

عرفان ابراهیمی سوز

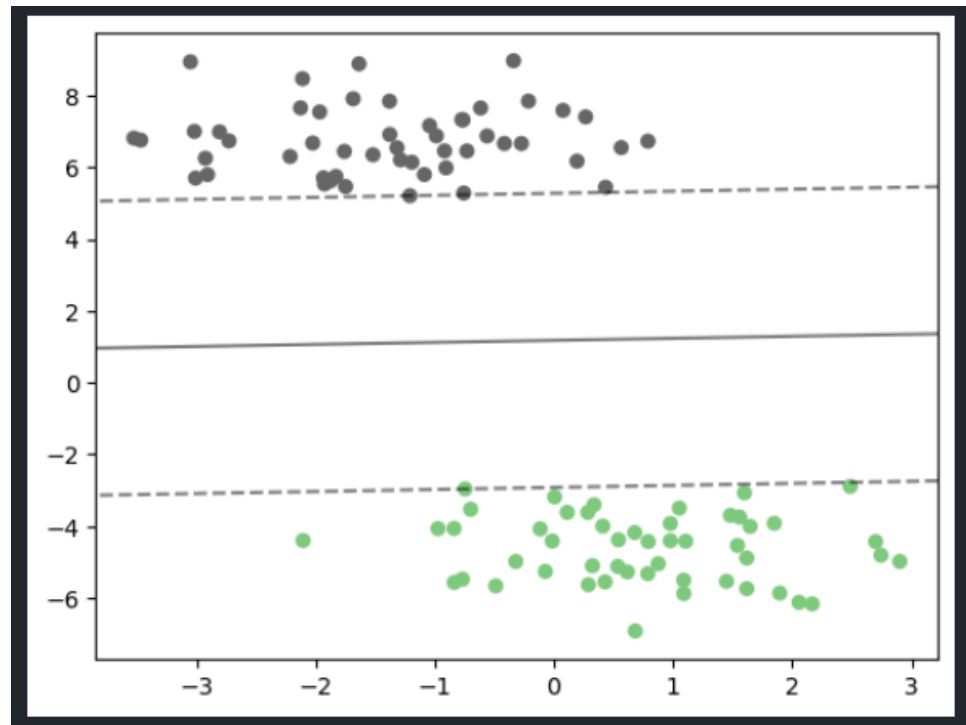
۹۹۵۲۲۶۶

اجرا کد:

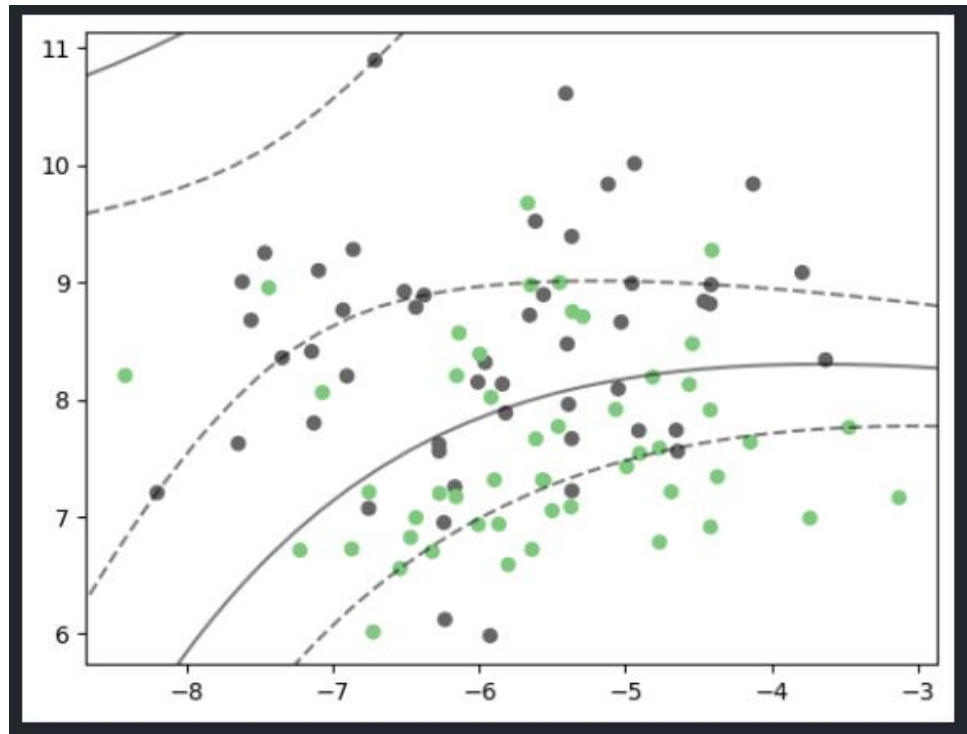
با زدن run all در جویپتر می توانید ران شدن برنامه را ببینید

