ממ"ן 22 כריית מידע

שם: נועם שדה

תאריך: 9.6.2023

שאלה 1 – למידה בייאסינית ולמידה מבוססת תצפיות

א. בחרתי באלגוריתם Naïve – Bays מהסיבות הבאות:

- קל לממש אלגוריתם זה, הוא פשוט להבנה ואף יעיל.
- יכול לעבוד עם הרבה ממדי נתונים ביעילות.
- יש לו ביצועים טובים על סט נתונים קטן.

ב. שלבי הפתרון הבעיה הנתונה תוך שימוש באלגוריתם K-NN:

1. הכנת סט הנתונים – טיפול בערכים חסרים, ערכים חריגים, בחירת פיצ'רים רלוונטים, חילוק סט הנתונים לאימון וולידציה. ביצענו שלב זה בממ"ן 21.
2. בחירת k – פרמטר המייצג את מספר השכנים הסמוכים כדי לבצע סיווג. ערכים קטנים כמו 3 או 5 יהיו יותר רגישים לרעש, לעומת ערכים גדולים כמו 21 שיגרמו לחוסר דיוק.
3. בחירת מדד המרחק – נבחר מדד זה כדי למדוד את הדמיון בין מופעי הנתונים. האפשרויות הם מרחק אוקלידי, מרחק מנהטן או מרחק מינקובסקי.
4. אימון מודל ה- k-NN - אימון המודל על סט האימון
5. הרצת המודל על סט הולידציה
6. הערכת הדיוק באמצעות מדדים כמו AUC-ROC Score, Confusion Matrix, precision, f1 score.
7. שינוי ההיפר פרמטרים בשלבים 2 – 3 וחזרה על השלבים 4-6 כדי לראות אם הגענו לתוצאות יותר טובות.

ג. אבחר באלגוריתם k-NN מהסיבות הבאות:

1. באופן כללי השיקול המרכזי הוא רמת הדיוק. כשהרצתי אלגוריתם זה קיבלתי תוצאות יותר מדויקות מאשר Naïve Bayes (נראה זאת בסעיף ד').
2. אלגוריתם זה יכול להתמודד עם סט נתונים לא מאוזן בצורה יעילה יותר מאשר Naïve Bayes.
3. אלגוריתם זה הוא לא פרמטרי, כלומר לא מניח שהתכונות הן בלתי תלויות.

ד. נריץ את ונדווח את התוצאות על המודל שבחרתי:

```
Model score for the algorithm k-NN:
ROC AUC: 0.9782608695652174
Confusion Matrix:
 [[44  2]
 [ 0 54]]
F1 Score: 0.9818181818181818
```

כלומר קיבלנו דיוק של 97.8% לפי מדד ROC-AUC , וקיבלנו רק שני False Negative לפי ה- Confusion Matrix.
למען סעיף ג' אראה שמודל זה אכן יותר מדויק מאשר אלגוריתם Naïve Bayes:

```
Model score for the algorithm Naive Bayes:
ROC AUC: 0.9565217391304348
Confusion Matrix:
 [[42  4]
 [ 0 54]]
F1 Score: 0.9642857142857143
```

ה. נסיק כי אחוז הדיוק באלגוריתם k-nn הוא גבוה מאוד (רק שתי טעויות) ונעדיף להשתמש בו מאשר באלגוריתם Naïve Bayes.
בהשוואה לממ"ן 21 , שבו הצלחתי להגיע לדיוק של 100% הצלחה בעזרת אלגוריתם יער אקראי, קיבלתי עדיין תוצאה יחסית טובה.
בחרתי באלגוריתמים שנוחים למימוש וקלים להבנה, אך כאשר ניגשים לבעיה מסוג זו עדיף לנסות את כל האלגוריתמים האפשריים על מנת למצוא את האלגוריתם המוצלח ביותר, ובפרט בתחום הרפואה שטעות בסיווג עלולה להיות בעייתית.

שאלה 2: ניתוחי אשכולות

א. מדדי איכות לאשכולות

בכדי למדוד את איכות האשכולות משתמשים במדדים כמו:

- הומוגניות – כמה העצמים בתוך האשכול דומים זה לזה, וכמה הם שונים מעצמים באשכולות אחרים. ככל שהדמיון בתוך האשכול והדמיון בין אשכולות קטן, החלוקה יותר איכותית.

