به نام خالق ستارگان

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

تمرین دوم درس جداسازی کور منابع (BSS)

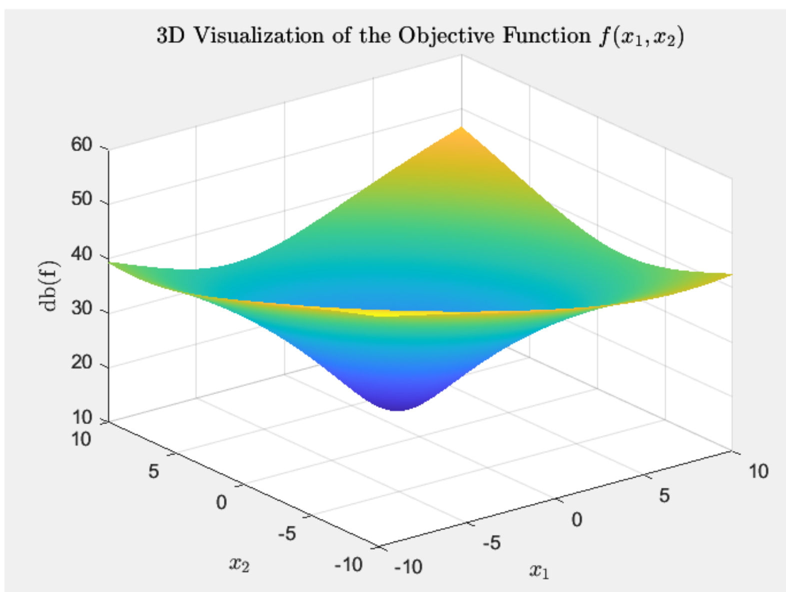
دکتر سعید اخوان

نام و نام خانوادگی	سیده غزل موسوی
شماره دانشجویی	۸۱۰۱۰۰۲۵۹
مهلت ارسال پاسخ	**۱۴۰۳.۱۲.۱۵**

سوال ۱.

$$f(\underline{x}) = x_1{}^2 + x_2{}^2 - 4x_1 - 6x_2 + 13 + x_1x_2$$
$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

ابتدا با استفاده از دستور meshgrid دومتغیر x_1, x_2 در بازه $[-10, 10]$ با گام‌های 0.01 تعریف کرده و ضابطه f را برحسب دومتغیر بیان کرده و درنهایت با استفاده از دستور mesh تابع هدف f به صورت db نسبت به متغیرهای x_2 و x_1 رسم شده است.

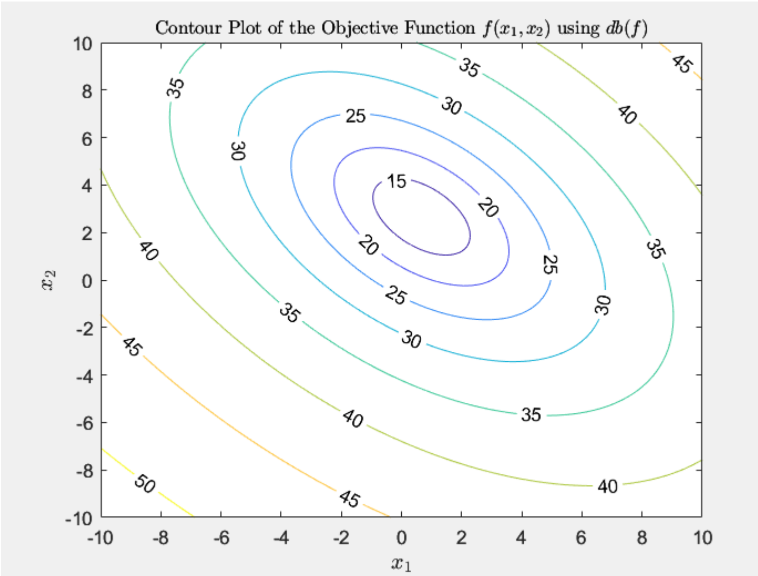


شکل ۱. نمودار سه بعدی تابع هدف برحسب x_1 و x_2

همانطور که مشاهده می‌شود، تابع هدف دارای یک مینیمم است یعنی convex است و شکل برحسب دو متغیر یک سهمی گون است.

