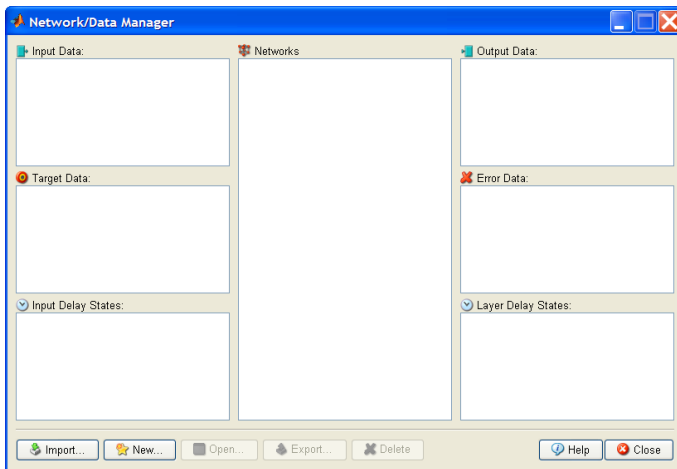
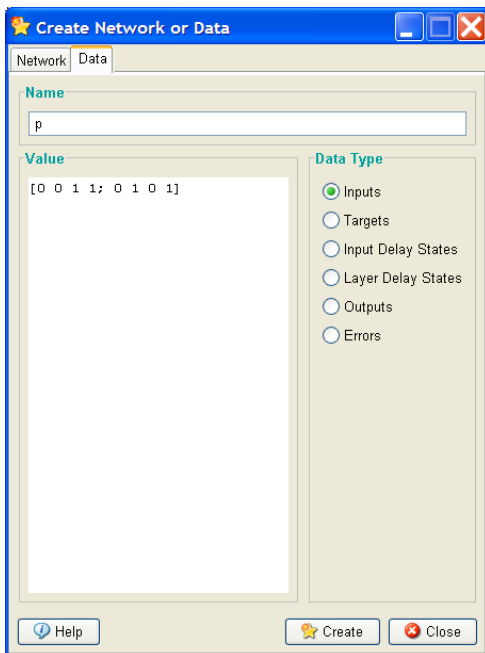


با اجرای دستور nntool تصویر زیر باز می شود:



با وارد کردن داده که آشنا هستید اما می توانید داده ی جدیدی نیز تولید کنید برای این کار دکمه ی new را بزنید تا صفحه ی زیر باز شود:

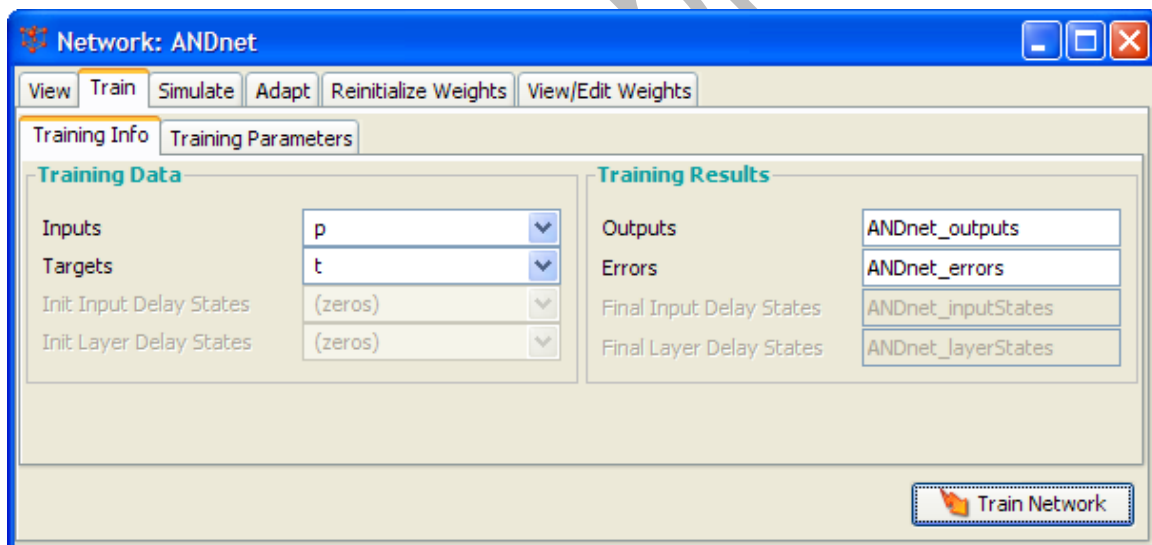


این صفحه سه بخش دارد. در نوار بالای نام در بخش سمت راست نوع آن که ورودی است یا خروجی مشخص می شود و در بخش سمت چپ مقادیر آن را می نویسید. می توانید برای مقادیر مانند ما عمل کنید:

$$P = [0 \ 0 \ 1 \ 1; 0 \ 1 \ 0 \ 1]$$

بعد از وارد کردن مقادیر با زدن دکمه ی **creat** داده ساخته می شود. برای داده ی خروجی هم می توانید از همین روش استفاده کنید.

در سربرگ **network** می توانید مشخصات شبکه تنظیم کنید. نوع شبکه را **perceptron** و ورودی خروجی ها را تعیین کنید. تابع انتقال را **HARDIm** و الگوریتم یادگیری را **LEARNnp** انتخاب کنید. سپس با زدن **creat** شبکه ایجاد می شود و می توانید صفحه ی **Network/Data Manager** **window** را ببینید. برای آموزش **ANDNet** را کلیک کنید تا برجسته شود و سپس **open** کنید تا تصویر زیر وارد شوید:



در سربرگ **view** می توانید شبکه را ببینید. در سربرگ **train** می توانید هم اطلاعات مربوط به ورودی و خروجی در بخش **training info** مشاهده کنید و هم در سربرگ **training parameters** می توانید پارامترهای مربوط به آموزش مانند تعداد **epoch** و یا خطای هدف را تعیین کنید. همچنین با زدن دکمه ی **train network** شبکه را آموزش دهید.

بعد از آموزش می توانید در صفحه ی تصویر قبلی در سربرگ **simulate** از منوی ورودی، **p** را به عنوان ورودی مشخص کنید و نام خروجی را **ANDNet_outputsSim** بنامید و سپس **Simulate** **Network** را بزنید در **Network/Data Manager** خروجی با نامی که گذاشتید را می بینید دوبل کلیک کنید تا مقادیر آن را مشاهده کنید.

جلسه ی بعدی **linear networks** را مورد بررسی قرار می دهیم.

www.matlabproject.ir