לימוד מכונה | קבוצה 2 | פרויקט חלק ב'

Smoking

רועי עזראי 206118754

ליטל זולוטריוב 208466524

תוכן עניינים

הקדמה

בשלב הראשון ביצענו התאמות על סט הנתונים שלנו בהתאם לשינויים שביצענו בחלק א'. טיפול בערכים חסרים והתמודדות עם ערכים חריגים. בנוסף החלטנו להחזיר את משתני age, weight להיות משתנים רציפים כפי שהיו בדאטה המקורי. עשינו זאת מכיוון שראינו לנכון להתייחס אליהם כמשתנים בעלי התנהגות רציפה ולא כחלוקה עצמאית לקבוצות כפי שביצענו בשלב הראשון. בנוסף עבור משתנה blood pressure ביצענו המרה שלו מערך קטגוריאלי לDummies אשר הוא מהווה כעת משתנה אינדיקטור.

עצי החלטה

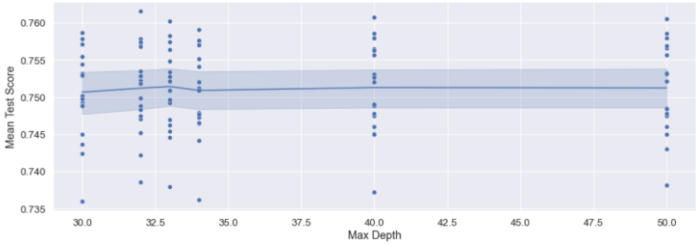
1. עץ החלטה מלא

עץ ההחלטה מכיל צמתים אשר כל צומת מכילה תצפיות בעלות ערכים מסוימים. בכדי להפריד בין המחלקות בצורה טובה, מתבצע פיצול של העץ אשר יכול להיות על פי ערך של מאפיין מסוים או סדרה של ערכים למאפיינים. סיווג תצפית הוא בהתאם למחלקה השכיחה ביותר באותו הצומת בעץ. על סמך מאפייני העץ, העץ מסביר בצורה טובה באיזה אופן ישויכו התצפיות למחלקות. בכדי לאמן את העץ חילקנו 20% מהנתונים עבור סט הבחינה (ולידציה) ו80% לסט האימון. מפני שיש לנו סט נתונים גדול החלטנו שזו תהיה הכמות הרצויה בכדי לאמן את אלגוריתם בצורה יעילה ולא לאבד מידע רב מידי. בחרנו למקסם את מדד אחוז הדיוק, אשר מהווה את סך הערכים עבורם מתקבל פסוק אמת מסך כלל הערכים. בחרנו במדד זה מכיוון שראינו משמעות גבוה בסיווג נכון ואמיתי של אדם כמעשן. מכיוון שמדובר בנתונים רפואיים והשלכות הסיווג הינן משמעותיות ישנה חשיבות רבה לדיוק גבוה. אחוז הדיוק על סט האימון הינו 100% ואחוז הדיוק על סט הבחינה הינו 76.89%. אחוז הדיוק על סט האימון נובע מהתאמת יתר, אשר כל תצפית בעץ מתאימה את עצמה לצומת בעץ בצורה מדויקת, אך לא לסט הבחינה מכיוון שאלו הן תצפיות אשר לא נחשף אליהן בסט האימון ואין ביכולתו יכולת הכללה בסיווג נכון בתצפיות אלו.

2. כוונון הפרמטרים לעץ ההחלטה

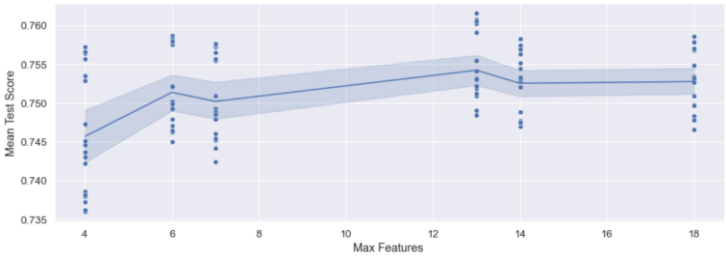
נבצע כוונון היפר פרמטרים על עצי ההחלטה על מנת למצוא את הקונפיגורציה המיטבית ולהתגבר על בעיית התאמת היתר והגדלת התאמת סט ולידציה. ביצועים מיטביים של העץ תלויים מאוד באופן כיוונון ההיפר פרמטרים שלו. על מנת לקבל סט ולידציה חזק, תחילה בחרנו את ההיפר פרמטרים בהם אנו רוצים להשתמש ובדקנו את הכוונון המיטבי של כל אחד בנפרד מתוך טווח ערכים רחב (נספח 1.01). לאחר מכן הרצנו את כוונון הפרמטרים בצורת Grid Search עבורם נבחרו הערכים שנתנו את הדיוק הגבוה ביותר עבור כל אחד מההיפר פרמטרים. Grid Search הוא טכניקת כוונון המנסה לחשב את הערכים האופטימליים של היפר פרמטרים. זהו חיפוש ממצה שמתבצע על ערכי פרמטר ספציפיים. ערכי ההיפר פרמטרים שנבחרו כפונקציה של אחוז הדיוק:

Mean Accuracy	STD Accuracy	Min Sample Leaf	Max Features	Max Depth	
0.7616	0.0062	1	13	32	1
0.7608	0.0068	1	13	40	2
0.7605	0.007	1	13	50	3
0.7602	0.0079	1	13	33	4
0.759	0.0082	1	13	34	5
0.7587	0.0097	1	6	30	6
0.7586	0.0047	1	18	50	7
0.7586	0.0047	1	18	40	8



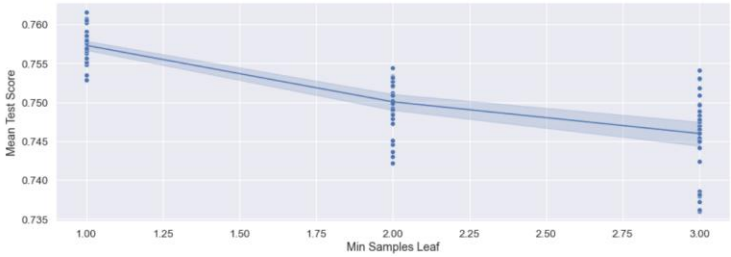
Max Depth -היפר פרמטר זה נוגע לעומק המירבי של העץ. בעבור עומק עץ גדול יותר, יתקיימו יותר פיצולים בעץ ובכל צומת פחות תצפיות. עץ עמוק מידי מעלה את הסיכויים להתאמת יתר. נשאף למצוא את הערך המתאים בכדי שהעץ יתאים בצורה הטובה ביותר עבור סט האימון וסט הבחינה. נעשה זאת על ידי קיטום בשלב מוקדם וכל צומת תייצג יותר תצפיות וכך תקטין את הסיכוי להתאמת יתר ומצד שני נשאף שעומק העץ לא יהיה קטן מידי כדי שהלמידה לא תעצור בטרם עת. תחילה בחרנו טווח ערכי העומק של העץ בצורה גנרית רחבה מ1- 50. מתוך הטווחים המיטביים שהתקבלו בחרנו את החמשת העומקים אשר נותנים את הדיוק הגבוה ביותר ועליהם הרצנו את G rid Search.

Max Features – עבור מספר מקסימלי של מאפיינים, היפר פרמטר זה מהווה את מספר המאפיינים של הפיצול הטוב ביותר, בחיפוש אחר המשתנה הטוב ביותר למקם בעץ. הפרמטר נותן אלמנט של אקראיות בבניה של העץ ועל כן מסייע בהקטנת בעיית התאמת היתר על סט האימון.



