מגישים: עומר ידגר 323040758 עידן ביבי 211375746

עבודה 2 – Deep Reinforcement Learning

הוראות הרצה – קבצי py

- בכל קובץ, יצרנו פעולות של run של המודל, לכן על מנת להריץ את הקוד יש להריץ את אחד הקבצים הנבחרים.
בנוסף, הפעולה run מחזירה 3 רשימות: רשימה של כל הEpisodes, Reward עבור כל Episode וערך הscore הממוצע עבור 100 Episodes רצופים.

חלק 1 – REINFORCE with value-function baseline

שאלה 1 - תשובה

הAdvantage מייצג את התגמול הנוסף שניתן להרוויח מביצוע פעולה במצב מסוים, ביחס לbaseline, ומוגדר כהפרש בין הreward המצטבר לValue-Function.

כדאי לעקוב אחרי הgradient אשר מחושב עם הadvantage מפני שבאלגוריתם REINFORCE ללא הadvantage, אנו מעדכנים את פרמטר המדיניות באמצעות עדכוני מונטה קרלו. זה גורם לgradients רועשים, שעלולים להוביל ללמידה לא יציבה ואיטית, ולהטיית חלוקת המדיניות לכיוון לא אופטימלי. אך, הפחתת הreward המצטבר מהbaseline גורמת לgradients קטנים יותר, ובכך לעדכונים קטנים ויציבים יותר.

שאלה 2 – תשובה

נרצה להוכיח את השוויון הבא:

$$E_{\pi_\theta}[\nabla log\pi_\theta(a_t|s_t)b(s_t)] = 0$$

מתוך **ההנחה כי פונקציית הbaseline $b(s_t)$ אינה תלויה בפעולה a_t**, נקבל:

$$E_{\pi_\theta}[\nabla log\pi_\theta(a_t|s_t)b(s_t)] = b(s_t) \cdot E_{\pi_\theta}[\nabla log\pi_\theta(a_t|s_t)]$$

לכן, נפתח את הצד השמאלי באמצעות חוקי תוחלות ונקבל:

$$b(s_t) \cdot E_{\pi_\theta}[\nabla log\pi_\theta(a_t|s_t)] = b(s_t) \cdot \int \nabla log\pi_\theta(a_t|s_t) \cdot \pi_\theta(a_t|s_t) d_{a_t|s_t}$$

כעת, נפתח על פי הזהות הנ"ל: $\nabla log\pi_\theta(a_t|s_t) = \frac{\nabla\pi_\theta(a_t|s_t)}{\pi_\theta(a_t|s_t)}$

$$b(s_t) \cdot \int \frac{\nabla\pi_\theta(a_t|s_t)}{\cancel{\pi_\theta(a_t|s_t)}} \cdot \cancel{\pi_\theta(a_t|s_t)} d_{a_t|s_t} = b(s_t) \cdot \int \nabla\pi_\theta(a_t|s_t) d_{a_t|s_t}$$

כעת, קיבלנו אינטגרל על פונקציית צפיפות של התפלגות מותנית, אשר מסתכמת ל1, ונקבל:

$$b(s_t) \cdot \int \nabla\pi_\theta(a_t|s_t) d_{a_t|s_t} = b(s_t) \cdot \nabla \int \pi_\theta(a_t|s_t) d_{a_t|s_t} = b \cdot \nabla 1 = 0$$

כלומר, קיבלנו ש:

$$\boldsymbol{E_{\pi_\theta}[\nabla log\pi_\theta(a_t|s_t)b(s_t)] = 0}$$

ניסויים – השוואה

REINFORCE:

Architecture – PolicyNetwork for regular **REINFORCE**

Layer	Units	Activation
Input	4 – Observation Space	-
Hidden	**12**	**Relu**
Output	1 – Action Space	Linear

REINFORCE with Baseline:

Architecture - PolicyNetwork for **REINFORCE with Baseline**

Layer	Units	Activation
Input	4 – Observation Space	-
Hidden	**12**	**Relu**
Output	1 – Action Space	Linear

