בינה מלאכותית- תרגיל רטוב 3- דו"ח המטלה

מגישים:

שקד מירום, 305258550

דור חוגי, 200952232

**חלק 1 - מדידת ביצועי המסווג**

1. מימשנו ואימנו עץ החלטה מבוסס ID3 כפי שנלמד בכיתה.
   1. מדד דיוק יחיד: כפי שנלמד בכיתה, על מנת לקבל מדד דיוק יחיד נחשב את הממוצע של 'הדיוקים' (accuracies) שהתקבלו. **הערך הממוצע שקיבלנו לאחר ביצוע ה- cross validation הוא דיוק של 72.1%.**

נציין, כי כאשר מבצעים k-fold cross-validation כדאי להתבונן גם בסטיית התקן של תוצאת הדיוק. זאת, מפני שלעיתים נעדיף מסווג עם דיוק מעט נמוך יותר, אך עם סטיית תקן קטנה, על מנת להבטיח את הצלחתו במרבית המקרים.

* 1. ישנן מספר דרכים לייצר מטריצה אחת מ-k ה confusion matrices שנוצרות בתהליך ה- cross validation. פתרון אפשרי אחד הוא לסכום את המטריצות. פתרון זה טוב מספיק, מאחר ואיננו מתעניינים באופן כללי במספרים האבסולוטיים שבמטריצה, אלא רק ביחסים ביניהם. כמובן שניתן לנרמל ערך זה על מנת שסדר הגודל הכולל לא יתנפח (באופן פרופורציוני ל -k), או לעשות ממוצע. במקרה זה בחרנו לסכום את הערכים. להלן המטריצה שהתקבלה:

|  |  |  |
| --- | --- | --- |
| False | True | actual/ predicted |
| 73 | 68 | True |
| 237 | 45 | False |

1. שימוש במטריצת בלבול לבחירת מסווג:
   1. במונחים של מטריצת הבלבול, ערך Positive מתייחס למעבד תקין, וערך Negative מתייחס למעבד תקול. נפרט את משמעות המושגים של ה confusion matrix בהקשר הזה:
      * True positive - מעבד תקין שהמסווג הכריז עליו כתקין
      * True negative - מעבד תקול שהמסווג הכריז עליו כתקול
      * False positive - מעבד תקול שהמסווג הכריז עליו כתקין
      * False negative - מעבד תקין שהמסווג הכריז עליו כתקול

לאחר התיקון של השאלה ב-FAQs, נרצה למזער את ה-False Negative, מפני שעלותו היא הגבוהה ביותר, מפני שכאשר המסווג קובע כי מעבד תקין הוא תקול, אז החברה מפסידה 9000 ש"ח, כלומר את שווי המעבד פחות עלות בדיקתו לפני הוצאתו לשוק). לעומת זאת, כאשר המסווג קובע כי מעבד תקול הוא תקין (False Positive), אז החברה מפסידה רק 1000 ש"ח, כי היא מבצעת בדיקה מיותרת.

* 1. נחשב את הרווח הצפוי כתוצאה מהשימוש בכל מסווג. נשים לב כי החברה מרוויחה רק במקרה של True Positive (כי אז המעבד מסווג כתקין, נבדק וגם נמכר), והחברה מפסידה רק במקרה של False Positive (כי אז המעבד מסווג באופן שגוי כתקין, ולכן נבדק ולא נמכר). במקרים האחרים לא מורצת בדיקה וגם המעבד לא נמכר, אז אין עלות ואין רווח. בנוסף, ניתן לראות כי הרווח ממכירת מעבד תקין הוא 9000 (10000 למכירת המעבד פחות עלות הבדיקה), וההפסד מבדיקה מיותרת הוא 1000. להלן החישוב עבור כל מסווג:
     + מסווג A: רווח:

* + - מסווג B: רווח:

* + - מסווג C: רווח:

לכן, לפי החישוב שלעיל, **הכי כדאי לחברה לבחור במסווג B**, כי באמצעותו החברה תמקסם את הרווח שלה.

