

שיעור 1 - תכנות פרספטרון בשפת Python

תרגיל 1:

שלב א'

ממשו מכונה לומדת מבוססת פרספטרון לסיווג הנתונים שבטבלה הבא:

data		label
Input A	Input B	Output
0	0	0
1	0	0
0	1	0
1	1	1

על הפעולה לעבור על כל אחד מהנתונים שבמבוא וללמוד אותו.

לצורך מימוש התרגיל נבצע את המשימות הבאות:

- בשלב הראשון נקלוט את נתוני הטבלה לתוך מערך numpy.
 - נפעיל את הפעולה train
 - נבחן את ביצועי המערכת תוך כדי עדכון הפרמטרים learningRate ו- epochs לרמה האופטימלית.
- לרשותכם המחלקה Perceptron כפי שלמדנו.

```
class Perceptron(object):
    def __init__(self, numOfInputs, epochs=200, learningRate=0.01):
        self.epochs = epochs
        self.learningRate = learningRate
        self.weights = np.zeros(numOfInputs)
        self.bios = 1

    def Activation(self, s):
        if s > 0:
            activation = 1
        else:
            activation = 0
        return activation

    def predict(self, inputs):
```

```

sum = np.dot(inputs, self.weights) + self.bios
out = self.Activation(sum)
return out

def train(self, inputs, labels):
    for _ in range(self.epochs):
        for i in range(len(inputs)):
            prd = self.predict(inputs[i])
            self.weights -= (prd - labels[i]) * inputs[i] * self.learningRate
            self.bios -= (prd - labels[i]) * self.learningRate

```

שלב ב'

למדו את פרספטרון לזהות את השערים באים: XOR NOT NOR OR
בדקו האם התוכנה מסוגלת ללמוד כל אחד מהם?

תרגיל 2: (פותרים את בעיית הקורונה בעולם!!!)

ממשו מכונה לומדת מבוססת פרספטרון לצורך ביצוע בדיקות קורונה תוך חלקי שניות בודדות.

לצורך כך במאמץ רב קיבלתם את מאגר הבדיקות הבא שישמש אתכם לצורך אימון הפרספטרון. להלן הנתונים:

איבוד חוש הריח	חום גוף מעל 38	נזלת	כאבי גוף	האם אובחן כחולה קורונה
כן	לא	לא	כן	כן
כן	לא	לא	לא	כן
לא	לא	כן	כן	לא
לא	כן	לא	לא	לא
כן	כן	לא	לא	כן
לא	כן	כן	כן	כן
לא	לא	לא	כן	לא
לא	לא	כן	לא	לא

ממשו תוכנה בשפת Python המבצעת אימון לפרספטורן על סמך הנתונים שהתקבלו. בשלב הבא על התוכנה לקלוט באופן מחזורי נתונים של נבדקים חדשים ולבצע להם בדיקת קורונה. על התוכנית להפסיק כאשר לא יהיו יותר נבדקים חדשים בקורונה בעולם (בקיצור כאשר נקלט 1 לתוך משתנה Exit)

להלן דוגמה לממשק המשתמש שהתבקשתם לממש:

```
Loss of smell (1=yes 0=no): 1
Body heat about 38 (1=yes 0=no): 1
Runny nose (1=yes 0=no): 1
Body pain (1=yes 0=no): 1
Exit (1=yes 0=no): 0
Coronavirus Positive!!!
Loss of smell (1=yes 0=no): 1
Body heat about 38 (1=yes 0=no): 1
Runny nose (1=yes 0=no): 0
Exit (1=yes 0=no): 0
Coronavirus Positive!!!
Loss of smell (1=yes 0=no): 0
Body heat about 38 (1=yes 0=no): 1
Runny nose (1=yes 0=no): 0
Body pain (1=yes 0=no): 0
Exit (1=yes 0=no): 0
Coronavirus Negative!!!
Loss of smell (1=yes 0=no): 0
Body heat about 38 (1=yes 0=no): 0
Runny nose (1=yes 0=no): 0
Body pain (1=yes 0=no): 0
Exit (1=yes 0=no): 0
Coronavirus Negative!!!
Loss of smell (1=yes 0=no): 0
Body heat about 38 (1=yes 0=no): 0
Runny nose (1=yes 0=no): 0
Body pain (1=yes 0=no): 1
Exit (1=yes 0=no): 1
```

