דאטא סיינס בתעשייה

**עבודה 1**

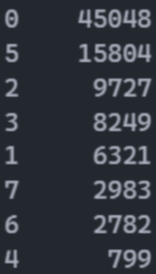
מגישים:

בן איידלין

שחר קרמר

תיאור כללי של ה-pipeline:

**הכנת נתונים**

1. החלפת ערכים חסרים- עבור משתנים רצפים החלפנו בממוצע, ועבור משתנים בדידים החלפנו בערך הנפוץ ביותר.
2. המרת משתנים מסוג STRING למשתנים נומריים.
3. הורדת פיצ'רים שאינם רלוונטים (עמודות המהוות ids)
4. ניקוי המידע-
   1. תחילה ביצענו חילוץ של טווח ההפרש בין הרבעון הראשון Q1 לרבעון שלישי Q3 עבור עמודת גיל.
   2. לאחר מכן סיננו את כל הרשומות אשר ערך הגיל שלהן גדול מהרבעון השלישי ועוד הטווח שחילצנו כפול משקל מסוים.
   3. ביצענו שינויים בנוסחה ובתנאים עד שקיבלנו את הנוסחה שהיתה לנו הכי הגיונית מבחינת התוצאות שלה, והיא-age > Q3 + 0.4\*IQR.
   4. קיבלנו כי:
      1. סה"כ נזרקו 7471.
      2. טווחי הגילאים שנזרקו נע בין 84-89.
      3. מתוך 7471 רשומות שנזרקו, ישנם 1042 מקרי ממוות (שהינם מהווים בערך 13.165% מתוך סך כלל הנפטרים ב-dataset).
   5. בסופו של דבר, קיבלנו החלטה שלא לבצע ניקוי של המידע ולא להשתמש בסינון הזה מאחר ואנחנו מאבדים אחוז יחסית גבוה של נפטרים, וגיל הוא בעיננו אחד הפיצ'רים המשמעותיים שיכול להוות סיבה הגיונית לסיווג או פיצול וקביעת מותו של חולה שהגיע למיון (הגיוני שככל שגיל המטופל עולה כך הסיכוי למותו גדל).
5. הוספת פיצ'רים-
   1. Hospital\_visit\_counts- כמות המבקרים סה"כ באותו בית חולים.
   2. Hospital\_death\_count- כמות המתים סה"כ באותו בית חולים.
   3. Precent\_of\_death\_in\_hospital- אחוז המתים באותו הבית חולים, חושב באמצעות שני הפיצ'רים מעלה (לאחר חישוב זה הם נמחקו).
   4. K-mean column- הוספנו פיצ'יר ע"י שימוש באלגוריתם זה, ככה שכמות הקלסטרים היא לוג בבסיס 4 של כמות הרשומות הקיימות (יצא 8 קלסטרים). להלן התפלגות הפיצ'ר (משמאל- מספר הקלסטר, מימין- כמות הרשומות שמופו לאותו הקלסטר).  
      
6. בחירת פיצ'רים- השתמשנו במבחן ANOVA F value ובפונקציה קיימת SelectKBest של sklearn המאפשרת בחירת פיצ'רים. סיננו את כל הפיצ'רים שה-p\_value שלהם גדול מ-0.01 . סה"כ נשארנו עם 167 פיצ'רים.

**מידול אבלואציה**

5 שיטות הקלספיקציה השונות שבחרנו הן:

1. Neural Network
2. Logistic Regression
3. Naive Bayes
4. Random Forest
5. KNN

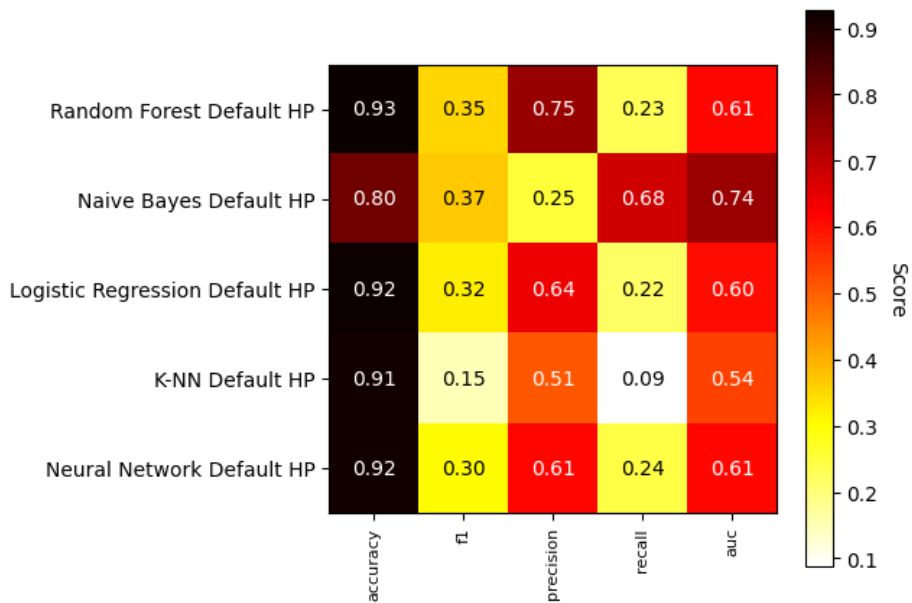
**ביצוע של 10-fold-cross-validation:**

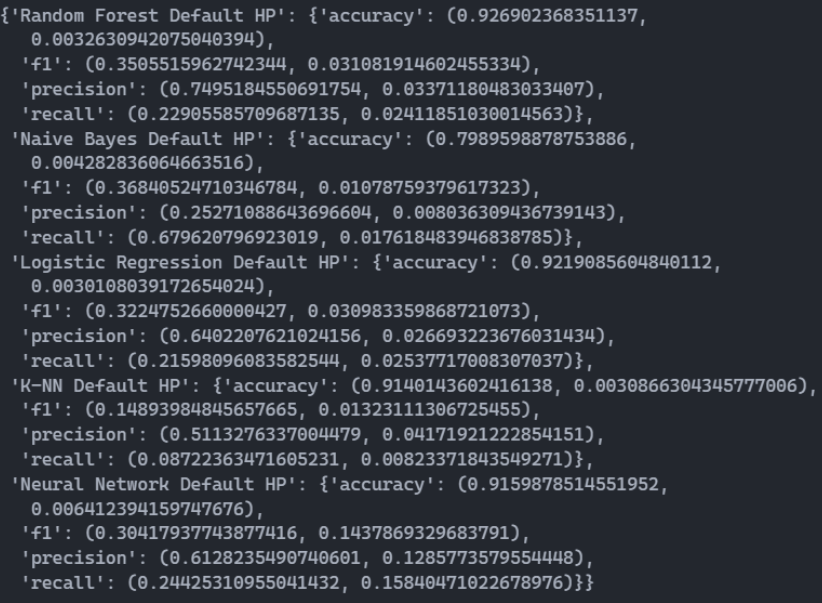
הגדרנו מחלקה הנקראת FOLD. המחלקה מייצגת fold ספציפי מתוך העשרה ומופע שלה מכיל את סט האימון והמבחן וטווחי ה-binnings. לבסוף יצרנו קובץ Pickle שישמור את אותם הfolds כדי שבמידה ונרצה להרי את הקוד מהתחלתו, כל החלק של הדסקרטיזציה (שהיה ארוך מאוד) יחסך ונקרא פשוט את הקובץ פיקל עם האובייקטים המייצגים את הfolds.

**לכל fold בוצעו השלבים הבאים:**

1. דסקרטיזציה- לכל פיצ'ר נומרי בחרנו את כמות הbins (המינימום בין 10 לכמות הערכים הייחודים) ואת שיטת החלוקה עומק שווה.
2. אימון על גבי סט האימון (תחת כל fold בנפרד) של כל אחד מחמשת סוגי המודלים.
3. בחינת המודל בעזרת סט המבחן.
4. הערכת התוצאות ע"י חמשת המדדים הבאים:
   1. Accuracy
   2. F1
   3. Precision
   4. Recall
   5. Auc

