ביולוגיה חישובית תרגיל 3
רעות לב 207385741 יעלה גרנות 209133107

<u>הוראות הרצה לקוד :</u>

1) **הרצה ידנית משורת הפקודה** - לחלק א: הרצת הפקודה python Buildnet0.py בשורת הפקודה ללא צורך בהתקנות נוספות. יש לוודא שקיימת באותה תיקייה את קבצי הדאטה "train0.txt" ו "test0.txt" .
לחלק ב: הרצת הפקודה python rurnnet0.py בשורת הפקודה ללא צורך בהתקנות נוספות. יש לוודא שקיימת באותה תיקייה את קבצי הדאטה testnet0.txt ו wnet0.json.

(עבור קובץ הקלט 1nn יש לבצע שנית את התהליך הנ"ל - אך כמובן עם הספרה "1" במקום "0").

2) **הרצת exe** - לחלק א: שני קבצי הרצה בשם"Buildnet0_1.exe" ו"Buildnet0_2": פתיחת הקישור לgit המצורף להגשה (https://github.com/reutlev98/ComputationalBiology-Ex3) והורדת שני קבצי הzip הללו ואיחודם לתייקיה אחת משותפת. לאחר מכן יש להריץ את קובץ הexe ע"י דאבל קליק. בתום הריצה קבצי הפלט "wnet0.json" יופיעו בתיקייה של קובץ הexe יחד עם קבצי הקלט.
לחלק ב: קובץ הרצה בשם"runnet0.exe" : יש להכניס לתיקיית ההרצה את קובץ "testnet0.txt" , קובץ "wnet0.json" יופיע בתיקייה.
שם קובץ הפלט של התוכנית יקרא "output0.txt" ובו ניתן לראות את הסיווגים.

(עבור קובץ הקלט 1nn יש לבצע שנית את התהליך הנ"ל - אך כמובן עם הספרה "1" במקום "0").

★ לצורך הנוחות, קבצי הדאטה של הטסט והאימון נמצאים בתיקייה המתאימה עבור הרצת תוכניות הBuildnet וכן קובץ הtestnet להרצת הrunnet. במידה ותרצו להשתמש בקבצי קלט אחרים יש להחליף אותם עם הקבצים הללו.

<u>ייצוג המשקולות:</u>

עבור שתי התוכניות ייצגנו את המשקולות שנמצאו כהכי טובות כאובייקט JSON המכיל שני שדות "best_solution" ו "structure". כאשר הstructure הוא מערך שמציין את מספר הנוירונים בכל שכבה ברשת. למשל "structure": [16, 2, 1]" מסמל רשת נוירונים בעלת שכבת ביניים אחת כאשר שכבת הקלט שלה מכילה 16 נוירונים שכבת הביניים 2 נוירונים ושכבת הפלט מכילה נוירון אחד.
ה best_solution הוא מערך המכיל את ערכי המשקולות השונים בין השכבות.

<u>תיאור אלגוריתם Buildnet למציאת משקולות הרשת</u>

- **גודל האוכלוסייה** 100.
- **מספר דורות** 150. בפועל ניתן לראות שעד מספר דורות של 80 יש עלייה משמעותית בקצב גבוה, ומדור 80 ומעלה יש עלייה מאוד מתונה מ0.97 עד 0.99.
- **מבנה הרשת**: קיבענו את מבנה הרשת להיות עם 3 שכבות ביניים עם מספר נוירונים עולה של 6,8,10 בהתאמה. השתמשנו בפונקציית אקטיבציה של Relu למציאת סיווג של דאטה לא לינארי.
- **אתחול האוכלוסייה:** כל פרט באוכלוסייה הוא אובייקט של מחלקת network המכילה שדה structure שהוא מערך המציין את מבנה הרשת ושדה של משקולות שעליהן הופעל האלגוריתם הגנטי.

