

מערכות לומדות תשפ"ד – פרויקט מסכם

התאמת מודל לנתונים בעזרת gradient descent ונושא בחירה

מפתח סימנים וצבעים לתרגיל:

בכל סעיף בו מופיע @ עליכם לרשום את התשובה/התוצאות במסמך התשובות.

בסעיפי בונוס תופיע המילה בונוס והסעיף יוצג בסגול.

הדגמות או פתרונות לדוגמה יוצגו בירוק.

חלק ראשון: פתרון בעזרת אופטימיזציה על ידי gradient descent - ירידה במורד השיפוע

נדגים את שלבי התהליך בעזרת המודל הפשוט $f(x) = ax + b$.

(למדנו כי אפשר למצוא את הפרמטרים של המודל הזה אנליטית בעזרת פתרון במונחי מינימום שגיאה ריבועית של $Ax=b$, אבל כאן נשתמש בו להדגמת האופטימיזציה.)

בעזרת נתונים x_i, y_i נמצא עבור המודל את הפרמטרים a, b המתאימים ביותר לנתונים.

א. נבחר loss function. כאן זו שגיאה ריבועית. (מדוע?)

1. נגדיר שגיאה עבור נתון יחיד x_i, y_i :

$$e_i = (f(x_i) - y_i)^2 = (ax_i + b - y_i)^2$$

2. נגדיר את השגיאה הכוללת לסט הנתונים כסכום השגיאות עבור הנתונים

$$E = \sum_i ((ax_i + b) - y_i)^2$$

ב. נגזור את השגיאה על פי הפרמטרים כדי לקבל את ווקטור הגרדיאנט

$$\nabla E = \begin{bmatrix} \frac{\partial E}{\partial a} \\ \frac{\partial E}{\partial b} \end{bmatrix} = \begin{bmatrix} 2 \sum_i (ax_i + b - y_i)x_i \\ 2 \sum_i (ax_i + b - y_i) \end{bmatrix}$$

ג. כעת נפנה לפתרון האיטרטיבי:

נבחר קבוע קצב למידה lr (learning rate),

נבחר (או נגדיל) ערכי התחלה עבור הפרמטרים: a_0, b_0 , ונציב אותם

$a = a_0$

$b = b_0$

ד. נחשב את ערך הגרדיאנט עבור ערכי הפרמטרים הנוכחיים.

ה. נחשב את ערך הפרמטרים הבאים: התקדמות מערך הפרמטרים הנוכחיים על ידי צעד בכיוון מורד השיפוע (מינוס קצב הלמידה כפול הגרדיאנט) כך

$$\begin{bmatrix} a \\ b \end{bmatrix}_{new} = \begin{bmatrix} a \\ b \end{bmatrix}_{old} - lr \nabla E$$

ו. אם לא הגענו לתנאי העצירה נחזור בלולאה לסעיף ד'.
תנאי עצירה מקובלים הם מספר צעדים גדול ממספר מסוים ו\או גודל התקדמות קטן מסף מסוים.

(שאלה 1)

כתיבת קוד ופתרון:

א. כתבו קוד המוצא את הפרמטרים של מודל בעזרת gradient descent עבור המודל $f(x) = ax + b$,
ושגיאה של סכום ההפרשים בריבוע $E = \sum_i (f(x_i) - y_i)^2$,
והנתונים האלה

$x = [-3.0, -2.0, 0.0, 1.0, 3.0, 4.0]$

$y = [-1.5, 2.0, 0.7, 5.0, 3.5, 7.5]$

הנחיות:

- הציגו את הנתונים בגרף בצורה ויזואלית.
 - חשבו את הביטויים לשני מרכיבי הגרדיאנט של השגיאה ביחס לפרמטרים (אפשר לחשב ידנית ולהכניס זאת hard coded לתוכנית במקומות שצריך).
 - בחרו ניחוש התחלתי לפרמטרים, למשל 1,1.
 - הציגו את הקו שמייצג המודל עבור הניחוש ההתחלתי על גבי הנתונים (וכך תראו בעין כמה זה מתאים).
 - בחרו קצב למידה קטן בהרבה מ 1, ומספר איטרציות מקסימלי לתנאי עצירה. שימו לב: המטרה היא לפתור כראוי את בעיית האופטימיזציה, כך שתצטרכו לבחור בצורה מושכלת את שני הערכים האלה אחרי שתסתכלו על התוצאות והגרפים המתקבלים.
 - כתבו לולאה לירידה במורד הגרדיאנט, שתמשיך עד הגעה לתנאי העצירה של מספר איטרציות מקסימלי.
 - שימו לב שבכל איטרציה השגיאה מחושבת כסכום על פני כל הנתונים. חשבו בכל איטרציה גם את גודל השגיאה (לצורך הצגה אח"כ).
 - שימרו בכל איטרציה את ערכי הפרמטרים (לצורך הצגה אח"כ).
 - בסיום הלולאה הציגו:
1. הדפיסו בתוכנית, @ ורשמו במסמך התשובות את ערכי הפרמטרים הסופיים שנמצאו.
 2. הציגו בגרף את המודל עם הפרמטרים הסופיים (קו ישר) על גבי הנתונים להמחשה של ההתאמה. הקפידו על יחס צירים 1:1 כך שאם מוצג הקו $y=x$ הוא יראה בשיפוע של 45 מעלות. @ צרפו את הגרף למסמך התשובות.

3. הציגו בגרף את ערך השגיאה כפונקציה של מספר האיטרציה. @ צרפו את הגרף למסמך התשובות.

4. חשבו את משטח השגיאה מעל הפרמטרים והציגו אותו בגרף תלת ממדי. הוסיפו להצגה את ערכי הפרמטרים שנמצאו במהלך כל האיטרציות. שימו לב לבחור את גבולות ההצגה (טווח ערכי הפרמטרים) כך שצורת המשטח תתגלה בצורה ברורה. השתמשו בכמה זוויות הצגה ובכמה בחירות של גבולות הצגה כדי להמחיש היטב את צורת המשטח ואת מסלול האיטרציה על פניו. @ צרפו את הגרפים למסמך התשובות.

ב. שכפלו את הקוד של הסעיף הקודם והשתמשו בו לכתיבת הקוד בסעיף זה. המירו את החישובים הידניים לחישובים אוטומטיים בעזרת הכלים הסימבוליים הנחיות:

- הגדירו משתנים סימבוליים x, y, a, b .
- הגדירו את המודל f במונחים של המשתנים הסימבוליים.
- הגדירו את השגיאה עבור נתון יחיד $e = (f - y)^2$.
- הגדירו את מרכיבי הגרדיאנט על ידי גזירה סימבולית של השגיאה לפי a , ולפי b .
- הגדירו את השגיאה הכוללת בעזרת סכום השגיאות על כל הנתונים.
- השתמשו בכל המרכיבים הללו כדי לחשב את הדרוש בקוד בשלבים השונים.

לפניכם הדגמה קצרה לשימוש בכלים סימבוליים:

```
import numpy as np
from sympy import symbols, lambdify, sin, diff
import matplotlib.pyplot as plt

x_data = np.linspace(-4, 4, 30)
x, y, a = symbols('x, y, a')

f = a*sin(x)
# create a function that can operate on numpy arrays
F = lambdify([a, x], f, 'numpy')

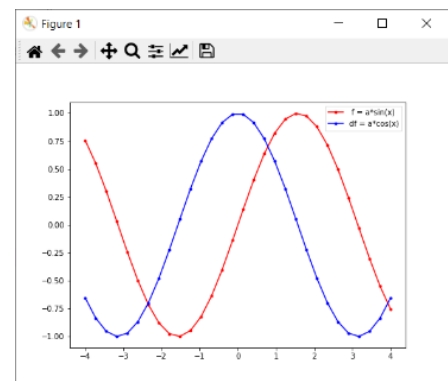
# apply the function to the data
a0 = 1
Y = F(a0, x_data)

plt.figure(1, figsize=(8, 6))
plt.plot(x_data, Y, '.-', color = 'red', label=' f = ' + str(f))

# the symbolic derivative
df = diff(f, x)
# apply the derivative to the data
DF = lambdify([a, x], df, 'numpy')

DY = DF(a0, x_data)

plt.plot(x_data, DY, '.-', color = 'blue', label='df = ' + str(df))
```



