

### تمرین شماره چهارم

این فرمول اشاره به ضریب شیب خط در رگرسیون استاندارد دارد. در مورد رگرسیون با اینکه میزان تأثیر هر متغیر بیان می شود اما عدم یکسان بودن واحد های دو متغیری که آورده می شود باعث می شود که نتوان به درستی تأثیر هر متغیر را در متغیر وابسته تفسیر نمود برای رفع این نقیصه و یکسان سازی واحد ها از استاندارد سازی بهره می بریم.

$b_i^* = \frac{\sum_{i=1}^n (x_{iz} - \bar{x})(y_{iz} - \bar{y})}{\sum_{i=1}^n (x_{iz} - \bar{x})^2}$	
---	--

نمره استاندارد Z در این ارتباط متغیرهای مستقل و وابسته همگی به نمره استاندارد تبدیل و ضرایب رگرسیون آنها محاسبه می شود برای حل این فرمول ابتدا باید نمره استاندارد هریک از متغیرها را با استفاده از تابع zscore یا فرمول استاندارد سازی حساب کنیم در اینجا از آنجای که هدف فرمول نویسی می باشد با استفاده از فرمول زیر نمره استاندارد را حساب شد:

$z = \frac{x_i - \bar{x}}{std}$	
---------------------------------	--

برای حل این فرمول همان متغیرهای فرمول سوم را در نظر بگیرید که برای استاندارد سازی آنها به شکل زیر انجام شد:

```
>> zx=(x-mean(x))/std(x);
```

برای متغیرهای y نمره استاندارد بر حسب دستورات زیر حساب شد:

```
>> zy(:,1)=(y(:,1)-mean(y(:,1)))/std(y(:,1));
>> zy(:,2)=(y(:,2)-mean(y(:,2)))/std(y(:,2));
>> zy(:,3)=(y(:,3)-mean(y(:,3)))/std(y(:,3));
>> zy(:,4)=(y(:,4)-mean(y(:,4)))/std(y(:,4));
>> zy(:,5)=(y(:,5)-mean(y(:,5)))/std(y(:,5));
```

نتایج حاصل از محاسبات بالا به صورت زیر می باشد:

zx =	zy =				
0.70625	0.70625	0.97888	0.96573	0.71987	0.48118
1.3254	1.3254	1.9263	2.2225	2.168	1.851
-0.34181	-0.34181	-0.49528	-0.75396	-1.0035	-0.55312
-0.21173	-0.21173	-0.30243	-0.341	0.0862	0.74367
1.4513	1.4513	0.34133	0.19541	0.50091	0.55831
-0.48759	-0.48759	-0.50288	-0.313	0.0802	0.61092
-1.9822	-1.9822	-1.6368	-1.3096	-1.4	-1.0395
-0.50979	-0.50979	-0.74935	-0.60833	-0.30492	-0.40485
0.18078	0.18078	0.59933	0.30278	-0.2019	-1.0059
-0.13067	-0.13067	-0.1591	-0.35973	-0.64492	-1.2418

## فصل سوم - اسکریپت نویسی در متلب

اشاره شد که این محاسبات را به راحتی می توان با دستورات زیر بدست آورد:

```
>> zx=zscore(x);
```

```
>> zy=zscore(y);
```

حالا صورت فرمول را متغیر X و ستون اول ماتریس Y حساب می کنیم که به شکل زیر می باشد:

```
sorat=sum((zx-mean(zx)).*(zy(:,1)-mean(zy(:,1))));
```

محاسبه مخرج فرمول

```
>> makh=sum((zx-mean(zx)).^2);
```

```
>> b=sorat/makh
```

```
b =
```

```
-0.23713
```

حالا اگر بخواهیم کل این دستور را در یک خط بنویسیم به شکل زیر خواهد بود:

```
>>b=(sum((zx-mean(zx)).*(zy(:,1)-mean(zy(:,1)))))/sum((zx-mean(zx)).^2)
```

```
b =
```

```
-0.23713
```