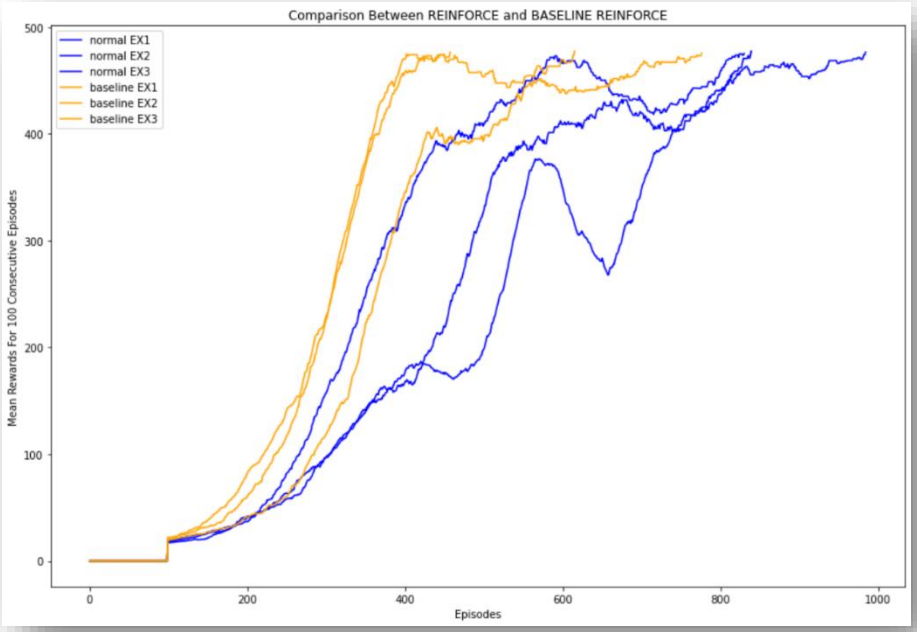
Architecture – StateValueNetwork for **REINFORCE with Baseline**

Layer	Units	Activation
Input	4 – Observation Space	-
Hidden	**8**	**Relu**
Hidden	**8**	**Relu**
Output	1 – Action Space	Linear

Final Hyper-parameters

- $discount\_factor = 0.99$
- $learning\_rate = 0.0004$
- $decay\_rate = 0.999$
- $max\_episodes = 5000$
- $\max\_steps = 501$

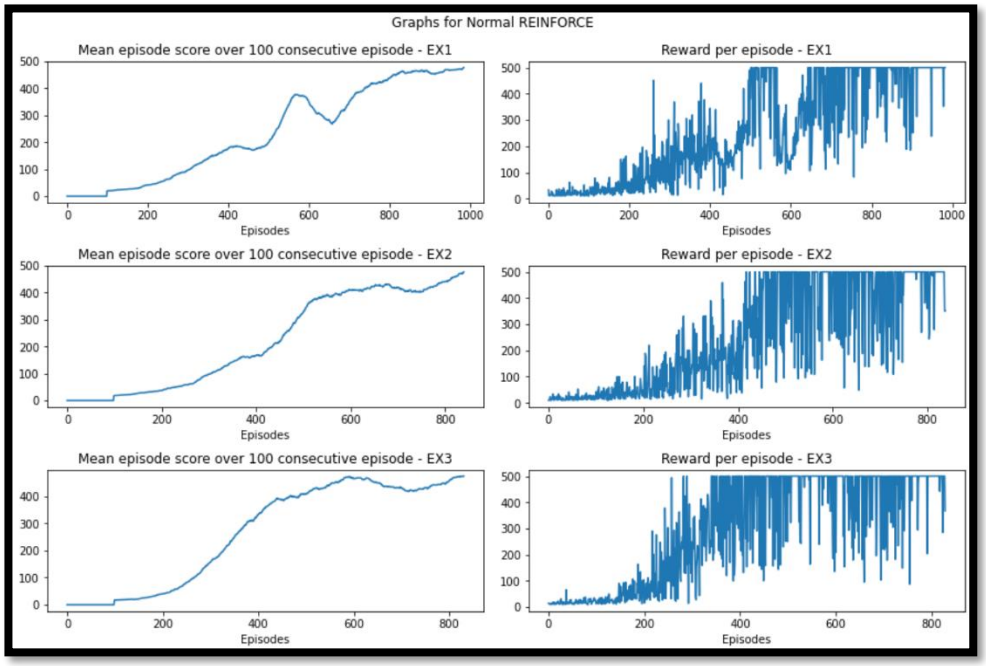
עבור הארכיטקטורה, הרצנו את המודל 3 פעמים בשביל להעריך את המודל עבור כמה הרצות שונות.



