$$

$\Delta \leftarrow \max(\Delta, |v - v_0|)$

retorna $v(s)$

És important tenir en compte que el mètode d'iteració per avaluar la política només convergirà exactament als valors d'estat en un nombre infinit de passos, per això definim un $\theta > 0$ com a llindar. Podríem considerar que per $\theta = 0$ les iteracions defineixen una sèrie que sí que convergeix exactament a v_π però algorismicament no té sentit considerar una iteració que no acabarà mai.

5.2 Millora de la política

Ara que tenim un mètode que convergeix a la política, buscarem una manera de millorar-la, que és l'objectiu principal de l'aprenentatge per reforç. Primer suposem que tenim una política π que no és òptima. En podem calcular els seu valors d'estat $v_\pi(s)$ per tot s. Ara veurem que podem millorar aquesta política si estem a un estat s i si tenim una acció a que té un valor d'acció $q_\pi(a, s)$ més gran que el valor d'estat actual $v_\pi(s)$.

Així doncs, tindríem una nova política π', on $\pi = \pi'$ per a tots estats excepte s, on prioritzarem una nova acció que compleixi que $q_\pi(a, s) > v_\pi(s)$. És important notar que aquí encara no hem avaluat la nova política π'. Veiem ara si realment ha millorat els valors d'estat respecte la política inicial.

Per fer-ho en tenim prou amb veure que $v'_\pi(s) > v_\pi(s)$, ja que, si això es compleix, com que la política només ha canviat en s, els retorns es mantindran iguals en totes les trajectòries que no passin per s i, en les trajectòries que passin per s hauran d'augmentar. Així doncs, tenim que:

$$
\begin{aligned}
v_\pi(s) &< q_\pi(s,a), \text{ on } a = \pi'(s) \\
q_\pi(s,a) &= \mathbb{E}[R_{t+1} + \gamma v_\pi(S_{t+1}) | S_t = s)] \leq \mathbb{E}[R_{t+1} + \gamma q_\pi(S_{t+1}, a) | S_t = s)] \\
&= \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 v_\pi(S_{t+2}) | S_t = s)] \\
&\leq \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 v_\pi(S_{t+3}) | S_t = s)] \\
&\vdots \\
&\leq \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} \cdots | S_t = s)] \\
&= v_{\pi'}(s)
\end{aligned}
$$

Així doncs, sempre que ens trobem en la hipòtesi de que per algun estat existeix una acció amb el valor d'acció més gran que el valor d'estat actual, podrem millorar la política. Ara bé, què passa si tenim una política que no podem millorar amb aquest sistema? Suposem que tenim una política per la qual tots els valors d'estat són més petits o iguals que el valor d'acció de les polítiques que en surten. Aleshores, vol dir que la política agafa les accions amb el valor d'acció igual al valor d'estat amb una probabilitat 1. Així doncs, estem en una situació equivalent a la de l'equació d'optimitat de Bellman on $\pi = \arg\max_\pi(r + \gamma_\pi v_p i)$ i per tant, ja és una política òptima.

def **millora de la política**(π_0, v_{π_0}): $\rightarrow \pi$

:paràmetre π: política inicial

:paràmetre v_{π_0}: valors d'estat de π_0

:retorn v_π: política millor que π_0

Si $\pi \neq \pi_0$ repeteix:

$\forall s \in \mathcal{S}$ repeteix:

$$\pi(s) \leftarrow \arg\max_a \sum_{s'} p(s', r | s, a)[r + \gamma v_{\pi_0}(s')]$$

retorna $\pi(s)$

5.3 Iteració de la política

Ara hem vist com millorar una política sempre que no sigui òptima amb el mètode anterior. Ara bé, només podem saber que una política és òptima si l'avaluem, ja que si no la podem millorar més amb el mètode descrit però hem canviat la política inicial, treballarem amb els valors d'estat de la política anterior. És a dir, si tenim una política no òptima, amb el mètode anterior podrem millorar-la. Però com que no actualitzem els valors d'estat, no podem assegurar que convergim

a una política òptima, encara que sempre millor serà que la inicial. Per solucionar això podem anar iterant el mètode d'avaluació de la política amb el de millora de la mateixa.

Així doncs, tindrem un mètode què funcionarà més o menys així:

$$\pi_0 \xrightarrow{\text{Avaluació}} v_{\pi_0} \xrightarrow{\text{Millora}} \pi_1 \xrightarrow{\text{Avaluació}} v_{\pi_1} \xrightarrow{\text{Millora}} \pi_2 \xrightarrow{\text{Avaluació}} v_{\pi_2} \cdots$$

Fins que $\pi_n = \pi_{n+1}$, que en un espai finit, trobarem en $n < \infty$ i que ens indicarà que π_n és una política òptima ja que no es pot millorar. Entenem que una política no es pot millorar quan s'ha avaluat, i per tant, se sap amb seguretat que compleix l'equació de optimitat de Bellman. En aquest cas, cal destacar que el pas de millora tant pot ser canviant una sola acció així com canviant múltiples estats de cop. El que és rellevant és que el mètode de millora sempre ens retorna una política estrictament millor que l'anterior.

