

תרגיל בית 3:

למידה ובחירת מאפיינים

הערות:

- תאריך הגשה: 25.1.18
- את המטלה יש להגיש בזוגות בלבד!
- **בתרגיל זה מומלץ ואף רצוי להשתמש בספריות חיצוניות כמו sklearn אלא אם כן צוין מפורשות לא להשתמש בספריות אלו.**
- צרפו דו"ח שעונה על כל השאלות המופיעות בתרגיל.
- שאלות בנוגע לתרגיל ניתן לשלוח לתיבת המייל הקורסית: ai.technion@gmail.com
- בקשות מוצדקות לדחייה יש לשלוח לאמיר: amirc@cs.technion.ac.il
- קראו היטב את ההסברים וההוראות במסמך זה, מטרתם לסייע לכם בהבנת הדרישות של התרגיל. שימו לב להוראות ההגשה המצורפות בסוף התרגיל.
- התעדכנו ברשימת ה-FAQ באתר הקורס בתדירות גבוהה, לפני פנייה בשאלות דרך המייל ולפני הגשת התרגיל. ההערות שתתפרסמנה באתר הקורס מחייבות את כלל הסטודנטים בקורס!
- הקוד שלכם יבדק על-ידי *Unit – Tests*, לכן עליכם לעקוב בתשומת לב רבה אחר הוראות ההגשה המצורפות במהלך התרגיל ובסופו לפני הגשתו.
- בתרגיל עץ החלטה הוא id3 כפי שנלמד בכיתה.

בתרגיל זה, כמו גם בתרגילים האחרים בקורס, הרצת הניסויים עשויה לקחת זמן רב ולכן מומלץ מאוד להימנע מדחיית העבודה על התרגיל לרגע האחרון. לא תינתנה דחיות על רקע זה

חלק 1 – מדידת ביצועי המסווג

תהליך הלמידה מנסה להביא לאופטימום את ביצועי המסווג על מדד מסוים (מדד נפוץ אותו נרצה לשפר הוא הדיוק) מדידת ביצועי המסווג חשובה מכיוון שבעזרתה ניתן לשפר את תהליך הלמידה. אנו נשתמש במדידת ביצועי המסווגים על מנת לבחור אלגוריתם למידה ולכוון את הפרמטרים שלו במטרה לשפר את התוצאות. בסיום התהליך נאמוד את ביצועי המסווג המתקבל.

כדי לכוון פרמטרים באופן יעיל נרצה להבין את הבעיה ואת ביצועי המסווג בבעיה זו. תובנות אלו יעזרו לכוון פרמטרים כך שישפרו את ביצועי המסווג.

דוגמה לבעיה שהבנתה תוכל לעזור בתהליך הלמידה היא כאשר יש חוסר איזון בין יחס המחלקות באוכלוסייה. למשל עבור בעיית סיווג בינארית (כל דוגמה מסווגת ידי סיווג שלילי או חיובי) אוכלוסייה לא מאוזנת יכולה להיות כזו ש 90% מהאוכלוסייה שייכת למחלקה השלילית ו-10% שייכת למחלקה החיובית.

במקרה זה סיווג של כל הדוגמאות כשליליות יגיע ל-90% דיוק. נרצה לנסות לפתור את הבעיה כך שהמסווג ילמד להפריד בין דוגמאות שליליות לחיוביות ולשפר את הדיוק למעל 90%. מטרתנו בחלק זה היא להבין את המשמעות של המדדים השונים, להכיר בעיות נפוצות, ולרכוש את הבסיס הנחוץ על מנת להתמודד עם מקרים שונים שאולי תתקלו בהם בעתיד.