תחילה, בדקנו את הטווח הרב ביותר של מספר המאפיינים, שהוא 19 כמספר המשתנים שלנו. לאחר ההרצה ראשונית זו, לקחנו את חמשת הערכים אשר קיבלו את הדיוק הגבוה ביותר, ואותם הכנסנו לGrid Search.

Min Sample Leaf – היפר פרמטר זה מהווה את מספר התצפיות המינימלי שנדרש כדי שהוא יהווה עלה. עבור עלה אשר לא מייצג מספיק תצפיות הוא לא יהיה מספק בביצוע סיווג תצפיות חדשות (אשר לא נראו בסט האימון) והעלים לא יהיו מבוססים על בסיס הפרדה אמיתית. בעבור ערך גדול מידי של היפר פרמטר זה, העץ עלול להימנע מלמצוא הפרדות. תחילה בדקנו טווח גנרי רחב עבור בדיקה של מאפיין זה בנפרד. הטווח הינו 10 דגימות לעלה. לאחר מכן לקחנו עבור Grid Search את החמשת הערכים אשר קיבל ו את אחוז הדיוק הגבוה ביותר.



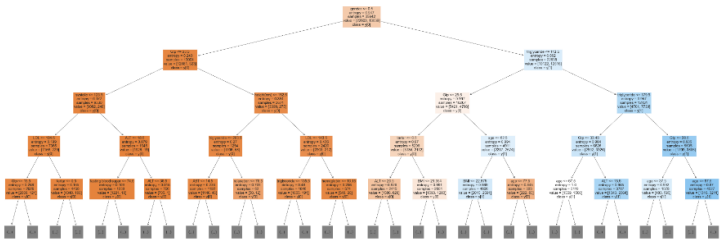
***הערה**- עבור כל היפר פרמטר, החלטנו בעבודה זו לקחת את החמשת המאפיינים בעלי אחוז דיוק הגבוה ביותר, לבדיקת Grid Search מאילוצי זמני ריצה ויכולות המעבד של המחשב האישי שלנו. שאפנו למצוא את הקומבינציות הטובות ביותר בבחירת טווחים אלו עקב מגבלות האילוצים. במידה והאפשרות הייתה קיימת היינו מגדילים את טווחי הערכים על מנת להגדיל את סיכויי המצאות קומבינציות טובות יותר.

3. אימון עץ ההחלטה

1. בחרנו את הקונפיגורציה אשר אחוז הדיוק שלה היה הגבוה ביותר (ראשונה בטבלה). כיוונון ההיפר פרמטרים מאפשר שתתבצע למידה מוכללת יותר אשר יכולה להתאים לנתונים אחרים מלבד סט האימון. אחוזי הדיוק על סט האימון: 99.87% אחוזי הדיוק על סט הבחינה:77.53%. עבור סט האימון ניתן לראות אנו נמצאים באחוז דיוק גבוה מאוד, אשר מעיד על כך שאנו עדיין נמצאים במצב של התאמת יתר, אך כעת היא לא מוחלטת ב100% בעקבות כוונון הרמטרים שביצענו. עבור אחוז הדיוק על סט הבחינה, ניתן לראות כי שיפרנו את הדיוק בזכות כוונון הפרמטרים.

2. עץ ההחלטה אשר התקבל:

(נספח 1.01) – הגדלה של התמונה לנוחות ציפייה.



3. רשומה לדוגמא מסט הבחינה:

לקחנו את הרשומה של הנ"ל והרצנו את הנתונים בעץ. הממצאים שהתקבלו:

ID	gender	age	height(cm)	weight(kg)	systolic	relaxation	fasting blood sugar	Cholesterol	triglyceride	HDL	LDL	hemoglobin	AST	ALT	Gtp	dental caries	tartar	BMI	Blood_Pressure	smoking	prediction
35783	1	30	175	90	111	77	97	177	122	61	91	15.7	40	75	40	1	1	29.3877551	Optimal	0	0

gender <= 0.5 נלך שמאלה, gtp <= 23.5 נלך שמאלה, Systolic <= 128.5 נלך ימינה, ALT <= 18.5 נלך שמאלה. ניתן לראות כי התקבל smoking = 0 אשר נותן חיזוי נכון.

4. תובנות מבנה העץ

עבור ההחלטה באיזו תכונה להתפצל בכל שלב בבניית העץ, השאיפה היא לשמור על העץ פשוט וקטן ככל הניתן. לשם כך, בכל שלב יש לבחור את הפיצול שמביא לצמתי הבת הטהורים ביותר (במדגם נבחר של מערך הנתונים כל הנתונים שייכים לאותה המחלקה). עבור כל צומת של העץ, ערך המידע מודד כמה מידע התכונה נותנת לנו על המחלקה. הפיצול עם רווח המידע הגבוה ביותר יילקח כפיצול הראשון והתהליך יימשך עד שכל צמתי הילדים יהיו טהורים, או עד שרווח המידע יהיה 0. בעץ ניתן לראות את התנאי לפיו העץ יבצע את הסיווג של כל תצפית. עבור מאפיין "entropy" אשר מודדת את האקראיות או חוסר הסדר בניבוי של המידע, הפיצול יתבצע בהתאם למזעור האנטרופיה באופן המירבי, ויתן את ההפרדה הטובה ביותר בין המחלקות. בעץ נתון ערכי samples, values אשר מציגים את מספר הדגימות ואת הפיצול בערך המטרה. בנוגע לצבעים בעץ, ככל שהצבע המתקבל חזק יותר, הסבירות שהדגימות הגיעו מהמחלקה הנ"ל תהיה גבוה יותר. הפיצולים מתבצעים בהתאם לדירוג חשיבות המשתנה. המשתנה בעל החשיבות הגבוה ביותר הינו מין הנבדק, הדבר מתיישב עם המציאות, מכיוון שבחלק א' ראינו כי למין השפעה משמעותית על הסיכוי של הנבדק להיות מעשן. המשתנה הבא הינו טריגליצרידים, שכן ראינו בחלק א' כי הקורלציה גבוה בינו לבין משתנה העישון ורמתו אכן משפיעה על סיכוי הנבדק להיות מעשן. עבור משתנה נוסף הממוקם גבוה בפיצול הינו Gtp אשר גם עבורו קיימת קורלציה גבוהה בינו לבין משתנה המטרה. נצפה שמשתנים אלו ימוקמו בגובה העץ כי הם מהווים את המסווגים הטובים ביותר.

5. Feature Importance

Feature	Importances
gender	24.22%
triglyceride	7.57%
Gtp	6.62%
fasting blood sugar	5.53%
LDL	5.51%
hemoglobin	5.50%
HDL	5.34%
systolic	5.29%
Cholesterol	5.19%
ALT	5.13%
AST	5.01%
relaxation	4.65%
BMI	4.22%
age	3.38%
height(cm)	2.07%
weight(kg)	1.98%
tartar	0.90%
Blood_Pressure_Normal	0.57%
dental caries	0.53%
Blood_Pressure_Normal High	0.31%
Blood_Pressure_Optimal	0.31%
Blood_Pressure_High	0.17%

התכונה מספקת דרך להעריך את רמת החשיבות של כל מדד בעת ביצוע החישוב. כלומר אילו מאפיינים מועילים יותר על פני האחרים. בעבור ערכים גבוהים יותר, החשיבות של אותו המאפיין בחיזוי יהיה גבוה יותר. התכונה מספקת את משמעות רלוונטיות כל מאפיין, מאפשרת לשפר את המודל על ידי קטימת הענפים עבורם התקבל מדד נמוך יותר. ניתן

לראות כי סדר המאפיינים של הרמות בעץ תואם את סדר הופעתם במדד חשיבות העץ בטבלת ה Feature Importance ולכן ניתן להגיד שהדבר אכן מתיישב עם הסעיף הקודם.

