به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

ماشین لرنینگ

تمرین ۴

نام و نام خانوادگی	محمد مشرقی
شماره دانشجویی	810199492
تاریخ ارسال گزارش	

فهرست

۱- ۵

۱ ..... ۵

۲ ..... ۶

۳ ..... ۷

۲ ..... ۸

۱ ..... ۸

۲ ..... ۸

۳ ..... ۸

2 ..... ۹

3 ..... ۹

4 ..... ۹

5 ..... ۱۰

3 ..... ۱۱

4 ..... ۱۲

1 ..... ۱۲

۲ ..... ۱۲

Linear ۱۳

Poly ۱۳

RBF ۱۳

RBF datatest ۱۴

Poly dataset ۱۵

۴ ..... ۱۷

regularization hyperparameter: ۱۷

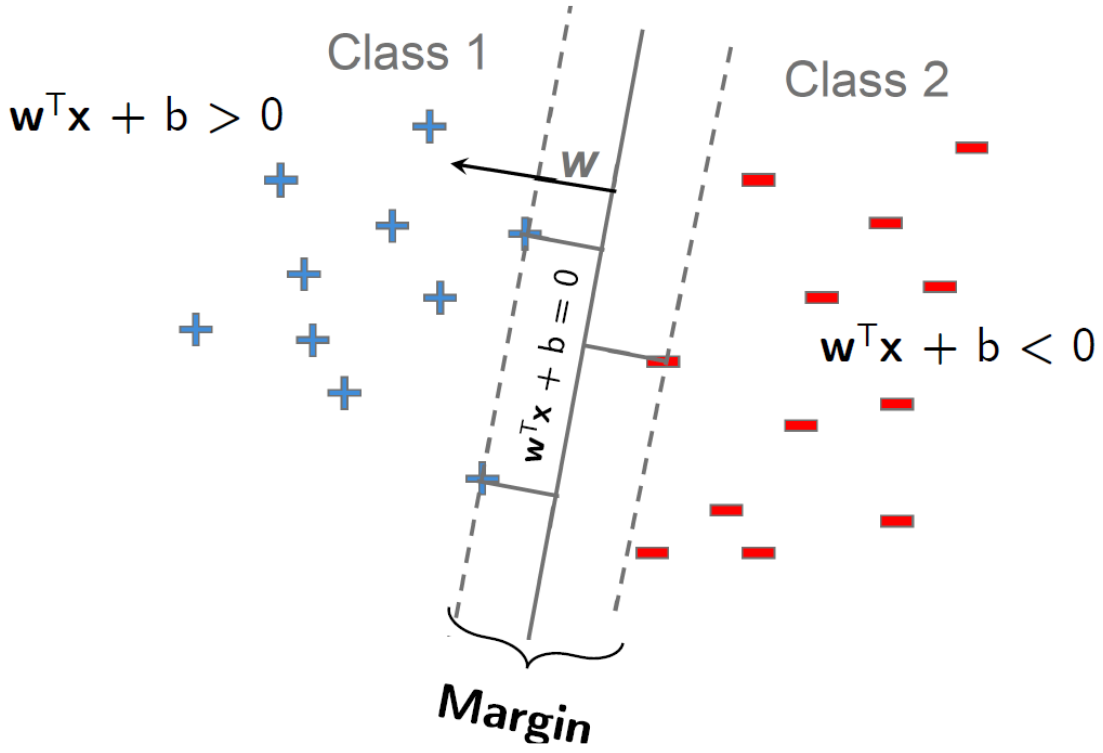
gamma Hyperparameter: ۱۸
تست مقادیر هایپرپارمتر روی RBF ۲۰
تست مقادیر هایپرپارمتر روی poly دیفالت درجه ۳ ..... ۲۶
تست مقادیر هایپرپارمتر روی linear ۳۲
5 ..... ۳۵
نتیجه گیری ۳۸
۶ ..... ۳۹
One vs all ۳۹
One vs one ۴۲
نتیجه کلی ۴۴
۵ ..... ۴۵
۱ ..... ۴۵
۲ ..... ۴۶
۳ ..... ۴۷
6 ..... ۴۸
7 ..... ۵۰
الف ۵۰
..... ۵۱
ب ۵۱
ب ۵۲

۱-

۱

با توجه به فرمول $(\mathbf{w} \cdot \mathbf{x}) + b = 0, \quad \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}$

می فهمیم که جهت قرار گیری خطوطیا صفحه طبق بردار w انتخاب می شود و بردار hyperplane عمود بر صفحه w است و مرز تصمیم گیریست و b هم افستی است که به آن می دهیم تا مکان صفحه مشخص شود.



۲

۲. در صورت وجود نویز در داده‌ی آموزش، آیا روش SVM توانای تفکیک آن‌ها را دارد؟ (بررسی کنید که چطور تفکیک انجام می شود)

در سوال وقتی نویز نداشته باشیم از hard SVM استفاده می کنیم که محدوده مارجین آن کم است اما وقتی نویز بیفتد دیگر نمی تواند به خوبی جدا کند برا هم سراغ softsvm با اینکار مقداری خطا را می پذیریم و در عوض محدوده حاشیه رابیشتر می کنیم با اینکار بهتر می توانیم داده ها را جدا سازی کنیم و و به خوبی جلو نویز گرفته میشود و در فرمول هزینه یه ترم اضافه می شود

$$\min \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{n}\zeta_i$$

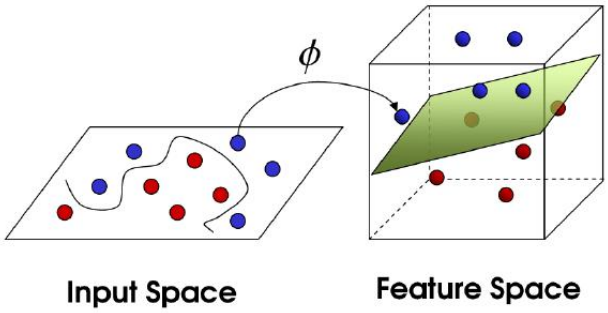
$$\textbf{s.t.} \quad y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \zeta_i \quad \forall i = 1, ..., n,\ \zeta_i \geq 0$$

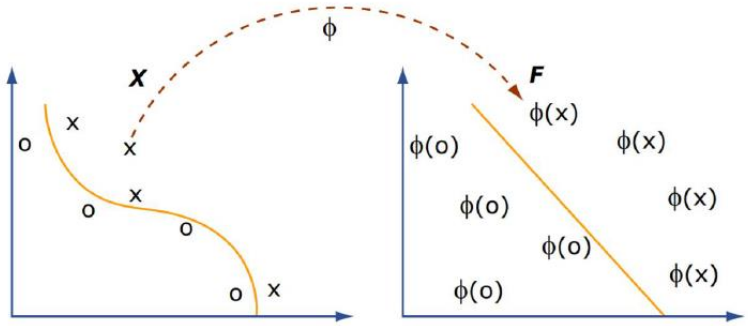
۳

۳. مفهوم کلی کرنل را بیان کنید. فرض کنید از دو کرنل چند جمله‌ای درجه دو زیر برای تعریف طبقه‌بندی SVM استفاده شده است. کدام یک از طبقه‌بندها حاشیه بزرگ‌تری را نتیجه می‌دهد؟ چرا؟

$$\varphi_1 = [x.x^2]^T$$
$$\varphi_2 = [2x.2x^2]^T$$

کرنل یه رابطه ریاضی است که با استفاده از آن می توان فضای ویژگی داده را بیشتر کنیم و نتیجتا باعث می شود با الگوریتم های خطی یا غیر خطی راحت تر داده ها را جدا کنیم. با استفاده از کرنل درست می توان روش را بهینه کرد و نتایج بهتری بدست آورد





دومی چون فاصله بین دو نقطه را بیشتر می کند. در نتیجه حاشیه آنها بیشتر و بزرگتر می شود.

۲

۱

۲

2-1)

$(x^T - y^T)(x-y)$

$(x-y)^T(x-y)$

$$K(x,y) = \exp\left(-\frac{\|x-y\|^2}{\sigma^2}\right)$$

۳

$$= \exp\left(-\frac{x^Tx - x^Ty - y^Tx + y^Ty}{\sigma^2}\right)$$

$$= \exp\left(-\frac{\|x\|^2 - \overbrace{(x^Ty + y^Tx)}^{2x^Ty} + \|y\|^2}{\sigma^2}\right)$$

جداکنیم

$$= \underbrace{\exp\left(-\frac{\|x\|^2}{\sigma^2}\right)}_{f(x)} \underbrace{\exp\left(-\frac{\|y\|^2}{\sigma^2}\right)}_{f(y)} \underbrace{\exp\left(\frac{2x^Ty}{\sigma^2}\right)}_{K_1(x,y)}$$

حال چون سه ترم بالا مثبت اند پس کرنل معتبر است

2.2) $K(x,y) = bK_1(x,y) + aK_2(x,y)$

$$\boxed{\forall g : \iint g(x)\, K(x,y)\, g(y)\, dx\, dy \geq 0}$$ $\to$ $P^T K P$

قابل

2

$$= b\iint g(x)\, K_1(x,y)\, g(x)\, dx\, dy \quad (1)$$

$$+ a\iint g(x)\, K_2(x,y)\, g(x)\, dx\, dy \quad (2)$$

در عبارت بالا در ترم داریم که این دو همیشه باید مثبت باشند

$a, b > 0$ ✓

$a, b < 0$ ✗

$a > 0, b < 0$ } نامعلوم

$b > 0, a < 0$

3

2.3) $K(x,y) = F(x)F(y)$ $\quad P^T K P \geq 0$

$$P^T K P = \sum_{i=1}^{n}\sum_{j=1}^{n} P_i P_j F(x_i) F(x_j) = \left(\sum_{i=1}^{n} P_i F(x_i)\right)^2 \geq 0$$

چون توان 2 دارد همواره بزرگتر مساوی صفر است

4

2.4) $P^T K P \geq 0$

$K_1(x,y) = \langle F(x), F(y) \rangle$ ✓

$K(x,y) = K_1(g(x), g(y)) = \langle F(g(x)), F(g(y)) \rangle$

2.5) $K(x,y) = K_1(x,y) \times K_2(x,y)$

$$= \sum_{i=1}^{m} a_i(x)\, a_i(y) \times \sum_{j=1}^{n} b_j(x)\, b_j(y)$$

$$= \sum_{j=1}^{m} \sum_{i=1}^{n} \underbrace{a_i(x)\, b_j(x)}_{c_{ij}(x)}\ \underbrace{a_j(y)\, b_j(y)}_{c_{i,j}(y)}$$

$$= \langle C(x)^T, C(y) \rangle$$

1- $P^T K P \geq 0$

در عبارت بالا باید ~~[illegible]~~ شرط اثبات شود

2- متقارن بودن

$K(x,y) = K(y,x)$

3

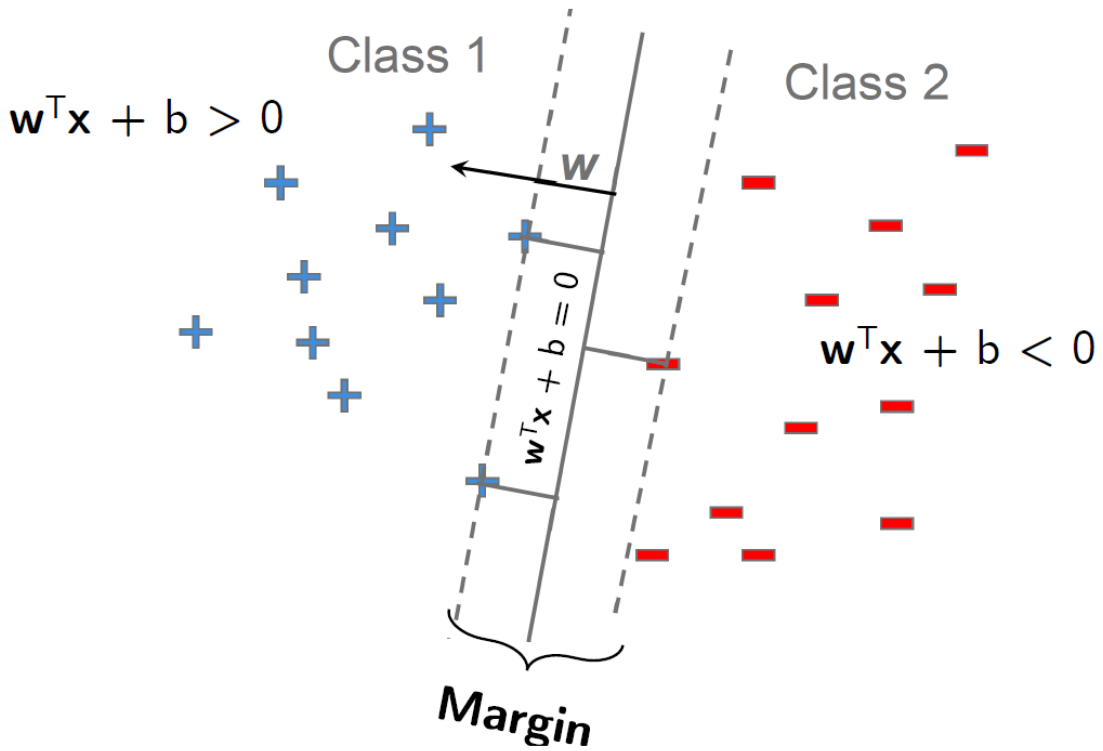
چنانچه تابع Cost function برای Soft margin SVM به صورت زیر تعریف شود، نشان دهید که آیا اینگار می تواند تبدیل به یک مسئله class-separable شود؟

$$\frac{1}{2}||w||^2 + \frac{c}{2}\sum_{i=1}^{N} \varepsilon^2$$

بله در روش svm می توان با انجام اینکار و ضریب دهی به مسئله class separable تبدیل کرد