- מגמתיות – במקרה שהחלוקה מניבה מבנים לא אקראיים זה עשוי לעזור לנו בגילוי של מגמה או תופעה לא טריוויאלית ולהסיק מסקנות על הנתונים.

בדומה לשאלה קודמת, ננסה להשתמש בניתוח אשכולות בכדי לקבל תובנות שיעזרו לנו למשימת הסיווג, לכן בהרצת האלגוריתם נתעלם מתכונת הסיווג, נגדיר מספר אשכולות כמספר הערכים של תכונת הסיווג, ונשווה בין תכונת הסיווג לחלוקת האשכולות שתתקבל.

ב. בחירת גישה לניתוח האשכולות
אבחר בגישה מבוססת חלוקה ואעזר באלגוריתם K-Means.
גישת החלוקה משייכת את העצמים למספר מחיצות מוגדר מראש, והשיוך מתבצע באמצעות פונקציית מרחק/דמיון.

האלגוריתם k-means מתייחס לכל רשומה כנקודה (ווקטור בעל מספר קורדינאטות כמספר התכונות) במרחב.

ראשית, נבחרות k נקודות רנדומליות (ניתן להגדיר בחירה יותר "חכמה") שמייצגות את נקודות המרכז של k האשכולות ולכל אחת מהנקודות הקיימות מחשב את המרחק בינה לבין כל אחת מ k מרכזי האשכולות, ומסווגת לאשכול שהיא הכי קרובה אליו (הכי קרובה למרכזו).

בכל שלב מרכז האשכול מחושב ע"י חישוב מרכז כל הנקודות שנמצאות באותו אשכול, ושוב מתבצע שיוך של נקודות לאשכולות. התהליך נגמר כאשר שינוי מרכזי האשכולות לא גורר יותר שינוי בשיוך הנקודות לאשכולות.

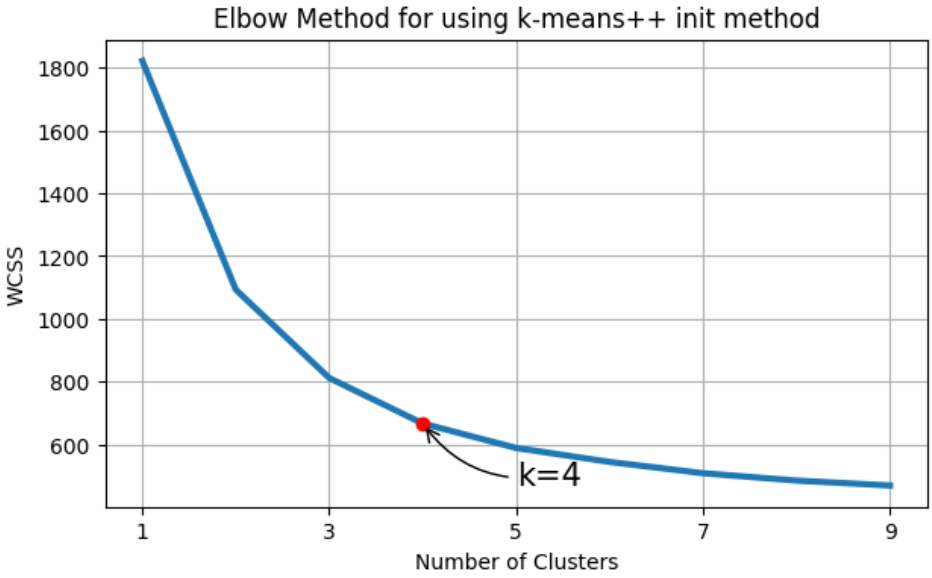
בחרתי אלגוריתם זה כיוון שהוא פשוט וקל להבנה, קל למימוש, ואף יכול להגיע לביצועים טובים.

