

=====有关 mapminmax 的用法详解=====

几个要说明的函数接口:

```
[Y,PS] = mapminmax(X)
```

```
[Y,PS] = mapminmax(X,FP)
```

```
Y = mapminmax('apply',X,PS)
```

```
X = mapminmax('reverse',Y,PS)
```

用实例来讲解,测试数据 $x1 = [1 \ 2 \ 4]$, $x2 = [5 \ 2 \ 3]$;

```
>> [y,ps] = mapminmax(x1)
```

```
y =
```

```
    -1.0000    -0.3333     1.0000
```

```
ps =
```

```
name: 'mapminmax'
```

```
  xrows: 1
```

```
  xmax: 4
```

```
  xmin: 1
```

```
 xrange: 3
```

```
 yrows: 1
```

```
 ymax: 1
```

```
 ymin: -1
```

```
 yrange: 2
```

其中 y 是对进行某种规范化后得到的数据,这种规范化的映射记录在结构体 ps 中. 让我们来看一下这个规范化的映射到底是怎样的?

$$y = (ymax-ymin)*(x-xmin)/(xmax-xmin) + ymin;$$

[关于此算法的一个问题. 算法的假设是每一行的元素都不想相同, 那如果都相同怎么办? 实现的办法是, 如果有一行的元素都相同比如 $xt = [1 \ 1 \ 1]$, 此时 $xmax = xmin = 1$, 把此时的变换变为 $y = ymin$, matlab 内部就是这么解决的. 否则该除以 0 了, 没有意义!]

也就是说对 $x1 = [1 \ 2 \ 4]$ 采用这个映射 $f: 2*(x-xmin)/(xmax-xmin)+(-1)$, 就可以得到 $y = [-1.0000 \quad -0.3333 \quad 1.0000]$

我们来看一下是不是: 对于 $x1$ 而言 $xmin = 1, xmax = 4$;

则:

$$y(1) = 2*(1 - 1)/(4-1)+(-1) = -1;$$

$$y(2) = 2*(2 - 1)/(4-1)+(-1) = -1/3 = -0.3333;$$

$$y(3) = 2*(4-1)/(4-1)+(-1) = 1;$$

看来的确就是这个映射来实现的.

对于上面 algorithm 中的映射函数 其中 ymin, 和 ymax 是参数, 可以自己设定, 默认为 -1, 1;

比如:

```
>>[y,ps] = mapminmax(x1);
```

```
>> ps.ymin = 0;
```

```
>> [y,ps] = mapminmax(x1,ps)
```

```
y =
```

```
    0    0.3333    1.0000
```

```

ps =
name: 'mapminmax'
xrows: 1
xmax: 4
xmin: 1
xrange: 3
yrows: 1
ymax: 1
ymin: 0
yrange: 1

```

则此时的映射函数为: $f: 1*(x-xmin)/(xmax-xmin)+(0)$ 。

如果我对 $x1 = [1 \ 2 \ 4]$ 采用了某种规范化的方式, 现在我要对 $x2 = [5 \ 2 \ 3]$ 采用同样的规范化方式[同样的映射], 如下可办到:

```

>> [y1, ps] = mapminmax(x1);
>> y2 = mapminmax('apply', x2, ps)
y2 =
    1.6667    -0.3333     0.3333

```

即对 $x1$ 采用的规范化映射为: $f: 2*(x-1)/(4-1)+(-1)$, (记录在 ps 中), 对 $x2$ 也要采取这个映射.

$x2 = [5, 2, 3]$, 用这个映射我们来算一下.

$y2(1) = 2(5-1)/(4-1)+(-1) = 5/3 = 1+2/3 = 1.66667$

$y2(2) = 2(2-1)/(4-1)+(-1) = -1/3 = -0.3333$

$y2(3) = 2(3-1)/(4-1)+(-1) = 1/3 = 0.3333$

$X = \text{mapminmax}('reverse', Y, PS)$ 的作用就是进行反归一化, 讲归一化的数据反归一化再得到原来的数据:

```

>> [y1, ps] = mapminmax(x1);
>> xtt = mapminmax('reverse', y1, ps)
xtt =

```

```

    1    2    4

```

此时又得到了原来的 $x1$ ($xtt = x1$);