עתה נעבור לחידוד של ההגדרות מהכיתה:

לצורך מדידת ביצועי המסווג, למדנו שאת הדוגמאות מחלקים לקבוצת אימון ולקבוצת מבחן. בנוסף למדנו כי אם ברצוננו לכוון פרמטרים, כיוון פרמטרים ייחשב כחלק מתהליך הלמידה ולכן נרצה להשתמש בקבוצת האימון לצורך בחירת הפרמטרים. חשוב לבצע את ההפרדה הזאת על מנת שכאשר נמדוד את ביצועי המסווג על קבוצת המבחן לא נקבל תוצאות אופטימיות מדי, תוצאות שהתקבלו ע"י התאמה של הלומד לקבוצת המבחן. תוצאות אלו לא ישיקפו את ביצועי הלומד בעולם האמיתי. לכן, על מנת לכוון פרמטרים נפצל את קבוצת האימון (Training set) לשתי קבוצות שונות:

1. $T' = \text{Subset of the Training set}$: סט אימון
2. $\text{Validation} = \text{Training set} - T'$: סט לצורך כוון פרמטרים.

שימו לב - אם אין צורך בבחירת פרמטרים אלא רק במדידת ביצועים, נוכל למדוד בעזרת שיטת cross validation במקרה זה כל חלוקה של המידע תהיה חלוקה ל Test set ו Training set

במהלך התרגיל נשתמש בנתונים אשר נאספו על מנת לאפשר סיווג של התפרצויות סולריות. הנתונים נלקחו מהאתר *UCI Machine Learning Repository: Data Sets*. הנתונים שזו מעט והגרסה הרלוונטית עבור תרגיל זה נמצאת בקובץ המצורף: *flare.csv*.

שאלה 1 (12 נק):

בשאלה זו נרצה לאמן מסווג וללמוד כיצד להציג את ביצועי המסווג בצורה אינפורמטיבית.

א. אמנו עץ החלטה ללא גיזום שמשמש ב-ID3 כפי שנלמד בכיתה.

לצורך כך הם השתמשו בנתונים המצורפים בקובץ *flare.csv* ואימנו בעזרת cross validation עם 4 חלקים (*folds*). כיצד עליכם לשלב את המידע שהתקבל ע"י אימון בשיטת cross validation כדי לקבל מדד דיוק יחיד? אמדו את הדיוק (*accuracy*) בשיטה המוצעת.

ב. בתרגול המבוא ללמידה דיברנו על סוגי שגיאות. וראינו את מטריצת הבלבול; למידע נוסף ראו את [הלינק](#). צרפו דיאגרמה של מטריצת הבלבול (*confusion matrix*) עבור התוצאות מהסעיף קודם (הרצת cross validation עם 4 חלקים) איך נשלב את התוצאות להצגת מטריצה יחידה?

עבור שאלה 1 הגישו קובץ פיתון בשם $q1.py$ אשר טוען את הנתונים, מאמן את הלומד ומדפיס את התוצאות באופן הבא: שורה ראשונה הדיוק שיצא, שורה שניה מטריצת בלבול בפורמט דומה לפורמט שראינו בתרגול המבוא ללמידה, ניתן להשתמש בפונקציות ספריה מוכנות כדי לייצא את המטריצה. דוגמה לפלט אפשרי:

0.911748804635

[[329 94]

[18 828]]

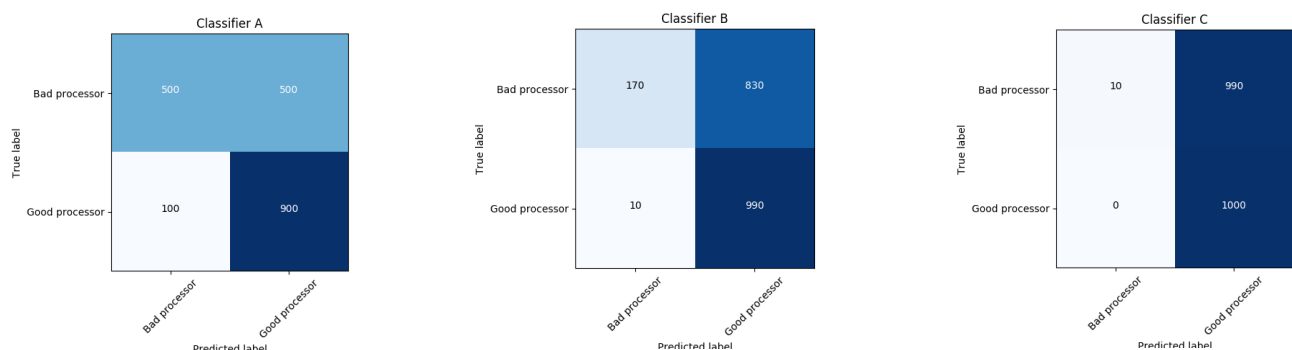
שאלה 2 (10 נק'):

בשאלה זו נרצה לראות היכן מדד הדיוק אינו מספיק ויש צורך במדדים אחרים, במקרה זה מטריצת הבלבול, על מנת לבחור נכונה מסווג.

