דו"ח סופי- פרויקט גמר ביולוגיה חישובית

שם הסטודנט: אבישי ויזל. מנחה: פרופ' סול עפרוני

הפרויקט: בחינת מודל לשימוש ברפרטואר של תאי T לצרכי אבחון

תוכן עניינים:

תקציר הפרויקט

הפרויקט עוסק במציאת מודל המסוגל לחזות תגובה לטיפול אימונותרפי, בהינתן רפרטואר תאי ה- T (המורכב מרפרטואר TRA, ורפרטואר TRB) מפציינט, טרום טיפול.

השלב המקדים של הפרויקט כלל ריצוף רפרטואר תאי ה- T של עכברים מושרי GBM, Alignment, מציאת הפפטידים הייחודים לכל עכבר, וחישוב 31 תכונות ביוכימיות של הפפטידים היחודיים.

בהמשך, בוצע עיבוד נוסף של נתוני הריצוף, עיבוד ראשוני בעזרת אלגוריתם K-Means לקבלת דאטה סט שמתאים ללימוד מכונה, והכנסת הדאטה סט ל- database של SQL.

בשלב הבא, הוספנו פיצ'ר חדש שהתמקד בתכונה ביוכימית מסוימת, הידרופוביות הפפטיד ונוסו מספרי אלגוריתמים ללמידת מכונה:

1. Perceptron- אלגוריתם הפרדה לינארית.
2. SVM- מפריד ליניארי אשר יוצר מרווח גדול ככל האפשר בין דוגמאות מקלאסים שונים עבור דוגמאות האימון המיוצגות כווקטורים במרחב ליניארי.
3. KNN- מסווג דוגמה לפי K הדוגמאות הקרובות ביותר לפי מרחק אוקלידי.
4. Random Forest- מסווג ע"פ מספר גדול של עצי החלטה בלתי תלויים.
5. XgBoost- בונה random forest כך שכל עץ ישפר את עצמו בעזרת העצים הקודמים.

בשלב האחרון נבחנו ביצועי האלגוריתם השונים, תוך התייחסות לשאלות:

1. האם הדאטה למיד.
2. מהו האלגוריתם האופטימלי ללימוד.
3. האם יש הבדל בין הרפרטואר של TRA ו- TRB.
4. האם הוספת פיצ'ר ההידרופוביות תרם או פגע בלמידה.

מבוא – רקע ביולוגי

מערכת החיסון היא מערכת שתפקידה להילחם בפולשים חיצוניים, לנטרל רעלים ולחסל גידולים סרטניים [1].

מערכת החיסון מורכבת מסוגי תאים שונים, לדוגמה תאים דנטרידטים, מקרופאג'ים ותאי B. מקובל לחלק את מערכת החיסון לשניים- מערכת החיסון הטבעית, ומערכת החיסון הנרכשת, ולכל מערכת תאים ייחודיים משלה. למשל, תאי T משתייכים למערכת החיסון הנרכשת. תפקידה של מערכת החיסון הטבעית הוא להוות קו ההגנה הראשון של הגוף נגד פתוגנים. בהיותה קו ההגנה הראשון, עליה להיות מסוגלת לפעול במהירות כנגד כל פתוגן שנכנס לגוף על כן, הזיהוי שלה הוא לא ספציפי ופחות יעיל [1].

מערכת החיסון הנרכשת נכנסת לפעולה בשלב מאוחר יותר, אבל הזיהוי שלה הוא ספציפי ולכן יותר יעיל.

תאי T משתייכים למערכת החיסון הנרכשת. תפקידם לזהות פתוגניים בעזרת קולטן הנקרא TCR - T cell receptor. TCRs הם קולטנים הנמצאים על גבי תאי T ותפקידם לזהות פתוגן, הנמצא בתוך תא קיים. התא מגיש כלפי חוץ רצף שמייצג את הפתוגן הנמצא בו, בעזרת קומפלקס חלבוני הנקרא MHC. למרבית התאים בעלי גרעין תא יש MHC. ה- TCR על פני תאי ה- T הוא זה שמתחבר ל-MHC ומזהה את הפתוגן[2].

כפי שהזכרנו, הזיהוי של תאי מערכת החיסון הנרכשת, הוא ספציפי. כדי שתאים יוכלו להיקשר לפתוגן באופן ספציפי, הם צריכים לעבור של "ארגון מחדש" (Rearrangement). בתהליך הזה, כל תא עובר סדרה של תהליכים רנדומליים, אשר בסופם מתקבלים תאים שונים המסוגלים לזהות פתוגנים שונים[3]. תהליך הארגון מחדש מערב רקומבינציה של גנים מסוג V, D ו-J (קיימים גנים שונים מכל סוג) המרכיבים יחד את הקולטן הבשל (רקומבינציית V(D)J). משערים שבתהליך הזה, יש פוטנציאל ליצירה של בין 10^{15} ל 10^{20} TCRs שונים. זהו מספר גדול בהרבה ממספר תאי הגוף, ולכן רק חלק קטן מה- TCRs אכן מיוצג בגוף. מחקרים הראו שרק 10% מרצפי ה- TCRs משותפים בין בני אדם ו-90% מהרצפים הם ייחודיים לאדם מסוים ומהווים "טביעת אצבע" חיסונית. יחד נקרא מגוון רצפי ה- TCRs, רפרטואר תאי ה-**T** [4]. הספציפיות נתרמת הודות לתהליך הארגון מחדש החל על קולטני ה –TCR כך שכל תא T מזהה פתוגן ספציפי, בעזרת ה - TCR שלו השונה בין תא לתא. השוני ברצפי ה- TCR תורם לגיוון ויוצר למעשה סוגים שונים של תאי T [3]. העובדה שמגוון המוטציות והפתוגניים הוא עצום מכתיבה למעשה את הצורך ביצירת מגוון אדיר של TCR .

ההתפתחות בטכנולוגיות הריצוף שהושגה בעשור האחרון, מקנה לנו כלים ללימוד מעמיק של מאפייני הרפרטואר. טכנולוגיית Rep-seq [5] מאפשרת לנו למדוד את כלל רצפי ה- TCRs מתוך דיגום של רקמה/דם ולאפיין את השונות והדינמיקה של מערכת החיסון בין בני פרטים שונים, על פני אינדיקציות רפואיות שונות, על פני ציר הזמן או תחת טיפול.

במחקר הזה ננסה לעמוד על הקשר בין רפרטואר ה- TCR לבין התגובה של אורגניזם לטיפול אימונותרפי נגד סרטן. טיפול אימונותרפי הוא טיפול שמתמקד בהפעלת מערכת החיסון נגד פתוגן [6].

המחקר יתמקד בטיפול אימונותרפי נגד סרטן **גליובלסטומה** (Glioblastoma-GBM) [7] מסוג anti PD1 [8]. התאים הסרטניים מתחמקים ממערכת החיסון בעזרת דרכים שונות. אחת הדרכים היא ביטוי של ליגנד הנקרא PD-L1, ונקשר לקולטן נוסף על פני תאי T, הנקרא PD-1. קישור

זה גורם לתאי ה-T להיות פחות פעילים, ומסייע לתאי הסרטן לחמוק ממערכת החיסון [9].
כיום מפותחות תרופות נגד סרטן, המונעות את הקישור בין הליגנד PD-L1 לקולטן PD-1 ובכך משפעלות את תאי ה-T עד לחיסולו של הגידול הסרטני. אחת הדוגמאות הנפוצות כיום כטיפול בקליניקה, היא שימוש בנוגדן Anti-PD1 המכוון כנגד הקולטן PD-1. הודות לקישור הנוגדן לקולטן, ימנע הקישור של הקולטן לליגנד שלו ותאי ה- T יוכלו להיות במצב משופעל.
טיפול ב- Anti PD-1 לא בהכרח מביא לריפוי; למעשה מחקרים מדווחים על אחוז הצלחה קטן של תגובה, בקרב חולי גליובלסטומה (פחות מ- 8%) [10] מכאן, המוטיבציה לאתר קשר בין רפרטואר תאי ה- T ובין התגובה לתרופה ככלי חיזוי טרום טיפול. באופן זה ייחסך סבל מיותר מהחולים וימנע מתן טיפול לא יעיל. כלי חיזוי המבוסס על מערכת החיסון יוכל למעשה להכוון טיפול אישי (Precision medicine).

