**אז אילו אלגוריתמים יש ל"תחרות"**

שילוב הפרמטרים והתכונות הטוב ביותר(אולי גם המקום השני במקרים מסוימים) ילכו לניסוי בקבוצת המבחן עצמה.

1. עץ החלטה בודד-לא ממש מיועד לשימוש ישיר אלא יותר עבור שימוש במסווגי boost. יש כמה עצים שהולכים סביר אבל כנראה overfit גדול. אין חשיבות לנרמול.

הולך טוב יותר משמעותית כאשר ממושקל.

הכי טוב בלי סינון תכונות:

0.011465824869612944

סינון תכונות ל50% הטובות משפר ביצועים! הכי טוב נצפה ב50% תכונות:

0.015529765155652629

1. יער עצי החלטה אקראיים-האלגוריתם לפי הספר. זמני ריצה גדולים, נראה שיחסית לא שמרן... אין חשיבות לנרמול. נבדק עם sampler בגלל זמני הריצה.

הכי טוב בלי סינון תכונות:

0.02780840991367306

הורדת תכונות פוגעת בדרך כלל, אבל נקודתית(בגלל הsampler ו"מזל") ב90% תכונות נמצאה תוצאה אחת טובה יותר:

0.030661064425770358

1. עצים אקראיים מאוד-האלגוריתם לפי הספר. מצליח פחות טוב מיער אקראי רגיל, אין חשיבות לנרמול, לא שמרן. זמני ריצה גדולים.

הכי טוב ללא סינון תכונות:

0.009139547710976274

סינון תכונות פוגע מאוד, כמה שיותר תכונות-עדיף. נראה שמשיג תוצאות פחות טובות מאשר יער רגיל מעניין אם הסיבה היא בגלל הsplit או בגלל הbootstrap.

1. רגרסיה לינארית: האלגוריתם הפשוט עובד לא רע, אין משמעות רבה לנרמול –כנראה כי אפשר גם בתוך האלגוריתם, ואין קורולציה טובה לגבי מה עדיף ואיפה בדיוק.

אין הבדל גדול מבחינת תכונות, אבל הכי טוב עם כל התכונות:

-0.023552165622446226

הכי טוב עם 76 פיצ'רים(נקודת המקסימום אם כי אין הבדל גדול)

-0.021077762619372484

ridge-מצליח טיפה יותר טוב, עובד יותר טוב עם נרמול שלנו. נראה שאין לרוב חשיבות גדולה לsolver מבחינת דיוק אלא אולי מבחינת זמן ריצה אבל כולם די מהירים.

באופן כללי די יציב.

הכי טוב בינתיים עם כל התכונות:

-0.01570103655210042

עם 90% תכונות נראה יתרון קל בפרמטרים הטובים ביותר, למרות שבאופן כללי עדיף 100% תכונות. הכי טוב:

-0.015139116202946033

יש לציין שאין הבדלים גדולים מבחינת התכונות.

elastic net-מצליח יפה, נרמול שלנו משפר מאוד את הביצועים. לא מעט פרמטרים, אבל כולם נבדקו.

נראה שהפחתת תכונות עוזרת לביצועים:

הכי טוב בינתיים עם כל התכונות:

-0.011520261082404133

עם 50% תכונות:

0.00959749796029372

\*לא\* ניתן להשתמש בהם לboosterים בגלל המימוש הייחודי.

1. רגרסיה לוגיסטית: יש 6 תת אלגוריתמים. משקול עוזר מאוד, ונרמול חובה. משום מה דווקא המקרים שאין התכנסות נותנים תוצאות עדיפות-אולי העלאת מספר האיטרציות תעזור למרות שזה מגדיל מאוד זמני ריצה. לא יציב ותלוי מאוד בפרמטרים.

רק עם הפותרים sag, saga יש תוצאות טובות-השאר נוראי בכל פרמטר או מספר תכונות.

הפחתת תכונות פוגעת בביצועים מאוד.

הכי טוב בינתיים-:

כל התכונות:

0.028667776852622815

אם כי תוצאה זו חריגה וייתכן שהיא אנומליה-הפרמטרים האחרים שנותנים תוצאות טובות מגיעים לרווח של בערך 1%.

1. רגרסיה לוגיסטית לא מולטי לייבל. \*לא\* ניתן להשתמש בהם לboosterים בגלל המימוש הייחודי. נרמול הכרחי.

מאוד יציב ונותן תוצאות עדיפות בדרך כלל מאשר מולטי, אבל כחלק מהיציבות-גם המקום הראשון פחות טוב... כחלק מזה יש לציין שכל הפותרים נותנים תוצאות דומות הפעם.

בניגוד לרגרסיה מולטי-דווקא הפחתת תכונות מועילה הפעם!

הכי טוב בינתיים:

-0.00707562568008703

1. Gradient boost-על העץ המוביל שלנו, על מסווג טיפש ועל לא מסווג בכלל.. המון פרמטרים שלא כולם נבדקו, תוצאות לא יציבות שמאוד תלויות פרמטרים. מביא לתוצאות מרשימות!

זמני ריצה עצומים לכן בוצע sampler.

נראה שמהמר מאוד "שמרני".

יש לציין שהפרמטר score שלו כמובן אינו בדיוק מה שאנחנו מחפשים(הוא מחפש דיוק בסיווג נטו) וזה מוביל ל"שמרנות" וגם לכך שיותר עצים/פרמטרים שאמורים לשפר את דיוק האלגוריתם לא בהכרח מתורגמים לתוצאות עדיפות בפועל.

נפריד לשלושה סוגים:

עץ משלנו:

בגדול נראה שיותר תכונות נותן תוצאות עדיפות.

הכי טוב כל התכונות:

0.03237664473684205

מסווג אפס:

נראה שבגדול פחות תכונות עדיף!

כל התכונות:

0.04500141602945353

מרגיש כמו אנומליה רצינית, שכן זה רחוק מאוד מהמקום השני.

הכי טוב ב25 תכונות:

0.02485008328706272

על מסווג טיפש:

בגדול יותר תכונות עדיף! אבל הבדלים לא משמעותיים.

הכי טוב:

50% תכונות:

0.022993559227107257