חברה לייצור מעבדים רוצה לבדוק שהמעבדים שהיא מוציאה עובדים כראוי. על מנת לעשות זאת הם מבצעים וורפיקציה על כל המעבדים לפני הוצאתם אל השוק. תהליכי הוורפיקציה מושלמים בכך שהם יודעים להגיד בדיוק של 100% האם מעבד תקין או לא. הבעיה בביצוע וורפיקציה על כל המעבדים היא שהרצת וורפיקציה על מעבד לוקחת המון משאבי מחשב וזמן ולכן יקרה מאד. הקריטריון אשר מעניין את החברה הוא מינימום עלות. לכן רוצים מסווג שידע להגיד בביטחון גבוה האם המעבד לא תקין מבלי לבזבז משאבים. מאחר והוצאת מעבד לא תקין לשוק היא קטסטרופלית מבחינת החברה, מהנדסי החברה מציעים את הפתרון הבא: במידה ואלגוריתם הסיווג אומר שהמעבד תקין נריץ עליו וורפיקציה כדי לוודא שאכן כך הדבר. במידה והמסווג אומר שהמעבד לא תקין נזרוק את המעבד מבלי להריץ עליו וורפיקציה. עלות החשמל עבור אלגוריתם הוורפיקציה היא 1,000 ₪. עלות של מעבד היא 10,000 ₪. שימו לב - לכל מעבד מורצת הוורפיקציה לפני שהוא יוצא לשוק, היא לא תורץ רק אם המסווג הגדיר את המעבד כלא תקין.

א. הגדירו את הבעיה במושגים של מטריצת הבלבול, התייחסו אל מעבד תקין בתור חיובי. (False Positive, False Negative, True Positive, True Negative).

ב. בהינתן הביצועים הממוצעים הבאים עבור שלושה מסווגים, איזה מסווג הכי כדאי לחברה לבחור? הסבירו



שאלה 3 (12 נק'):

כעת ננסה לענות על השאלה האם יכולת ההכללה של הלומד מספיק טובה כדי לסווג גם דוגמאות חדשות שלא ראה לפני כן?

נחלק את השאלה לשני מקרים:

האם הלומד בעל יכולת תיאור מספיק חזקה כדי לתפוס את הקשר בין המאפיינים למחלקות? כאשר התשובה שלילית, אומרים כי הלומד אינו מצליח ללמוד את מושג המטרה וכי הוא בעל הטיה גבוהה (high bias) ויש לנו בעיה של תת התאמה (under fitting). אם הלומד התאים את עצמו יותר מדי לסט האימון (over fitting) השגיאה על סט האימון נמוכה אך עדיין יכולת ההכללה של המסווג לדוגמאות שלא ראה נמוכה. במקרה זה אומרים הלומד הוא בעל שונות גבוהה (High variance)

עבור קובץ הנתונים שלנו *flare.csv*, חלקו את הנתונים לסט אימון בגודל 75% מהנתונים וסט מבחן בגודל 25% מהנתונים ואמנו מסווג עץ אשר מדגים *under fit* ו *over fit*. סך הכול 2 מסווגים. על מנת לבדוק האם הלומד בעל הטיה גבוהה או האם הוא בעל שונות גבוהה, מומלץ לבדוק מהו הדיוק שקיבלתם על סט האימון ועל סט המבחן.

- א. מהם הפרמטרים אשר אותם שיניתם כדי לגרום להתאמת יתר?
 - ב. מהם הפרמטרים אשר אותם שיניתם כדי לגרום לתת התאמה?
 - ג. מהו הדיוק שקיבלתם על סט האימון וסט המבחן בכל אחד מן המקרים? הסבירו מדוע תוצאות אלו מתאימות להתאמת יתר או תת התאמה.
- במידה ואינכם בטוחים בתשובותיכם הסבירו למה.