תחילה ביצענו את השלבים המתוארים מעלה עבור היפר-פרמטרים דיפולטיביים עבור כל אחד מהמודלים שבחרנו. סה"כ לאחר חישוב משוקלל של מדדי הערכה על גבי כל הfolds תחת כל אלגוריתם הניבו את התוצאות הבאות:

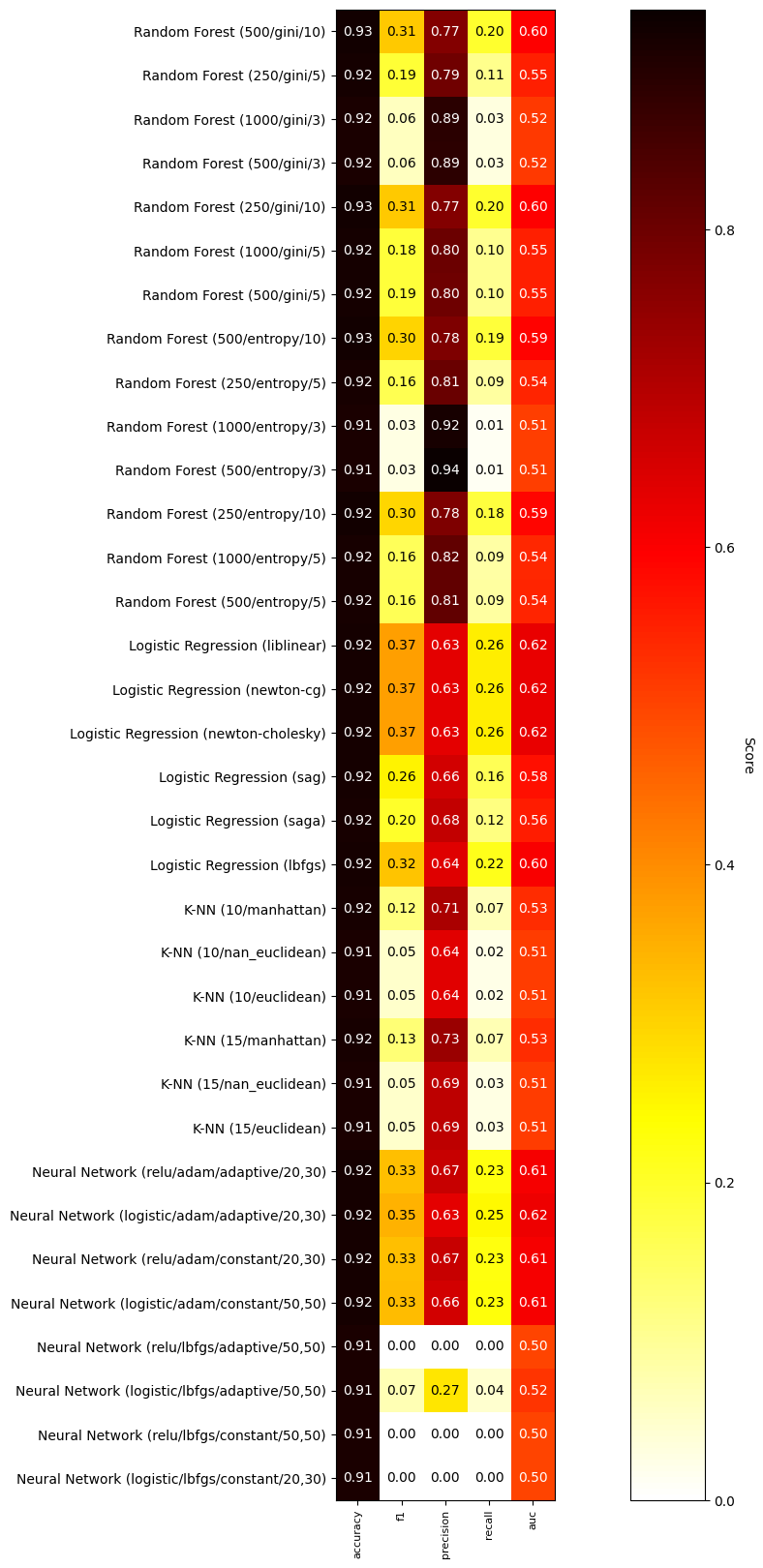




אפשר לראות שמדד הדיוק הוא מאוד גבוה בכל המדדים בזמן שמדדים אחרים די חלשים יחסית.