قسمت الف:

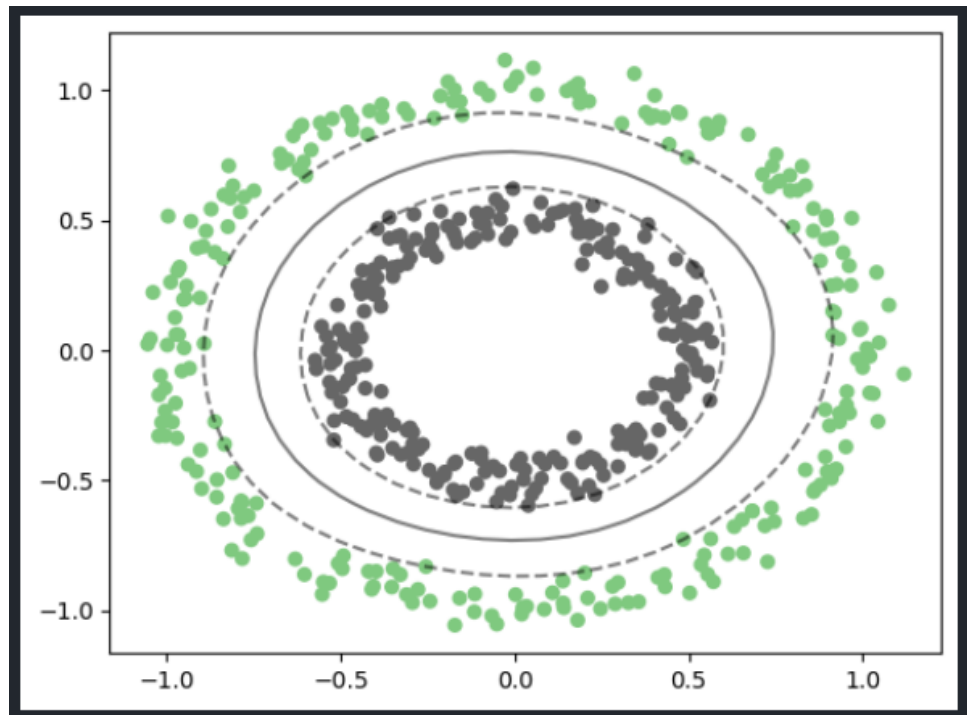
مرحله اول ایجاد دیتاست است که با استفاده از تابع های آماده sklearn این کار را انجام می دهیم اولین تابع make_blobs است که دایره هایی درست می کنن که از هم جدا هستند و آن ها را می توان به راحتی با استفاده از kernel linear جدا کنیم.



