به نام خدا

دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده برق و کامپیوتر

درس سیستم‌های هوشمند

تمرین شماره ۱

نام و نام خانوادگی : سیاوش شمس

شماره دانشجویی : ۸۱۰۱۹۷۶۴۴

آبان ۱۴۰۰

فهرست سوالات

سوال ۱

برای محاسبه نقاط کمینه بخش الف ابتدا گرادیان تابع داده شده را حساب میکنیم و سپس با به دست آوردن هسین نوع آن را تعیین میکنیم. در بخش ب دو تکرار روش گرادیان نزولی را به صورت دستی انجام دادیم و سپس این روش را به کمک متلب پیاده سازی کردیم و نتایج را وارد کردیم، در قسمت ج هم روش نیوتن را بررسی میکنیم. در قسمت د فرم ماتریسی قسمت الف را محاسبه کرده و با فرم ماتریسی این بخش مقایسه میکنیم.

الف:

$$\nabla f = \begin{bmatrix} 6x_1 + 6x_2 + 12 \\ 16x_2 + 6x_1 + 8 \end{bmatrix}$$

$$\nabla f = 0 \ \rightarrow \ x_1 = \frac{-12}{5}, \ x_2 = \frac{2}{5}$$

$$H(x_1, x_2) = \begin{bmatrix} 6 & 6 \\ 6 & 16 \end{bmatrix}$$

$$det(H - \lambda I) = 0 \ \rightarrow \begin{vmatrix} 6-\lambda & 6 \\ 6 & 16-\lambda \end{vmatrix} = \lambda^2 - 22\lambda + 60 = 0 \ \rightarrow \lambda_1 = 11 - \sqrt{61}, \ \lambda_2 = 11 + \sqrt{61}$$

هر دو مقادیر ویژه پیدا شده برای تابع H مثبت هستند پس این تابع مثبت معین [1] است. در نتیجه نقطه پیدا شده مینیمم محلی است و چون H همیشه به ازای همه x ها مثبت معین است این نقطه مینیمم سراسری [2] است.

ب:

$$p = -\nabla f_{|x=(1,1)} = \begin{bmatrix} -24 \\ -30 \end{bmatrix}$$

$$\varphi(\alpha) = f(x^k + \alpha p^k) = 13248\alpha^2 - 1476\alpha + 37$$

$$\frac{d\varphi}{d\alpha} = 26496\alpha - 1476 = 0 \ \rightarrow \ \alpha = 0.056$$

تکرار اول:

$$x^2 = x^1 + \alpha^1 p^1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + 0.056 \begin{bmatrix} -24 \\ -30 \end{bmatrix} = \begin{bmatrix} -0.344 \\ -0.68 \end{bmatrix}$$

[1] Positive-definite
[2] Global

تکرار دوم:

$$p = -\nabla f_{|x=(-0.344,-0.68)} = \begin{bmatrix} -5.86 \\ 4.944 \end{bmatrix}$$

$$\varphi(\alpha) = f(x^k + \alpha p^k) = \ 124.73\alpha^2 - 58.76\alpha - 4.11$$

$$\frac{d\varphi}{d\alpha} = 249.46\alpha - 58.76 = 0 \ \rightarrow \ \alpha = 0.235$$

$$x^3 = x^2 + \alpha^2 p^2 = \begin{bmatrix} -0.344 \\ -0.68 \end{bmatrix} + 0.235 \begin{bmatrix} -5.86 \\ 4.944 \end{bmatrix} = \begin{bmatrix} -1.72 \\ 0.48 \end{bmatrix}$$

1.0000	1.0000
-0.3370	-0.6712
-1.8068	0.5047
-2.0401	0.2131
-2.2965	0.4183
-2.3372	0.3674
-2.3819	0.4032
-2.3890	0.3943
-2.3968	0.4006
-2.3981	0.3990
-2.3995	0.4001
-2.3997	0.3998
-2.3999	0.4000
-2.3999	0.4000
-2.4000	0.4000

شکل ۱-۱ – روند تغییرات x در روش Gradient Descent به کمک MATLAB

0
0.0557
0.2470
0.0557
0.2470
0.0557
0.2470
0.0557
0.2470
0.0557
0.2470
0.0557
0.2470
0.0557
0.2470

شکل۲-۱ – روند تغییرات طول گام در روش Line Search به کمک MATLAB

میبینیم که نتایج به دست آمده از شبیه سازی به کمک MATLAB با تکرار های دستی تقریبا یکسان است. و نتیجه نهایی نیز طبق انتظار به دست آمد.