עבור שאלה 3 הגישו קובץ פיתון בשם *q3.py* אשר טוען את הנתונים, מאמן את הלומדים ומדפיס את התוצאות באופן הבא: שורה ראשונה הדיוק שיצא תחת התאמת יתר בעץ, שורה שניה הדיוק שיצא תחת תת התאמה בעץ.

חלק 2 – בחירת מאפיינים.

בחלק זה נתמקד בבעיית בחירת תת קבוצה של מאפיינים מתוך קבוצת המאפיינים המלאה. בהינתן קבוצת המאפיינים המלאה בגודל N , $S = \{x_1, x_2, \dots, x_N\}$, נרצה לבחור מתוך מאפיינים אלו רק תת קבוצה של מאפיינים בגודל b ($b < N$), אשר ישמשו אותנו בבניית המסווג.

הסיבות שבגינן נרצה לבצע בחירת תת קבוצה של מאפיינים הן מגוונות, להלן חלק מהסיבות:

- מחיקת מאפיינים מיותרים, כלומר כאלו אשר אינם קשורים ואינם תורמים מידע עבור הפרדה בין המחלקות השונות, או לחלופין מאפיינים אשר ניתן לבטא אותם ע"י מאפיינים אחרים בקבוצה ולכן לא תורמים לנו מידע רלוונטי ויכולים אף לפגוע בביצועי המסווג.
- מתן תובנות על בעיית הסיווג: בחירת המאפיינים תיתן לנו מידע לגבי הרלוונטיות של המאפיינים לתהליך הסיווג.
- הקטנת הסיבוכיות החישובית.

לצורך משימת בחירת המאפיינים נבחר שתי שיטות לבחירת תת קבוצה של מאפיינים ונשווה בין ביצועי המסווגים המתקבלים ע"י שימוש בכל אחת מהשיטות.

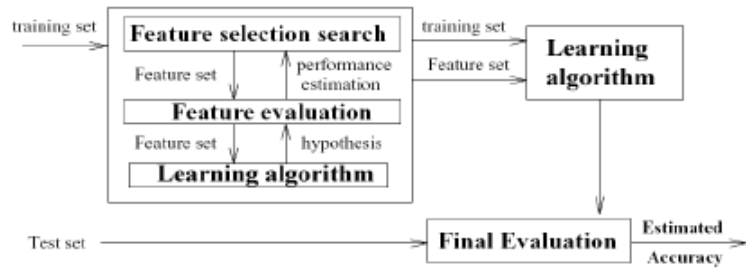
להלן תיאור כללי של השיטות לבחירת מאפיינים שאותן נבחר (בהמשך נרחיב על כל אחת מהשיטות):

- *Wrapper method* - בשיטה זו בחירת המאפיינים נעשית כשלב מקדים לאימון המסווג. נתייחס לבעיית בחירת תת קבוצת המאפיינים כבעיית חיפוש לוקלי ונפתח פונקציית עוקב ופונקציית תועלת כדי להנחות את החיפוש.
- *Embedded method* - בשיטה זו בחירת מאפיינים באופן מתבצעת מובלע בתהליך האימון.

Wrapper method

נרצה למצוא תת קבוצה טובה של מאפיינים בגודל b עבורה הדיוק של אלגוריתם הלמידה יהיה אופטימלי. בחלק זה נניח כי גודל תת הקבוצה b נתון, אך נציין כי בבעיות מהעולם האמיתי מספר המאפיינים הרלוונטיים אינו ידוע בהכרח והוא פרמטר של בעיית הלמידה. כדי למצוא תת קבוצה של מאפיינים קבוצת שיטות ה-*wrapper* משתמשות במסווג בתור קופסא שחורה. בכל שלב מאמנים את המסווג מחדש על תת קבוצה של המאפיינים ומנסים למצוא את קבוצת המאפיינים שתביא למקסימום את ביצועי המסווג.

להלן סכמה של שיטת בחירת המאפיינים:



שאלה 4 (5 נק'):

נגדיר את כל תתי הקבוצות של S כ $P(S)$. כלומר, כל תתי הקבוצות האפשריות בגודל קטן או שווה ל N .