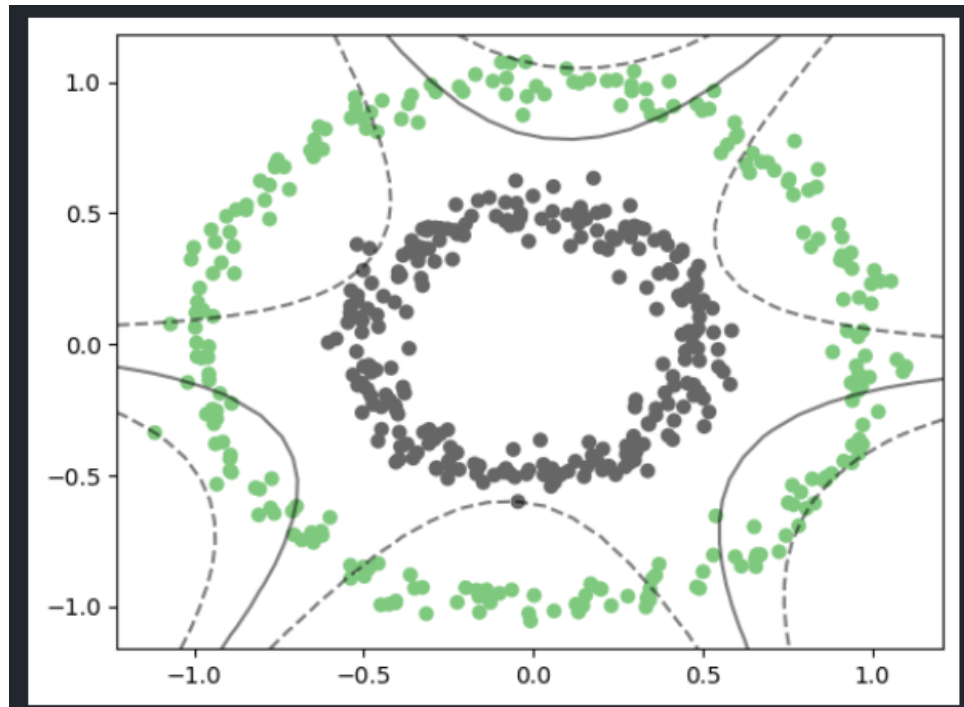
اگر در همین تابع از $random = 5$ استفاده کنیم نقطه ها در هم می روند و دیگر نمی توان آن ها را به راحتی جدا کرد یا حداقل در دو بعد نمی توان آن جدا کردن یا تقسیم کردن آن ها را به راحتی نشان داد



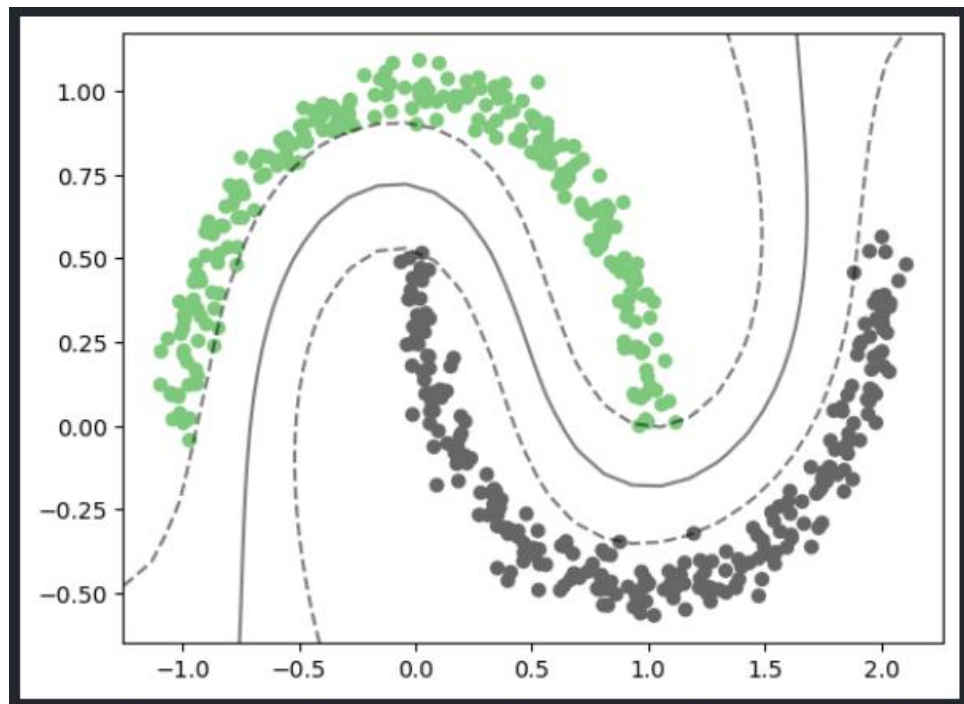
حال از `make_circle` استفاده می کنیم در این صورت می توانیم دایره درست کنیم و با استفاده از `rbf` آن را از هم جدا کنیم