ج:

$$p = -\left(\nabla^2 f_{|x=(1,1)}\right)^{-1}.\nabla f_{|x=(1,1)} = \begin{bmatrix} -3.4 \\ -0.6 \end{bmatrix}$$

$$x^2 = x^1 - p = \begin{bmatrix} -2.4 \\ 0.4 \end{bmatrix}$$

$$p = -\left(\nabla^2 f_{|x=(-2.4,0.4)}\right)^{-1}.\nabla f_{|x=(-2.4,0.4)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$x^3 = x^2 - p = \begin{bmatrix} -2.4 \\ 0.4 \end{bmatrix}$$

بله روش نیوتن برای این سوال کاربرد دارد، همانطور که میبینیم مقادیر بردار x با اولین تکرار به مقادیر بهینه رسیدند، زیرا این تابع دارای گرادیان و هسین پیوسته است همچنین حدس اولیه دور از نقطه بهینه نیست.

د:

$$\nabla f = \frac{1}{2}(A + A^T)x + \begin{bmatrix} 12 \\ 8 \end{bmatrix} = \begin{bmatrix} 6\mathrm{x}_1 + 6x_2 + 12 \\ 16\mathrm{x}_2 + 6x_1 + 8 \end{bmatrix}$$

$$\nabla^2 f = \frac{1}{2}(A + A^T) = \begin{bmatrix} 6 & 6 \\ 6 & 16 \end{bmatrix}$$

$$\nabla \mathrm{f} = 0 \ \rightarrow \ x_1 = \frac{-12}{5}, \ x_2 = \frac{2}{5}$$

فرم ماتریسی الف:

$$f(x_1, x_2) = \frac{1}{2} x^T \begin{bmatrix} 6 & a \\ b & 16 \end{bmatrix} x + \begin{bmatrix} 12 \\ 8 \end{bmatrix}^T x, \qquad a + b = 12$$

از فرم ماتریسی قسمت الف می فهمیم که دو درایه a, b دو درایه کم اهمیت هستند و با انتخاب دلخواه آنها به شرطی که $a + b = 12$ شود، گرادیان و هسین تابع یکی هستند و در نتیجه نقاط کمینه یکسان هستند.

سوال ۲

در بخش اول با روش گردایان نزولی با گام ثابت با نقطه اولیه رفتار تابع را بررسی کردیم، در بخش ب دو روش تحلیلی و آرمیجو را برای تابع بررسی کردیم و از نظر سرعت همگرایی آنها را مقایسه کردیم، سپس در قسمت ج به کمک روش فراابتکاری تبرید شبیه سازی[1] مقدار بهینه تابع را پیدا کردیم

الف:

با نقطه اولیه (5,5) و گام ثابت 0.01 شروع کردیم و در تکرار ۳۵ ام به مقدار کمینه محلی رسیدیم.

1	5	5
2	5.0500	4.8000
3	5.1038	4.8408
4	5.1685	4.8354
5	5.2440	4.8393
6	5.3326	4.8419
7	5.4361	4.8456
8	5.5569	4.8500
9	5.6976	4.8553
10	5.8606	4.8618
11	6.0484	4.8697
12	6.2627	4.8791
13	6.5043	4.8900
14	6.7718	4.9025
15	7.0616	4.9163
16	7.3668	4.9306
17	7.6776	4.9446
18	7.9824	4.9575
19	8.2698	4.9685
20	8.5307	4.9773
21	8.7594	4.9839
22	8.9543	4.9887
23	9.1167	4.9921
24	9.2496	4.9944
25	9.3572	4.9959
26	9.4434	4.9970
27	9.5121	4.9977
28	9.5666	4.9982
29	9.6097	4.9986
30	9.6438	4.9988
31	9.6706	4.9990
32	9.6918	4.9992
33	9.7084	4.9992
34	9.7214	4.9993
35	9.7317	4.9994

