רוני מינדלין מילר 302242870
עמית מגן 206348005

למידה חישובית – עבודה 1

בחרנו לעשות את העבודה בשפת python.

שינינו במחלקה Tree של האלגוריתם RandomForestClassifier בחבילה scikit learn את הפונקציות fit לצורך חיזוי על פי Naive Bayes, ואת הפונקציה ו-predict_proba כדי להחזיר חיזוי בהתאם.

בקובץ Tree יצרנו שני מילונים – אחד עבור X והשני עבור Y, כך שעבור כל עלה בעץ נשמור במילון את הרשומות (הערכים וה-label שלהם במילונים X ו-Y בהתאם) שהגיעו לאותו עלה. המטרה ביצירת מילונים אלה היא שבסוף בניית העץ נדע את התפלגות הרשומות בכל עלה, ונוכל לבנות על פיהם מודל naïve Bayes מתאים. הערה: גילינו איזה רשומות הגיעו לכל עלה באמצעות הפונקציה apply על סט הרשומות שהתקבל ב-fit (X).

לאחר מציאת התפלגויות הסיווגים בכל אחד מהעלים, יצרנו מודל naïve Bayes לכל אחד מהעלים, ושמרנו מודלים אלה במילון גלובלי הממפה בין עלה למודל. על פי מילון זה בהמשך נבצע את הסיווג.

בפונקציה predict_proba ביצענו את השינוי הבא:

כעת בכל פעם שהתבקשנו לבצע סיווג עבור סט רשומות X, עברנו על כל רשומה ובדקנו לאיזה עלה היא מגיעה בעץ, וכך ידענו איזה מהמודלים להפעיל על הרשומה. ניגשנו למודל המתאים במילון המודלים על פי האינדקס של העלה, וביצענו סיווג לרשומה באמצעות מודל זה.

<u>שיפור ביצועי המודל:</u>

כדי לשפר את ביצועי ה-random forest כאשר אנו מחזיקים מודלי naïve Bayes בעלים, שינינו בעיקר את הפרמטרים שקשורים לכמות הרשומות שמגיעות לעלה, כיוון שעבור מודל naïve Bayes חשוב שיגיעו מספיק רשומות מכל מחלקת סיווג על מנת שהמודל ילמד כמו שצריך.

עבור כל פרמטר בחרנו את הערך שנראה לנו המתאים ביותר עבור מודל naïve Bayes. על מנת לבנות את המודל הטוב ביותר ביצענו ניסיונות אמפיריים לקומבינציות של פרמטרים לאלגוריתם במטרה למצוא את הקומבינציה שמניבה לנו את רמת הדיוק הגבוהה ביותר.

נשתמש בסימונים הבאים:

N – מספר ה-samples ב-database.

C – מספר ה-classes ב-database

F – מספר הפיצ'רים.

להלן הפרמטרים ששינינו במהלך ניסיונותינו:

1. Max_feature –מספר הפיצ'רים שלפיו נפצל כל node משפיע במודל ה-random forest על רכיב הרנדומליות של העץ. התחלנו עם הערך הדיפולטיבי (שורש מספר הפיצ'רים).
הערכים שהזנו הינם מתוך האפשרויות שהמודל יכול לקבל. להלן המשמעות של כל אחד מהם:
   - Auto – שורש מספר הפיצ'רים במודל.
   - Log2 – log2 של כמות הפיצ'רים במודל.
   - None – התחשבות בכלל הפיצ'רים.

רוני מינדלין מילר 302242870
עמית מגן 206348005

2. Min_sample_split – מספר הרשומות המינימלי שצריך שיגיעו ל-node על מנת שיהיה ניתן לפצל אותו. בחרנו להתחיל עם הערך N/2*F כיוון שנרצה שערך זה יהיה פרופורציונלי לכמות הרשומות ב-database ולכמות הפיצ'רים של המודל. ככל שיש יותר פיצ'רים נרצה שיהיו יותר פיצולים ולכן נהיה פחות קשוחים בהחלטה האם לפצל או לא. פרמטר זה שולט בכמות הרשומות שמגיעות לכל עלה ולכן עשוי לשפר את ביצועי המודל.
3. Mean_sample_leaf – כמות הרשומות המינימלית שצריכה להיות בעלה בעץ. בחרנו להתחיל עם הערך C*10 כיוון שבמודל ה-Naïve Bayes שנבנה בכל אחד מהעלים, נרצה שלכל עלה יגיעו מספר רשומות מכל מחלקת סיווג. פרמטר זה שולט על כמות הרשומות שמגיעות לכל עלה ולכן עשוי לשפר את ביצועי המודל.
4. N_estimators – כמות העצים שנבנים כחלק מה-forest. התחלנו עם הערך 100.
5. Max_depth – העומק המקסימלי של העץ. גם פה נרצה שיהיה תלוי בכמות הפיצ'רים במודל. כאשר הפיצ'רים הם בינאריים ניתן לומר כי העומק המקסימלי של העץ אמור להיות Log(F). כיוון שהפיצ'רים לא בהכרח יהיו בינאריים ב-dataset התחלנו עם הערך Log(F)+1 ועלינו בהדרגה. פרמטר זה שולט בכמות הרשומות שמגיעות לכל עלה ולכן עשוי לשפר את ביצועי המודל.

