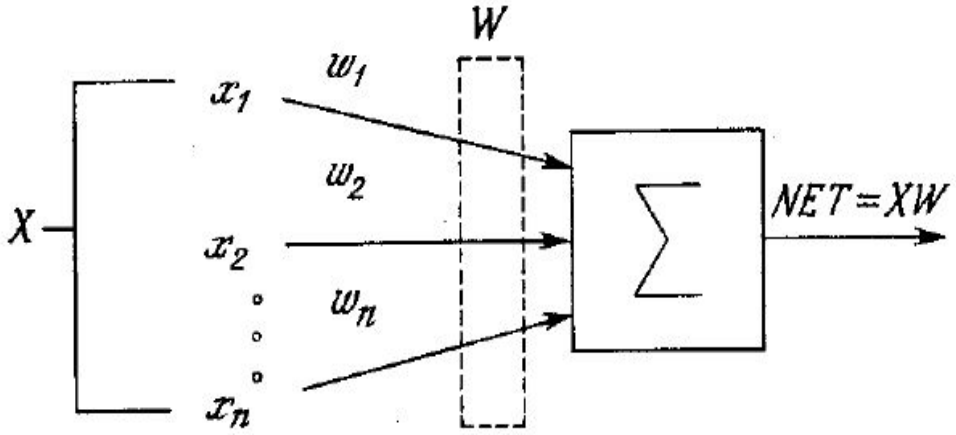
Нейронные сети

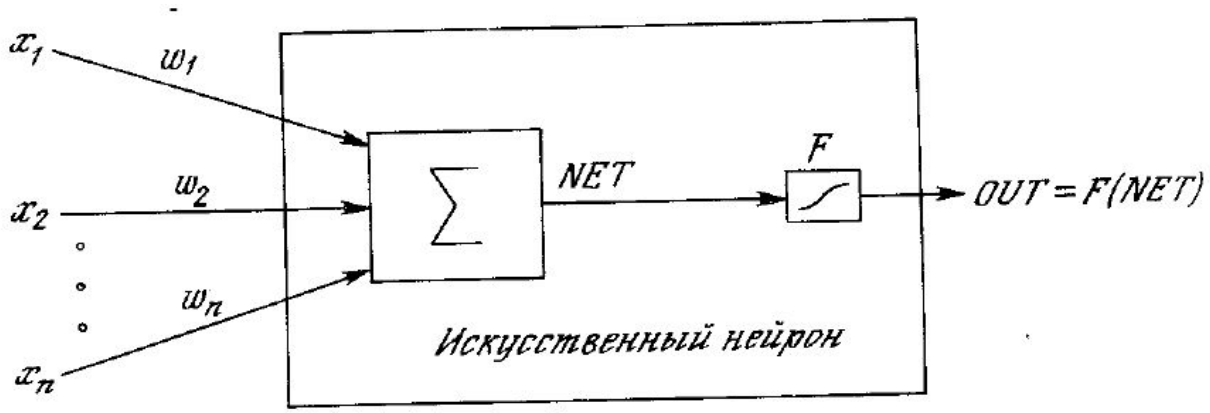
Реализация нейронных сетей в среде MatLab

Основные понятия.
Нейрон.



- X – входной вектор;
- W – вектор весов;
- Σ – суммирующий блок;
- NET – выход.

Основные понятия.
Нейрон.

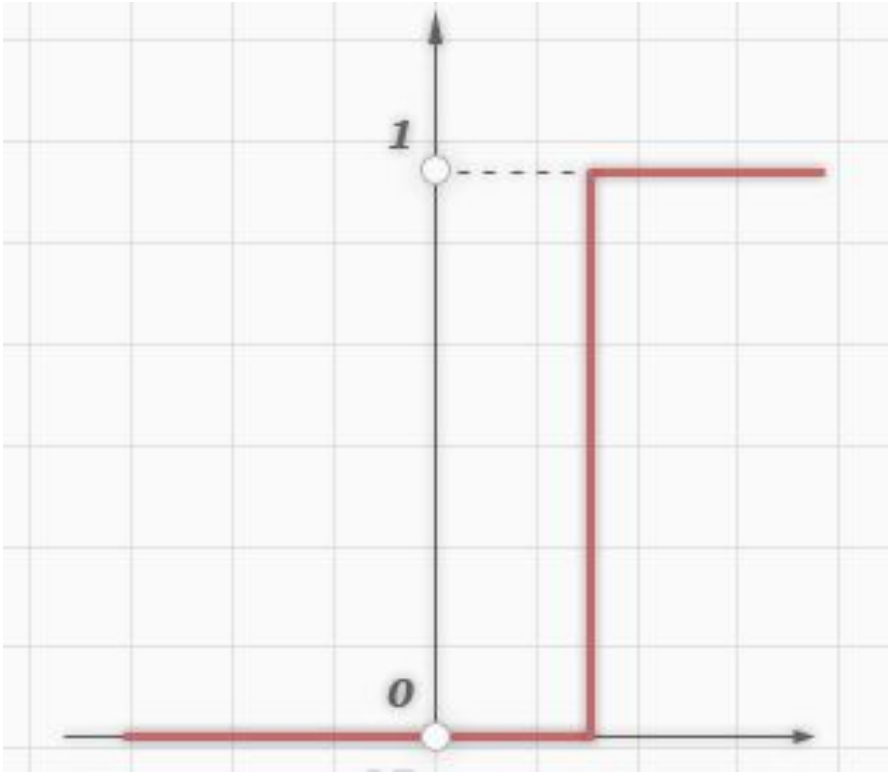


- NET – промежуточный выход;
- F – активационная функция;
- OUT – выход.

Активационная функция должна иметь ограниченное множество значений, к примеру [0;1]. Это необходимое условие для нормализации выходного значения.

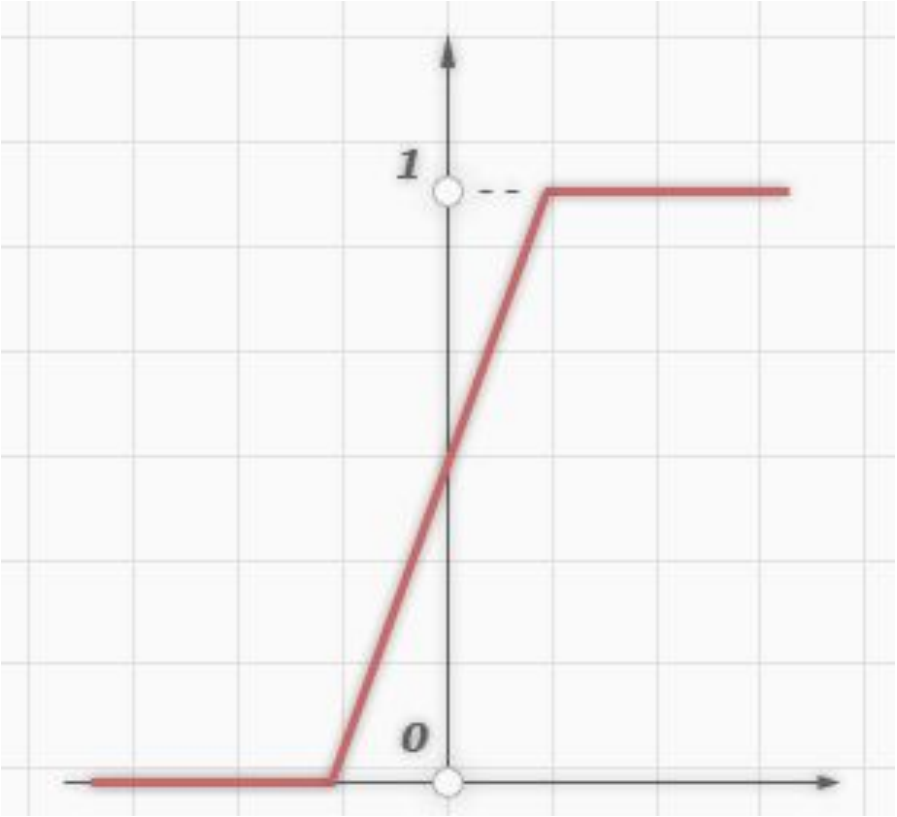
Основные понятия.
Примеры активационных функций.

