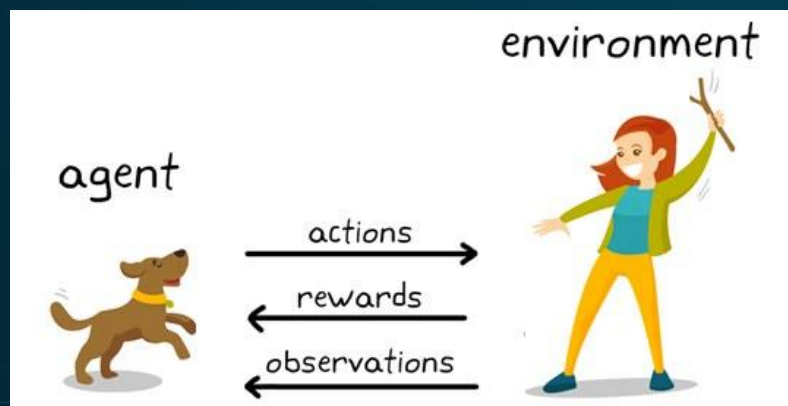




קריית החינוך
פארק המדע
בית לערכים
למצוינות ולחדשנות

מבוא ללמידת חיזוק



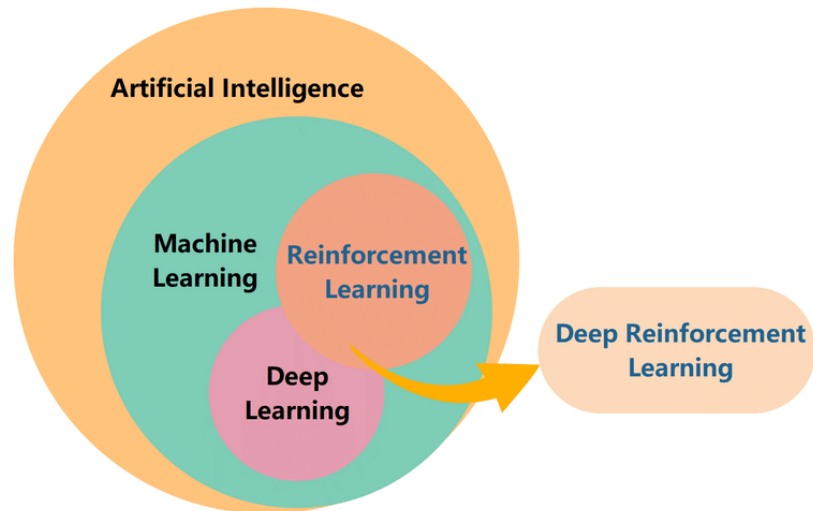
גלעד מרקמן



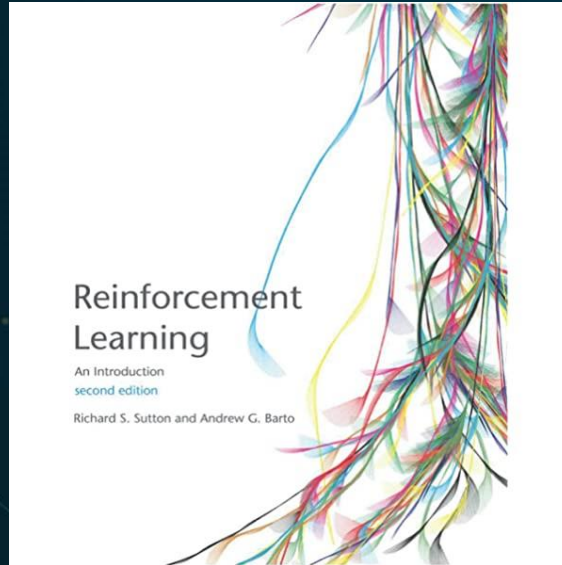
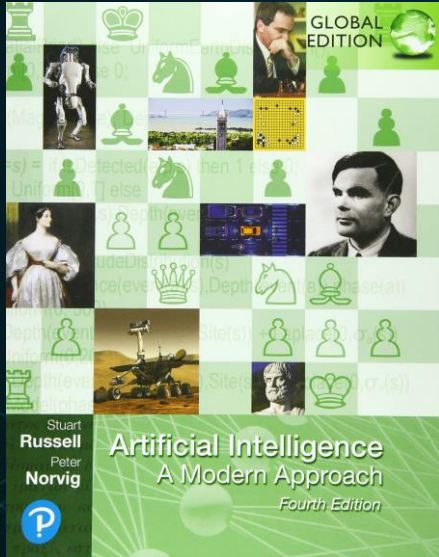
קריית החינוך
פארק המדע
בית לערכים
למצוינות ולחדשנות