6. **דוגמאות של מאפינים בעלי חשיבות שונה מתוך פלט העץ הנלמד:**

- **gender** - משמעות מאפיין זה הינו מין הנבדק והוא נמצא בראש העץ, לכן על פי אלגוריתם עץ ההחלטה יש לו חשיבות גבוהה בצורה תואמת לטבלת ה Feature Importance. דבר זה אכן מתיישב עם האיטואיציה שלנו מכמה סיבות, אחת היא עבור סט הנתונים שלנו בחלוקה לפי מין קיבלנו שהוא אינו מאוזן לטובת הגברים (כ64% גברים לפי הניתוחים של חלק א'). בנוסף, לפי מחקר שעשינו בחלק א' עבור חלוקה זאת נראה כי אכן יש הבדל מהותי בין מעשנים בקרב גברים ונשים לטובת הגברים.

- **fasting_blood_sugar** – משמעות מאפיין זה הינו רמת הסוכר אצל הנדבק לאחר צום והוא נמצא בתחתית העץ הנלמד מהסעיף הקודם. למאפיין זה יש את חשיבות נמוכה יותר בקרב הנבדקים בעץ הנלמד, למרות שעל פי טבלת ה Feature Importance הוא נמצא בבמקום הרביעי מבין כל המאפיינים. ניתן להסיק מכך שבהתאם לטבלה חשיבותו אכן פחות ממאפיין המין, אך עדיין אחד מהמאפיינים החשובים בעץ. מאפיין זה נמצא בחלק התחתון של העץ מפני שהעץ מוגבל עד 4 רמות (התבקשו להדפיס עץ עד לגודל ברור) , דבר המתיישב עם הממצאים והנתונים שלנו.

רשתות נויורונליות

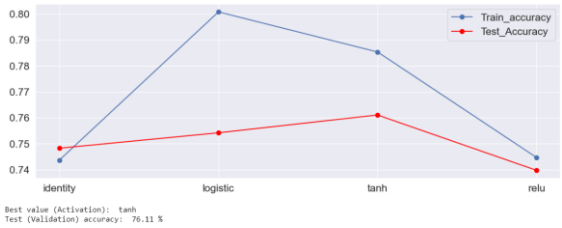
1. אימון הרשת בערכי ברירת מחדל

בשלב הראשון ביצענו המרה של הנתונים שלנו להתפלגות נורמלית סטנדרטית באמצעות "StandartScaler". לאחר מכן הרצנו את הרשת בערכי ברירת המחדל עבורה השתמשנו בקונפיגורציה הדיפולטיבית של MLP. עבורה גודל שכבה חבויה הינה 100 נוירונים, פונקציית אקטיבציה דיפולטיבית של השכבות החבויות הינה "relu" אשר מהווה את פונקציית היחידה הליניארית המתוקנת. עבור solver אופטימלי, "Solver= "adam אשר מהווה בסיס לאופטימליות המשקל, אשר מבוסס על גרדיאנט סטוכסטי. "adam" עובד טוב על מערכי נתונים גדולים יחסית מבחינת זמן אימון וציון ולידציה. לאחר הרצת הרשת בערכי ברירת המחדל על המודל התקבלו אחוזי הדיוק הבאים: עבור סט האימון התקבל אחוז דיוק של 80.42%, ועבור סט הבחינה התקבל אחוז דיוק של 75.56%. עבור סט האימון התקבל דיוק נמוך יותר מהדיוק שהתקבל על סט האימון בעצי ההחלטה (כאן הדיוק אינו מושלם). הסיבה לכך נובעת כנראה מהעובדה שנדרשות איטרציות נוספות לכך שתתבצע התאמה מלאה וכן הערך הדיפולטיבי עוצר ב200 איטרציות. דיוק סט האימון גבוה יותר מדיוק סט הבחינה, נשאף לכוונן את הפרמטרים על מנת למצוא את הקונפיגורציה המיטבית ולשפר את הדיוקים.

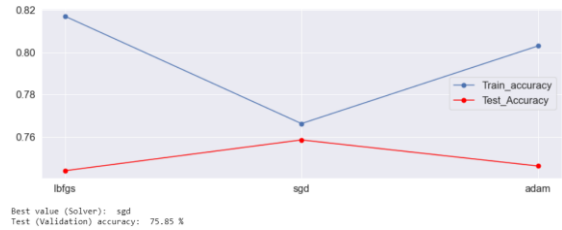
2. כיוונון פרמטרים לרשת ניורונאלית

נבצע תהליך של כוונון פרמטרים למציאת הקונפיגורציה המתאימה ביותר עבור מודל זה. בדומה לתהליך כיוונון בעצים, תחילה בחרנו את ההיפר פרמטרים בהם ראינו לנכון להשתמש ובדקנו את ערכי הכוונון המיטבי של כל אחד בנפרד מתוך טווח ערכים רחב, לאחר מכן הרצנו את כוונון הפרמטרים בצורת Grid Search עבורם נבחרו הערכים שנתנו את הדיוק הגבוה ביותר עבור כל אחד מההיפר פרמטרים.

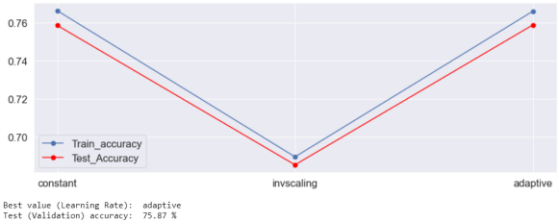
ערכי היפר פרמטרים כפונקציה ומשמעותן בכוונון



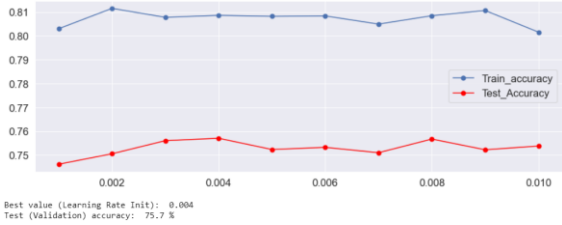
activation- עבור פונקציית האקטיבציה, בחירת הפונקציה בשכבה הנסתרת תשלוט עד כמה מודל הרשת לומד את מערך ההדרכה. בחירת פונקציית האקטיבציה בשכבת הפלט תגדיר את סוג התחזיות שהמודל יכול לעשות. תחילה הערכנו את פונקציית האקטיבציה אשר נותנת את הדיוק הגובה ביותר באופן עצמאי. טווח הפונקציות שנלקח הינו הרחב ביותר בכדי להגדיל את הסיכויים למציאת פונקציית אקטיבציה המתאימה ביותר. (נספח 2.01)



