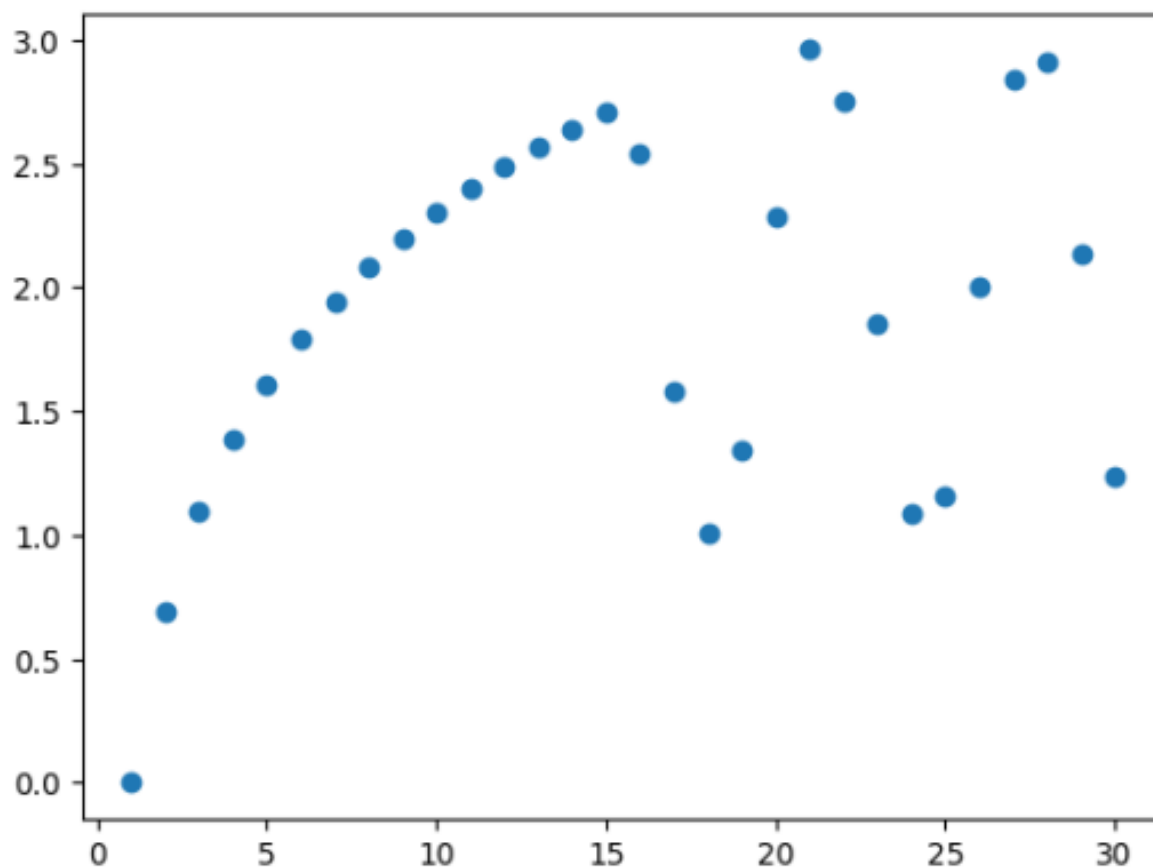


## پارت اول :

خطی:

ابتدا تعدادی نقطه، از 1 تا  $n$ ،  $n/2$  نقطه اول را به تابع اول (نمونه تابعی که در شکل زیر میبینیم برای  $n/2$  نقطه اول لگاریتم) و  $n/2$  نقطه بعد را به تابعی دیگر (که در شکل زیر کسینوسی است) میدهم و  $n/2$  تای اول کلاس 1 و به  $n/2$  تای آخر کلاس 0 دادم. من در ورودی تعداد نقاط یعنی  $n = 30$  در نظر گرفتم.