سوال ۲.

با استفاده از دستور contour و فعال کردن ShowText سطوح هم پتانسیل که در واقع $f(\underline{x}) = c$ را در صفحه x_2 و x_1 رسم شده است.لازم به ذکر است که دراین بخش نیز تابع هدف به صورت db در نظر گرفته شده‌است.



شکل ۲. نمودار contour تابع هدف

سوال ۳ و ۴.

بردار گرادیان

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2x_1 + x_2 - 4 \\ 2x_2 + x_1 - 6 \end{bmatrix}$$

ماتریس هسین

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial {x_1}^2} & \frac{\partial^2 f}{\partial x_1 x_2} \\ \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial {x_2}^2} \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

اثبات convex بودن تابع هدف:

$$x^T H x = [x_1 \quad x_2] \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = [2x_1 + x_2 \quad x_1 + 2x_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 2{x_1}^2 + x_1x_2 + x_1x_2 + 2{x_2}^2$$
$$= 2({x_1}^2 + x_1x_2 + {x_2}^2) = {x_1}^2 + (x_1 + x_2)^2 + {x_2}^2$$

برای تمامی مقادیر x عبارت بالا همواره مثبت است در نتیجه یک مینیمم دارد و convex است.

در متلب نیز متغیرهای x_2 و x_1 به صورت syms تعریف کرده و با استفاده از دستورات gradient و hessian بردار گرادیان و ماتریس هسین را به دست‌آورده و به نتایج مشابه تئوری می‌رسیم.

```
g =

2*x1 + x2 - 4
x1 + 2*x2 - 6
```

```
h =

     2     1
     1     2
```

$$\nabla f = \begin{bmatrix} \dfrac{\partial f}{\partial x_1} \\ \dfrac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2x_1 + x_2 - 4 \\ 2x_2 + x_1 - 6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow \begin{matrix} 2x_1 + x_2 - 4 = 0 \\ 2x_2 + x_1 - 6 = 0 \end{matrix} \rightarrow \begin{bmatrix} 2/3 \\ 8/3 \end{bmatrix} \rightarrow f_{min} = \frac{11}{3}$$

```
Minimum of cost function f: 3.666667 at point = (0.666667, 2.666667)
```

شکل ۳. نتیجه صفر قرار دادن گرادیان‌ها و به دست آوردن مینیمم تابع هدف

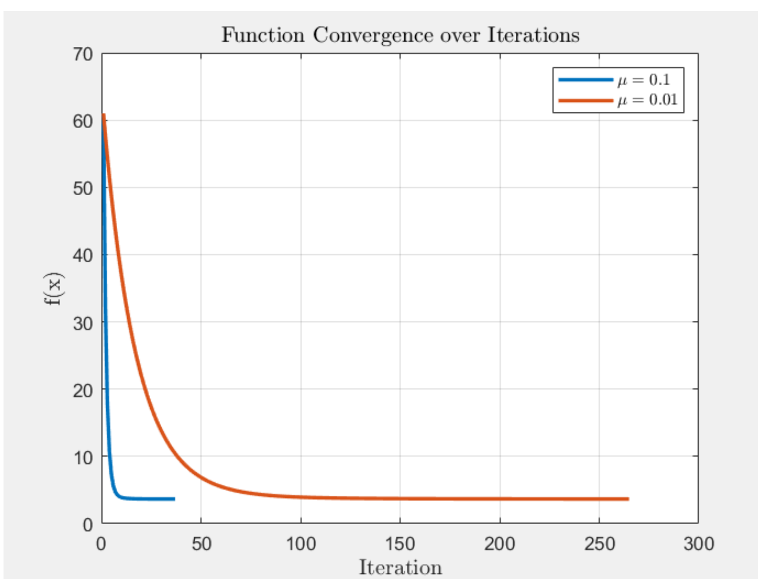
سوال ۵.