מבוא- רקע חישובי

למידת מכונה (Machine Learning) היא תחום במדעי המחשב, שעוסק בפיתוח אלגוריתמים, שנותנים למחשב את היכולת ללמוד מתוך דוגמאות.
תוכנה שמבוססת על למידת מכונה תוכל לקבל מספר דוגמאות לאימון ("סט אימון"), ולאחר מכן לחזות דוגמה שאיננה חלק מסט האימון.

דוגמה מורכבת מוקטור (או מערך) של נתונים. כל וקטור מורכב מיחידות שנקראות "פיצ'רים".

למידת מכונה מסווגת לשני תחומים עיקריים[11]:

Supervised Learning – למידה מפוקחת : כאשר החיזוי של הדוגמות האימון נתון לאלגוריתם.

Unsupervised Learning – למידה לא מפוקחת: כאשר החיזוי של הדוגמות האימון אינו נתון לאלגוריתם.

הבעיות העיקריות שניתן לפתור בעזרת למידת מכונה הם [12]:

Regression – חיזוי של ערך מספרי. למשל: חיזוי גודל הרפרטואר של פרט.

Classification – סיווג הדוגמה לאחת מכמה קטגוריות נתונות. למשל: . נתון פרט, וידוע שהוא שייך לאחת מהקטגוריות "מגיב" או "לא מגיב". הקטגוריות נתונות לאלגוריתם, והוא יכריע לאיזה קטגוריה הפרט שייך.

Clustering - קיבוץ הדוגמאות לאשכולות. למשל: נתונה לנו קבוצה גדולה של משתמשים, ונרצה שהאלגוריתם יחלק אותם לקבוצות ("אשכולות"), ע"פ נתוני השימוש שלהם.

השלבים בעבודה של למידת מכונה [13]:

1. הגדרת הבעיה

מהו הקלט של האלגוריתם, האם החיזוי יהיה נתון (Supervised/Unsupervised Learning), וכיצד יבוצע החיזוי (Regression/ Classification/ Clustering).

2. איסוף נתונים (**Data Collection**)

איסוף הנתונים הנדרשים. ניתן למצוא במסדי נתונים שונים, או לבנות באופן עצמאי את בסיס הנתונים.

3. ניקוי וניתוח הנתונים (**Data analysis**)

מעבר על הנתונים וזיהוי נתונים חריגים או חסרים.

4. בחירת האלגוריתם

בחירת אלגוריתם ע"פ הדרישות הבאות:

- קלט מתאים
- גודל מאגר הנתונים
- אורך זמן ריצה
- רמת דיוק

5. אימון המודל (**train**)

אימון האלגוריתם שבחרנו ע"י "סט אימון". סט האימון הוא חלק ממאגר הנתונים שאספנו (בד"כ 80%). בשלב זה, האלגוריתם ילמד את הנתונים, ובסופה יוכל לתת חיזוי עבור דוגמה מסט האימון המהווה את 20% הנותר ממאגר הנתונים.

6. בחינת המודל (**test**)

הזנת האלגוריתם בנתונים שלא נבחרו לסט האימון, חיזוי שלהם, והשוואה. לתיוג הנכון כפי שנתון במאגר הנתונים.

מבוא- תיאור הניסוי המקדים:

הניסוי הוא במודל עכברי, והוא כלל 7 עכברי ביקורת ו-24 עכברים מושרי GBM שטופלו ב- anti-PD1. מכל עכבר נדגם דם (ימים 0, 7, 21, 35, 49, 63) ככל ששרד ונוטרו ממדי הגידול אחת לשבוע. סה"כ הופקו 124 דגימות.

הפקת RNA מתאי דם לבנים (לימפוציטים ומונוציטים):

מתוך דגימות הדם שנאספו במהלך הניסוי, מוצתה פאזת תאי הדם הלבנים המועשרת בין היתר בתאי T והופק RNA.

ריצוף ה-TCR ו-Alignment:

ה- RNA שמיצינו מכל העכברים על פני נקודות הזמן השונות, ישמש לבניית ספריות DNA מועשרות ברצפי ה-TCRs. ספריות אלה יעלו לריצוף על גבי פלטפורמת Miseq (אילומינה). רצפי ה- FASTQ שיתקבלו יעברו עיבוד ראשוני דרך alignment בעזרת תוכנות המותאמות לניתוח מידע גנומי שמקורו ב- TCRs (רצפים עם שונות גבוהה) כדוגמת MiXCR[14]. בשלב הבא, נוכל לאפיין הרפרטואר במתודות שונות השאולות מעולם האקולוגיה: מדדי ה- clonality וה- diversity[15] של כל רפרטואר בהשוואה לרפרטואר האחרים המשתייכים

לאותה קבוצה (מספר פרטים שונים יאוגדו תחת אותה הקבוצה, אם הדגימו אותו שיעור תגובה לתרופה). כמו כן, ניתן יהיה לחלץ רצפים ספציפיים האופייניים לקבוצה זו או אחרת, וכך נקבע איזה TCR קיימים ונוכל גם לקבוע את רמת הביטוי שלהם.

<u>חישוב תכונות ביוכימיות:</u>

לכל רצף TCR חושבו 31 תכונות ביוכימיות שונות:

['GlutamicAcid', 'Glycine', 'Serine', 'Isoleucine', 'Cysteine',
'Methionine', 'NegativelyCharged', 'Aliphatic', 'Tyrosine',
'FrameShift', 'Valine',
'PositivelyCharged', 'Asparagine', 'Lysine',
'Polar', 'StopCodon', 'MolecularMass', 'Arginine',
'length', 'Glutamine', 'Aromatic',
'IsoelectricPoint', 'Histidine',
'Proline', 'Tryptophan', 'AsparticAcid', 'Phenylalanine', 'Leucine',
'Threonine', 'Alanine', 'Hydrophobicity']

סה"כ הנתונים בתחילת הפרויקט כללו:

1. 2 טבלאות (עבור TRA ו- TRB) עם כל הרצפים הייחודיים והתכונות הביוכימיות שלהם ("טבלאות התכונות")[16]
2. 124*2 קבצי CSV עם נתוני הריצוף מכל דגימה ("קבצי הניסוי")[17].
3. metadata של העכברים[18]. הטבלה מכילה את העמודות:
   [mice id, experimental group, survive]
4. metadata של הדגימות[19]. הטבלה מכילה את העמודות:
   [sample id, mice id, time point, source]

דוגמה של טבלאות התכונות:

sequence	n	AsparticAcid	Phenylalanine	Leucine	Threonine	Alanine
CAAAASSGSWQLIF		-0.560676	-0.183178	-0.283984	-1.0459	2.55218
CAAAATSSGQKLVF		-0.560676	-0.183178	-0.283984	0.16676	2.55218
CAAADSNYQLIW		1.76447	-2.03773	-0.0839905	-1.0459	2.00389
CAAADTNAYKVIF		1.58561	-0.0405203	-1.48395	0.260041	2.88959
CAAADYANKMIF		1.76447	0.125914	-1.48395	-1.0459	3.28323
CAAAGGRNAKLTF		-0.560676	-0.0405203	-0.191679	0.260041	2.88959
CAAAGGSNAKLTF		-0.560676	-0.0405203	-0.191679	0.260041	2.88959
CAAAGMHAGAKLTF		-0.560676	-0.183178	-0.283984	0.16676	3.64876
CAAAGNTGKLIF		-0.560676	0.125914	-0.0839905	0.36887	2.00389
CAAAGSNTNKVVF		-0.560676	-0.0405203	-1.48395	0.260041	1.70866

טבלה 1: דוגמה חלקית של טבלת התכונות.