که با توجه به مقدار نهایی عبارت از طریق فرمول $(\mathbf{w} \cdot \mathbf{x}) + b = 0, \quad \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}$

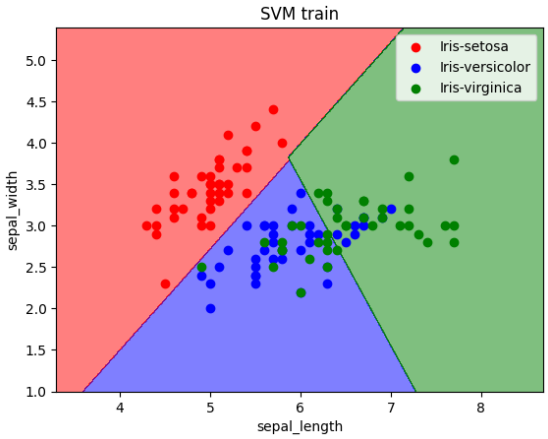
اگر مثبت شود برای یه کلاس و اگر منفی شود یک کلاس دیگر است بدست می آید و می توان جدا کرد

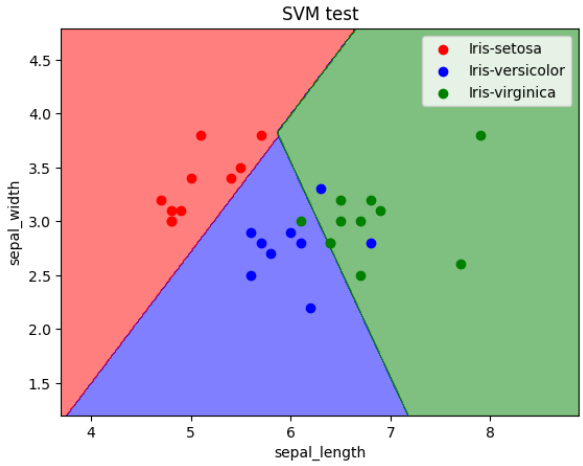


4

1

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score =   0.896
                                                          f1_score =   0.898
```

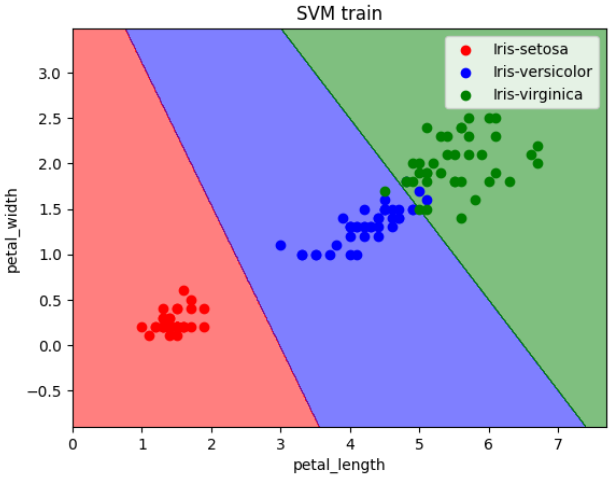


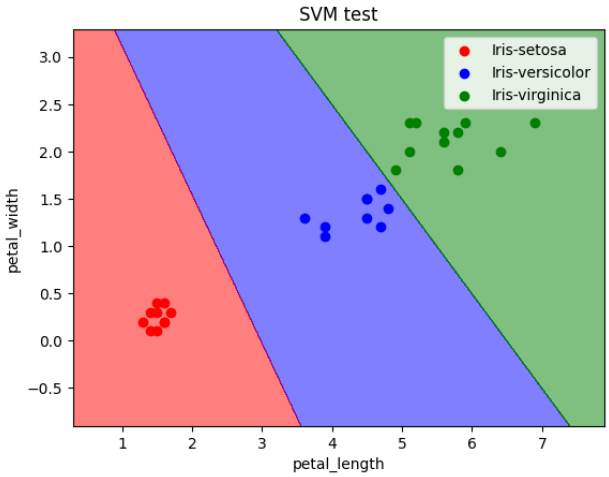


می بینیم که داده ها اصلا به خوبی جدا نشدن و فیچر های بدی داریم

۲

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
                                                =   1.000   f1_score =   1.000
```





با توجه به نتایج می بینیم که این مدل ویژگی خیلی بهتر بوده است

۳

Linear

کرنل خطی که به آن "غیر کرنل " نیز گفته می شود، کرنل خطی ساده ترین کرنل از همه کرنل ها است. وقتی از این کرنل استفاده می‌شود، داده‌ها از لحاظ فنی به ابعاد بالاتر نمایش داده نمی‌شوند، بنابراین فقط حاصل ضرب درونی x و y با یک عبارت ثابت اختیاری c است.

مزیت کرنل خطی این است که فوق العاده ساده است و فقط عبارت ثابت c را به عنوان پارامتر دارد. کرنل خطی معمولاً در مجموعه داده‌هایی با مقادیر زیادی ویژگی استفاده می‌شود، زیرا افزایش ابعاد در این مجموعه داده‌ها لزوماً تفکیک‌پذیری را بهبود نمی‌بخشد.

Poly

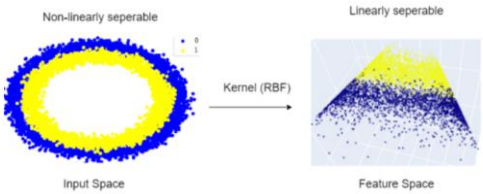
بر خلاف کرنل خطی، کرنل چند جمله ای شامل گرفتن محصول داخلی از فضای ابعاد بالاتر است.

با اینکار محاسبات بیشتر می شود و پارامتر های بیشتری را برای کنترل کردن داریم در عکس پایین ۳ پارامتر برای کنترل کردن داریم تا کاری کنیم برازش خوبی داشته باشیم و اورفیت نشود.

بیشتر وقتی استفاده می شود که می فهمیم داده ها با چند جمله ای بهتر جدا می شوند.

RBF

کرنل تابع پایه شعاعی (RBF) رایج ترین کرنل مورد استفاده در SVM است. به عنوان معادله کرنل تابع پایه شعاعی تعریف می شود که در آن γ یک پارامتر آزاد است که میزان تأثیر دو نقطه را بر یکدیگر مقیاس می کند. بر خلاف کرنل چند جمله ای که به d بعد اضافی نگاه می کند، RBF به تعداد نامتناهی ابعاد گسترش می یابد. این به دلیل گسترش نمایی است.



Linear Kernel: $K(X,Y) = X^T Y$

Polynomial kernel: $K(X,Y) = (\gamma \cdot X^T Y + r)^d, \gamma > 0$

Radial basis function (RBF) Kernel: $K(X,Y) = \exp(\|X-Y\|^2/2\sigma^2)$ which in simple form can be written as $\exp(-\gamma \cdot \|X-Y\|^2), \gamma > 0$

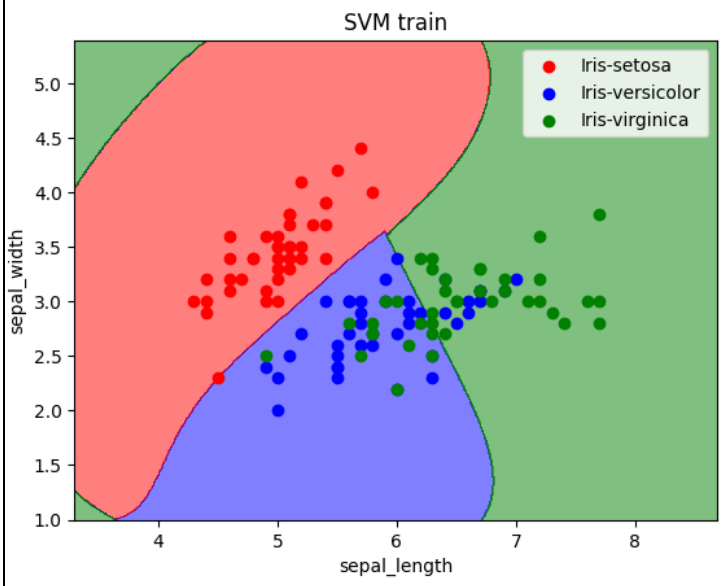
Sigmoid Kernel: $K(X,Y) = \tanh(\gamma \cdot X^T Y + r)$ which is similar to the sigmoid function in logistic regression.

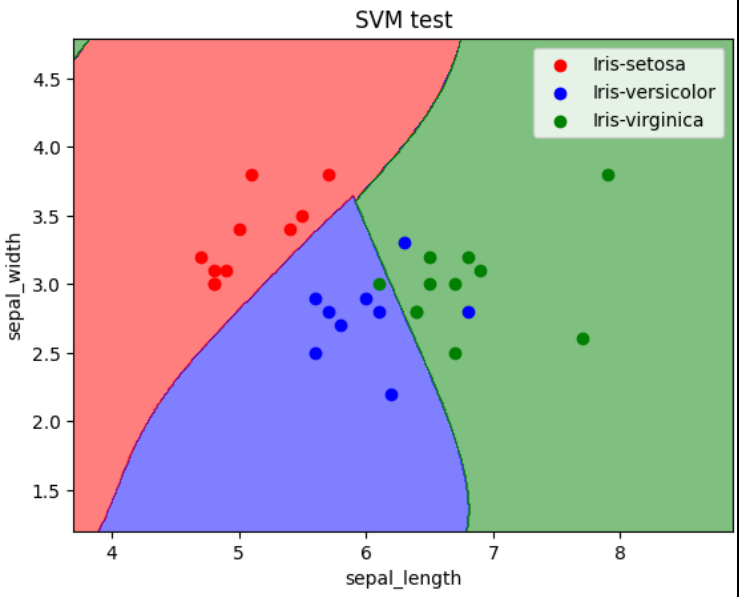
Here r, d, and γ are kernel parameters.

RBF datatest

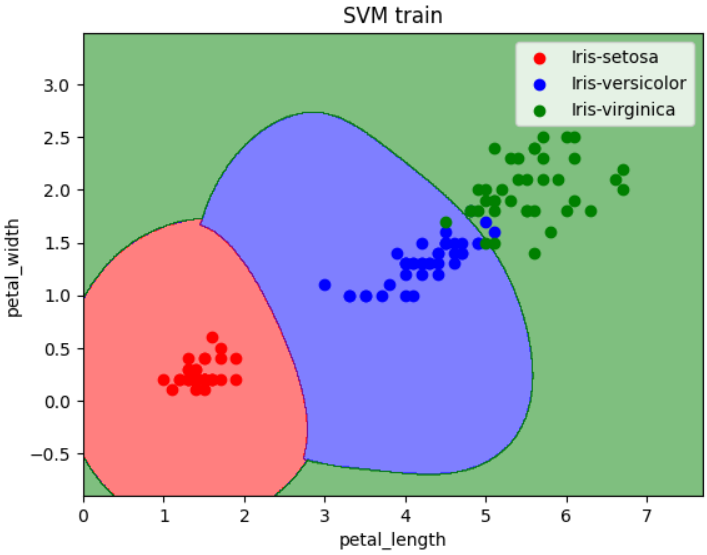
در این مسائل گاما ۰.۸ است.

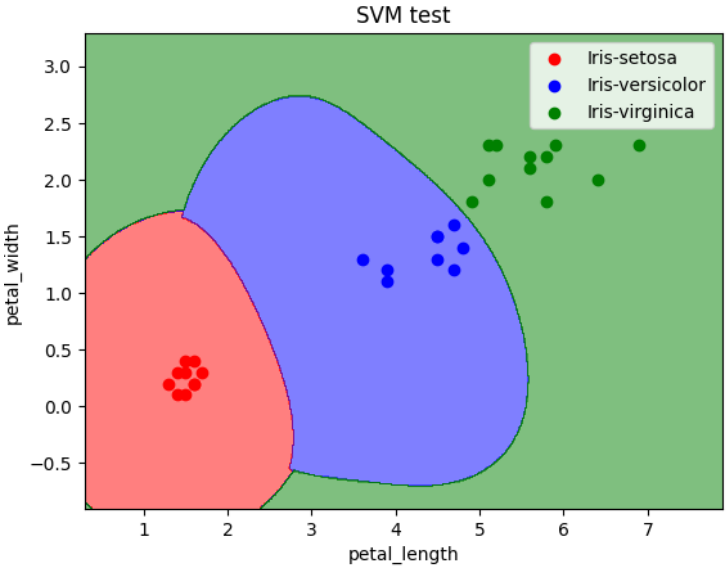
```
[[10  0  0]
[ 0  7  2]
[ 0  1 10]]
accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```