1	25
2	23.1486
3	22.7573
4	22.3027
5	21.6818
6	20.8310
7	19.6682
8	18.0854
9	15.9450
10	13.0781
11	9.2922
12	4.3916
13	-1.7791
14	-9.2673
15	-17.9305
16	-27.3792
17	-36.9962
18	-46.0675
19	-53.9809
20	-60.3891
21	-65.2475
22	-68.7358
23	-71.1361
24	-72.7363
25	-73.7789
26	-74.4472
27	-74.8708
28	-75.1371
29	-75.3036
30	-75.4072
31	-75.4716
32	-75.5115
33	-75.5361
34	-75.5514
35	-75.5608

شکل ۱-۲ – روند تغییرات بردار x در جدول راست و مقدار تابع در جدول چپ

[1] **Simulated Annealing**

با نقطه اولیه (5-,5-) و گام ثابت 0.01 شروع کردیم و در تکرار ۳۴ ام به مقدار کمینه محلی رسیدیم.

1	-5	-5
2	-4.7500	-5
3	-4.5541	-4.9988
4	-4.3999	-4.9936
5	-4.2774	-4.9800
6	-4.1792	-4.9499
7	-4.0993	-4.8884
8	-4.0334	-4.7671
9	-3.9801	-4.5344
10	-3.9458	-4.1096
11	-3.9601	-3.4372
12	-4.0697	-2.7583
13	-4.2261	-2.5419
14	-4.3660	-2.5375
15	-4.4904	-2.5402
16	-4.6006	-2.5416
17	-4.6980	-2.5423
18	-4.7840	-2.5426
19	-4.8597	-2.5426
20	-4.9264	-2.5423
21	-4.9850	-2.5420
22	-5.0366	-2.5416
23	-5.0820	-2.5411
24	-5.1219	-2.5407
25	-5.1571	-2.5402
26	-5.1880	-2.5398
27	-5.2153	-2.5394
28	-5.2393	-2.5390
29	-5.2604	-2.5387
30	-5.2790	-2.5384
31	-5.2955	-2.5381
32	-5.3099	-2.5378
33	-5.3227	-2.5376
34	-5.3340	-2.5374

1	75
2	69.4281
3	66.0010
4	63.8634
5	62.4827
6	61.4700
7	60.3238
8	57.7750
9	49.8269
10	25.8992
11	-22.5493
12	-56.6899
13	-61.4549
14	-63.3045
15	-64.7642
16	-65.9093
17	-66.8029
18	-67.4978
19	-68.0369
20	-68.4545
21	-68.7778
22	-69.0280
23	-69.2217
24	-69.3717
25	-69.4878
26	-69.5778
27	-69.6476
28	-69.7018
29	-69.7439
30	-69.7765
31	-69.8019
32	-69.8217
33	-69.8370

شکل ۲-۲ – روند تغییرات بردارx در جدول راست و مقدار تابع در جدول چپ

تا وقتی که طول گام به اندازه ای نباشد تا باعث شود از روی مینیمم محلی پرش داشته باشیم می توان آن را انتخاب کرد، در غیر اینصورت این کار منجر به واگرا شدن الگوریتم می شود، در این مورد با انجام آزمایش طول گام باید کمتر از 0.11 باشد تا الگوریتم همگرا شود. همچنین مقدار دقیق طول گام را میتوان با استفاده از قانون پیوستگی گرادیان لیپشیتس[1] به دست آورد.