- Единичный скачок или жесткая пороговая функция



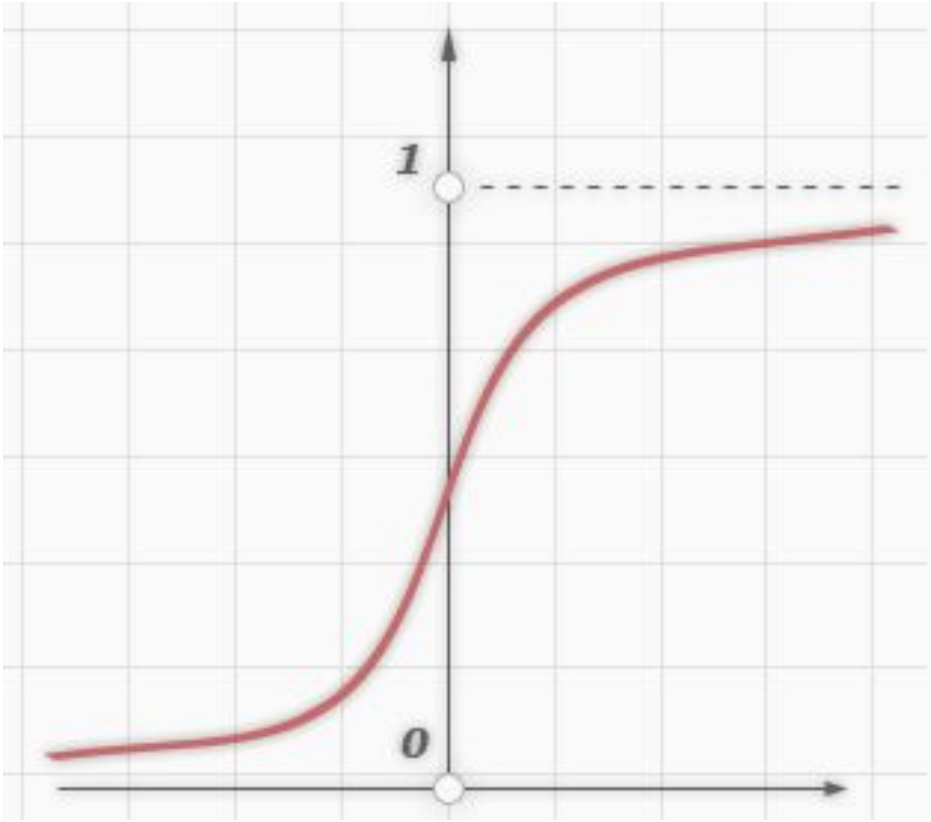
Основные понятия.
Примеры активационных функций.

- Линейный порог или гистерезис

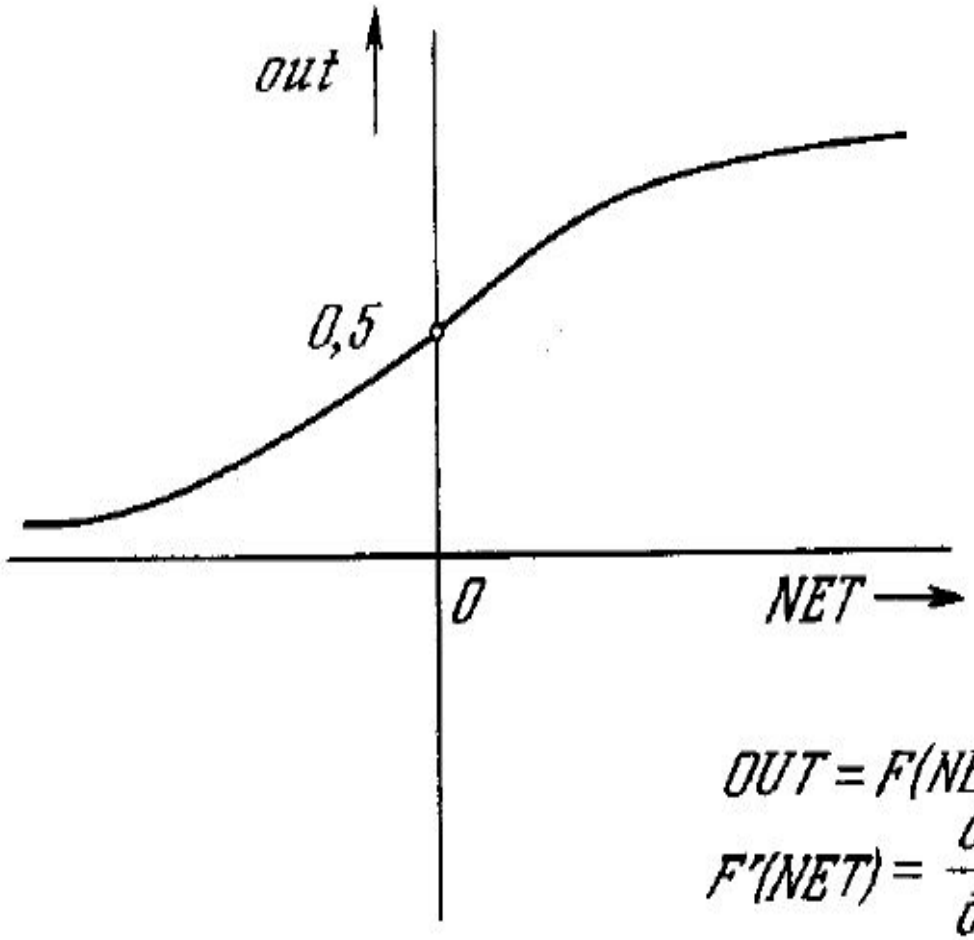


Основные понятия.
Примеры активационных функций.

- Сигмоид



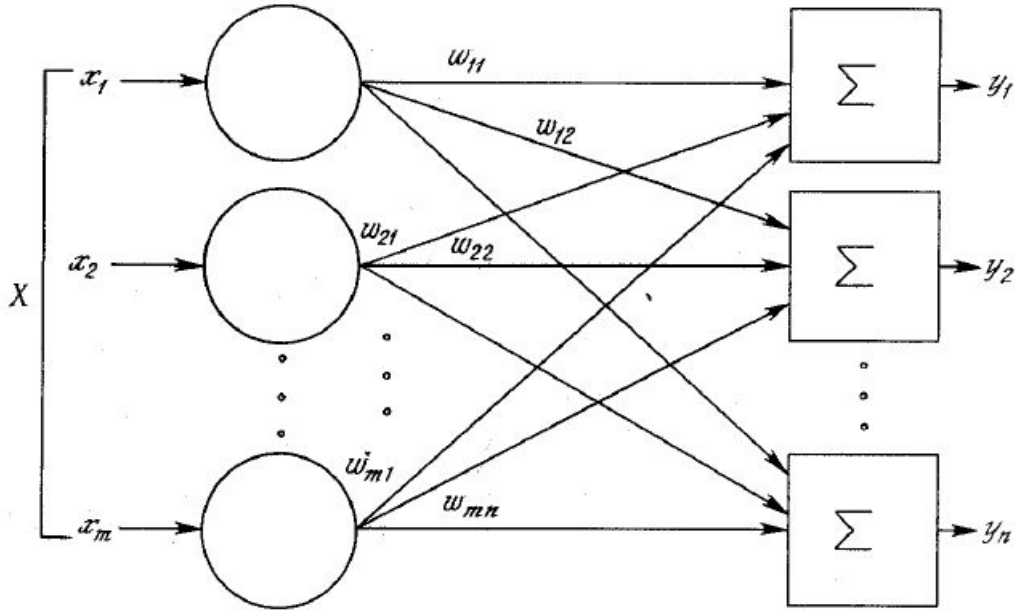
Основные понятия.
Примеры активационных функций.



$$OUT = F(NET) = 1/(1 + e^{-NET})$$

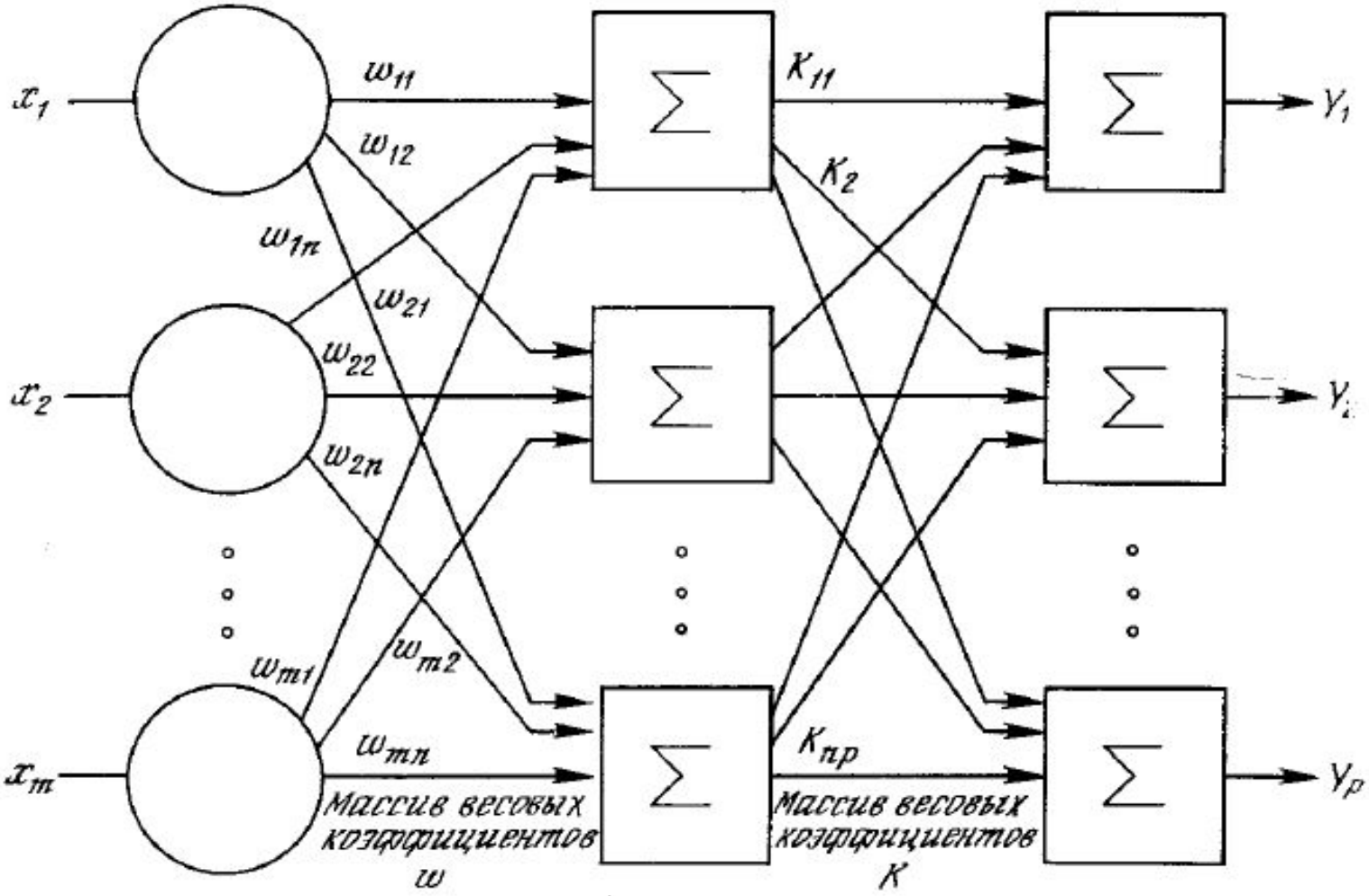
$$F'(NET) = \frac{\partial_{OUT}}{\partial^{NET}} = OUT(1-OUT)$$

Основные понятия.
Нейронная сеть. Однослойная сеть.