```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

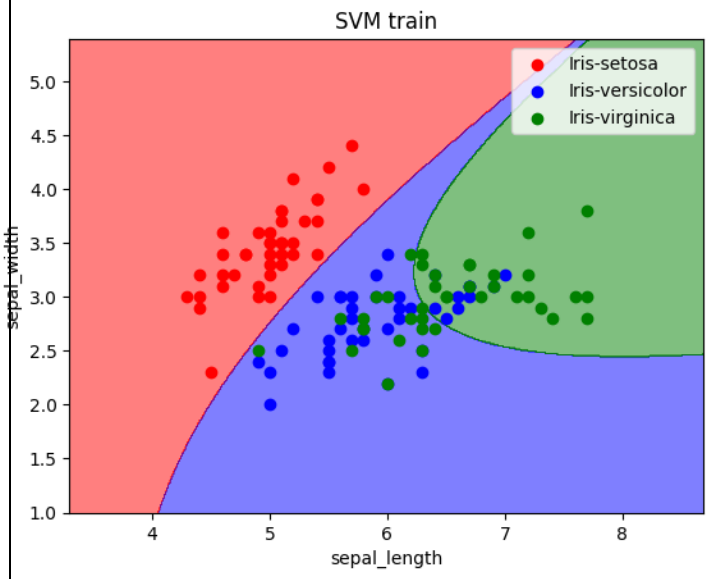


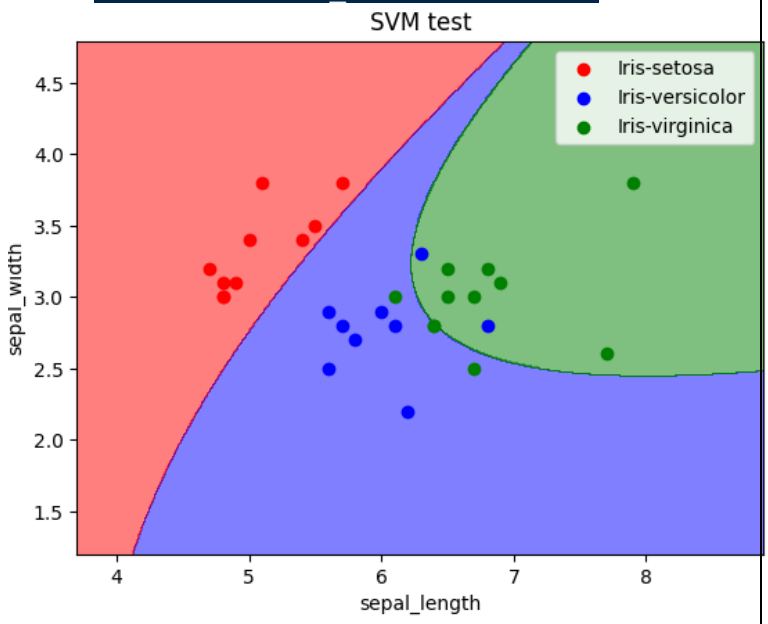


Poly dataset

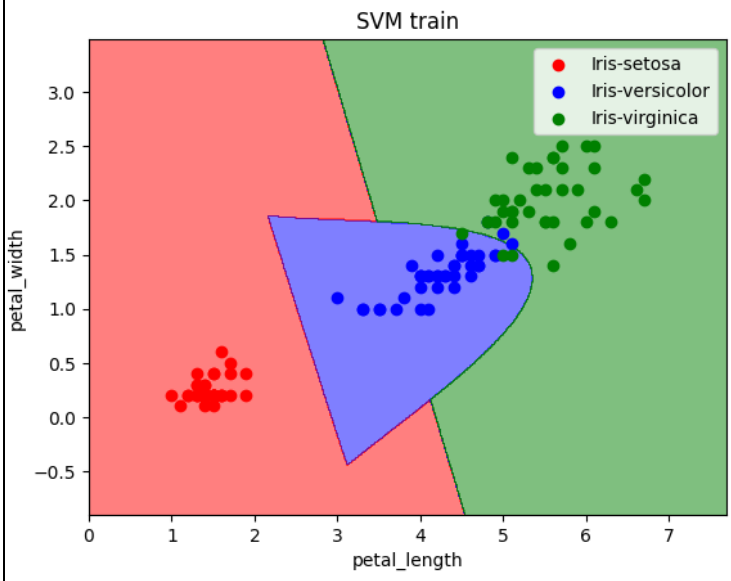
گاما ۰.۸ و درجه سه هست

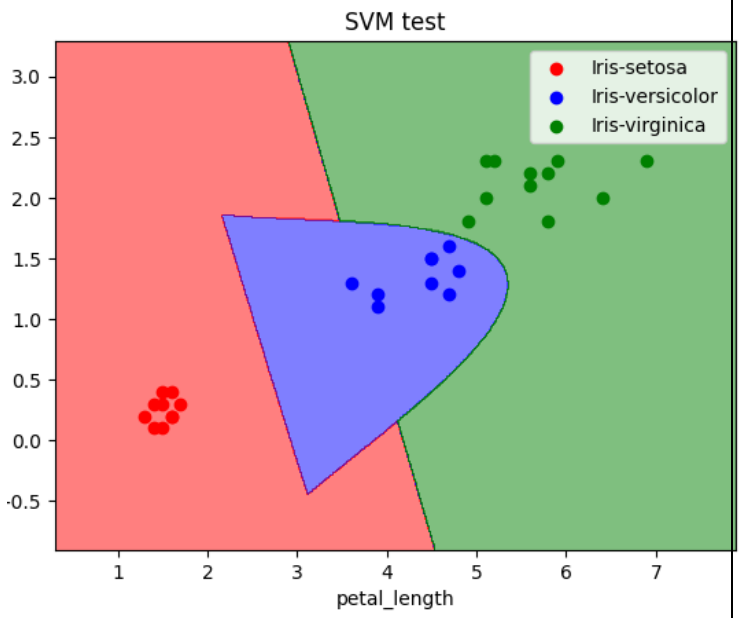
```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  4  7]]
 accuracy_score =  0.800    precision_score =  0.805   recall_score
                                          =   0.805   f1_score =   0.800
```






```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
                                          =   1.000   f1_score =   1.000
```



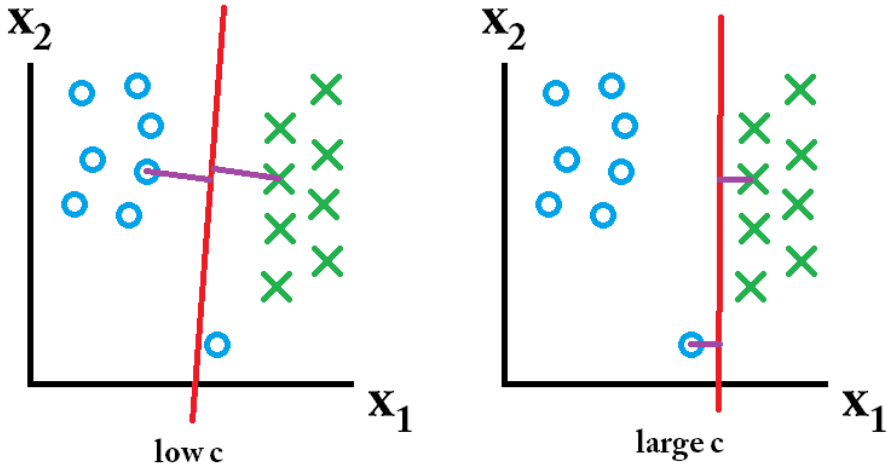


در نتایج فقط بخش sepal در چند جمله ای نتایج بدتر شد آن هم به دلیل درجه و بقیه بخش ها نتایج عوض شد و همانطور که می تونید ببینید مرز جدا کنندگیشونم تغییر کرده که این به دلیل خاصیت آن کرنل هستش

۴

:regularization hyperparameter

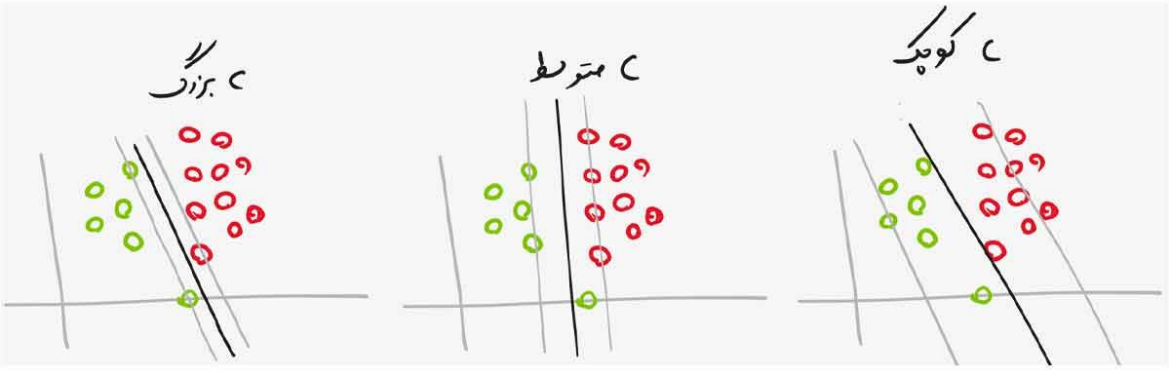
پارامتر C : یک ترم ثابت مثبت است که توسط کاربر تعیین می شود وحاشیه امن (Margin) را کنترل می کند. و نسبت آن عکس اندازه حاشیه امن است.



- **C بزرگ**: C اگر مقدار بزرگی داشته باشد در نتیجه آن قیدهای مسئله بهنیه سازی SVM شدید می شوند و اگر خیلی خیلی بزرگ باشد، مسئله بهینه سازی شبیه Hard Margin می شود. یعنی خطای طبقه بندی باید صفر باشد!
که منظور همون overfit هستش.

- **Cمتوسط**: C اگر مقدار متوسط و حد معمولی داشته باشد در نتیجه آن شدت قیدهای مسئله بهنیه سازی SVMکمتر می شود و به مدل اجازه میدهد مقداری خطا داشته باشد (ولی خطای حداقل نه خیلی زیاد) و همین باعث میشود که مرز مناسبی برای مسئله طبقه بندی پیدا کند.

- **Cکوچک**: C اگر مقدار کوچکی داشته باشد در نتیجه آن شدت قیدهای مسئله بهنیه سازی SVM خیلی کمتر می شود و به مدل اجازه میدهد خطای زیادی داشته باشد که در نتیجه آن ممکن است مدل به داده های پرت (outlier) حساس شود و در نتیجه مرز مناسبی بدست نیاید. مقدار C نباید خیلی کوچک باشد!

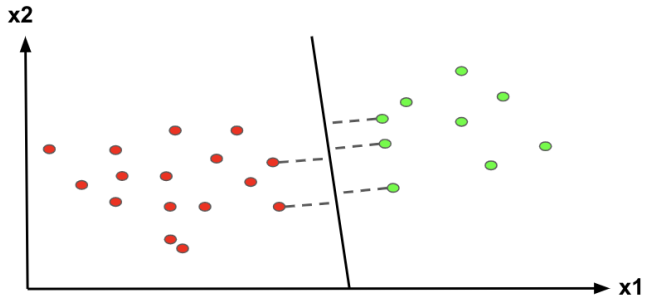


:gamma Hyperparameter

در مورد کرنل RBF یا Poly مفید است. وقتی هسته شما خطی باشد هیچ نقشی ندارد. با توجه به اینکه ما می خواهیم یک تابع پیچیده را تحقق بخشیم،

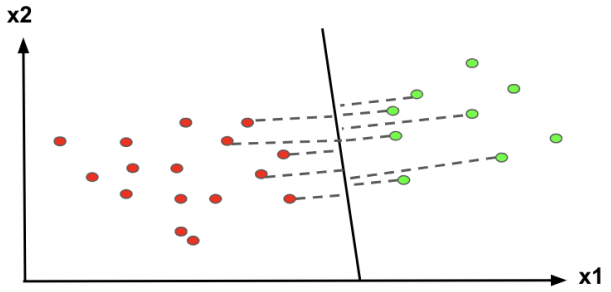
گاما اساساً فاصله تأثیرگذاری یک نقطه تمرین را کنترل می کند.

- مقادیر پایین گاما نشان دهنده یک شعاع شباهت زیاد است که منجر به گروه بندی نقاط بیشتر با هم می شود. یعنی باعث می شود داده های اموزشی دور تر هم اثر گذار باشند و در پیدا کردن خط جدایی دو کلاس تاثیر بگذارند و اگر گاما خیلی پایین باشد احتمالا دچار underfitting می شویم.
- برای مقادیر بالای گاما، نقاط باید بسیار نزدیک به هم باشند تا در یک گروه در نظر گرفته شوند.در این حالت فقط داده های مرزی مشارکتشون بیشتر میشه و ما بیشتر طبق اونها می سنجیم در این صورت ممکنه دچار overfitting شویم



High Gamma

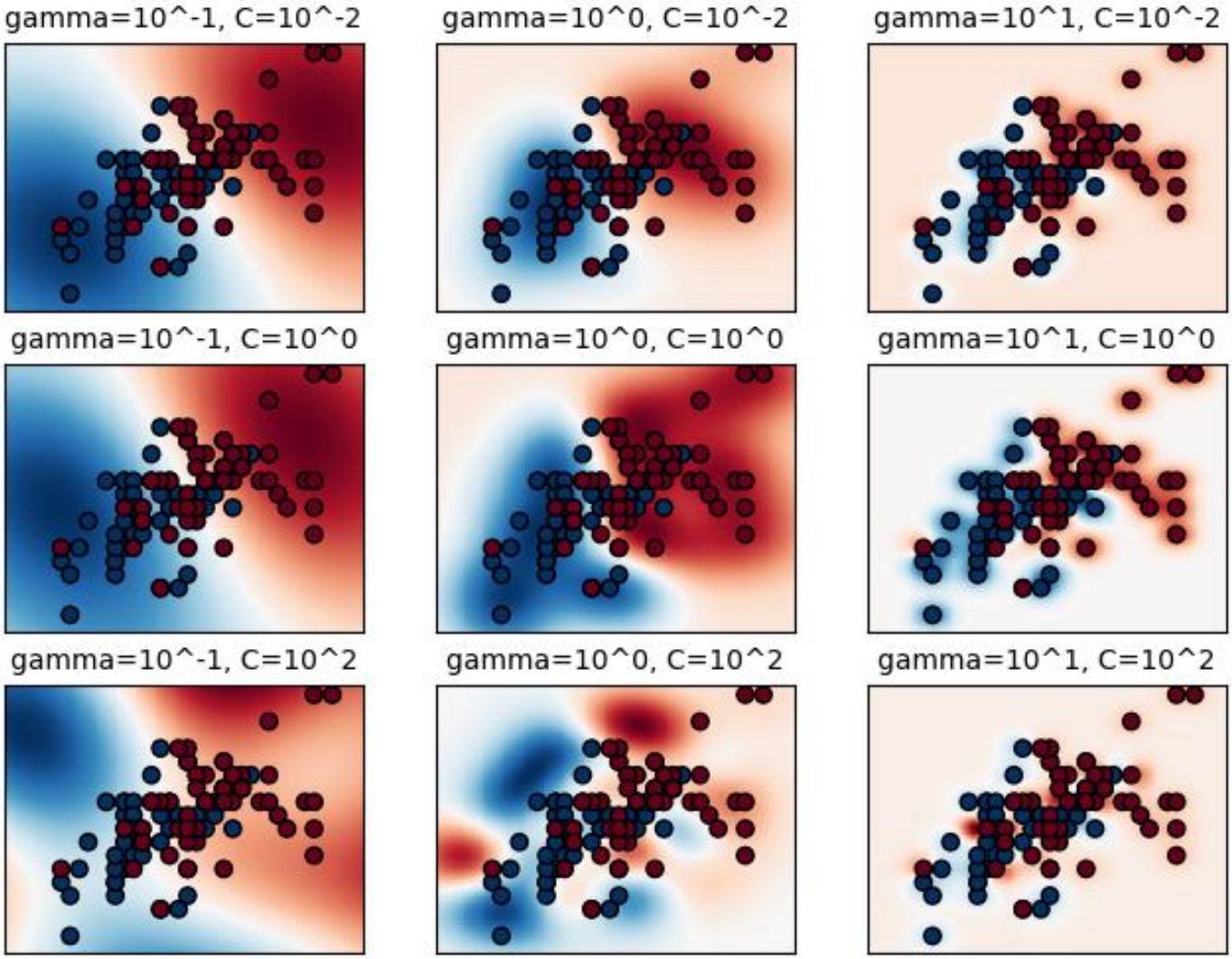
- only near points are considered.