- בדקו שתוצאות הקוד המשתמש בחישובים הסימבוליים זהות לזה של הסעיף הקודם. הקוד צריך לייצר את אותן הצגות ואותם גרפים כמו בסעיף א' אך אין צורך לצרף אותם למסמך הפתרון.

ג. שכפלו את הקוד של הסעיף הקודם והשתמשו בו לכתיבת הקוד בסעיף זה. כתבו קוד המוצא את הפרמטרים של מודל בעזרת gradient descent וחישובים סימבוליים עבור המודל $f(x) = a \sin(bx)$, שגיאה של סכום ההפרשים בריבוע $E = \sum_i (f(x_i) - y_i)^2$, והנתונים:

```
x = [-5., -4.5, -4., -3.5, -3., -2.5, -2., -1.5, -1., -0.5, 0.,
      0.5, 1., 1.5, 2., 2.5, 3., 3.5, 4., 4.5, 5.]
y = [-2.16498306, -1.53726731, 1.67075645, 2.47647932, 4.49579917,
      1.14600963, 0.15938811, -3.09848048, -3.67902427, -1.84892687,
      -0.11705947, 3.14778203, 4.26365256, 2.49120585, 0.55300516,
      -2.105836, -2.68898773, -2.39982575, -0.50261972, 1.40235643,
      2.15371399]
```

שימו לב:

- המודל הוא \sin - רציף וחלק. על כן לצורך הצגת המודל יש לדגום את ציר ה X בצפיפות רבה יותר (מאשר הנתונים שעל פי הם מתבצעת ההתאמה).
- חשוב למצוא תנאי התחלה טובים, וטווח ערכים מתאים להצגת משטח השגיאה. יש לבחור את גבולות ההצגה (טווח ערכי הפרמטרים) כך שצורת המשטח תתגלה בצורה ברורה. השתמשו בכמה זוויות הצגה ובכמה בחירות של גבולות הצגה כדי להמחיש היטב את צורת המשטח ואת מסלול האיטרציה על פניו.

@ צרפו למסמך התשובות את שלושת הגרפים (בדומה לסעיף א' של שאלה זו).
 @ צרפו למסמך התשובות את הפרמטרים שמצאתם, ואת קצב הלמידה ומספר האיטרציות בהם השתמשתם.

ד. השתמשו ב `optimize.curve_fit` של החבילה `scipy` כדי לפתור עבור המודל והנתונים של הסעיף הקודם.

@ צרפו תצוגה של המודל עם הפרמטרים שנמצאו על ידי השיטה כשהם מוצגים על גבי הנתונים, כך שאפשר יהיה לבחון בעין את טיב ההתאמה.
 @ צרפו למסמך התשובות את הפרמטרים שמצאתם בשיטה זו.
 @ בחנו את התוצאות והסבירו.

ה. הציעו מודל לא לינארי בפרמטרים המכיל 3 פרמטרים ומעלה לפי בחירתכם. חובה לקבל באימייל אישור מן המרצה על המודל שאתם בוחרים.

ייצרו נתונים סינתטיים מורעשים (למשל על ידי הוספת רעש המתפלג גאוסיאני) עבור מודל זה. מצאו את הפרמטרים של המודל בעזרת gradient descent וסכום הפרשים בריבוע. הסתמכו על הקוד של סעיף ג'. מומלץ לייצר קוד כללי ונוח לשימוש.

@ במסמך התשובות הציגו את המודל שבחרתם:

מה המודל, ומה ערכי הפרמטרים הראשוניים.