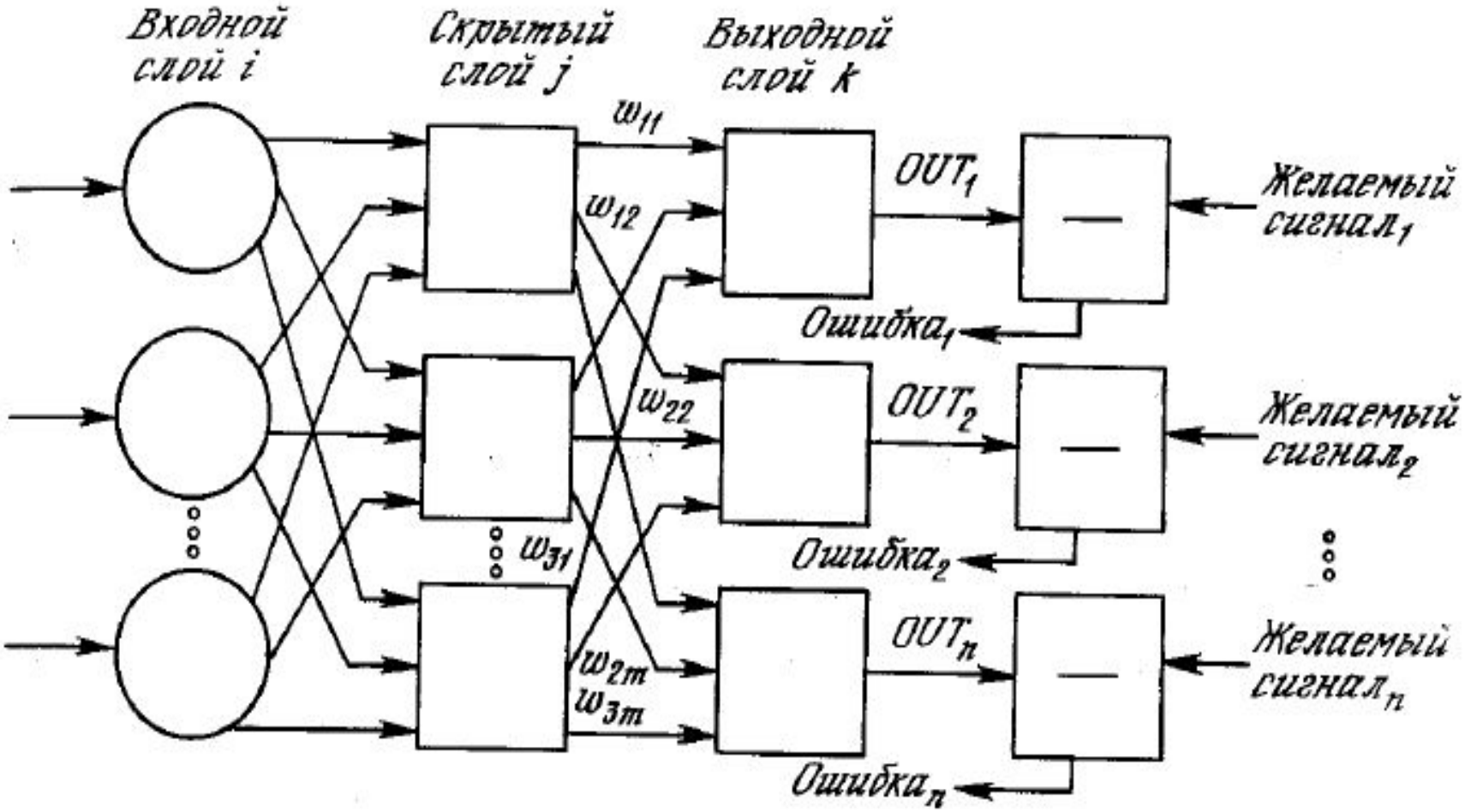


- X – входной вектор;
- W_{ij} – вес, определенный для значения x_i у нейрона y_j;
- Y – выходной вектор.

Основные понятия.
Нейронная сеть. Многослойная сеть.



Обучение сети.
Нейронная сеть с обратным распространением.



Обучение сети. Алгоритм.

- Обучение сети обратного распространения требует выполнения следующих операций:

1. Выбрать очередную обучающую пару из обучающего множества; подать входной вектор на вход сети.
2. Вычислить выход сети.
3. Вычислить разность между выходом сети и требуемым выходом (целевым вектором обучающей пары).
4. Подкорректировать веса сети так, чтобы минимизировать ошибку.
5. Повторять шаги с 1 по 4 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня.

Обучение сети. Алгоритм (продолжение).

- Новое значение весов считается по формуле:

$$W_{pq} = W_{pq} + \mu\delta_q OUT$$

- где *p* и *q* – номера нейронов.
- Для выходного слоя:

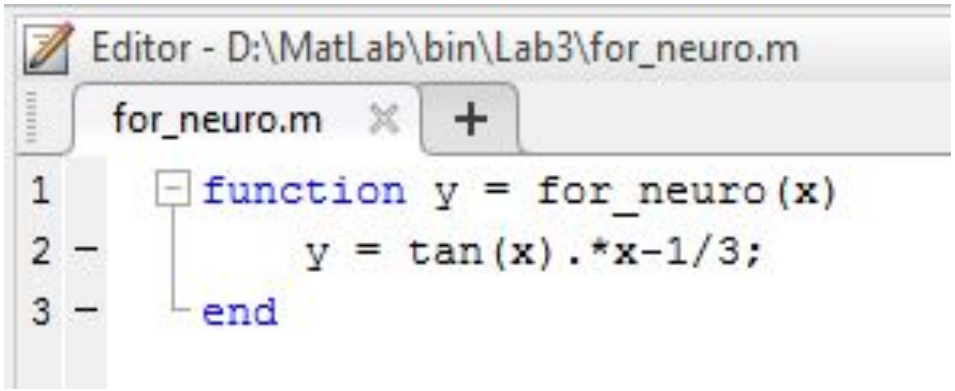
$$\delta = OUT(1 - OUT)(T - OUT)$$

- где *T* – желаемый результат, а *OUT(1-OUT)* – производная сигмоида.
- Для остальных слоёв:

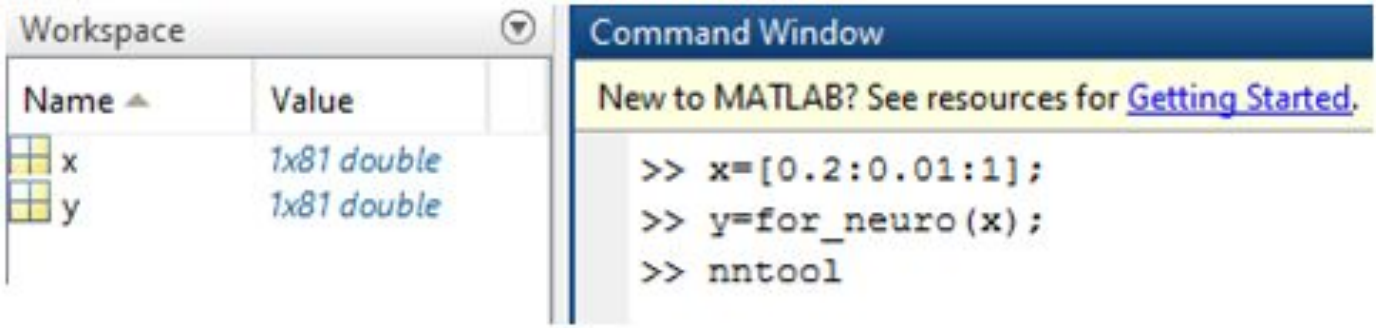
$$\delta = OUT_{p,i}(1 - OUT_{p,i})\left[\sum_q \delta_q W_{pq}\right]$$