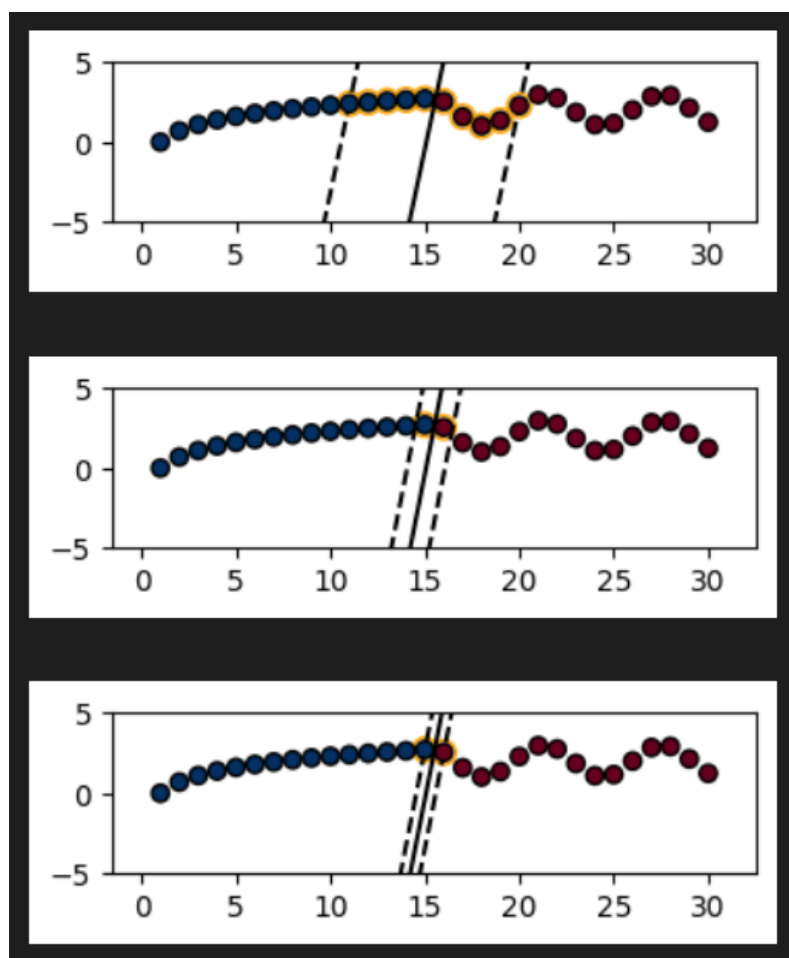


سپس برای C های مختلف مدل SVM با هسته Linear ساخته و آموزشش داده می شود.

سپس weight vector که نشان دهنده ضرایب ویژگی های مورد استفاده برای تعیین مرز تصمیم گیری است و slope\_parameter که شیب تمایل مرز تصمیم را در فضای ویژگی تعیین میکند و مرز تصمیم را از طریق فرمول :

$$y\_values = slope\_parameter * x\_values - (clf\_model.intercept\_ [0]) / weight\_vector$$

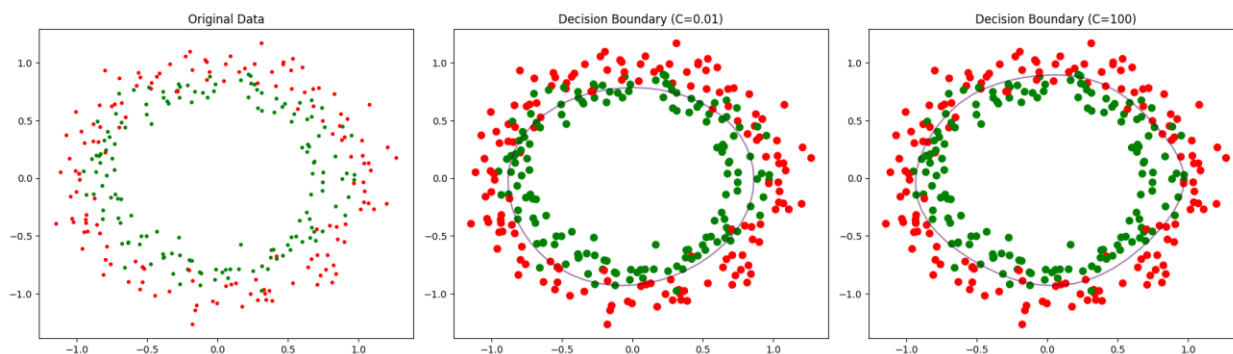
و مارجین های بالا و پایین را محاسبه کرده و حاصل را نمایش میدهم:



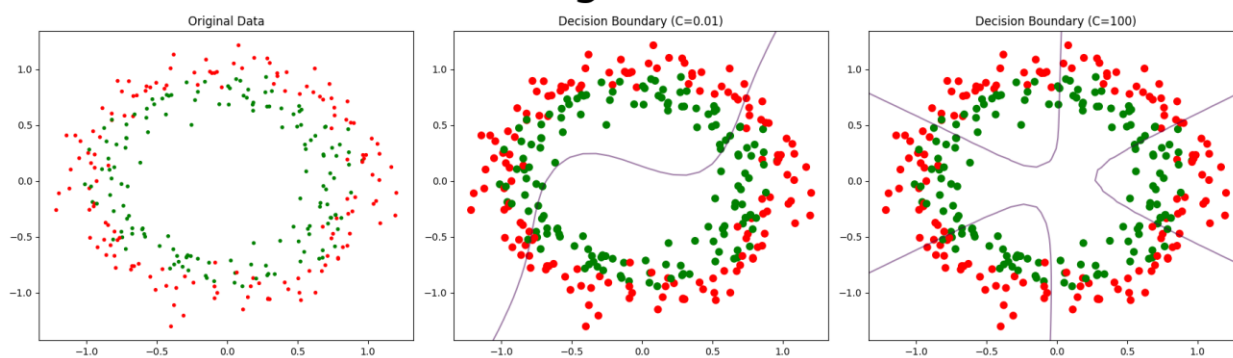
غیرخطی :

