תרגיל בית 3 – MDP ומבוא ללמידה

רון דהן 208835637
רנן קנטור 314998931

מומלץ לחזור על שקפי ההרצאות והתרגולים הרלוונטיים לפני תחילת העבודה על התרגיל.

חלק א' – MDP (30 נק')

רקע

בחלק זה נעסוק בתהליכי החלטה מרקובים, נתעניין בתהליך עם **אופק אינסופי** (מדיניות סטציונרית) ובתועלת המחושבת בעזרת-Discounted rewards.

חלק א' - חלק היבש ✍

1. בתרגול ראינו את משוואת בלמן כאשר התגמול ניתן עבור המצב הנוכחי בלבד, כלומר $R: S \to \mathbb{R}$, למתן תגמול זה נקרא "תגמול על הצמתים" מכיוון שהוא תלוי בצומת שהסוכן נמצא בו.

   בהתאם להגדרה זו הצגנו בתרגול את האלגוריתמים Value iteration ו-Policy Iteration למציאת המדיניות האופטימלית.

   כעת, נרחיב את ההגדרה הזו, לתגמול המקבל את המצב הנוכחי, הפעולה לביצוע והמצב הבא שהסוכן הגיע אליו בפועל (בין אם הסוכן בחר לצעוד לכיוון הזה ובין אם לא), כלומר: $R: S \times A \times S' \to \mathbb{R}$, למתן תגמול זה נקרא "תגמול על הקשתות".

   א. (יבש 1 נק') התאימו את הנוסחה של התוחלת של התועלת מהתרגול, עבור התוחלת של התועלת המתקבלת במקרה של "תגמול על הקשתות"?

   פתרון:

$$E_\pi[\sum_{t=0}^{\infty} \gamma^t R(S_t, \pi(S_t), S_{t+1}) | S_0 = s]$$

   ב. (יבש 1 נק') כתבו מחדש את נוסחת משוואת בלמן עבור המקרה של "תגמול על הקשתות".

   פתרון:

$$U(s) = \underbrace{max}_{a \in A(s)} \sum_{S'} [P(S'|a,s) \cdot R(s,a,S') + \gamma P(S'|a,s) \cdot U(S')]$$

   ג. (יבש 2 נק') נסחו את אלגוריתם Value Iteration עבור המקרה של "תגמול על הקשתות".

<u>פתרון:</u>

```
while True:
   U = U'
   delta = 0
   for s in states:
      U'[s] = max_action(Sum_on_all_next_states(gamma * P(s'|a, s) * U(s') + P(s'|a, s) + R(s, a, s')))
      if abs(U'[s] - U[s]) > delta:
         delta = abs(U'[s] - U[s])
   if gamma == 1:
      if delta == 0
         return U
   else:
      if delta < epsilon * (1 - gamma) / gamma:
         return U
```

ד. (יבש 2 נק') נסחו את אלגוריתם Policy Iteration עבור המקרה של "תגמול על הקשתות".

<u>פתרון:</u>

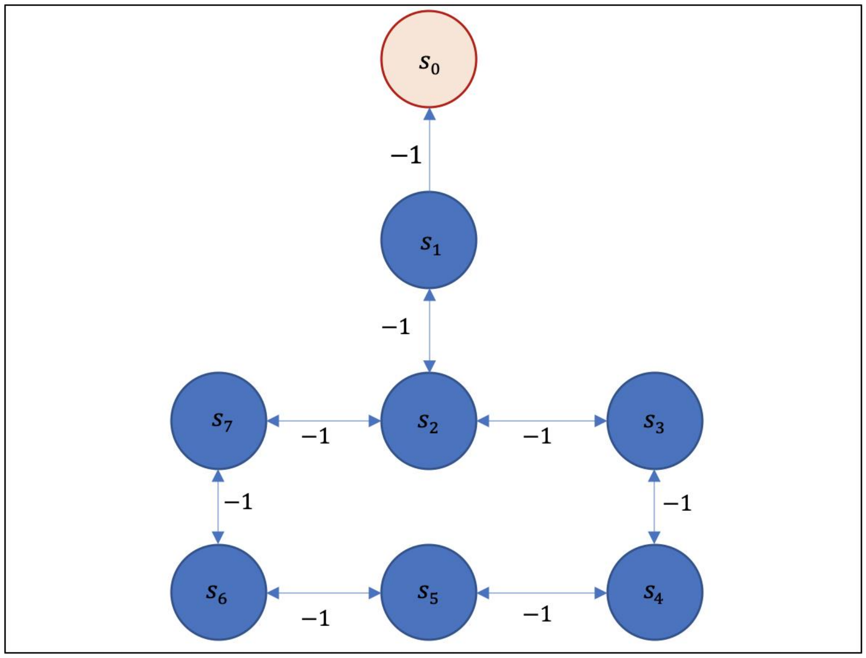
function POLICY-ITERATION(mdp) **returns** a policy
  inputs: mdp, an MDP with states S, actions $A(s)$, transition model $P(s' \mid s, a)$
  local variables: U, a vector of utilities for states in S, initially zero
    π, a policy vector indexed by state, initially random