def **iteració de la política**(π_0, v_0, θ): $\rightarrow \pi^*$

:paràmetre π_0: política inicial

:paràmetre v_{π_0}: valors d'estat de π_0

:paràmetre Θ: llindar a partir del qual aturem les iteracions

:retorn π^*: política òptima per θ prou petit

Si $\Delta < \theta$ repeteix:

$\Delta \leftarrow 0$

$\forall s \in \mathcal{S}$ repeteix:

$v(s) \leftarrow v_0(s)$

$$v_0(s) \leftarrow \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma v(s')\right]$$

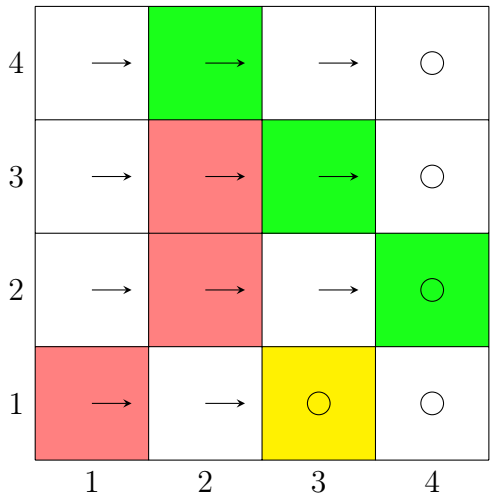
$\Delta \leftarrow \max(\Delta, |v - v_0|)$

Si $\pi \neq \pi_0$ repeteix:

$\forall s \in \mathcal{S}$ repeteix:

$$\pi(s) \leftarrow \arg\max_a \sum_{s'} p(s', r|s, a)[r + \gamma v_{\pi_0}(s')]$$

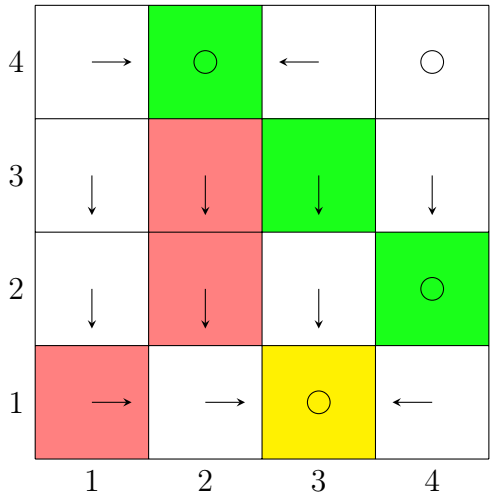
retorna $\pi(s)$



(a) Política inicial qualsevol

4	-9.1	-9	-10	-10
3	-10.9	-11	-9	-10
2	-4.5	–3.9	-1	0
1	4.2	8	10	-10
	1	2	3	4

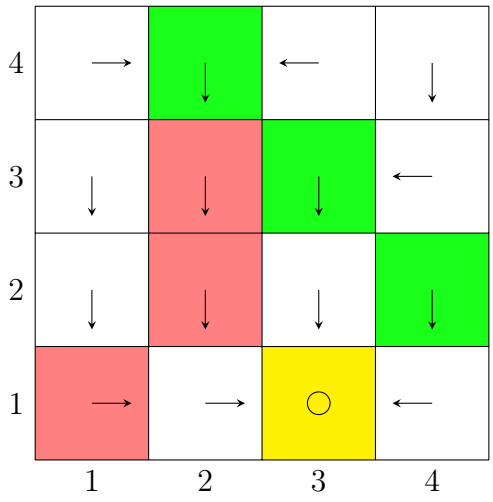
(b) Valors d'estat



(c) Política millorada una vegada

4	-1	0	-1	-10
3	1.5	0.8	7.2	-1
2	2.8	4.2	8	0
1	4.2	8	10	8
	1	2	3	4

(d) Valors d'estat



(e) Política millorada dues vegades

4	0.7	1	-0.4	-1.3
3	1.5	0.8	7.2	0.4
2	2.8	4.2	8	7.2
1	4.2	8	10	8
	1	2	3	4

(f) Valors d'estat

En aquest exemple hem iterat tres vegades el mètode per veure com evoluciona la política. En aquest cas, hem considerat que el subjecte comença en una posició qualsevol, per tant, el primer estat on es troba també influeix en els valors d'estat. Hem començat amb una política

determinista, però si haguéssim començat amb una estocàstica, hauríem acabat igualment amb una determinista.

5.4 Millores de la iteració de la política

L'eficiència del mètode d'iteració de la política descrit a 5.3 és relativament baixa. Per cada millora que podem fer hem de reavaluar tot els valors d'estat. Hem de pensar que per un espai d'estats gran aquest pas pot ser molt costós computacionalment parlant. En el nostre exemple només tenim 16 estats, però en un joc com els escacs, on cada estat seria una posició del tauler podríem estar parlant d'un nombre que té un ordre proper a 13^{64} ja que tenim 13 possibles peces diferents (comptant que la casella pot estar buida) a cada casella i 64 caselles. A més, la convergència amb els valors d'estat està garantida al límit, però evidentment de manera pràctica hem de fer un nombre limitat d'iteracions. És notable el fet que un cop millorem la política s'ha de recalculat altra vegada per tots els estats; el fet és que només necessitem saber si un valor d'estat és més gran que un altre sense importar el valor exacte d'aquests. Així doncs, tot i que de manera teòrica és molt important tenir clara la convergència, a la pràctica s'introdueixen estratègies per millorar l'eficiència dels mètodes.