برای سایر ستونها

```
>> b(1,2)=(sum((zx-mean(zx)).*(zy(:,2)-mean(zy(:,2)))))/sum((zx-mean(zx)).^2);
```

```
>> b(1,2)=(sum((zx-mean(zx)).*(zy(:,5)-mean(zy(:,5)))))/sum((zx-mean(zx)).^2);
```

```
>> b(1,2)=(sum((zx-mean(zx)).*(zy(:,2)-mean(zy(:,2)))))/sum((zx-mean(zx)).^2);
```

```
>> b(1,3)=(sum((zx-mean(zx)).*(zy(:,3)-mean(zy(:,3)))))/sum((zx-mean(zx)).^2);
```

```
>> b(1,4)=(sum((zx-mean(zx)).*(zy(:,4)-mean(zy(:,4)))))/sum((zx-mean(zx)).^2);
```

```
>> b(1,5)=(sum((zx-mean(zx)).*(zy(:,5)-mean(zy(:,5)))))/sum((zx-mean(zx)).^2)
```

```
b =
```

```
-0.23713   -0.32434   -0.31902   -0.22116    0.061923
```

برای فرمول نویسی در این فرمول دوتا ساختار دستور اصلی داریم یکی مربوط به نمره Z و یکی مربوط به خود

فرمول یعنی b که به ترتیب در زیر آورده شده است:

```
zy(:,j)=(y(:,j)-mean(y(:,j)))/std(y(:,j));
```

```
b(1,j)=(sum((zx-mean(zx)).*(zy(:,j)-mean(zy(:,j)))))/sum((zx-mean(zx)).^2);
```

شکل برنامه نویسی این فرمول به طریق زیر می باشد:

```
zx=(x-mean(x))/std(x);
```

```
for j=1:5
```

```
zy(:,j)=(y(:,j)-mean(y(:,j)))/std(y(:,j));
```

```
end
```

```
i=1
```

```
while i<=5
```

```
b(1,i)=(sum((zx-mean(zx)).*(zy(:,i)-mean(zy(:,i)))))/sum((zx-mean(zx)).^2);
i=i+1;
end
```

### تمرین شماره پنجم

در این قسمت از فرمول سعی می‌کنیم با استفاده از روش‌های تبدیل یک سری متغیر را به نمره معیار تبدیل کنیم. فرمول زیر که در واقع تبدیل کمینه مشاهدات می‌باشد را در نظر بگیرید:

$KM = 0.8 \left( \frac{x_i - x_{min}}{x_{max} - x_{min}} \right) + 1$	
---	--

اگر دقت کنید این یک فرمول بسیار ساده می‌باشد که بیشتر بر پایه کمینه و بیشینه مشاهدات تکیه دارد. در واقع مثل فرمول آنومالی می‌اشد با این تفاوت که متغیرها را به جای اینکه از میانگین هر ستون کم کند از بیشینه هر ستون کم می‌کند. بنابراین ابتدا صورت فرمول را با استفاده از دستور زیر بر روی داده‌های آب قابل بارش ایران طی پنج دوره ۱۰ ساله منتهی به سال حساب می‌کنیم بنابراین ستون اول آب قابل بارش برای ۱۳۴۰-۱۳۵۰، ستون دوم ۱۳۵۰-۱۳۶۰، ستون سوم ۱۳۶۰-۱۳۷۰، ستون چهارم ۱۳۷۰-۱۳۸۰ و ستون پنجم ۱۳۸۱-۱۳۹۰:

bb =

13.868	14.412	13.902	13.765	13.166
15.309	13.975	14.262	13.569	13.663
14.521	13.639	13.922	14.599	13.238
15.533	14.067	14.039	13.825	13.212
15.584	14.672	14.248	13.401	13.772
14.695	14.869	14.255	14.358	13.594
14.694	13.939	14.408	14.342	13.822
14.832	14.534	13.808	12.97 2	14.361
14.188	14.008	14.111	12.945	13.903
13.892	14.163	13.794	13.466	13.662