לכל אחד מהפרמטרים ניסינו לתת ערכים שונים וראינו איך הדבר משפיע על מידת הדיוק. כיוון שלאלגוריתם יש רכיב רנדומי לא נוכל להסתפק בתוצאות של הרצה אחת (כיוון שיכול להיות שבמקרה תצא לנו הרצה לא מייצגת), ולכן חישבנו את דיוק המודל ע"י הרצת 10-fold cross validation ומיצוע של התוצאות המתקבלות.

בכל איטרציה שינינו ערך של פרמטר אחד בלבד, מצאנו את הערך שממקסם את הדיוק והמשכנו איתו.

להלן פירוט ההרצות שביצענו עבור כל אחד מהפרמטרים:

Mean_sample_leaf	Max_depth	N_estimators	Min_sample_split	Max_feature	accuracy
C*10	Log(F)+1	100	N/(2*F)	Auto	0.74
C*10	Log(F)+1	100	2*N/F	Auto	0.76
C*10	Log(F)+1	100	3*N/F	Auto	0.74
C*10	Log(F)+1	100	1.5*N/F	Auto	0.74
C*20	Log(F)+1	100	2*N/F	Auto	0.77
C*25	Log(F)+1	100	2*N/F	Auto	0.79
C*30	Log(F)+1	100	2*N/F	Auto	0.74
C*25	Log(F)*2	100	2*N/F	Auto	0.76
C*25	Log(F)*1.5	100	2*N/F	Auto	0.77
C*25	F	100	2*N/F	Log2	0.8
C*25	Log(F)*1.5	100	2*N/F	Log2	0.78
C*10	Log(F)+1	100	2*N/F	Log2	0.78
C*10	Log(F)+1	100	2*N/F	None	0.93
C*25	F	150	2*N/F	Log2	0.81
C*25	F	300	2*N/F	Log2	0.8
C*25	F	50	2*N/F	Log2	0.7
C*25	F	100	2*N/F	Auto	0.82
C*25	Log(F)+1	100	2*N/F	None	0.91

רוני מינדלין מילר 302242870
עמית מגן 206348005

מכלל הניסיונות שביצענו ערכי הפרמטרים שהניבו לנו את מידת הדיוק הגבוהה ביותר הם:

Mean_sample_leaf	Max_depth	N_estimators	Min_sample_split	Max_feature	accurancy
None	C*10	Log(F)+1	100	2*N/F	None

כעת נציג את ביצועי החיזוי של המודל שלנו בהשוואה לתוצאות החיזוי של המודל המקורי על 10 datasets שונים. נבצע הרצה אחת עבור שני האלגוריתמים ללא הפרמטרים ששינינו והרצה שנייה עם הפרמטרים ששינינו.

להלן תוצאות הריצה עם הפרמטרים שבחרנו:

dataset	Our algorithm	Original algorithm	The best algorithm
Chess	0.912	0.934	original
Audiology	0.74	0.304	Our
Balance-scale	0.78	0.73	Our
Balloons	1	0.5	Our
Breast-cancer	0.61	0.78	original
Car	0.81	0.74	our
Connect-4	0.28	0.67	Original
Hayes-roth	0.7	0.39	Our
lymphography	0.8	0.63	our
Mushroom	0.84	0.98	Original

להלן תוצאות הריצה עם הפרמטרים הדיפולטיביים:

dataset	Our algorithm	Original algorithm	The best algorithm
Chess	0.8	0.98	Original
Audiology	0.48	0.75	Original
Balance-scale	0.62	0.8	Original
Balloons	0.78	0.92	Original
Breast-cancer	0.74	0.72	our
Car	0.78	0.95	our
Connect-4	0.62	0.79	original
Hayes-roth	0.68	0.88	original
lymphography	0.65	0.82	original
Mushroom	0.8	1	original

ניתן לראות כי לאחר הוספת הפרמטרים ברוב המקרים המודל שלנו מצליח לקבל ביצועים טובים יותר מאשר המודל של ה-majority vote. בנוסף במרבית מה-datasets רמת הדיוק עלה כאשר השתמשנו בפרמטרים שאנחנו בחרנו ולא בפרמטרים הדיפולטיביים.