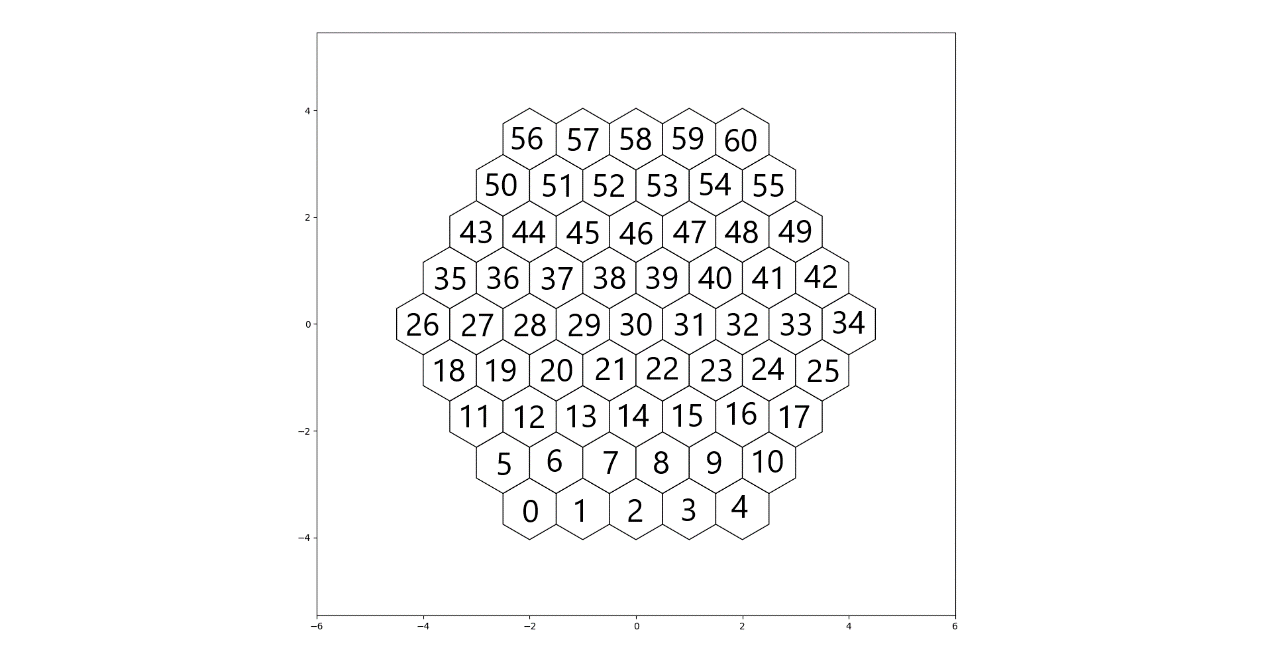
**ביולוגיה חישובית   
80-512 89-512**

**תרגיל 3 – רשתות נוירונים** SOM **מגישים (הגשה משותפת סטודנטים מביולוגיה חישובית ומדעי המחשב):   
 אייל הבר - 203786298 (ביולוגיה חישובית)   
אייל ברילינג - 208204230 (מדעי המחשב)**

בתרגיל זה התבקשנו ליישם **SOM** על תוצאות הבחירות בכנסת ה-24.

* לצורך הרצת התרגיל, יש לקרוא את קובץ ה-**README.txt** המצורף.
* קובץ הקלט **Elec\_24.csv** המצורף מכיל את תוצאות הבחירות לכנסת ה-24 עבור כ 200 ישובים.