Noise = 0.1

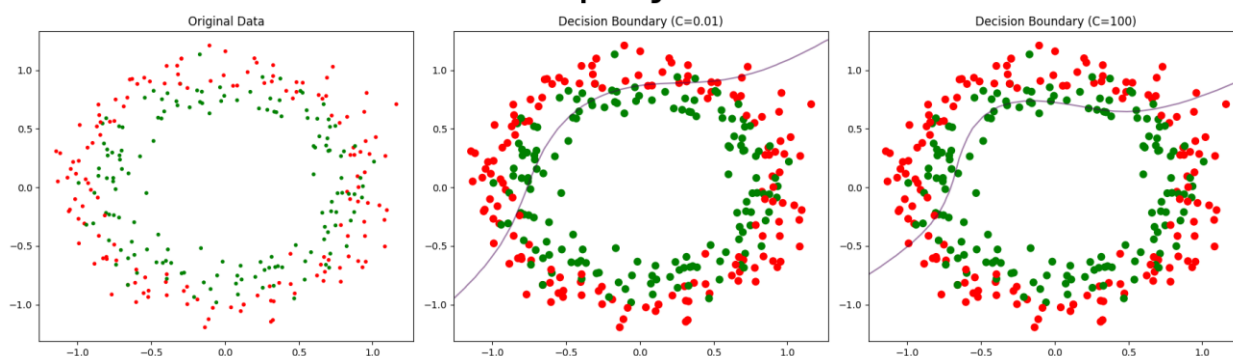
rbf



sigmoid

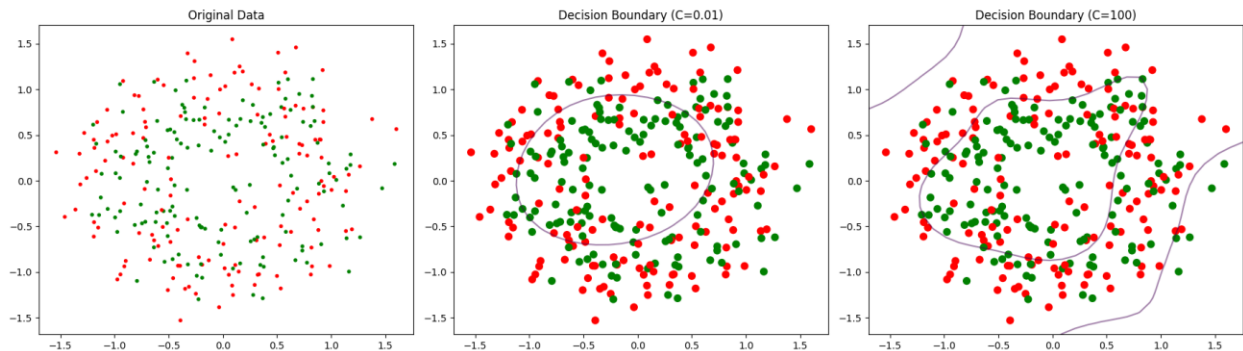


poly

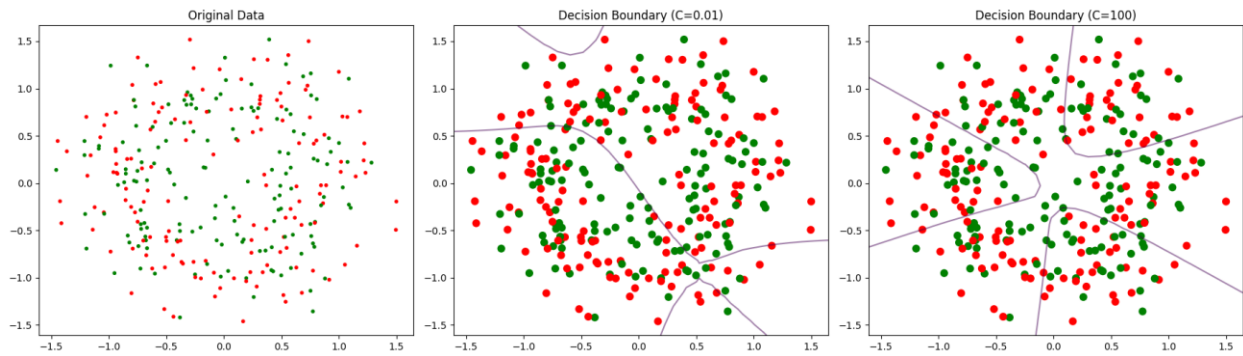


Noise = 0.3

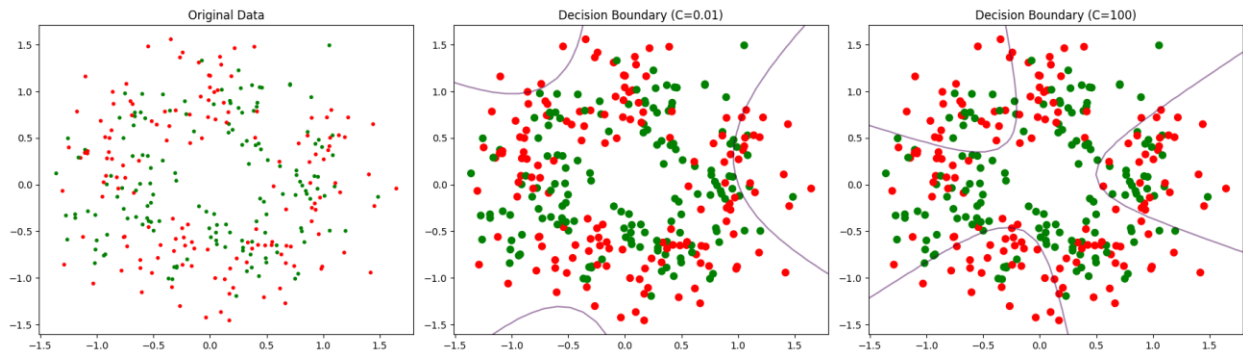
rbf



sigmoid

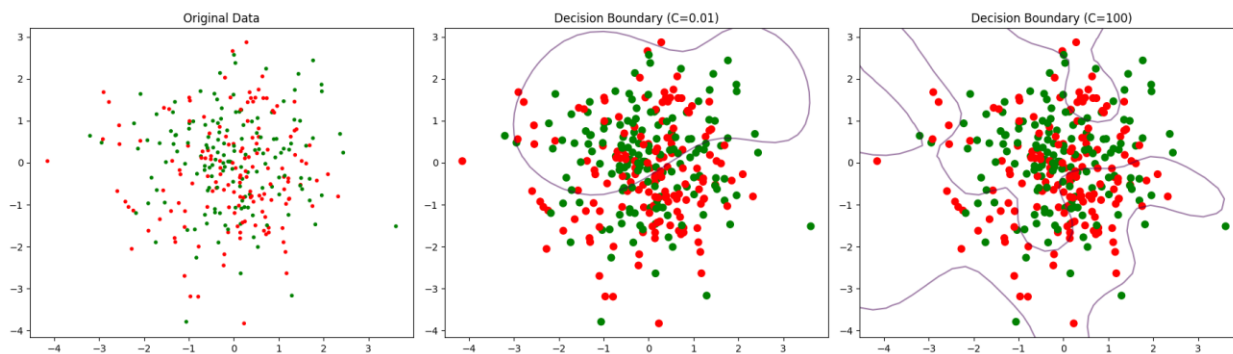


poly