  repeat
    $U \leftarrow$ POLICY-EVALUATION(π, U, mdp)
    $unchanged? \leftarrow$ true
    for each state s **in** S **do**
      if $\max_{a \in A(s)} \sum_{s'}$ P(s'|a,s)*(R(s,a,s')+gamma * U(s') > $\sum_{s'}$ P(s'|pi(s),s)*(R(s,pi(s),s') + gamma * U(s')
        $\pi[s] \leftarrow \operatorname{argmax}_{a \in A(s)} \sum_{s'} P(s' \mid s, a)\ U[s']$
        $unchanged? \leftarrow$ false
  until $unchanged?$
  return π

- לגבי סעיפים 3 ו 4, אם גאמה שווה אחד לא מובטח לנו התכנסות אם אין מצב סופי. לכן יש להבטיח מצב סופי ושכל מדיניות תסתיים בו כדי לקבל אופטימליות.

הערה: בסעיפים 3 ו-4 התייחסו גם למקרה בו $\gamma = 1$, והסבירו מה לדעתכם התנאים שצריכים להתקיים על הסביבה\mdp על מנת שתמיד נצליח למצוא את המדיניות האופטימלית.

נתון הגרף הבא:



נתונים:

- $\gamma = 1$ (Discount factor).
- אופק אינסופי.
- $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$ – קבוצת המצבים – מתארים את מיקום הסוכן בגרף.
- $S_G = \{s_0\}$ – קבוצת המצבים הסופיים.
- קבוצת הפעולות לכל מצב (על פי הגרף), לדוגמא: $A(s_2) = \{\uparrow, \rightarrow, \leftarrow\}$.
- תגמולים ("תגמול על הקשתות"):

$$\forall s \in S \backslash S_G, a \in A(s), s' \in S: \quad R(s, a, s') = -1$$

- מודל המעבר הוא דטרמיניסטי, כלומר כל פעולה מצליחה בהסתברות אחת.

ה. (יבש 4 נק') הרץ את האלגוריתם Value iteration שכתבת על הגרף הנתון. ומלא את הערכים בטבלה הבאה, כאשר $\forall s \in S: U_0(s) = 0$. (ייתכן שלא צריך למלא את כולה).

	$U_0(s_i)$	$U_1(s_i)$	$U_2(s_i)$	$U_3(s_i)$	$U_4(s_i)$	$U_5(s_i)$	$U_6(s_i)$	$U_7(s_i)$	$U_8(s_i)$
s_1	0	-1	-1	-1	-1	-1	-1	-1	**-1**
s_2	0	-1	-2	-2	-2	-2	-2	-2	**-2**
s_3	0	-1	-2	-3	-3	-3	-3	-3	**-3**
s_4	0	-1	-2	-3	-4	-4	-4	-4	**-4**
s_5	0	-1	-2	-3	-4	-5	-5	-5	**-5**
s_6	0	-1	-2	-3	-4	-4	-4	-4	**-4**
s_7	0	-1	-2	-3	-3	-3	-3	-3	**-3**

ו. (יבש 4 נק') הרץ את האלגוריתם Policy iteration שכתבת על הגרף הנתון. ומלא את הערכים בטבלה הבאה, כאשר המדיניות ההתחלתית π_0 מופיעה בעמודה הראשונה בטבלה. (ייתכן שלא צריך למלא את כולה).

