

گزارش الگوریتم ژنتیک

علیرضا ابراهیمی

۸۱۰۳۰۱۰۱۷

در این پروژه هدف استفاده از الگوریتم ژنتیک به منظور انتخاب ۱۰٪ از باندهای یک عکس hyperspectral برای طبقه‌بندی عکس است. بدین منظور از بین ۲۰۰ باند عکس به صورت رندوم ۲۰ باند انتخاب می‌کنیم و سپس این عکس را به دو قسمت train و test تقسیم می‌کنیم. سپس ابتدا با نمونه آموزشی با استفاده از پکیج libsvm یک مدل آموزش می‌دهیم و با قسمت تست عکس برای آن دقت به دست می‌آوریم.

نتایج دقت برای باندهای متفاوت به صورت زیر است:

دقت	شماره باند
51.8554%	197 ,193 ,190 ,177 ,161 ,158 ,147 ,143 ,129 ,118 ,93 ,87 ,81 ,60 ,58 ,46 ,33 ,12 ,11 ,9
52.0457%	190 ,168 ,157 ,156 ,156 ,144 ,140 ,127 ,125 ,120 ,115 ,105 ,92 ,75 ,71 ,67 ,61 ,60 ,60 ,38
52.7593%	194 ,185 ,179 ,175 ,168 ,160 ,144 ,140 ,134 ,134 ,133 ,129 ,115 ,89 ,64 ,23 ,21 ,15 ,6 ,4

به طور مثال مشاهده می‌شود که بین سه ست انتخاب باند، ست آخر بهترین نتیجه را داده است. بنابراین به عنوان باندهای مناسب انتخاب می‌شوند.

```
from libsvm.svml import *
import scipy.io
import numpy as np
import random
DATA = scipy.io.loadmat('/home/alireza/Desktop/seg/Indian_pines_corrected.mat')
label = scipy.io.loadmat('/home/alireza/Desktop/seg/Indian_pines_gt.mat')
DATA = DATA['indian_pines_corrected']
lbl = label['indian_pines_gt']
lbl = lbl.flatten()

def flattening(data):
    data = data
    image = np.zeros((data.shape[0]*data.shape[1] , 1))
```

```

    for i in range(data.shape[2]):
        band = data[:, :, i]
        band = band.flatten()
        band = band.reshape((-1, 1))
        image = np.append(image, band, axis=1)
    image = image[:, 1:]
    return image

ind = []
for i in range(20):
    rand = random.randint(0, 200)
    ind.append(rand)

data = DATA[:, :, ind]
data = flattening(data)
rate = 0.1
k = round(rate * data.shape[0])
test_ind = random.sample(range(0, data.shape[0]), k)

test = data[test_ind]
test_gt = lbl[test_ind]
train = np.delete(data, test_ind, axis = 0)
train_gt = np.delete(lbl, test_ind, axis = 0)
print(f'data:"total data : {data.shape}/ test : {test.shape} / train : {train.shape}')

model = svm_train(train_gt, train)
p_labels, p_acc, p_vals = svm_predict(test_gt, test, model)
print(sorted(ind))

```