@ צרפו למסמך התשובות גרף ובו הנתונים שייצרתם לפני ואחרי הוספת הרעש, (בקוד הקפידו על שימוש בגרעין הגרלה קבוע כך שהתוצאות יהיו זהות בכל פעם).

@ צרפו למסמך התשובות את הגרפים בדומה לסעיף א' של שאלה זו, מלבד ההצגה של משטח השגיאה.

@ הציגו בגרף את שני המודלים (המודל עם הפרמטרים של GD והמודל עם הפרמטרים של curve_fit) על גבי הנתונים – הנקודות המורעשות. שימו לב לבחירת תנאי ההתחלה בשתי השיטות.

@ הציגו את הפרמטרים שמצאתם בעזרת gradient descent לאלה שמחזירה curve_fit.

@ הציגו את הזמן שלוקח למצוא את הפרמטרים בכל שיטה.

@ תארו את ההבדלים בין שתי השיטות על פי מה שהצגתם בשלושת הסעיפים הקודמים והסבירו מדוע לדעתכם יש הבדלים (אם אכן יש).

בנוסף (עד 5 נקודות): ממשו דרך להצגת משטח (מרחב?) השגיאה, ועליו את הפרמטרים שנמצאו בכל האיטרציות. הסבירו את השיטה והציגו את התוצאות.

חלק שני: בחירה מבין הנושאים הבאים - ענו על אחת מהשאלות 2-4

השאלות בחלק זה שונות זו מזו באופיין, בדרך הפתרון ובמה שיש להגיש. **בשאלה 2** עליכם להפגין עצמאות ויצירתיות. השאלה מנוסחת בצורה כללית ועליכם להשלים את הפרטים. הציון יינתן על פי טיב התוצאות ורמת ההעמקה שלכם: בניתוח הבעיה, בחיפוש פתרונות, במימוש פתרון, בניתוח התוצאות, ובהצעת רעיונות להמשך. **בשאלה 3** עליכם להפגין דיוק ויכולת תכנות גבוהה. השאלה מנוסחת במפורש ועליכם לעקוב אחר ההנחיות ולבצע אותן כלשונן. **בשאלה 4** עליכם להתמודד עם בעיה מאתגרת גם בהיבטים הטכניים וגם בהיבטים של ההבנה. בחרו את השאלה על פי הנטייה והיכולות שלכם, ועל פי הנושא שתרצו להעמיק בו.

(שאלה 2)

שימוש באשכול (clustering) לעזרה בביצוע קלסיפיקציה

בתרגילים האחרונים השתמשנו בסט הספרות הכתובות בכתב יד. הפעלנו עליהם אלגוריתמים שונים לביצוע קלסיפיקציה, מציאת מאפיינים אוטומטית וחיפוש אבות טיפוס בדרכים שונות. כאמור, אם נתוני כל מחלקה מקורם בדגימה מתוך התפלגות גאוסיאנית ספרית (spherical) אזי אב טיפוס מועיל לכל מחלקה יהיה מרכז המחלקה. אך הנחה זו לא מתאימה לסט הספרות הכתובות בכתב יד. לכל אחת מן המחלקות יש כמה ווריאנטים עיקריים, ועל כן יש צורך בתיאור סטטיסטי יותר מתוחכם כדי לזהות בצורה טובה את הווריאנטים ולהשתמש בהם בהצלחה במשימת סיווג.

בשאלה זו עליכם להשתמש בסט הספרות בכתה יד 28x28 המכונה mnist_784. ראו

[Visualization of MLP weights on MNIST — scikit-learn 1.1.1 documentation](#)

<https://www.openml.org/search?type=data&sort=runs&id=554&status=active>



המשימה

עליכם לפתח מערכת המשלבת אישכול (מציאת אשכולות \ צברים \ clustering) וסיווג ולבחון את ביצועיה. הרעיון הוא להפעיל אלגוריתם אישכול על כל מחלקה בפני עצמה ואחר כך להשתמש במידע המתקבל ככלי עזר לסיווג 10 המחלקות.