תרגיל 3: (מכונה לומדת מבוססת פרספטורן לזיהוי תמונות)

לצורך זיהוי פריטים בתמונה או בקיצור זיהוי תמונות על ידי מכונה לומדת יש צורך לארגן מערך של תמונות מתוגיות. נבחן את הסוגיה על ידי התבוננות על התמונה הבא:



Image by [maja7777](#) from [Pixabay](#)

כיצד אם כן נאפיין מי זה הכלב ומי החתול בתמונה? האם כל הלבנים חתולים? האם כל בעלי הפרווה כלבים? האם לכל מי שיש אף בצורת משולש הוא חתול? האם כל מי שיש לו אוזן ורודה הוא כלב?

נראה שיש צורך לחדד את הנושא ולעזור לנו לאפיין את הגורמים המבדילים בין פריטים. תהליך זה נקראה במכונות לומדות: Feature Selection in Machine Learning והוא בא לספק לנו כלי להגדרת המאפיינים שיכולים להבדיל בין דברים שאנו רוצים לסווג.

במקרה שלנו כאשר מדובר בתמונות, אנו מתייחסים ליכולת לאפיין מספר תכונות בתוך תמונה במטרה לסווג פריטים בתוך תמונה ברמת וודאות גבוהה.

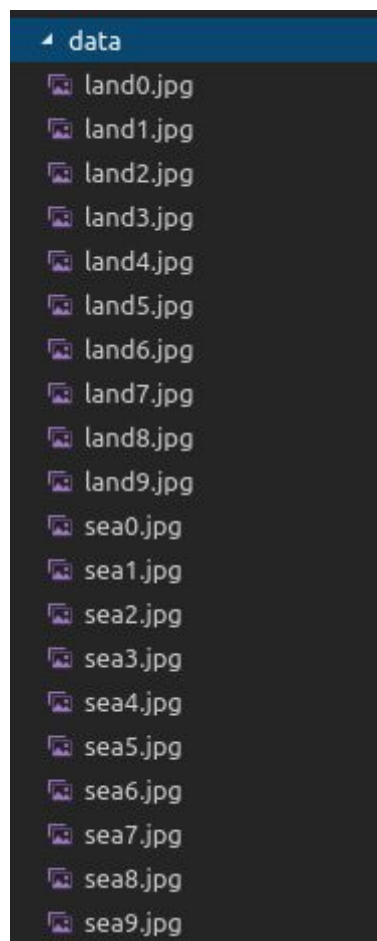
לצורך התנסות בנושא ובמטרה לתרגל עבודה עם תמונות נחפש תכונות Features שיכולים להבדיל בין תמונות ים לבין תמונות יבשה. בקיצור משחק ים יבשה בגרסת מכונה לומדת (נו טוב לא ממש דומה למשחק הילדים ששיחקנו פעם).

נוריד מהאינטרנט 10 תמונות של ים ו-10 תמונות של יבשה ונשמור אותם בתיקייה בשם data תחת השמות sea ומספר התמונה עבור כל התמונות המתארות ים ו-land ומספר התמונה עבור כל התמונות המתארות יבשה.

ניתן להוריד תמונות שאינם מוגנות מזכויות יוצרים מאתר:

<https://pixabay.com/>

להלן דוגמה למבנה הקבצים בתיקייה data הכולל את התמונות.



לאחר שהורדנו למחשב 20 תמונות נחפש מאפיינים שיכולים להבדיל בין תמונות נוף של ים ליבשה. האפשרות שנבדוק כמאפיין אפשרי יהיה היחס בין ממוצע הצבע הירוק בתמונות יבשה לבין יחס ממוצע הצבע הכחול בתמונות היבשה.

להלן דוגמה לתמונות:



Image by [Sasin Tipchai](#) from [Pixabay](#)

Image by [Michelle Maria](#) from [Pixabay](#)

שלב א'

כתבו קטע קוד המציג מתוך תמונה בודדת של ים ומתוך תמונה אחת של יבשה את ממוצע הצבע הירוק ואת ממוצע הצבע הכחול בכל אחת מהם.

