

אלגוריתם Percpetron

באלגוריתם Percptron יש שני פרמטרים epoc, eta אשר יכולים להשפיע על הדיוק של החיזוי.
הפרמטר epoc – מספר ההרצות לדיוק האימון באלגוריתם.
הפרמטר eta קבוע המכונה 'קצב הלמידה' (מספר קטן)
כיצד בדקנו איזה epoc, eta כדאי לקחת ?
הרצנו את הקוד הבא:

```
def try_precptron(full_train_x, full_train_y):
    sizeTrainingSet = 355
    max = 0
    eta_val = 0
    k_val = 0
    kf = KFold(n_splits=5)
    KFold(n_splits=5, random_state=None, shuffle=False)
    for i in range(60,100):
        j = 0.1
        while j < 1:
            comp = 0
            for train_ind, test_ind in kf.split(full_train_x):
                x_train, x_test = full_train_x[train_ind], full_train_x[test_ind]
                y_train, y_test = full_train_y[train_ind], full_train_y[test_ind]

                res = perceptron(x_train, y_train, x_test, i,j)
                comp += np.sum(res == y_test)

            if (comp / sizeTrainingSet > max):
                max = comp / sizeTrainingSet
                eta_val = j
                k_val = i
            j += 0.1
        print("epoc: ", i, "accuracy:", comp / sizeTrainingSet, "eta: ", j)
    print("max accuracy: ", max, "eta: ", eta_val, "epoc: ", k_val)
```

בלולאה הראשית בדקנו epoc בין 60 ל100 ובלולאה הפנימית בדקנו eta בין 0.1 ל1. חישבנו את הדיוק עבור חלוקה של הtrain לחמש חלקים שונים והרצה כל פעם של חלק שונה כ test ואת השאר לחלק של הלמידה

$$\text{accuracy} = \frac{\sum_{i=0}^5 \text{comp}}{\text{size training test}}$$

כאשר comp הוא מספר הtags הנכונים שהאלגוריתם חזה. זה size training test הוא 355.

לבסוף אנחנו שומרים את הeta, epoc שעבורם אנחנו מקבלים את הaccuracy הגבוה ביותר.

```
epoc: 99 accuracy: 0.6873239436619718 eta: 0.4
epoc: 99 accuracy: 0.6394366197183099 eta: 0.5
epoc: 99 accuracy: 0.6929577464788732 eta: 0.6
epoc: 99 accuracy: 0.6112676056338028 eta: 0.7
epoc: 99 accuracy: 0.7690140845070422 eta: 0.8
epoc: 99 accuracy: 0.7098591549295775 eta: 0.9
epoc: 99 accuracy: 0.6535211267605634 eta: 1.0
max accuracy: 0.7690140845070422 eta: 0.5 epoc: 76
```

קיבלנו שעבור epoc=76 וeta=0.5 נקבל דיוק גבוה ביותר העומד על 0.7690141 ולכן נריץ את האלגוריתם שלנו עם ערכים אלו.

אלגוריתם passive agressive

באלגוריתם passive agressive יש פרמטר אחד שיכול להשפיע על הדיוק של החיזוי והוא epoc.
הפרמטר epoc – מספר ההרצות לדיוק האימון באלגוריתם.
כיצד בדקנו איזה epoc לקחת ?
הרצנו את הקוד הבא:

```
def pa(train_data, train_lbl, test_data, i):  
    train_data, test_data, w = initialize(train_data, test_data, 0.0)  
    predict = []  
  
    for e in range(i):  
        c = list(zip(train_data, train_lbl))  
        np.random.shuffle(c)  
        for x, y in c:  
            y = int(y)  
            temp = replace(w, y)  
  
            y_hat = np.argmax(np.dot(temp, x))  
            max_val = max(0.0, 1.0 - np.dot(w[y], x) + np.dot(w[y_hat], x))  
            tau = max_val / (2 * ((np.linalg.norm(x)) ** 2))  
  
            if y != y_hat:  
                w[y, :] = w[y, :] + tau * x  
                w[y_hat, :] = w[y_hat, :] - tau * x  
  
        for test in test_data:  
            mx = np.argmax(np.dot(w, test))  
            predict.append(mx)  
  
    return predict
```

בלולאה הראשית בדקנו epoc בין 60 ל 100 וחישבנו את הדיוק עבור חלוקה של הtrain לחמש חלקים שונים והרצה כל פעם של חלק שונה כ test ואת השאר לחלק של הלמידה

$$\text{accuracy} = \frac{\sum_{i=0}^5 \text{comp}}{\text{size training test}}$$

כאשר comp הוא מספר tags הנכונים שהאלגוריתם חזה. זה size training test הוא 355.