נקודות למחשבה והתייחסות:

א. עיבוד מקדים \ נורמליזציות.
במה כדאי להשתמש?

ב. באיזה אלגוריתם אישכול כדאי להשתמש?
עליכם לקרוא על האלגוריתמים השונים (למשל בתיעוד של scikit-learn אבל רצוי גם במקורות אחרים), להבין את היתרונות והחסרונות שלהם ולבחור מתוכם.
כדאי לנסות כמה אלגוריתמים ולתעד את הביצועים שלהם כך שתוכלו לתמוך בהחלטה שלכם ולהסביר אותה.

ג. האם כדאי לבחור את אותו מספר אשכולות לכל מחלקה או לא?
נסו, תעדו, והסבירו.

ד. כיצד המידע המושג מאלגוריתם האשכול ינוצל לצורך הסיווג?
איזה מידע יעבור?
כאן אפשר לחשוב על כמה רעיונות. הציעו שיטה/ות ונסו.

ה. איזה אלגוריתם סיווג מתאים למערכת הנבנית. אולי זה יהיה תלוי בדרך בה אתם מעבירים את המידע בסעיף הקודם.
אם כן, יתכן ויש מסלולים אלטרנטיביים של: שיטת אישכול --> העברת מידע --> סיווג.

ו. כיצד לבחון את המערכת?
ראוי לפתח אמת מידה בסיסית לביצועים ללא תרומת האשכול, ואז אפשר יהיה להשוות אליה את הגרסאות השונות שתפתחו.

ז. ובהמשך לסעיף הקודם: באילו המחשבות ויזואליות כדאי להשתמש להצגת המערכת, תוצאות הסיווג ומדדי הביצוע ?

ח. איך יש לחלק את סט הנתונים לצורך הפיתוח של השיטה, הערכות הביניים של הביצועים והתוצאות, ולבסוף ההערכה המסכמת שבסופה תוכלו לומר מה טיב ביצועי השיטה.

ט. איזה שיפורים נוספים ורעיונות להמשך להציע לאחר סיום המשימה?

שימו לב שסדר הנקודות כאן אינו בהכרח הסדר בו תרצו להתקדם, וכן שבוודאי יש עוד נקודות למחשבה והתייחסות במשימה זו.

@ צרפו למסמך הפתרון הסבר מלא הכולל את השלבים שביצעתם, ההחלטות שלקחתם, מקורות המידע שהשתמשתם בהם, הניסיונות שניסיתם, התוצאות שקיבלתם, ניתוח התוצאות והרעיונות להמשך.

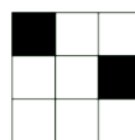
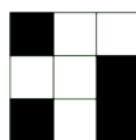
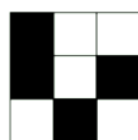
@ הגישו את כל הקוד הדרוש לפתרון השאלה.

(שאלה 3)

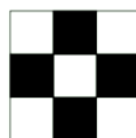
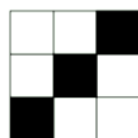
מרחב הפתרונות

בתרגיל 1 הוצגה חידה הממחישה את העובדה שכאשר יש מעט דוגמאות בסט האימון (ביחס לגודל מרחב הפתרונות), קיימים הרבה פתרונות קונסיסטנטיים עם סט האימון שעשויים להיות סותרים זה את זה.

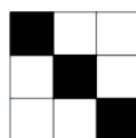
להזכירכם, הנה החידה (הפעם התוצאות הן True או False):



$f = \text{True}$



$f = \text{False}$



$f = ?$

כל אחד מן הנתונים במשימת הסיווג שבחידה הוא מטריצה 3 על 3 ובה ערכים בינאריים. מספר המטרices האפשריות השונות זו מזו הוא $2^9=512$, וזה גם גודל מרחב הקלט. זהו מספר יחסית קטן, אך גדול בהרבה ממספר הנתונים בסט האימון המוצג בחידה.