[1] Lipschitz Continuous Gradient

ب:

1	0	0
2	1.0207	0.6804
3	0.5724	1.3528
4	0.3215	1.1855
5	0.3860	1.0888

1	0
2	-13.9936
3	-9.3986
4	-10.4956
5	-10.3997

شکل ۳-۲ – روند تغییرات بردارx در جدول چپ و مقدار تابع در جدول راست با روش تحلیلی برای به دست آوردن طول گام

1	0
2	0.0680
3	-0.0855
4	0.0413
5	-0.0699

شکل ۴-۲ – روند تغییرات طول گام با روش تحلیلی

1	0	0
2	0.6000	0.4000
3	1.0408	0.5220
4	1.7744	0.0456
5	3.2231	0.3565
6	3.3018	-0.1319
7	3.5268	0.0121
8	3.7834	-0.2172
9	3.8899	-0.0478
10	4.2604	-0.2067
11	4.3948	0.0413
12	4.4339	-0.0464
13	4.6554	-0.2037
14	4.6924	-0.1167

1	-11.0859
2	-15.1591
3	-23.6226
4	-31.4949
5	-38.5656
6	-40.3850
7	-41.8559
8	-43.4759
9	-45.4165
10	-46.1344
11	-47.1688
12	-47.8300
13	-48.7435
14	-49.3453

شکل ۵-۲ – روند تغییرات بردارx در جدول چپ و مقدار تابع در جدول راست با روش آرمیجو برای به دست آوردن طول گام

1	0.0400
2	0.0400
3	0.0939
4	0.1297
5	0.0155
6	0.0252
7	0.0325
8	0.0136
9	0.0502
10	0.0205
11	0.0064
12	0.0357
13	0.0068
14	0.0430

شکل ۶-۲ – روند تغییرات طول گام با روش آرمیجو [1]

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
18.0278	11.4335	9.3130	11.4240	31.8248	10.5858	10.5751	14.6987	8.0366	13.7616	14.9589	7.6158	13.9807	6.5194	14.1705	6.3032	16.3837	16.0

43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
3.3561	1.6465	1.6339	5.2610	1.8266	1.5353	1.9498	2.1954	1.3312	1.8910	2.8403	1.9922	1.2120	1.7104	2.6161	1.8680	1.0644	1.3490

error function with
backtracking line search

	1	2	3	4
1	18.0278	9.4531	7.2932	1.6645

error function with exact
line search method

شکل ۷-۲ – مقایسه تابع خطای دو روش شبیه سازی شده

با مقایسه تابع خطای دو روش و رابطه زیر میبینیم نرخ همگرایی در روش تحلیلی به دست آوردن طول گام در این مسئله بهتر است و سریعتر به جواب می رسد، اما با مقایسه طول گام در دو روش متوجه می شویم روش آرمیجو پایداری بیشتری دارد و احتمال واگرایی کمتر است.

$$\frac{e(x_{k+1})}{e(x_k)}$$

توضیحات روش آرمیجو: در این روش از مقدار $c = 0.01$ و $beta = 0.2$ استفاده کردیم، و برای محاسبه طول گام اولیه بعد از تکرار دوم از رابطه زیر استفاده کردیم.

[1] Armijo

$$\alpha^0 = \frac{2(f_k - f_{k-1})}{\nabla f_k^T p_k}$$

این روش در نهایت با ۶۰ گام به مقدار کمینه محلی 56- در نقطه (6.95,-0.015) همگرا شد.

ج:

```
Mimimum found for the function equals -75.543108
Vector that results in minimum is 9.732527 5.011950
```