בינה מלאכותית ולמידת מכונה

- תחום הבינה המלאכותית הוא תחום רחב מאוד, הכולל בתוכו את למידת המכונה.
- תחום למידת מכונה מתחלק לשלושה סוגים:
 - למידה מונחית (supervised learning)
 - למידה בלתי מונחית (unsupervised learning)
 - למידת חיזוק (reinforcement learning)
- Deep Learning – שימוש ברשת נוירונים בכל אחד מהסוגים לעיל.



ביבליוגרפיה



• R. Sutton, A. Barto, Reinforcement Learning, 2nd ed.

• אתר של פרופ' דיויד סילבר הכולל סדרת הרצאות בנושא למידת מכונה.

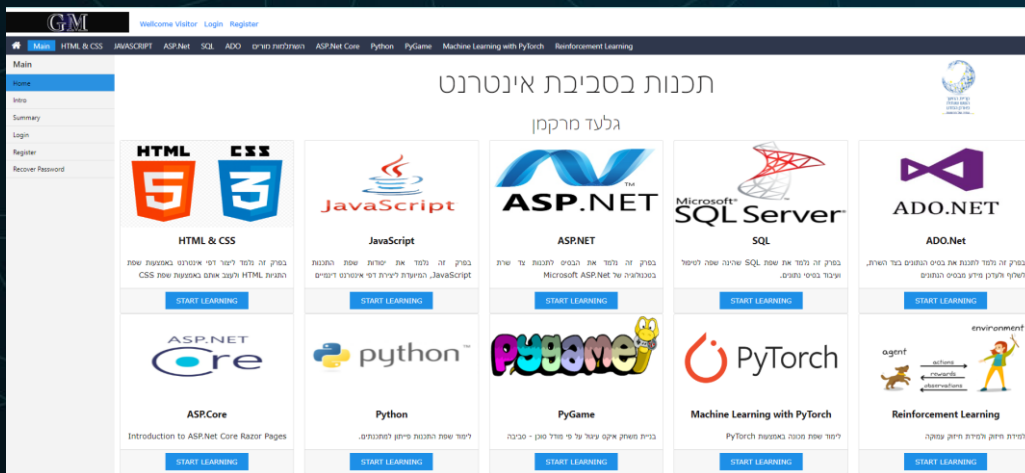
• <https://www.davidsilver.uk/>

• S. Russel, P. Norvig, Artificial Intelligence - A Modern Approach, 4th ed.

• האתר תכנות באינטרנט – גלעד מרקמן

• <https://webprogramming.azurewebsites.net/>

• הכולל סרטוני הרצאות, מצגות ודוגמאות קוד.



מודל סביבה סוכן

- הצגנו בשיעורים הקודמים מודל לבניית מערכת לפתרון בעיות באמצעות בינה מלאכותית הנקרא מודל סביבה - סוכן.
- המודל מאפשר לנו לתאר את העולם שסביבנו, ואת היחסים בין מרכיבי העולם לסביבה.
- המודל מתאים גם לבניית משחקים בהם המחשב מהווה את אחד המשתתפים במשחק.
- מודל זה מהווה בסיס למודל מתמטי לפתרון בעיות הנקרא MDP-Markov Decision Process, שאותו נלמד היום.

MDP – Markov Decision Process

• סביבה – Environment

• מתאר את העולם בו אנו נמצאים.

• סוכן – Agent

• הרובוט / התוכנה הפועלת בסביבה.

• הסוכן קולט את הסביבה ומבצע פעולות המשנות אותה.

• מצב – state

• תיאור של מצב רגעי (הקפאה) של הסביבה.