נראה כי Naïve base יוצא דופן- הוא היחיד שמדד הrecall שלו גבוה יבחס לשאר, ומדד הprecision שלו נמוך ביחס לשאר.

לאחר הרצה ראשונית, התחלנו "לשחק" עם ההיפר-פרמטרים של כל מודל, ובחנו את תוצאות המדדים בכל הרצה. להלן התוצאות:



**בחירת האלגוריתם:**

נדגיש כי התוצאות המוצגות מעלה אינן מספקות לרמת הdeployment production אבל עושות את העבודה עבור שלב בחירת המודל.

נראה כי מבחינת דיוק, לכלל המודלים יש ביצוע דומה וכי מדד הדיוק גבוה בכולם ובטווח זהה.

בהיבט של recall אל מול precision, לדעתנו מדד הprecision הינו חשוב יותר למודל שנבחר מאחר וחשוב לנו יותר לזהות את אלו שישארו בחיים לא פחות מאשר לזהות את אלו שימותו. נסביר- נסתכל על מקרה הקיצון בו על כל חולה נחזה שימות. כלומר משמעות מקרה זה הוא recall גבוה. בהנחה והמודל שלנו יעזור לרופאים במיון להחליט איך לחלק את תשומת הלב שלהם, עדיף להפנות את המשאבים למקרים בהם הסיכויים לחיות גדולים יותר. לכן מקרה הקיצון המתואר פחות עדיף במקרה זה, ונרצה שהמודל פחות יחמיר ויוכל לחזות גם במקרים בהם החולה לא ימות.

בנוסף, אנחנו גם יותר נתחשב במדד הAUC כמדד חשוב מאחר והוא מעריך בצורה טובה עד כמה המודל מצליח להבדיל בין רשומות שהם positive ל-negative.

זאת ועוד, מדד f1 פחות יהווה גורם משמעותי בהחלטת המודל מאחר והינו שיקלול של מדדי הrecall והprecision ונותן לכל אחד מהם משקל דומה, וכפי שהזכרנו למעלה אנחנו רוצים לתת חשיבות גבוהה יותר למדד הprecision.

תחילה נבחר עבור כל מודל את ההיפר-פרמטרים האופטימלים:

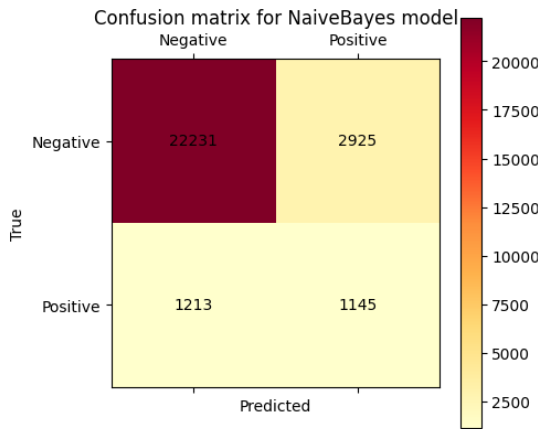
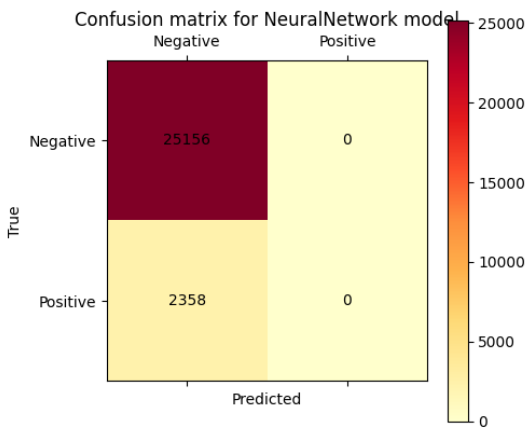
1. **Neural Network**:
   1. הפרמטר solver עם הערך lbfgs הניב את התוצאות הכי נמוכות (0 בכל המדדים).
   2. שאר תוצאות ההרצה די דומות, ולכן ניקח את קומבינציית הפרמטרים שממקסמת את הprecision (על חשבון הrecall).
   3. קומבינציית הפרמטרים הסופית- relu, adam, adaptive, 20, 30
2. **Logistic Regression**:
   1. קומבינציית הפרמטרים הסופית- liblinear
3. **Naive Bayes**:
   1. אין למודל זה היפר-פרמטרים שאפשר לשחק איתם, ולכן ניקח את תוצאותיו מההרצה הראשונית.
4. **Random Forest**:
   1. קומבינציית פרמטרים הסופית- 250, gini, 10
5. **KNN**:
   1. קומבינציית פרמטרים הסופית- 15, manhattan