עזרה:

לצורך קליטת תמונה ניתן להשתמש בקוד הבא:

```
img = Image.open("data/sea0.jpg")
img.load()
sea0 = np.array(img, dtype=np.uint8)
```

כדי לבדוק את ממוצע הצבע הירוק ניתן לכתוב את הקוד הבא:

```
print ("Green AVG =" , sea0[:, :, 1].sum() / sea0[:, :, 1].size)
```

האם זה הגיוני שהאלגוריתם מצא צבע ירוק בתמונה הים? נמקו את תשובתכם.

שלב ב'

בשלב הבא בנו 2 רשימות הכוללות את ממוצע הצבע הירוק בכל אחד מ-20 התמונות לפי הפרדה בין תמונות ים לתמונות יבשה.

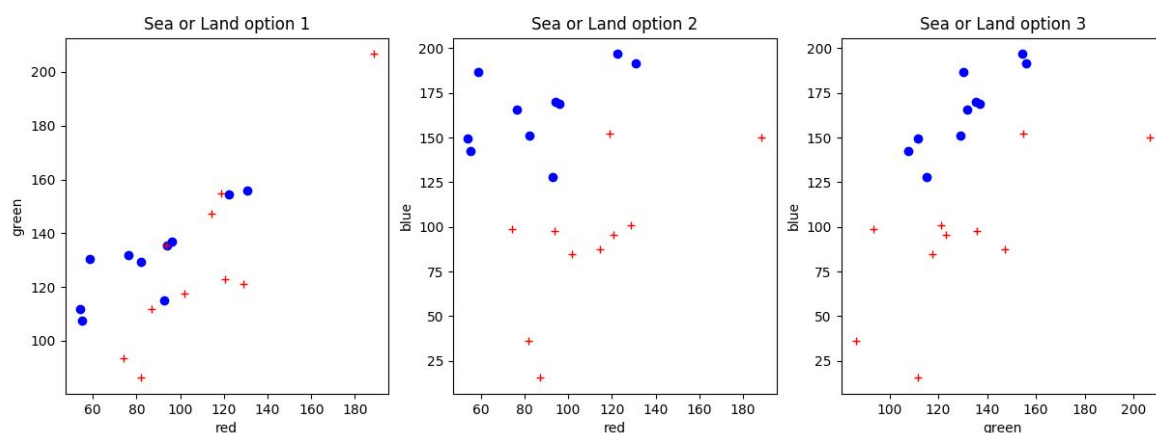
דוגמה לפלט אפשרי:

```
sea_list:
[130.4101615341768, 131.96297276468894, 154.34126688346825, 111.7069587628866, 115.01519280118855, 155.91324938555555, 129.21721614583333, 135.44244079172367, 137.0602974487363, 107.63084548611111]
land_list:
[121.23325928276738, 86.22154513888889, 147.22994865262988, 117.407721875, 122.99220265188997, 206.6399573089777, 154.72347739692003, 111.65416666666667, 135.62284365079364, 93.52566318926075]
```

שלב ג'

בשלב הבא הציגו גרפים המציגים את הנתונים שקיבלנו מהשלב הקודם אך הפעם נציג את היחס בין ממוצע כל 2 צבעים בכל תמונה לפי חלוקה בין תמונות ים לתמונות יבשה.

להלן דוגמה לפלט אפשרי:



מתוך הגרפים שקיבלתם בחרו את זוג הצבעים שנותן את ההפרדה הטובה ביותר בין תמונות הים והיבשה.

שלב ד'

ממשו אלגוריתם למידת מכונה מבוסס פרספטרון לביצוע סיווג של תמונות יום/יבשה תוך כדי ביצוע השלבים הבאים:

- ממשו את המחלקה perceptron
- צרו את מערך הנתונים המבוסס על התמונות.
- בצעו עיבוד לנתונים כדי לספק אותם במבנה מתאים ל- perceptron
- הפעילו את תהליך הלמידה של ה- perceptron ובדקו את איכות הלמידה על ידי מערך נתוני בדיקה. כלומר על מערך תמונות אחר מזה שנבדק בשלב האימון. דוגמה למערך תמונות בדיקה ניתן לראות באיור הבא:

