

به نام خدا

## پروژه : SVM

محمد حسین میرزائی 99522158

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import svm
from sklearn.datasets import make_blobs
```

کتاب خانه های بالا را اضافه می کنیم .(برای این پروژه از colab استفاده شده است)

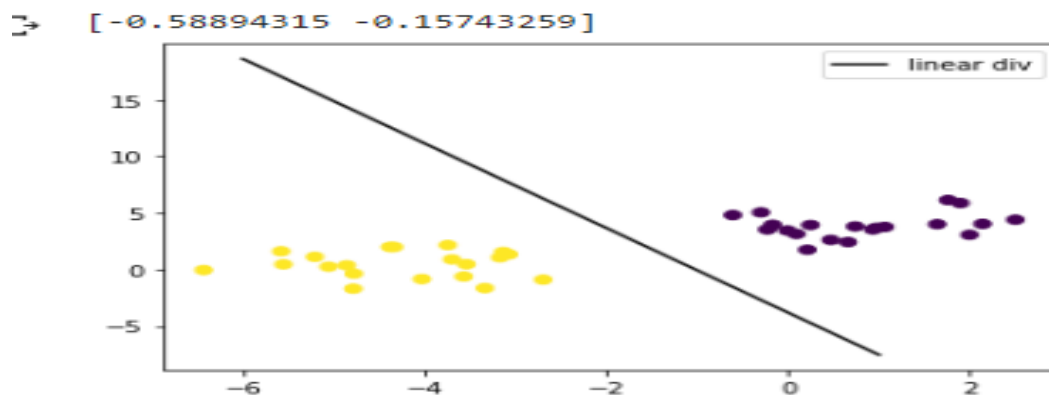
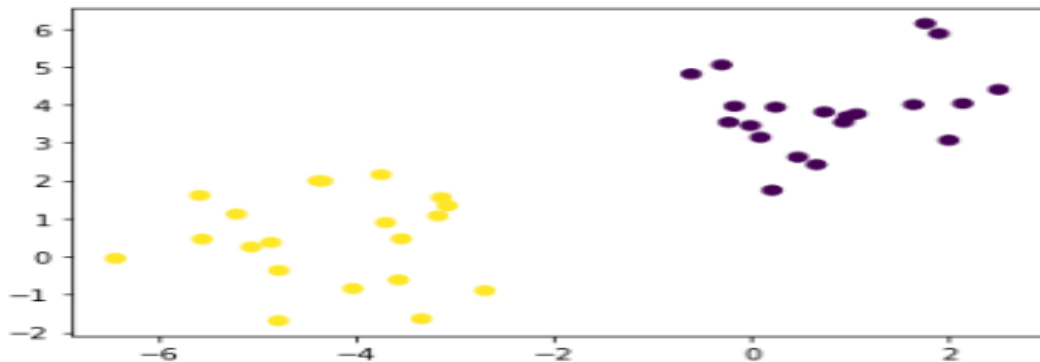
با استفاده از دستور make\_blob ابتدا سوال را برای شرایط راحت تر پیاده می کنیم ، این دستور برای ساخت نقطه به صورت 2 بعدی می باشد .

حال با توجه به دستورات SVM را با حالت مورد نظر پیاده می کنیم ، معمولا 'rbf' و یا 'linear' و گاهی نیز حالت 'poly' نیز می گیریم .

با صدا زدن تابع fit ، SVM داده های ساخته شده را می شناسد و در ادامه با صدا زدن scatter روی اطلاع صدا می زنیم تا بتوانیم اطلاعات را به شکل تبدیل کنیم و چون در 2 بعد می باشد می توان آن را روی شکل به راحتی نشان داد .

در مرحله بعدی اگر داده ها را صرفا خطی بررسی کنیم ، می توانیم از coef\_ یا همان coefficient

را صدا می زنیم و با استفاده از آن خط `margin` میان این داده های اولیه را پیاده کرده و نشان می دهیم.



حال برای پیاده سازی بهتر می توان با استفاده از `gca` و `get_xlim, get_ylim` پی می بریم که بازه اطلاعات ما در کدام قسمت قرار دارد .

در ادامه با استفاده از `linspace` خطوطی را بدست می آوریم که به 2 دسته نزدیک هستند و خطی میان آن ها را برای `margin` در نظر می گیریم ؛ بخش تقسیم برای - استفاده و برای نزدیک به داده ها با -- نشان داده شده است .

دقت شود که برای این بخش نوع `SVM` را می توان به صورت دقیق تعیین و جواب را برای بازه ی مورد نظر بررسی کرد (با توجه به `get_xlim, get_ylim` )

در مرحله بعدی داده ها را برای عکس باز سازی می کنیم .

ابتدا با استفاده از `os.listdir` تمام عکس ها را می خوانیم و با استفاده از `imread` داده ها را به صورت ماتریس در می آوریم و سائز آن ها را بررسی میکنیم و داده ها را `flatten` می کنیم و همه اعضا را به ماتریس تبدیل می کنیم و خروجی را با توجه به نوع داده ها به چند بخش تقسیم می کنیم که اگر ورودی از نوع `x` و یا `y` خروجی به این ورودی ها حساس باشد .

حال داده ها را به 2 بهش `train` و `test` تبدیل می کنیم و برای آن ها `SVM` را صدا می کنیم و در نهایت داده ها را با استفاده از `classification report` پیش بینی نهایی می کنیم .