1. **ראשית, נתחיל באבחנה**: המסווג ID3 הוא **מסווג** **עקבי,** ולכן כאשר **המידע** **עקבי** (כלומר, שכל שתי דגימות זהות הן בעלות סיווג אמיתי זהה בהכרח), אז מתקבל דיוק של 100% על סט האימון **.** אולם, במקרה זה, **המידע שלנו איננו עקבי** (קיימות דגימות זהות אשר הסיווג האמיתי שלהן שונה), ולכן למרות שהמסווג עקבי אז עדיין מתקבל דיוק לא מושלם על סט האימון. כעת, לאחר אבחנה זו (אשר תסייע בהסבר התוצאות), ניגש למענה על השאלות:
   1. על מנת לגרום **להתאמת יתר**, החלטנו **לא להגביל את עומק העץ, וכך העץ יגיע לעומקו המקסימלי**. נסביר מדוע קביעת העומק למקסימום מובילה במקרה זה להתאמת יתר. במקרה של עומק מרבי, עץ ההחלטה שלנו הינו אלגוריתם **עקבי**, ולכן הוא מפתח צמתים (חלוקות של הדגימות) כל עוד הוא מסוגל לעשות זאת (כאשר כל העלים מכילים דגימות רק מאחת המחלקות או כאשר לא ניתן לחלק יותר את העלה, למשל במקרה של דגימות זהות בעלות סיווג שונה). לאחר הרצת התוצאות עם העץ בעל עומק מקסימלי ראינו כי מתקבל דיוק מיטבי על סט האימון כ-92% (כמוסבר באבחנה לעיל, מפני שהמידע אינו עקבי אז דיוק זה אינו 100%), אך דיוק משמעותית פחות טוב על סט המבחן. לעומת זאת, כאשר מגבילים את עומק העץ, אז מתקבל דיוק טוב יותר על סט המבחן (ופוחת הדיוק על סט אימון). מכך ניתן להסיק, כי העץ בעל עומק מרבי נמצא במצב של התאמת יתר.
   2. על מנת לגרום ל**תת-התאמה**, **החלטנו להגביל את העץ לעומק 1** (כלומר, העץ מבצע רק חלוקה אחת של המידע). בצורה כזו, יכולת התיאור של העץ מוגבלת מאוד, והוא לא מצליח "להסביר" בצורה נכונה את סט האימון (ואכן מתקבל דיוק נמוך על סט האימון, כפי שמוצג בסעיף הבא), ולכן מגיעים למצב של תת-התאמה. נשים לב כי הדיוק על סט המבחן השתפר (יכולת ההכללה עלתה).
   3. להלן הדיוק שקיבלנו בכל אחד מהמקרים:

|  |  |  |
| --- | --- | --- |
| **Accuracy on Test** | **Accuracy on Train** | **Model** |
| 63% | 91.2% | **Over-fitted** |
| 70% | 70% | **Under-fitted** |

נסביר את ההתאמה של התוצאות אלו להתאמת-יתר ותת-התאמה:

ניתן לראות כי במקרה של התאמת יתר, מתקבל דיוק מירבי על סט אימון (מרבי בהתחשב באבחנה שלעיל שהמידע לא עקבי), אך הדיוק על סט המבחן נמוך. בכך, תוצאות אלו קוהרנטיות עם התאמת יתר, כי המודל מתאים את עצמו יותר מדי לסט האימון (ולכן מקבל תוצאות טובות בסיווגו), אך פוגע בהכללה ולכן מקבל תוצאות לא טובות על סט המבחן.

כמו כן, ניתן לראות כי במקרה של תת-התאמה, מתקבל דיוק נמוך על סט האימון (כי המודל איננו מורכב מספיק על מנת לתאר בצורה נכונה את המושג), אך מתקבל דיוק טוב יותר (מאשר במקרה של over-fitting) על סט המבחן, כי נותרה יכולת הכללה סבירה (ולכן, ב-under-fitting סט המבחן מסווג באותו דיוק כמו סט האימון). בכך יש תיאום בין התוצאות לבין הצפוי ממקרים של התאמת יתר ותת התאמה.