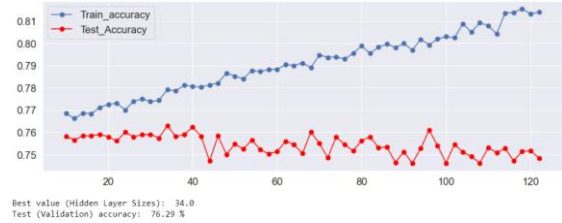
solver- פרמטר זה מציין את האלגוריתם לאופטימיזציה של משקל על פני הצמתים. בשלב זה, הערכנו את הsolver אשר נותן את הדיוק הגבוה ביותר מכלל הסוגים הקיימים על מנת להגדיל את טווח האופציות שלנו. (נספח 2.02)



learning_rate- פרמטר זה נוגע לקצב הלמידה. עבור כל איטרציה הוא מציין את פרופורציית השינוי לנגזרת החלקית של השגיאה ביחס למשקולות. הגדלת הערך יוביל לקצב מהיר יותר של שינוי המשקולות, וככל שנקטין, הקצב יאט בהתאם. בעבור ערכים גדולים, יתכנו פספוסים של שגיאה, ועבור ערכים ערך נמוך יתכן והוא יתקע במינימום מקומי. בעבור כיוונון היפר פרמטר זה בפני עצמו, בחרנו את כלל טווח הערכים על מנת להגדיל את סיכויי ההצלחה שלנו. (נספח 2.03)



learning_rate_init-פרמטר זה נוגע לקצב הלמידה הראשוני בו נעשה שימוש. הוא שולט בגודל המדרגה בעדכון המשקולות. בחרנו טווח ערכים רחב, החל מ0.001 ל0.01 להגדלת סיכויי דיוק גבוה. (נספח 2.04)



hidden_layer_sizes- פרמטר זה מאפשר להגדיר את מספר השכבות ואת מספר הצמתים שאנו רוצים שיהיו ברשת. כל אלמנט ב-tuple מייצג את מספר הצמתים במיקום ith שבו i הוא האינדקס של ה-tuple. לפיכך אורך ה-tuple מציין את המספר הכולל של שכבות נסתרות ברשת. תחילה הערכנו את מספר השכבות החבויות בכך שחישבנו מהן מספר הנוירונים המיטבי עבור היפר פרמטר זה בלבד. הטווח של מספר הנוירונים שנלקח הינו 10-122. טווח ערכים רחב לטובת בחינה של תחום רחב להגדלת סיכויי ההצלחה. לאחר מכן, עבור מספר נוירונים אשר הביאו את הדיוק הגבוה ביותר, חישבנו את מספר השכבות המיטבי, בטווח ערכים יחסית רחב גם כן על מנת שנוכל להקיף כמה שיותר אופציות, אך בזמן ריצה סביר. (נספח 2.05)

	hidden_layer_sizes	activation	activation	solver	learning_rate	learning_rate_init	std_test_score	mean_test_score
0	(96, 96, 96, 96)	tanh	tanh	sgd	constant	0.004	0.004640	0.755261
1	(96, 96, 96, 96)	tanh	tanh	sgd	adaptive	0.004	0.004640	0.755261
2	(96, 96, 96)	tanh	tanh	sgd	constant	0.004	0.006786	0.750771
3	(96, 96, 96)	tanh	tanh	sgd	adaptive	0.004	0.006786	0.750771
4	(96, 96, 96)	logistic	logistic	sgd	adaptive	0.004	0.003601	0.746619
5	(96, 96, 96)	logistic	logistic	sgd	constant	0.004	0.004356	0.746591
6	(96, 96, 96, 96)	logistic	logistic	sgd	constant	0.004	0.000087	0.634168
7	(96, 96, 96, 96)	logistic	logistic	sgd	adaptive	0.004	0.000087	0.634168

3. אימון רשת נוירונים

רשת הנוירונים הינה פונקציה מורכבת, וכמות הטרנספורמציות גדולה מאוד, מה שמתבטא בזמני ריצה גדולים מאוד ולא ראליים לאילוצי המחשב האישי שלנו ולאילוצי זמן. מסיבה זו, בשלב הראשון צמצמנו את האפשרויות על ידי הגבלת מספר האיטרציות ל150, לקיחת שתי

השכבות החבויות בעלות הדיוק הגבוה ביותר, פונקציית אקטיבציה של שני הערכים הגבוהים ביותר, פונקציית ה solver של הערך גבוה ביותר שהתקבל, קצב לימוד של שתי הערכים הטובים ביותר, וגודל המדרגה בעדכון המשקולות בקצב הלמידה אשר נתן את הדיוק הגבוה ביותר בהרצות היפר פרמטרים בנפרד. כל זה תחת cv=10. מתוך הבנה שהאופטימום לא מתכנס במספר איטרציות שהוגדר, ביצענו את הקונפיגורציה אשר קיבלה את המקום הראשון והרצנו אותה עבור מספר איטרציות גדול (1000) אשר עבורו קיבלנו את האחוז דיוק הגבוה ביותר: עבור הדיוק שהתקבל על סט האימון הינו:98.67% והדיוק על סט הבחינה:78.36% .

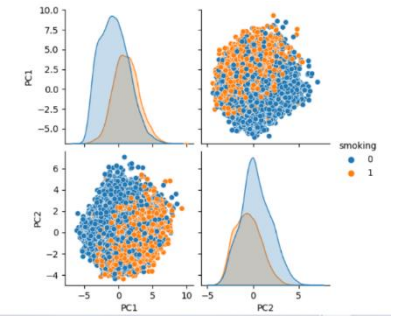
	Hidden Layer Sizes	Train_Accuracy	Test_Accuracy
0	(96, 96, 96, 96)	0.986729	0.783638

K-Means

האלגוריתם משמש אלגוריתם פופולרי ויחסית פשוט בלמידת מכונה ללא פיקוח. הוא מסיק מסקנות ממערכי נתונים תוך שימוש בווקטורי הקלט בלבד, ללא התייחסות לתוצאות ידועות. מטרתו היא לקבץ נקודות נתונים דומים ולגלות דפוסים בסיסיים. האלגוריתם מחפש מספר קבוע (k) של אשכולות במערך הנתונים, אשר אשכול מוגדר כאוסף של נקודות נתונים המצטברות יחד בגלל קווי דמיון מסוימים. במילים אחרות, אלגוריתם K-means מזהה מספר k של מוקדים, ולאחר מכן מקצה כל נקודת נתונים לאשכול הקרוב ביותר, תוך שמירה על המוקדים קטנים ככל האפשר. כדי לעבד את נתוני הלמידה, אלגוריתם K-means מתחיל בקבוצה ראשונה של מרכזים שנבחרו באקראי, המשמשים כנקודות התחלה לכל אשכול, ולאחר מכן מבצע חישובים איטרטיביים כדי לייעל את מיקומי הצנטרואידים. התהליך נעצר כאשר מתקיים אחד מהשניים: הסנטרואידים התייצבו - אין שינוי בערכים שלהם מכיוון שהאשכול הצליח או המספר המוגדר של איטרציות הושג.

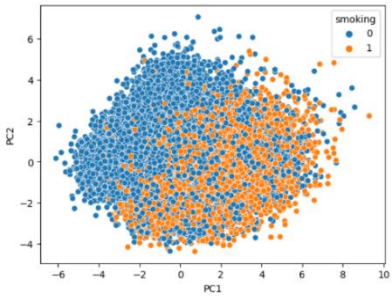
1. הרצת ערכי ברירת מחדל

בעבור הרצת האלגוריתם עם ערכי ברירת המחדל, מלבד מספר האשכולות אותו הגדרנו כך שיתאים לבעיה עימה אנו עובדים, ולכן הגדרנו את מספר האשכולות להיות 2. השתמשנו באלגוריתם PCA על מנת שנוכל להציג את הנתונים בממד נמוך יותר בתצוגת ויזואלית (דו ממדי). בצורה זו נוכל להציג בתרשים את הקשר בין משתנה המטרה לשונות הפיזור של המאפיינים. ניתן לראות בגרף את הפיזור ביחס לשני מרכיבים מרכזיים. על אף שהאלגוריתם מנסה לשמור על כמה שיותר שונות מהדאטה הכללי, מבדיקה שביצענו גילינו כי pca מחזיק ב 34.62% מתוך השונות.