**אופן יישום התרגיל**

בשלב הראשון יצרנו את מפת ההקסגונים עליה נרצה לעבוד, למקם בכל תא את הישובים המתאימים לו ביותר, לאחר מכן לצבוע אותם בהתאם לממוצע הדירוג הכלכלי ולבסוף להציגה.   
מפה זאת מכילה את הקסגונים עם אינדקסים באופן הבא:

**בהתאם למפת האינדקסים הנ"ל, בנינו פונקציה** initiate\_neighbors() **המייצרת מילון של רשימת אינדקס שכנים מתאימה עבור כל אינדקס**

**השלב הבא היה ביצוע טעינה של הקובץ בעזרת הפונקציה** load\_file().  
**פונקציה זאת תקרא את קובץ הקלט שורה אחרי שורה ותייצר וקטור ערכים מתאים בהתאם לתוצאות ההצבעה למפלגות באותו היישוב. לאחר מכן הוקטור יעבור "הערכה" ויקבל ציון מתאים בעזרת הפונקציה** get\_score(vector) **שתחשב את אחוזי ההצבעה לכל מפלגה בהתאם לכמות המצביעים הספציפית (על מנת לייצר נרמול מסוים בים ישובים בעלי מספר תושבים שונה באופן מהותי) כל ערך בוקטור יקבל ערך "מקדם" בהתאם להשקפה והנטייה הפוליטית של המפלגה אותה הוא מייצג (מפלגות עם השקפות דומות יקבלו את אותו ה"מקדם", מפלגות מנוגדות יקבלו "מקדמים" שונים יותר וכו').**

**לאחר מכן הרצנו 10 פעמים הרצה רנדומלית של ערכי ההקסגונים במפה בעזרת הפונקציה** generate\_random\_score\_for\_each\_hexagonal\_cell() **ואז יצירת** SOM **בהתאמה בעזרת הפונקציה** generate\_SOM()**. יישום אופן יצירת** **ה**-SOM **והצגת הפלט הסופי יפורט במענה על שאלות התרגיל.**

**מענה על שאלות התרגיל**

1. **איך חישבתם את המרחק בין ישוב בקלט לבין התא בגריד שמייצג אותו?**כאמור, כל תא מקבל ציון התאמה. אותה פעולה תתבצע גם על כל ישוב שנטען בקובץ. הישוב ישווה את ציון ההתאמה שהוא קיבל אם כל התאים על הגריד ויבחר את התא עם ציון ההתאמה בעל המרחק הכי קצר אליו, לאחר ריצה על כל התאים יישמר גם התא בעל הציון השני הכי קרוב (בשביל חישוב שגיאה טופולוגית TE בהמשך)
2. **באיזה אופן קירבתם את התא אליו מופה קלט מסוים לאותו קלט ובאיזה אופן שונו שכניו?**לאחר בחירת התא המתאים, ביצענו "קירוב" של ערך ציון ההתאמה בתא אל ערך ציון ההתאמה של היישוב שבחר בו, עשינו זאת בעזרת השמת ערך חדש המתקרב ב-30% אל ערך ציון היישוב: new = 0.7\*old + 0.3\*cityלאחר מכן נעזרנו במילון השכנים שבנינו בשביל למצוא את רשימת השכנים של אותו האינדקס, ושם על כל אינדקס ביצענו קירוב באופן דומה אך הפעם בשיעור של 20%. לבסוף פעלנו במעגל שני, כך שקירבנו גם שכן של שכן בקירוב של 10% (בשלב זה ווידאנו שאנו לא מקרבים מחדש ערך תא באינדקס שכבר קירבנו)
3. **האם זה משנה באיזה סדר מציגים את הישובים למערכת?**כן, לאחר בדיקות רבות, מצאנו כי כאשר אנו משווים SOM בסדר ישובים מקורי ו-SOM לאחר shuffle של הישובים מקבלים כי ה-error נמוך יותר ב-SOM שעבר shuffle.  
   ניתן לבדוק זאת בעזרת העברת מצב TEST\_MODE = False למצב TEST\_MODE = True