اما اگر از `poly` استفاده کنیم نمی تواند به خوبی نشان دهد زیرا دقت آن کمتر است:

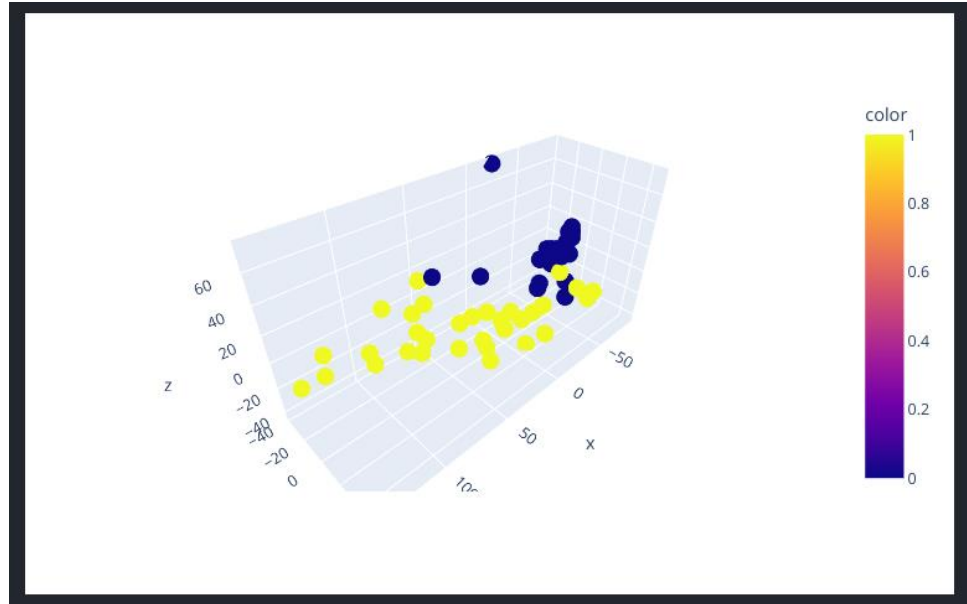


با استفاده از تابع ها می توان ماهم درست کرد که با rbf می توان خط و margin را پیدا کرد:



قسمت ب:

اول باید بتوانیم عکس ها را بخوانیم با استفاده از `imread` می توان عکس ها را خواند و به سیاه سفید تبدیل کرد و همین باید سائز عکس ها با هم برابر باشند حال این عکس ها درواقع یک ماتریس هستند که باید به لیست تبدیل شوند در نهایت عکس ها به ۶۰۰۰۰ ها تا دیتا تبدیل می شوند که باید آن ها را با `pca` به سه یا هر چند تا که لازم است تقسیم کرد که بعد از تقسیم کردن خروجی آن ها به این صورت می شود.



حال با `gridsearchCV` بهترین پارامتر ها را برای `svm` پیدا می کنیم و مدل خود را می سازیم بعد باید `crossval` بزنییم تا بفهمیم دیتا ها برای `svm` خوب هستند یا نه بعد می توانیم `learn` کنیم بعد از `learn` کردن می توانیم دقت آن را بسنجیم.

