

عملیات بر روی آرایه ها

عملیات ریاضی بر روی ماتریس ها می تواند بر روی تک تک آرایه ها اجرا شود. به این عملیات، عملیات آرایه ای گویند. ضرب آرایه ها نمونه ای از آن می باشد. در صورتیکه A و B دو ماتریس با ابعاد یکسان باشند و هریک از اعضاء این دو ماتریس به صورت a_{ij} و b_{ij} قابل نمایش باشند، بنابراین فرمان :

```
>> C=A.*B
```

ماتریس دیگری به نام C با ابعاد مشابه A و B ایجاد می کند که اعضاء آن به صورت $c_{ij}=a_{ij}*b_{ij}$ هستند. برای مثال استفاده از ماتریس های 2×3 با نام های A و B به صورت زیر، مشاهده می شود که :

```
>> A=[۱ ۲ ۳; ۲ ۵ ۴];
>> B=[۲ ۰ -۱; ۲ ۲ ۷];
>> C=A.*B
C =
     ۲     ۰    -۳
     ۴    ۱۰    ۲۸
```

عملیات بر روی هر یک از اعضاء یک ماتریس باشد از فرمان $A.^2$. استفاده می شود. برای مثال، در صورتیکه هدف ایجاد ماتریس جدیدی باشد که اعضاء آن مربع اعضاء ماتریس A هستند، می توان نوشت:

```
>> A.^2
ans =
     ۱     ۴     ۹
     ۴    ۲۵    ۱۶
```

همچنین همانگونه که در سایت سیم پاور توضیح دادم یک نوع تقسیم آرایه ای برای دو ماتریس هم بعد وجود دارد که دو ماتریس را عضو به عضو برهم تقسیم می کند:

```
>> D=[۱ ۳ ۵; -۲ ۴ -۱];
>> A./D
ans =
    ۱.۰۰۰۰    ۰.۶۶۶۷    ۰.۶۰۰۰
   -1.0000    ۱.۲۵۰۰   -۴.۰۰۰۰
```

حال توابع `sum` و `cumsum` را بررسی می کنیم. این ها توابعی هستند که اغلب با عملگر نقطه (.) به کار می روند. هنگامی که آرگومان تابع `sum` یک بردار باشد، پاسخ تابع، عددی برابر مجموع تمام آرایه های بردار می باشد

و وقتی که آرگومان تابع sum یک ماتریس باشد، پاسخ یک بردار سطری است که هر عضو آن از مجموع اعضای ستون نظیر آن در ماتریس اصلی به دست می‌آید. پس اگر Z یک ماتریس (3×4) با اعضای Z_{ij} باشد، آنگاه :

$$\text{sum}(Z) = [\sum_{n=1}^3 Z_{n1} \quad \sum_{n=1}^3 Z_{n2} \quad \sum_{n=1}^3 Z_{n3} \quad \sum_{n=1}^3 Z_{n4}] \rightarrow (1 \times 4)$$

یک بردار ۴ عضوی است. همین‌طور

$$\text{sum}(Z') = [\sum_{n=1}^4 Z'_{1n} \quad \sum_{n=1}^4 Z'_{2n} \quad \sum_{n=1}^4 Z'_{3n}] \rightarrow (1 \times 3)$$

یک بردار سه عضوی است.

تابع cumsum برای یک بردار v متشکل از n عضو v_j که خود بردار n عضوی دیگری است بدین شکل تعریف میشود:

$$y = \text{cumsum}(v) = [\sum_{k=1}^1 v_k \quad \sum_{k=1}^2 v_k \quad \dots \quad \sum_{k=1}^n v_k] \rightarrow (1 \times n)$$

به عبارت دیگر اگر w یک ماتریس $(m \times n)$ متشکل از اعضای w_{jk} باشد، آنگاه $\text{cumsum}(w)$ ماتریس زیر است:

$$y = \text{cumsum}(w) = \rightarrow (m \times n)$$

$$\begin{bmatrix} \sum_{k=1}^1 w_{k1} & \sum_{k=1}^1 w_{k2} & \dots & \sum_{k=1}^1 w_{kn} \\ \sum_{k=1}^2 w_{k1} & \sum_{k=1}^2 w_{k2} & & \\ \vdots & & \ddots & \\ \sum_{k=1}^m w_{k1} & & & \sum_{k=1}^m w_{kn} \end{bmatrix}$$

برای نشان دادن کاربرد sum معادله زیر را ملاحظه کنید:

$$Z = \sum_{m=1}^4 m^m$$

دستور محاسبه عبارت بالا به صورت زیر است:

```
>> m=1:4;
>> z=sum(m.^m);
```

هنگامی که دستور فوق اجرا شود، $Z = 288$ را نتیجه می‌دهد. این دستور را می‌توان به صورت خلاصه‌تر نیز نوشت:

```
>> z=sum ((1:4). ^ (1:4));
```

حال نتایج این قسمت را با محاسبه عبارت زیر برای $N=305$ و پنج مقدار x از بازه $0 \leq x \leq 2$ در فواصل یکسان به طور خلاصه شرح می‌دهیم:

$$\operatorname{sech} x = 4\pi \sum_{n=1,3,5}^{N \rightarrow \infty} \frac{n(-1)^{(n-1)/2}}{(n\pi)^2 + 4x^2}$$

باید همچنین همانگونه که در سایت سیم پاور توضیح دادم مقادیر جمع شده با مقادیر درست مقایسه شود. دستورات زیر را برای محاسبه مثال ذکر شده، اجرا کنید:

```
nn=1:2:305;                               %(1x153)
xx=linspace(0,2,5);                         %(1x5)
[x,n]=meshgrid(xx,nn);                     %(153x5)
s=4*pi*sum(n.*(-1).^((n-1)/2)./((pi*n).^2+4*x.^2)); %(1x5)
se=sech(xx);                                %(1x5)
compare=[s' se']                             %(5x2)
```

با اجرا در پنجره فرمان نتیجه می‌دهد :

```
1.0021    1.0000
0.8889    0.8868
0.6501    0.6481
0.4272    0.4251
0.2679    0.2658
```

تابع `meshgrid` برای ایجاد ماتریس‌های (153×5) انتخاب شده است که `sum` جمع تمام سطرها را ستون به ستون انجام می‌دهد. دستورات بالا را می‌توان به صورت خلاصه‌تر زیر نوشت:

```
[x,n]=meshgrid (linspace (0,2,5),1:2:305);
s=4*pi*sum (n.*(-1). ^ ((n-1)/2)./ ((pi*n).^2+4*x.^2)) ;
Compare =[s' sech (linspace (0,2,5)')]
```