Low Gamma

- far away points are also considered

در کل داریم:

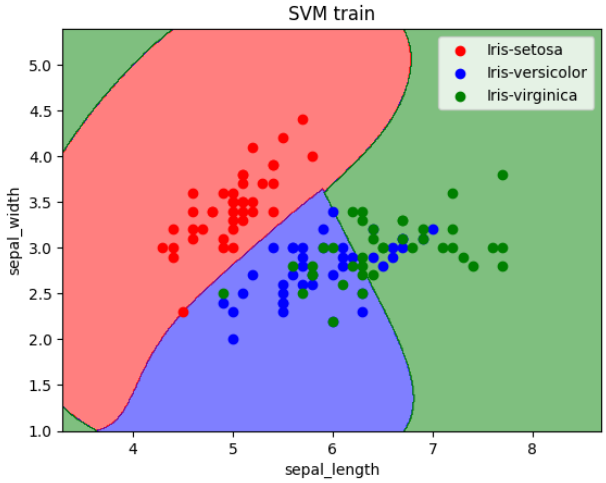

gamma=10^-1, C=10^-2
gamma=10^0, C=10^-2
gamma=10^1, C=10^-2
gamma=10^-1, C=10^0
gamma=10^0, C=10^0
gamma=10^1, C=10^0
gamma=10^-1, C=10^2
gamma=10^0, C=10^2
gamma=10^1, C=10^2

تست مقادیر هایپرپارمتر روی RBF

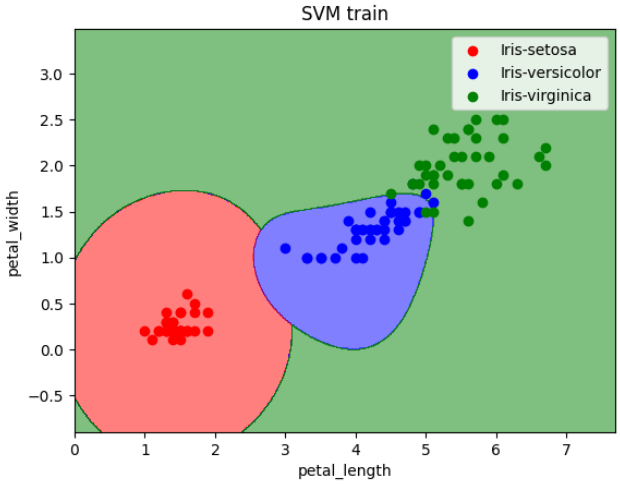
تاثیرC

گاما ۰.۸ و C = 100

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
accuracy_score =  0.900    precision_score =  0.903    recall_score
=   0.896    f1_score =   0.898
```

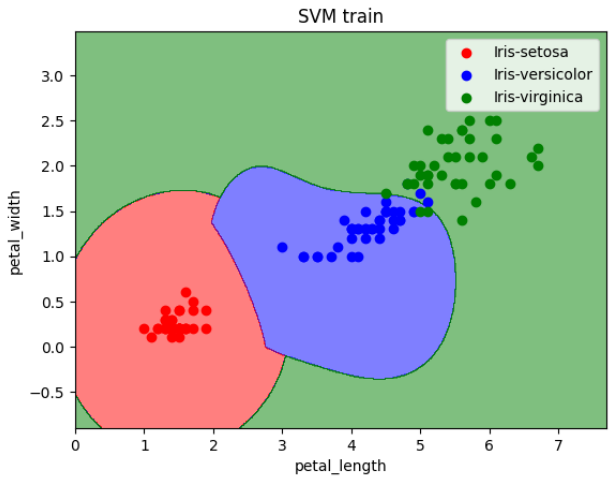



```
confusion matrix :
[[10  0  0]
 [ 0  6  3]
 [ 0  1 10]]
accuracy_score =  0.867    precision_score =  0.875    recall_score
=   0.859    f1_score =   0.861
```

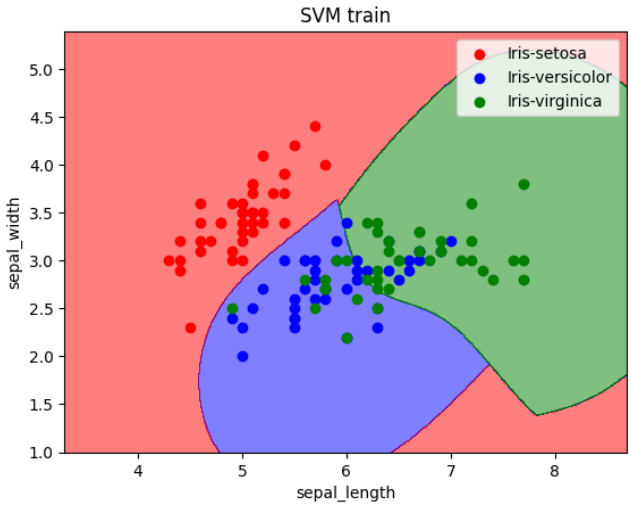


گاما ۰.۸ و C = 10

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

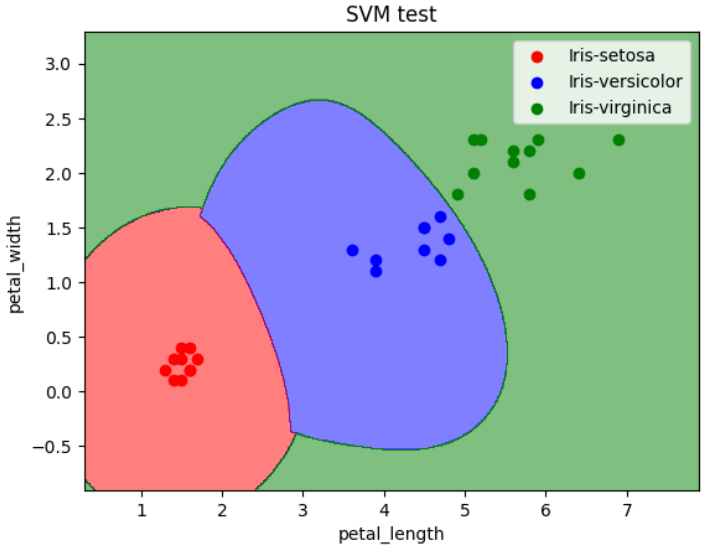



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

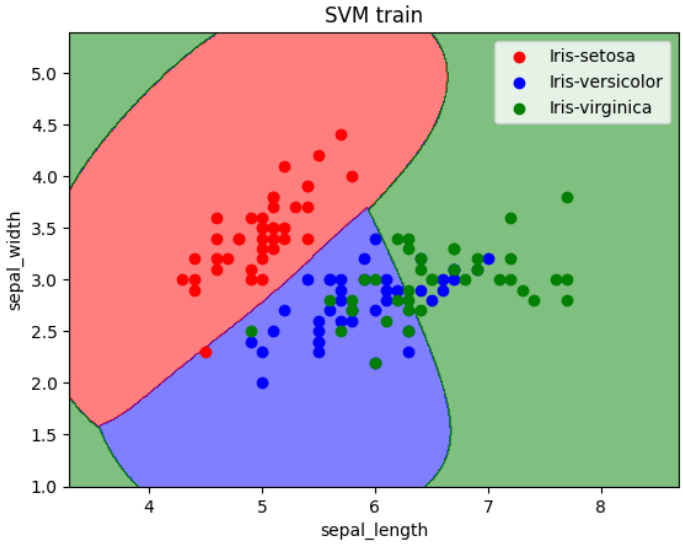


گاما ۰.۸ و C = 0.5

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```




```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

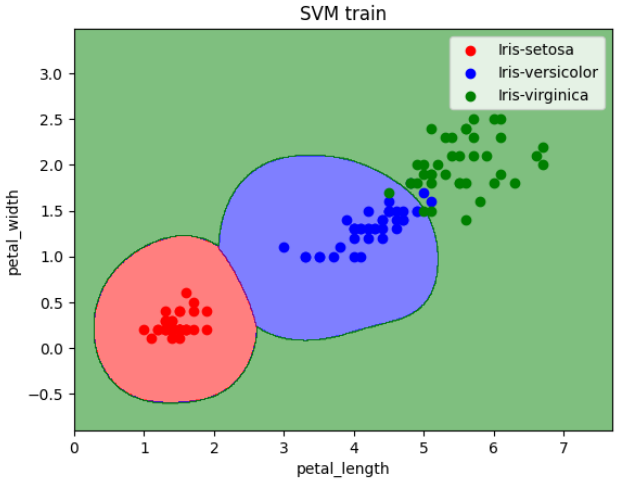


با توجه به مقادیر مختلفc می بینیم که با زیاد شدنش باعث overfitt شدن میشه و باید در رنج مشخصی باشه

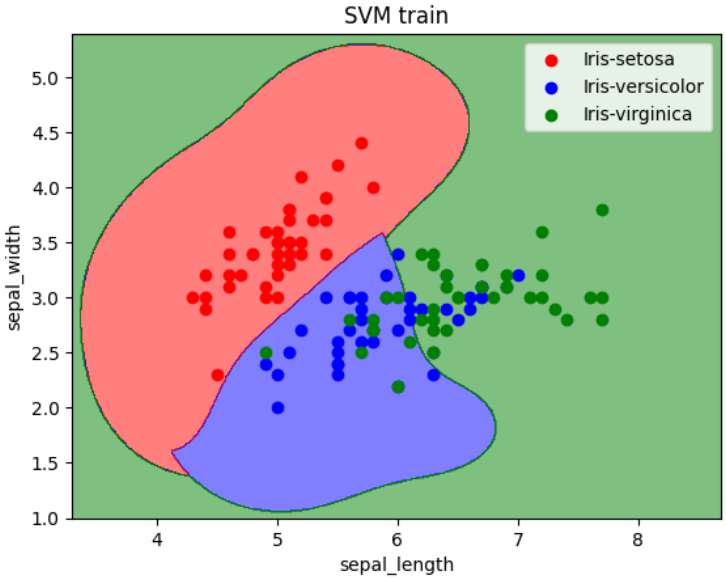
تاثیر گاما

گاما ۲ و C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
                                          =   1.000   f1_score =   1.000
```

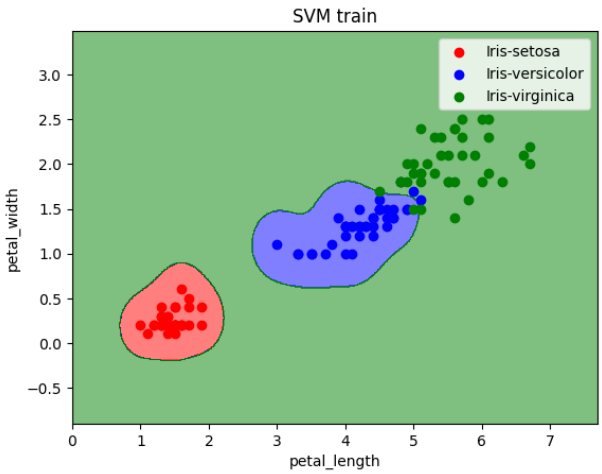



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  0 11]]
 accuracy_score =  0.933    precision_score =  0.949   recall_score
                                          =   0.926   f1_score =   0.931
```