Una estratègia que hem de fer servir és la d'aturar les iteracions quan la diferència entre l'últim valor i el nou valor és inferior a un llindar determinat. En aquest cas, com més petit sigui el llindar, més exactitud obtindrem en els valors d'estat, però haurem d'iterar més vegades el mètode. Per altra banda, per evitar iterar per tots els estats possibles cada vegada que aproximem els valors d'estat, podem fer-ho només per uns quants sota certes condicions. En concret, si definim una successió on tots els estats surten infinites vegades, podem actualitzar només un grup d'estats, seguin l'ordre de la successió, cada iteració i la convergència seguirà estan assegurada.

Hem de tenir clar que disminuir el nombre d'estats que actualitzem és una estratègia que pot millorar l'eficiència del mètode, però també la pot fer disminuir. Si alguns nodes clau no s'actualitzen, podríem trobar el cas en què la política no millora en alguna de les iteracions, el que podríem pensar que indica que la política és òptima, quan no necessàriament ho és.

6 Mètodes de Montecarlo

En aquest capítol introduïm nous mètodes. El mètodes de Montecarlo ens serviran per avaluar i millorar polítiques sota condicions més generals que els de programació dinàmica. Una de les seves característiques principals és que no necessiten el coneixement total de l'entorn, en comptes, el va descobrint a mesura que interactuen amb els estats a cada iteració.

6.1 Predicció de Montecarlo per valors d'estat

El primer mètode que tractarem servirà per avaluar polítiques, és a dir, calcular-ne els valors d'estat. Aquest mètode es basa en la idea d'anar fent la mitjana dels retorns que tens a partir d'una trajectòria. Entenent que una trajectòria segueix de manera aleatòria les accions segons la política que estem avaluant. En aquest cas, només sabrem la recompensa de cada acció quan l'executem.

Així doncs, per obtenir $v_\pi(s)$, aniríem generant trajectòries aleatòries, i per cada trajectòria, quan passem per s, si hi passem, en calculem el retorn. Això h podem fer per tots els estas de la trajectòria. Per la llei dels grans nombres, la mitjana dels retorns és el retorn esperat. Com que no necessàriament passarem una sola vegada per s en una trajectòria, podem diferenciar dues maneres de fer aquesta aproximació: podem agafar només el retorn de la primera vegada que hi passem o, alternativament, fer la mitjana amb totes les interaccions que tinguem amb l'estat s. Cal destacar que ambdós mètodes convergeixen a $v_\pi(s)$ i els podem anomenar mètode de la primera interacció i mètode de totes les interaccions respectivament. En aquest treball es farà servir el mètode de primera interacció.

def **predicció de Montecarlo per valors d'estat**(π, N): $\rightarrow v_\pi(s)$

:paràmetre π: política a evaluar

:paràmetre N: nombre d'exploracions

:retorn $v_\pi(s)$: valor d'estat de s amb la política

$n \leftarrow 1$

$v(s) \leftarrow 0$

Si $n < N$ repeteix:

Executa una trajectòria seguint π

$v(s) \leftarrow \frac{n-1}{n} v(s) + \frac{1}{n} \text{retorn}(s)$

$n \leftarrow n + 1$

retorna $v(s)$

On el retorn(s) és la suma de totes les recompenses que es reben a partir de l'estat s. Igual que amb el mètode de programació dinàmica, per infinites iteracions tindrem convergència al valor d'estat sempre que passem infinites vegades per aquest estat.

6.2 Predicció de Montecarlo per valors d'acció

El mateix mètode que hem definit per als valors d'estat pot ser aplicat als valors d'acció. Començarem a un estat i anirem seguint una política, aleshores, cada vegada que prenem una acció per primera vegada en calcularem el retorn i finalment, amb vàries repeticions d'aquest procés farem la mitjana del retorn esperat d'aquella acció. Anàlogament, per la llei dels grans nombres, tenim un mètode que convergeix als valors d'acció.

Ara bé, no podem perdre de vista el nostre objectiu; volem millorar la política. Aquest mètode de càlcul pels valors d'estat ens genera un inconvenient per aquest objectiu, només trobarem el valor d'acció de les accions amb probabilitat d'esdevenir-se no nul·les. És a dir, les accions que per la política no prenguem mai, no podrem saber si millorarien o no la política, ja que no tindrem mostra per calcular el seu valor d'acció. Aquest problema és especialment aparent si pensem en una política determinista. En aquest cas, per cada estat la política només contempla una sola acció, així doncs, podrem trobar amb l'exactitud que desitgem els valors d'acció de totes les accions que tenen probabilitat 1, però com que no tenim cap altre valor per comparar-les no serà possible valorar si la política és bona o dolenta, ni sabrem com millorar-la. Una manera de no trobar-se amb aquest problema és crear polítiques on qualsevol parella estat-acció té una probabilitat no nul·la de ser escollida.