לבסוף אנחנו שומרים את ה epoc שעבורו אנחנו מקבלים את ה accuracy הגבוה ביותר.

```
epoc:    93 accuracy: 0.6563380281690141  
epoc:    94 accuracy: 0.7211267605633803  
epoc:    95 accuracy: 0.5492957746478874  
epoc:    96 accuracy: 0.7436619718309859  
epoc:    97 accuracy: 0.7633802816901408  
epoc:    98 accuracy: 0.647887323943662  
epoc:    99 accuracy: 0.6591549295774648  
max accuracy: 0.7690140845070422 epoc:    65
```

קיבלנו שעבור epoc=65 ה accuracy הוא הגבוה ביותר ועומד על 0.76901408 ולכן נריץ את האלגוריתם שלנו עם epoc זה.

אלגוריתם KNN

באלגוריתם Knn יש פרמטר אחד- k אשר יכול להשפיע על הדיוק של החיזוי.
הפרמטר K – מספר השכנים אותם ניקח (לפי tag של רוב השכנים נבחר את tag שלנו)
הפרמטר K צריך להיות קטן ממספר הרשומות בtrain ומומלץ שיהיה אי זוגי.
כיצד בדקנו איזה K כדאי לקחת ?
הרצנו את הקוד הבא:

```
def try_knn(full_train_x, full_train_y):  
    sizeTrainingSet = 355  
    max = 0  
    kf = KFold(n_splits=5)  
    KFold(n_splits=5, random_state=None, shuffle=False)  
    for k_num in range(1, 100):  
  
        comp = 0  
        for train_ind, test_ind in kf.split(full_train_x):  
            x_train, x_test = full_train_x[train_ind], full_train_x[test_ind]  
            y_train, y_test = full_train_y[train_ind], full_train_y[test_ind]  
  
            res = knn(x_train, y_train, x_test, k_num)  
            comp += np.sum(res == y_test)  
  
        if (comp/sizeTrainingSet > max):  
            max = comp/sizeTrainingSet  
            epok_val = k_num  
        print("k: ", k_num, "accuracy:", comp / sizeTrainingSet)  
    print("accuracy:", max, "k_number:", epok_val)
```

בלולאה הראשית בדקנו K בין 1 ל 100 וחישבנו את הדיוק עבור חלוקה של הtrain לחמש חלקים שונים והרצה כל פעם של חלק שונה כ test ואת השאר לחלק של הלמידה

$$accuracy = \frac{\sum_{i=0}^5 comp}{size\ training\ test}$$

כאשר comp הוא מספר הtags הנכונים שהאלגוריתם חזה. זה size training test הוא 355.

לבסוף אנחנו שומרים את ה K שעבורו אנחנו מקבלים את ה accuracy הגבוה ביותר.

```
k: 91 accuracy: 0.7267605633802817  
k: 92 accuracy: 0.7267605633802817  
k: 93 accuracy: 0.7323943661971831  
k: 94 accuracy: 0.7267605633802817  
k: 95 accuracy: 0.7267605633802817  
k: 96 accuracy: 0.7211267605633803  
k: 97 accuracy: 0.7295774647887324  
k: 98 accuracy: 0.7211267605633803  
k: 99 accuracy: 0.7267605633802817  
accuracy: 0.7971830985915493 k_number: 8
```

קיבלנו שעבור k=8 ה accuracy הוא הגבוה ביותר ועומד על 0.72676 ולכן נריץ את האלגוריתם שלנו עם k זה.