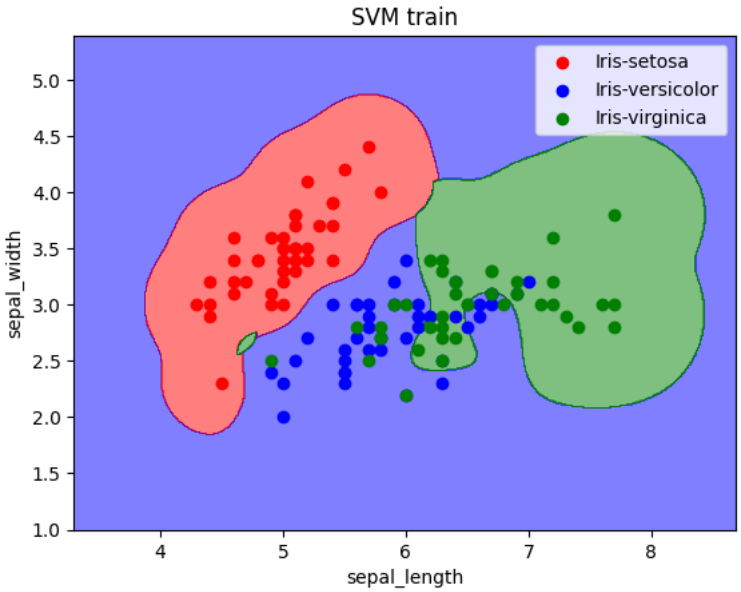


گاما ۱۰ و c=1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

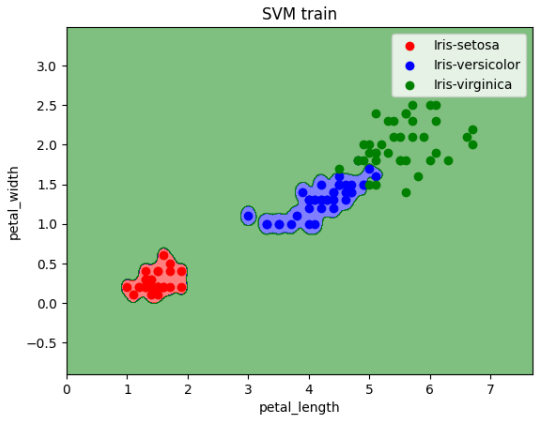


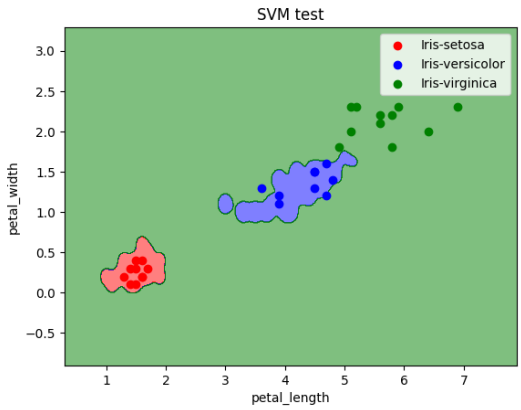

```
confusion matrix :
[[10  0  0]
 [ 0  8  1]
 [ 0  4  7]]
 accuracy_score =  0.833    precision_score =  0.847   recall_score
=   0.842   f1_score =   0.833
```



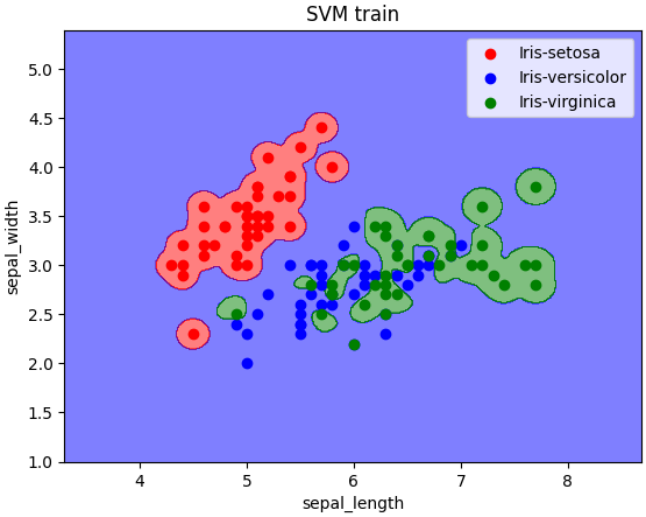
گاما 100 و c=1

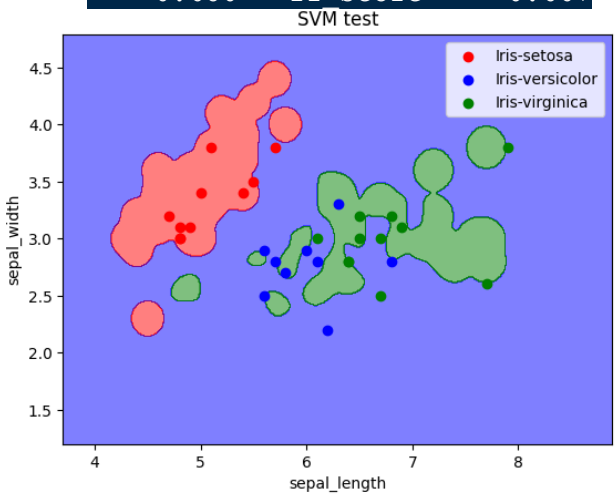
```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  0 11]]
 accuracy_score =  0.933    precision_score =  0.949   recall_score
                                          =   0.926   f1_score =   0.931
```






```
confusion matrix :
[[9 1 0]
 [0 7 2]
 [0 7 4]]
 accuracy_score =  0.667    precision_score =  0.711   recall_score
                                          =   0.680   f1_score =   0.667
```





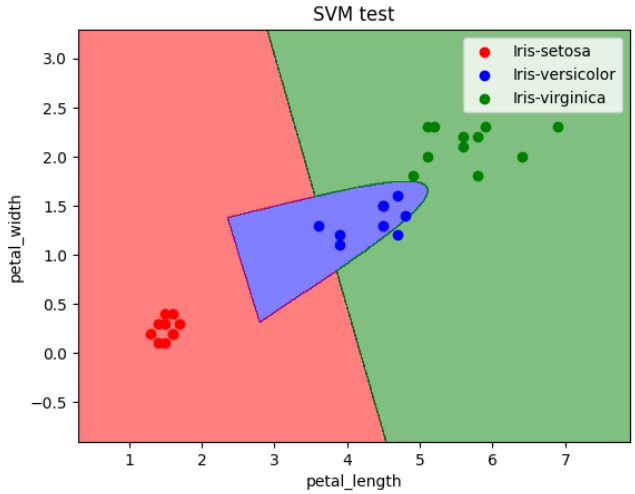
با توجه به نتایج هرچی گاما بیشتر باشد overfitt و هرچی کمتر underfit که قشنگ و واضح می توان دید

تست مقادیر هایپرپارمتر روی poly دیفالت درجه ۳

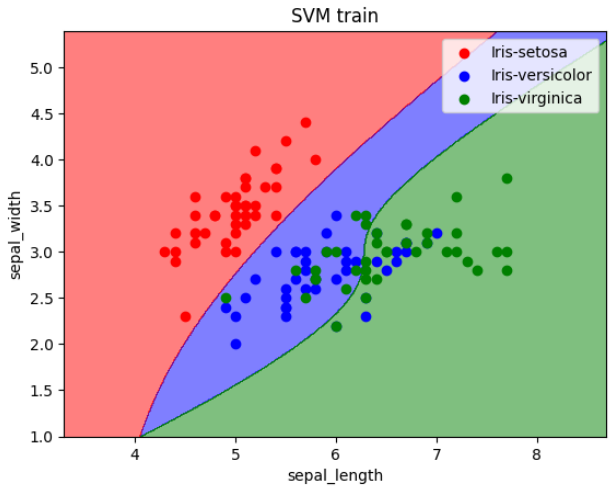
تاثیرC

گاما ۰.۸ و C = 100

```
confusion matrix :
[[10  0  0]
 [ 0  8  1]
 [ 0  0 11]]
 accuracy_score =  0.967    precision_score =  0.972   recall_score
=   0.963   f1_score =   0.966
```

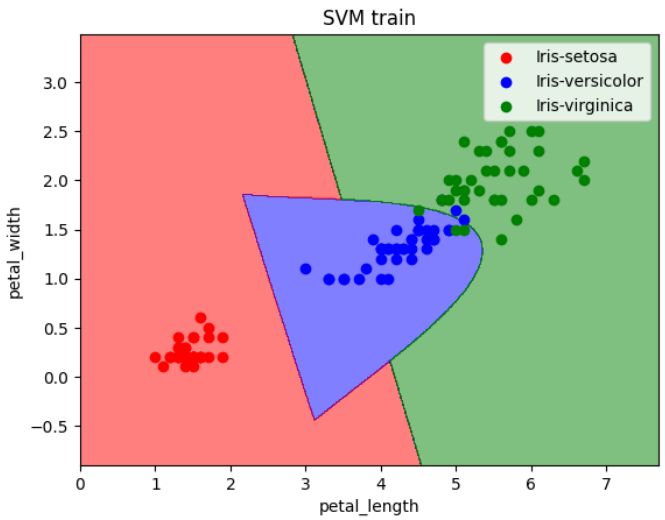



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

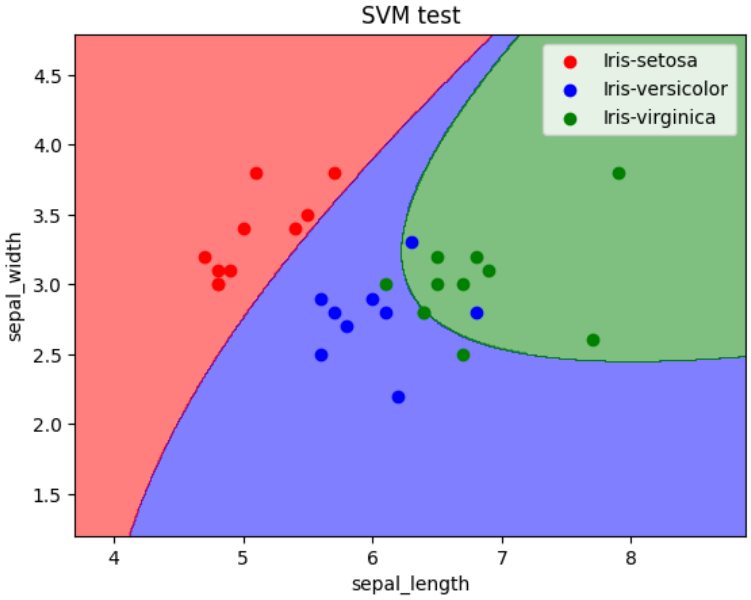


گاما ۰.۸ و C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

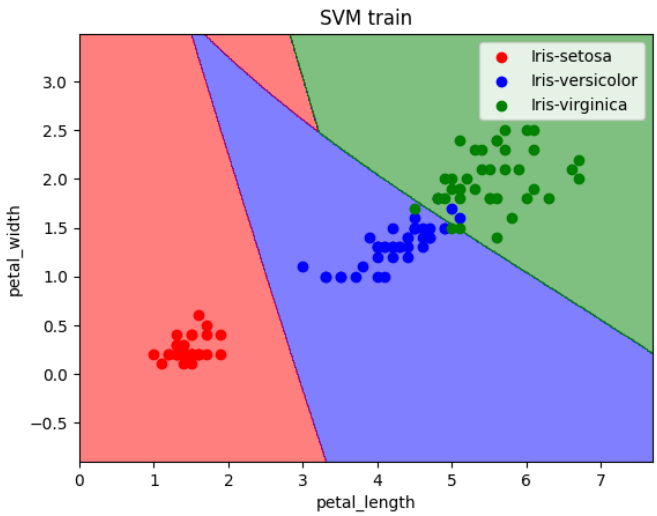



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  4  7]]
 accuracy_score =  0.800    precision_score =  0.805   recall_score
=   0.805   f1_score =   0.800
```

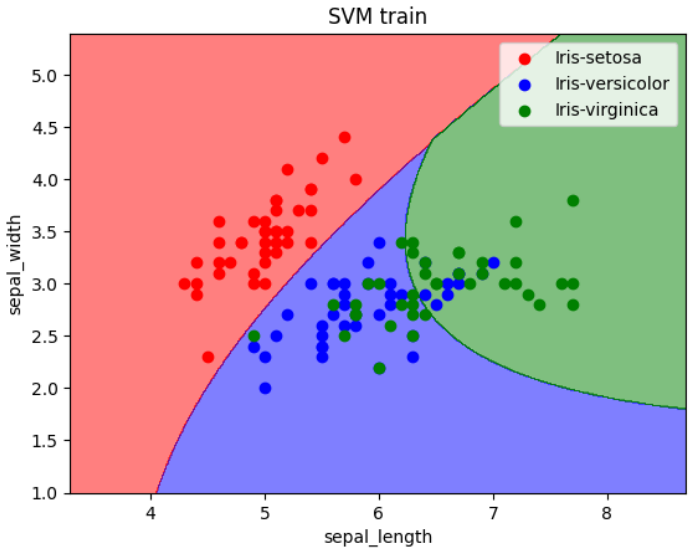


گاما ۰.۸ و C = 0.1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```