6.3 Iteració de la política amb Montecarlo

Seguint la mateixa idea que amb la iteració del capítol de programació dinàmica, iterant els mètodes d'aproximació nous i aprofitant el mateix mètode de millora de la política que a la secció 5.2, podem definir un mètode que intercali iteracions del dos per aconseguir aproximar una política òptima sense conèixer l'entorn. En aquest cas, necessitem començar amb una política que ens permeti accedir a tots els estats per poder garantir la convergència a la política òptima, ja que altrament hi ha accions que no podrem descobrir mai.

def **Gradients de la política amb Montecarlo**(π_0): $\to \pi^*$

:paràmetre π_0: política inicial (ha de ser no nul·la per tota parella (a, s))

:retorn π^*:política òptima

$\pi \leftarrow \pi_0$

Tantes vegades com vulguis, repeteix:

Executa una trajectòria seguint π

Per tot $s \in \mathcal{S}$, si passes per s:

$$q(s) \leftarrow \frac{n-1}{n} v(s) + \frac{1}{n} \text{retorn}(s)$$

$$\pi(s) \leftarrow \arg\max_a q(s, a)$$

retorna π

Recordem que estem suposant que les trajectòries passen per totes les parelles estat-acció. Però això és una condició molt forta que ens agradaria treure. Ho analitzarem a continuació.

6.4 Iteració amb Montecarlo generalitzada

Com podem evitar dependre de la "sort" de trobar-nos amb polítiques que ens permetin accedir a totes les accions? Recordem que necessitem que totes les parelles estat-acció siguin visitades indefinidament per a assegurar la convergència a una política òptima. Per tal de corregir aquest defecte, canviarem una mica el sistema de millora de la política. Això ens canviarà lleugerament la política, però ara veurem que no ens suposarà un problema.

La idea és que les polítiques que fem servir siguin quasi deterministes en lloc de deterministes. Amb això volem dir que tot i que exigirem que $\pi(a|s) > 0$, permetrem que aquests valor es facin arbitràriament petits per tal d'apropar-nos a la política determinista a la qual estem acostumats a convergir. Així doncs, definirem un ε per fitar el límit inferior del valor que pot tenir $\pi(a|s)$. Concretament, fitarem de la següent manera:

$$\pi(a|s) > \frac{\varepsilon}{|\mathcal{A}(s)|}$$

on recordem que $\mathcal{A}(s)$ és l'espai d'accions per a l'estat s. D'aquesta manera, enlloc d'assignar probabilitat 1 a l'acció amb el màxim valor d'acció; el que farem és assignar-li $1 - \varepsilon + \frac{\varepsilon}{|\mathcal{A}(s)|}$ i llavors assignar $\frac{\varepsilon}{|\mathcal{A}(s)|}$ a la resta d'accions possibles. Adaptem el pseudocodi anterior per als nous canvis:

def **Gradients de la política amb Montecarlo**(π_0, ε): $\to \pi^*$

:paràmetre π_0: política inicial (ha de ser no nul·la per tota parella (a, s))

:paràmetre ε: llindar inferior positiu de la $\pi(a|s)$

:retorn π^*:política òptima

$\pi \leftarrow \pi_0$

Tantes vegades com vulguis, repeteix:

Executa una trajectòria seguint π

Per tot $s \in \mathcal{S}$, si passes per s:

$$q(s) \leftarrow \frac{n-1}{n} v(s) + \frac{1}{n} \text{retorn}(s)$$

$$\pi(s) \leftarrow \arg\max_a q(s, a)$$

Per tot $a \in \mathcal{A}(s)$:

si $a = \arg\max_a q(s, a)$

$$\pi(s|a) \leftarrow 1 - \varepsilon + \frac{\varepsilon}{|\mathcal{A}(s)|}$$

si $a \neq \arg\max_a q(s, a)$

$$\pi(s|a) \leftarrow \frac{\varepsilon}{|\mathcal{A}(s)|}$$

$\varepsilon \leftarrow 0$

retorna π

En aquest cas, a l'infinit tenim convergència igual que abans a una política determinista òptima. Es pot veure que mai podem estar segurs si la política que tenim és òptima donat que els valors d'estat i d'acció son aproximats amb el mètode de Montecarlo i podrien variar. Així doncs, només podem anar repetint indefinidament l'algorisme fins que tinguem un resultat suficientment bo, sense condició per aturar quan arribem a una política concreta. S'ha de tenir en compte, però, que només pot existir una política lleugerament millor. Concretament l'error que cometem al aproximar un valor d'estat és el mateix que pot millorar la política. Això no és una qualitat exclusiva d'aquest mètode.

Tots els mètodes que hem definit estan pensats per a no haver de calcular exactament els valors d'acció ja que és un procés que implica invertir matrius de dimensions enormes, concretament $|\mathcal{S}| \times |\mathcal{S}|$. Pensem que en el nostre exemple només hi ha 16 estats, però en un joc de taula n'hi pot haver milers de milions com hem vist abans amb els escacs. Per posar un altre exemple, un joc molt menys complicat que els escacs com el parxís, que té 16 fitxes (4 de cada color), i que poden estar en 77 posicions cada una, per tant, obviant que hi ha posicions

impossibles, com tres fitxes en una mateixa casella, estaríem parlant de $\left(\frac{80!}{4!76!}\right)$ estats, de l'ordre de 10^{24} i aleshores, la matriu tindria uns 10^{48} elements. Així doncs, molt ràpidament se'ns fa impossible treure el resultat a força de computació.