	$\pi_0(s_i)$	$\pi_1(s_i)$	$\pi_2(s_i)$	$\pi_3(s_i)$	$\pi_4(s_i)$	$\pi_5(s_i)$	$\pi_6(s_i)$	$\pi_7(s_i)$	$\pi_8(s_i)$
s_1	↑	↑	↑	↑	↑	↑	↑	↑	↑
s_2	↑	↑	↑	↑	↑	↑	↑	↑	↑
s_3	←	←	←	←	←	←	←	←	←
s_4	↑	↑	↑	↑	↑	↑	↑	↑	↑
s_5	→	→	→	→	→	→	→	→	→
s_6	→	→	↑	↑	↑	↑	↑	↑	↑
s_7	↓	→	→	→	→	→	→	→	→

חלק ב׳ - מבוא ללמידה (70 נק׳)

✍️ חלק ב׳ – חלק היבש (28 נק׳)

1. (8 נק׳) השתמשו בdataset הנתון על מנת ללמוד מסווג (עץ) אשר יחזה האם התלמדים יעברו את הקורס באלגברה א׳

Average	Studied	Passed
Low	No	No
Low	Yes	Yes
Medium	No	No
Medium	Yes	Yes
High	No	Yes
High	Yes	Yes

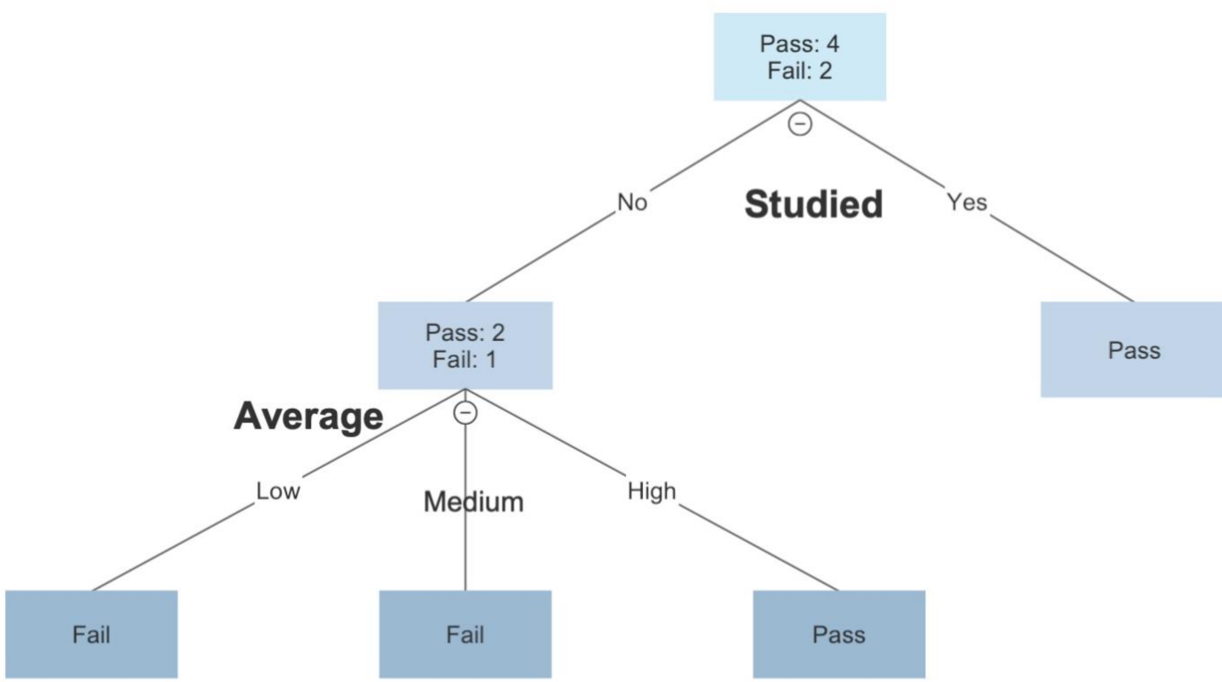
הערה, ניתן לרשום את התשובות בעזרת $\log_2$

א. מה האנטרופיה H(Passed)? 0.9183

ב. מה האנטרופיה H(Passed | Average)? 0.666

ג. מה האנטרופיה H(Passed | Studied) ? 0.459

ד. צייר את עץ ההחלטה הנלמד עבור הdataset הנתון



.2 (12 נק') נגדיר דאטה סט $D = \{(x_1, y_1), \dots, (x_n, y_n)\}$ שבו n דוגמאות מתויגות עם סיווג בינארי $y_i \in \{0,1\}$. כל דוגמה היא וקטור תכונות המורכב משתי תכונות רציפות $x_i = (v_{1,i}, v_{2,i})$.

הניחו כי קיים מסווג מטרה $f(x): R^2 \to \{0,1\}$ שאותו אנו מעוניינים ללמוד (הוא אינו ידוע לנו) וכן שהדוגמאות ב-D עקביות עם מסווג המטרה (כלומר שאין דוגמאות רועשות ב- D).

בסעיפים הבאים, עבור KNN, הניחו פונק' מרחק אוקלידי.