ג. שלבי ניתוח האשכולות

1. הכנת הנתונים – בממ"ן 21 טיפלתי בערכים חריגים, ערכים חסרים, הפכתי נתונים קטגורים לבינארים. לא נירמלתי את הנתונים כיוון שקיבלתי ביצועים מדויקים ומהירים עם המודלים בהם השתמשתי בממ"ן 21. כעט, אנרמל את הנתונים המספריים כך שיהיו בטווח הערכים 0-1, וכך אלגוריתם K-means יעבוד בצורה יותר טובה. אבצע זאת באמצעות המתודה MinMaxScaler המצויה בספרייה sklearn בפייתון.
2. הרצת האלגוריתם והגדרת הפרמטרים – אריץ אלגוריתם זה באמצעות הספרייה sklearn.cluster הנמצאת בפייתון.

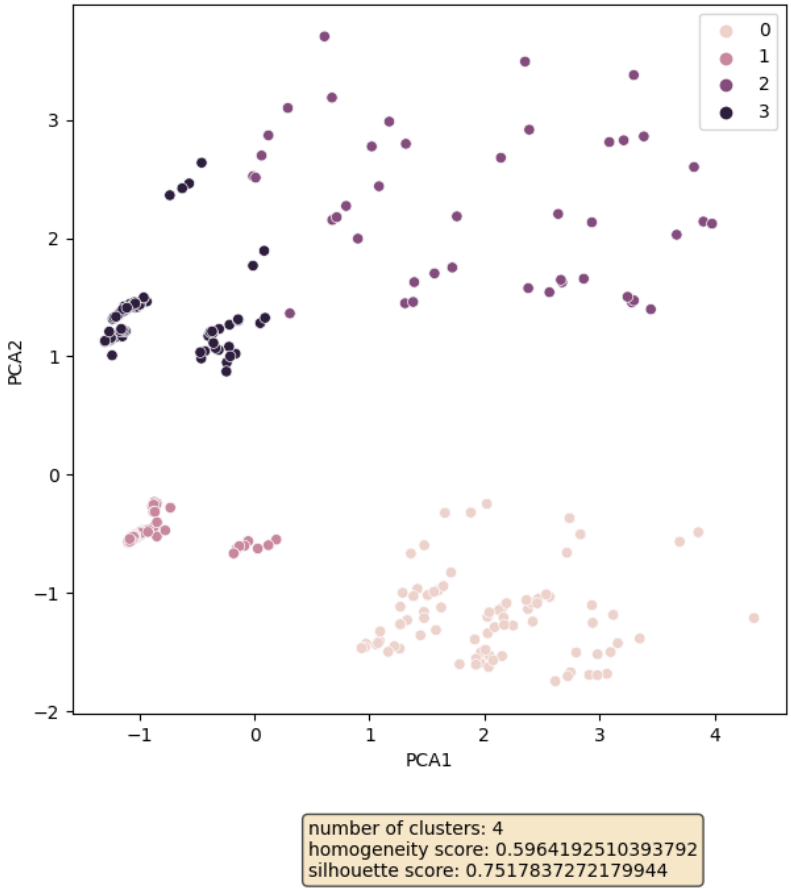
- בעזרת שיטת Elbow Method אמצא את הפרמטר k האופטימלי.
- אגדיר את אתחול האשכולות לפי המתודה 'k-means++' כך שהאשכולות יותחלו בצורה טובה.
- אגדיר את מטריצת המרחק להיות אוקלידית

ד. הרצה ותוצאות – בעזרת WCSS (Within-Cluster Sum of Square) אנתח את דיוק האלגוריתם ואמצע את ה-k האופטימלי בעזרת שיטת Elbow Method:



כלומר עבור k=4 האלגוריתם יעבוד בצורה הטובה ביותר.

אשתמש במדד סילואט ובמדד הומוגניות על מנת להעריך את תוצאות האלגוריתם.
כדי שנוכל ליצור גרף מסוג plot scatter ביצעתי טרנספורמציה לינארית למזעור ממדי הטבלה בעזרת אלגוריתם PCA:



לפי הגרף ניתן לשים לב שהאשכולות מופרדים בצורה יחסית טובה.

