***דו״ח תרגיל 2 למידת מכונה***

עבור כל אלגוריתם בניתי פונקציה שתבדוק את הערכים של ההיפר-פרמטרים שמניבים את אחוזי

ההצלחה הטובים ביותר.

כאשר השתמשתי ב10 אחוזים מהקלט שנתון לי כvalidation ו90 אחוז מהקלט שלחתי לאימון של האלגוריתם. לאחר שקיבלתי את w חזרה מהמודל בדקתי על סט validation את אחוזי ההצלחה.

כך ביצעתי וערכתי כל פעם את הvalidation להיות 10 אחוזים אחרים מהקלט כדי לקבל ממוצע של אחוזי ההצלחה בין כל 10 הפעמים עבור ערכי ההיפר-פרמטרים ששלחתי .

עבור knnהאלגוריתם חיפש את הערך הטוב ביותר עבור ההיפר-פרמטר k

וקיבלתי את ממוצע אחוז ההצלחה הגבוה ביותר עבור k=3

the best percentage success is : 93.51851851851852% for k = 3

def k\_parm\_knn(x\_train, y\_train):  
 per\_valid = 10  
 len\_valid = int(len(x\_train)/per\_valid)  
 max\_success = 0  
 max\_k =1  
 for k in range(1, len(x\_train)):  
 i = 0  
 success = 0  
 while i < per\_valid:  
 begin = i \* len\_valid  
 end = begin + len\_valid - 1  
 tmp\_x = x\_train.copy()  
 tmp\_y = y\_train.copy()  
 validation\_x = tmp\_x[begin:end]  
 validation\_y = tmp\_y[begin:end]  
 for j in range(len\_valid):  
 tmp\_x = np.delete(tmp\_x, begin, axis=0)  
 tmp\_y = np.delete(tmp\_y, begin, axis=0)  
 for x,y in zip(validation\_x,validation\_y):  
 y\_hat = knn\_model(k, tmp\_x, tmp\_y, x)  
 if y == y\_hat:  
 success += 1  
 i += 1  
 success /= per\_valid  
 if success > max\_success:  
 max\_success = success  
 max\_k = k  
 print(f" the percentage success is : {(success/len(validation\_x))\*100}% for k = {k} ")  
  
 print(f" the best percentage success is : {(max\_success/len(validation\_x))\*100}% for k = {max\_k} ")  
 return max\_k

עבור אלגוריתם perceptron ניסיתי למצוא את הערכים הכי טובים עבור כמות epochs וערך הלמידה .

def parms\_percepteron(x\_train, y\_train):  
 per\_valid = 10  
 len\_valid = int(len(x\_train)/per\_valid)  
 max\_n2 =1  
 max\_epoch=1  
 max2\_success=0  
 for epoch in range(1, 200):  
 max\_n = 1  
 n = 0.01  
 max\_success = 0  
 while n < 1:  
 success = 0  
 i = 0  
 while i < per\_valid:  
 begin = i \* len\_valid  
 end = begin + len\_valid - 1  
 tmp\_x = x\_train.copy()  
 tmp\_y = y\_train.copy()  
 validation\_x = tmp\_x[begin:end]  
 validation\_y = tmp\_y[begin:end]  
 for j in range(len\_valid):  
 tmp\_x = np.delete(tmp\_x, begin, axis=0)  
 tmp\_y = np.delete(tmp\_y, begin, axis=0)  
 w = per\_train(n, epoch, tmp\_x, tmp\_y)  
 for x,y in zip(validation\_x, validation\_y):  
 y\_hat = np.argmax(np.dot(w, x))  
 if y == y\_hat:  
 success += 1  
 i += 1  
 success /= per\_valid  
 if success > max\_success:  
 max\_success = success  
 max\_n = n  
 n += 0.1  
 if max\_success > max2\_success:  
 max2\_success = max\_success  
 max\_n2 = max\_n  
 max\_epoch = epoch  
 print(f" the percentage success is : {(max\_success / len(validation\_x)) \* 100}% with n = {max\_n} for epoch = {epoch} ")  
 print(f" the best percentage success is : {(max2\_success/len(validation\_x))\*100}% with n = {max\_n2} for epoch = {max\_epoch} ")  
 return max\_epoch,max\_n2

לאחר הרצה ממושכת קיבלתי שעבור n=0.83 וכמות ephochs=102 האלגוריתם נותן את האחוזים הגבוהים ביותר. אך בהסתכלות בנתונים עבור שאר ההדפסות גילתי שהפער כמעט לא ניכר עבור ערכי n שונים ובאחוזים מאוד מאוד קטנים אז ניתן לומר שערך הלמידה לא כל כך שינה.

