توابع متلب سایت سیم پاور

متلب سایت سیم پاور اساساً دارای دو نوع تابع است: توابع عددی و توابع نمادین.

- توابع عددی در حقیقت یک برنامه کوتاه هستند که بر روی اعداد عمل کرده و عدد دیگری را به وجود می آورند.
- توابع نمادین بر روی متغیرهای نمادین عمل کرده و نتایج نمادین می‌دهند. این توابع نمادین با عملیات-هایی چون دیفرانسیل‌گیری و انتگرال‌گیری به خوبی، سازگار می‌باشند که در قسمت بعد درباره آن‌ها صحبت خواهد شد.

متلب سایت سیم پاور دارای توابع مفیدی مانند: $\sin(x)$ ، $\cos(x)$ ، $\tan(x)$ ، $\exp(x)$ ، $\log(x)$ و $\sqrt{x}$ است. این توابع می‌توانند بر روی ماتریس‌ها نیز به صورت آرایه ای عمل کنند . به همین دلیل به این توابع array- smartگفته می‌شود. برای مثال تابع کسینوس می‌تواند بر روی یک ماتریس مانند T اعمال گردد:

```
>> T= [۲ ۳ pi; ۸  pi/۲  ۱];
>> cos(T)
ans =
     −۰.٤١٦١   −۰.۹۹۰۰   −۱.۰۰۰۰
     −۰.۱٤٥٥    ۰.۰۰۰۰    ۰.٥٤۰۳
>>  sqrt(T)
ans  =
    ۱.٤١٤۲   ۱.۷۳۲۱   ۱.۷۷۲٥
    ۲.۸۲۸٤   ۱.۲٥۳۳   ۱.۰۰۰۰
```

به علاوه توابع ویژه دیگری نیز در دسترس می‌باشند. تابع خطا که توسط فرمان $\text{erf}(x)$ صدا زده می‌شود و تابع بسل در همه مرتبه‌ها از آن جمله هستند. همچنین همانگونه که در سایت سیم پاور توضیح دادم توابع خطی نیز برای بدست آوردن اطلاعات درباره ماتریس‌ها وجود دارد، مانند $\text{eig}(A)$ که مقادیر ویژه ماتریس A را پیدا می‌کند.

با این وجود، اغلب نیاز است که توابع عددی با یک، دو و یا سه متغیر ساخته شوند. در این بخش تنها به توابع یک متغیره پرداخته می‌شود. درباره توابع با متغیرهای بیشتر در فصول آینده بحث خواهد شد.

ساخت توابع عددی در خط فرمان بسیار آسان می‌باشد. به این توابع عددی inline function می‌گویند. در اینجا یک مثال ساده از یک تابع عددی بیان می‌شود:

```
>> f=inline('x^۳+x-۱');
```

برای بدست آوردن مقدار تابع $f(x)=x^۳+x-۱$ در $x=۲$ ،کافی است $f(۲)$ را در خط فرمان تایپ کنید.

```
>> f(۲)
ans = ۹
```

در صورتی‌که بخواهیم این تابع array-smart باشد، یعنی بر روی ماتریس به صورت آرایه ای یا به عبارتی بر روی تک تک اعضا ماتریس عمل کند، عبارت تابع به صورت زیر نوشته می‌شود:

```
>> f=inline('x.^۳+x-۱');
```

در اینجا از عملگر .^ برای عملیات آرایه ای استفاده شده است. توابعی که بدین روش ایجاد می‌شوند، می‌توانند بردارها و ماتریس‌ها را به عنوان ورودی بپذیرند. بدین ترتیب تابع به هریک از اعضاء ماتریس یا بردار اعمال می-شود. بطور مثال در صورتی که ماتریس A به صورت زیر تعریف شود:

```
>> A=[۱ ۲ ۳; ٤ ٥ ٦];
```

سپس می‌توان مقدار تابع f را در تک تک عناصر ماتریس A به صورت زیر محاسبه کرد:

```
>> B=f (A)
B =
    ۱     ۹    ۲۹
   ٦۷   ۱۲۹   ۲۲۱
```

یکی از متداول‌ترین اشتباهات این است که فراموش می‌شود توابع عددی با وارد کردن نقطه قبل از عملگرهای *, /, ... به توابع array-smart تبدیل شوند.

متأسفانه با ضرب و اضافه کردن توابع inline نمی‌توان تابع جدیدی ایجاد کرد. اگر تابع inline با نام g به صورت زیر تعریف شود:

```
>> g=inline(' cos(x)+x' );
```

می‌توان از فرمان زیر برای تعریف تابع h استفاده کرد:

```
>> h=f+g
```

برای این منظور باید یک تابع inline جدید به صورت زیر تعریف کرد:

```
>> h=inline(´x.^٣+٢*x-١+cos(x)´ );
```