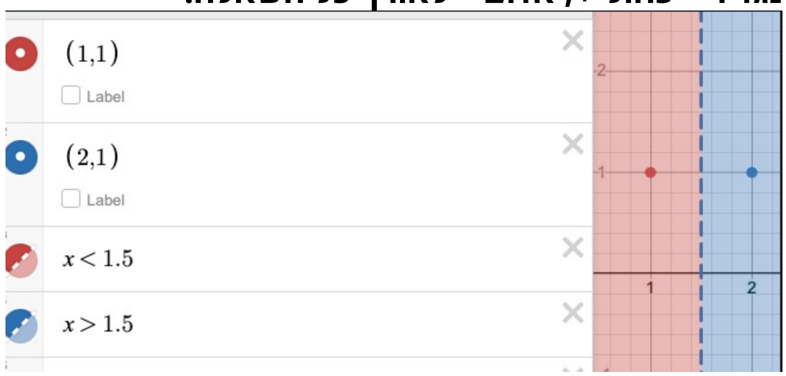
כמו כן, הניחו שאם קיימות נקודות במרחב כך שעבורן יש מספר דוגמאות במרחק זהה, קודם מתחשבים בדוגמאות עם ערך v_1 מקסימלי ובמקרה של שוויון בערך של v_1, מתחשבים קודם בדוגמאות עם ערך v_2 מקסימלי. הניחו כי אין דוגמאות זהות לחלוטין (כלומר גם עם ערך v_1 זהה וגם עם ערך v_2 זהה).

בכל סעיף, **הציגו מקרה המקיים את התנאים המוצגים בסעיף, הסבירו במילים, וצרפו תיאור גרפי (ציור) המתאר את המקרה (הכולל לפחות תיאור מסווג המטרה והדוגמאות שבחרתם)** . סמנו דוגמאות חיוביות בסימן '+' (פלוס) ודוגמאות שליליות בסימן '-' (מינוס). בכל אחת מתתי הסעיפים הבאים אסור להציג מסווג מטרה טריוויאלי, דהיינו שמסווג כל הדוגמאות כחיוביים או כל הדוגמאות כשליליים.

[3 שורות לכל סעיף, אין הגבלה על הגרפים, מלל ופתרון שאינו מוגדר היטב כמתבקש לא יקבל ניקוד]

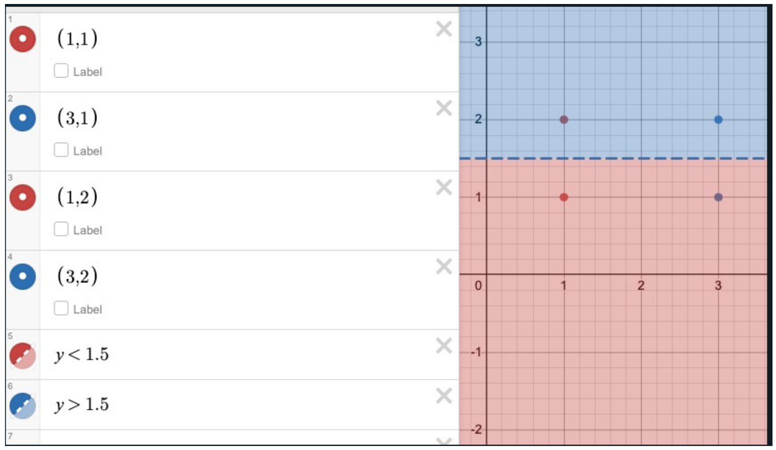
א. (3 נק') הציגו מסווג מטרה $f(x): R^2 \to \{0,1\}$ וקבוצת אימון בעלת לכל היותר 10 דוגמאות כך שלמידת עץ ID3 תניב מסווג אשר עונה נכון עבור כל דוגמת מבחן אפשרית (כלומר יתקבל מסווג המטרה), אך למידת KNN תניב מסווג שעבורו קיימת לפחות דוגמת מבחן אחת עליה הוא יטעה, לכל ערך K שייבחר.

נגדיר – כחול +, אדום – לאורך כל השאלה.