در ابتدا صورت فرمول را با استفاده از دستورات زیر برای هر ستون حساب کردیم:

```
>> sor(:,1)=bb(:,1)-min(bb(:,1));
>> sor(:,2)=bb(:,2)-min(bb(:,2));
>> sor(:,3)=bb(:,3)-min(bb(:,3));
>> sor(:,4)=bb(:,4)-min(bb(:,4));
>> sor(:,5)=bb(:,5)-min(bb(:,5));
>> sor(:,5)=bb(:,5)-min(bb(:,5));
```

اگر دقت کرده باشید در دستورات بالا با استفاده از تابع min هر ستون تک تک درایه‌های آن را از کمینه کم شده است. بنابراین باید در هر ستون یک عدد صفر وجود داشته باشد. چون کمینه را از کمینه کم کنیم عدد

حاصله صفر می‌شود. به نتایج دقت کنید:

0	0.761105	0.107895	0.820127	0
1.441022	0.336625	0.467424	0.623526	0.494586
0.652718	0	0.125495	1.653865	0.072341
1.665067	0.428056	0.24429	0.879879	0.045845
1.7152	1.033306	0.453805	0.455924	0.604683
0.826086	1.230316	0.460516	1.413408	0.427978
0.825937	0.300112	0.613648	1.397382	0.65392
0.963637	0.895123	0.01363	0.024709	1.194735
0.319769	0.36879	0.316973	0	0.737034
0.021431	0.524512	0	0.521146	0.496293

اگر دقت کنید در هر ستون یک عدد صفر داریم که با توجه به صورت فرمون می‌توان گفت که جای که مقدار آن صفر باشد محل کمینه هر ستون می‌باشد. حالا با توجه به دستور min و max مخرج فرمول را حساب می‌کنیم:

```
>> makh(1,1)=max(bb(:,1))-min(bb(:,1));
>> makh(1,2)=max(bb(:,2))-min(bb(:,2));
>> makh(1,3)=max(bb(:,3))-min(bb(:,3));
>> makh(1,4)=max(bb(:,4))-min(bb(:,4));
>> makh(1,5)=max(bb(:,5))-min(bb(:,5));
```

makh =

1.7152	1.2303	0.61365	1.6539	1.1947
--------	--------	---------	--------	--------

در مرحله آخر باید 0.8 را در حاصل صورت بر مخرج به شرح زیر تقسیم کنیم:

```
>> nb(:,1)=0.8*(sor(:,1)./makh(1,1))+1;
>> nb(:,2)=0.8*(sor(:,2)./makh(1,2))+1;
>> nb(:,3)=0.8*(sor(:,3)./makh(1,3))+1;
>> nb(:,4)=0.8*(sor(:,4)./makh(1,4))+1;
>> nb(:,5)=0.8*(sor(:,5)./makh(1,5))+1;
```

nb=

1	1.494901	1.14066	1.396708	1
1.672118	1.218887	1.60937	1.301609	1.331177
1.304439	1	1.163605	1.8	1.04844
1.776617	1.278339	1.318476	1.425611	1.030698
1.8	1.671897	1.591616	1.220537	1.404899
1.385301	1.8	1.600365	1.683687	1.286576
1.385232	1.195144	1.8	1.675935	1.437868
1.449458	1.582045	1.017769	1.011952	1.8
1.149146	1.239802	1.413231	1	1.493521
1.009996	1.341059	1	1.252086	1.33232

همان طوری که اشاره شده است می توان این دستورات را به صورت خلاصه تر مطابق دستورت پایین نوشت:

```
>> kol(:,1)=0.8*((bb(:,1)-min(bb(:,1)))/(max(bb(:,1))-min(bb(:,1))))+1;
>> kol(:,2)=0.8*((bb(:,2)-min(bb(:,2)))/(max(bb(:,2))-min(bb(:,2))))+1;
>> kol(:,3)=0.8*((bb(:,3)-min(bb(:,3)))/(max(bb(:,3))-min(bb(:,3))))+1;
>> kol(:,4)=0.8*((bb(:,4)-min(bb(:,4)))/(max(bb(:,4))-min(bb(:,4))))+1;
>> kol(:,5)=0.8*((bb(:,5)-min(bb(:,5)))/(max(bb(:,5))-min(bb(:,5))))+1;
```