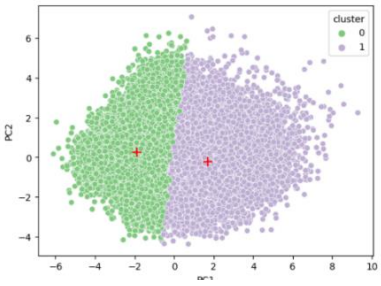
2. דיון בתוצאות

תחילה נתבונן האם החלוקה של pca נותנת לנו מידע. ניתן לראות שקיימות שתי קבוצות (גרף 1). על אף הפיזור הגבוה יחסית ביניהן, ניתן להבדיל בצורה מסוימת בין שתי הקבוצות (צפיפות תצפיות של מעשנים סביב מרכז מסוים לעומת לצפיפות התצפיות של לא מעשנים ביחס למרכז מסוים).

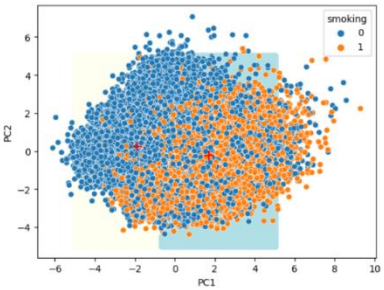


בשלב הבא, נחפש את הקבוצות הללו מתוך הנתונים (בהינתן ואין ברשותנו את משתנה המטרה). אלגוריתם kmeans מחפש 2 קבוצות דומות מתוך הנתונים בכך שהוא מוצא את מרכז האשכול עבור כל משתנה ולאחר מכן חוזה את התצפיות בכך שלוקח כל נקודה ושם אותו בצנטרואיד שהכי קרוב אליו, ובעזרת הpca נוכל לראות את החלוקה, את הצנטרואידים שנמצאו, אפשר לראות על הגרף של pca1,pca2 (גרף 2) ביחס למה

שאנחנו יודעים על העולם האמיתי (גרף 1), יחסית קיים דמיון מסוים על אף שאלו לא בדיוק אותן קבוצות כפי שהן בעולם האמיתי שלנו.



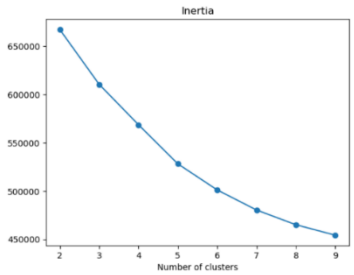
בשלב הבא ביצענו טרנספורמציה הפוכה והחזרנו אותו לצורה המקורית (גרף 3), נבצע פרדיקציות לנקודות במרחב. ניתן לשייך את הנקודות למרכזים בהתאם לשטח בהן הן נופלות כאשר הצבעים של הנקודות הן הנקודות האמיתיות והאשכולות הן הרקע. ניתן להסיק כי קיים חוסר כלשהו של הפרדה, אך הדבר הגיוני מכיוון שאנו בלמידה לא מפוקחת.



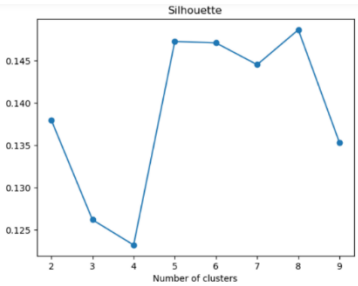
אחוז הדיוק על סט האימון: 67.20% אחוז הדיוק על סט הבחינה: 66.63%.

3. מדדים עבור K-means

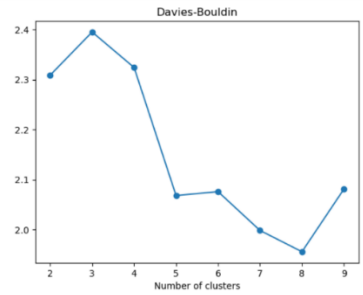
כאשר אנו לא יודעים לכמה קבוצות עלינו לחלק את העולם, נעזר במדדים שעוזרים לנו למדוד את ביצועי האשכול שלנו. לצורך כך אימנו 8 מודלים של אשכול עם ערכי K משתנים. על מנת למצוא את מספר האשכולות המתאים ביותר השתמשנו במדדים הבאים:



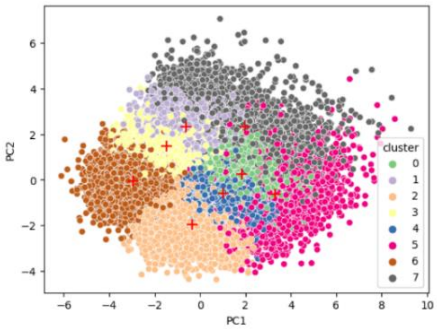
Inertia- המדד מעריך עד כמה הנתונים היו מקובצים על ידי K-Means. הוא מחושב על ידי מדידת המרחק בין כל נקודת נתונים למרכז שלה, סכום המרחקים בריבוע של הדגימות עבור מרכז האשכול הקרוב אליהן ביותר. עבור מודל טוב הוא מודל עם אינרציה נמוכה ומספר נמוך של אשכולות (K). עם זאת, זהו פשרה מכיוון שככל ש-K עולה, האינרציה פוחתת. כדי למצוא את ה-K האופטימלי עבור מערך נתונים, נמצא את הנקודה שבה הירידה באינרציה מתחילה להאט. בגרף שלנו נבחר ב3 אשכולות מכיוון שהשינוי בין 2-3 אשכולות הוא המשמעותי ביותר.



Silhouette- נשתמש במדד זה כדי להעריך את מרחק ההפרדה בין האשכולות המתקבלים. עליית המדד מתארת עד כמה קרובה כל נקודה באשכול אחד לנקודות באשכולות הסמוכים ובכך מספקת דרך להעריך פרמטרים כמו מספר אשכולות באופן ויזואלי. הוא מחושב על ידי הפרש של המרחק הממוצע של האשכול הקרוב ביותר מהמרחק הממוצע בתוך האשכול וחילוק התוצאה במקסימום מבין שני הערכים. נרצה שציון זה יהיה כמה שיותר גבוה מכיוון שזה מעיד שהתצפיות הקרובות ביותר באותו אשכול. מהגרף שלנו נראה כי הערך הגבוה ביותר מתקבל עבור לk=8.



Davies bouldin- הציון מוגדר מהווה מדד הדמיון הממוצע של כל אשכול עם האשכול הדומה לו ביותר, כאשר הדמיון הוא היחס בין מרחקים בתוך אשכול למרחקים בין אשכול. לפיכך, אשכולות שנמצאים רחוק יותר זה מזה ופחות מפוזרים יביאו לציון טוב יותר. הציון המינימלי הוא אפס, כאשר עבור ערכים נמוכים יותר הם מצביעים על קיבוץ טוב יותר. בגרף שלנו ניתן לראות שעבור k=8 מתקבל ערך הנמוך ביותר ולכן ייתן את הציון הטוב ביותר.



נבחר ב k=8, הפיזור של הנתונים שלנו רב מאוד, ולדעתנו זהו מספר אשכולות הגיוני. ניתן לראות כי שניים מהמדדים נותנים מספר אשכולות השווה ל-8 ומדד אחד הנותן 3 אשכולות. ערך זה אינו מתקשר לסיפור שלנו, מכיוון שבסיפור שלנו קיימים 2 אשכולות בלבד- האם הנבדק מעשן או לא מעשן. האלגוריתם בבסיסו מסתכל על מרחקים מהממוצע ושם את הדגש על הפיזור בין הנתונים ולכן בוחר במספר אשכולות רב יותר מהסיפור שלנו. ניתן לשער שהאלגוריתם עשוי להסתכל על רמות שונות של עישון ולאו דווקא על משתנה המטרה כמקבל ערך בינארי של מעשן או לא מעשן. בגרף הבא מתוארת החלוקה לאשכולות עבור k=8.