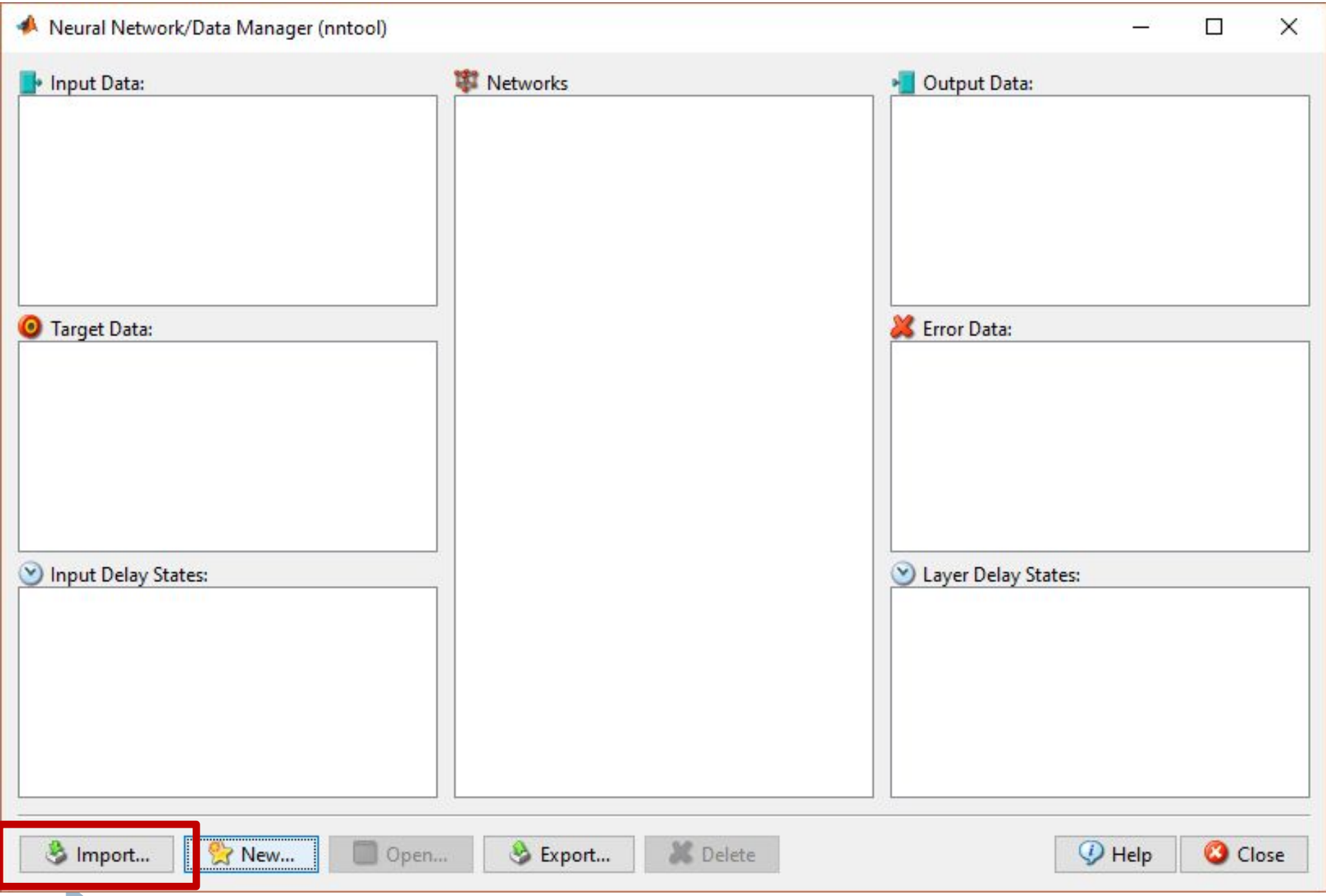
Работа в пакете MATLAB. Подготовка обучающих выборок

- Скрипт с описанием исследуемой функции

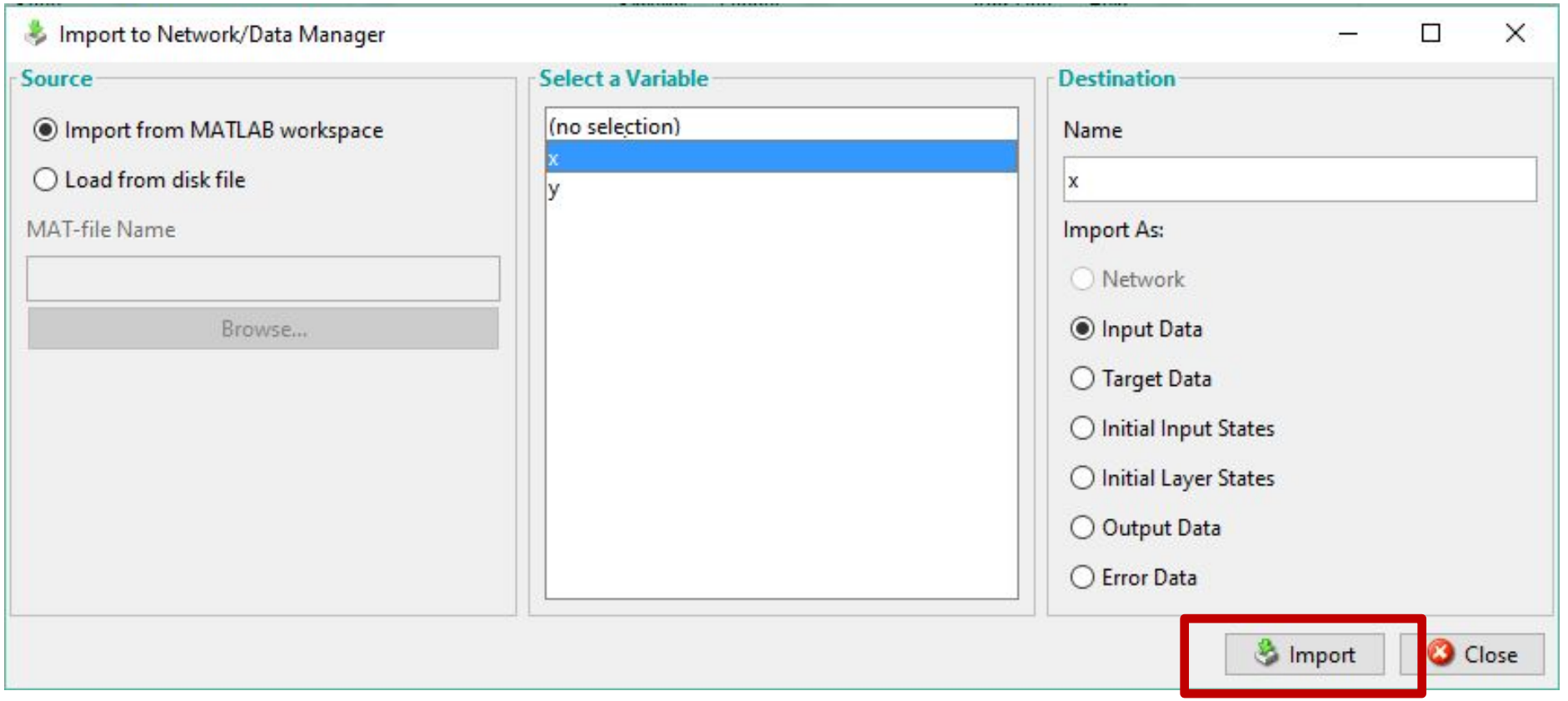


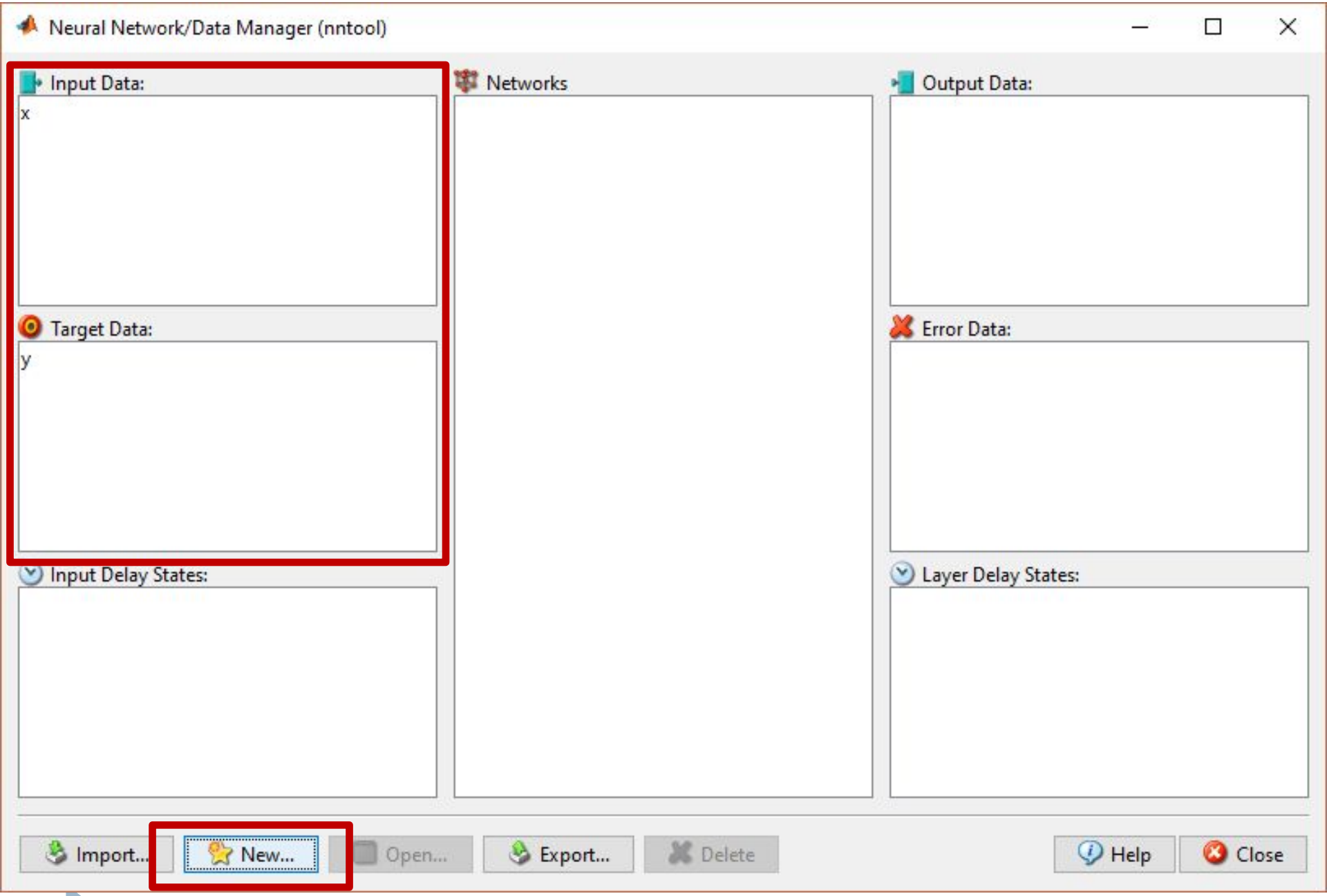
- Генерация выборок

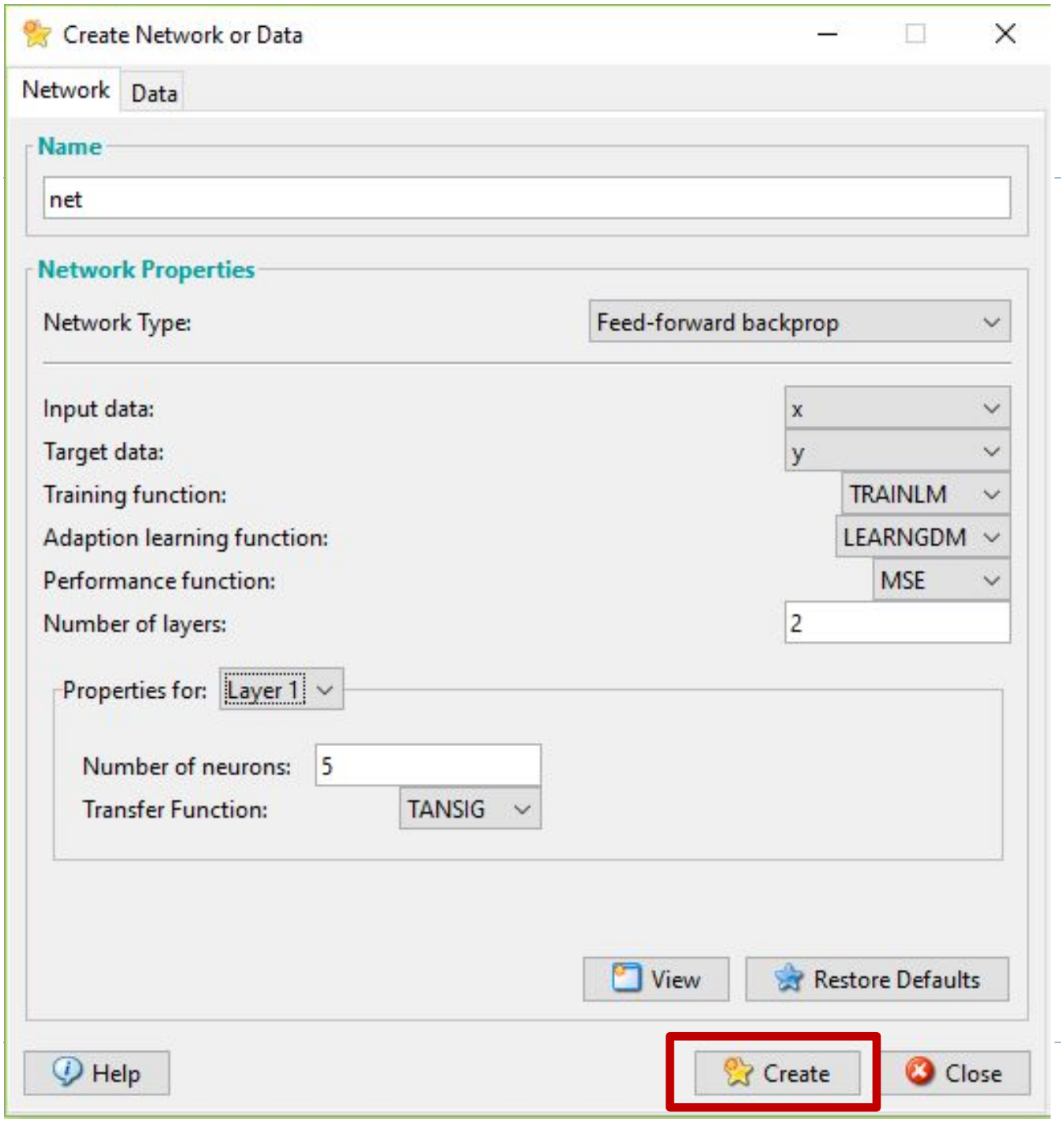



Neural Network/Data Manager (nntool)
Input Data:
Networks
Output Data:
Target Data:
Error Data:
Input Delay States:
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close

Работа в пакете MATLAB. Импорт данных и создание нейронной сети.




Neural Network/Data Manager (nntool)
Input Data:
x
Target Data:
y
Input Delay States:
Networks
Output Data:
Error Data:
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close


Create Network or Data
Network
Data
Name
net
Network Properties
Network Type:
Feed-forward backprop
Input data:
x
Target data:
y
Training function:
TRAINLM
Adaption learning function:
LEARNGDM
Performance function:
MSE
Number of layers:
2
Properties for:
Layer 1
Number of neurons:
5
Transfer Function:
TANSIG
View
Restore Defaults
Help
Create
Close

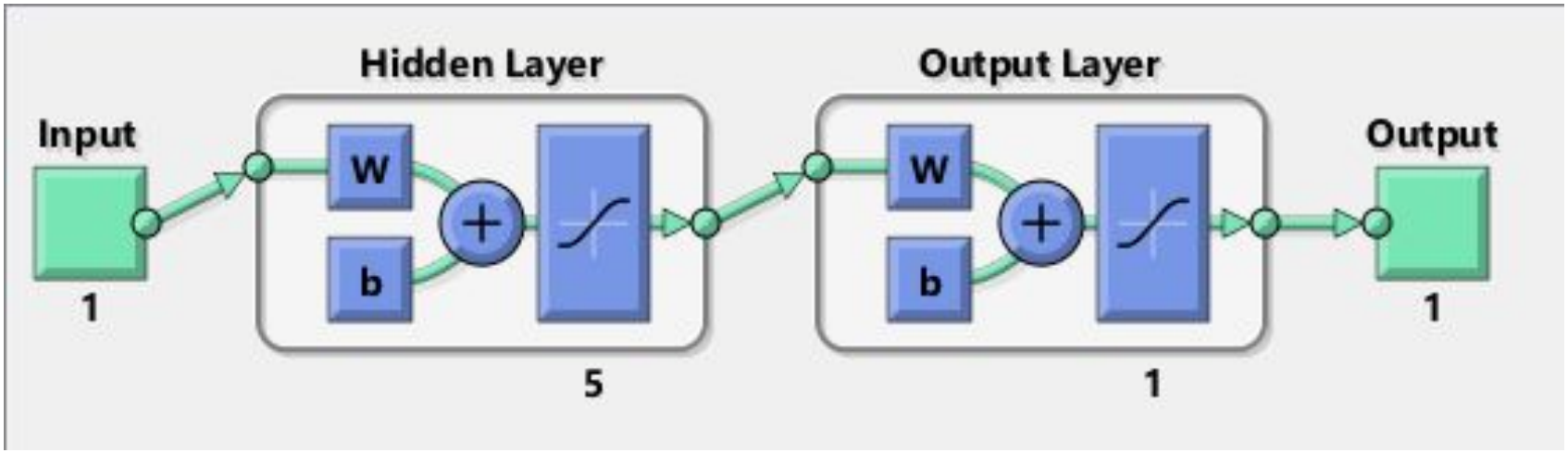
Работа в пакете MATLAB. Параметры нейронной сети.

