

# تکلیف سری اول

## درس روشهای عددی بهینه‌سازی

مسعود بابایی زاده

### ۱ هدف

در این تکلیف هدف آن است که روش Golden Section Search (GSS) برای مینیمم کردن یک تابع یک متغیره در MATLAB پیاده‌سازی گردد. این برنامه بعداً (در تکالیف بعدی) برای حداقل کردن توابع چندمتغیره نیز مورد استفاده قرار خواهد گرفت، زیرا خواهیم دید که بسیاری از روشهای تکراری (iterative) برای حداقل کردن توابع چندمتغیره، در هر تکرار نیاز دارند که یک تابع یک متغیره حداقل شود.

برنامه نهایی باید به صورت یک m فایل بنام GSS.m نوشته شود. در واقع یکی از آرگومانهایی که این تابع می‌گیرد باید نام تابعی باشد که قرار است حداقل شود (اینکار را می‌توان با استفاده از handle به توابع انجام داد که در قسمت بعد می‌توانید توضیحاتی در مورد آن مشاهده کنید). به این ترتیب می‌توان از این m فایل برای حداقل کردن هر تابعی استفاده کرد.

به طور خلاصه، ورودی و خروجی‌های این تابع به شرح زیر هستند:

$[x\_min, N] = GSS(f, a, b, \epsilon, \text{varargin})$

که در آن:

- $f$  یک هندل به تابعی است که باید حداقل شود.
- $a$  ابتدا و  $b$  انتهای بازه‌ای است که قرار است برای حداقل کردن تابع  $f$  مورد جستجو قرار گیرد.
- $\epsilon$  تolerانس مورد قبول در جواب نهایی است. به این ترتیب که هرگاه هنگام جستجو الگوریتم به این نقطه رسید که حداقل کننده تابع در یک فاصله  $[a_i, b_i]$  قرار دارد بطوریکه  $(b_i - a_i) < \epsilon$  است، تابع GSS مقدار  $\frac{a_i + b_i}{2}$  را به عنوان جواب بر می‌گرداند.
- $\text{varargin}$  بقیه پارامترهای احتمالی‌ای است که تابع  $f$  به آنها نیاز دارد. این را به این دلیل می‌گذاریم که ممکن است شخصی بخواهد از GSS برای حداقل کردن یک تابع یک متغیره‌ای استفاده کند که غیر از متغیر  $(x)$ ، پارامترهای دیگری نیز می‌گیرد. به عبارتی درون تابع GSS، هرگاه به مقدار تابع  $f$  در نقطه  $x$  نیاز بود، این مقدار با استفاده از دستور  $f(x, \text{varargin})$  محاسبه می‌شود.
- $x\_min$  حداقل کننده‌ای که الگوریتم پیدا کرده است (خروجی).
- $N$  تعداد تکراری که الگوریتم برای محاسبه این حداقل کننده و با این تolerانس و با این مقادیر اولیه مجبور به استفاده شده است (خروجی).

### مثالها

- برای حداقل کردن تابع  $\sin(x)$  در فاصله  $[0, 3]$  رادیان و با تolerانس  $0.01$  می‌نویسیم:  
 $GSS(@sin, 0, 3, 0.01)$

- فرض کنید یک تابع  $f(x, n)$  در MATLAB نوشته‌ایم که مقدار  $x^n$  را حساب می‌کند و  $n$  را هم به عنوان پارامتر از ورودی می‌گیرد. در اینصورت برای حداقل کردن تابع  $x^2$  در فاصله  $[-1, 1]$  و با تکران ۱٪ می‌نویسیم:

```
GSS(@f, -1, 1, 0.01, 2)
```

و برای حداقل کردن تابع  $x^4$  در فاصله  $[-1, 1]$  و با تکران ۱٪ می‌نویسیم:

```
GSS(@f, -1, 1, 0.01, 4)
```

## ۲ توضیحاتی در مورد function ها در MATLAB

### ۱.۲ handle به یک function

یک روش در MATLAB برای ارسال نام یک تابع به عنوان پارامتر به تابع دیگر استفاده از handle به توابع است. یک handle به یک تابع با استفاده از @ به صورت زیر بدست می‌آید:

```
fhandle = @FunctionName
```

در اینصورت صدا زدن این تابع می‌تواند بجای نام اصلی با این handle صورت گیرد. مثال. فرض کنید تابعی به صورت زیر داریم:

```
function x=MyPlot(fhandle, data)
plot(data, fhandle(data))
```

مثلاً با استفاده از این تابع می‌توان تابع  $\sin$  را در فاصله  $(-\pi, \pi)$  با دستور زیر رسم کرد:

```
MyPlot(@sin, -pi:0.01:pi)
```

توجه ۱. بجای استفاده از handle ها، می‌توانستیم MyPlot را طوری بنویسیم که نام تابع با استفاده از استرینگها به آن ارسال شود (`MyPlot('sin', -pi:0.01:pi)`). می‌توان یک string و handle را با استفاده از دستورات `func2str` و `str2func` به یکدیگر تبدیل کرد.

توجه ۲. با استفاده از handle ها می‌توان بدون نوشتن M-فایل در MATLAB تابع تعریف کرد (Anonymous Functions). مثلاً:

```
sqr = @(x) x.^2;
```

توجه ۳. برای توضیحات بیشتر به راهنمای MATLAB مراجعه کنید:

Help → Contents → MATLAB → Programming → Function Handles

### ۲.۲ تعداد آرگومان متغیر در توابع

در MATLAB می‌توان تعداد ورودی یا خروجی متغیر داشت. اینکار با استفاده از `varargin` و `varargout` صورت می‌گیرد. در صورتی که آرگومانهای عادی با `varargin` و `varargout` مخلوط شوند، باید حتماً `varargin` و `varargout` در آخر لیست آرگومانها ظاهر شوند.

برای توضیحات بیشتر به راهنمای MATLAB مراجعه کنید:

Help → Contents → MATLAB → Programming → Function Arguments