קיבלנו ציון 0.75 עבור מדד סילואט ועבור מדד ההומוגניות קיבלנו ציון 0.59. ציונים אלה מעידים על רמה מתונה של הפרדה ולכידות בין האשכולות, ובנוסף מצביע על כך שנקודות הנתונים בתוך כל אשכול מקובצים בצורה סבירה ומופרדים מאשכולות אחרים.
להלן טבלה המראה את הערכים בכל אשכול:

אשכול/עמודת מטרה	0	1	2	3
notckd	48	87	40	75
ckd	150	0	0	0

ניתן לראות שרוב הנקודות נמצאות באשכול הראשון, כלומר אחוז הדיוק של אלגוריתם זה אינו גבוה במיוחד. מכך נסיק שאנו נתקלים בבעיה מסוג זו נעדיף להשתמש ב-supervised learning מאשר unsupervised learning.

שאלה 3: רשת נוירונים מלאכותית

א. ארכיטקטורת הרשת: הרשת תהיה מורכבת מ – 4 שכבות כך שבכל שכבה כל נוירון מחובר בקשת לכל נוירון בשכבה הבאה (dense layers).

- בשכבה הראשונה הקלט הוא כמספר הפיצ'רים של סט האימון X (אני השתמשתי ב-21 פיצ'רים).
- 2 שתי שכבות חבויות, השכבה השנייה מורכבת מ-64 נוירונים והשכבה השלישית תהיה מורכבת מ-32 נוירונים. פונקציית האקטיבציה של שכבות אלו היא relu. בחרתי שכבות אלו לפי ניסוי וטעיה.
- שכבת פלט המורכבת מנוירון אחד, כאשר פונקציית האקטיבציה שלה היא sigmoid מכיוון שאנו מסווגים ערך בינארי (0 לא חולה במחלת כליית כרונית ו-1 כן חולה).

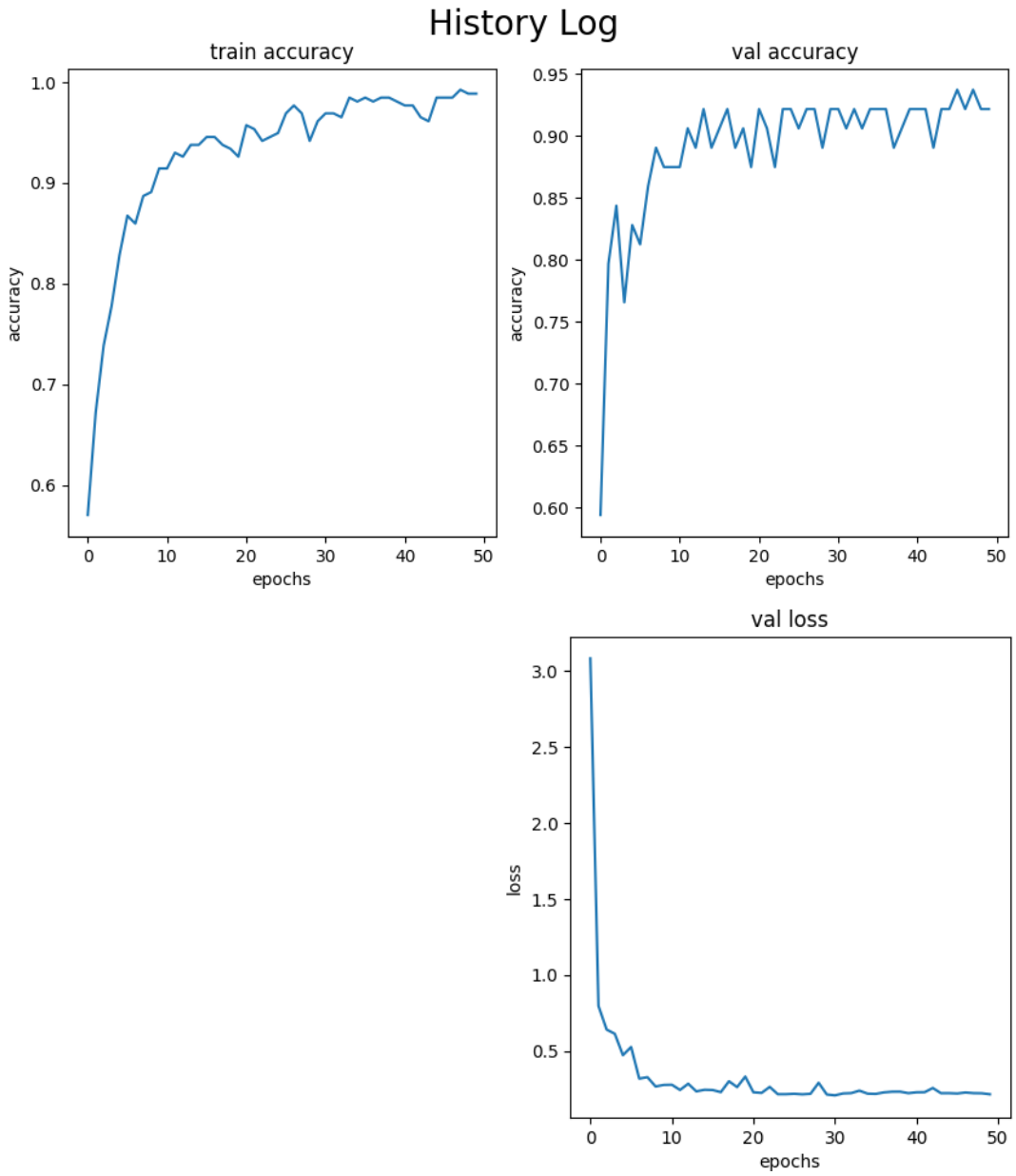
להלן סיכום הארכיטקטורה:

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_3 (Dense)             (None, 64)                1408

 dense_4 (Dense)             (None, 32)                2080

 dense_5 (Dense)             (None, 1)                 33

=================================================================
Total params: 3,521
Trainable params: 3,521
Non-trainable params: 0
_________________________________________________________________
```

ב. הפרמטרים של תהליך האופטימיזציה:

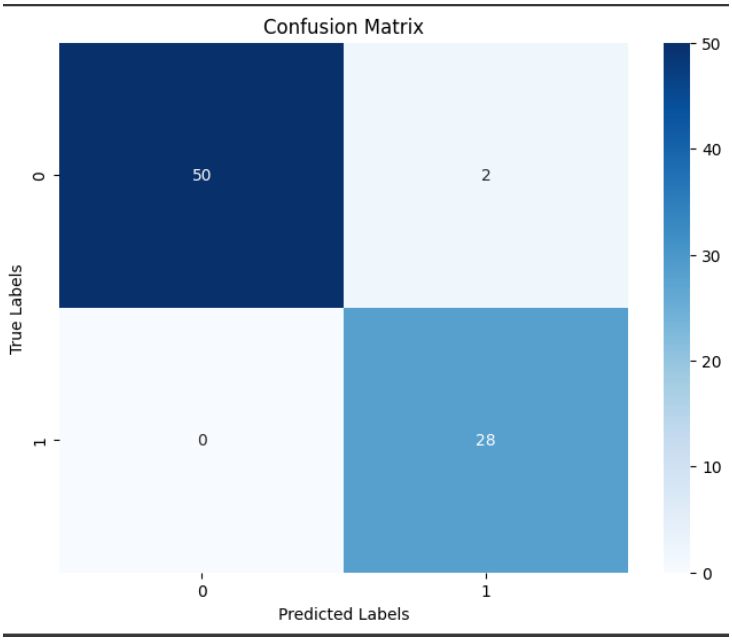
- פונקציית השגיאה היא Binary Cross Entropy מכיוון שאנו חוזים ערך בינארי.
- השתמשתי באופטימייזר הנקרא "Adam" המאוד פופולרי ומוכיח את עצמו בביצועיו הטובים בשימוש ברשתות נוירונים. קצב הלמידה שלו מוגדר באופן ברירת מחדל להיות 0.001.
- גודל ה-batch הוא 32, לא גדול מדי ולא קטן מדי. עדכון הפרמטרים של הרשת על סמך מדגם נתונים בודד עלול לגרום לרעש לא רצוי בתהליך ירידת הגרדיאנט, ולכן שימוש בbatch בגודל 32 יכול להוביל להתכנסות טובה יותר. בנוסף, באופן זה ניתן לבצע חישוב מקבילי באימון הרשת.

ג. הערכת ביצועי הרשת לאורך ה- Epochs של האימון, עבור 50 Epochs:



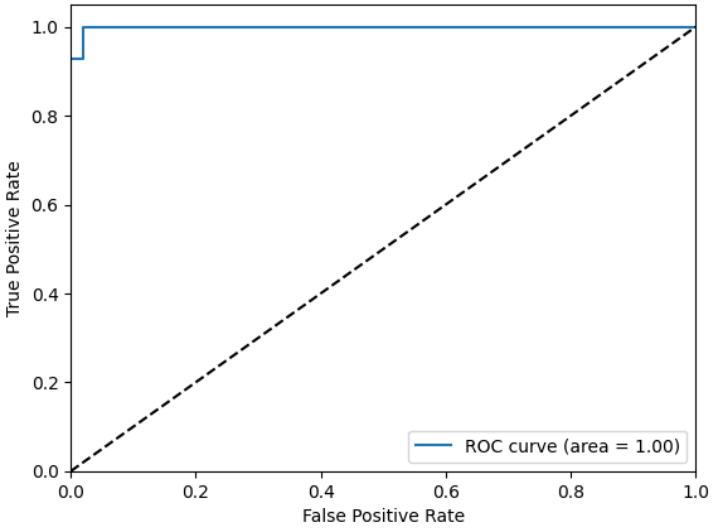
דיוק הרשת על סט המבחן: 97.5% הצלחה.

ד. מקרים חריגים בהם בוצע סיווג שגוי:
מתוך 80 מדגמי הנתונים מסט המבחן, פעמיים המודל סיווג באופן שגוי False Negative (כלומר בוצע סיווג של 0 – לא חולה במחלת כלייה כרונית, במקום 1- חולה במחלה).



ה. ניתוח התוצאות והסקת מסקנות:

גרף ROC-AUC:



לפי גרף ה- ROC-AUC וה-Confusion Matrix נסיק כי ביצועי הרשת טובים מאוד כאשר מתוך 80 מדגמי הנתונים של סט המבחן המודל סיווג פעמיים באופן שגוי, כלומר דיוק של 99.75%. לפי גרף ה-epochs ניתן לראות שעצרנו את אימון הרשת לפני שהגענו ל- overfitting.

שאלה 4 – סיכום ומסקנות

בממן זה השתמשנו באלגוריתמים נוספים על מנת לחזות את עמודת המטרה שלנו, שהיא האם אדם חולה במחלת כליה כרונית או לא.

מבין כל המודלים והאלגוריתמים שהשתמשנו בהם במהלך ממן זה וממן 21, אלגוריתם היער האקראי הניב את התוצאות הטובות ביותר עם אחוזי הדיוק הגבוהים ביותר (100%). אף על פי שמודל זה קיבל את הדיוק הכי גבוה, גם מודלים אחרים בהם השתמשנו כמו k-NN, naïve bayes, עצי החלטה – קיבלו דיוק מאוד גבוה על סט המבחן (כולם מעל 95% דיוק).

הערכנו את ביצועי המודלים באמצעות מדדי הערכה שונים כגון דיוק, F1, Roc-Auc ו Confusion Matrix כדי לקבל תובנות מעמיקות יותר לגבי התנהגות המודלים.

סט הנתונים שקיבלנו הינו יחסית קטן (400 שורות) ולכן שימוש בשיטת רשתות נוירונים על המשימה שקיבלנו פחות אופטימלית מאשר אלגוריתמים של supervised learning. אלגוריתמים של unsupervised learning גם פחות יעילים לפתירת בעיה זו, אלא רק עוזרים לנתח את הנתונים בצורה טובה יותר. אף על פי כן עדיין קיבלנו דיוק גבוה בשימוש ברשתות נוירונים מכיוון שסט נתונים זה הוא "קל" לחיזוי.

פרויקט זה מוכיח ששימוש באלגוריתמים של למידת מכונה יעיל בסיווג בעיות מהסוג שקיבלנו, ויכול לשמש במקרים אמיתיים בתחום הרפואה ועוד.