בשאלה זו תחקרו קצת יותר לעומק את מרחב הפתרונות למשימת הסיווג שבחידה.

א. @ בהמשך לחידה שהוצגה בעמוד הקודם הגישו בקובץ התשובות: 3 פתרונות המסווגים את המטריצה שמתחת לקו בתור True, ו 3 פתרונות המסווגים את המטריצה שמתחת לקו בתור False. הנחיות:

- כל פתרון חייב לסווג נכונה את כל סט האימון.
- הפתרון יוצג על ידי ביטוי לוגי שתוצאתו True או False.
- הביטוי הלוגי יכלול אך ורק ערכי פיקסלים, והאופרטורים הלוגיים: not, and, or, וסוגריים. אסור להשתמש באופרטורים אחרים.
- פיקסל שחור משמעו True, פיקסל לבן משמעו False.
- הפיקסלים ימוספרו כך:

p1	p2	p3
p4	p5	p6
p7	p8	p9

הנה דוגמה לכלל לוגי המסווג נכונה את כל סט האימון:

p1

הנה דוגמה הממחישה את צורת הכתיבה לכלל לוגי מורכב:

$(p1 \text{ and } p2) \text{ or } (p3 \text{ and not } p4)$

ב. @ כמה כללים מסווגים נכונה את סט האימון?

הסבירו בקובץ התשובות את החישוב במפורש.


(מומלץ להסתכל על הסרטונים Welch Labs | Learning to see, ובמיוחד חלק 7.)

[Learning To See - YouTube](#)

ג. כתבו תוכנית פייתון ובה חמישה חלקים:

1. חלק המגריל באקראי כלל לוגי (בפורמט שתואר בסעיף א).

2. חלק הבודק האם הכלל המוגרל תואם את סט האימון שהוצג בתחילת השאלה, ואם כן קובע שהכלל חוקי.

3. חלק המפעיל כלל חוקי על הדוגמה .

4. לולאה המשתמשת בחלקים הקודמים למציאת 20 כללים המניבים True על הדוגמה  ו-20 כללים המניבים False על אותה דוגמה.

5. כתיבה של התוצאות לקובץ טקסט על פי ההנחיות האלה:

- שם הקובץ יהיה rules.txt.
 - השורה הראשונה בקובץ תהיה
- ```
Rules that give True on test
```
- אחריה יופיעו 20 הכללים המניבים true על צורת המבחן, כאשר כל כלל כתוב בשורה אחת בלבד.
  - השורה ה-22 בקובץ תהיה
- ```
# Rules that give False on test
```
- אחריה יופיעו 20 הכללים המניבים false על צורת המבחן, כאשר כל כלל כתוב בשורה אחת בלבד.
 - לא יהיו שורות ריקות בקובץ.

הנה המחשה לצורת הקובץ הרצויה עם 2 כללים לכל קבוצה:

```
# Rules that give True on test
p1
p1 and not p2
# Rules that give False on test
p1 and not p5
p1 and p6
```

אתם יכולים לבחור אם להשתמש בסוגריים או לא. ללא סוגריים הביטויים הלוגיים יפורשו על פי סדר הקדימות של האופרטורים.

קובץ הפלט שלכם ייבדק בצורה אוטומטית כך שעליכם לייצר אותו בדיוק בפורמט שתואר.

@ הגישו את כל הקוד הדרוש לפתרון השאלה, ואת קובץ הפלט שהקוד שלכם מייצר.

(שאלה 4)

שימוש ב ICA להפרדת מקורות אודיו ואז למידה לצורך קלסיפיקציה

בשאלה זו תשתמשו ב Independent Component Analysis – ICA להפריד בין אותות קול שתערבבו בצורה סינטטית, ואח"כ תסווגו את האותות בעזרת אלגוריתם קלסיפיקציה שתפתחו.