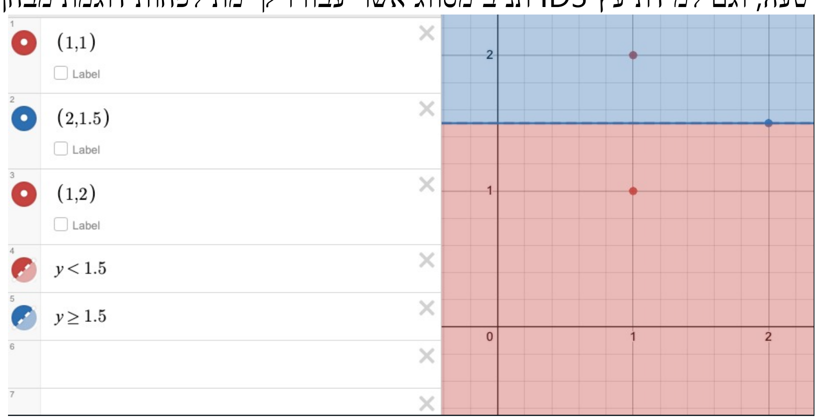
ברור כי לכל K, המסווג יטעה כי עבור K=1, לפי הפיאצה, הדוגמא לא רואה את עצמה ולכן תמיד יבחר בדוגמא השניה שזהו הסיווג ההפוך. בנוסף ה K מוגדרים, לפי הפיאצה, רק על מספרים אי זוגיים לכן אין עוד K שעליו מוגדר ה KNN (כי K לא יכול להיות יותר גדול ממספר הדוגמאות שלנו). בלמידת העץ 3ID, נגדיר שאם v1 גדול מ1.5 אתה כחול, ואדום אחרת (ובבירור, 3ID כזה מסווג נכון את קבוצת המבחן).

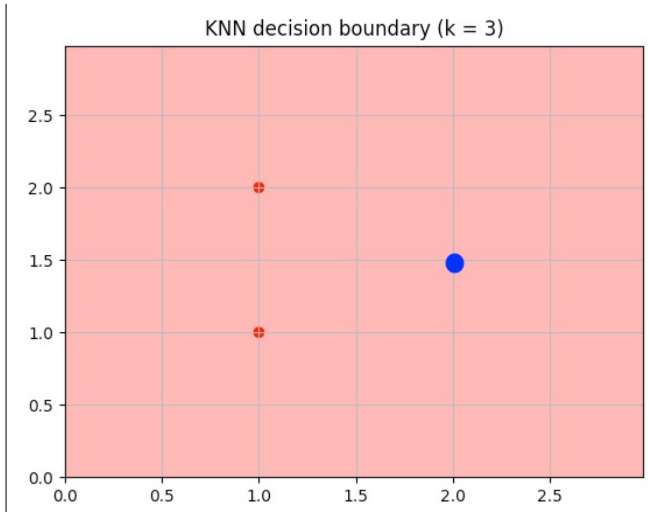
ב. (3 נק') הציגו מסווג מטרה $f(x): R^2 \rightarrow \{0,1\}$ וקבוצת אימון בעלת לכל היותר 10 דוגמאות כך שלמידת מסווג KNN עבור ערך K מסוים תניב מסווג אשר עונה נכון עבור כל דוגמת מבחן אפשרית (כלומר יתקבל מסווג המטרה), אך למידת עץ ID3 תניב מסווג אשר עבורו קיימת לפחות דוגמת מבחן אפשרית אחת עליה הוא יטעה.



עבור K=1, תמיד נבחר את אותו סיווג של הדוגמה (נשים לב כי המרחק בX גדול ממש מהמרחק בY). נגדיר בלמידת העץ שכל דוגמה עם v2 גדול מ 1.5 היא אדומה. על כן, על נקודה 3,2 המסווג יטעה (כי יגיד שאדומה למרות שכחולה).

ג. (3 נק') הציגו מסווג מטרה $f(x): R^2 \rightarrow \{0,1\}$ וקבוצת אימון בעלת לכל היותר 10 דוגמאות כך שלמידת מסווג KNN עבור ערך K מסוים תניב מסווג אשר עבורו קיימת לפחות דוגמת מבחן אפשרית אחת עליה הוא יטעה, וגם למידת עץ ID3 תניב מסווג אשר עבורו קיימת לפחות דוגמת מבחן אחת אפשרית עליה הוא יטעה.