מטרות הפרויקט

1. ניתוח מאפייני הדאטה
2. הכנסת הדאטה סט ל- SQL.
3. עיבוד הדאטה סט כך שיותאם באופן המיטבי ביותר ללמידת מכונה:

המטרה היא ליצור מטריצה לייצוג העכברים באופן המתאים ביותר: כל שורה תייצג דגימה, וכל עמודה תייצג פיצ'ר של אותה הדגימה, ע"פ רצפי ה- TCR שרוצפו ממנה.

4. הוספת פיצ'ר הידרופוביות לדאטה סט:

נרצה להוסיף לרשימת 31 התכונות הביוכמיות של רצפי ה- TCR, תכונה נוספת – הידרופוביות של הפפטיד.

5. מציאת אלגוריתם, לחיזוי האם אורגניזם יגיב בהצלחה לטיפול אימונותרפי שניתן לו, טרום טיפול.

בתחילה, רפרטואר תאי ה-T של עכברים מושרי גליובלסטומה שטופלו ב- anti pd-1 יתויגו באשר להצלחת הטיפול.
לאחר מכן, האלגוריתם שנבנה, יוזן בנתוני הרפרטואר של העכברים המטופלים ובאמצעותם ינסה לחזות האם הרפרטואר הבא שנזין, יהיה מסוגל להגיב בהצלחה לאותו טיפול. לאחר מכן, נתקף את תוצאת האלגוריתם שלנו ונבחן האם הצליח לחזות נכון.

תוצאות הפרויקט

1. ניתוח מאפייני הדאטה

על טבלאות התכונות הוכן EDA (Exploratory data analysis)[20] [21]. ה- EDA נכתב בעזרת סקריפט בשפת R [22]

2. הכנסת הדאטה סט ל- SQL

הדאטה סט הוכנס לתוך Database של SQL, בעזרת sqlite3[23]. בנוסף, הוכן ERD[24] (Entity–relationship model) שיתאר את ה- database.

3. ניתוח ועיבוד הדאטה סט כך שיותאם באופן המיטבי ביותר ללמידת מכונה

שלב 1:

כל קבצי הניסוי אוחדו לשני קבצים – קובץ אחד עבור TRA והשני עבור TRB.
לאחר מכן נוקה הדאטה כך שכל שורה תכיל:

א. רצף ח"א (aaSeqImputedCDR3)
ב. רצף נוקלאוטידי (nSeqImputedCDR3)
ג. הדגימה ממנה הרצף הגיע (sample_id)
ד. אחוז ה- clone (cloneFraction)

האיחוד והניקוי נעשו ע"י סקריפט ב- UNIX [25]. דוגמה חלקית מהתוצאה סופית:

Samples union example

1	cloneFraction	nSeqImputedCDR3	aaSeqImputedCDR3	sample_id
2	0.005475106	TGTGCACTCATAACAG	CALITGNTGKLIF	1
3	0.002486824	TGTGCAGCAAGTGCAG	CAASADTGANTGKLTF	1
4	0.001698818	TGTGCAGCAAGGCCGA	CAARPTNSAGNKLTF	1
5	0.001637415	TGTGTGGTGGGGGATA	CVVGDRGSALGRLHF	1
6	0.001576012	TGTGTGGTGGGCGATA	CVVGDRGSALGRLHF	1
7	0.001350867	TGTGCAGTGAATTATAA	CAVNYNQGKLIF	1
8	0.001289464	TGTGCAGCTAGTGAGC	CAASEPGTGGYKVVF	1
9	0.001115489	TGTGTGGTGGGTGATA	CVVGDRGSALGRLHF	1
10	0.00106432	TGTGCTATGAGAGAGA	CAMRENMGYKLTF	1

טבלה 2: דוגמה חלקית של טבלת כל הרצפים, והדגימה ממנה הם רוצפו.

שלב 2:

בחלק הראשון נירמלנו את הנתונים. הנורמליזציה בוצעה ע"פ שיטת Z-Score normalization. את הנרמול עשינו בעזרת החבילה sklearn, שבה יש אפשרות מובנית לנורמליזציה ע"פ Z-Score [26].

בחלק השני, ביצענו הורדת ממדים (feature reduction) לטבלאות התכונות ע"י אלגוריתם K-Means[27]. לכל רצף בטבלאות התכונות, ישויך קלאסטר (Cluster), וכך נוריד את הממדים מ- 31 התכונות, לתכונה אחת. הורדת הממדים נעשית מכיוון שהדאטה המלא הוא גדול ומדי, ויהיה קל יותר לעבוד עם דאטה בנפח קטן יותר. לאחר מכן, לכל רצף נוסיף את הדגימה ממנה היא רוצפה, ע"י שימוש בטבלה 1. דוגמה חלקית מהתוצאה הסופית:

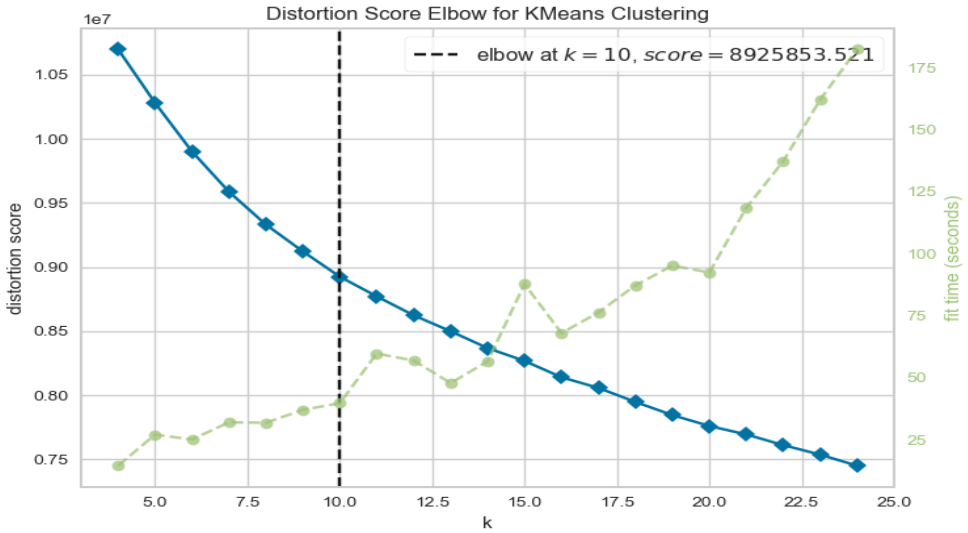
Sequence -Cluster Table example

Index	Sequence	sample_id	cluster
0	CALITGNTGKLIF	1	2
1	CAASADTGANTGKLTF	1	4
2	CAARPTNSAGNKLTF	1	0
3	CVVGDRGSALGRLHF	1	2
4	CVVGDRGSALGRLHF	1	2
5	CAVNYNQGKLIF	1	1
6	CAASEPGTGGYKVVF	1	2
7	CVVGDRGSALGRLHF	1	2
8	CAMRENMGYKLTF	1	4
9	CAVSMLAGGYKVVF	1	2

טבלה 3: הרצפים הייחודיים והקלאסטר אליו הם שויכו.

מספר ה- clusters (K) הוא היפר פרמטר שנבחר ע"י המשתמש ולכן השתמשנו בשיטת ה- elbow method[28] כדי לקבוע את K.

בשיטה זאת, בוחרים מספר cluster שונה בכל הרצה, ומודדים את המרחק של כל הנקודות למרכז ה-cluster. הקונבנציה היא לבחור K כאשר המרחק הממוצע קטן בקצב נמוך. במקרה הנוכחי נבחר k =10.