در این بخش با نوشتن یک حلقه روش Steepest descend را اجرا کردیم و شرط توقف ان حلقه زمانی است که مقدار تابع هدف در نقطه فعلی با نقطه قبلی به اندازه 0.0001 اختلاف داشته باشد.

همانطور که در نتایج زیر مشاهده می‌شود، هردو الگوریتم همگرا شدند ولی الگوریتم با $\mu = 0.1$ نسبت به $\mu = 0.01$ سریع‌تر همگرا شد زیرا گام‌های تغییر با μ بیشتر، بزرگتر است و سریع‌تر به نقطه مینیمم می‌رسیم ولی هرچه μ کوچک‌تر باشد گام‌های تغییر کوچک تر بوده و دیرتر همگرا می‌شود. لازم به ذکر است که μ نمی‌تواند خیلی هم بزرگ باشد زیرا باعث می‌شود از نقطه مورد نظر رد شود و نتواند همگرا شود.

```
=========================================================================================
Steepest Descent converged with mu = 0.10 after 37 iterations.
Minimum of cost function f with Steepest Descend (mu = 0.10): 3.667078 at point = (0.6870, 2.6464)
=========================================================================================
Steepest Descent converged with mu = 0.01 after 265 iterations.
Minimum of cost function f with Steepest Descend (mu = 0.01): 3.671532 at point = (0.7377, 2.5983)
=========================================================================================
```

شکل ۴. مقدار بهینه برای روش Steepest descend با $\mu = 0.01$ و $\mu = 0.1$



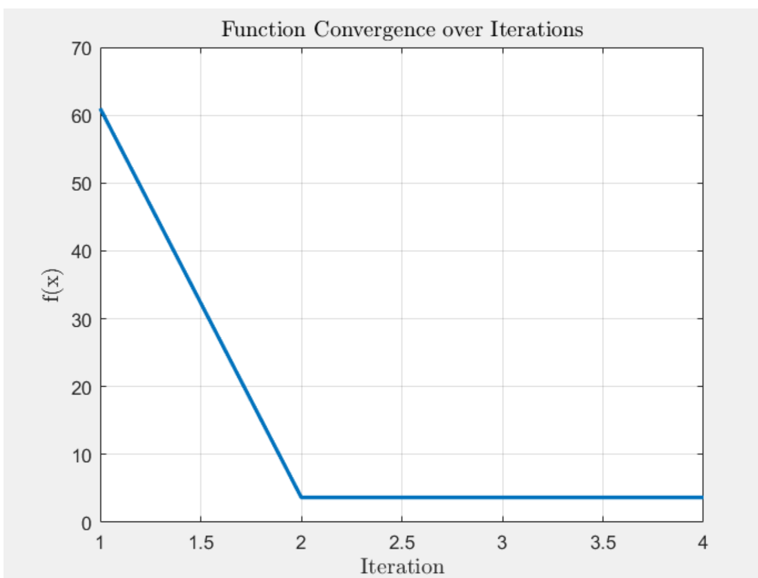
شکل ۵. نمودار همگرایی برحسب تعداد iteration برای الگوریتم steepest descend

سوال ۶.