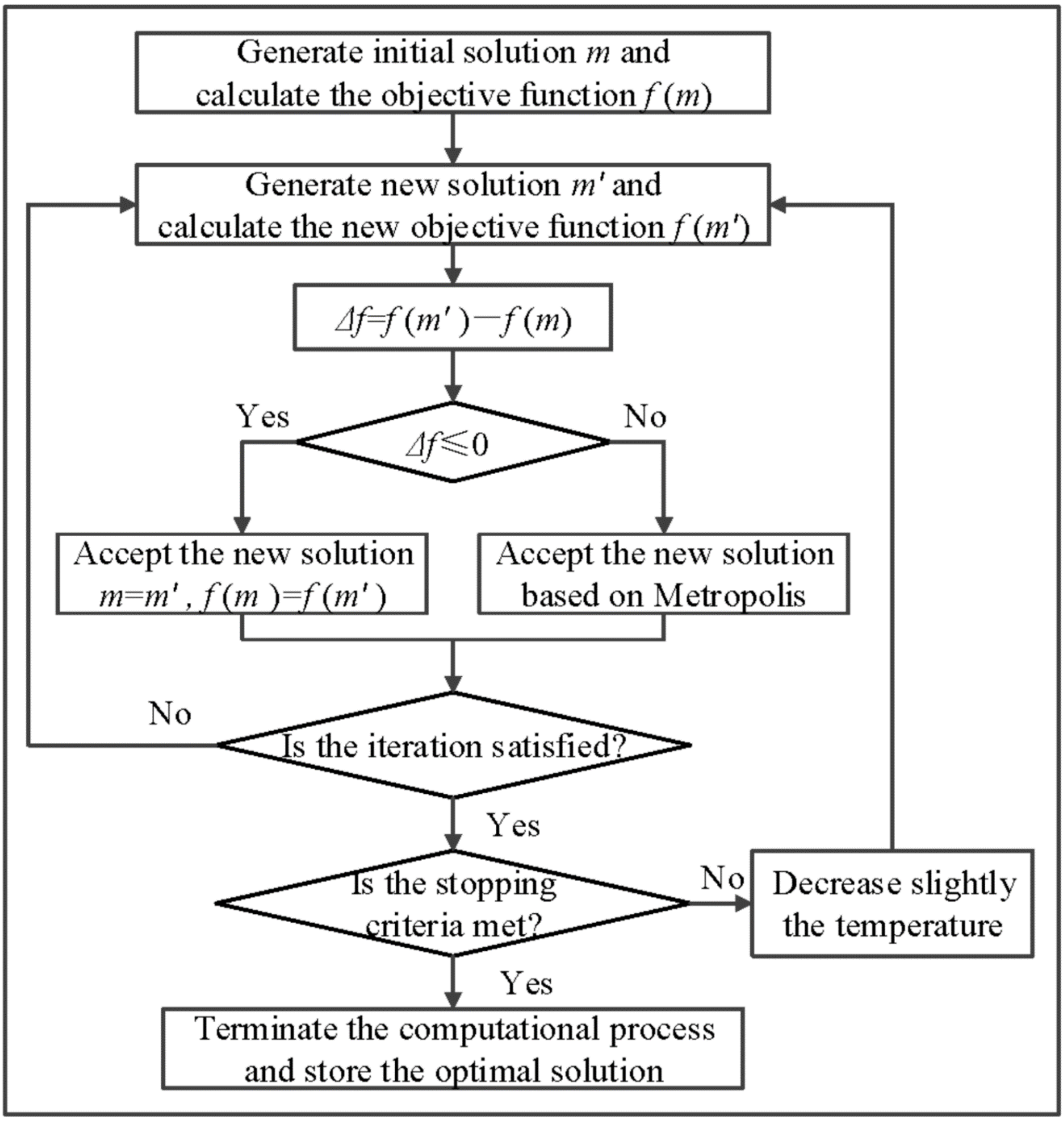
شکل ۸-۲ – خروجی پیاده سازی الگوریتم تبرید شبیه سازی [1]

پارامتر های انتخاب شده در الگوریتم برابر $\mathrm{maxtemps} = 1000$ و نقطه اولیه $x = (0,0)$ می باشد، همچنین برای به روزرسانی بردار x در هر مرحله، عددی تصادفی گوسی با میانگین x قبلی و واریانس T با آن جمع کردیم. دلیل انتخاب $maxtemps\ =\ 1000$ در الگوریتم افزایش اطمینان در پیدا کردن مقدار بهینه است، و انتخاب نقطه اولیه به صورت دلخواه انجام شده است.

a	[44.4327,49.8459]
del	4.5748e+03
f	@(x)(x(1)^2-10*x(2)*...
f0	-75.5431
fx	5.3296
fxx	4.5801e+03
iter	100
k	1000
maxtemps	100
T	99
x	[-3.2961,3.2066]
x0	[9.7325,5.0119]
xx	[41.1366,53.0525]

شکل ۹-۲ – متغیر های خروجی بعد از شبیه سازی

[1] **Simulated Annealing**



شکل ۱۰-۲ – الگوریتم استفاده شده در قسمت ج

سوال ۳

در قسمت الف با نوشتن روابط، معادله خط جدا کننده را می یابیم، در قسمت ب برای روش های بدون استفاده از کتابخانه sklearn داده های با برچسب 0 در مقابل سایر داده ها با برچسب 1 و 2 طبقه بندی شد. اما در قسمت استفاده از کتابخانه sklearn هر ۳ نوع داده را طبقه بندی کردیم. به همین دلیل دقت به دست آمده در بخش دوم کمتر است چون ۳ نوع طبقه بندی داریم.