	Kmeans	אשכול היררכי
silhouette	0.148	0.332
davies_bouldin	1.956	0.447

4. שיטת אשכול נוספת- אשכול היררכי

אשכול היררכי הוא אלגוריתם גמיש אשר משתמש בגישה של "מלמטה למעלה" כך שבשלב הראשון כל תצפית מתחילה באשכול משלה וזוגות של אשכולות מתמזגים ככל שמתקדמים במעלה ההיררכיה על סמך מדד דמיון.
היתרונות של אשכול היררכי ביחס לkmeans הוא שבאשכול היררכי הנחה על מספר מסוים של אשכולות.
נשווה בין מספר האשכולות מהסעיף הקודם (k=8) עבור אשכול היררכי בערכי ברירת מחדל ביחס לkmeans ונעמוד על ההבדלים בין המדדים: אחוז דיוק על סט האימון באשכול ההיררכי: 63.4% ועל סט הבחינה: 62.6% נראה כי לפי מדדים אלו ואחוז הדיוק נעדיף להשתמש באלגוריתם kmeans. (נספח 3.01)
הערה – עקב אילוצי המחשבים האישיים שלנו, חישוב הגרף גרם לקריסה של שני המחשבים שלנו לכן לא יכולנו לצרף אותו.
בנספח (ובקוד) תוכל לראות את הקוד עבור הוצאת פלט הגרף. (נספח 3.02)

השוואה בין מודלים

בעבור משימת הסיווג, נעדיף להשתמש בלמידה מפוקחת ונבחר במודל מכוונן של רשתות נוירונליות עבורו קיבלנו את אחוז הדיוק הגבוה ביותר. המדד בו השתמשנו להשוואת המודלים (מודלים לא מונחה ומודלים מונחים) היא באמצעות מדד הדיוק. מדד הדיוק מגדיר כמה ערכים נותנים פסוק אמת מסך כלל הערכים ובכך אנו יודעים בכמה מהחיזויים המודל שלנו צדק. בלמידה מפוקחת, האלגוריתם לומד ממערך הנתונים של האימון על ידי ביצוע ניבויים איטרטיביים על הנתונים והתאמה לתשובה הנכונה. מודלים של למידה בפיקוח נוטים להיות מדויקים יותר ממודלים של למידה ללא פיקוח, אך הם דורשים התערבות אנושית מראש כדי לתייג את הנתונים כראוי. בלמידה מפוקחת, המטרה היא לחזות תוצאות עבור נתונים חדשים כאשר אנו יודעים מראש לאיזה סוג תוצאות לצפות. בעבור למידה לא מפוקחת, המטרה היא לקבל תובנות מכמויות גדולות של נתונים חדשים וכאשר אין ברשותנו את משתנה המטרה. בבעיה שלנו גם הרשת הנוירונלית וגם עץ ההחלטה משתמשות במשתנה המטרה על מנת לנבות תצפיות עתידיות בעוד ששיטות מסוג אשכול יוצרות את המחלקות על בסיס הפיזור של הנתונים הקיימים ללא משתנה המטרה ומסיבה זו היא משימה מורכבת יותר. עבור השוואה בין שלושת המודלים, השתמשנו באותו מדד השוואה והוא מדד הדיוק בהתאם ערכים הבאים:

המודל הנבחר

המודל הנבחר הוא מודל רשת הנוירונלית בעבורה קיבלנו את אחוז מדד הדיוק הגבוה ביותר, הקונפיגורציה עבורה התקבל המודל הטוב ביותר הינה: מספר שכבות חבויות הינו (96,96,96,96), פונקציית אקטיבציה:'tanh', solver: 'sgd', קצת למידה: 'constant' וגודל מדרגה של קצב למידה: 0.004. מצאנו כי האופטימום התכנס בעבור 498 הרצות. בחרנו את מאפייני הקונפיגורציה הזו, מכיוון שבשלב הראשון ביררנו מהן הערכים המיטביים עבור כל פרמטר

בנפרד. לאחר מכן הרצנו באמצעות grid search את מאפייני הקונפיגורציה אשר קיבלו את הערכים הטובים ביותר למציאת השילוב המיטבי. מאילוצי זמן ריצה וכוח עיבוד הגבלנו את מספר ההרצות ל150 בהבנה שהאופטימום לא התכנס בשלב זה. לאחר שהערכנו מהי הקונפיגורציה המיטבית, הרצנו אותה עבור מספר איטרציות גדול (1000 הרצות) ונמצאה הקונפיגורציה הנ"ל. יתכן וקיים פתרון טוב יותר, אך מאילוצים שצוינו, נשארנו עם הפתרון איתו ביצענו את החיזויים הסופיים.

שיטה	דיוק ממוצע סט בחינה	דיוק ממוצע סט אימון
עץ החלטה	77.53%	99.87%
רשת נוירונים	78.36%	98.67%
Kmeans	66.63%	67.20%
אשכול היררכי	62.60%	63.40%

מטריצת המבוכה

מטריצת המבוכה היא מטריצה של מספרים שמעידה לנו היכן מודל שוגה. זוהי התפלגות של ביצועי הניבוי של מודל סיווג, כלומר היא דרך מאורגנת למיפוי התחזיות למחלקות המקוריות שאליהן שייכים הנתונים. ניתן להשתמש במטריצת המבוכה רק כאשר מדובר בלמידה מפוקחת. המטריצה מסייעת להעריך את מדדים שניתן להשתמש בהם כדי להעריך את המודל. המטריצה יכולה עוזרת בהשוואה על נקודות החוזק והחולשה כדי להשיג את הביצועים האופטימליים.

True Positive - הוא מספר התצפיות שסווגו כהלכה כ"חיוביות".

True Negative - הוא מספר התצפיות שסווגו כהלכה כ"שליליות".

False Positive - הוא מספר התצפיות שסווגו בטעות כ"חיוביות".

False Negative - הוא מספר התצפיות שסווגו בטעות כ"שליליות".

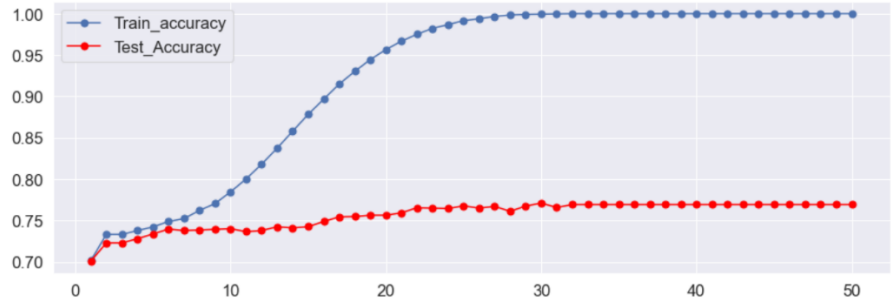
		Actual	
		Negative	Positive
Predicted	Positive	4561	1025
	Negative	989	2336

נספחים

1. עץ החלטה :