- **אתחול המשקולות:** ערכי המשקולות הוגרלו בצורה רנדומלית תוך שימוש בפונקציית אתחול xavier מהתפלגות אחידה.
כאשר המכנה סוכם את מספר הנוירונים בשכבת הקלט והפלט.

$$[-x,x] \text{ for } x = \sqrt{\frac{6}{inputs + outputs}}$$

- **חישוב fitness**: ציון הfitness מודד את הaccuracy של הרשת, כלומר את מספהסיווגים שכל רשת הצליחה לחזות נכון ביחס לסך כל הדוגמאות. בשלב החיזוי הכפלנו את המשקולות של כל פרט באוכלוסייה בכל אחת מהדוגמאות ,וספרנו את מספר הסיווגים שכל פרט סיווג נכונה ע"י השוואה של פלט המודל מול הסיווג האמיתי. הכרעת הסיווג בהינתן פלט המודל נקבע לפי ערך סף של 0.5 כך שבמידה והוא גדול מ0.5 הסיווג הוא 1, ואחרת 0.
- **פעולות גנטיות:**
  - מוטציות: קצב המוטציות באוכלוסייה נקבע להיות 0.15 , כלומר בממוצע רק 15% מערכי המשקולות של כל פרט שעובר תהליך של מוטציה יעברו שינוי.
    כדי להימנע מהתכנסות מוקדמת הגדרנו שעבור כל 4 דורות בהם לא חל שיפור בציון של האוכלוסייה, קצב המוטציות יועלה באותו דור ספציפית ל0.25.
    בתהליך המוטציה השתמשנו בהגרלה של ערך דלתא בין 0.1 ל 0.1- בהתפלגות אחידה והוספה של ערך זה למשקולת הנוכחית.
    לאחר ביצוע מוטציה השתמשנו בפקודת clip על מנת לוודא שערך המשקולת נותר בין 1- ל1.
  - crossover: בהינתן שני הורים יצרנו 2 צאצאים תוך שימוש בהכלאה אחידה (uniform crossover). בשיטה זו עבור כל ערך של משקולת ספציפית מגרילים האם הצאצא הראשון יירש אותו מהורה א' או ב' בהתפלגות אחידה,כאשר את ערך המשקולת מההורה השני שלא נבחר - יורש הצאצא השני.

- **יצירת הדור הבא:**
הרכב הדור חדש תלוי בדור הקודם באופן הבא:
  - אליטה - <u>10%</u> פרטים מהאוכלוסייה בעלי הציונים הטובים ביותר בדור הנוכחי מועברים ישירות לדור הבא.
  - מוטציות לאליטה - צירפנו לדור הבא עוד <u>10%</u> העתקים של האליטה שעברו תהליך של מוטציות וזאת מתוך כוונה להגדיל את השונות ולהתקדם לעבר הפתרון הנכון.
  - הכלאות -ביצענו 35 פעם בחירה אקראית של 2 הורים מתוך האליטה וביצענו בניהם הכלאות ולאחר מכן מוטציות, באופן הזה הוספנו לדור הבא עוד <u>70%</u> פרטים חדשים.
  - הגירה - יצירה של <u>10%</u> פרטים חדשים באוכלוסייה המגדילים את המגוון הגנטי ומונעים התכנסות מוקדמת.

- **אופטימיזציות** - אחת לעשר דורות השתמשנו באלגוריתם גנטי לפי שיטת למארק, שבו כל פתרון עובר תהליך של מוטציה כפולה ובדיקה שאכן מדובר באופטימיזציה. במידה וכן, נשמרו השינויים עבור אותו פתרון, אחרת הפתרון חזר לגרסתו המקורית. האוכלוסייה לאחר האופטימיזציה היא זו שיצרה את הדור הבא.

★ האלגוריתם שתיארנו למעלה מתאר עקרונית את תוכנית 0nn אך גם מתאים לתוכנית השנייה של 1nn מלבד השינויים הבאים:
  - מספר דורות 100.

- הרשת כללה שכבת ביניים אחת עם 2 נוירונים.
- פונקציית האקטיבציה עבור תוכנית זו היא sigmoid.