در الگوریتم Newton در هر iteration یک سهمی نسبت به آن نقطه فعلی فیت می‌کنیم و سپس تصویر مینیمم آن سهمی فیت‌شده را روی تابع هدف به عنوان نقطه جدید درنظر می‌گیریم. در اینجا چون تابع هدف ما نسبت به دومتغیر سهمی است، سهمی ای که در نقطه اولیه فیت می‌کنیم روی خود تابع هدف می‌افتد و در نتیجه مینیمم سهمی فیت شده همان مینیمم تابع هدف است و به همین دلیل بعد از ۲ iteration الگوریتم همگرا می‌شود و مشاهده می‌کنیم که سرعت این الگوریتم نسبت به الگوریتم قبلی بیشتر است.

```
================================================================================
Newton converged after 2 iterations.
Minimum of cost function f with Newton: 3.666667 at point = (0.6667, 2.6667)
================================================================================
```

شکل ۶. مقدار بهینه با روش Newton



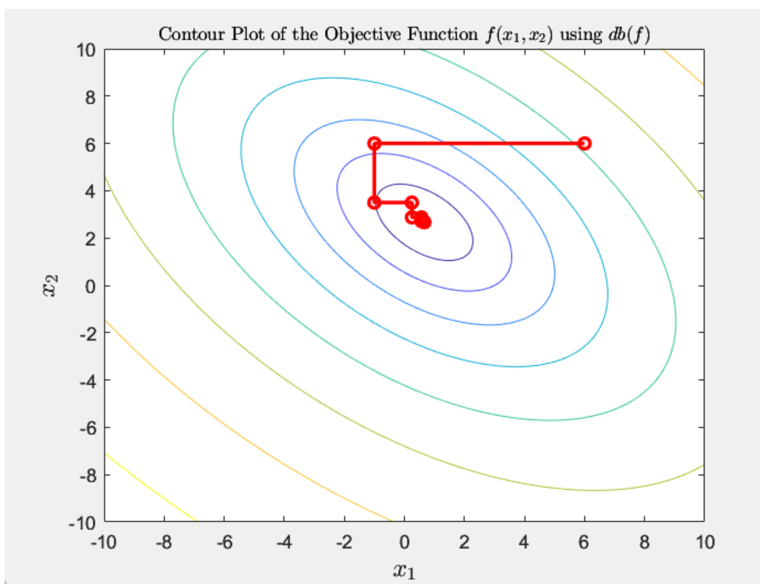
شکل ۷. نمودار همگرایی برحسب iteration برای الگوریتم Newton

سوال ۷.

در این بخش از الگوریتم alternation minimization استفاده شده است. تابع هدف برحسب دومتغیر x_2 و x_1 نوشته شده است و در مرحله یک متغیر را ثابت درنظر می‌گیریم و متغیر دیگری را به‌گونه ای محاسبه می‌کنیم که گرادیان آن نسبت به همان متغیر صفر شود. یعنی در یک مرحله x_1 ثابت درنظر گرفته می‌شود و x_2 از رابطه $x_2 = \frac{6-x_1}{2}$ به دست می‌آید و بار دیگر x_2 ثابت درنظر گرفته می‌شود و x_1 از رابطه $x_1 = \frac{4-x_2}{2}$ به دست می‌آید.

```
=========================================================================================
Alternation Minimization converged after 11 iterations.
Minimum of cost function f with Alternation Minimization: 3.666675 at point = (0.6650, 2.6699)
=========================================================================================
```

شکل ۸. مقدار بهینه با روش alternation minimization



شکل ۹. نمودار contuor و نمایش گام‌ها الگوریتم alternation minimization

در شکل ۵ می‌توان مشاهده کرد که به جای اینکه به صورت یک خط صاف به سمت نقطه مینیمم که مرکز بیضی است حرکت کنیم به صورت پله‌ای حرکت کردیم زیرا درهرگام یک متغیر را ثابت درنظر گرفته و تلاش کردیم دیگری را مینیمم کنیم.

