דו"ח מסכם

תרגיל בית 3

יונתן בתן

302279138

Yonibettan@gmail.com

נדב אליהו

303086854

Neliahu@gamil.com

שאלה 1:

א. על מנת לאמן את עץ ההחלטה השתמשנו בחבילה decision-tree-id3 תוך חלוקת המידע שניתן לנו ל3 קבוצות של 106 וקבוצה של 105 כלומר ל4 קבוצות כמעט שוות. אימנו את העץ 4 פעמים כל פעם הוצאנו קבוצה אחרת מהמידע והפכנו אותה לקבוצה מבחן. עבור כל איטרציה חישבנו את הדיוק של העץ (בכמה מהחיזויים צדקנו) באמצעות sklearn. לבסוף קיבלנו 4 מדדי דיוק עבור כל עץ. לבסוף איחדנו את המדד דיוק על ידי ממוצע בין 4 המדדים.

ב. בנוסף למוזכר בסעיף א חישבנו את מטריצת הבלבול לכל איטרציה באמצעות sklearn, על מנת לאחד בין המטריצות פשוט נסכום אותן למטריצה אחת, מטריצה של כל איטרציה מכילה את הבלבול עבור קבוצת מידע שונה ולכן המטריצה הסופית תכיל את הבלבול עבור כלל המידע כאשר נסכום.

התוצאה עבור חלק 1:

```
0.716307277628
[[235 47]
[73 68]]
```

שאלה 2:

א. נגדיר את הבעיה במושגי מטריצת הבלבול:
   a. סיווג מוטעה של מעבד כתקין - False Positive
   b. סיווג נכון של מעבד כתקין - True Positive
   c. סיווג נכון של מעבד כלא תקין - True Negative
   d. סיווג מוטעה של מעבד כלא תקין - False Negative

ב. על פי התיקון בFAQ, עלות יצור מאבד אפסית, עלות ביצוע ורפיקציה אשר מבוצעת על כל מחשב שסווג כתקין היא 1000 ₪ ורווח עבור מעבד שנמצא תקין ויצא לשוק הוא 10,000 ₪. החברה מעוניינת למקסם את הרווח שלה.
   a. עבור מסווג a סווגו כתקינים - 500 מעבדים לא תקינים, 900 תקינים. כלומר החברה בזבזה על ורפיקציה 1400*1000 ₪ והרוויחה 10,000*900 ₪. סך הכל רווחים: 7,600,000 ₪.
   b. עבור מסווג b סווגו כתיקנים – 830 מעבדים לא תקינים ו990 תקינים. כלומר החברה בזבזה על ורפיקציה 1820*1000 ₪ והרוויחה 10,000*990 ₪. סך הכל רווחים: **8,080,000 ₪**.
   c. עבור מסווג c סווגו כתיקנים – 990 מעבדים לא תקינים ו1000 תקינים. כלומר החברה בזבזה על ורפיקציה 1990*1000 ₪ והרוויחה 10,000*1000 ₪. סך הכל רווחים: 8,010,000 ₪.

לכן המסווג באמצעותו תמקסם החברה את הרווחים הוא מסווג b.

שאלה 3:

א. על מנת לקבל התאמת יתר הגדרנו כי עומק העץ המרבי יהיה 32 (כמספר המאפיינים) כלומר אנו לא מגבילים פיצולים בעץ בגלל עומק המסלול ובכך מאפשרים תשובה מאוד מדויקת ביחס לסט האימון.

ב. על מנת לקבל תת התאמה הגדרנו את עומק העץ המרבי ל1 – כלומר הרבה אנו מגבילים את הפיצולים ובכך נקבל תשובה לא מדויקת על סט האימון

ג. עבור התאמת יתר קיבלנו כי על סט האימון הדיוק הוא 0.89 – כלומר מאוד גבוהה (היינו מצפים לראות דיוק 1 אולם ישנן דוגמאות סותרות בflare הגורמות לאי דיוק זה). הדיוק שקיבלנו עבור סט המבחן הוא 0.76 – מה שמראה שהתאמת יתר אכן גורמת לעץ ההחלטה להתאים עצמו יתר על המידה לסט האימון אבל זה לא מעיד על מידע חדש שאינו מכיר.
עבור תת התאמה קיבלנו כי על סט האימון הדיוק הוא 0.70 ואילו על סט המבחן 0.72 - שני הדיוקים מאוד דומים מאחר ועץ ההחלטה כמעט ולא מושפע מסט האימון והתגובה שלו לסט האימון וסט המבחן דומה.