الف:

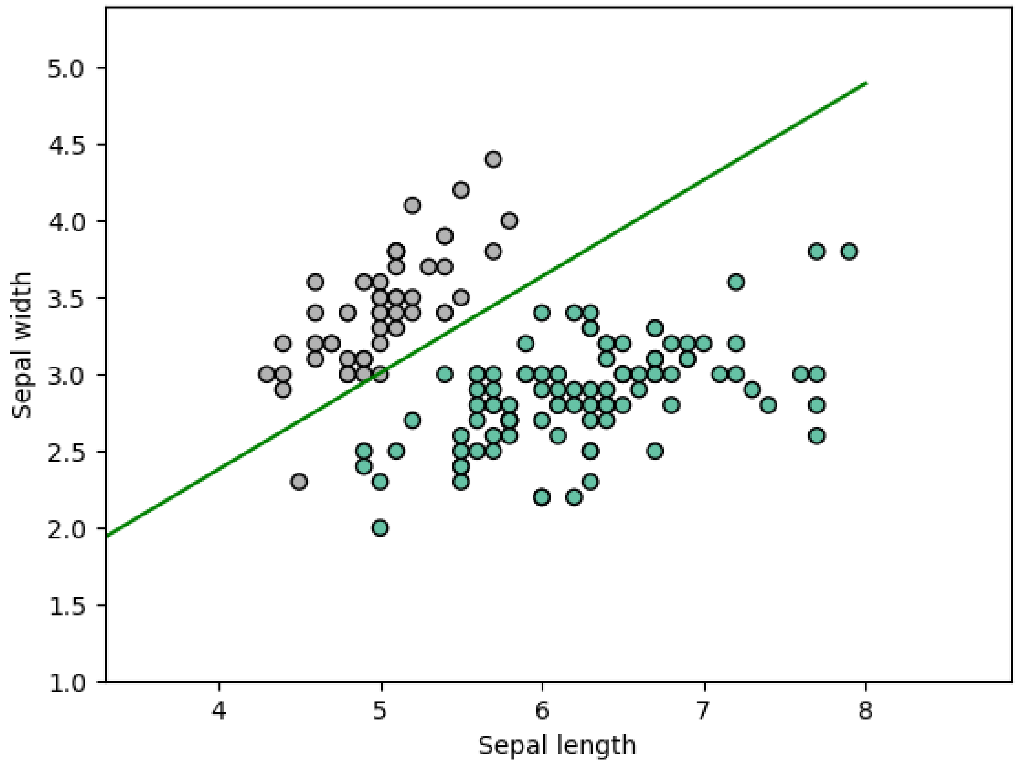
$$\widetilde{S_1} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}, \widetilde{S_2} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \widetilde{S_3} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

$$\alpha_1 \widetilde{S_1}.\widetilde{S_1} + \alpha_2 \widetilde{S_2}.\widetilde{S_1} + \alpha_3 \widetilde{S_3}.\widetilde{S_1} = -1$$
$$\alpha_1 \widetilde{S_1}.\widetilde{S_2} + \alpha_2 \widetilde{S_2}.\widetilde{S_2} + \alpha_3 \widetilde{S_3}.\widetilde{S_2} = 1$$
$$\alpha_1 \widetilde{S_1}.\widetilde{S_3} + \alpha_2 \widetilde{S_2}.\widetilde{S_3} + \alpha_3 \widetilde{S_3}.\widetilde{S_3} = 1$$

$$\begin{cases} 2\alpha_1 + \alpha_2 = -1 \\ \alpha_1 + 2\alpha_2 + \alpha_3 = 1 \\ \alpha_2 + 2\alpha_3 = 1 \end{cases}$$
$$\rightarrow \alpha_1 = -1, \alpha_2 = 1, \alpha_3 = 0$$

$$\widetilde{W} = \sum_i \alpha_i . \tilde{S}_i = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \rightarrow w = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, b = 0$$
$$y = \vec{w}x + b \rightarrow y = -x$$