1. מהו מספר כל תתי הקבוצות של הקבוצה S ?
2. מהו מספר כל תתי הקבוצות בגודל b של הקבוצה S ?

בשאלה 4 ראינו כי מספר תת הקבוצות האפשריות של מאפיינים הוא לא פולינומיאלי בגודל הקבוצה, באופן כללי מציאת קבוצת המאפיינים האופטימלית עבור מושג מטרה היא בעיה NP קשה. לכן בחלק זה נתייחס אל בעיית בחירת מאפיינים כאל בעיית אופטימיזציה ונפתור אותה בעזרת חיפוש לוקאלי. מכאן שניתן לחלק את משימת בחירת המאפיינים לשלוש תתי בעיות:

- הגדרת מרחב המצבים
 - הגדרת פונקציית תועלת U אשר נותנת ציון לבחירה מסוימת של תת קבוצה של מאפיינים (הפונקציה U , תרגול 7).
 - הגדרת פונקציית עוקב (*Successor*) שתדריך את החיפוש ותמנע פיתוח של צמתים במרחב המאפיינים.
- נתחיל בהגדרת מרחב המצבים. אנו רוצים לבחור תת קבוצה של מאפיינים בגודל b ולכן נבחר להסתכל על וקטורים בינאריים ממימד N כאשר ביט דולק מסמן מאפיין שנבחר. פונקציית תועלת U תוגדר כפונקציה המקבלת תת קבוצה של קבוצת המאפיינים מאמנת את המסווג על תת קבוצת מאפיינים זו ומחזירה מדד לביצועי המסווג. את אמידת ביצועי המסווג נבצע בעזרת שיטת *cross validation* עם 4 חלקים. הערכה לביצועי המסווג על תת קבוצת מאפיינים היא הדיוק (אחוז הסיווגים הנכונים) של אלגוריתם הלמידה.
- כלומר $U : P(s) \rightarrow (0,1)$

שאלה 5 (5 נק'):

על איזו קבוצה (קבוצת אימון ולידציה או מבחן), יש לבדוק את הביצועים? הסבר.

האלגוריתם החיפוש הלוקלי בו נשתמש הוא *Sequential forward selection (SFS)*.
כעת נגדיר את פונקציית העוקב (*Successor*) עבור אלגוריתם זה.

האלגוריתם מתחיל בקבוצה ריקה של מאפיינים ובכל שלב האלגוריתם מוסיף מאפיין נוסף.

להלן תיאור של האלגוריתם:

נתחיל מהקבוצה הריקה $V_0 = \emptyset$

נסמן ב V_k את תת הקבוצה של מאפיינים בגודל k של המאפיינים שכבר בחרנו.

1. יהי v_j מאפיין מתוך קבוצת המאפיינים שעדיין לא בחרנו.

כלומר, $v_j \in \{x_1, x_2, \dots, x_N\} - V_k$ כך שמתקיים:

$U(v_j \cup V_k) \geq U(v_i \cup V_k) \quad \forall v_i \in \{x_1, x_2, \dots, x_N\} - V_k$. כלומר המאפיין v_j נבחר כמאפיין

המשפר את פונקציית התועלת U בצורה המקסימלית.

2. נוסיף את v_j לקבוצת המאפיינים שכבר נבחרו. $V_{k+1} = V_k \cup \{v_j\}$

3. נחזור על שלבים 1,2 עד שיהיו בידנו b מאפיינים.

שאלה 6 (20 נק'):

ממשו בקובץ `sfs.py` את הפונקציה `sfs`, פונקציה עוטפת `Sequential forward selection` גנרית אשר יכולה לקבל כל אלגוריתם למידה בתור פרמטר לפונקציה. אלגוריתם הלמידה יאומן מחדש עבור כל תת קבוצה של מאפיינים ובכל שלב יבחר המאפיין הממקסם את פונקציית התועלת. למידע נוסף לגבי הפרמטרים אותה הפונקציה מקבלת ראו את התיעוד בקובץ `sfs.py`.
הגישו את קובץ הפיתון המכיל את המימוש שלכם (שימוש בספריות קיימות לא נחשב).