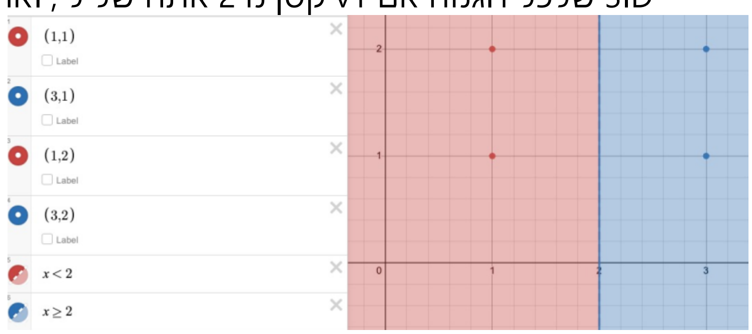




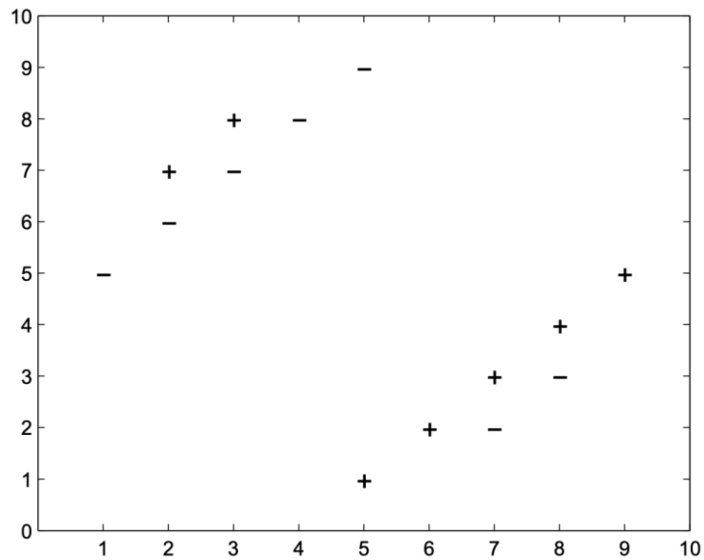
ב KNN עבור K=3 נסווג תמיד שלילי לכל דוגמא וזה כמובן טועה עבור נקודה 2,1.5
עבור עץ ID3 נבחר בלימדת העץ שכל דוגמא עם v2 גדול שווה ל 1.5 הוא חיובי ואחרת שלילי, וזה טועה עבוא דוגמא 1,1 כפי שניתן לראות באיור.

ד. (3 נק') הציגו מסווג מטרה $f(x): R^2 \rightarrow \{0,1\}$ וקבוצת אימון בעלת לכל היותר 10 דוגמאות כך שלמידת מסווג KNN עבור ערך K מסוים תניב מסווג אשר עונה נכון עבור כל דוגמת מבחן אפשרית (כלומר יתקבל מסווג המטרה), וגם למידת עץ ID3 תניב מסווג עונה נכון עבור כל דוגמת מבחן אפשרית (כלומר יתקבל מסווג המטרה).

שוב, נבחר את אותה דוגמה עבור סעיף ב. K=1 וראינו כי הוא צודק. נגדיר בלמידת עץ ה ID3 שלכל דוגמה אם v1 קטן מ 2 אתה שלילי, ואחרת חיובי.



3. (8 נק') בשאלה נשתמש במסווג k-nearest neighbour באמצעות מרחק אוקלידי, במשימת סיווג בינארי.
אנו מגדירים את הסיווג של נקודת המבחן להיות הסיווג של רוב ה-K השכנים הקרובים ביותר (שימו לב שנקודה יכולה להיות שכנה של עצמה).



א. (2 נק') איזה ערך של k ממזער את שגיאת האימון עבור קב׳ הדגימות הנ״ל? מהי שגיאת האימון כתוצאה מכך?

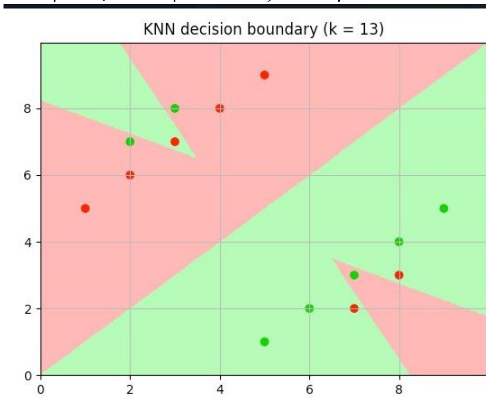
עבור k=1, מכיוון שכל נקודה שכנה של עצמה, תמיד נבחר את עצמה בתור השכן הכי קרוב (כי המרחק הוא 0). לכן הסיווג תמיד יהיה נכון – כלומר השגיאה תהיה 0.

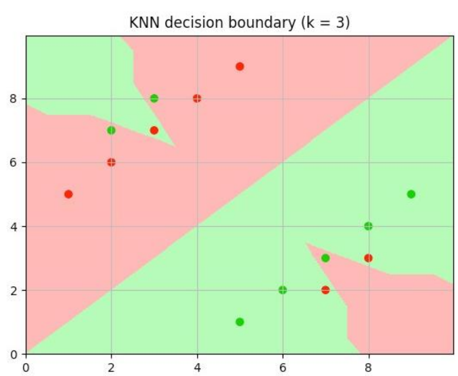
ב. (2 נק') מדוע שימוש בערכי k גדולים מדי יכול להיות גרוע עבור קב׳ הדגימות הנ״ל? למה אולי גם ערכים קטנים של k זה רעיון רע?