**חלק 2 - בחירת מאפיינים**

**Wrapper Method**

1. מגדירים את כל תתי-הקבוצות של S כ-:
2. מספר כל תתי הקבוצות של הקבוצה S הוא: . ההסבר לכך הוא שלכל איבר קיימות שתי אפשרויות בנוגע לכל תת-קבוצה: להיכלל בקבוצה או לא להיכלל בקבוצה. לכן, ניתן לייצג כל תת-קבוצה באמצעות וקטור בינארי באורך N, כאשר כל מספר בווקטור תואם לאחד מהאיברים ב-S. במידה והמספר הינו 1, אז האיבר שייך לתת-קבוצה, ואחרת האיבר לא שייך אליה. מכאן שמספר תתי-הקבוצות שווה למספר הווקטור הבינאריים באורך N.
3. מספר כל תתי הקבוצות בגודל b של הקבוצה S הוא: . זאת, מפני שמספר תתי-הקבוצות בגודל b שווה למספר האפשרויות לבחור b איברים מתוך N, ללא חזרות.

נשים לב כי שני הגדלים שלעיל אינם פולינומיאליים ב-N (חזקה ועצרת).

1. בגלל שהשאלה לא ברורה, גם לאחר ההבהרה באתר, התייעצנו עם הסגל במיילים, והחלטנו לחלק את התשובה ל-2, בהתאם לכוונתכם:
   1. במידה והכוונה היא לבדוק את הביצועים של אלגוריתם למידה שיאומן רק על תת קבוצת המאפיינים שנבחרה, על מנת להשוות אותם לאלגוריתם למידה שאומן על קבוצת מאפיינים אחרת (למשל, כזו שנבחרה בדרך אחרת, או פשוט קבוצת כל המאפיינים), אז יש להשתמש בסט הולידציה. זאת, מפני שמדובר בבדיקת ביצועים לצורך החלטה של "היפר-פרמטר" (שההיפר פרמטר במקרה זה הוא באיזו שיטה לבחור את המאפיינים, או כמה מאפיינים לבחור או האם בכלל להשתמש בכל המאפיינים), ועל כן אם נשתמש בסט המבחן אז אנחנו עלולים לפגוע ביכולת ההכללה, כי נבצע בחירה של ההיפר-פרמטרים בהתאם לסט המבחן.
   2. במידה והכוונה היא לבדוק את הביצועים של אלגוריתם למידה שיאומן רק על תת קבוצת המאפיינים שנבחרה, לאחר שהחלטנו כי באלגוריתם זה ובמאפיינים אלו נשתמש במערכת, וזאת על מנת לחזות את הביצועים הסופיים שיהיו לאלגוריתם על מידע חדש (עם אותם סוגי מאפיינים כמובן) שעתיד להתקבל, אז יש להשתמש בסט המבחן. הסיבה לכך היא שזו מטרת סט המבחן- לחזות את יכולת הסיווג והכללה של המערכת.
2. מימשנו את sfs. המימוש מצורף.
3. להלן התוצאות:
4. השתמשנו באלגוריתם KNN עם K=5 כדי לסווג את הנתונים, ללא בחירת מאפיינים. הדיוק שהתקבל הוא **0.72** (כלומר, המסווג סיווג נכונה 72% מדגימות סט המבחן). נציין, כי דיוק זה רגיש לחלוקה האקראית בין סט האימון לסט המבחן, וממספר הרצות הדיוק נע בין 0.62 ל-0.77.
5. השתמשנו בפונקציה sfs שמימשנו על מנת לבחור את 8 המאפיינים המועילים ביותר עבור ה-KNN (עם K=5). הדיוק שהתקבל הוא **0.77** (כלומר, 77% מדגימות המבחן סווגו נכונה). זאת, עם אותה החלוקה לסט אימון וסט מבחן אשר הביאה את ה-KNN ללא בחירת מאפיינים לדיוק של 0.72. כלומר, כצפוי, **בחירת המאפיינים אפשרה לשפר את ביצועי המסווג**.

**שאלת בונוס:**

נציע מקרה שבו המאפיינים הרצויים הם , למרות שהתועלת של גבוהה יותר, כאשר מתבוננים בכל מאפיין לכשלעצמו:

מקרה זה יתרחש כאשר השילוב בין ייתן תועלת רבה יותר מאשר כל שילוב אחר של זוג מאפיינים.

למשל:

נניח 4 מאפיינים בינאריים בנוגע לשאלה: האם אדם צפוי לחלות בסרטן ריאות.

להלן המאפיינים והתועלת הישירה שלהם (לפי המידע הקיים):

1. האם לאדם יש היסטוריה של מחלות סרטן במשפחה: אם כן, 60% סיכוי לחלות.
2. האם לאדם יש מערכת חיסונית חלשה: אם כן, 50% סיכוי לחלות.
3. האם האדם מעשן: אם כן, 40% סיכוי לחלות.
4. האם האדם לא עוסק בספורט: אם כן, 30% סיכוי לחלות.

כמו כן, נתון כי אם אדם מעשן וגם לא עוסק בספורט, אז יש לו 70% סיכוי לחלות.

לעומת זאת, במקרה זה, נתון כי כל צירוף אחר של מאפיינים לא מוסיף לתועלת המשותפת שלהם (למשל, אם לאדם יש היסטוריה של מחלות סרטן במשפחה וגם יש לו מערכת חיסונית חלשה, אז עדיין יש לו 50% סיכוי לחלות).

במקרה זה, במידה ויש לבחור זוג מאפיינים, נעדיף להשתמש ב-, ולא בכל זוג אחר, מפני שהתועלת של השילוב ביניהם היא הגבוהה ביותר.

כלומר, הם המאפיינים הרצויים.

נשים לב כי אם נשתמש באלגוריתם SFS, אז לא נמצא את השילוב הרצוי, מפני שהוא ייבחר תחילה את .

**Embedded Method**

1. להלן התוצאות:
   1. הדיוק המתקבל עבור עץ החלטה ללא גיזום הוא: **0.75** (כלומר, 75% מהדגימות של סט המבחן סווגו נכונה).
   2. אימנו עץ החלטה עם גיזום מוקדם כנדרש. הדיוק המתקבל הוא: **0.8** (כלומר, 80% מהדגימות של סט המבחן סווגו נכונה).
   3. **האלגוריתם של עץ החלטה עם גיזום מוקדם נתן דיוק גבוה יותר**. בדקנו את שני האלגוריתמים על סט המבחן, לאחר שאימנו אותם על סט האימון. וידאנו כי שני האלגוריתמים משתמשים באותה חלוקה של סט אימון וסט מבחן, על מנת ליצור בסיס זהה להשוואה.
2. על מנת להשוות בין השיטות השונות והאלגוריתמים השונים, ביצענו מיצוע של הדיוק שלהם על סט המבחן, על פני 10 חלוקות אימון-מבחן שונות. להלן תוצאות הדיוק:

KNN mean accuracy, without feature selection, is: 0.754716981132

KNN mean accuracy, with feature selection, is: 0.764150943396

ID3 mean accuracy, without pre-pruning, is: 0.735849056604

ID3 mean accuracy, with pre-pruning, is: 0.770754716981

ניתן לראות כי **קיבלנו שיפור בביצועים באמצעות בחירת מאפיינים, בשתי השיטות**. בפרט, ניתן לראות כי ה-Wrapper method הובילה לשיפור בדיוק של ה-KNN, וכי ה-Embedded method הובילה לשיפור בדיוק של ID3.

**קיבלנו תוצאות טובות יותר עבור השיטה של Embedded method, אך לא בפער ניכר לעומת ה-Wrapper method**.

1. יתרון וחסרון של Wrapper Method לעומת Embedded Method:
   1. **יתרון**: ה-Wrapper Method ניתנת למימוש כפונקציה גנרית אשר תפעל על כל מסווג, ועל כן ניתן להשתמש בה לטובת מגוון אלגוריתמי סיווג (reuse). לעומת זאת, יש להתאים את ה-Embedded Method לכל מסווג בנפרד.
   2. **חסרון**: זמן הריצה הכולל של שימוש ב-Wrapper Method (בחירת המאפיינים ולאחריה אימון) יהיה ארוך יותר בדרך כלל מהשימוש ב-Embedded Method. זאת, מפני שב-Wrapper Method מבצעים פעמים רבות אימון של המסווג עם קבוצות הולכות וגדלות של מאפיינים, ורק לאחר מכן מאמנים את המסווג עם סט המאפיינים הנבחר. לעומת זאת, ב-Embedded Method מבצעים את בחירת המאפיינים תוך כדי תהליך האימון (כמובן שחסרון זה תלוי באופן המימוש של כל אחת מהשיטות).

**תפקידי הקבצים בתיקייה שהגשנו:** הקובץ היחיד שהוספנו למעט הקבצים שנדרשנו באופן ישיר לספק הוא הקובץ parse\_data.py. קובץ זה מכיל את ה API הדרוש לקריאת קובץ ה CSV ובתוך כך ביצוע ה Parsing.