שאלה 7 (10 נק'):

בשאלה זו נשתמש ב `SFS` כאמצעי לבחירת מאפיינים. חלקו את הנתונים לסט אימון בגודל 75% מהנתונים וסט מבחן בגודל 25%.

1. השתמשו באלגוריתם `KNN` כדי לסווג את הנתונים בקובץ `flare.csv`, ניתן לממש בעצמכם או

להשתמש ב `Sklearn`, $(K = 5)$ ללא בחירת מאפיינים. מהו הדיוק המתקבל?

2. השתמשו בפונקציה `sfs.py` שמימשתם בסעיף קודם, השתמשו באלגוריתם `KNN` $(K = 5)$

בתור אלגוריתם הלמידה שהפונקציה מקבלת. עבור אלגוריתם `SFS` בחרו רק $b = 8$ מאפיינים.

מהו הדיוק המתקבל?

בנוס (10 נק'):

מאחר מדובר באלגוריתם חיפוש לוקאלי יתכן ולא ייתן את התוצאה האופטימלית. בשאלה הבאה נראה מקרה כנ"ל אשר מבוסס על מאמר מאת *Thomas M. Cover*
The Best Two Independent Measurements are Not the Two Best

נתבונן בארבעת המאפיינים: x_1, x_2, x_3, x_4 ונניח כי מתקיים $U(\{x_1\}) \geq U(\{x_2\}) \geq U(\{x_3\}) \geq U(\{x_4\})$
ונניח כי ברצוננו לבחור שני מאפיינים בלבד. כלומר, במקרה זה $N = 4, b = 2$.
הציעו מקרה שעבורו המאפיינים הנבחרים הם x_3, x_4 . כלומר $U(\{x_3, x_4\})$ הוא מקסימלי עבור הזוג

Embedded method

בשיטות אלו מנגנון בחירת המאפיינים הוא חלק מפרוצדורת האימון של אלגוריתם הלמידה.
אלגוריתמים מקבוצה זו מנסים לבצע בחירת תת קבוצה של מאפיינים באופן מובלע בתהליך האימון.
דוגמה לאלגוריתם ממשפחה זו הוא עץ החלטה עם גיזום, כאשר העץ מבצע גיזום האלגוריתם בוחר לא להשתמש במאפיינים נוספים, מתוך כך נובע שיתכן שייבחרו רק חלק מן המאפיינים.

שאלה 8 (11 נק'):

צרפו את הקוד עבור שאלה זו בקובץ *Q8.py*
בשאלה זו נשווה שני ווריאנטים של עץ החלטה, בלי גיזום ועם גיזום.
בעץ עם גיזום מוקדם נשתמש כאמצעי לבחירת מאפיינים.
עבור הנתונים עליהם עבדתם (*flare.csv*) חלקו את הנתונים לקבוצת מבחן וקבוצת אימון (כמו בשאלה 8).

גם בשאלה זו נכוון פרמטרים בעזרת *cross validation* עם 4 חלקים (*folds*):

- מהו הדיוק המתקבל עבור עץ החלטה ללא גיזום?
- אמנו עץ החלטה עם גיזום מוקדם. פתחו את עץ ההחלטה עד אשר מספר הדוגמאות בצומת ≤ 20 , בשלב זה הפסיקו את פיצול הצומת והפכו אותו לעלה. כרגיל, נבצע את ההחלטה לפי רוב הדוגמאות בצומת. מהו הדיוק המתקבל?
- איזה אלגוריתם נתן דיוק גבוה יותר, על איזו קבוצה בדקתם?
- מהו הדיוק המתקבל עבור האלגוריתם הנבחר, על איזו קבוצה אימנתם $(T \setminus \text{Training} \setminus \text{Validation})$? על איזה קבוצה נמדד הדיוק?

שאלה 9 (10 נק'):

האם קיבלתם שיפור בביצועים בעזרת בחירת מאפיינים? עבור איזו שיטה (*Embedded Wrapper*) קיבלתם תוצאות טובות יותר?