גרף 1: *Elbow method* עבור טבלאות התכונות.

<u>שלב 3:</u> יצירת טבלה, שבה לכל דגימה, יהיו עמודות כמספר הקלאסטרים, ובכל תא יהיה אחוז הקלאסטר בדגימה זו. למשל:

8% sequences from Sample 2 are cluster 1

0	1	2
2	0.0832957	0.0447436
3	0.0829283	0.0455068

תמונה 1: דוגמה חלקית לטבלת הפיצ'רים של הדוגמאות (Samples).

לאחר מכן הורדנו את הדוגמאות שהגיעו מעכברי ביקורת.

4. הוספת פיצ'ר הידרופוביות לדאטה סט

אנו מעוניינים להוסיף לרשימה תכונה שתנקד כל פפטיד על סמך מדד ההידרופוביות שלו.
אחרי סקירה ספרותית בנושא, נמצא כי קיימות מספר שיטות למדידת הידרופוביות, ביניהם לפחות 7 מדדי הידרופוביות מובילים. המדדים עצמם בנויים בצורה דומה, כך שבכל מדד, יש ציון לכל אחת מ- 20 ח. האמינו, והציון הכולל של הפפטיד הוא ממוצע של כל ח. האמינו המרכיבות אותו.
בשלב הבא, היינו צריכים להחליט באיזה מדד להשתמש. בין יתר השיקולים שיילקחו בהעדפת מדד זה או אחר, ניטה לבחור מדד שיש לו פונקציה מובנית בחבילה מוכרת.
האלגוריתם והרשימה של הרצפים עם 31 התכונות שלהם כפי שנותחו במעבדה, כתובים בפייתון, ולכן חיפשנו חבילות ופונקציות מובנות בפייתון.
מצאנו שיש פונקציה מובנית בחבילת bio-python[29] שמחשבת את ההידרופוביות של פפטיד לפי Kyte-Doolittle[30], שהוא אחד מן המדדים השונים שאספנו.
ולכן, חישבנו לכל פפטיד את מדד ההידרופוביות שלו, והוספנו לטבלאות התכונות:

sequence	n	AsparticAcid	Phenylalanine	Leucine	Threonine	Alanine
CAAAASSGSWQLIF		-0.560676	-0.183178	-0.283984	-1.0459	2.55218
CAAAATSSGQKLVF		-0.560676	-0.183178	-0.283984	0.16676	2.55218
CAAADSNYQLIW		1.76447	-2.03773	-0.0839905	-1.0459	2.00389
CAAADTNAYKVIF		1.58561	-0.0405203	-1.48395	0.260041	2.88959
CAAADYANKMIF		1.76447	0.125914	-1.48395	-1.0459	3.28323
CAAAGGRNAKLTF		-0.560676	-0.0405203	-0.191679	0.260041	2.88959
CAAAGGSNAKLTF		-0.560676	-0.0405203	-0.191679	0.260041	2.88959
CAAAGMHAGAKLTF		-0.560676	-0.183178	-0.283984	0.16676	3.64876
CAAAGNTGKLIF		-0.560676	0.125914	-0.0839905	0.36887	2.00389
CAAAGSNTNKVVF		-0.560676	-0.0405203	-1.48395	0.260041	1.70866

sequence	cid	Phenylalanine	Leucine	Threonine	Alanine	Hydrophobicity
CAAAASSGSWQLIF		-0.183178	-0.283984	-1.0459	2.55218	1.4883
CAAAATSSGQKLVF		-0.183178	-0.283984	0.16676	2.55218	1.04503
CAAADSNYQLIW		-2.03773	-0.0839905	-1.0459	2.00389	0.0407604
CAAADTNAYKVIF		-0.0405203	-1.48395	0.260041	2.88959	0.842583
CAAADYANKMIF		0.125914	-1.48395	-1.0459	3.28323	0.687189
CAAAGGRNAKLTF		-0.0405203	-0.191679	0.260041	2.88959	0.0370312
CAAAGGSNAKLTF		-0.0405203	-0.191679	0.260041	2.88959	0.588982
CAAAGMHAGAKLTF		-0.183178	-0.283984	0.16676	3.64876	1.18356
CAAAGNTGKLIF		0.125914	-0.0839905	0.36887	2.00389	1.23666
CAAAGSNTNKVVF		-0.0405203	-1.48395	0.260041	1.70866	0.544229

תמונה 2: המחשה של הוספת פיצ'ר ההידרופוביות.

5. מציאת אלגוריתם לחיזוי הצלחה לטיפול אימונותרפי, טרום טיפול.

על הדאטה הופעלו מספר אלגורתמי machine learning. הסקריפט[31] נכתב בשפת פייתון ובעזרת חבילת scikit-learn[32]. בנוסף, ביצענו את אותם האלגוריתמים על דאטה סט רנדומי (control), כאשר ה- labels מציגים את אותו היחס בין responders ל- no responders כמו הדאטה סט הרגיל. נתוני הדאטה:

- 4 datasets: TRA, TRB, with and without hydrophobicity.
- 98 samples for each dataset.
- 10 features.
- Features: [0,1].
- 59.1% non-responders, 40.9% responders.

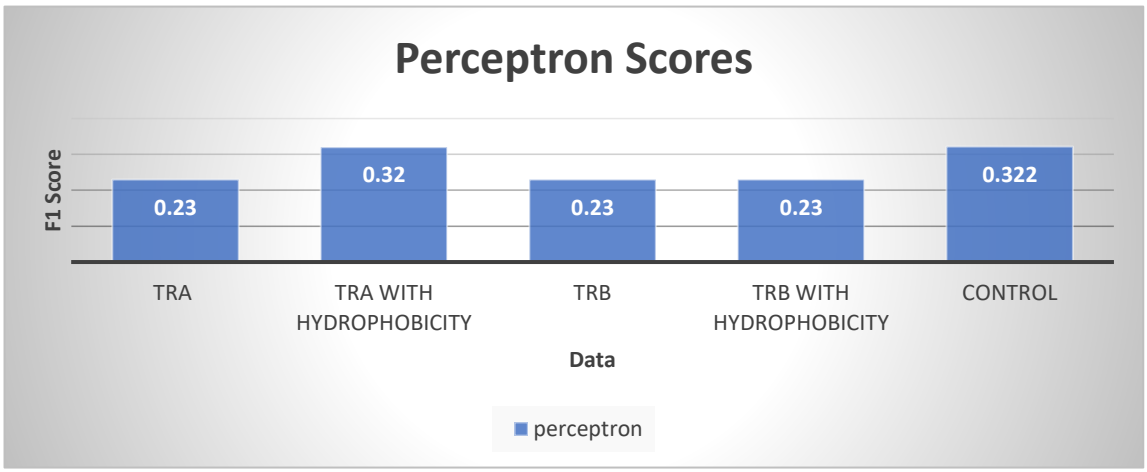
מכיון שלדאטה יש class imbalance, החלטנו להשתמש במדד הדיוק F1 Score [33]. במידה והיינו משתמשים במדד ה- Accuracy, היינו יכולים להגיע גם עם bad classifier לדיוק של כמעט 60%, אם הוא היה למשל חוזה שכל רפרטואר שהיה מוזן לו הוא non-responder (כי 60% מהדאטה סט הוא non-responders) .
היתרון במדד זה הוא השילוב בין ה- recall וה- Precision:

$$F1 = \frac{2 * precision * recall}{precision + recall}$$

כך, אפשר לזהות מקרים שבהם הדיוק נראה גבוה, אבל ה- recall או ה- precision נמוכים.

Perceptron:

פרספטרון הוא מפריד לינארי, שמוצא ישר שהוא המפריד הטוב ביותר בין הדוגמאות. תוצאות האלגוריתם:



גרף עמודות 1: תוצאות F1 Score של אלגוריתם ה- Perceptron על ה- datasets השונים.