```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

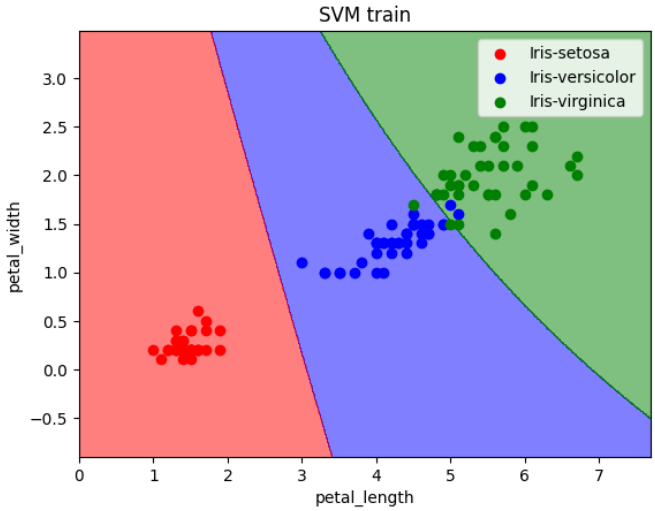


با توجه به نتایج طبق توضیحات گفته بودیم چه اتفاقی می افته

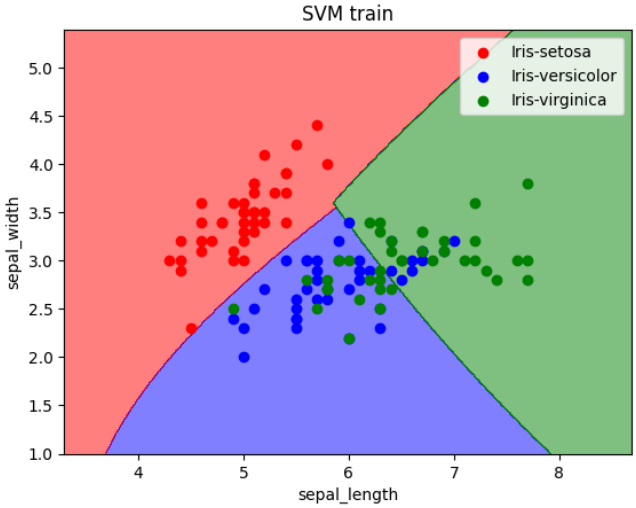
تاثیر گاما

گاما ۰.۱ و C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

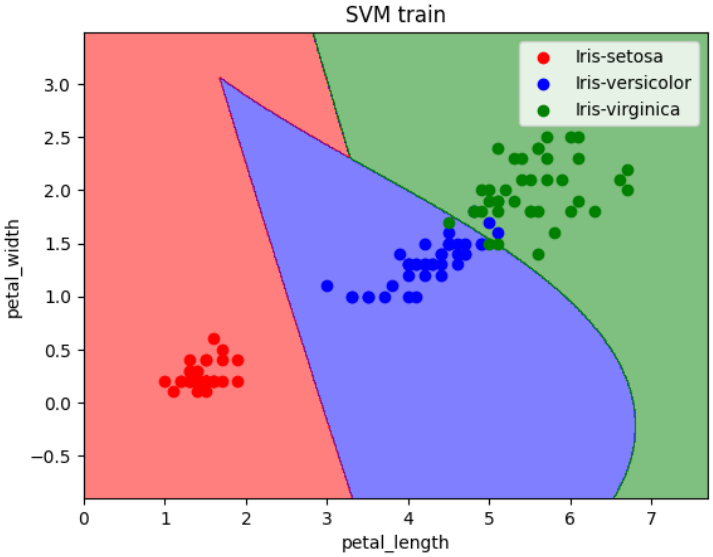



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  3  8]]
 accuracy_score =  0.833    precision_score =  0.833   recall_score
=   0.835   f1_score =   0.833
```

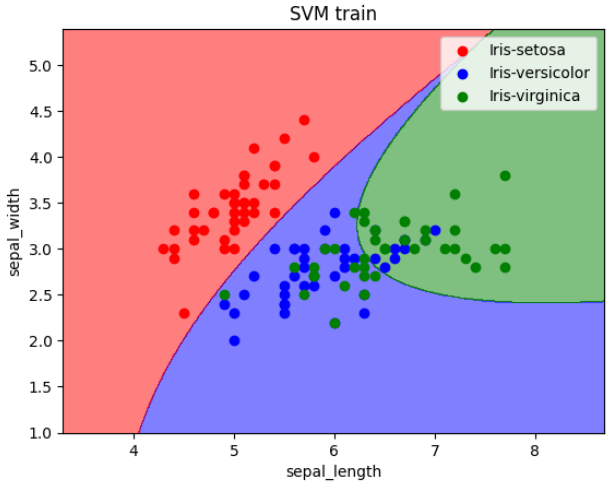


گاما ۱۰۰ و C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

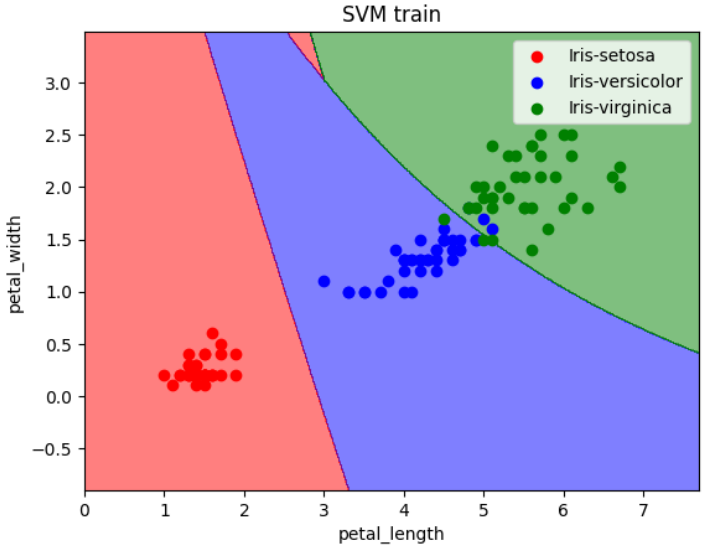



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  4  7]]
 accuracy_score =  0.800    precision_score =  0.805   recall_score
=   0.805   f1_score =   0.800
```

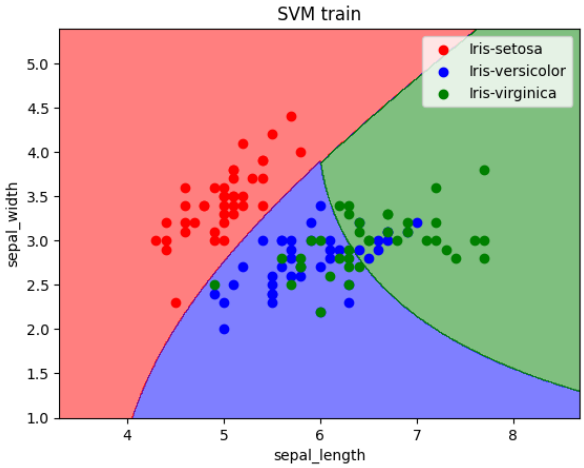


گاما ۱۰ و C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```




```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  3  8]]
 accuracy_score =  0.833    precision_score =  0.833   recall_score
=   0.835   f1_score =   0.833
```



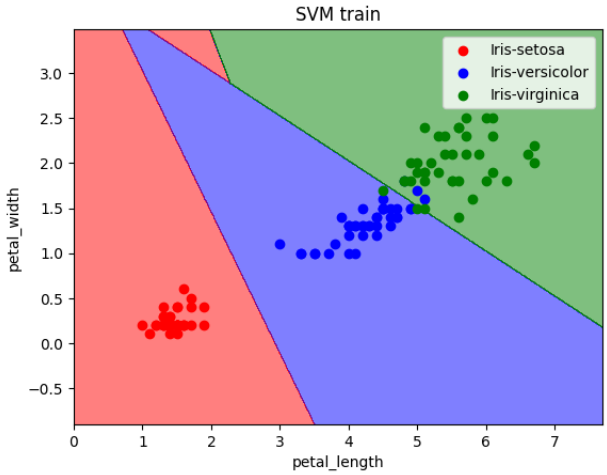
با توجه به نتایج طبق توضیحات گفته بودیم چه اتفاقی می افته

تست مقادیر هایپرپارمتر روی linear

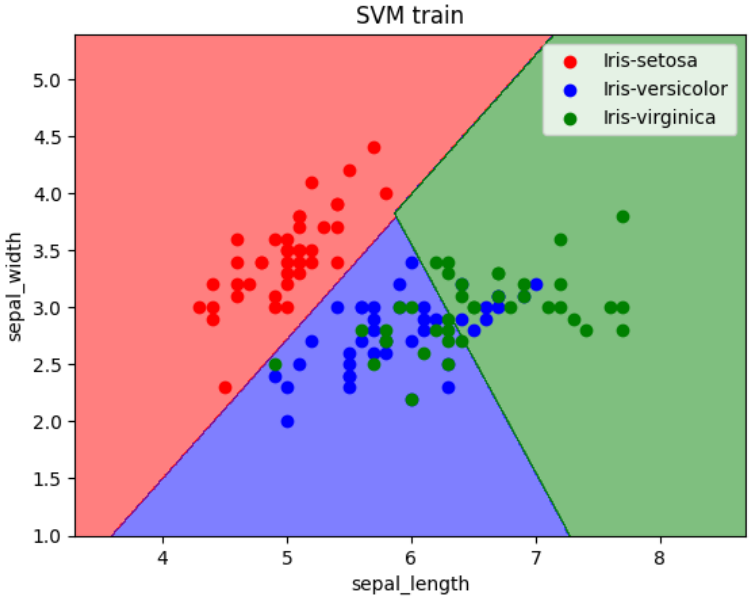
تاثیر C در این بخش گاما نداریم

C = 1000

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```

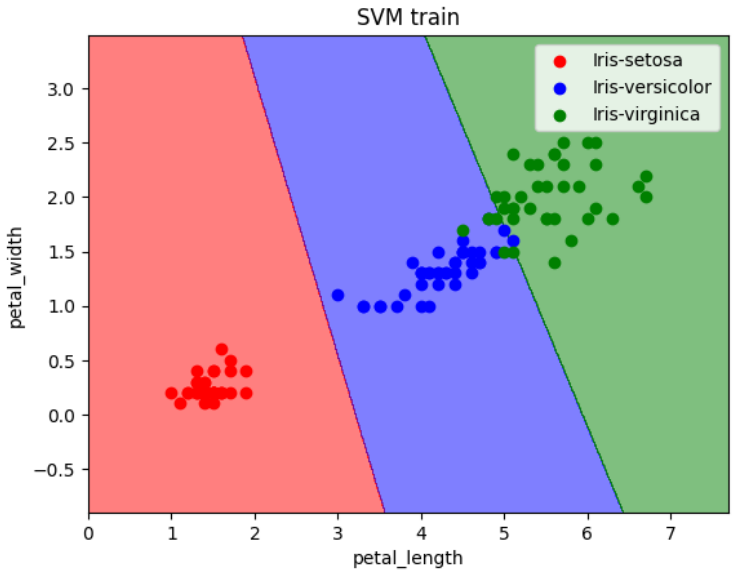



```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

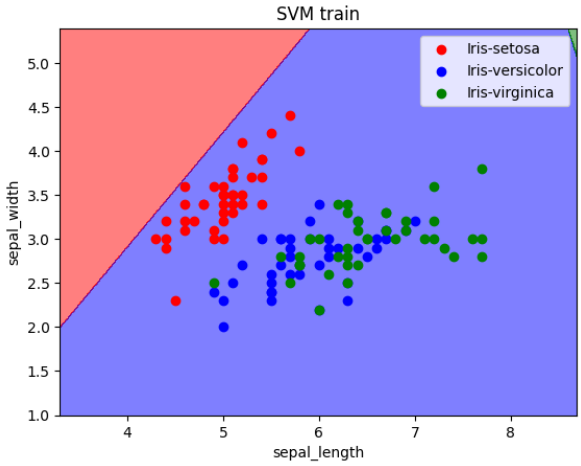


C = 0.01

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  1 10]]
 accuracy_score =  0.967    precision_score =  0.967   recall_score
                                          =   0.970   f1_score =   0.967
```

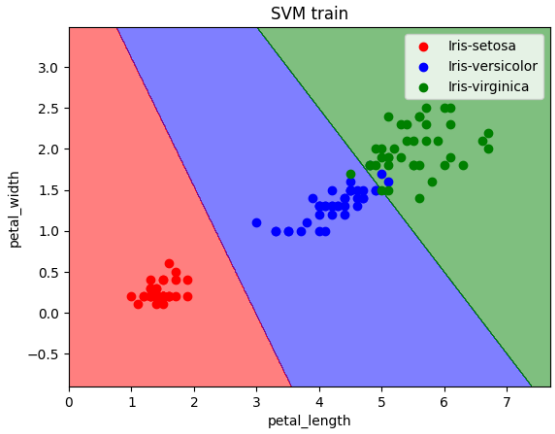



```
confusion matrix :
[[ 0 10  0]
 [ 0  9  0]
 [ 0 11  0]]
 accuracy_score =  0.300    precision_score =  0.100   recall_score
                                          =   0.333   f1_score =   0.154
```

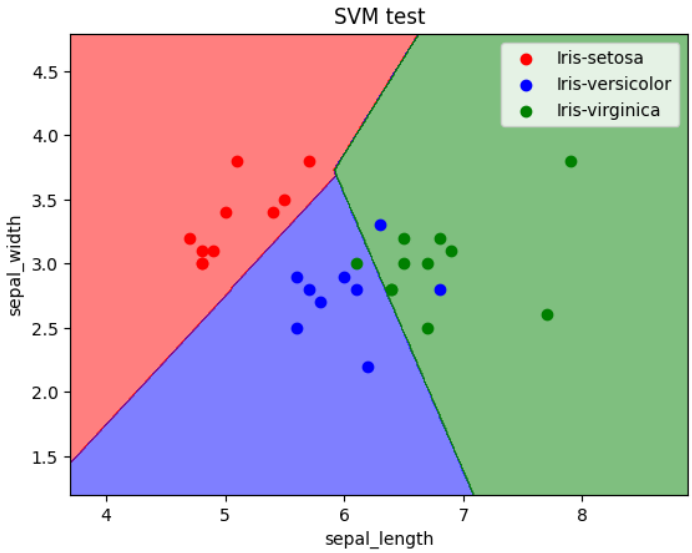


C = 1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
                                        =   1.000   f1_score =   1.000
```




```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
                                        =   0.896   f1_score =   0.898
```



با توجه به نتایج نباید هم زیاد باشه هم کوچیک باشه

5