7 Mètodes de gradient de la política

En aquesta secció introduïm un nou tipus de mètodes. Fins ara hem basat els mètodes en triar l'acció a partir del càlcul dels valors d'estat o d'acció. En aquests nous mètodes no sempre farem servir aquests valors. En els mètodes de gradient considerarem un vector de paràmetres θ que influirà la política. Així doncs, ara tenim que $\pi(a|s,\theta) = \mathbb{P}(A_t = a|S_t = s, \theta_t = \theta)$ com la probabilitat de que l'acció a sigui l'escollida si estem a l'estat s amb el paràmetre de la política θ. El fet que introduïm un paràmetre per alterar l'acció escollida no vol dir que necessàriament no utilitzem una funció per valorar com de bona és una política.

Els mètodes de gradient de la política cerquen el valor del paràmetre a través del gradient d'una funció del propi paràmetre, aquesta funció escalar la denotarem que $J(\theta)$. Així doncs, buscarem maximitzar el valor de la funció seguint el gradient de la mateixa respecte el paràmetre. Podem expressar la idea com:

$$\theta_{t+1} = \theta_t + \alpha \nabla J(\theta) \tag{3}$$

On α és la longitud dels passos que fem. És a dir, per α molt gran, la política evolucionarà molt ràpid, però la precisió serà més baixa; contràriament, per α molt petit avançarem molt lentament, però seguirem una corba de màxima pendent amb molta exactitud. Així doncs, sovint ens interessa començar amb una α relativament alta i disminuir-la a mesura que ens aproximem al valor final.

Els principals avantatges d'aquest mètode són el fet de que podem considerar espais d'estat no discrets. També podem reduir el nombre de dimensions del problema en el cas que sigui discret, ja que no necessàriament la dimensió de θ ha de ser tan gran com el cardinal de l'espai d'estats. En definitiva, computacionalment podem reduir el cost dels mètodes anteriors.

Pel que fa a les limitacions, centrant-nos en la funció que determina la política, $\pi(a|s,\theta)$, ha de complir certs criteris. En un espai discret, la suma dels valors per tot $s \in \mathcal{S}$ ha de ser igual a 1, ja que ha de formar una distribució de probabilitat. És a dir:

$$\sum_s \pi(a|s,\theta) = 1, \forall \theta$$

altrament, en un espai d'estats continu, ha d'integrar 1 en aquest espai.

$$\int_s \pi(a|s,\theta) = 1, \forall \theta.$$

Una altra característica que han de tenir aquestes funcions és ser derivables respecte el paràmetre, ja que ens seria impossible calcular el gradient si no fos així.

Si tenim un espai d'accions finit, un dels exemples de funció amb aquestes propietats més

importants és la funció soft-max, que ens retorna una distribució de probabilitat a partir d'un vector de preferències v. Aquestes preferències s'expiressen com a nombres reals; sent més exactes, si la posició i del vector v conté un valor més gran que la resta, l'acció corresponent a i, serà la més probable de totes. Per tant, una puntuació de preferència més gran resultarà en una probabilitat més gran d'escollir l'acció. La funció soft-max per a a si tenim n accions possibles és la següent:

$$\pi(a|s,\theta) = \frac{e^{v_\theta(a)}}{\sum_{i=1}^{n} e^{v_\theta(i)}}$$

És a dir, quan el vector paràmetre té un valor θ i estem a l'estat s, amb la funció soft-max podem definir la probabilitat de triar l'acció a d'aquesta manera. Podem comprovar que és una distribució ja que l'exponencial sempre serà no negatiu i a més, la suma de les probabilitats de les accions és 1.

7.1 Mètriques per a polítiques òptimes

Si fem servir una funció de θ com a política, com determinem quan és òptima? La primera mètrica pot ser la mitjana dels valors d'estat. En aquest cas, podem determinar uns pesos d_π per a cada estat de manera que, amb $\sum d_\pi(s) = 1$ ens queda una mitjana ponderada:

$$\bar{v}_\pi = \sum_s d_\pi(s) v_\pi(s)$$

o vectorialment:

$$\bar{v}_\pi = d_\pi^T v_\pi$$

Ara bé, aquests pesos d'on surten i perquè serveixen? Doncs podríem agafar una distribució uniforme on $d_\pi(s) = \frac{1}{|\mathcal{S}|}$, però si hi ha estats més concorreguts, és millor assignar un pes major a aquests. Per exemple, podríem pensar agafar

$$d_\pi(s) = \mathbb{P}(S_t = s)$$

és a dir, la probabilitat de que en un moment qualsevol l'estat on estiguem sigui s. D'aquesta manera, els estat que quasi no es visiten no tenen tanta importància com els estats on és molt probable passar. Una altra possibilitat és considerar l'equació

$$d_\pi^T P_\pi = d_\pi^T$$

on P és la matriu probabilitat de transició d'un estat a un altre que hem fet servir anteriorment. Podem observar que com que $\sum d_\pi(s) = 1$, en les dues definicions tindrem els mateixos pesos, és a dir, la columna i de P és la probabilitat de cada estat a acabar a l'estat i, d'aquesta manera, si ho multipliquem per d_π^T, tindrem la probabilitat total d'acabar a aquell estat.