اگر بخواهیم اسکریپتی برای این فرمول بنویسیم با آشنایی با ساختار دستور به راحتی می توان برنامه ای برای این فرمول را نوشت. ابتدا ساختار دستور فرمول را می نویسیم:

```
kol(:,j)=0.8*((bb(:,j)-min(bb(:,j)))/(max(bb(:,j))-min(bb(:,j))));
```

در اسکریپت کافیسیت به جای خط تیره ها در ساختار دستور j یعنی حلقه را قرار داد:

```
for j=1:5
kol(:,j)=0.8*((bb(:,j)-min(bb(:,j)))/(max(bb(:,j))-min(bb(:,j))))+1;
end
```

بنابراین با اجرای برنامه نتایجی مشابه با آنچه که در بالا حساب کردیم به ما می دهد:

nb=

1	1.494901	1.14066	1.396708	1
1.672118	1.218887	1.60937	1.301609	1.331177
1.304439	1	1.163605	1.8	1.04844
1.776617	1.278339	1.318476	1.425611	1.030698
1.8	1.671897	1.591616	1.220537	1.404899
1.385301	1.8	1.600365	1.683687	1.286576
1.385232	1.195144	1.8	1.675935	1.437868
1.449458	1.582045	1.017769	1.011952	1.8
1.149146	1.239802	1.413231	1	1.493521
1.009996	1.341059	1	1.252086	1.33232

تمرین شماره شش

در این قسمت یک فرمول دیگر را که از خانواده تبدیلات خطی می باشد بر روی همین متغیرها (یعنی ۵ دوره آب قابل بارش ایران پیاده خواهیم کرد:

$O_i = \frac{x_i}{\sum_{i=1}^n x_i}$	
--------------------------------------	--

این نسبت به فرمول های دیگر بسیار آسان تر می باشد. در فرمول هر متغیر را بر مجموع کل متغیرهای در هر ستون تقسیم می کنیم:

```
>> su=sum(bb(:,1));
>> O=bb(:,1)./su;
O=
0.094269
```

0.10406  
0.098706  
0.10559  
0.10593  
0.099884  
0.099883  
0.10082  
0.096443  
0.094415

برای سایر ستون‌ها مطابق دستورات بالا می‌نویسیم. اما اگر بخواهید شکل کلی دستورات را برای همه ستون‌ها اجرا کنید به شکل زیر خواهد بود:

```
>> Oi(:,1)=bb(:,1)./sum(bb(:,1));
>> Oi(:,2)=bb(:,2)./sum(bb(:,2));
>> Oi(:,3)=bb(:,3)./sum(bb(:,3));
>> Oi(:,4)=bb(:,4)./sum(bb(:,4));
>> Oi(:,5)=bb(:,5)./sum(bb(:,5));
```

Oi =

0.094269	0.101218	0.098775	0.1003	0.096534
0.104064	0.098235	0.101329	0.098867	0.10016
0.098706	0.095868	0.0989	0.106375	0.097064
0.105587	0.098877	0.099744	0.100735	0.09687
0.105928	0.103132	0.101232	0.097646	0.100967
0.099884	0.104516	0.10128	0.104623	0.099672
0.099883	0.097978	0.102368	0.104506	0.101328
0.100819	0.10216	0.098105	0.094504	0.105294
0.096443	0.098461	0.10026	0.094324	0.101938
0.094415	0.099555	0.098008	0.098121	0.100173

برای نوشتن اسکریپت این فرمول مطابق ساختار به شکل زیر خواهد بود:

ساختار دستور این فرمول

```
>> Oi(:,j)=bb(:,j)./sum(bb(:,j));
```

```
for j=1:5
    Oi(:,j)=bb(:,j)./sum(bb(:,j));
end
```