```
grid_parameters_rbf = {'kernel': ['rbf'],'gamma': [10, 1, 0.1, 0.01, 0.001,
0.0001],'C': [0.1, 1, 10, 100, 1000]}
grid_parameters_linear = {'kernel': ['linear','C': [0.1, 1, 10, 100, 1000]}
grid_parameters_poly = {'kernel': ['poly'],'gamma': [10, 1, 0.1, 0.01, 0.001,
0.0001],'C': [0.1, 1, 10, 100, 1000]
                        ,'degree': [2,  3 , ١]}
```

باتوجه به مقادیر برای خروجی RBF داریم:

```
for petal features :
best parameter :
  {'C': 100, 'gamma': 10, 'kernel': 'rbf'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      0.89      0.94         9
           2       0.92      1.00      0.96        11

    accuracy                           0.97        30
   macro avg       0.97      0.96      0.97        30
weighted avg       0.97      0.97      0.97        30

for sepal features :
best parameter :
  {'C': 1000, 'gamma': 0.01, 'kernel': 'rbf'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       0.64      0.78      0.70         9
           2       0.78      0.64      0.70        11

    accuracy                           0.80        30
   macro avg       0.80      0.80      0.80        30
weighted avg       0.81      0.80      0.80        30
```

باتوجه به مقادیر برای خروجی poly داریم:

```
for petal features :
best parameter :
  {'C': 0.1, 'degree': 3, 'gamma': 10, 'kernel': 'poly'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      1.00      1.00         9
           2       1.00      1.00      1.00        11

    accuracy                           1.00        30
   macro avg       1.00      1.00      1.00        30
weighted avg       1.00      1.00      1.00        30

for sepal features :
best parameter :
  {'C': 0.1, 'degree': 1, 'gamma': 1, 'kernel': 'poly'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       0.88      0.78      0.82         9
           2       0.83      0.91      0.87        11

    accuracy                           0.90        30
   macro avg       0.90      0.90      0.90        30
weighted avg       0.90      0.90      0.90        30
```

باتوجه به مقادیر برای خروجی linear داریم:

```
for petal features :
best parameter :
  {'C': 100, 'kernel': 'linear'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      1.00      1.00         9
           2       1.00      1.00      1.00        11

    accuracy                           1.00        30
   macro avg       1.00      1.00      1.00        30
weighted avg       1.00      1.00      1.00        30

for sepal features :
best parameter :
  {'C': 0.1, 'kernel': 'linear'}
report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       0.88      0.78      0.82         9
           2       0.83      0.91      0.87        11

    accuracy                           0.90        30
   macro avg       0.90      0.90      0.90        30
weighted avg       0.90      0.90      0.90        30
```

نتیجه گیری

باتوجه به نتایج بهترین گزینه برا طبقه بندی آنها همان خطی است البته به دلیل کم بودن داده ها مشخص نیست اما اگر تعداد داده بیشتر بود نتایجی میتوانست روش دیگری را بهتر نشان دهد

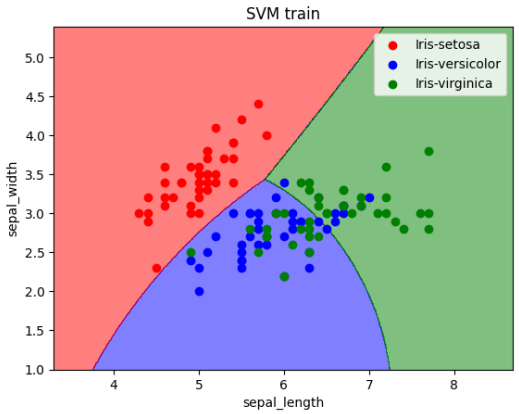
۶

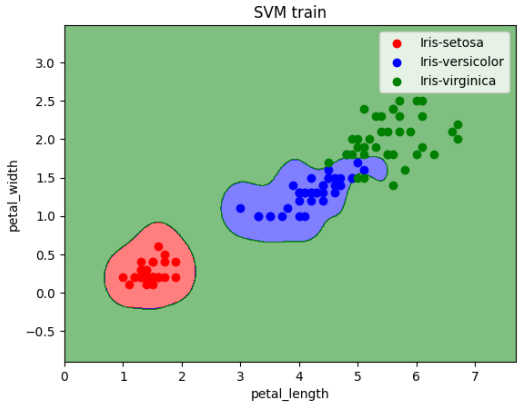
One vs all

RBF

باتوجه به نتایج بخش قبل برای sepal گاما 0.01 و c=1000 و برای petal گاما 10و c=100

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```





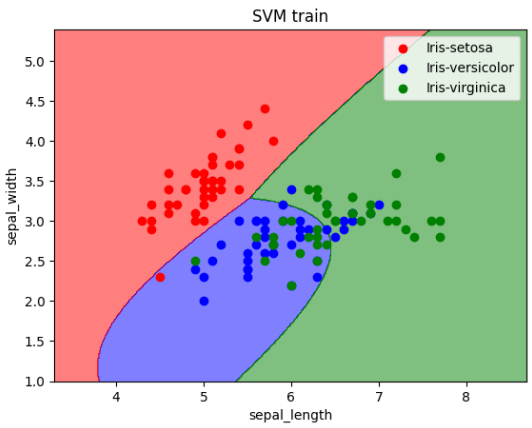

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  4  7]]
 accuracy_score =  0.800    precision_score =  0.805   recall_score
=   0.805   f1_score =   0.800
```

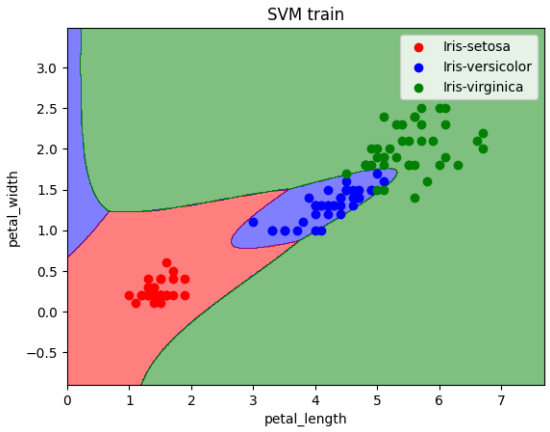
اگه می بینید نتایج فرق کرده به این دلیل که داده های تست و ترین با بخش ۵ فرق داشته

Poly

باتوجه به نتایج بخش قبل برای sepal گاما 0.1 و c=0.1 و برای petal گاما 10 c=0.1

```
confusion matrix :
[[10  0  0]
 [ 0  8  1]
 [ 0  0 11]]
 accuracy_score =  0.967    precision_score =  0.972   recall_score
=   0.963   f1_score =   0.966
```





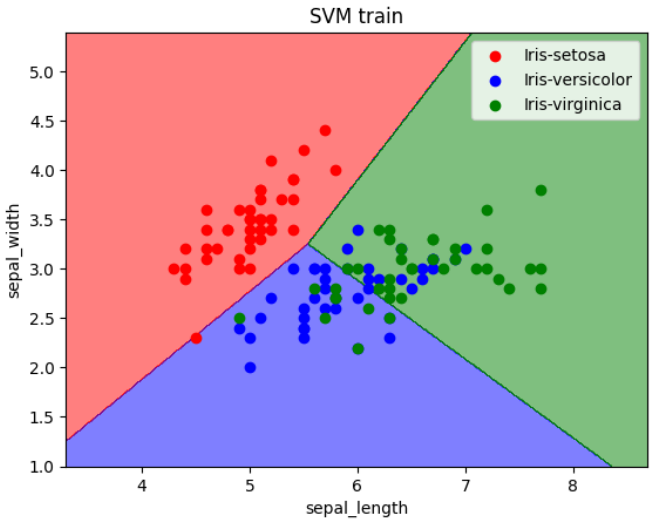

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  3  8]]
 accuracy_score =  0.833    precision_score =  0.833   recall_score
=   0.835   f1_score =   0.833
```

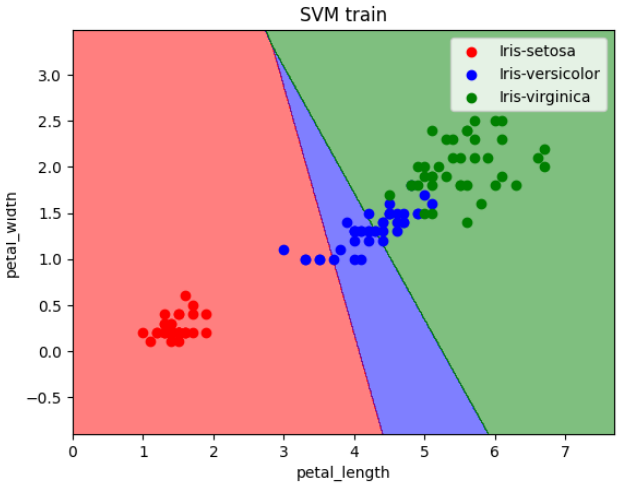
در اینجا هم می بینیم که سپال برای خطی بهتر بود و اینجا ضعف دارد چون در صورت سوال نوشته درجه سه

Linear

باتوجه به نتایج بخش قبل برای sepal c=0.1 و برای petal c=0.1

```
confusion matrix :
[[10  0  0]
 [ 0  3  6]
 [ 0  0 11]]
 accuracy_score =  0.800    precision_score =  0.882   recall_score
=   0.778   f1_score =   0.762
```






```
confusion matrix :
[[10  0  0]
 [ 0  6  3]
 [ 0  0 11]]
 accuracy_score =  0.900    precision_score =  0.929   recall_score
=   0.889   f1_score =   0.893
```

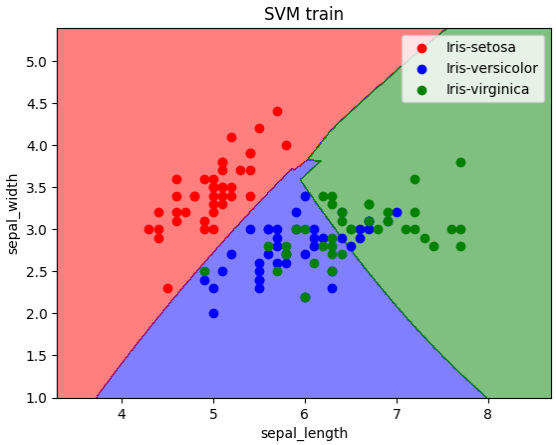
می بینیم که داده های های آبی وسطی بیشترین ضرر رو کردن و محدودشون از بقیه کمتر شده

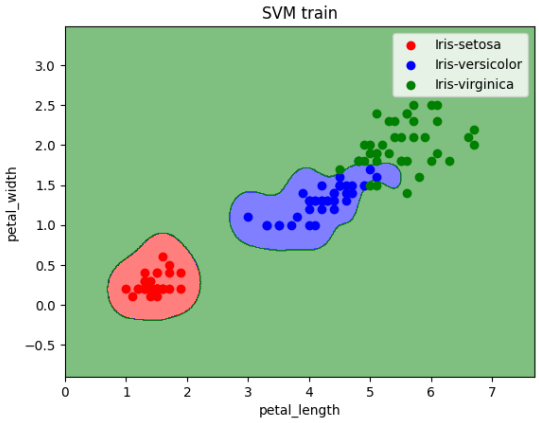
One vs one

RBF

باتوجه به نتایج بخش قبل برای sepal گاما 0.01 و c=1000 و برای petal گاما 10و c=100

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```



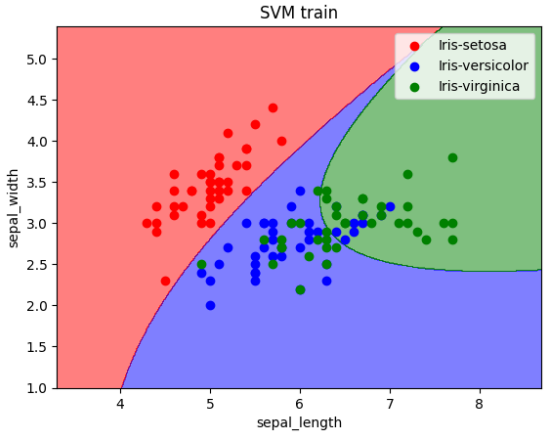


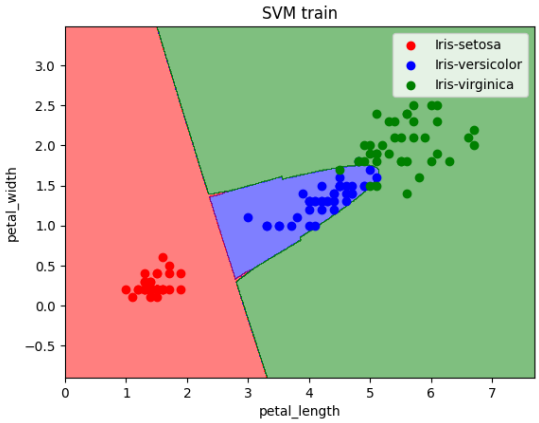

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  3  8]]
 accuracy_score =  0.833    precision_score =  0.833   recall_score
=   0.835   f1_score =   0.833
```

Poly

باتوجه به نتایج بخش قبل برای sepal گاما 0.1 و c=0.1 و برای petal گاما 10 c=0.1

```
confusion matrix :
[[10  0  0]
 [ 0  8  1]
 [ 0  0 11]]
 accuracy_score =  0.967    precision_score =  0.972   recall_score
=   0.963   f1_score =   0.966
```