Una altra mètrica habitual és la mitjana de les recompenses esperades de cada estat. Aquí considerem les recompenses de cada estat com $r_\pi(s) = \sum_a \pi(a|s)r(a,s)$. La mètrica també podrà ser una mitjana ponderada igual que en la mitjana dels valors d'estat. Així doncs, la mètrica la podrem representar com:

$$\bar{r}_\pi = \sum_s d_\pi(s) r_\pi(s)$$

o vectorialment:

$$\bar{r}_\pi = d_\pi^T r_\pi$$

on d_π són pesos igual que en la primera mètrica.

Per últim, en un cas més concret, la millor mètrica és el valor d'estat d'un estat determinat. Aquesta mètrica és útil quan el problema sempre ha de tenir el mateix estat inicial, ja que només ens interessa maximitzar el valor d'aquest estat en concret, independentment dels altres estats. Realment aquesta mètrica és un subcas de la primera, on els pesos tots son nuls excepte el de l'estat inicial que és 1.

Així doncs, hem definit unes funcions que com que totes depenen de π, tenen com a paràmetre θ. Aquestes són algunes de les funcions que volem maximitzar aprofitant el gradient respecte θ.

7.2 Gradient de les mètriques

En aquest apartar calculem el gradient de les mètriques que hem definit abans. Primer veiem que en el cas que la taxa de descompte $\gamma < 1$, les dues mètriques són proporcionals, concretament:

$$(1-\gamma)\bar{v}_\pi = \bar{r}_\pi$$

i per tant, ens donen la mateixa informació, ja que són tenen una dependència lineal. Veiem que realment són iguals:

Recordant l'equació de Bellman en la seva forma vectorial: $v_\pi = r_\pi + \gamma P_\pi$
Tenint que $\bar{v}_\pi = d_\pi^T v_\pi, \bar{r}_\pi = d_\pi^T r_\pi$ i $d_\pi^T P_\pi = d_\pi^T$.
Multiplicant per d_π^T als dos costats de l'equació de Bellman obtindrem:
$\bar{v}_\pi = \bar{r}_\pi + \gamma d_\pi^T P_\pi v_\pi = \bar{r}_\pi + \gamma \bar{v}_\pi$
D'on surt la identitat que es volia demostrar.

Vist això, hem comprovat que només necessitarem calcular un sol gradient per les tres mètriques que havíem plantejat al començament del capítol. Així doncs calculem quin és aquest gradient:

$$\nabla_\theta \bar{r}_\pi(\theta) \simeq \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

Aquesta expressió és una igualtat si $\gamma = 1$ i una aproximació altrament. Com més proper a 1 és el valor de γ, millor és l'aproximació. Això quedarà més clar quan es calculi el gradient, però la idea és que quan γ tendeix a 1, hi ha uns termes que tendeixen a 0. No els conversem perquè impliquen un altre gradient per la regla de la cadena que no podem calcular. A més, podem expressar el gradient d'una forma tancada més simple a través de l'esperança:

$$\nabla_\theta \bar{r}_\pi(\theta) \simeq \mathbb{E}\left[\nabla_\theta \log \pi(A|S,\theta) q_\pi(S,A)\right]$$

on $S \sim d_\pi$ i $A \sim \pi(s)$

La segona expressió no es demostrarà ja que fa servir resultats llargs que no s'han fet servir fins ara, però es deixa la referencia [5] per si el lector hi té especial interès. Ara es demostra la primera expressió, calculant primer l'expressió de $\nabla_\theta \bar{v}_\pi$. Volem veure que:

$$\nabla_\theta \bar{v}_\pi \simeq \frac{1}{1-\gamma} \sum_s d^\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a)$$

i

$$\nabla_\theta \bar{r}_\pi \simeq \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

La correspondència entre una i l'altre bé donada pel terme lineal que les feia proporcionals com hem vist a , on hem vist que eren proporcionals amb un factor de $1-\gamma$. Així doncs, calculem només el primer gradient. Per calcular el gradient de $\bar{v}_\pi$, primer calcularem el gradient de $v_\pi(s)$.

$$\nabla_\theta v_\pi(s) = \sum_{s'} \mathbb{P}_\pi(s'|s) \sum_a \nabla_\theta \pi(a|s',\theta) q_\pi(s',a)$$

on

$$\mathbb{P}_\pi(s'|s) = \sum_{k=0}^{\infty} \gamma^k \mathbb{P}(S_{t+k} = s'|S_t = s, \pi) = [(I - \gamma P_\pi)^{-1}]_{ss'}$$

és la probabilitat amb descompte de passar de l'estat s a s' amb la política π. Veiem que $\mathbb{P}(S_{t+k} = s'|S_t = s, \pi)$ és la probabilitat de passar de s a s' en exactament k passos si la política que seguim és π.