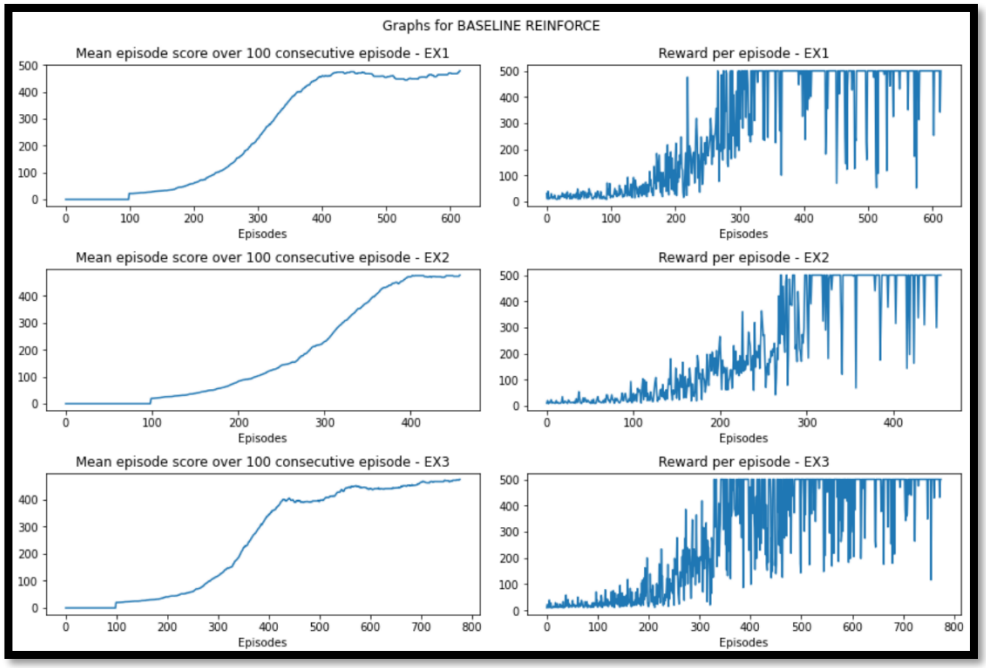
<u>התכנסות</u>: אצלנו, אפשר להגיד שהמודל התכנס אם הוא הגיע למצב שממוצע 100 הEpisodes האחרונים שלו הוא מעל 475.

לפי הגרף לעיל, אפשר לראות שהמודל עם ה BASELINE מתכנס יותר מהר מהמודל בלי ה BASELINE. עבור מודל BASELINE אפשר לראות הepisode הממוצע שלו להתכנסות הוא באזור ה600 לעומת המודל הרגיל שהוא מתכנס בממוצע אחרי 850.

בנוסף, נציג את הגרפים עבור ערכי הReward בכל Episode וערך הscore הממוצע עבור 100 Episodes רצופים עבור הרצה של אלגוריתם REINFORCE ו- Baseline REINFORCE:



ניתן לראות כי יש מגמת עליה אחרי בערך 200 episodes וגם שהמודל מגיע לפרס המקסימלי אחרי בערך 400 episodes.



ניתן לראות כי במודל של ה BASELINE ההתכנסות יותר מהירה והוא מגיע לפרס מקסימלי כבר באזור ה 300 episodes שזה 100 פחות מהמודל הרגיל.

בנוסף, ניתן לראות שבמודל ללא הBaseline יש הרבה קפיצות קיצוניות בערכי הReward לעומת המודל עם הBaseline אשר עם ערכים קבועים יותר יחסית.