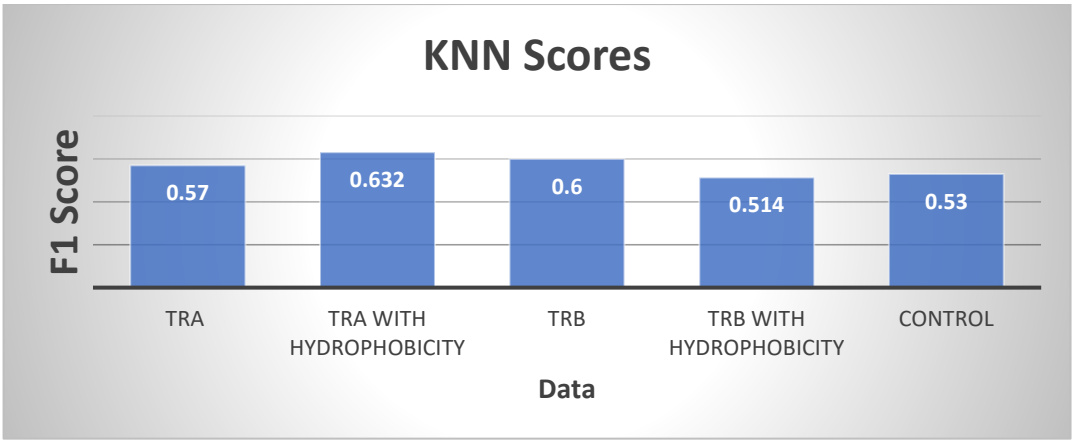
מהתוצאות נראה שלא הצלחנו לחזות את התוצאות, ולכן הדאטה לא פריד לינארית.

נשים לב שאם הציון היה מחושב לפי Accuracy, היינו יכולים "להפוך" את החיזוי ולקבל תוצאות טובות, אבל מכיון שהציון נחושב ע"ב F1 Score, אין לנו אפשרות כזאת, ולכן האלגוריתם לא מצליח לחזות.

KNN:

KNN הוא מפריד לא לינארי, שמסווג דוגמאות חדשות ע"פ מרחקם האוקלידי מדוגמאות האימון.

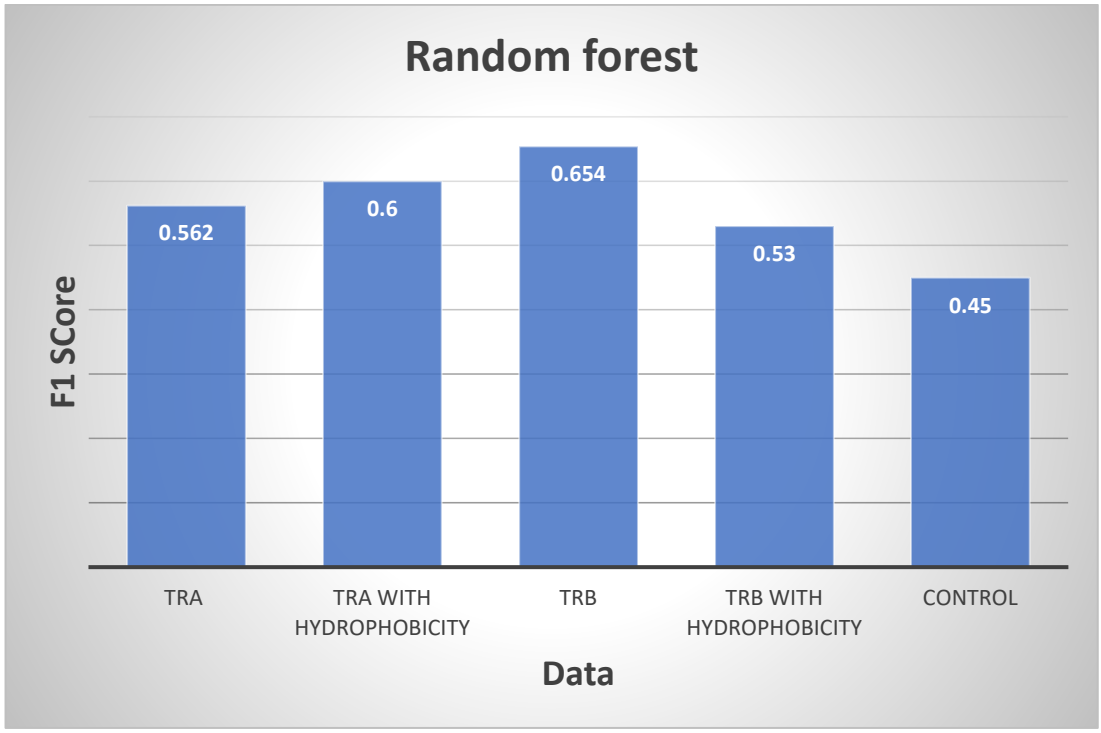
הדאטה פוצל עם stratify[34] ל- train ו- test בחלוקה של 75% ו- 25% בהתאמה. לאחר מכן האלגוריתם הורץ עבור K= [1,3,5,7,9,11], ו- 100 חזרות עבור כל K, בעזרת gridSearchCV [35] כדי למנוע תוצאות חריגות.
לבסוף, נמדד F1 Score הממוצע בין כל ההרצות:



גרף עמודות 2: תוצאות F1 Score של אלגוריתם ה- KNN על ה- datasets השונים.באלגוריתם KNN אפשר לראות שיש למידה מסוימת, בעיקר ב TRA With hydrophobicity ו- TRB.

Random forest:

Random forest מסווג ע"פ מספר גדול של עצי החלטה בלתי תלויים. הדאטה פוצל ל- train ו- test כמו באלגוריתם KNN, ובוצעו 50 חזרות:



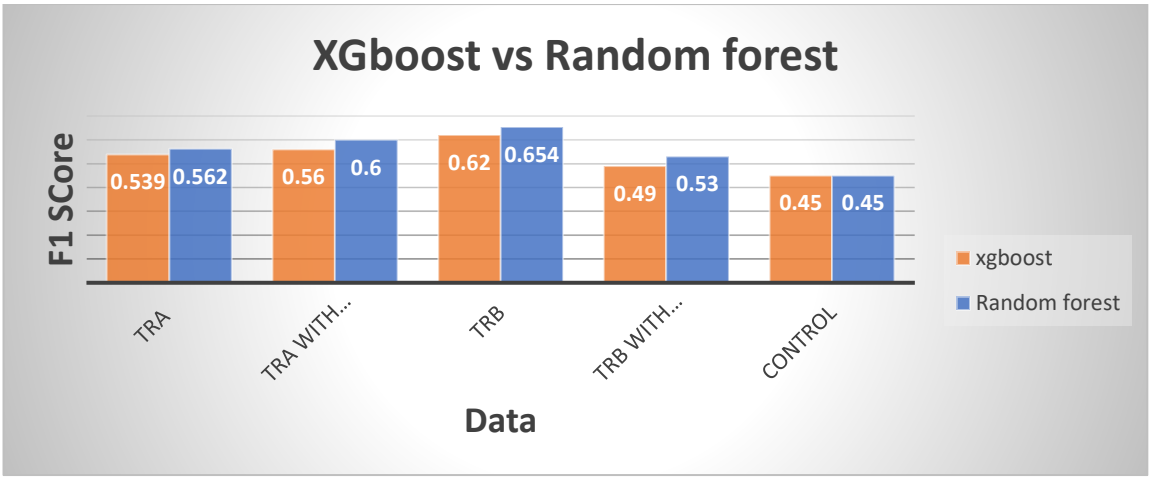
גרף עמודות 3: תוצאות F1 Score של אלגוריתם ה- Random Forest על ה- datasets השונים.גם כאן TRA עם הידרופוביות ו- TRB נלמדו במידה מסוימת, בעוד ש- TRA ו- TRB עם הידרופוביות נלמדו במידה מועטה.