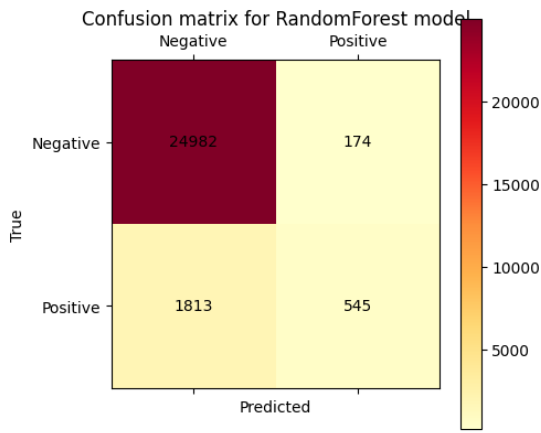
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **ACC** | **F1** | **PRECISION** | **RECALL** | **AUC** | **Run time in sec** | **Run time std** |
| **ANN** | 0.92 | 0.33 | 0.67 | 0.23 | 0.61 | 104.47 | 80.32 |
| **LR** | 0.92 | 0.37 | 0.63 | 0.26 | 0.62 | 31.912 | 48.89 |
| **NB** | 0.798 | 0.36 | 0.25 | 0.67 | 0.74 | 0.124 | 0.023 |
| **RF** | 0.93 | 0.31 | 0.77 | 0.2 | 0.6 | 45.296 | 22.456 |
| **KNN** | 0.92 | 0.13 | 0.73 | 0.07 | 0.53 | 107.842 | 65.701 |

המודלים שבחרנו להמשיך איתם הם Naive Bayes, Neural Network ו-Random Forest.

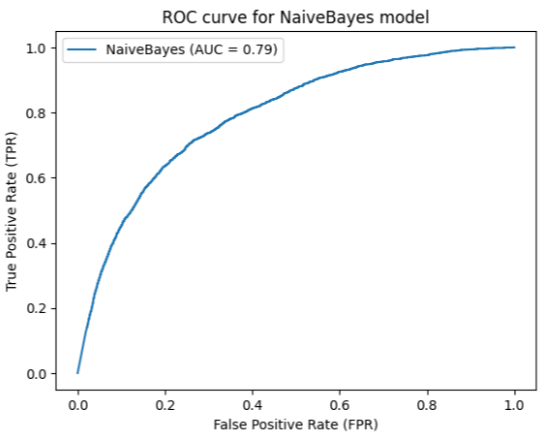
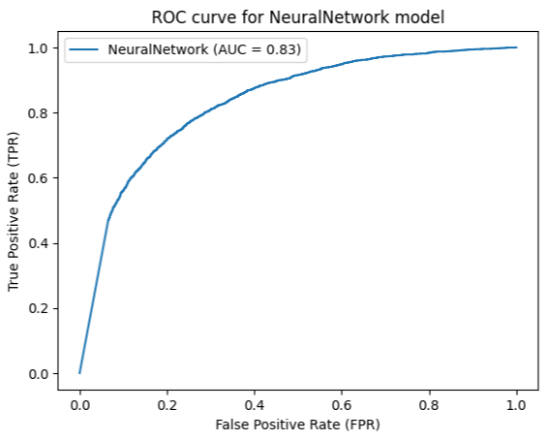
עבור כל אחד מהמודלים ביצענו הרצה נוספת ללא חלוקה ל-folds, והגדלנו משמעותית את כמות הbins בשלב הbinning כיוון שראינו שדבר זה משפר את תוצאות המודלים. בהרצה זו החלוקה לסט אימון ומבחן על גבי על הדאטא סט נעשתה רנדומלית כך ש-70% מהרשומות היו תחת סט האימון והשאר תחת סא המבחן.