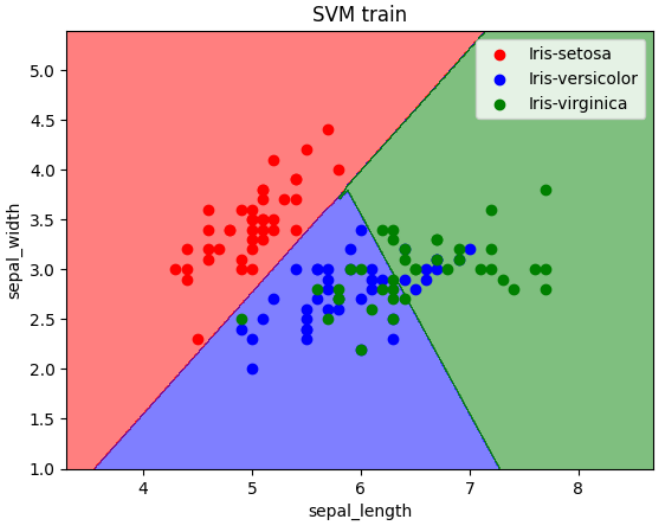

```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  4  7]]
 accuracy_score =  0.800    precision_score =  0.805   recall_score
=   0.805   f1_score =   0.800
```

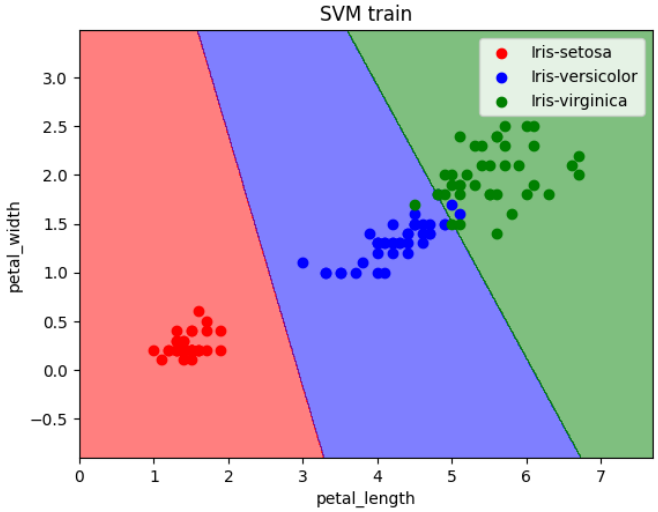
در اینجا هم می بینیم که سپال برای خطی بهتر بود و اینجا ضعف دارد چون در صورت سوال نوشته درجه سه

Linear

باتوجه به نتایج بخش قبل برای sepal c=0.1 و برای petal c=0.1

```
confusion matrix :
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
 accuracy_score =  1.000    precision_score =  1.000   recall_score
=   1.000   f1_score =   1.000
```






```
confusion matrix :
[[10  0  0]
 [ 0  7  2]
 [ 0  1 10]]
 accuracy_score =  0.900    precision_score =  0.903   recall_score
=   0.896   f1_score =   0.898
```

نتیجه کلی

با توجه به نتایج فک می کنم که برای قسمت آبی (گل وسطی) در روش one vs all کمتر ناحیه هست اما در روش one vs one مرز تصمیم گیریش بیشتر میشه و نتایج هم بهتر است.

5) 1)

$$\|\theta(x)-\theta(y)\|^2 = \|\theta(x)\|^2 + \|\theta(y)\|^2 - 2\theta^T(x)\theta(y)$$

$$\|\theta(x)\|^2 = \theta^T(x)\theta(x) = K(x,x) = 1$$

$$\|\theta(y)\|^2 = K(y,y) = 1$$

$$\Rightarrow \|\theta(x)-\theta(y)\|^2 = 1+1-2\theta^T(x)\theta(y)$$

$$= 2 - 2e^{\left(-\frac{1}{2}\|x-y\|^2\right)}$$

$$\|x-y\|^2 > 0 \Rightarrow e^{\left(-\frac{1}{2}\|x-y\|^2\right)} > 0$$

$$2 - 2e^{\left(-\frac{1}{2}\|x-y\|^2\right)} < 2$$ تصحیح نمره

$$\|\theta(x)-\theta(y)\|^2 < 2$$ تصحیح نمره

۱۳۹۶ شهریور ۶ دوشنبه
28Aug.2017 ■ ۶ ذی الحجه ۱۴۳۸ ■

5.2) $K(x,y) = (x^T y + 1)^2 = (x^T y)^2 + 1 + 2x^T y$

$$= \left(\sum_{i=1}^{n} x_i y_i\right)^2 + 2\sum_{i=1}^{n} x_i y_i + 1$$

$$= \underbrace{\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i x_j y_i y_j)}_{\underbrace{\sum_{i=1}^{n} x_i^2 y_i^2}_{①} + \underbrace{2\sum_{i\neq j} x_i y_i x_j y_j}_{②}} + \underbrace{2\sum_{i=1}^{n} x_i y_i}_{③} + \underbrace{1}_{④}$$

① $\Rightarrow (x_1^2, x_2^2 \cdots x_n^2)$

② $\Rightarrow (\underbrace{x_1x_2, x_1x_3 \cdots x_1x_n}_{x_1}, \underbrace{x_2x_3 \cdots x_2x_n}_{x_2}, x_{n-1}x$

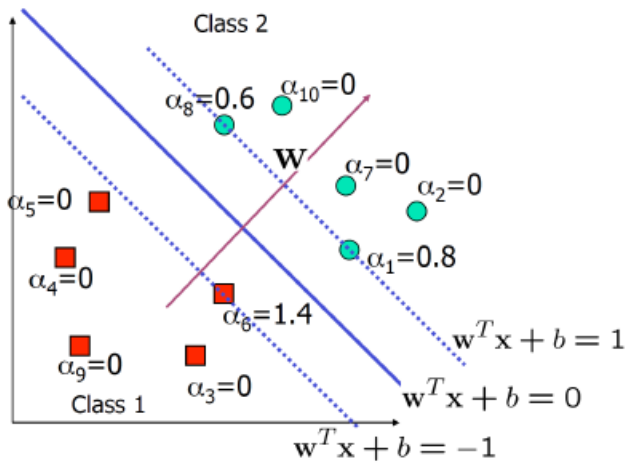
③ $\Rightarrow (x_1, x_2, \cdots, x_d)$

$\Rightarrow$ ④ $\Rightarrow (1)$

تعداد بعد $= \binom{n+2}{2} = \frac{(n+2)(n+1)}{2}$

۳

باتوجه به عکس تعداد داده های لازم برای دو دسته سه تا هست که می تونید ببینید



و این تعداد ممکنه بیشتر بشه اما کمتر ۳ نمی تونه بشه برا همین اگه یه داده نامعلوم دیگر اضافه کنیم ممکنه همین ۳ بمونه یا حتی به تعداد sv برسه

6) 1) $N = 5$ $\qquad P(\text{پشت}) = P(\text{رو}) = \frac{1}{2}$

$$P(\text{error}) = P(\text{error} \mid \text{رو})\,P(\text{رو}) + P(\text{error} \mid \text{پشت})\,P(\text{پشت})$$

$$= \left(P(\text{error} \mid \text{رو}) + P(\text{error} \mid \text{پشت})\right) P(\text{رو})$$

$$= \left(\underbrace{P(\text{error} \mid \text{رو})}_{①} + \underbrace{P(\text{error} \mid \text{پشت})}_{②}\right) \frac{1}{2}$$

$$① = \binom{5}{2}(0.51)^3(0.49)^2 + \binom{5}{1}(0.51)^4(0.49)^1 + \binom{5}{0}(0.49)^5$$

$$= 0.48125$$

$① = ②$

$$P(\text{error}) = \frac{1}{2}(0.48125 + 0.48125) = 0.48125$$

$$P(\text{true}) = 1 - P(\text{error}) = 0.51875$$

2) $N=9$

$P(\text{error} \mid \text{رد}) = P(\text{error} \mid \text{پذیرش})$

$$= \binom{9}{9}(0.51)^9 + \binom{9}{8}(0.51)^8(0.49) + \binom{9}{7}(0.51)^7(0.49)^2$$

$$+ \binom{9}{6}(0.51)^6(0.49)^3 + \binom{9}{5}(0.51)^5(0.49)^4 \simeq 0.524$$

$P(\text{error}) := 1 - P(\text{error}) = 0.476$

ج- دقت به کمک 100 تا مبلغ کند چون تعداد که بیشتر شود خطا کمتر می‌شود

و دقت بالا می‌رود

---

4)

$P(\text{error} \mid \text{پذیرش}) = P(\text{error} \mid \text{رد})$

$$= \binom{5}{2}(0.5)^5 + \dots + \binom{5}{0}(0.5)^5 = 0.5$$

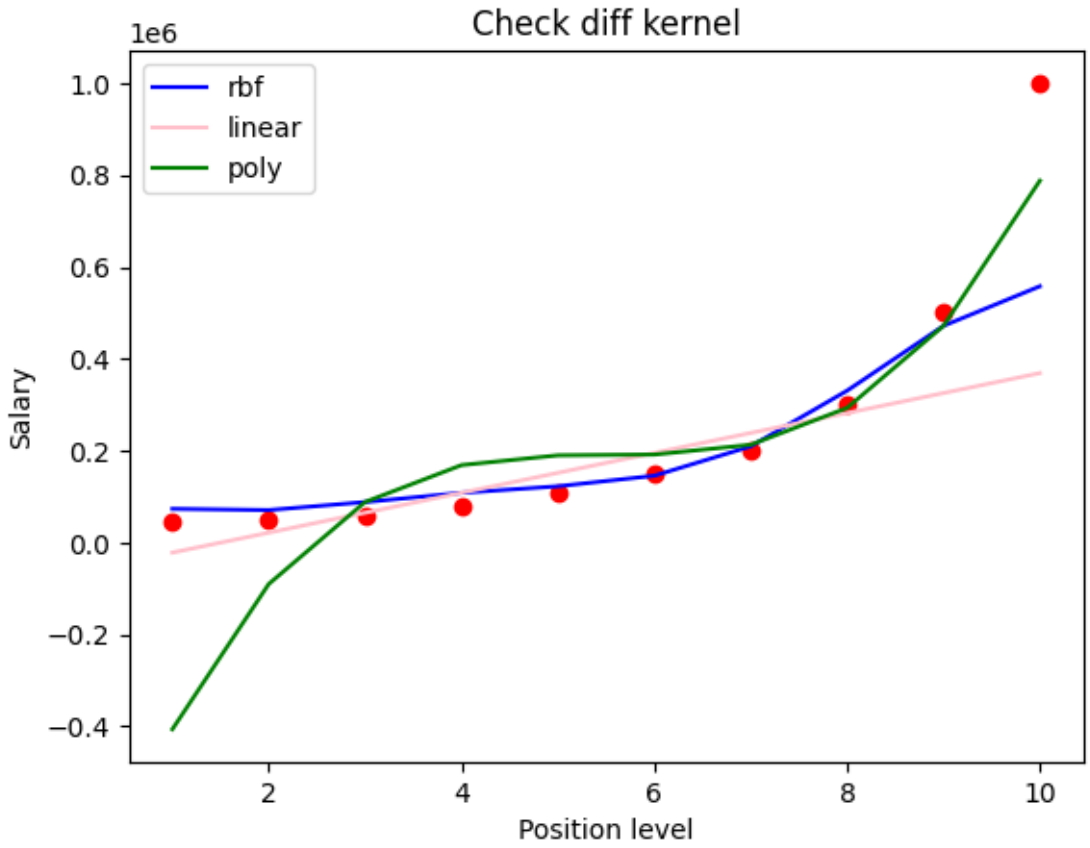
$P(\text{true}) = 1 - 0.5 = \boxed{0.5}$

با توجه به نتایج می‌فهمیم که اگر دقت طبقه‌بندی $\frac{1}{2}$ باشد با افزایش تعداد

طبقات خطا و دقت ثابت می‌ماند و بد

7

الف



نتیجه خروجی نهایی به شکل رو به رو هست همینطور که می بینید هر کدام یک مشکلی دارد

برای مثال linear خیلی سادس و خطای آن زیاد

و برای poly با افزایش position level خیلی خوب همگام می شود و خطای کمی دارد اما ابتدای position level مقادیر درآمد منفی است که اصلا ایده آل نیست

برای RBF می بینیم مقادیر به خیلی خیلی خوب فیت شدن اما آخرین داده که خیلی بزرگ هست رو نتونسته به خوبی پیش بینی کنه

در کل RBF بهترین عملکرد رو داره که البته با تنظیم هایپر پارامتر ها میشه همشون رو بهتر کرد

```
rbf value :
 [[ 73474.15615697]
 [ 70786.94584626]
 [ 88213.00902103]
 [108254.98574956]
 [122574.52250599]
 [145503.10688572]
 [209410.23132923]
 [330606.89204569]
 [471671.95587315]
 [557821.75899897]]
linear value :
 [[-21802.18980308]
 [ 21598.9053814 ]
 [ 65000.00056589]
 [108401.09575037]
 [151802.19093486]
 [195203.28611934]
 [238604.38130383]
 [282005.47648831]
 [325406.5716728 ]
 [368807.66685728]]
```

```
poly value :
 [[-406153.33381986]
 [ -90097.02410228]
 [  88401.09900247]
 [ 168643.37452662]
 [ 189932.14150242]
 [ 191569.7389621 ]
 [ 212858.50593789]
 [ 293100.78146205]
 [ 471598.9045668 ]
 [ 787655.21428438]]
real value :
 [[  45000.]
 [  50000.]
 [  60000.]
 [  80000.]
 [ 110000.]
 [ 150000.]
 [ 200000.]
 [ 300000.]
 [ 500000.]
 [1000000.]]
```

ب

در این مسیر بعضی از داده که تعریف نشده بودن را حذف بعضی داده ها که رنج قیمت یک شب را صفر بود را حذف کردیم

سپس نرمالایز کردیم و ترین RBF با مقادیر دیفالت داریم :

خروجی نهایی

```
SVR Training MSE: 0.6605444862777878
SVR Training R2 Score: -0.4248320147706788
```

فایل مورد نظر پیوست شد