* בקוד הדפסנו את תוצאות הדיוק רק עבור סט האימון כפי שצוין בFAQ.

שאלה 4:

א. כל תת קבוצה בS ניתן לייצג כווקטור באורך N כאשר עבור מאפיין i נגדיר את ערך הווקטור במיקום i כ0 אם הוא לא בקבוצה ו1 אם הוא בקבוצה. כל ווקטור שונה מייצג תת קבוצה שונה – מספר הווקטורים הבינאריים השונים בעלי N ספרות הוא 2^N.

ב. עתה אנו נדרשים לדעת כמה תתי קבוצות בגודל b יש בS, כלומר כמה אופציות בחירה של b מאפיינים מתוך N יש לנו כלומר $\binom{N}{b}$ תתי קבוצות.

שאלה 5:

האלגוריתם לבחירת קבוצת הפרמטרים המתאימה משתמש בשיטת cross validation עם 4 חלקים כלומר הוא בודק את אמינות הסיווג עבור כל קבוצת פרמטרים 4 פעמים על קבוצות הוולידציה והממוצע שלהן הוא הדיוק עבור אותם פרמטרים.
לאחר בחירת הפרמטרים מבצעים אימון נוסף עם כלל הסט אימון (גם חלקי הוולידציה) ובודקים את הביצועים שלו על קבוצת המבחן. כלומר לאחר שנבחרו הפרמטרים יש לבדוק את הביצועים על פי קבוצת המבחן. על מנת לבחור את הפרמטרים יש לבדוק את הביצועים על קבוצת הוולידציה.

שאלה 6:

קוד מצורף.

שאלה 7:

א. הדיוק המתקבל עבור KNN ללא בחירת פרמטרים הוא 0.707

ב. הדיוק המתקבל עבור KNN עם בחירת 8 פרמטרים הוא 0.792

בונוס:

נחפש דוגמא בה סיווג באמצעות מאפיין אחד יביא את הדירוג הנתון $U(\{x_1\}) \geq U(\{x_2\}) \geq U(\{x_3\}) \geq U(\{x_4\})$ אבל עבור צמדים נקבל כי מאפיינים 3,4 הם עם הדיוק הטוב ביותר.

	x_4	x_3	x_2	x_1	סיווג
1	1	0	0	0	True
2	1	0	1	1	True
3	1	0	1	1	True
4	0	0	0	0	False
1	0	1	0	0	True
2	0	1	1	1	True
3	0	1	1	1	True
4	1	1	0	0	False

נשים לב כי על פי מאפיין 1 נקבל דיוק של 75%, באופן דומה עבור מאפיין 2. עבור מאפיינים 3 ו4 נקבל דיוק 50%. ולכן מתקיים התנאי הראשוני $U(\{x_1\}) \geq U(\{x_2\}) \geq U(\{x_3\}) \geq U(\{x_4\})$ ובעקבות כך המאפיין הראשון שיבחר יהיה 1 או 2.

נשים לב כי אם נבחר במאפיינים 3 ו4 מxor שלהם נקבל דיוק של 100% אולם הבחירות הבאות יביאו לדיוק פחות:

1 ו2 – לא מוסיפים מידע אחד לשני ולכן דיוק 50%.

1 ו3 או 2 ו3 – אם ננסה לסווג דוגמא בה 2 התכונות 0 נוכל לקבל אמת או שקר לכן הדיוק יהיה פחות מ100%.

1 ו4 או 1ו4 – אם ננסה לסווג דוגמא בה 2 התכונות 0 נוכל לקבל אמת או שקר לכן הדיוק יהיה פחות מ100%.

כלומר בשום מצב לא נקבל את הצמד הרצוי שהוא 3 ו4 עבורו נקבל דיוק 100%.