חלק 2 – Advantage Actor-Critic

שאלה 1 – תשובה

TD error היא ההפרש בין התגמול העתידי הצפוי לאומדן הנוכחי של פונקציית הערך. זה ניתן על ידי המשוואה:

$$\delta_t = R_{t+1} + \gamma V(s_{t+1}) - V(s_t)$$

פונקציית היתרון היא אומדן של כמה טובה פעולה נתונה בהשוואה לפעולה הממוצעת במצב נתון. היא מוגדרת כשגיאת *TD* הצפויה עבור אותה פעולה, ניתנת על ידי:

$$A_{\pi\theta}(s,a) \;=\; Q_{\pi\theta}(s,a) \;-\; v_{\pi\theta}(s)$$

אפשר להגיע לפונקציית היתרון על ידי נטילת התוחלת לשגיאת *TD* על פני כל הפעולות האפשריות:

$$E_{\pi\theta}[\delta_{\pi\theta}|s,a] = \; E_{\pi\theta}[R_{t+1} \;+\; \gamma v_{\pi\theta}(S_{t+1})|s,a] - \; v_{\pi\theta}(s) =$$
$$= Q_{\pi\theta}(s,a) \;-\; v_{\pi\theta}(s) \;=\; A_{\pi\theta}(s,a)$$

לכן, השימוש בשגיאת *TD* של פונקציית הערך עבור עדכון פרמטרי רשת המדיניות זהה לשימוש הערכת היתרון.

שאלה 2 - תשובה

ה (π) policy-function היא פונקציית הActor, וה (v) value-function היא פונקציית הCritic.

- תפקידו של הActor הוא ללמוד את הPolicy שתמקסם את התגמול הצפוי על ידי בחירת הפעולה הטובה ביותר בכל מדינה.
- תפקידו של הCritic, לעומת זאת, הוא ללמוד את הvalue-function, שמעריכה את התגמול העתידי הצפוי עבור מצב נתון.

ניסויים – השוואה

- הארכיטקטורה עבור האלגוריתמים REINFORCE ו- REINFORCE with Baseline מוגדרים כפי שהצגנו בחלק 1.

Actor-Critic:

Architecture - PolicyNetwork for **Actor-Critic**

Layer	Units	Activation
Input	4 – Observation Space	-
Hidden	**12**	**Relu**
Output	1 – Action Space	Linear

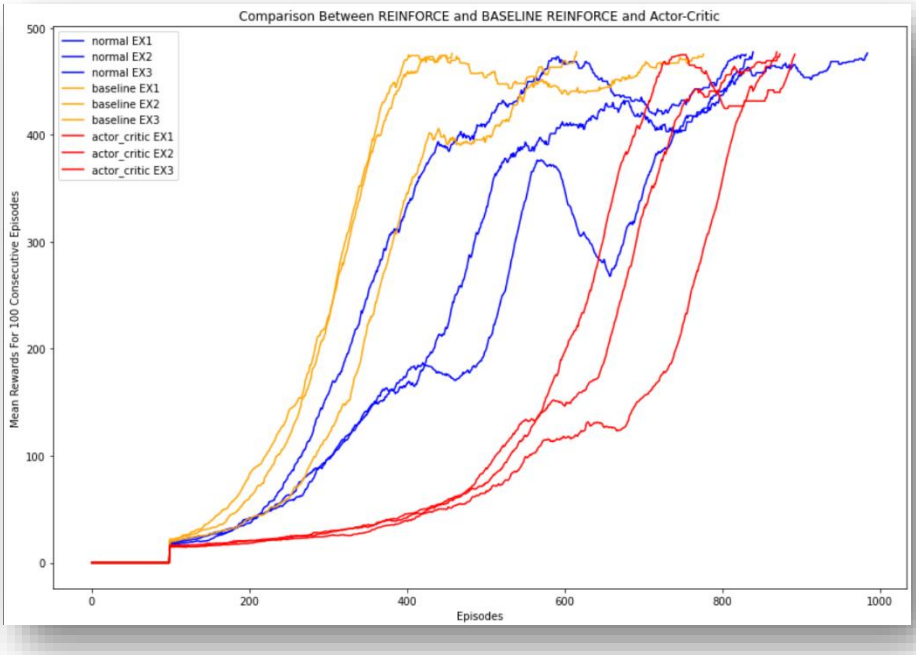
Architecture – StateValueNetwork for **Actor-Critic**