XgBoost[36]:

Boosting היא שיטה לבניית random forest כך שכל עץ ישפר את עצמו בעזרת העצים הקודמים: אחרי כל עץ מחושבת פונקציית loss, והעץ הבא יבנה כך שפונקציית ה- Loss תמוזער.

XgBoost הוא אלגוריתם חדש יחסית (פותח ב- 2016), שהיווה שיפור משמעותי ל- Boosting בפרט ול- Random Forest בכלל, תוך שהוא זוכה במספר תחרויות Machine Learning.[37]

אי לכך, ציפינו שהאלגוריתם יצליח יותר מ- Random Forest. האלגוריתם הורץ עם gridSearch כדי למצוא את הפרטרים האופטימלים, ולבסוף הורץ עם אותם פרמטרים 50 חזרות.
התוצאות:



גרף עמודות 4: השוואה בין אלגוריתם XGboost ל- Random forest.

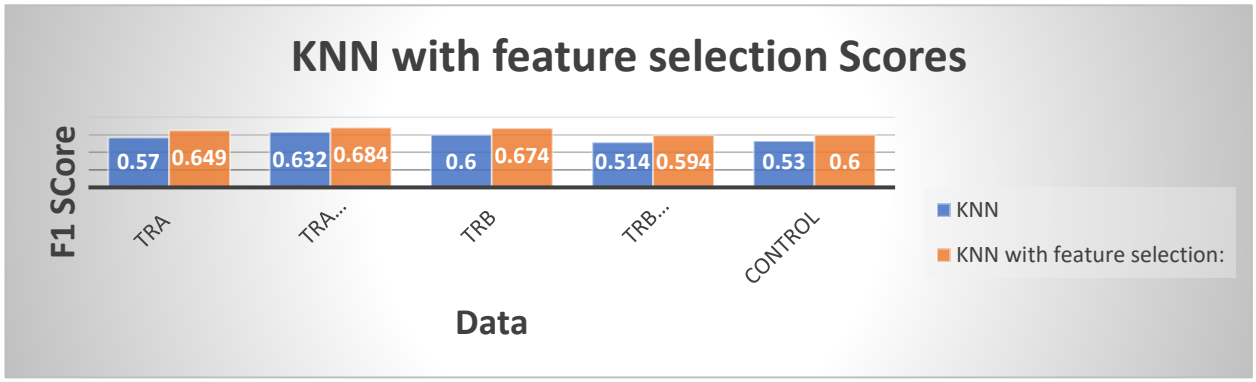
ניתן לראות ש- xgboost חוזה פחות טוב את הדאטה מאשר Random Forest. העובדה הזאת הייתה מפתיעה, ולכן הועלו שתי השערות אפשריות:

א. כיוונון ההיפר פרמטרים לא היה מספיק טוב. יכול להיות שעם יותר ניסיון והבנה של האלגוריתם, היינו מצליחים להוציא תוצאות טובות יותר.
ב. XgBoost לא מתאים ללימוד של Dataset קטנים מאוד, כמו בפרויקט זה.

בשלב הבא ניסינו מחדש את חלק האלגוריתמים, הפעם בשיטת forward feature selection.
זוהי שיטה שמשמשת ל- dimensionality reduction. בשיטה זאת, בודקים את האלגוריתם עם פיצ'ר אחד בלבד, ולאחר מכן מנסים להוסיף פיצ'ר נוסף. אם האלגוריתם הצליח יותר, נמשיך לפיצ'ר הבא עם שני הפיצ'רים. אם ההצלחה הייתה פחותה- הפיצ'ר לא יילקח.
לבסוף, נקבל תת רשימה של פיצ'רים, מתוך סט הפיצ'רים המקורי.

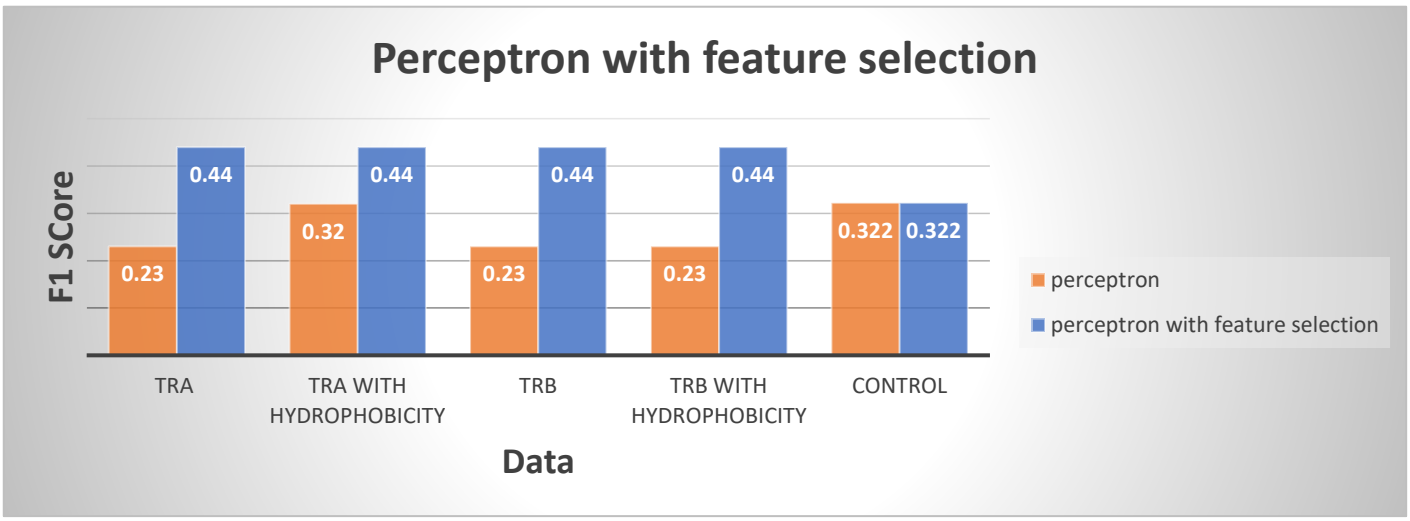
KNN with feature selection:

בתחילה, נבחר ה-K האופטימלי כמו באלגוריתם המקורי. לאחר מכן, נבצע forward feature selection, נקבל את תת הרשימה, ועליה נבצע 100 הרצות עם פיצול של הדאטה כמו באלגוריתם המקורי, וחישוב F1 Score הממוצע:



גרף עמודות 5: השוואה בין אלגוריתם KNN עם feature selection ל- KNN ללא feature selection. אלגוריתם ה- Feature selection שיפר מאוד את ה- F1 Score, בכל הdatasets. נשים לב שגם ה- Control עלה באופן משמעותי, מה שאומר שלא ניתן בהכרח להסיק שהאלגוריתם יותר טוב מאשר ללא Feature selection.