בספריה scikit-learn נמצא מימוש ידוע של אלגוריתם ICA שנקרא FastICA:

<http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.FastICA.html>

http://scikit-learn.org/stable/auto_examples/decomposition/plot_ica_blind_source_separation.html

א. הורידו דוגמאות קול
עליכם להוריד את הקבצים הבאים

<http://research.ics.aalto.fi/ica/cocktail/source1.wav>

<http://research.ics.aalto.fi/ica/cocktail/source2.wav>

<http://research.ics.aalto.fi/ica/cocktail/source3.wav>

<http://research.ics.aalto.fi/ica/cocktail/source4.wav>

<http://research.ics.aalto.fi/ica/cocktail/source5.wav>

<http://research.ics.aalto.fi/ica/cocktail/source6.wav>

על קריאת קבצי wav בפייתון ניתן לקרוא כאן:

<https://docs.python.org/3/library/wave.html>

ב. הציגו את הספקטרוגרמה (spectrogram) של כל אחד מן האותות.

ראו למשל כאן

[Plotting a Spectrogram using Python and Matplotlib | Pythonic.com](#)

[Hands-On Tutorial on Visualizing Spectrograms in Python \(analyticsindiamag.com\)](#)

[scipy.signal.spectrogram — SciPy v1.8.1 Manual](#)

ג. הגרילו מטריצת ערבוב אקראית (סדר 6×6) מהתפלגות אחידה של ערכים בין 0.5 ל 2.5.

ד. צרו אותות מעורבבים ממקורות האודיו על ידי הכפלתם במטריצת הערבוב. שימרו קבצי wav של האותות המעורבבים לצורך השמעה ואיתור תקלות.

ה. הפעילו ICA כדי להפריד את 6 האותות המעורבבים ל 6 אותות משוחזרים.

ו. שימו לב שאלגוריתם ICA אינו יכול לשחזר את סדר האותות, ולא את ה scale שלהם. נרמלו את האותות המשוחזרים כך ש א. ממוצע כל אות יהיה 0. ב. כל אות יהיה וקטור יחידה. גם כאן שימרו את התוצאות לקבצי wav לצורך השמעה.

ז. הציגו את הספקטרוגרמה (spectrogram) של כל אחד מן האותות המשוחזרים המנורמלים.

כעת לסיווג. כאן נדמה מקרה אמיתי בו יש לנו דגימות קול קצרות ולהן תיוג ידוע, ודגימות קול קצרות אחרות אותן אנו רוצים לסווג.

ח. ברשותכם רק 6 האותות המקוריים ועליכם להחליט כיצד להשתמש בהם ליצירת סט אימון וסט מבחן. הרעיון העקרוני הוא לחתוך כל דוגמה לכמה קטעים קצרים וכך לייצר את הסטים.

ט. הציעו כמה מאפיינים למשימת הסיווג. המירו למרחב מאפיינים, אמנו מסווג ובדקו את איכות הסיווג על סט המבחן.

י. שלב אחרון: השתמשו במסווג שפיתחתם לסיווג של האותות המשוחזרים משלב ו.

בנוסף: ביצוע השאלה על 6 קבצי האודיו.
ללא בנוסף: ביצוע על 4 קבצי אודיו מתוך ה 6 לבחירתכם.

@ צרפו למסמך הפתרון :

א. הסבר מלא של השיטה שלכם לביצוע הסיווג, כולל ההחלטות על בניית הסטים השונים, ובחירת המאפיינים.

ב. תצוגות הספקטרוגרמות שקיבלתם.

ג. הצגת תוצאות ההפרדה ודיון ביקורתי בתוצאות: האם ICA מצליח להפריד את 6 האותות כראוי?

ד. הצגת איכות הקלסיפיקציה על האותות המקוריים ודיון ביקורתי בתוצאות.

ה. הצגת איכות הקלסיפיקציה על האותות המשוחזרים ודיון ביקורתי בתוצאות.

ו. הציעו שיפורים להמשך פיתוח או מחקר, והסבירו אותם ברמה שתאפשר מימוש.