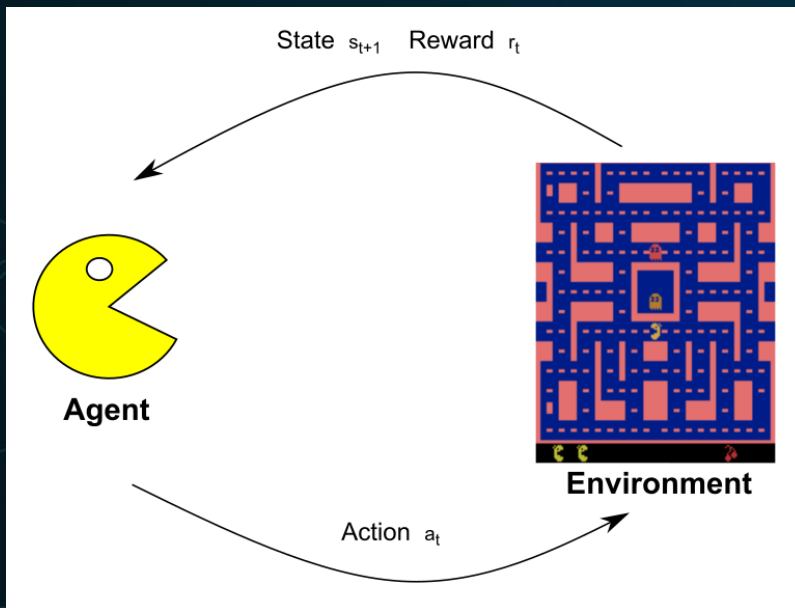
• פעולה – action

• פעולות אותן יכול לבצע הסוכן (צעדי המשחק).

• תגמול – Reward

• תגמול מספרי שהסוכן מקבל בעת ביצוע פעולה.

• התגמול יכול להיות חיובי או שלילי או אפס.



תגמול - Reward

• תגמול (Reward)

- פונקציית התגמול מקבלת מצב של הסוכן, הפעולה שביצע, והמצב אליו הגיע הסוכן ומחזירה ערך מספרי.
 $R(s,a,s') \rightarrow \text{reward}$
- התגמול יכול להיות חיובי, שלילי או אפס.

• דוגמאות לתגמול:

- במשחק שחמט: 0 על כל צעד, 1 על נצחון, -1 על הפסד, 0 על תיקו.
- במשחק מחשב (פקמן): התגמול כמספר הנקודות שמקבל פקמן מאכילת המזון / רוחות; 100- במקרה שנאכל על ידי רוח.
- סוכן המטיס רחפן: 1 על כל שניה שהרחפן באויר; -50 על התרסקות.
- סוכן המשקיע בבורסה: התגמול בהתאם לרווחים ההפסדים היומיים.

מטרה ומדיניות

• המטרה (Goal)

• המטרה של הסוכן למקסם את התגמול הכולל לאורך זמן.

• $G = R_0 + R_1 + R_2 + \dots + R_T$

• במודל שלנו אנו מניחים כי ניתן להגדיר כל מטרה כמקסום (maximization) של התגמול העתידי של הסוכן.

• מדיניות (Policy)

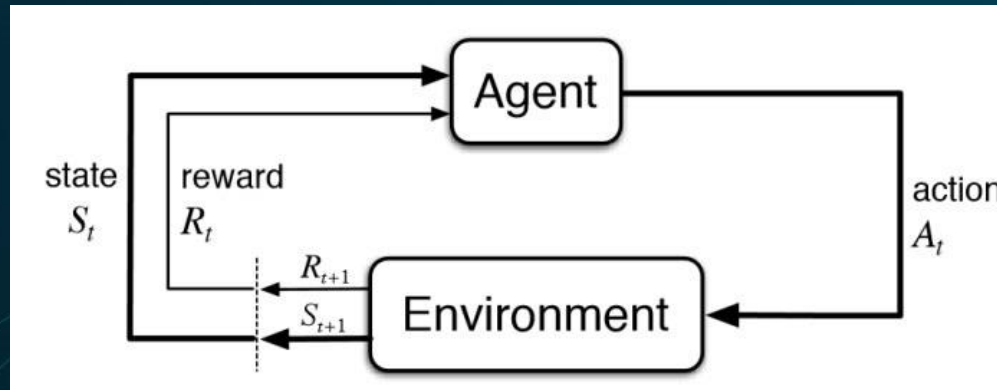
• זוהי פונקציה המקבלת מצב ומחזירה את הפעולה בה נוקט הסוכן (נפרט בהמשך).