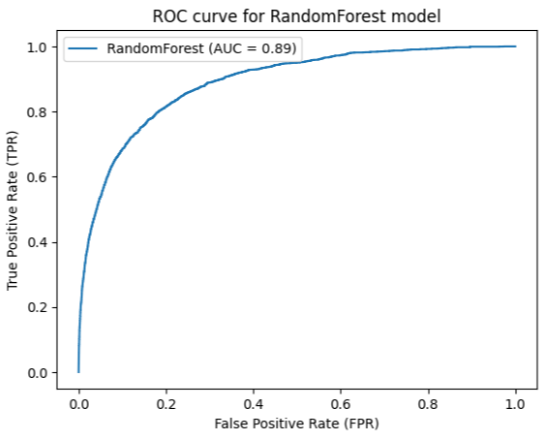
**confusion matrix:**

1. Naive Bayes  
   
2. Neural Network  
   
3. Random Forest



**ROC curve:**

1. Naive Bayes  
   
2. Neural Network  
   
3. Random Forest



**אימון המודלים הנבחרים ותוצאות הערכה החיצונית**

הרצנו את שלב הpre-processing על הנתונים הנמצאים בקובץ הunlabeled (בצורה שבה הפעלנו את אותה חוקיות שעשינו על הdataframe המקורי, גם על הקובץ הלא מתוייג).

ביצענו אימון ראשוני כך שקובץ האימון הוא כל הדאטא סט שעבר pre-processing ולאחר אימון כל אחד מהמודליםת ביענו פרדיקציה עבור הקובץ unlabeled.

את התוצאות העלנו ל-kaggle ולהלן ציוני ה-auc שהעניק:



זמני הריצה:

|  |  |
| --- | --- |
|  | **Run time in sec** |
| **ANN** | 91.62 |
| **NB** | 1.41 |
| **RF** | 72.86 |

לאחר מכן הגדלנו את קובץ האימון פי 3 ובחנו את המודלים. להלן התוצאות:



ראינו כי יש שיפור בכל המודלים ולכן בחרנו להגדיל את קובץ האימון פי 10 ולאמנם מחדש. להלן התוצאות:



זמני ריצה

|  |  |
| --- | --- |
|  | **Run time in sec** |
| **ANN** | 512 |
| **NB** | 7.52 |
| **RF** | 1577.1 |

ניתן לראות כי עבור מודל הNB הגדלת קובץ האימון לא שינתה בהרבה (לא להגדלה של פי 3 או 10).

לעומתו, מודל הRF בעל ביצועים גבוהיים יותר בהגדלת סט האימון פי 3, אך בעל ביצועים פחותים יותר בהגדלה פי 10.

עבור המודל ANN כל הגדלה של קובץ האימון שיפרה את הביצועים.

ניתן לראות כי המגמה עקבית למידי עם ההערכה הפנינמית חוץ מכך שNB סיפק את תוצאת הAUC הטובה ביותר בהערכה הפנימית ואת הגרועה ביותר בהערכה החיצונית (הציון שניתן מקגל).

**נעשה ניסיון לבצע אנסמבל המשלב את תוצאות שלושת המודלים יחדיו.**

נעשו שלושה נסיונות כאלו:

1. באופן נאיבי נתן משקול שווה לכל אחת מהתוצאות של המודלים  
   
2. משקול תוך התחשבות בתוצאות הערכה חיצונית- כלומר מודל בעל הערכה חיצונית גבוהה יותר קיבל משקול גובהה יותר ולהפך (לדוג'-RF קיבל את הציון הטוב ביותר)  
   
3. האנסמבל השלישי והמורכב יותר- מודל הNB בעל ציון מדדי recall וauc הגבוהים ביותר, ובעל ציון precision הנמוך ביותר. משמעות הדבר הזה בעיננו היא שמודל זה מרבה לחזות מוות של חולה ולכן הFP שלו גבוה, אך מנגד כאשר המודל חוזה כי החולה לא ימות, חיזוי הזה הוא חזק יותר מאשר החיזוי של שאר המודלים, כלומר הTN שלו גבוה. לכן ניסינו לבצע משקול לפי התיאור הבא- עבור דגימת חולה, כארש NB נתן חיזוי שבו החולה לא ימות, נתנו משקל גבוה יותר עבור NB לעומת שאר המודלים, ולהפך- כאשר NB חזה כי החולה ימות נתנו לו את המשקל הנמוך ביותר.