ז. הפניה למקורות בהם השתמשתם מעבר לאלה שבתרגיל.

@ הגישו את כל הקוד הדרוש לפתרון השאלה.

@ צרפו את קבצי ה wav שהקוד שלכם יצר.

נספח כלים שימושיים

1. בתרגיל נשתמש בחבילה לפתרון סימבולי SymPy

[SymPy 1.12 documentation](#)

התקינו את החבילה (לגבי התקנות הסתכלו שוב בתרגיל 0)

```
conda install sympy  
conda update sympy
```

להכרות עם היכולות הרבות מאוד של SymPy הסתכלו ב [Introductory Tutorial - SymPy 1.12 documentation](#)

2. . לעניין חישובי הנגזרות והכלים הסימבוליים מומלץ להסתכל גם במקורות האלה

[Maxima vs Minima and Global vs Local | My Journey with Deep Learning and Computer Vision \(wordpress.com\)](#)

[Gradient Descent Optimization \[Part 1\] | My Journey with Deep Learning and Computer Vision \(wordpress.com\)](#)

[sympy | My Journey with Deep Learning and Computer Vision \(wordpress.com\)](#)

3. בשאלה 1 בתרגיל עליכם למצוא את הפרמטרים המתאימים למודלים שונים על ידי קוד לאופטימיזציה שתכתבו על פי ההנחיות.

כמובן ששיטות אופטימיזציה ועוד הרבה יותר נמצאים בחבילות השונות של פייתון. ראויה לציון החבילה SciPy – אוסף חבילות פייתון לחישוב מדעי על היבטיו השונים.

ראו במיוחד

[Scipy Lecture Notes — Scipy lecture notes \(scipy-lectures.org\)](#)

[scipy.linalg.lstsq — SciPy v1.12.0 Reference Guide](#)

[Fitting to polynomial — Scipy lecture notes \(scipy-lectures.org\)](#)

[scipy.optimize.curve_fit — SciPy v1.12.0 Reference Guide](#)

[Least squares fitting with Numpy and Scipy \(mmas.github.io\)](#)

4. בהצגה תלת ממדית לרוב צריך לבחור זווית תצוגה מתאימה בה רואים היטב את מה שמנסים להציג. אני מזכיר שכדי לקבל תצוגה אינטראקטיבית בה אפשר לשנות זווית הצגה יש לרשום ב prompt של Spyder

%matplotlib auto

הגשת הפרויקט

- א. תאריך הגשה:
למעוניינים ללמוד בקורס "מעבדת תכנות מתקדמת ב AI"
עד יום ראשון, 24.3.24, בשעה 23:55.
לשאר הסטודנטים
עד יום ראשון, 31.3.24, בשעה 23:55.
- ב. ניתן להגיש ביחידתם, בזוגות או בשלשות.
המעוניינים להגיש בשלשה יפנו אלי לקבלת אישור ומשימה נוספת.
- ג. יש לכתוב **שם \ שמות + ת"ז** בראשית כל מסמך מוגש (**כולל בקבצי הקוד**).
- ד. כל מגיש (ביחיד או בצוות) צריך לדעת להסביר כל מה שנעשה בפתרון המוגש. חלק מן המגשים ידרשו להסביר את הפתרון שלהם למרצה.
- ה. יש להגיש מסמך Word המכיל את כל התשובות לתרגיל. שם מסמך זה יהיה **project.docx**.
הקפידו שמספור סעיפי התשובות שלכם יהיה זהה למספור סעיפי השאלות.
- ו. לכל פונקציה צריך להיות תיעוד.
- ז. יש להגיש את כל הקוד לתרגיל במספר קבצים לפי בחירתכם.
- ח. כל הקבצים ישכנו בתוך תיקייה הכוללת את שמכם.
שם התיקיה למגיש יחיד:
ProjectFamilyName
שם התיקיה לשני מגשים:
ProjectFamily1Family2
התיקיה תארז לקובץ **zip** בעל אותו שם כשל התיקיה.

בהצלחה!