Primer, per a qualsevol $s \in S$, es compleix que

$$\nabla_\theta v_\pi(s) = \nabla_\theta \left[\sum_a \pi(a|s,\theta) q_\pi(s,a) \right] = \sum_a \left[\nabla_\theta \pi(a|s,\theta) q_\pi(s,a) + \pi(a|s,\theta) \nabla_\theta q_\pi(s,a) \right]$$

on $q_\pi(s,a)$ és el valor de l'acció donat per

$$q_\pi(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) v_\pi(s').$$

Aquí, $r(s,a) = \sum_r r p(r|s,a)$ és independent de θ. Per tant,

$$\nabla_\theta q_\pi = 0 + \gamma \sum_{s'} p(s'|s,a) \nabla_\theta v_\pi(s'),$$

substituint això a l'equació anterior s'obté

$$\nabla_\theta v_\pi(s) = \sum_a \left[\nabla_\theta \pi(a|s,\theta) q_\pi(s,a) + \pi(a|s,\theta) \gamma \sum_{s'} p(s'|s,a) \nabla_\theta v_\pi(s') \right] \tag{4}$$

És notable que $\nabla_\theta v_\pi$ apareix als dos costats de l'equació anterior. Per calcular $\nabla_\theta v_\pi$, podem utilitzar la tècnica de descompressió [4]. Aquí, utilitzem la forma vectorial, ja que és més simple que la element a element. En particular, sigui

$$u(s) = \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

Fixem-nos que l'equació 4 es pot escriure en forma vectorial com

$$\begin{bmatrix} \vdots \\ \nabla_\theta v_\pi(s) \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ u(s) \\ \vdots \end{bmatrix} + \gamma (P_\pi \otimes I_m) \begin{bmatrix} \vdots \\ \nabla_\theta v_\pi(s') \\ \vdots \end{bmatrix}$$

que es pot reescriure com

$$\nabla_\theta v_\pi = u + \gamma (P_\pi \otimes I_m) \nabla_\theta v_\pi.$$

On m és la dimensió del paràmetre θ i $\otimes$ és el producte de Kronecker, que s'explica més endavant a 6. El producte de Kronecker apareix perquè $\nabla_\theta v_\pi$ ja és un vector, així doncs necessitem una 'matriu de matrius' per tal que el resultat sigui un vector.

L'equació anterior és una equació lineal de $\nabla_\theta v_\pi$, que es pot resoldre com:

$$\nabla_\theta v_\pi = (I_{nm} - \gamma P_\pi \otimes I_m)^{-1} u = (I_n \otimes I_m - \gamma P_\pi \otimes I_m)^{-1} u = (I_n - \gamma P_\pi)^{-1} \otimes I_m u$$

la forma element per element de la solució és:

$$\nabla_\theta v_\pi(s) = \sum_{s'} (I_n - \gamma P_\pi)^{-1}_{ss'} \, u(s') = \sum_{s'} (I_n - \gamma P_\pi)^{-1}_{ss'} \sum_a \nabla_\theta \pi(a|s', \theta) q_\pi(s', a)$$

on $[\cdot]_{ss'}$ és l'element a la fila s i la columna s'. La quantitat $[(I_n - \gamma P_\pi)^{-1}]_{ss'}$ té una interpretació clara en termes de probabilitat. En particular, ja que:

$$(I_n - \gamma P_\pi)^{-1} = I + \gamma P_\pi + \gamma^2 (P_\pi)^2 + \cdots,$$

tenim que:

$$(I_n - \gamma P_\pi)^{-1}_{ss'} = [I]_{ss'} + \gamma [P_\pi]_{ss'} + \gamma^2 [(P_\pi)^2]_{ss'} + \cdots = \sum_{k=0}^{\infty} \gamma^k [(P_\pi)^k]_{ss'}.$$

s'ha de tenir en compte que $[(P_\pi)^k]_{ss'}$ és la probabilitat de passar de s a s' utilitzant exactament k passos; per tant, $[(I_n - \gamma P_\pi)^{-1}]_{ss'}$ és la probabilitat total (descomptada) de passar de s a s' utilitzant qualsevol nombre de passos.

Ara tenim el valor del gradient de $v_\pi(s)$, però recordem que estem buscar el gradient de $\bar{v}_\pi$, així doncs, considerem la definició d'aquest:

$$\nabla_\theta \bar{v}_\pi = \nabla_\theta \sum_s d_\pi(s) v_\pi(s) = \sum_s \nabla_\theta d_\pi(s) v_\pi(s) + \sum_s d_\pi(s) \nabla_\theta v_\pi(s) \tag{5}$$

Aquesta equació conté dos termes. D'una banda, substituint l'expressió de $\nabla_\theta v_\pi(s)$ al segon terme obtenim:

$$\begin{aligned}
\sum_s d_\pi(s) \nabla_\theta v_\pi(s) =& (d_\pi^T \otimes I_m) \nabla_\theta v_\pi \\
=& (d_\pi^T \otimes I_m) \left((I_n - \gamma P_\pi)^{-1} \otimes I_m \right) u \\
=& d_\pi^T (I_n - \gamma P_\pi)^{-1} \otimes I_m u
\end{aligned}$$

Notem que

$$d_\pi^T (I_n - \gamma P_\pi)^{-1} = \frac{1}{1-\gamma} d_\pi^T,$$

es pot veure fàcilment multiplicant $(I_n - \gamma P_\pi)$ a ambdós costats de l'equació.