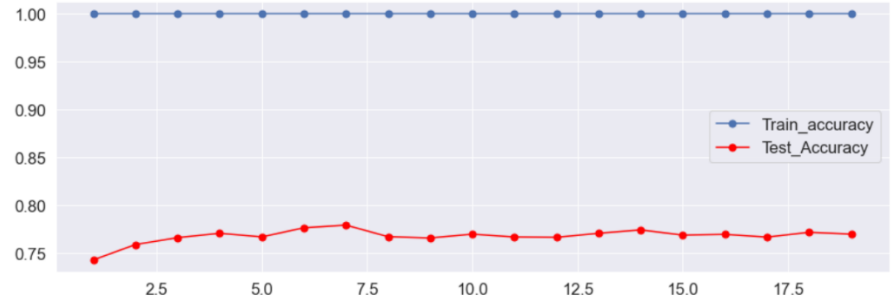
נספח 1.01 - כוונון המיטבי של כל היפר פרמטר בנפרד

	Max_Depth	Train_Accuracy	Test_Accuracy
0	30.0	0.999383	0.770957
1	50.0	1.000000	0.769162
2	40.0	1.000000	0.769162
3	32.0	1.000000	0.769162
4	33.0	1.000000	0.769162
5	34.0	1.000000	0.769162



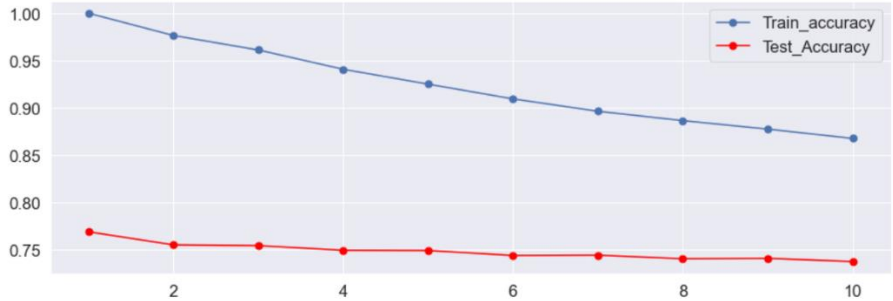

```
Best value (Max Depth):  30.0
Test (Validation) accuracy:  77.1 %
```

	Max_Features	Train_Accuracy	Test_Accuracy
0	7.0	1.0	0.778925
1	6.0	1.0	0.776119
2	14.0	1.0	0.773875
3	18.0	1.0	0.771406
4	13.0	1.0	0.770396
5	4.0	1.0	0.770396



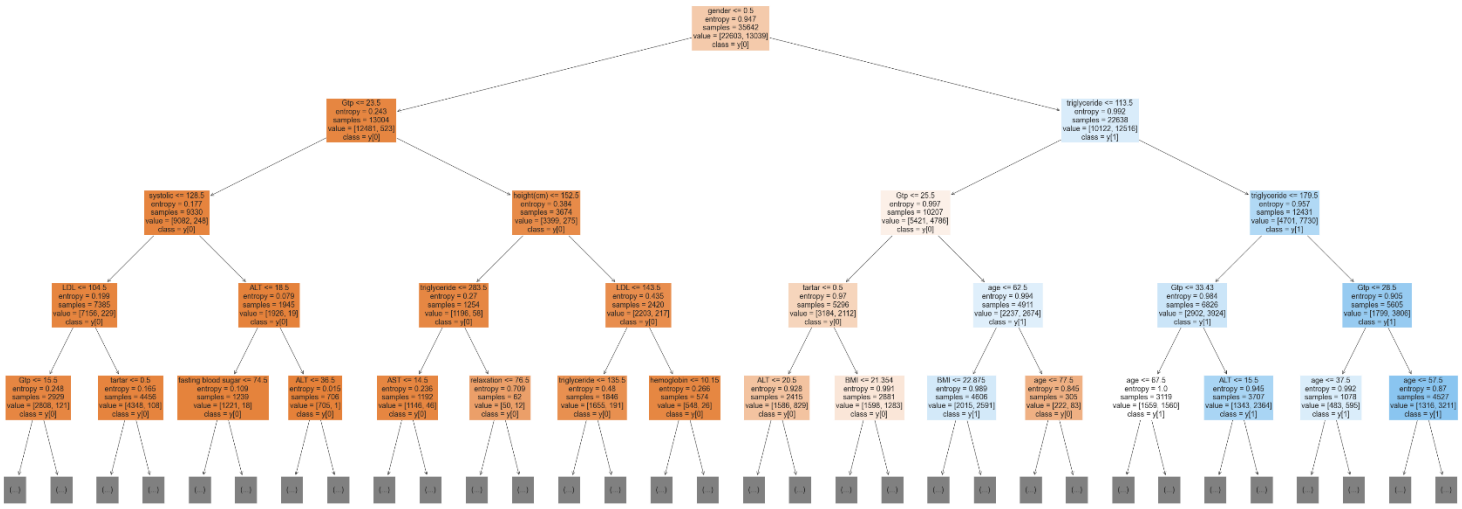

```
Best value (Max Features):  7.0
Test (Validation) accuracy:  77.89 %
```

	Min_Samples_Leaf	Train_Accuracy	Test_Accuracy
0	1.0	1.000000	0.769162
1	2.0	0.976488	0.755359
2	3.0	0.961169	0.754573
3	4.0	0.940800	0.749635
4	5.0	0.925088	0.749411
5	7.0	0.896442	0.744473




```
Best value (Min Samples Leaf):  1.0
Test (Validation) accuracy:  76.92 %
```

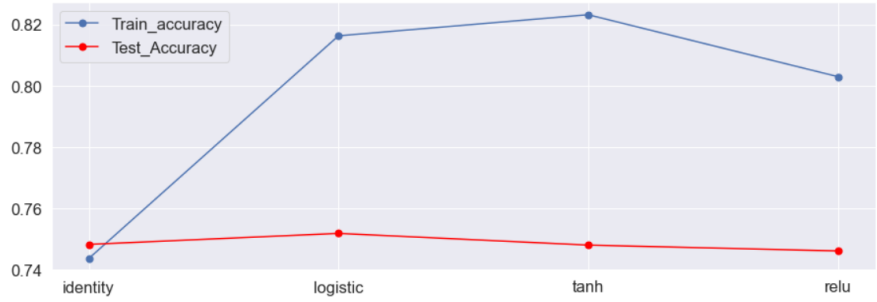
נספח 1.02 – הגדלת התמונה עבור העץ ההחלטה



2. רשתות נויורונליות

נספח 2.01

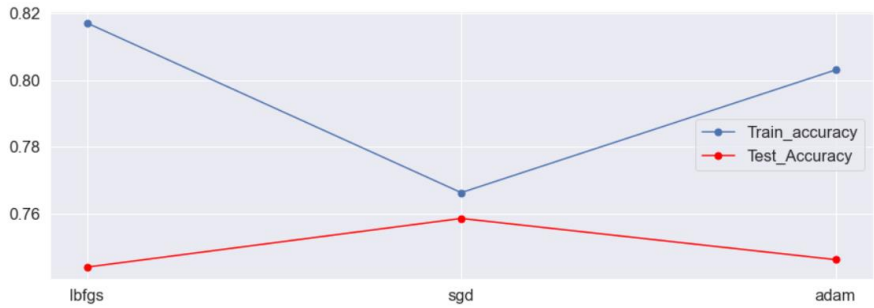
	Activation	Train_Accuracy	Test_Accuracy
0	logistic	0.816284	0.751880
1	identity	0.743701	0.748289
2	tanh	0.823158	0.748064
3	relu	0.802985	0.746156




```
Best value (Activation):  logistic
Test (Validation) accuracy:  75.19 %
```