שאלה 8:

א. עבור עץ החלטה ללא גיזום מתקבל דיוק 0.745
ב. עבור עץ עם גיזום מוקדם מתקבל דיוק 0.735
ג. האלגוריתם עם הדיוק הגבוהה יותר התקבל עבור זה ללא הגיזום אולם לא בהרבה. את האלגוריתם אימנו עם 75% מהמידע ובחנו על ה25% הנותרים. לבחירת 75% הרשומות הייתה השפעה על טיב האלגוריתם – כלומר אם מתבצעת חלוקה רנדומלית של המידע במקרים מסוימים אלגוריתם הגיזום טוב יותר ובאחרים ללא גיזום טוב יותר – הסיבה תלויה במידע עליו אנחנו מאמנים את האלגוריתם – אם במידע יש רעש (סיווגים מוטעים) יש יתרון לגיזום אשר מונע מרעש קטן לשנות החלטה מצד שני אם נגזום יותר מדי אנו כבר נפגע בעץ ההחלטה ו"נסנן" גם דוגמאות נכונות אשר יכלו לגרום לשינוי ההחלטה ללא גיזום. מאחר ואנחנו בוחרים בצורה רנדומלית את 75% המידע עליו אנו מאמנים את האלגוריתם כמות הרעש אינה קבועה ומשפיעה על התוצאות ספציפית עם מאגר המידע בו אנו משתמשים.
ד. סעיף זה בוטל (FAQ).

שאלה 9:

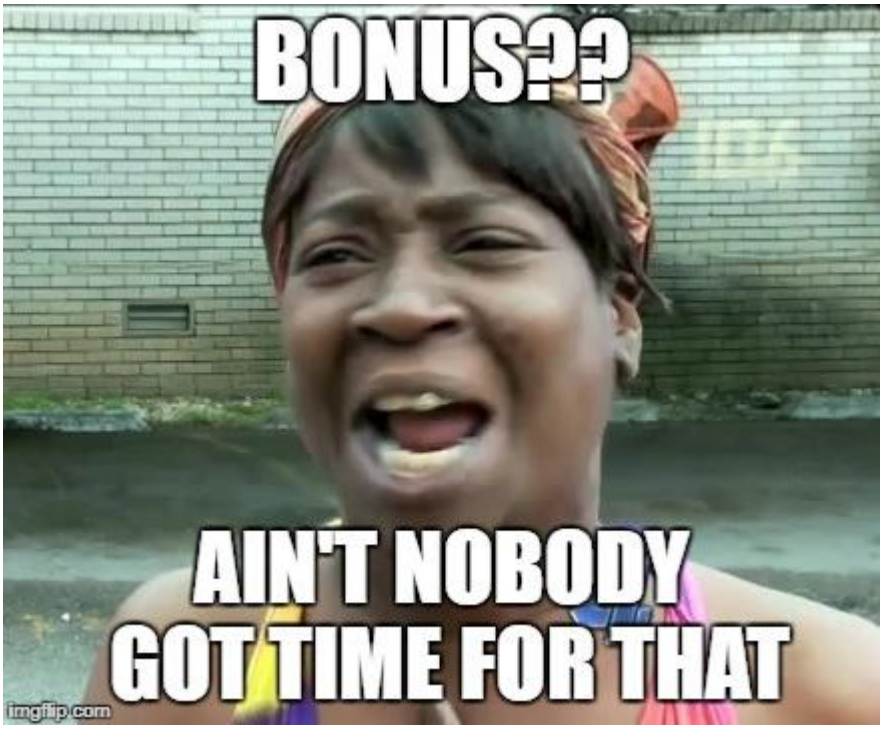
קיבלנו שיפור ביצועים כאשר השתמשנו בבחירת מאפיינים עבור שיטת הwrapper, עבור שיטת הembedded קיבלנו שיפור לפעמים כן ולפעמים לא (תלוי בבחירת 75% מהמידע לאימון) – בהשוואה בין 2 השיטות כאשר בחרנו את אותו סט אימון קיבלנו כי שיטת הwrapper שיפרה את הדיוק ואילו שיטת הembedded פגעה בו, אם היינו מנסים לבצע embedded עם סף גיזום קטן יותר יתכן והיינו מקבלים תוצאות טובות יותר בהתבסס על סט האימון המדובר.

שאלה 10:

יתרון של wrapper methods על embedded הוא שאנו מנסים לוודא כי בחרנו את המאפיינים הטובים ביותר על ידי אימון האלגוריתם והשוואת התוצאות. עם זאת יש מקרים בהם אנו מפספסים כמו בתרחיש המתואר בבונוס שאלה 7 מאחר והחיפוש שלנו לוקלי.

חסרון של wrapper methods על embedded הוא זמן הריצה – אנו מאמנים את האלגוריתם מספר רב של פעמים לעומת אימון אחד החוסך בפיתוח צמתים ולכן שלב האימון והלמידה בwrapper methods משמעותית איטי יותר.

בונוס תחרות:



סוף סמסטר ויש עוד קורסים עמוסים, נאלץ לוותר על הבונוס ☹.