Per tant, l'equació es converteix en:

$$\sum_s d_\pi(s)\nabla_\theta v_\pi(s) = \frac{1}{1-\gamma} {d_\pi}^T \otimes I_m \, u = \frac{1}{1-\gamma} \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

D'altra banda, el primer terme de l'equació 5 implica $\nabla_\theta d_\pi$. No obstant això, atès que el segon terme conté $\frac{1}{1-\gamma}$, aquest segon terme es torna dominant i el primer terme es torna insignificant quan $\gamma \to 1$. Per tant,

$$\nabla_\theta \bar{v}_\pi \simeq \frac{1}{1-\gamma} \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

A més, es desprèn de $\bar{r}_\pi = (1-\gamma)\bar{v}_\pi$ que

$$\nabla_\theta \bar{r}_\pi = (1-\gamma)\nabla_\theta \bar{v}_\pi \simeq \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

L'aproximació anterior requereix que el primer terme no tendeixi a l'infinit quan $\gamma \to 1$.

El producte de Kronecker, que és un cas especial de producte de tensors, el definim amb un exemple de la següent manera:

Exemple de **Producte de Kronecker:**
Si tenim les següents matrius:

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} \tag{6}$$

el seu producte de Kronecker $A \otimes B$ es defineix com:

$$A \otimes B = \begin{pmatrix} 1 \cdot B & 2 \cdot B \\ 3 \cdot B & 4 \cdot B \end{pmatrix}$$

aleshores, calculant cada bloc, obtenim:

$$A \otimes B = \begin{pmatrix} 1 \cdot \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} & 2 \cdot \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} \\ 3 \cdot \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} & 4 \cdot \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} & \begin{pmatrix} 0 & 10 \\ 12 & 14 \end{pmatrix} \\ \begin{pmatrix} 0 & 15 \\ 18 & 21 \end{pmatrix} & \begin{pmatrix} 0 & 20 \\ 24 & 28 \end{pmatrix} \end{pmatrix}$$

unificant les matrius blocs, obtenim la matriu resultant:

$$A \otimes B = \begin{pmatrix} 0 & 5 & 0 & 10 \\ 6 & 7 & 12 & 14 \\ 0 & 15 & 0 & 20 \\ 18 & 21 & 24 & 28 \end{pmatrix}$$

És fàcil veure doncs que el producte de Kronecker és una operació entre dues matrius de mida qualsevol. A més no és commutatiu, tot i que no és rellevat en el nostre cas.

7.3 Gradient de la política amb Montecarlo

Ara que tenim els gradients, podem utilitzar aquests per millorar les polítiques a partir del mètode de gradient que s'ha definit al començament de la secció. Recordem el que havíem considerat a 3, al principi del capítol:

$$\theta_{t+1} = \theta_t + \alpha \nabla J(\theta)$$

on α és la mida dels canvis que fem al paràmetre. Ara podem substituir $J(\theta)$ per la mètrica que es vulgui.

def **Gradients de la política amb Montecarlo**(π, θ_0): $\rightarrow \theta^*$

:paràmetre π: política depenent d'un paràmetre θ

:paràmetre θ_0: valor inicial del paràmetre

:retorn θ^*: paràmetre θ que maximitza $J(\theta)$

$\alpha \leftarrow \alpha_0$

Tantes vegades com vulguis, repeteix:

Executa una trajectòria seguint π

Aproxima $\nabla J(\theta)$

$\theta \leftarrow \theta + \alpha \nabla J(\theta)$

retorna θ

Aquest és l'últim mètode que tractem. És interessant perquè és més flexible que totes les anterior. També té la avantatja que computacionalment, pot ser molt més fàcil aproximar valors de funcions com els gradients que quantitats molt grans d'operacions com en el cas dels valors d'estat. Cal destacar que la mètrica també és pot minimitzar si ho necessitem en algun cas concret, en aquest cas només hem de posar un α negatiu. A més, quan utilitzem un mètode de Montecarlo, no necessitem un coneixement complet de l'entorn.

Referències

[1] M. Pinsky, S. Karlin, *An introduction to stochastic modeling (3rd Edition),*Academic Press, *(1998)*
https://appliedmath.arizona.edu/sites/default/files/0f04d86a836182cbf608dfc86c7a70f5e5f6_0.pdf

[2] M. L. Puterman, *Màrkov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons. *(2009)*
Es pot trobar a la biblioteca de la facultat.

[3] R. S. Sutton, A. G. Barto, *Reinforcement Learning, An Introduction, second edition.* MIT Press, *(2018)*
https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

[4] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, *Policy gradient methods for reinforcement learning with function approximation, (1999)*,
https://proceedings.neurips.cc/paper_files/paper/1999/file/464d828b85b0bed98e80ade0a5c43b0f-Paper.pdf

[5] S. Zhao, *Mathematical Foundation of Reinforcement Learning*, *(2022)*,
https://github.com/MathFoundationRL/Book-Mathematical-Foundation-of-Reinforcement-Learning