נשים לב שהקבוצות מחולקות כמעט לחיובי ושלילי (וגם יש תמונת מראה). לכן באופן אינטואיטיבי נרצה לסווג דגימה על פי איזו צד "במראה" היא יותר קרובה.

אם נבחר k קטן מדי, למשל k = 3, עבור דגימה באינדקס 7,3 נבחר (-) כי יש שכן אחד (+) ושניים (-) (למרות שבבירור היינו רוצים שהדגימה תצא (+)).

אם נבחר K גדול מדי, נוכל להגיע למצב שאנחנו מכילים את שני צידי המראה, ונקבל תוצאה לא נכונה – <u>K=13. לפי הגרף הבא, עבור נקודה 3,5 נקבל</u> תוצאה שגויה:





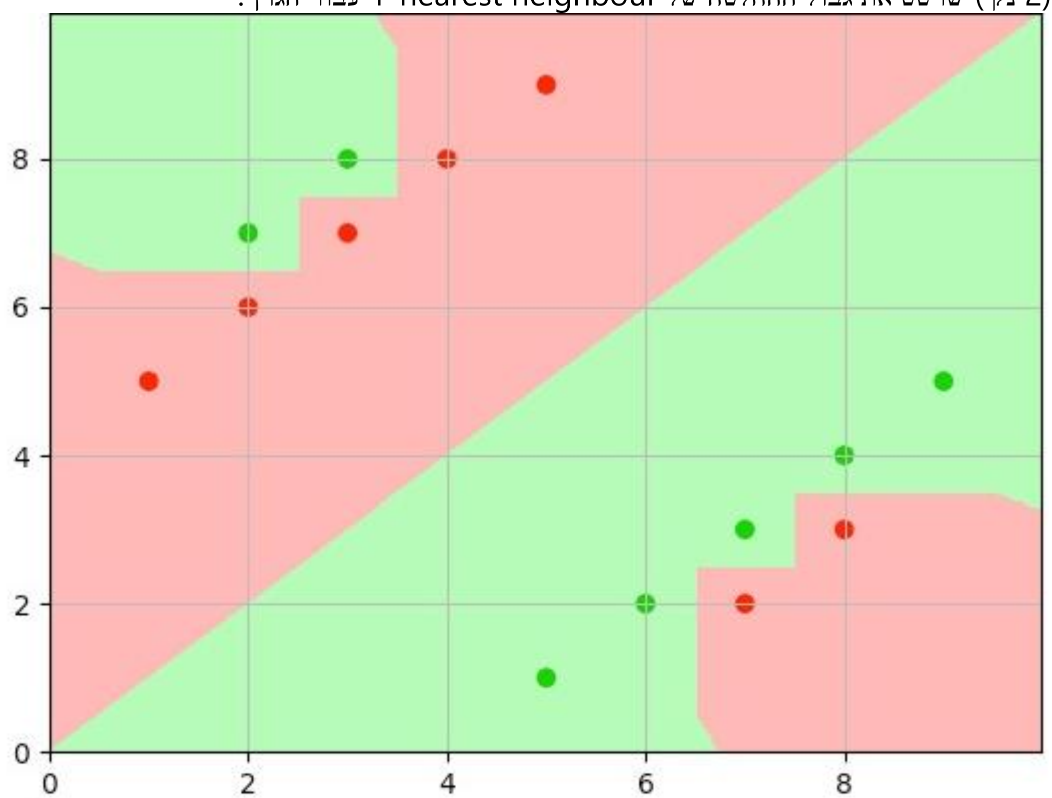
ג. (2 נק') איזה ערך של k ממזער את שגיאת Leave-One-Out Cross Validation עבור קב׳ הדגימות? מהי השגיאה שנוצרה?

https://www.statology.org/leave-one-out-cross-validation/

עבור k=5,6,7 השגיאה המתקבלת היא הימינימלית, וערכה – 28.57% (השגיאה המקסימלתי מתקבלת למשל, עבור k=9 ואז יש 100% שגיאה).

```
For k =  1  error =  71.42857142857143 %
For k =  2  error =  71.42857142857143 %
For k =  3  error =  42.85714285714286 %
For k =  4  error =  64.28571428571428 %
For k =  5  error =  28.57142857142857 %
For k =  6  error =  28.57142857142857 %
For k =  7  error =  28.57142857142857 %
For k =  8  error =  64.28571428571428 %
For k =  9  error =  100.0 %
For k =  10  error =  71.42857142857143 %
For k =  11  error =  100.0 %
For k =  12  error =  71.42857142857143 %
For k =  13  error =  100.0 %
```

ד. (2 נק') שרטט את גבול ההחלטה של 1-nearest neighbour עבור הגרף.