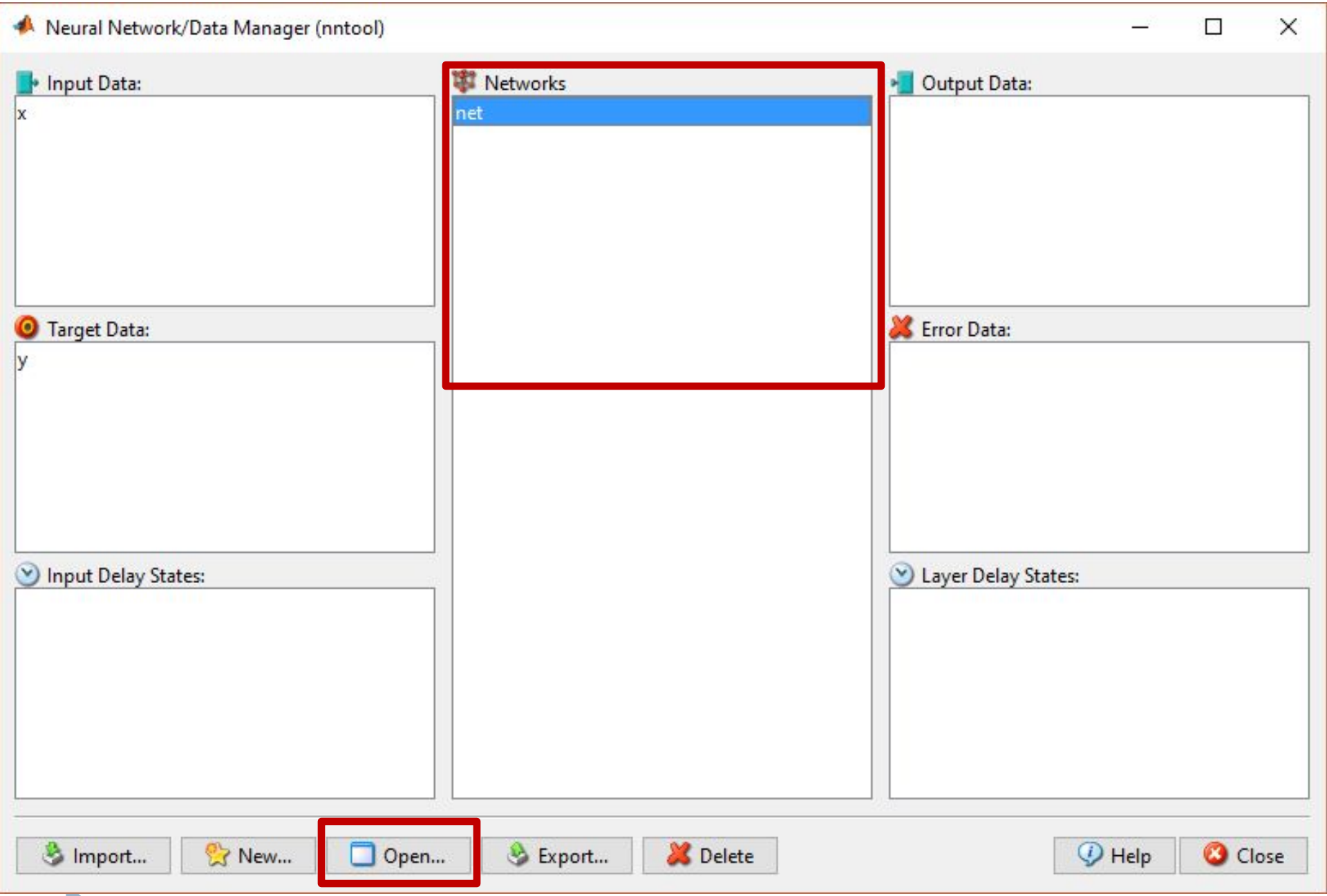
- **Network type** – список сетей, доступных для работы.
- **Input Data, Target Data** – данные, представляющие входной и выходной векторы соответственно.
- **Training function** – обучающая функция (по умол. метод оптимизации Левенберга-Маркара).
- **Adaptation learning function** - функция, отвечающая за обновление весов и смещений сети в процессе обучения (по умол. метод градиентного спуска)
- **Performance function** - функция оценки качества обучения (по умол. среднеквадратичная ошибка)
- **Number of layers** – число слоев сети

Работа в пакете MATLAB. Параметры нейронной сети.

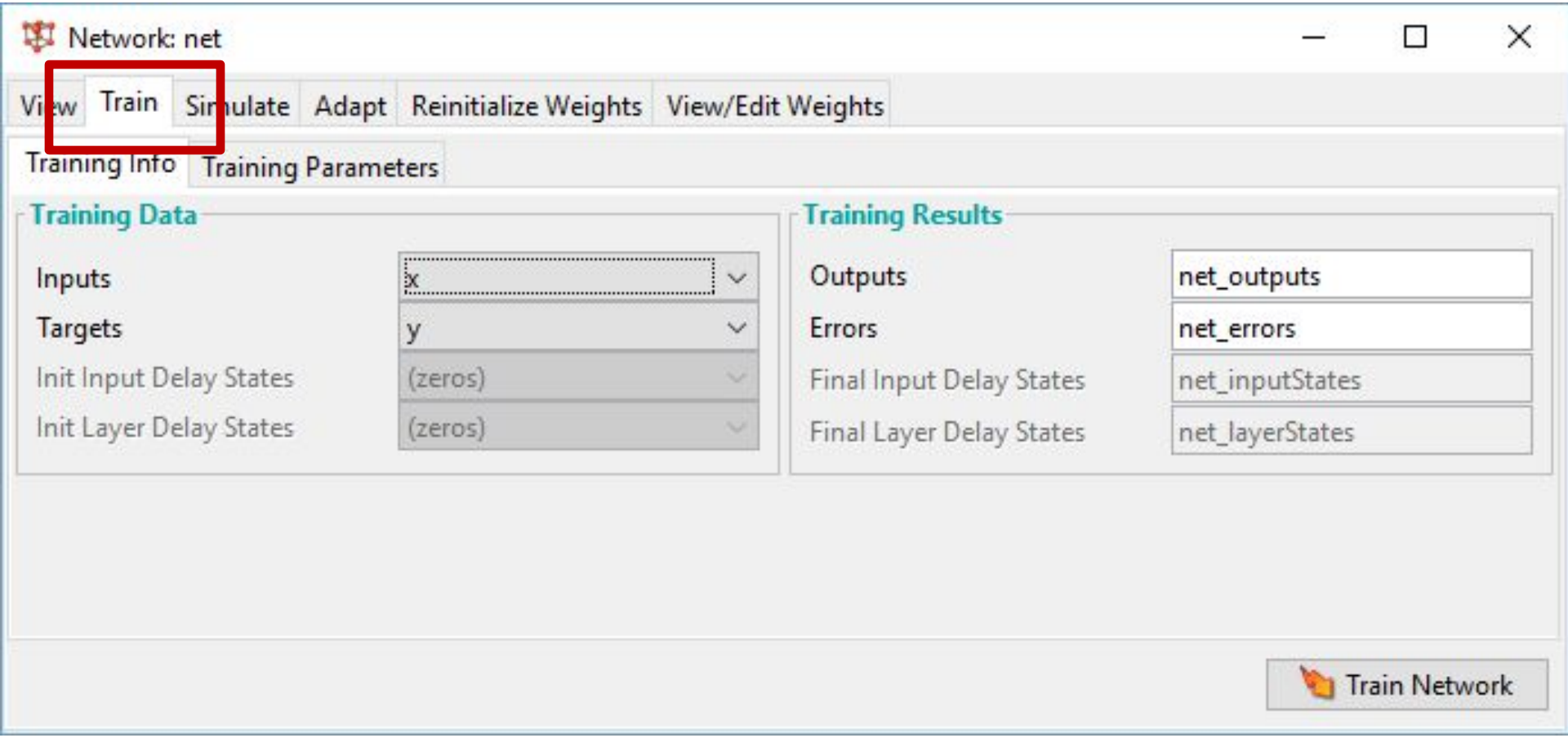
- **Number of neurons** – число нейронов.
- **Transfer function** – активационная функция.

Работа в пакете MATLAB. Нейронная сеть.

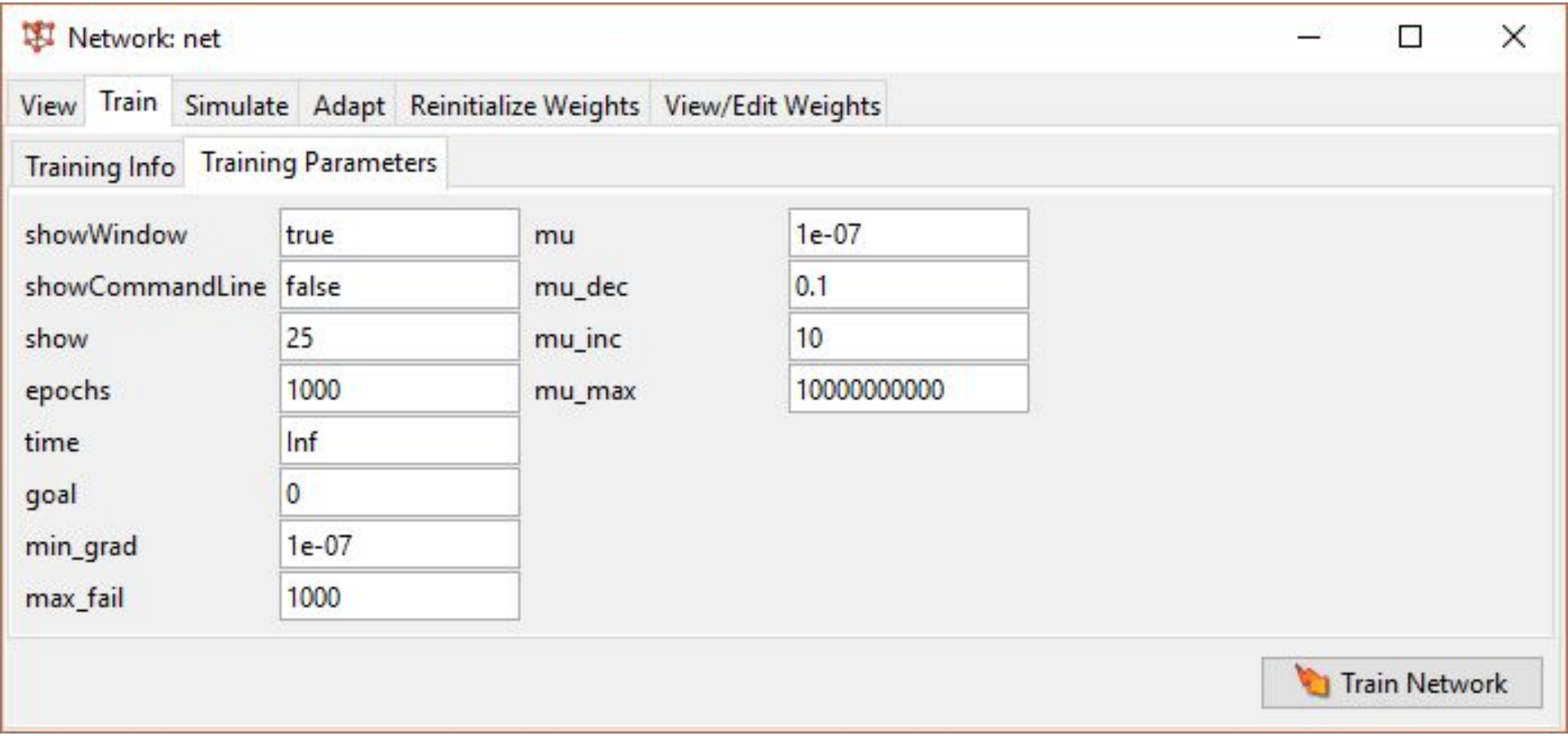



Neural Network/Data Manager (nntool)
Input Data:
x
Networks
net
Output Data:
Target Data:
y
Error Data:
Input Delay States:
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close

Работа в пакете MATLAB. Обучение нейронной сети.