הגישו קובץ פייתון בשם *part2.py* אשר יאמן את הלומדים ויבצע בחירת מאפיינים כפי שהוגדר. בנוסף הדפיסו את הביצועים הממוצעים של כל אחד מהלומדים משאלות 7-8 לפי הסדר הנשאל, כל אחד בשורה.

שאלה 10 (5 נק'):

תנו יתרון אחד וחסרון אחד של *Wrapper methods* לעומת *Embedded method*.

בונוס תחרות

בחלק זה נבקש מכם לאמן מסווג על משימה חדשה, חיזוי רמת הצלילות של יהלומים. אמנו מסווג על הסט `diamonds.csv`, אתם רשאים לאמן כל מסווג שתבחרו ולהשתמש בכל טכניקה שתמצאו באימון המסווג. מצורף סט הנקרא `unlabeled_diamonds.csv`, בסט זה הושמטו התיוגים ומשימתכם לתייג אותו מחדש. הקבוצה שתוציא את הדיוק הגבוהה ביותר אותו נמדוד על הסט הלא מתיוג תנצח. יש להגיש את הקובץ המלא כאשר התיוגים מופיעים בעמודה חדשה (סדר השורות חייב להישמר).

הבונוס יינתן לפי ביצועי המסווג שלכם על סט המבחן. התרגיל שיקבל על קבוצה זו את הדיוק הגבוה ביותר יקבל בונוס של 10 נקודות מקום שני יקבל בונוס של 7 נקודות ומקום שלישי 5 נקודות (לציון תרגיל זה). לצורך הבדיקה צרפו סקריפט אשר טוען סט אימון ומאמן את המסווג על קבוצה זו `bonus.py`. והגישו אותו יחד עם הקובץ עם התוויות שחזיתם.

הוראות הגשה

- הגשת התרגיל תתבצע אלקטרונית בלבד.
- עליכם להגיש קובץ ארכיון יחיד בשם: `AI3_<id1>_<id2>.zip` (ללא הסוגריים המשולשים). קובץ זה יכיל:
 - קובץ בשם `readme.txt` בפורמט הבא:

```
name1 id1 email1
name2 id2 email2
```
 - קובץ בשם `AI_HW3.PDF` המכיל את דו"ח הניסויים שערכתם, תשובות לחלק היבש והערות לקוד שהגשתם (כולל תפקיד כל קובץ הנמצא בתיקייה שהגשתם).
 - קובץ בשם `requirements.txt` שיכיל כל חבילה חיצונית שאינה מותקנת כחלק מ-Anaconda Python. על אחריותכם לוודא כי ניתן להתקין כל חבילה כנ"ל באמצעות הרצת השורה: `pip install -r requirements.txt`
 - כל הפונקציות שהתבקשתם לממש בתרגיל זה.
 - כל קוד עזר שכתבתם/השתמשתם בו לשם הרצת הניסויים או יצירת הגרפים.
 - כל קובץ פלט שהתבקשתם ליצור לצורך התרגיל.
- אין להעתיק את הקבצים המסופקים לכם אל תוך תיקיית ההגשה. הניחו כי קבצים אלו יהיו זמינים בעת בדיקת התרגיל.
- שימו לב שכל הפנייה למיקום קובץ/תיקייה כלשהם בקוד תהיה רלטיבית (*relative path*) ולא אבסולוטית, כך שהקוד יעבוד כפי שהוא על כל מחשב בכל מיקום שנבחר לתיקיית הפרוייקט. הקפידו לבדוק זאת לפני ההגשה!
- "המצאת" נתונים לצורך בניית הגרפים **אסורה** ותוביל לדיון בבית הדין המשמעתי של הטכניון.
- אתם רשאים לעשות שימוש בכל קוד שתמצאו ברשת, אך כל קוד חיצוני **מחייב הצהרה מפורשת** על המקור שלו בקובץ `AI_HW3.PDF`. אי-קיום דרישה זו מהווה עבירה משמעטית!
- הקפידו על קוד **ברור, קריא ומתועד!** עליכם לתעד כל חלק שאינו טריוויאלי בקוד שלכם. בפרט, אם השתמשתם בקוד שנמצא ברשת וביצעתם בו שינויים, עליכם לתעד זאת.

בהצלחה!