מאחר ותוצאות האנסמבל של הנסיון הראשון והאחרון לא יוצאות דופן באף אחד מהניסיונות ואף נמוכות יותר, זנחנו את הרעיון להשתמש בו.

הניסיון השני שמשקלל את כל אחד מהמודלים במשקל שונה לפי ההערכה החיצונית, נתן ביצועים דומים מאוד למודל הRF שהוא בעל הביצועים הגבוהים ביותר, ולכן השערתנו היא שיכול להיות כי בעולם האמיתי הוא יכול להיות אפילו יותר טוב.

**סיכום ותובנות**

בפרויקט הזה ניסינו להשתמש בגישות שונות כלפי הדאטא מבחנת מורכבות הניתוח שעשינו. כבר מלכתחילה בשלב האבלואציה (10 fold cross) בחרנו מודלים מרמות מורכבות שונות- מרמה פשוטה יותר (כמו naïve bayse) ועד לרמה מורכבת (כמו רשת נוירונים).

גם כאשר המשכנו לשלב הבא של הניתוח (מבחנים מול הדאטא הלא מתויג והערכה חיצונית ע"י Kaggle) השתמשנו בשלושה מודלים ברמות שונות, שלכל אחד יתרונות וחסרונות בהיבטים שונים- לדוגמא המודל הפשוט ביותר שבחרנו NB בעל מדדי הערכה הכי נמוכים מלבד מדד הערכה AUC שקיבל את הציון הגובה ביותר משאר המודלים. בנוסף, מודל זה בעל הביצועים המהירים ביותר (זמן אימון נמוך משמעותית). זאת ועוד, במודל הRF קיבלנו את התוצאות הטובות ביותר מבחינת המדדים אך מצד שני הוא בעל זמן ריצה הארוך ביותר. בסופו של דבר בחירת שלושת המודלים נעשתה משיקולים שונים שיתחשבו ברמות מורכבות שונות תוך בחירה פנימית של הביצועים הטובים ביותר פר מודל על גבי הרצות שונות עם קומבינציות של היפר-פרמטרים.

בשלב האחרון רצינו לחשוב על שיטות שונות לאמן את המודלים שבחרנו עם ההיפר-פרמטרים האופטימלים עבור כל המודל, כדי לקבל את ההערכה החיצונית הכי מוצלחת. תחילה, הרצנו הרצת אימון ראשונית ונאיבית לכל אחד מהמודלים עם הסט מבחן הלא מתויג, ואת התוצאות העלנו ישירות להערכה חיצונית ע"י Kaggle. ראינו כי התוצאות אכן לא מרשימות במיוחד ומכאן פתחנו בשלב אופטימיזציה של תהליך האימון ע"י שיפור הן כמות ה-binnings והן גודל סט האימון, ואף ניסיון לביצוע אנסמבל המשלב את שלושת המודלים הנבחרים.

הגדלת כמות הBins שיפר בצורה דרמטית את ביצועי המודלים. הגדלת סט האימון שיפר באופן חלקי (תלוי במודל ותלוי בכמות ההגדלה). ביצוע אנסמבל היה דומה בתוצאותיו לתוצאות של המודל הטוב ביותר שהוא הrandom forest (בניסיון אנסמבל ספציפי).

לסיכום, למדנו כי המודל המורכב הוא לא תמיד הכי טוב והמורכבות צריכה להיות תואמת לדאטא עצמנו ולדומיין עצמנו. בהקשר שלנו, הינו בוחרים להמשיך עם מודל הrandom forest ולהמשיך לבחון את האופציה לאנסמבל.