قبل از `pca`

	0	1	2	3	4	5	6	7	8	9	...	67491	67492	67493	67494	67495	67496	67497	67498	67499	Target
0	0.897723	0.970112	0.987153	0.897723	0.970112	0.987153	0.897723	0.970112	0.987153	0.897723	...	0.862745	0.960784	0.988235	0.862745	0.960784	0.988235	0.862745	0.960784	0.988235	0
1	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0
2	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0
3	0.996079	0.996079	0.996079	0.996079	0.996079	0.996079	0.996375	0.996375	0.996375	0.999777	...	1.000000	1.000000	1.000000	0.999987	0.999987	0.999987	0.999987	0.999987	0.999987	0
4	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000	1.000000	0.998706	1.000000	1.000000	0.999998	1.000000	1.000000	1.000000	0
...
65	0.718931	0.718931	0.718931	0.723172	0.723172	0.723172	0.726831	0.726831	0.726831	0.729648	...	0.343630	0.324487	0.312722	0.340615	0.321446	0.309681	0.338502	0.319273	0.307508	1
66	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1
67	0.492812	0.768377	0.899329	0.496332	0.772480	0.900138	0.498135	0.774474	0.899770	0.501114	...	0.653272	0.589154	0.487160	0.639701	0.585260	0.487088	0.630298	0.572908	0.478461	1
68	0.894118	0.905882	0.941177	0.894308	0.906072	0.941366	0.896911	0.908676	0.943970	0.895299	...	0.145789	0.173240	0.196769	0.143830	0.171281	0.194811	0.137704	0.165155	0.188684	1
69	0.846108	0.846330	0.838914	0.853008	0.853265	0.845724	0.861533	0.862627	0.851928	0.868103	...	0.208152	0.121661	0.044617	0.205749	0.123636	0.054289	0.203111	0.130020	0.081507	1

بعد از `pca`

	pc1	pc2	pc3
0	-24.725225	8.138552	-0.757345
1	50.706895	28.718483	3.879869
2	-73.226021	-2.583709	-1.021038
3	-67.959864	-5.983150	-2.603453
4	33.098721	33.109401	18.385309
5	-73.697208	-20.617357	-4.372538
6	-55.293352	29.985158	-17.195601
7	-83.409254	-9.499236	12.168271
8	30.980490	16.556267	-33.433690
9	43.760191	0.460264	-4.034236
10	-7.783909	14.388837	-43.399891
11	13.738398	-12.000306	12.309821
12	141.069607	-22.733260	-10.759881
13	119.627430	-36.859410	-8.956739
14	-80.668424	-9.293920	7.763743
15	55.323516	-20.688411	5.433466
16	65.326514	-6.624427	-9.334358
17	76.808424	-36.345646	3.995328
18	56.574930	25.500278	22.770714
19	70.952727	30.332671	10.978969
20	3.685747	38.209973	-12.189976
21	72.712317	-9.178934	13.406927
22	55.943213	36.727016	6.964189
23	-50.361374	-7.557148	10.229122
24	63.201480	-18.909222	-12.451694
25	126.920771	-21.669533	-10.274742
26	48.785857	-34.414674	13.627989
27	31.169130	47.968816	25.144734
28	-50.442340	-12.497542	-27.865767
29	-73.539859	-5.042531	-0.213128
30	53.469288	47.420302	25.389884
31	-44.182121	39.977451	-0.614749
32	-79.221916	-10.702802	3.744240
33	-79.105322	-6.319533	9.701426
34	-20.444317	16.605498	-13.320844
35	44.666593	36.478728	20.284355
36	-81.902065	-11.769011	11.452038
37	-76.416865	-11.641818	-0.184064
38	-20.720823	-38.366597	74.227680
39	-78.643777	-9.295092	5.124075
40	-66.510962	-14.038004	-12.236209
41	-42.393746	19.358384	-15.364334
42	77.844087	-9.701413	-13.755780
43	-74.367187	-16.227332	-2.975016
44	64.303430	3.303336	10.306060