

پیش پردازش و پس پردازش بر روی داده ها:

پیش پردازش بر روی داده ها، ورودی ها را به فرم بهتری برای استفاده شبکه تبدیل می کند.

یکی از توابع مهم برای پیش پردازش بر روی داده ها، تابع `fixunknowns` است که در حقیقت بر روی مقادیر بدون مقدار عمل می کند. داده های زیر را در نظر بگیرید:

```
p1 = [1 NaN 3 2 NaN; 3 1 -1 2 4];
```

شبکه با این مقادیر نمیتواند کار کند. ولی می توانیم سطری که دارای مقدار `NaN` است را تبدیل به دو سطر کرده و به صورت کد گذاری تبدیل کنیم.

```
[p2,ps] = fixunknowns(p1);
```

با اجرای این دستور می توانید خروجی زیر را مشاهده کنید:

```
p2 =  
1 2 3 2 2  
1 0 1 1 0  
3 1 -1 2 4
```

سطر اول همان سطر داده های اصلی است فقط مقادیر `NaN` را با میانگین مقادیر دو سمت خود جانشین کرده ایم. و در سطر دوم مشخص شده است کدام داده اصلی است و کدام داده غیر اصلی و این با مقادیر `0` و `1` مشخص شده است. برای خروجی، تابعی که برای `backpropagation` استفاده شده اند، مقادیر `NaN` برای خروجی را جز داده های بی اهمیت می داند.

:Min and Max (mapminmax)

یکی دیگر از پیش پردازش ها این است که همه ی داده ها را در بازه ی خاصی قرار دهیم. تابع `mapminmax` ورودی ها را در رنج `[-1,1]` قرار می دهد. ورودی ها و خروجی ها در ماتریس های `p` و `t` قرار دارند. با دستورات زیر این رنج بندی را اعمال می کنیم.

```
[pn,ps] = mapminmax(p);
```

```
[tn,ts] = mapminmax(t);
```

سپس برای شبکه از متغیرهای `pn` و `tn` به صورت زیر می توانیم استفاده کنیم:

```
net = train(net,pn,tn);
```

زمانی که شبکه آموزش دیده می شود و می خواهید برای نمونه جدید خروجی را ببینید، باید خروجی شبکه را به حالت اول برگردانید. برای این کار از متغیر **ts** استفاده کنید. به دستورات زیر توجه کنید:

```
an = sim(net,pn);
```

```
a = mapminmax('reverse',an,ts);
```

با استفاده از این دستور می توانید خروجی را به حالت اول ببینید.

www.matlabproject.ir