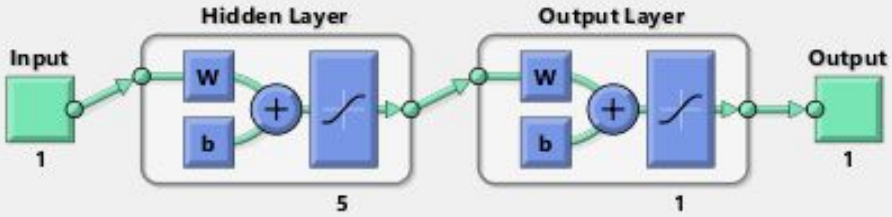
Работа в пакете MATLAB. Обучение нейронной сети.



Работа в пакете MATLAB. Параметры обучения нейронной сети.

- **showWindow** – вывод процесса обучения в графическом режиме.
- **showCommandLine** - вывод процесса обучения в командную строку.
- **show** - период обновления графика кривой обучения, выраженный числом эпох
- **epoch** – число эпох, по прошествии которых обучение заканчивается.
- **time** – время, по истечении которого обучение прекращается
- **goal** – значение функции ошибки, при которой цель будет считаться достигнутой.
- **min_grad** – минимальный градиент.
- **max_fail** – максимальное число ошибок.
- **mu** - начальное значение mu.
- **mu_dec** - коэффициент убывания mu.
- **mu_inc** - коэффициент возрастания mu.
- **mu_max** - максимальное значение mu.

Neural Network



Algorithms

Data Division:	Random	(dividerand)
Training:	Levenberg-Marquardt	(trainlm)
Performance:	Mean Squared Error	(mse)
Calculations:	MEX	

Progress

Epoch:	0	1000 iterations	1000
Time:		0:00:02	
Performance:	0.200	3.35e-07	0.00
Gradient:	0.349	6.50e-05	1.00e-07
Mu:	1.00e-07	1.00e-09	1.00e+10
Validation Checks:	0	659	1000

Plots

Performance (plotperform)

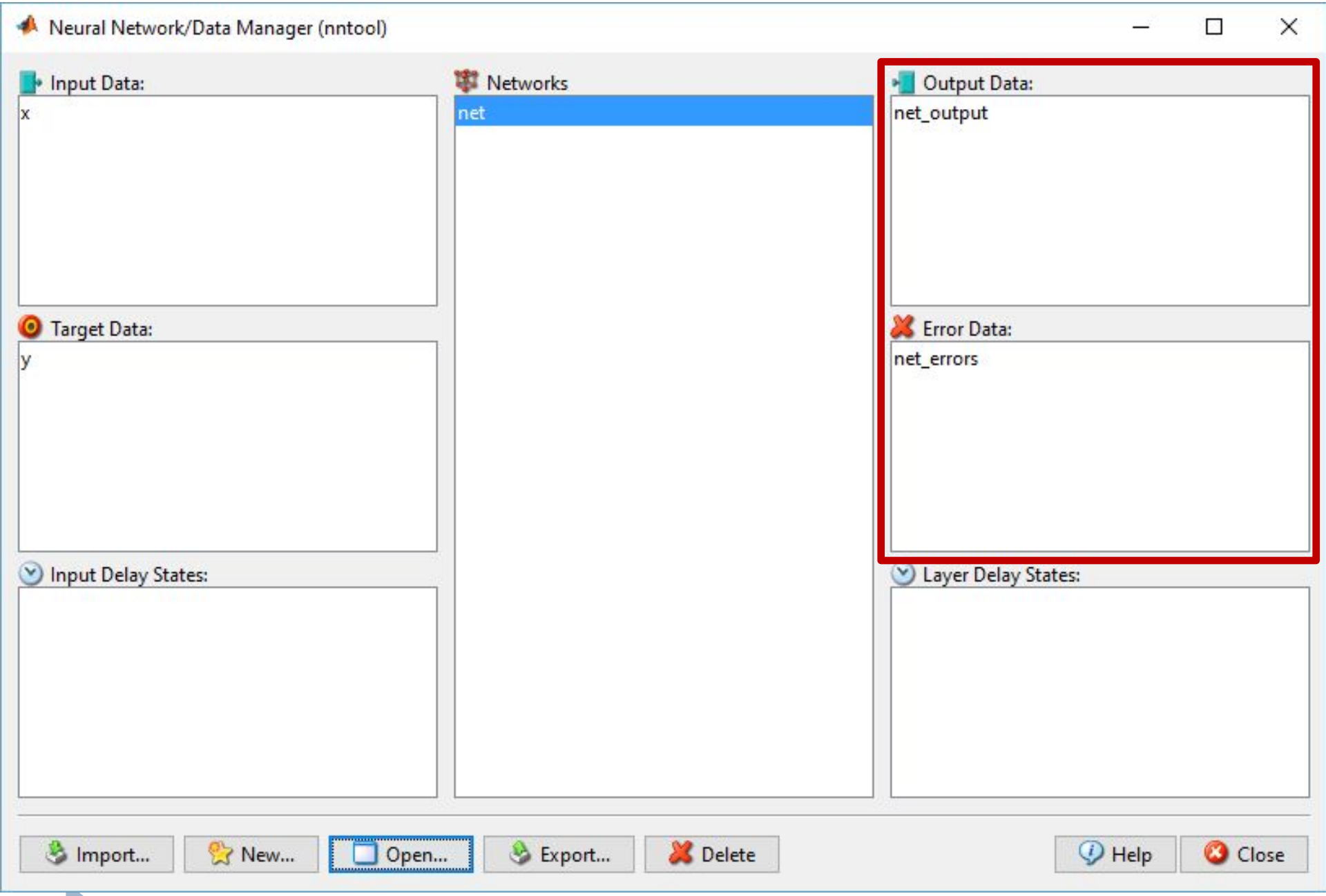
Training State (plottrainstate)

Regression (plotregression)