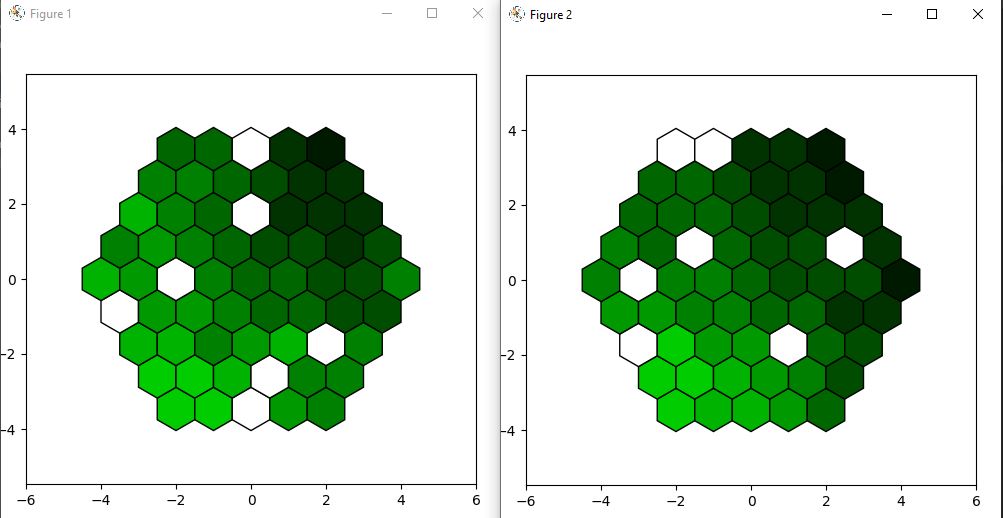
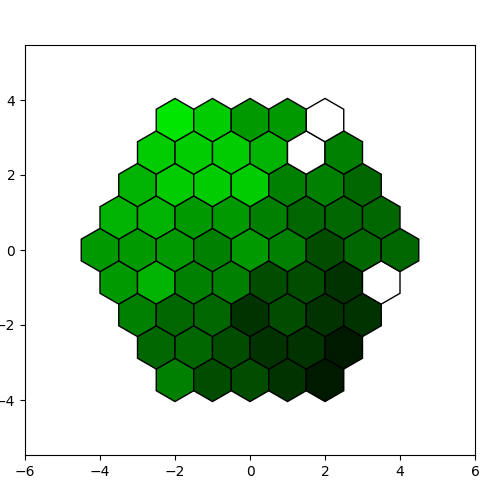
 מצב יציג לנו שני figures:

Figure1 מייצג את original city SOM error  
Figure2 מייצג את shuffled city SOM error נמוך יותר ולכן עדיף   
כלומר, הגרף הימני "טוב" יותר ולכן הגדרנו שה-SOM הגנרי שלנו יהיה במצב shuffled

1. **כיצד בחרתם את הפתרון להגשה מבין הפתרונות השונים?**לאחר שבחרנו מצב יצירת SOM ב-shuffled יצרנו 10 גרסאות SOM שונות.  
   מתוך 10 הגרסאות הללו בחרנו את ה-SOM עם ה-error הנמוך ביותר ואותו הצגנו ושמרנו**.  
   קובץ הריצה ייצר שני פלטים:**1. קובץ טקסט בשם: **Municipalities\_distribution\_in\_SOM.txt** המכיל רשימת ישובים בהתאמה לתא, לדוג':  
   תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי  
   2.קובץ תמונה **SOM.png** המציג את ה-SOM הנבחר, לדוג':

****

* **הגדרת צבעים:**הגדרנו את הצבעים כך שייצגו Economic Cluster כך שככל שהממוצע של הדירוג הכלכלי בין הישובים שנמצאים באותו תא משותף גבוה יותר (מעוגל כלפי מטה) ככה הגוון הירוק יהיה בהיר יותר ולהפך.

(תאים לבנים מייצגים תאים ריקים מישובים)

**מסקנה סופית**  
ניתן לראות בעזרת ניתוח גרף ה-SOM הנ"ל בהשוואה עם רשימת הישובים ומפת האינדקסים   
כי קיימת קורלציה ישירה בין גוון צבע התא המייצג דירוג כלכלי לבין הישובים אותם התא מייצג.