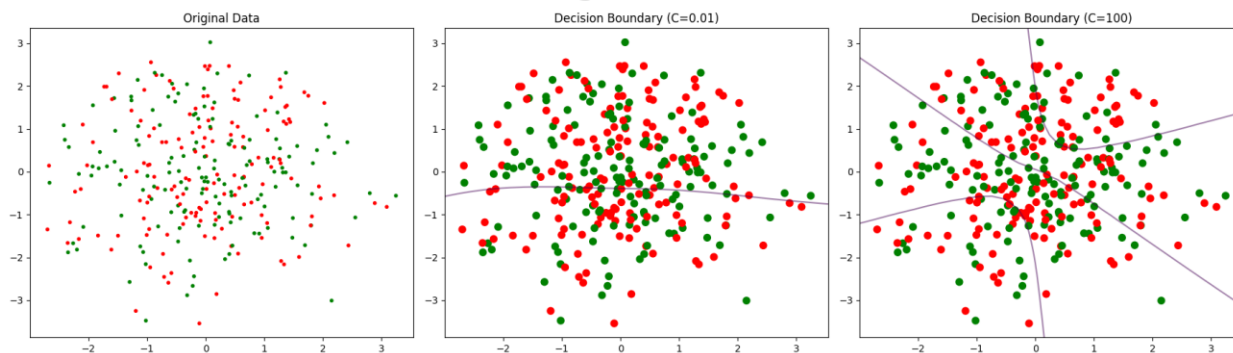


Noise = 1

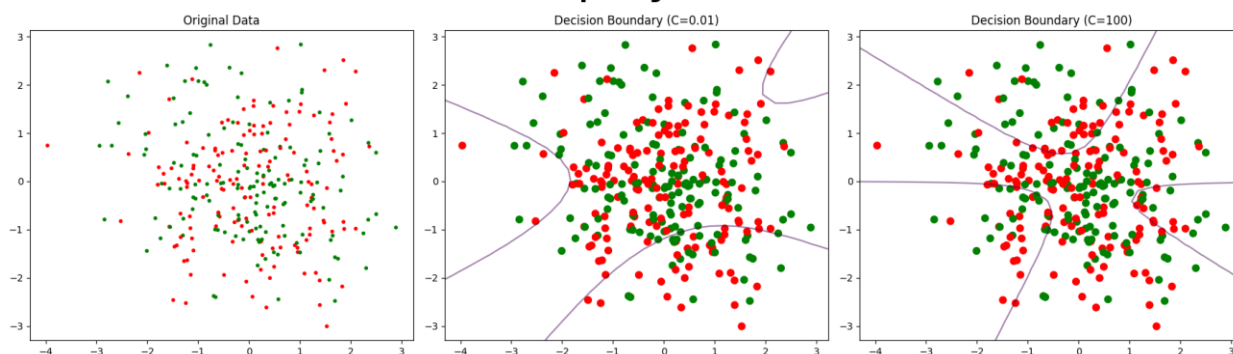
rbf



sigmoid



poly



دیتا با پیچیدگی کم:

کرل linear احتمالاً خوب عمل کند اگر دیتا با یک خط صاف قابل جدا کردن باشد.

RBF و Polynomial ممکن است لازم نباشد و موجب بیش برازش شود.

