מערכות דינאמיות ובקרה

לביא שפיגלמן

מבוא לReinforcement Learning

- הקשר בין DP וRL

הקשר בין DP וRL

כזכור, בבעית הבקרה הדיסקרטית, משוואת הדינאמיקה היא

$$x_{t+1} = f(x_t, u_t, t)$$

הcost to go, (שסומן כJ^0 או V^0) על המסלול האופטימלי הוא

$$V^0(x_k, k) = \min_{\mathbf{u}} \left\{ \phi(x_N, N) + \sum_{t=k}^{N-1} L(x_t, u_t, t) \right\}$$

או, כנוסחת נסיגה

$$V^0(x_k, k) = \min_{u_k} \left\{ L(x_k, u_k, k) + V^0\left(f(x_k, u_k, k), k+1\right) \right\}$$

בDP משתמשים בנוסחא הנ"ל (ובפונקציית המחיר הרגעי, L ומשוואות הדינאמיקה, f) למציאת $V^0(x_k, k)$ לכל x וk ע"י איטרציה. כפי שצויין, זמן החישוב, (עבור מספר צעדים, N, הוא $\mathcal{O}(N|\mathcal{U}||\mathcal{X}|)$) עלול להיות גדול ופונקליות הדינאמיקה והמחיר הרגעי לא תמיד ידועות.

התחום, Reinforcement Learning (RL) בא להתמודד עם הבעיות הנ"ל. לשם כך יש צורך בהתאמת מושגים קלה. הloss הרגעי נלקח כשלילי ונקרא reinforcement, מחיר הנקודה הסופית נלקח גם הוא כשלילי ונקרא גם כן reinforcement.

כשם שבבעיית בקרה האופטימלית רצינו למזער את המחיר הכולל, בRL רוצים למקסם את (תוחלת) סכום הreinforcements.

סימונים (ותרגומם):

שם בRL	סימון בRL	סימון בבקרה	שם בבקרה
state	$s \in S$	$\mathbf{x}$	state
action	$a \in A(S)$	$\mathbf{u}$	control sig.
(stochastic) policy	$\pi(s,a) = Pr(a\|s)$	$\mathbf{u}(\mathbf{x},t)$	control function
(stochastic) state transition	$P^a_{ss'}$	$\mathbf{f}(\mathbf{x},\mathbf{u},t,\omega)$	state dynamics
(stochastic) reward	r	$-L$	instantenous loss
commulative expected reward	V	$-J$,$-V$	cost to go

התמונה הטיפוסית בRL היא של סוכן (agent) שחש את סביבתו באופן חלקי ומייצג את ההערכה שלו של מצבו בעולם ע"י s. בכל צעד, הסוכן מבצע פעולה באופן סטוכסטי (או דטרמיניסטי). התפלגות הפעולות מיוצגת ע"י $Pr(a|s) = \pi(s,a)$ (במונחים של בקרה, $\mathbf{u}(\mathbf{x})$ זו פונקציה הסתברותית של המצב). לאחר ביצוע הפעולה המצב משתנה ע"פ התפלגות (ידועה או לא ידועה)

$$P^a_{ss'} = Pr\left\{s_{t+1} = s' | s_t = s, a_t = a\right\}$$

במונחים של בקרה $P^u_{xx'}$ מקבילה להסתברות ש$x' = f(x,u,\omega)$ (הסטוכסטיות נובעת מרעש תהליך ω). לאחר מכן מתקבל reward (loss במונחים של בקרה). הreward ניתן מתוך התפלגות ותוחלתו

$$R^a_{ss'} = E_{r_{t+1}}\left\{r_{t+1} | s_t = s\ ,\ a_t = a\ ,\ s_{t+1} = s'\right\}$$

מטרת הסוכן היא לבצע policy אופטימלי, כלומר לבצע פעולות כך שסכום הrewards יהיה מירבי. סכום זה נלקח בדרך כלל כסכום אינסופי מונחת (discounted). הסכום של ריצה ספציפית יסומן כך:

$$\begin{aligned} R_t &= r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... \\ &= r_{t+1} + \gamma R_{t+1} \end{aligned}$$

ותוחלת הסכום (הרווח) תסומן

$$\begin{aligned} V^\pi(s) &= E_\pi\left\{R_t | s_t = s\right\} \\ &= E_\pi\left\{r_{t+1} + \gamma V\left(s_{t+1}\right) | s_t = s\right\} \\ &= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'}\left[R^a_{ss'} + \gamma V^\pi(s')\right] \end{aligned} \quad (1)$$

זה כמובן הcost to go (בהצגה סטוכסטית), $J(x,t)$.נשים לב כי כאן מניחים שההתפלגויות אינן תלויות בזמן (הנחת סטציונריות) והמחיר הוא עבור זמן ריצה אינסופי (תוך דעיכת הrewards ע"י גורם γ) ולכן הפונקציה היא של המצב בלבד. במקרה זה פתרון הDP מצריך מילוי של וקטור ערכים, $\mathbf{v}$, על סמך ערכו הקודם במקום מילוי של עמודות עוקבות במטריצה.