Layer	Units	Activation
Input	4 – Observation Space	-
Hidden	**64**	**Relu**
Hidden	**32**	**Relu**
Output	1 – Action Space	Linear

Final Hyper-parameters

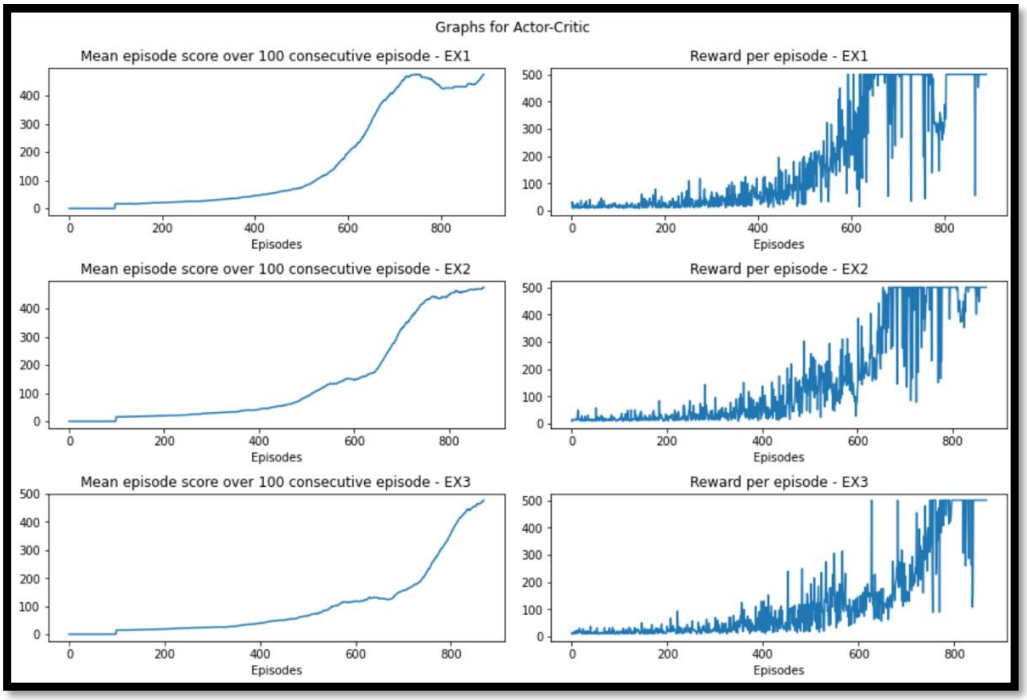
- $discount\_factor = 0.99$
- $learning\_rate\ PolicyNetwork = 0.0001$
- $learning\_rate\ StateValueNetwork = 0.0005$
- $decay\_rate = 0.999$
- $max\_episodes = 5000$
- $\max\_steps = 501$

עבור הארכיטקטורה, הרצנו את המודל 3 פעמים בשביל להעריך את המודל עבור כמה הרצות שונות.



אחרי שהרצנו מספר ניסויים ושיחקנו עם פרמטרים ראינו שהביצועים של מודל *actor-critic* הם פחות טובים מהביצועים של BASELINE REINFORCE.
באופן ממוצע הוא מתכנס לאחר 800 episodes בדומה לBASE REINFORCE שמתכנס לאחר 850. אבל המגמת עלייה שלו מתבצעת מאוחר יותר באזור ה 600 episodes.

נציג את הגרפים עבור ערכי הReward בכל Episode וערך הscore הממוצע עבור 100 Episodes רצופים עבור הרצה של אלגוריתם Actor-Critic:



ניתן לראות כי המודל מגיע לפרס מקסימלי לראשונה בממוצע באזור ה 750 episodes שזה כמעט פי 2 ממודל ה BASELINE REINFORCE.