נספח 2.02

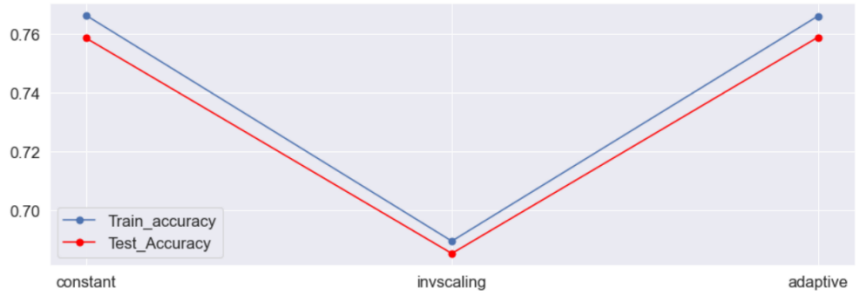
	Solver	Train_Accuracy	Test_Accuracy
0	sgd	0.766231	0.758501
1	adam	0.802985	0.746156
2	lbfgs	0.817070	0.743912




```
Best value (Solver):  sgd
Test (Validation) accuracy:  75.85 %
```

נספח 2.03

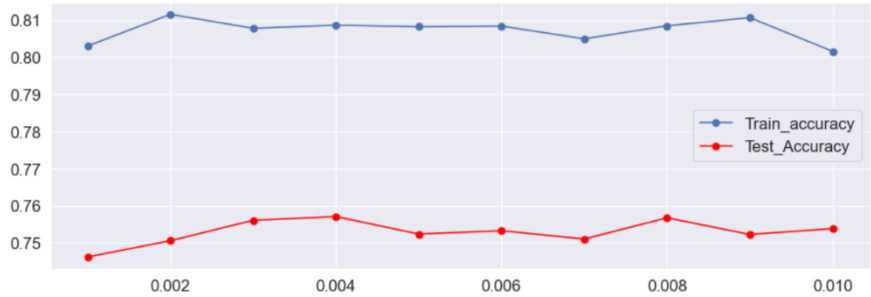
	Learning Rate	Train_Accuracy	Test_Accuracy
0	adaptive	0.765922	0.758725
1	constant	0.766231	0.758501
2	invscaling	0.689524	0.685333




```
Best value (Learning Rate):  adaptive
Test (Validation) accuracy:  75.87 %
```

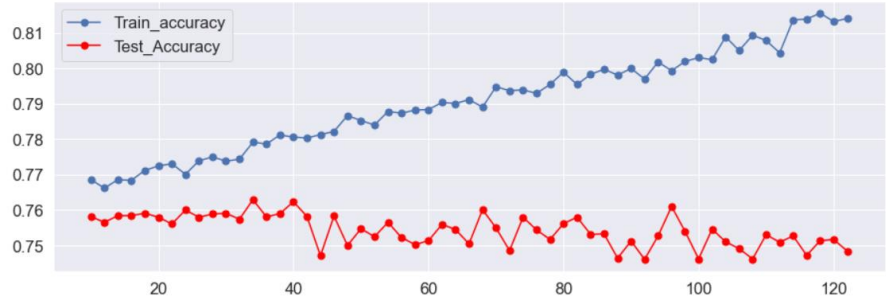
נספח 2.04

	Learning Rate Init	Train_Accuracy	Test_Accuracy
0	0.004	0.808597	0.757042
1	0.008	0.808428	0.756705
2	0.003	0.807755	0.756032
3	0.010	0.801554	0.753787
4	0.006	0.808344	0.753226




```
Best value (Learning Rate Init):  0.004
Test (Validation) accuracy:  75.7 %
```

נספח 2.05




```
Best value (Hidden Layer Sizes):  34.0
Test (Validation) accuracy:  76.29 %
```

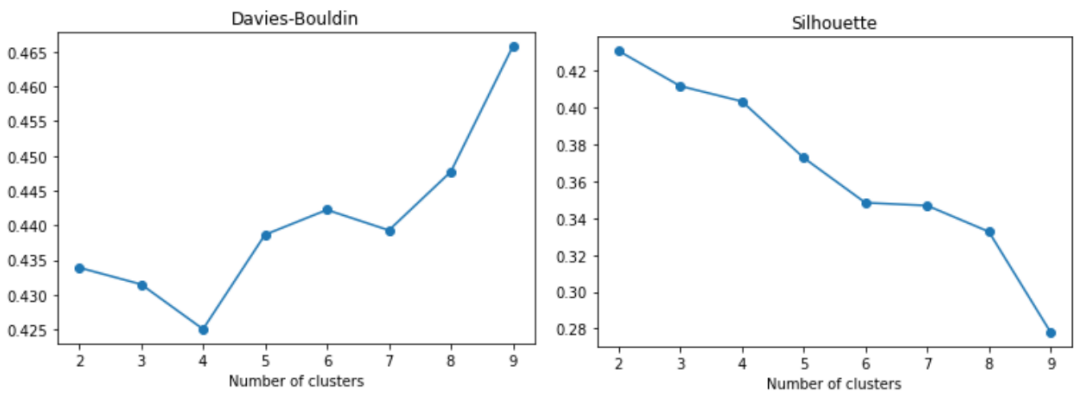
ממצאים עבור מספר הנוירונים לשכבה אחת ועבור מספר שכבות שונות.

	Hidden Layer Sizes	Train_Accuracy	Test_Accuracy
0	34.0	0.779109	0.762877
1	40.0	0.780568	0.762316
2	96.0	0.799282	0.760970
3	68.0	0.789013	0.760072
4	24.0	0.770047	0.759960
5	30.0	0.773778	0.759062
6	18.0	0.771169	0.759062
7	38.0	0.781129	0.758950
8	28.0	0.774957	0.758837
9	16.0	0.768363	0.758389

	Hidden Layer Sizes	Train_Accuracy	Test_Accuracy
0	(96, 96, 96, 96)	0.986729	0.783638
1	(96, 96, 96)	1.000000	0.782404
2	(96, 96)	0.943606	0.768376
3	34.0	0.779109	0.762877
4	40.0	0.780568	0.762316
5	96.0	0.799282	0.760970
6	(40, 40, 40, 40)	0.899585	0.753451
7	(40, 40)	0.820465	0.749411
8	(34, 34)	0.808092	0.747840
9	(34, 34, 34)	0.840189	0.745932

3. מדדים לאשכול היררכי:

נספח 3.01



Graph

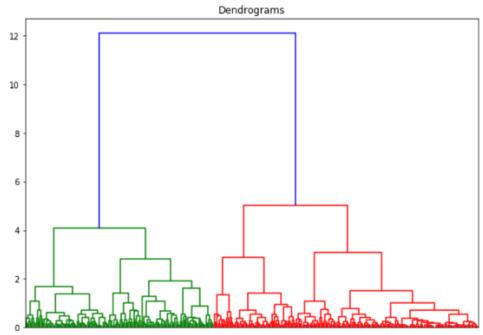
In [ ]:
```
import scipy.cluster.hierarchy as shc
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 7))
plt.title("Dendrograms")
dend = shc.dendrogram(shc.linkage(X_train_scaled, method='ward'))
plt.axhline(y=8, color='r', linestyle='--')
```

In [*]:
```
import scipy.cluster.hierarchy as shc
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 7))
plt.title("Dendrograms")
# Create a dendrogram using the linkage matrix
dend = shc.dendrogram(shc.linkage(data_pca, method='single'))
# Add a title to the plot
plt.title("Agglomerative Clustering Dendrogram")
# Display the plot
plt.show()
```

ציפינו לקבל גרף הדומה לגרף זה:



שבו ניתן לראות את הסיווג ל2 פתרונות אפשריים.