★ תוכניות runnet - ביצעו קריאה של קבצי הjson ולפיה ביצעו בנייה מתאימה של שכבות הרשת והמשקולות. פונקציית הפרדיקציה בתוכניות הrunnet הייתה זהה לאלו שבתוכניות הBuildnet בהתאמה.(גם מבחינת פונקציית האקטיבציה וגם בפירוש הפלט של הרשת וההמרה שלו לסיווג).

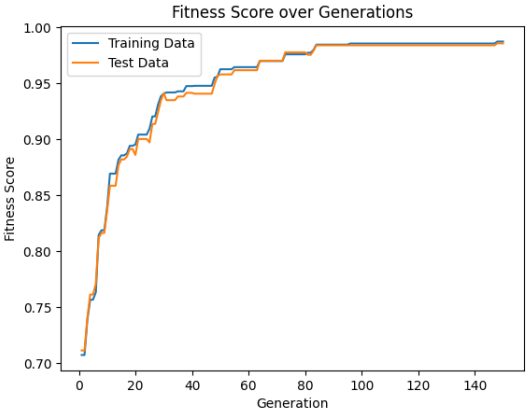
<u>תיאור ביצועים בשלב הלמידה ובשלב הבדיקות :</u>

לצורך מציאת המשקולות הטובות ביותר החלטנו לחלק את הדאטה שלנו ביחס של 80:20 לסט אימון וסט מבחן.

תיאור ויזואלי של התוצאות :

- Buildnet0:

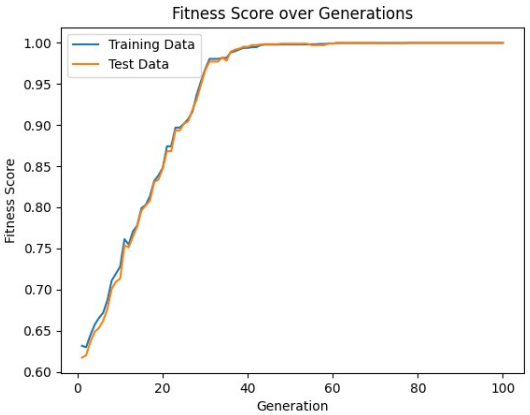
```
Generation number: 147 unchanged_gens:  50
Best fitness: 0.985625
Generation number: 148 unchanged_gens:  51
Best fitness: 0.987375
Generation number: 149 unchanged_gens:  0
Best fitness: 0.987375
Generation number: 150 unchanged_gens:  1
Best fitness: 0.987375
Test Accuracy: 0.98575
```



כמו שניתן לראות, גם סט האימון וגם סט המבחן הצליחו להגיע לפתרון בעל דיוק גבוה של מעל 98.5%, כאשר כפי שציפינו הדיוק בסט המבחן הוא מעט קטן יותר ,שכן המודל לא נחשף לסט המבחן ולא אומן עליו. לכן אפשר להגיד שהמודל שלנו מצליח לפתור את הבעיה בצורה טובה ומכלילה ומסוגל להימנע מoverfitting.

- Buildnet1:

```
Generation number: 99 unchanged_gens:  20
Best fitness: 1.0
Generation number: 100 unchanged_gens:  21
Best fitness: 1.0
Test Accuracy: 1.0
```



אפשר לראות שהצלחנו למצוא את המשקולות האופטימליות ולקבל תוצאות בדיוק מקסימלי של 100% גם על הסט אימון וגם על סט המבחן. אפשר לראות שבריצה זו הצלחנו להגיע לדיוק מירבי כבר בדור 78 אך החלטנו להשאיר את מספר הדורות המקורי (100) גם בגלל שהיו ריצות בהם הדיוק המירבי הגיע יותר מאוחר וגם בכדי להגדיל את הסיכוי לקבל ציון גבוה על סט המבחן ולהקטין את שגיאת ההכללה.

<u>חוקיות של התבניות בשני המקרים :</u>

nn0 : סיווג 1 עבור מספר אחדות בין 8 ל12 , ו0 אחרת.
nn1: סיווג 1 עבור מספר אחדות קטן מ8, ו0 אחרת.