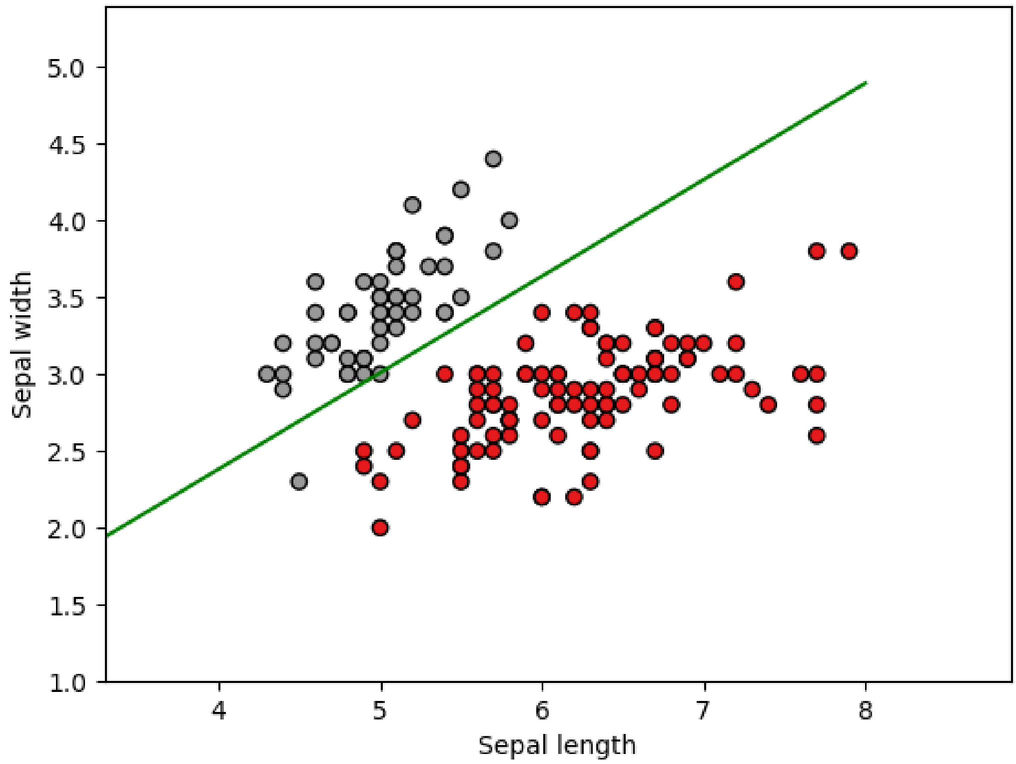
ب:



شکل ۱-۳ داده های مرتبط با دسته اول به رنگ خاکستری و بقیه داده ها به رنگ سبز و خط جدا کننده SVM به رنگ سبز روش Stochastic Gradient Descent

```
predicted   0  1
actual
0          22  0
1           0  8
predicted    0    1
actual
0          1.0  0.0
1          0.0  1.0
Accuracy is 100.0
```