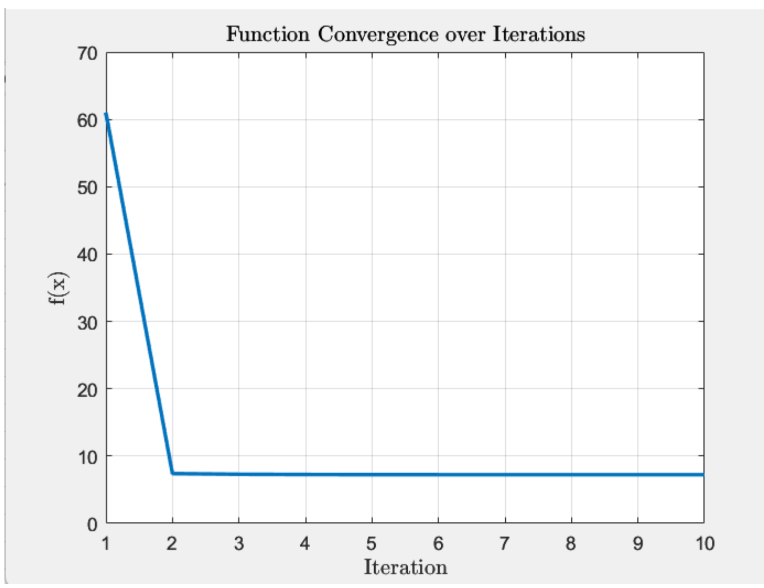
سوال ۸.

با توجه به قید ذکر شده می‌توانیم درهرمرحله بدون توجه به قید با یکی از روش‌های بهینه‌سازی جلو برویم و سپس برای شروع مرحله جدید تصویر نقطه جدید را روی فضای قید انتخاب کنیم.

یعنی در این سوال درهرمرحله با روش steepest descend جلو می‌رویم و برای شروع مرحله جدید بردار x به‌دست‌آمده را نرمالایز می‌کنیم.

```
==========================================================================================
Gradient Projection converged after 10 iterations.
Minimum of cost function f with Gradient Projection: 7.236774 at point = (0.4972, 0.8676)
==========================================================================================
```

شکل ۱۰. مقدار بهینه با روش gradient projection



شکل ۱۱ نمودار همگرایی روش gradient projection

سوال۹.

در این بخش تابع هدف $f(x,x_2)$ و قید مسئله $g(x,x_2)=k$ است.

$$\nabla f(x,x_2) = \lambda \nabla g(x,x_2)$$

$$g(x,x_2)=k \rightarrow g(x,x_2) == x_1{}^2 + x_2{}^2 = 1$$

روابط بالا را به صورت ساده‌تر بازنویسی می‌کنیم:

$$f_{x_1} = \lambda g_{x_1} \qquad f_{x_2} = \lambda g_{x_2}$$

$$f_{x_1} = 2x_1 + x_2 - 4 = \lambda g_{x_1} = \lambda(2x_1)$$
$$f_{x_2} = 2x_2 + x_1 - 6 = \lambda g_{x_2} = \lambda(2x_2)$$

با ساده‌سازی دو متغیر x_2 و x_1 را برحسب λ نوشته و درقید مسئله قرار می‌دهیم و مقادیر آن را به دست می‌آوریم:

$$\begin{cases} x_1 = \dfrac{-2(4\lambda-1)}{3-8\lambda+4\lambda^2} \\ x_2 = \dfrac{-12\lambda+8}{3-8\lambda+4\lambda^2} \end{cases} \overset{\text{قید}}{\Rightarrow} \left(\frac{-2(4\lambda-1)}{3-8\lambda+4\lambda^2}\right)^2 + \left(\frac{-12\lambda+8}{3-8\lambda+4\lambda^2}\right)^2 = 1$$

$$\Rightarrow \lambda = 5.08 \rightarrow \begin{cases} x_1 = -0.59 \\ x_2 = -0.81 \end{cases} \rightarrow f(-0.59,-0.81) = 21.70 \quad ✖$$

or

$$\lambda = -2.16 \rightarrow \begin{cases} x_1 = 0.49 \\ x_2 = 0.87 \end{cases} \rightarrow f(0.49,0.87) = 7.24 \quad ✔$$

و از آنجایی که ما به دنبال مینمم تابع بودیم جواب دوم صحیح است و نتیجه به دست‌آمده با جواب به دست آمده در سوال۸ تا حد خوبی یکسان است.