המחיר תחת בקרה אופטימלית, $J^0(x,t)$, יסומן כ

$$V^*(s) = \max_\pi V^\pi(s)$$

והpolicy האופטימלי הוא $\pi^*(s)$. כפי שהגדרנו את $J^1(\mathbf{x},\mathbf{u},t)$ להיות מחיר ביצוע אות בקרה כלשהו, $\mathbf{u}$ ולאחריו ביצוע בקרה אופטימלית, כך בRL מוגדרת פונקציית

הoptimal action-value (באופן רקורסיבי):

$$\begin{aligned} Q^*(s,a) &= E\left\{r_{t+1} + \gamma V^*(s_{t+1})\right\} \\ &= E\left\{r_{t+1} + \gamma \max_{a'} Q^*(s',a')\right\} \\ &= \sum_{s'} P^a_{ss'}\left[R^a_{ss'} + \gamma \max_{a'} Q^*(s',a')\right] \end{aligned}$$

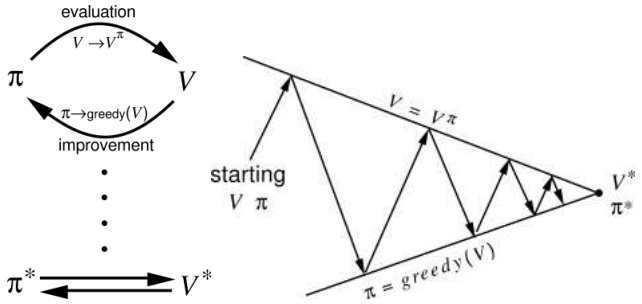
זה האנלוג לתנאי האופטימליות של Bellman מבקרה אופטימלית. הpolicy האופטימלי במונחים של Q הוא

$$\pi^*(s) = \arg\max_{a \in A(s)} Q^*(s,a)$$

כשם שמצאנו את $J(x,t)$ (ואת $J^1(x,u,t)$) בשיטות DP, ניתן למצוא (בהינתן פונקציות L ו$\mathbf{f}$, סטוכסטיות, כלומר בהינתן $P^a_{ss'}$ ו$R^a_{ss'}$) policy אופטימלי באותה סיבוכיות זמן ריצה.

כשהדינאמיקה לא ידועה במלואה או המחיר הרגעי אינו ידוע במלואו או זמן הריצה אינו ארוך דיו (כפי שקורה למשל בשש בש בו יש כ-10^{20} מצבים) RL מציב אלגוריתמים שמתכנסים (בד"כ, תחת תנאים מסוימים אך ריאלים) לפיתרון האופטימלי (כלומר מתכנסים לקיום תנאי האופטימליות של Bellman). ישנה גם הרחבה של RL לזמנים רציפים.

כל שיטות הRL וכן DP (וגם EM) מבצעות איטרציות בהן בצעד אחד ישנו הערכה של הvalue function עבור הpolicy הנתונה וצעד שנו בו יש שיפור של הpolicy על סמך הvalue function המעודכנת.



שיטות מעין זו מכונות Generalized Policy Iteration (GPI) הDP הקלאסי שהוצג כאן מבצע סריקה של כל המצבים ועדכון של $V(s)$ לכל s לפני שπ מעודכן [1]. דבר זה אינו הכרחי. Asychnous DP מעדכן את $V(s)$ עבור קבוצה חלקית של S (אפילו s בודד) ולאחר מכן מעדכן את הpolicy, כלומר, במקום לעדכן את וקטור הערכים במלואו על סמך וקטור קודם, מעכנים מצבים בודדים על סמך הוקטור הקודם או , אפילו ערך בודד על סמך שאר הערכים (ללא זיכרון זמני נוסף). באופן זה ניתן לרכז את מאמצי החישוב במצבים שסבירות ההגעה אליהם גבוהה.

[1] בהצגת הנוסחאות שלנו הדבר נבע מכך ש$V_{i,k}$ חושב לכל i ע"י בחירת u שמיזער את V על סמך $V_{:,k+1}$