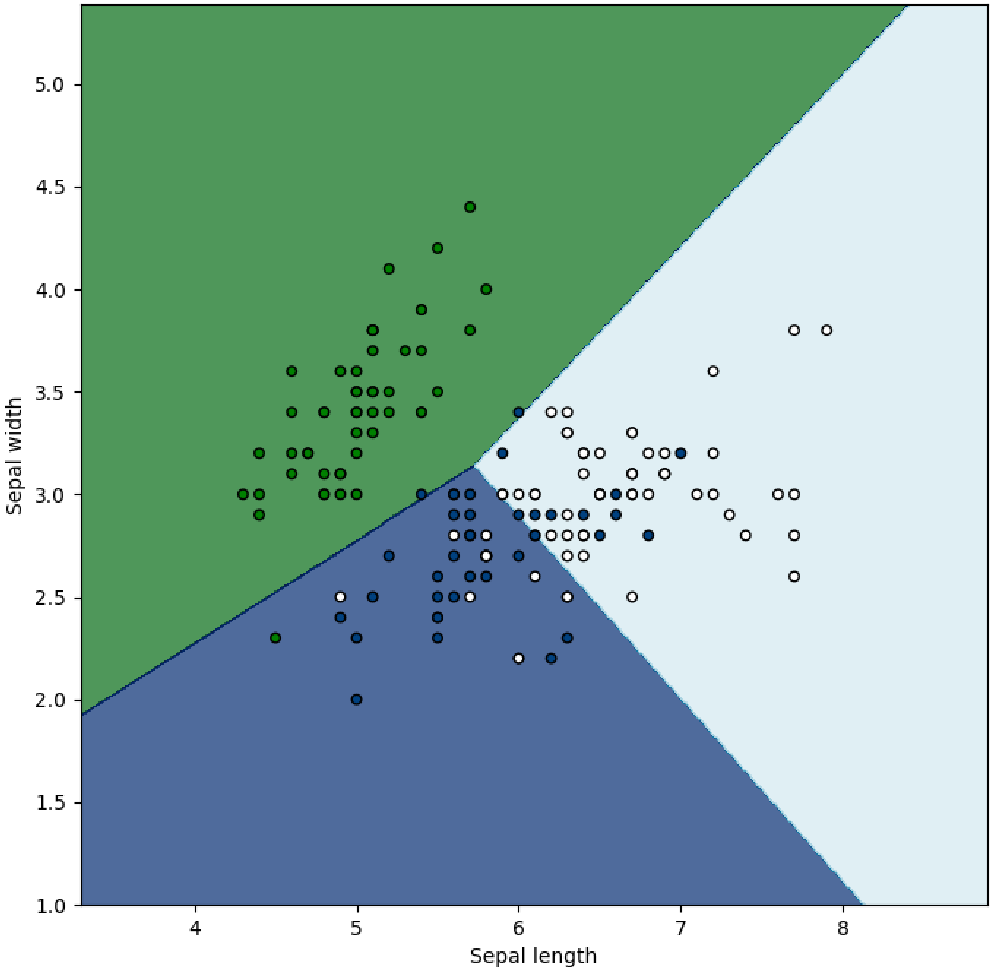
شکل ۲-۳ ماتریس آشفتگی و اطمینان و دقت آموزش روش Stochastic Gradient Descent



شکل ۳-۳ داده های مرتبط با دسته اول به رنگ خاکستری و بقیه داده ها به رنگ قرمز و خط جدا کننده SVM به رنگ سبز روش Gradient Descent با Armijo

```
predicted   0  1
actual
0          20  0
1           1  9
predicted     0    1
actual
0          1.00  0.0
1          0.05  0.9
Accuracy is  96.66666666666667
```

شکل ۳-۴ ماتریس آشفتگی و اطمینان و دقت آموزش روش Gradient Descent با Armijo



شکل ۵-۳- مناطق مشخص شده در طبقه بندی با استفاده از SVM

```
Confusion Matrix
 [[16  0  0]
 [ 1  9  8]
 [ 0  3  8]]
Confidence Matrix
 [[1.         0.         0.        ]
 [0.05555556 0.5        0.44444444]
 [0.         0.27272727 0.72727273]]
Accuracy: 81 %
```

شکل ۶-۳- ماتریس آشفتگی و اطمینان و دقت آموزش

در موارد بالا ماتریس اطمینان همان نرمالایز شده ماتریس آشفتگی می باشد و دقت با استفاده از ماتریس آشفتگی به صورت جمع مولفه های قطر اصلی تقسیم بر جمع کل درایه های ماتریس به دست می آید، همچنین از نتایج بالا میفهمیم که روش Stochastic Gradient Descent با تعداد تکرار کمتر دقت بهتری به ما می دهد و همچنین با داده های کمتری می توان از این روش برای آموزش استفاده کرد.