دیتا با پیچیدگی متوسط:

RBF و Polynomial ممکن است شروع به عملکرد بهتری از هسته Linear بکنند.  
هسته RBF انعطاف پذیرتر است و می تواند اشکال مختلفی را در decision boundary هندل کند.

دیتا با پیچیدگی زیاد:

RBF و Polynomial ممکن است روی داده های بسیار پیچیده و غیرخطی عملکرد خوبی داشته باشند.  
با این حال خطر بیش برآزش وجود دارد . تنظیم دقیق hyperparameter ها بسیار مهم است.  
هسته sigmoid همچنین ممکن است برای انواع خاصی از داده های پیچیده مفید باشد.

## پارت دوم:

ابتدا تصویر ها را لود کرده و نرمالایز کردیم. برای چهار مدل Sigmoid ، RBF، Poly،Linear داده ها را آموزش داده و تست را انجام داده و accuracy برای هر هسته چاپ میشود.

در گام بعد برای هر هسته hyperparameter tuning به وسیله grid search انجام می شود (از cross-validation با  $k = 5$  استفاده میشود) تا بهترین parameter روی داده های آموزشی پیدا شود. سپس دقت را روی داده های تست بدست آورده و بهترین پارامتر و دقت برای هر هسته چاپ میشود.

```
Accuracy with Linear kernel: 0.9282511210762332
*****
Accuracy with Poly kernel: 0.9451918285999004
*****
Accuracy with RBF kernel: 0.9471848530144494
*****
Accuracy with Sigmoid kernel: 0.8784255107125062
*****
Best parameters found for Linear kernel: {'svc__C': 1, 'svc__kernel': 'linear'}
Accuracy with best Linear model: 0.9282511210762332
*****
Best parameters found for Poly kernel: {'svc__C': 100, 'svc__kernel': 'poly'}
Accuracy with best Poly model: 0.9506726457399103
*****
Best parameters found for RBF kernel: {'svc__C': 10, 'svc__kernel': 'rbf'}
Accuracy with best RBF model: 0.9516691579471849
*****
Best parameters found for Sigmoid kernel: {'svc__C': 1, 'svc__kernel': 'sigmoid'}
Accuracy with best Sigmoid model: 0.8784255107125062
*****
```

- از لحاظ دقت هر چهار هسته با بهترین پارامتر دقت بالاتری از درصد دقت نتیجه من در پروژه شبکه عصبی کسب کردند.
- از لحاظ هایپرپارامتر ها هم ANN لازم دارد تا تعداد بیشنری از آنها را استفاده کند که میتواند کمی چالشی باشد.
- SVM برای binary و multiclass classification tasks عملکرد خوبی دارند و بدلیل عملکرد تعمیم خوب خود شناخته شده اند.

## پارت سوم:

در این پارت من از یک دیتا ست حاوی تصاویر دایره، مربع و مثلث استفاده کردم.  
ابتدا عکس ها را لود میکنیم و در ادامه همان کارهایی که در پارت دو توضیح داده شد  
انجام میدهیم.

نتایج :

```
Accuracy with Linear kernel: 0.9794871794871794
```

```
*****
```

```
Accuracy with Poly kernel: 0.6461538461538462
```

```
*****
```

```
Accuracy with RBF kernel: 0.958974358974359
```

```
*****
```

```
Accuracy with Sigmoid kernel: 0.9128205128205128
```

```
*****
```

```
Best parameters found for Linear kernel: {'svc__C': 1, 'svc__kernel': 'linear'}  
Accuracy with best Linear model: 0.9794871794871794
```

```
*****
```

```
Best parameters found for Poly kernel: {'svc__C': 100, 'svc__kernel': 'poly'}  
Accuracy with best Poly model: 0.958974358974359
```

```
*****
```

```
Best parameters found for RBF kernel: {'svc__C': 1000, 'svc__kernel': 'rbf'}  
Accuracy with best RBF model: 0.9948717948717949
```

```
*****
```

```
Best parameters found for Sigmoid kernel: {'svc__C': 1, 'svc__kernel': 'sigmoid'}  
Accuracy with best Sigmoid model: 0.9128205128205128
```

```
*****
```