• $P(s) \rightarrow action$

תכונת מרקוב ושרשרת מרקוב

• שרשרת מרקוב

- ניתן לייצג את פעילות הסוכן בסביבה כשרשרת מרקוב הבאה, הנקבעת לפי המדיניות:
- $S_0, A_0, R_0, S_1, A_1, R_1 \dots, S_T$



• תכונת מרקוב

- הערך של מצב מסויים תלוי אך ורק במצב בו נמצאים, ואין חשיבות למצבים בהם ביקרנו בעבר (אין חשיבות להסטוריה).
- אם למשל אנחנו נמצאים במצב מסוים במשחק. הערך של מצב זה אינו תלוי כיצד הגענו למצב זה. אין חשיבות אם, לדוגמה, הגענו למצב S באמצעות פעולה $A1$ ממצב $S1$ או באמצעות פעולה $A2$ ממצב $S2$.

מדיניות

• מדיניות (Policy)

- זוהי פונקציה המקבלת מצב ומחזירה את הפעולה בה נוקט הסוכן.

$$P(s) \rightarrow action \quad | \quad \Pi(s) \rightarrow action$$

- המדיניות קובעת את שרשרת מרקוב של הסוכן. היא קובעת אילו פעולות הוא יעשה בכל אחד מהצעדים.

- $S_0, A_0, R_0, S_1, A_1, R_1 \dots, S_T$

- המדיניות לא חייבת להיות אופטימלית. למשל אם במבוך אנחנו תמיד הולכים ימינה זו מדיניות אך היא אינה אופטימלית.

• מדיניות אופטימלית Π^* | P^*

- היא המדיניות אשר בכל מצב אם נפעל על פיה נקבל את סכום התגמולים העתידיים המקסימלי.

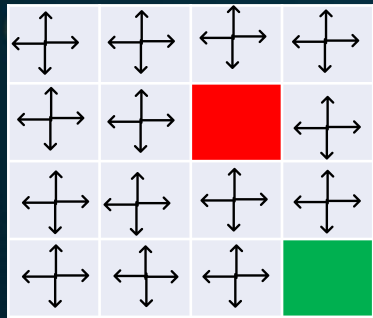
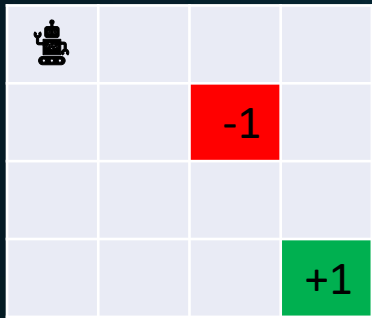
הדגמה של תהליך מרקוב

- נתון לנו לוח של 4×4 . הסוכן מתחיל בקובייה השמאלית עליונה
- המטרה - להגיע לקובייה הימנית.
- Reward - התגמול בהגעה לקובייה הימנית הוא 1, לקובייה האדומה הוא -1. התגמול לכל צעד אחר הוא 0.
- Action – up, down, left, right.
- Policy - המדיניות (policy) של הסוכן היא ללכת תמיד למעלה אלא אם הוא רואה (צמוד) לקובייה הימנית ואז הולך לכיוונה.
- State – מיקום הסוכן (x, y) .

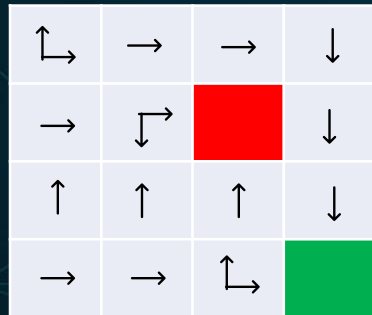
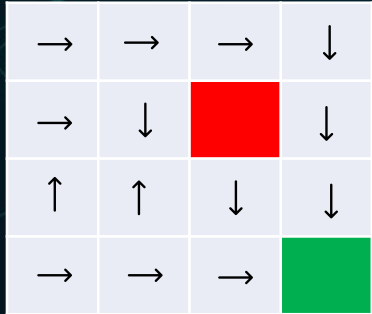
Policy			
↑	↑	↑	↑
↑	↑		↑
↑	↑	↑	↓
↑	↑	→	

♟			
		-1	
			+1

הדגמה של תהליך מרקוב



Policy



• דוגמאות לפונקציית מדיניות נוספת:

- הסוכן תמיד הולך למטה.
- הסוכן בוחר צעד רנדומלי.
- הפונקציה יכולה לקבוע פעולה לכל מצב בהתבסס על טבלה, אך זה אינו הכרחי (ולעיתים לא אפשרי).