Perceptron with feature selection:

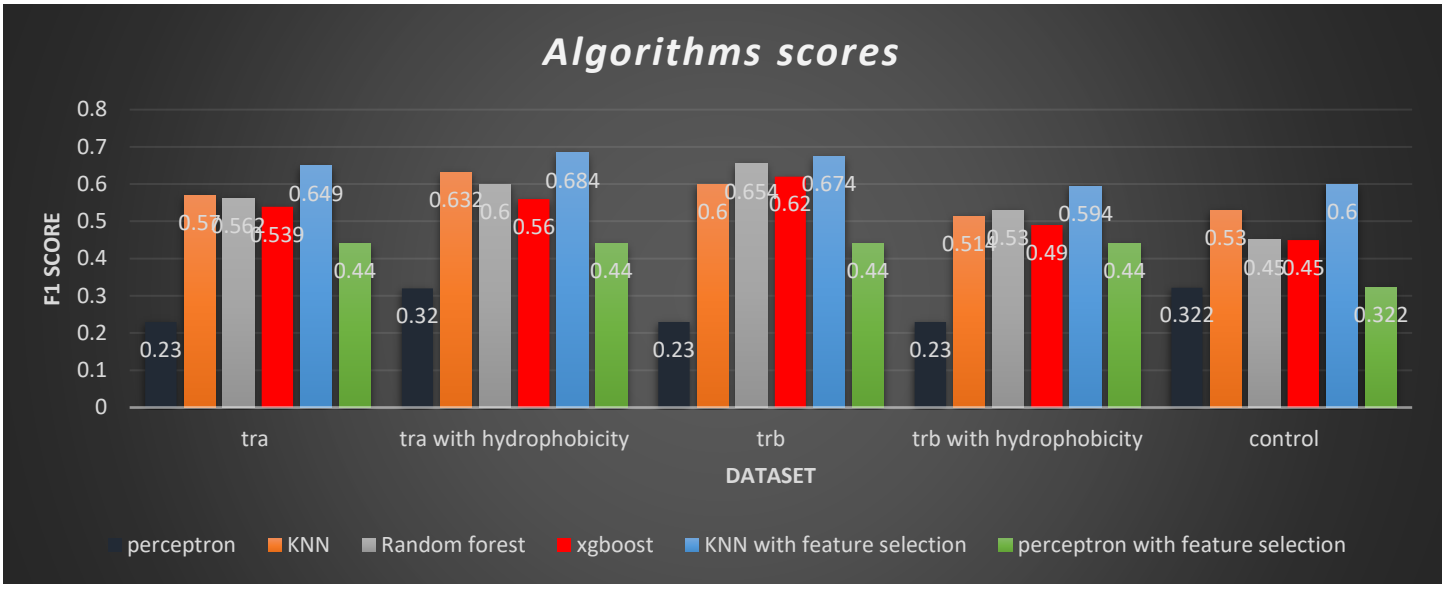


גרף עמודות 6: השוואה בין אלגוריתם ה- Perceptron עם feature selection ל- Perceptron ללא feature selection.

ניתן לראות עלייה קטנה ב- F1 Score ב- datasets שאינם control, אך עדיין התוצאות נמוכות יחסית.

דיון

Best algorithm:



גרף עמודות 7: תוצאות כל האלגוריתמים עבור כל Dataset.

ניתן לראות שהאלגוריתם שהציג ביצועים הכי טובים לפי F1- Score, בכל ה- datasets, הוא KNN with feature selection. אך עם נשווה את התוצאה עם התוצאה של ה- control, נראה שהמצב משתנה: למשל, KNN with feature selection על TRB with hydrophobicity, קיבל אמנם תוצאה גבוהה יותר מאשר Random forest, אך בהשוואה ל- control, לא היה ב KNN למידה כלל.

האלגוריתמים הכי טובים בהשוואה ל- control:

TRA: Random Forest

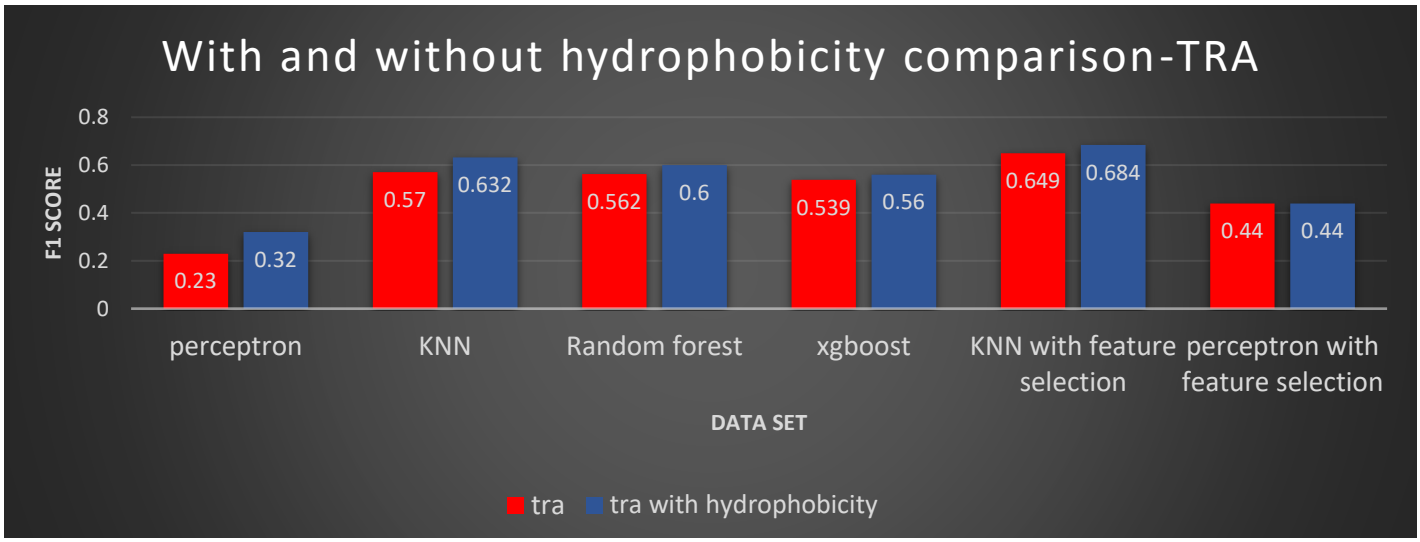
TRA with Hydrophobicity: Random Forest

TRB: Random Forest

TRB with Hydrophobicity: Random Forest

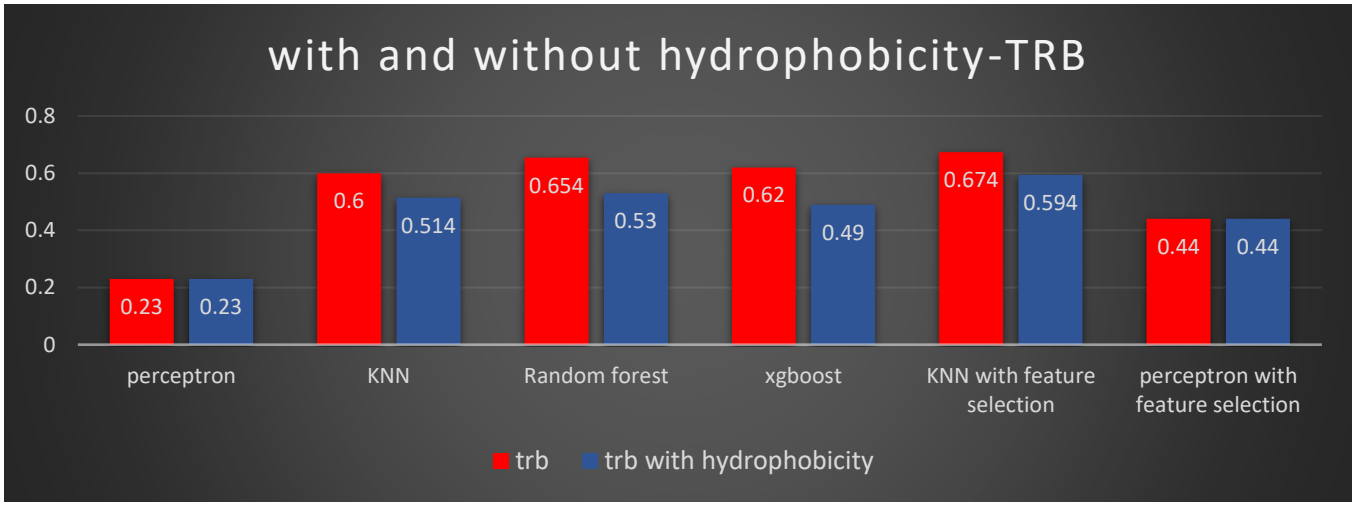
האם הוספת ההידרופוביות הועילה?