the best percentage success is : 97.39130434782608% with = 0.8300000005 for epoch = 102

עבור אלגוריתם svm הצטרכתי למצוא את הערכים הכי טובים עבור ערכי הלמדא, קצת הלמידה

וכמות epochs

def parms\_svm(x\_train, y\_train):  
 per\_valid = 10  
 len\_valid = int(len(x\_train) / per\_valid)  
 max\_ep=1  
 max3\_success=0  
 max\_3n=1  
 max\_lamda3=1  
 for epoch in range(1,200):  
 max\_lamda2 = 1  
 max2\_success = 0  
 max\_n = 1  
 n = 1  
 while n > 0.0000001:  
 max\_lamda = 1  
 lamda = 0.1  
 max\_success = 0  
 while lamda >= 0.00000000001:  
 success = 0  
 i = 0  
 while i < per\_valid:  
 begin = i \* len\_valid  
 end = begin + len\_valid - 1  
 tmp\_x = x\_train.copy()  
 tmp\_y = y\_train.copy()  
 validation\_x = tmp\_x[begin:end]  
 validation\_y = tmp\_y[begin:end]  
 for j in range(len\_valid):  
 tmp\_x = np.delete(tmp\_x, begin, axis=0)  
 tmp\_y = np.delete(tmp\_y, begin, axis=0)  
 w = svm\_train(lamda, n, epoch, tmp\_x, tmp\_y)  
 for x, y in zip(validation\_x, validation\_y):  
 y\_hat = np.argmax(np.dot(w, x))  
 if y == y\_hat:  
 success += 1  
 i += 1  
 success /= per\_valid  
 if success > max\_success:  
 max\_success = success  
 max\_lamda = lamda  
 lamda /= 10  
 if max\_success > max2\_success:  
 max2\_success = max\_success  
 max\_lamda2 = max\_lamda  
 max\_n = n  
 n /= 10  
 if max2\_success > max3\_success:  
 max3\_success = max2\_success  
 max\_3n = max\_n  
 max\_lamda3 = max\_lamda2  
 max\_ep = epoch  
 print(f" the percentage success is : {(max2\_success / len(validation\_x)) \* 100}% with n = {max\_n} for lamda = {max\_lamda2} for epoch = {epoch} ")  
  
 print(f" the best best percentage success is : {(max3\_success / len(validation\_x)) \* 100}% with n = {max\_3n} for lamda = {max\_lamda3}for epoch = {max\_ep} ")  
 return max\_lamda3, max\_3n,max\_ep

the best best percentage success is : 96.52173% with n = 0.001 for lamda = 0.1 for epoch = 89

ועבור אלגוריתם pa היה צריך רק למצוא את כמות ה epochs

def epoch\_parm\_pa(x\_train, y\_train):  
 per\_valid = 10  
 len\_valid = int(len(x\_train) / per\_valid)  
 max\_epoch = 1  
 max\_success = 0  
 for epoch in range(1, 200):  
 success = 0  
 i = 0  
 while i < per\_valid:  
 begin = i \* len\_valid  
 end = begin + len\_valid - 1  
 tmp\_x = x\_train.copy()  
 tmp\_y = y\_train.copy()  
 validation\_x = tmp\_x[begin:end]  
 validation\_y = tmp\_y[begin:end]  
 for j in range(len\_valid):  
 tmp\_x = np.delete(tmp\_x, begin, axis=0)  
 tmp\_y = np.delete(tmp\_y, begin, axis=0)  
 w = pa\_train(epoch, tmp\_x, tmp\_y)  
 for x, y in zip(validation\_x, validation\_y):  
 y\_hat = np.argmax(np.dot(w, x))  
 if y == y\_hat:  
 success += 1  
 i += 1  
 success /= per\_valid  
 if success > max\_success:  
 max\_success = success  
 max\_epoch = epoch  
 print(f" the percentage success is : {(success / len(validation\_x)) \* 100}% with for epoch = {epoch} ")  
 print(f" the best percentage success is : {(max\_success / len(validation\_x)) \* 100}% for epoch = {max\_epoch} ")  
 return max\_epoch

the best percentage success is : 96.08695652173914% for epoch = 86

לגבי אלגוריתמי הלמידה השתמשתי בדברים הבאים בשביל להגדיל את אחוז ההצלחה:

Feature selection כאשר בניסוי וטעיה החלטתי להוריד את הפיצ׳ר האחרון

בחירת היפר-פרמטרים שמניבים אחוזים הטובים ביותר על סט הלמידה

הוספת bias

נרמול הנתונים לפי פונקציית z-score