- בלמידת חיזוק אנחנו מחפשים את הפוליסי המיטבית P^* , אשר בכל מצב נותנת את הפעולה שתזכה אותנו בתגמול העתידי המצטבר הטוב ביותר.

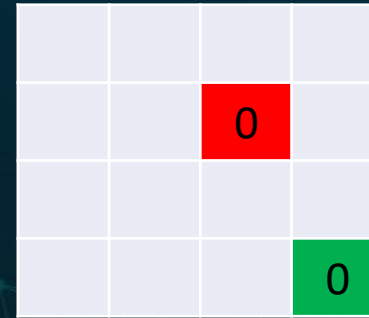
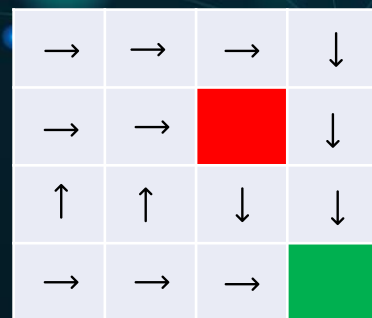
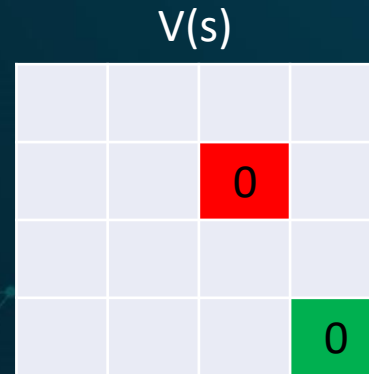
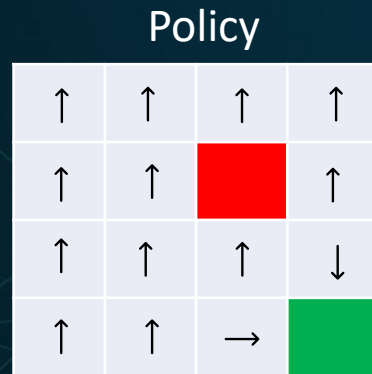
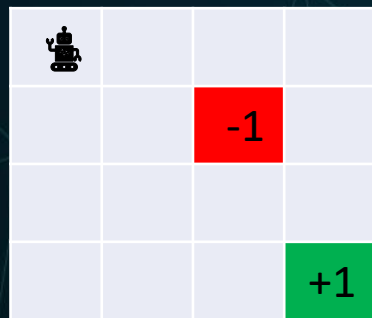
פונקציית הערך

• פונקציית הערך:

• פונקציה המקבלת מצב ומחזירה את הסכום המצטבר של התגמול אם הסוכן יפעל על פי המדיניות שנקבעה.

$$V_p(s) = value = R_0 + R_1 + \dots + R_T$$

• פונקציית הערך תלויה במדיניות (פוליסי). אם המדיניות תשתנה גם פונקציית הערך תשתנה.



פונקציית ערך פעולה

• פונקציית ערך פעולה (מצב-פעולה)

- $Q_P(s, a) = value$.
- פונקציה המקבלת מצב ופעולה ומחזירה את הסכום המצטבר של התגמול אם הסוכן יפעל על פי המדיניות שנקבעה.
- הערך הוא סכום מצטבר של התגמול: $value = R_0 + R_1 + R_2 + \dots + R_T$
- התגמולים נקבעים בהתאם לשרשרת הפעולות: $S_0, A_0, R_0, S_1, A_1, R_1 \dots, S_T$
- שרשרת הפעולה נקבעת בהתאם למדיניות (הפוליסי).
- בפונקציית ערך הפעולה – הפעולה הראשונה מועברת לפונקציה, ולאחר מכן כל פעולה נוספת מבוצעת בהתאם לפוליסי.

תגמול עתידי – פקטור γ

- בעת חישוב התגמול העתידי אנחנו יוצאים מתוך הנחה שכל שתגמול מתקבל בעתיד הרחוק יותר הוא שווה פחות. תגמול מידי עדיף על תגמול מאוחר.
- לכן, אנחנו נעדכן את נוסחת התגמול ונוסיף פקטור להפחתת תגמול עתידי.
- $G = R_0 + \gamma R_1 + \gamma^2 R_2 + \gamma^3 R_3 + \dots + \gamma^n R_n$


- פקטור $\gamma = [0,1]$, ובדרך כלל משתמשים במספר קרוב ל 1, כגון 0.90 או 0.99.

• מדוע משתמשים בפקטור?

- תואם את ההתנהגות האנושית לפיה תמורה מיידית עדיפה על תמורה בעתיד.
- נוח מבחינת חישובים מתמטיים.
- מונע לולאות אין סופיות – שכן בסוף שואף ל-0.

תגמול עתידי γ

- נחשב את פונקציית הערך בדוגמאות הבאות שהבאנו מקודם כאשר $\gamma = 0.9$.
- $G = R_0 + \gamma R_1 + \gamma^2 R_2 + \gamma^3 R_3 + \dots \gamma^n R_n$

			
		-1	
			+1

Policy			
→	→	→	↓
→	↓	-1	↓
↓	↓	→	↓
→	→	→	+1

V(s)			
		0	
			0

→	→	→	↓
→	→	-1	↓
↑	↑	↓	↓
→	→	→	+1

		0	
			0

0.590	0.66	0.729	0.81
0.66	0.729	0	0.9
0.729	0.81	0.9	1
0.81	0.9	1	0

0.590	0.66	0.729	0.81
-0.9	-1	0	0.9
-0.81	-0.9	0.9	1
0.81	0.9	1	0

מודל Model

- **מודל (Model) – צופה מה הסביבה תעשה בעקבות פעולה של הסוכן.**
- State - המודל צופה מה יהיה המצב הבא.
- Reward - המודל צופה מה יהיה התגמול המידי שיקבל הסוכן בעקבות הפעולה.

• מודל ידוע / מודל לא ידוע

- בחלק מהמקרים המודל ידוע – למשל במשחק פאזל המספרים / קובייה הונגרית
- אנחנו יודעים מה יהיה המצב הבא בעקבות פעולה שעשינו.
- במקרים אחרים המודל לא ידוע לנו.
- במשחק כנגד יריב המודל לא ידוע – בעקבות פעולה שלנו לא ידוע לנו מה יעשה היריב ומה יהיה המצב הבא (שאנחנו נקבל כדי להחליט איזו פעולה לנקוט).
- במשחקים הסתברותיים (עם קוביות/קלפים) – לא ידוע לנו מה יהיה המצב הבא שכן הוא תלוי בתוצאת הקוביות.

מודל אקראי – מודל דטרמיניסטי

- מודל MDP אקראי (סטוכסטי) - הינו מודל בו הסוכן לא יודע בוודאות לאיזה מצב יגיע בעקבות פעולה מסויימת.
- נהיגה על קרח – פעולה של פניה ימינה יכולה להוביל בחלק מהמקרים להחלקה קדימה.
- משחקי מזל (קוביות) – תוצאת פעולה הינה אקראית ותלויה במזל (בתוצאות הטלת הקוביות).
- משחקי קלפים – תוצאות פעולה כמו תן קלף נוסף אינה חד משמעית והינה אקראית / הסתברותית.
- מודל דטרמיניסטי (לא אקראי) – הינו מודל בו ידוע לנו מראש מה תהיה התוצאה בביצוע פעולה מסויימת.
- מודל MDP כולל גם התחשבות באקראיות (הסתברות). אנחנו נפשט את המודל ונלמד בהתחלה את המודל הדטרמיניסטי.

מטרת למידת חיזוק

• המטרה שלנו לפתח אלגוריתם למציאת המדיניות המיטבית:

- $P^*(s) \rightarrow action$

- למדיניות מיטבית יש גם פונקציית ערך תואמת. נכונה אותה פונקציית ערך מיטבית:

- $V^*(s) \rightarrow value$

- $Q^*(s, a) \rightarrow value$

- תחום למידת החיזוק פיתח את האלגוריתמים למציאת המדיניות המיטבית, הן כאשר המודל ידוע והן כאשר המודל לא ידוע.

שעורי בית

• ב Grid World שהצגנו:

- הצע מדיניות (פוליסי) מיטבית נוספת על המדיניות שהצענו בשיעור.
- חשב עבורה את פונקציית הערך כאשר $\gamma = 0.9$.
- מדוע $v(s_{end_state}) = 0$?
- הצע מדיניות מיטבית כאשר $\gamma = 1$ וחשב את פונקציית הערך המתאימה.
- כיצד משתנה התנהגות הסוכן (על פי מדיניות מיטבית) בין מקרה בו הפקטור γ שווה ל-1 לבין מקרה בו הפקטור קטן מ-1, מבחינת מספר הצעדים עד למטרה.