נשווה בין ה- datasets עם ההידרופוביות, ל- datasets ללא ההידרופוביות:



גרף עמודות 8: השוואה בין TRA Data ללא הידרופוביות ל-TRA Data עם הידרופוביות.

ניתן לראות שעבור כל אלגוריתם, חוץ מפרספטרון, יש עלייה ב- F1 Score. נשווה עם TRB:



גרף עמודות 8: השוואה בין TRB Data ללא הידרופוביות ל-TRB Data עם הידרופוביות.

כאן רואים בדיוק את המצב ההפוך – הוספה של הידרופוביות מורידה את ה- F1 Score בכל האלגוריתמים, חוץ מפרספטרון.

ניתן אולי להסיק מכאן שלהידרופוביות יש תפקיד מעודד או מעכב בקשירת ה- TRA אל הפתוגן, מה שתורם לנו בחיזוי האם עכבר יגיב או לא. לעומת זאת, ל TRB אין תפקיד כזה, והוספת פיצ'ר שאינו מועיל ללמידה מורידה את אחוזי ההצלחה.

סיכום

- הרפרטואר של תאי ה-T ניתן ללמידה ברמה מסוימת, בהקשר של תגובה לתרופה אימונותרפית.
- Random Forest הוא האלגוריתם שהצליח ללמוד את הדאטה באופן המיטבי ביותר
- מכיוון שהדאטה הכיל סט נתונים קטן מאוד, וישנה אפשרות של סטייה במדידות השונות. יש לבדוק את ההשערות הנ"ל על דאטה סט גדול יותר.

שיטות עבודה

שפות תכנות:

- R: ה- EDA נכתב בשפת R ע"י RStudio, כאשר הויזואליזציה נכתבה בעזרת חבילת ggplot[38].
- SQL: מסד הנתונים נכתב בעזרת חבילת sqlite3, שמאפשרת כתיבת התוכנית בשפת פיתון[39]
- UNIX: קבצי הניסוי נערכו בשפת UNIX, בעזרת תוכנת CYGWIN[40].
- Python: האלגוריתמים נכתבו בשפת פייתון על SPYDER 4.0[41] ועם החבילות Scikit-learn[42] ו- Yellowbrick[43].

ביבליוגרפיה:

[1] Owen, Punt, Stranford. (2009). *KUBY Immunology* (7 ed.), p.1, p.16 New York: W. H. Freeman and Company.
[2] Owen, Punt, Stranford. (2009). *KUBY Immunology* (7 ed.), p.247. New York: W. H. Freeman and Company.
[3] Owen, Punt, Stranford. (2009). *KUBY Immunology* (7 ed.), p.251. New York: W. H. Freeman and Company.
4 Daniel J. Laydon, Charles R. M. Bangham, Becca Asquith, "Estimating T-cell repertoire diversity: limitations of classical estimators and a new approach," Philosophical Transactions of The Royal Society B, 2015.

[5] Jennifer Benichou, Rotem Ben-Hamo, Yoram Louzoun, Sol Efroni, "Rep-Seq: uncovering the immunological repertoire through next-generation sequencing," *Immunology,* vol. 135, 2011.
[6] Owen, Punt, Stranford. (2009). *KUBY Immunology* (7 ed.), p.644 New York: W. H. Freeman and Company.
[7] lizabeth A. Maher, Robert M. Bachoo, Rosenberg's Molecular and Genetic Basis of Neurological and Psychiatric Disease (Fifth Edition), Chapter 78 - Glioblastoma, Academic Press, 2015, Pages 909-917
[8] Buchbinder, E. I., & Desai, A. (2016). CTLA-4 and PD-1 Pathways: Similarities, Differences, and Implications of Their Inhibition. *American journal of clinical oncology*, *39*(1), 98–106. https://doi.org/10.1097/COC.0000000000000239

[9] Iwai Y, Hamanishi J, Chamoto K, Honjo T. Cancer immunotherapies targeting the PD-1 signaling pathway. J. Biomed. Sci. 2017
[10] Filley, A. C., Henriquez, M. & Dey, M. Recurrent glioma clinical trial, CheckMate-143: the game is not over yet. *Oncotarget* **8**, 91779 (2017).
[11] Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, 2014, https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/, ISBN 978-1-107-05713-5 Hardback, 1.3 - Types of Learning
[12] https://machinelearningmastery.com/types-of-learning-in-machine-learning/
[13] https://cloud.google.com/ai-platform/docs/ml-solutions-overview
[14] Bolotin DA, Poslavsky S, Mitrophanov I, Shugay M, Mamedov IZ, Putintseva EV, Chudakov DM. MiXCR: software for comprehensive adaptive immunity profiling. Nat Methods. 2015 May;12(5):380-1. doi: 10.1038/nmeth.3364. PMID: 25924071.

[15] Chiffelle J, Genolet R, Perez MA, Coukos G, Zoete V, Harari A. T-cell repertoire analysis and metrics of diversity and clonality. Curr Opin Biotechnol. 2020 Oct;65:284-295. doi: 10.1016/j.copbio.2020.07.010. Epub 2020 Sep 2. PMID: 32889231.
[16] https://drive.google.com/file/d/1r9WAMRxGxM483DXXE6MfDsvBg-FrmiX7/view?usp=sharing
[17] https://drive.google.com/drive/folders/1NJYL8umAjnOgHz3hutoEDvoyFeHsUIvP
[18]https://github.com/avishai987/Micedb/blob/70b95ed6b71caeac4dd602fa5ad68305ba64f8f3/MICE.xlsx
[19] https://github.com/avishai987/Mice-db/blob/70b95ed6b71caeac4dd602fa5ad68305ba64f8f3/sample.csv
[20] https://www.stat.cmu.edu/~hseltman/309/Book/chapter4.pdf
[21] https://github.com/avishai987/Mice-db/blob/98e71e090a3396414cb326243c8ddb2620c05ef5/EDA2.pdf
[22] https://github.com/avishai987/Mice-db/blob/98e71e090a3396414cb326243c8ddb2620c05ef5/EDA_project.R
https://docs.python.org/3/library/sqlite3.html [23]
[24] https://vulms.vu.edu.pk/Courses/CS619/Downloads/CarMatch%20ERD.pdf
[25] https://github.com/avishai987/Mice-db/blob/70b95ed6b71caeac4dd602fa5ad68305ba64f8f3/unix.txt
[26] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html
[27] Hartigan, John A., and Manchek A. Wong. "Algorithm AS 136: A k-means clustering algorithm." Journal of the royal statistical society. series c (applied statistics) 28.1 (1979): 100-108.
[28] https://en.wikipedia.org/wiki/Elbow_method_(clustering)
[29] https://biopython.org/docs/1.75/api/Bio.SeqUtils.ProtParam.html, gravy() method
[30] [Kyte and Doolittle, 1982] Kyte, J. and Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. J Mol Biol, 157(1):105--132.

https://github.com/avishai987/Mice-db/blob/5b1c4462b21346429003da21abf6a31ac2302c5e/algorithms.py [31]
[32] https://scikit-learn.org/stable/
[33] https://deepai.org/machine-learning-glossary-and-terms/f-score
[34] https://towardsdatascience.com/stratified-splitting-of-grouped-datasets-using-optimization-bdc12fb6e691
[35] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html
[36] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 785–794. DOI:https://doi.org/10.1145/2939672.2939785
[37] https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions
[38] https://ggplot2.tidyverse.org/reference/ggplot.html
[39] [39] https://docs.python.org/3/library/sqlite3.html

[40] https://www.cygwin.com/

[41] https://www.spyder-ide.org/
[42] https://scikit-learn.org/stable/
[43] https://www.scikit-yb.org/en/latest/