חלק ג׳ – חלק רטוב ID3 (42 נק')

מותר להשתמש בספריות:

All the built in packages in python, sklearn, pandas ,numpy, random, matplotlib, argparse, abc, typing,

אך כמובן שאין להשתמש באלגוריתמי הלמידה, או בכל אלגוריתם או מבנה נתונים אחר המהווה חלק מאלגוריתם למידה אותו תתבקשו לממש.

4. (5 נק') השלימו את הקובץ utils.py ע"י מימוש הפונקציות $l2\_dist$ ו $-accuracy$. קראו את תיעוד הפונקציות ואת ההערות הנמצאות תחת התיאור TODO.
(הריצו את הטסטים המתאימים בקובץ $unit\_test.py$ לוודא שהמימוש שלכם נכון)

5. (25 נק') **אלגוריתם ID3:**

a. ממשו את אלגוריתם $ID3$ כפי שנלמד בהרצאה. TODO
שימו לב שכל התכונות רציפות. אתם מתבקשים להשתמש בשיטה של חלוקה דינמית המתוארת בהרצאה. כאשר בוחנים ערך סף לפיצול של תכונה רציפה, דוגמאות עם ערך השווה לערך הסף משתייכות לקבוצה עם הערכים הגדולים מערך הסף. במקרה שיש כמה תכונות אופטימליות בצומת מסוים בחרו את התכונה בעלת האינדקס המקסימלי.
המימוש צריך להופיע בקובץ בשם $ID3.py$ (השלימו את הקוד החסר אחרי שעיינתם והפנמתם את הקובץ $DecisionTree.py$ ואת המחלקות שהוא מכיל).

b. ממשו $basic\_experiment$ שנמצא ב $ID3\_experiments.py$ TODO
והריצו את החלק המתאים ב $main$ ציינו בדו"ח את הדיוק שקיבלתם. ✍

הדיוק לפני הגיזום – 94.69%

6. (12 נק') **גיזום מוקדם.**

פיצול צומת מתקיים כל עוד יש בו יותר דוגמאות מחסם המינימום m, כלומר בתהליך בניית העץ מבוצע "גיזום מוקדם" כפי שלמדתם בהרצאות. שימו לב כי פירוש הדבר הינו שהעצים הנלמדים אינם בהכרח עקביים עם הדוגמאות .לאחר סיום הלמידה (של עץ יחיד), הסיווג של אובייקט חדש באמצעות העץ שנלמד מתבצע לפי רוב הדוגמאות בעלה המתאים.

✍ **a.** הסבירו מה החשיבות של הגיזום באופן כללי ואיזה תופעה הוא מנסה למנוע?
גיזום נועד לצמצם את גודל העץ כמה שיותר – עצים קטנים בדרך כלל יביאו לנו תוצאות טובות יותר, גם על פי Occams Razor. בנוסף, זה יכול לעזור לנו במניעת overfitting, כך שהעץ שלנו מותאם יתר על המידה על train set שלנו, ויכול גם לטפל ברעשים – דוגמה אחת לא "תזהם" node שלם שמכיל דוגמאות עם label אחר.

b. ממשו את הגיזום המוקדם כפי שהוגדר בהרצאה. הפרמטר M מציין את מספר המינימלי בעלה לקבלת החלטה. על המימוש של הגיזום המוקדם להיות גם כן בתוך המחלקה ID3 שנמצאת בקובץ ($ID3.py$ TODO

c. השתמשו באלגוריתם ID3 עם הגיזום המוקדם כדי ללמוד מסווג מתוך **כל** קבוצת האימון ולבצע חיזוי על קבוצת המבחן. השתמשו בערך ה- M האופטימלי שמצאתם בסעיף c. (ממשו $best\_m\_test$ שנמצאת ב $ID3\_experiments.py$ והריצו את החלק המתאים ב $main$)
ציינו בדו"ח את הדיוק שקיבלתם. האם הגיזום שיפר את הביצועים ביחס להרצה ללא גיזום בשאלה 5?

דיוק אחרי גיזום עבור m=50: 97.35%. קיבלנו שיפור של 2.66 אחוז בתוצאות.