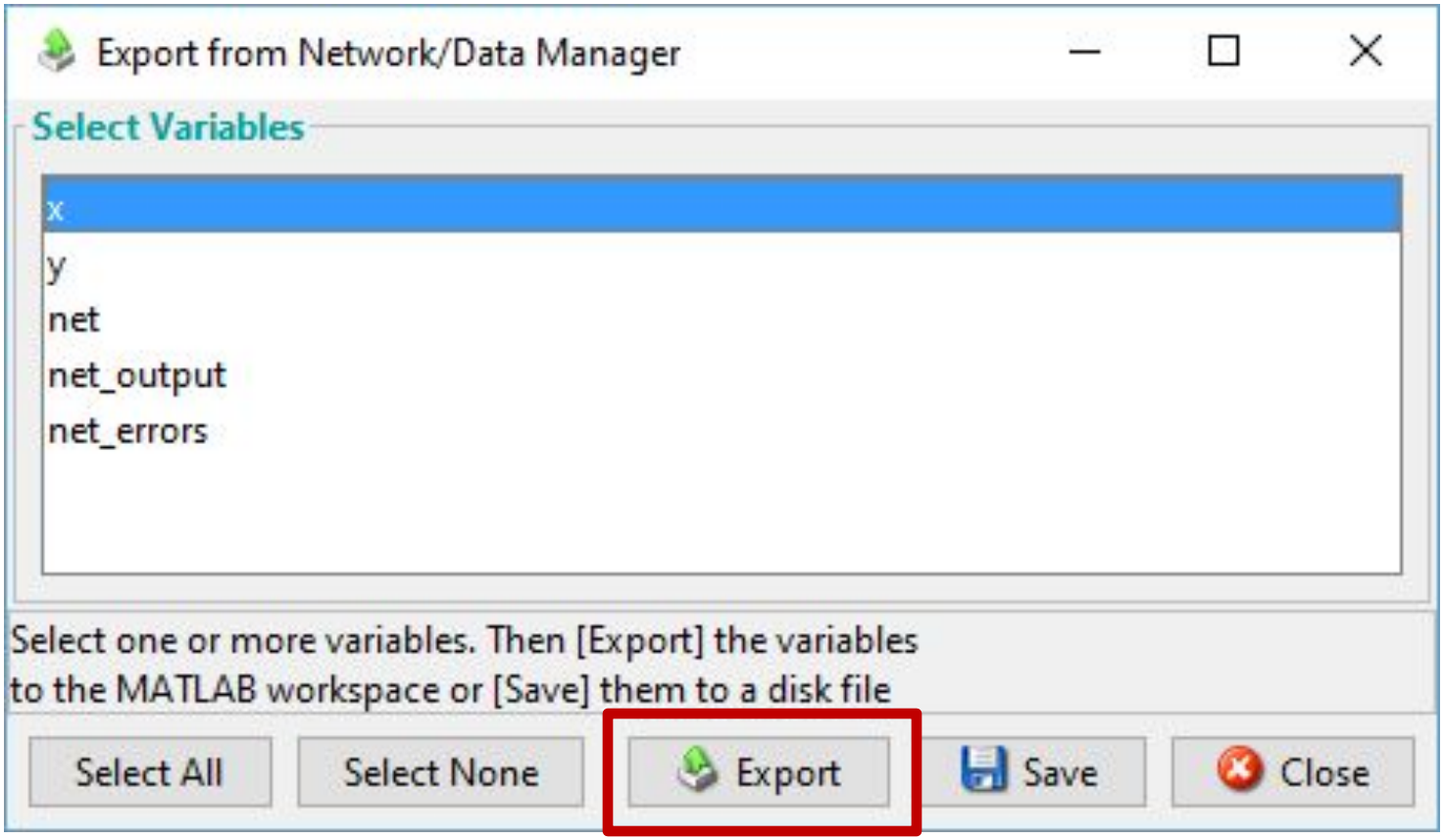
Plot Interval: 1 epochs

Maximum epoch reached.

Stop Training Cancel

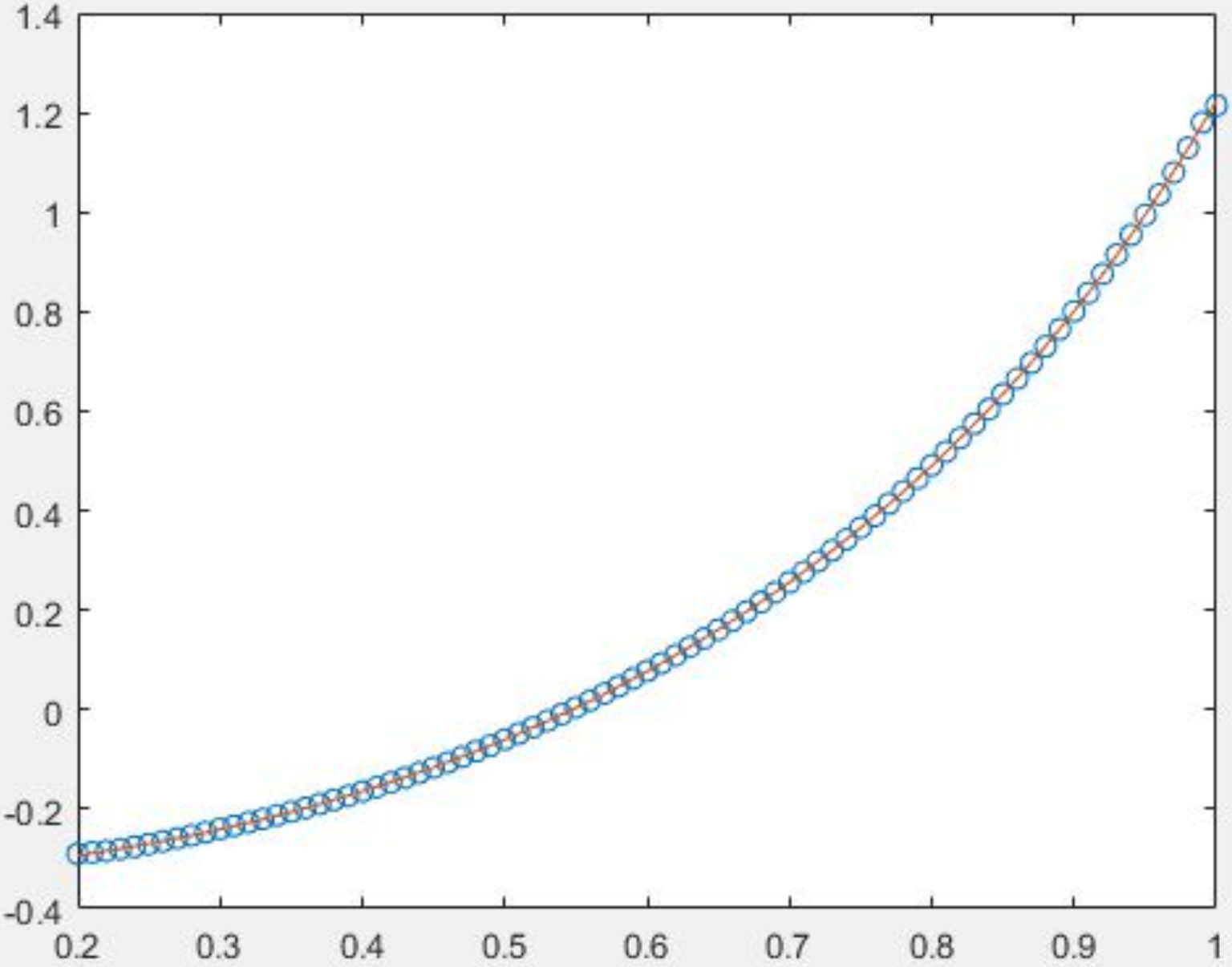

Neural Network/Data Manager (nntool)
Input Data:
x
Target Data:
y
Input Delay States:
Networks
net
Output Data:
net_output
Error Data:
net_errors
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close

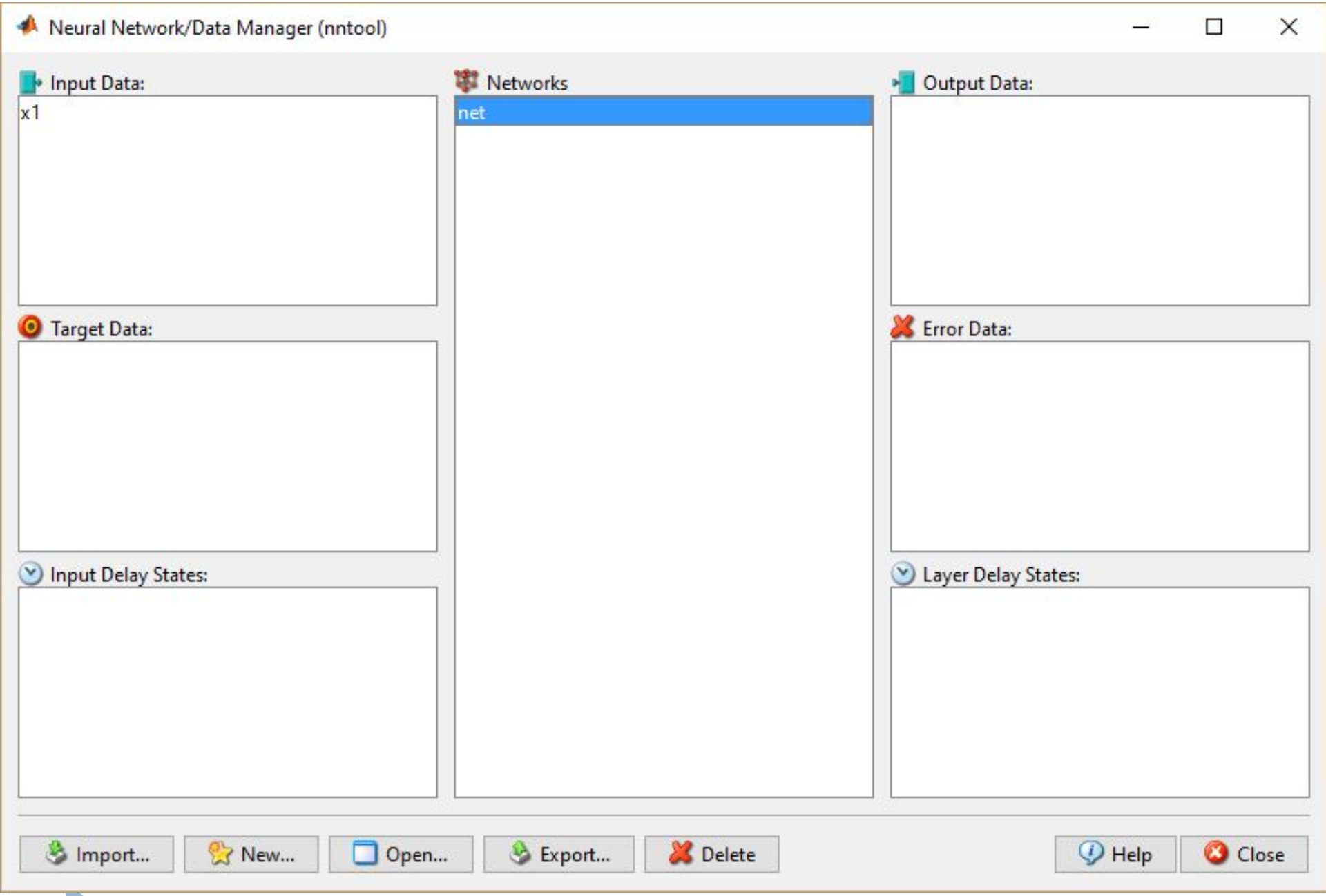
Работа в пакете MATLAB. Экспорт данных.



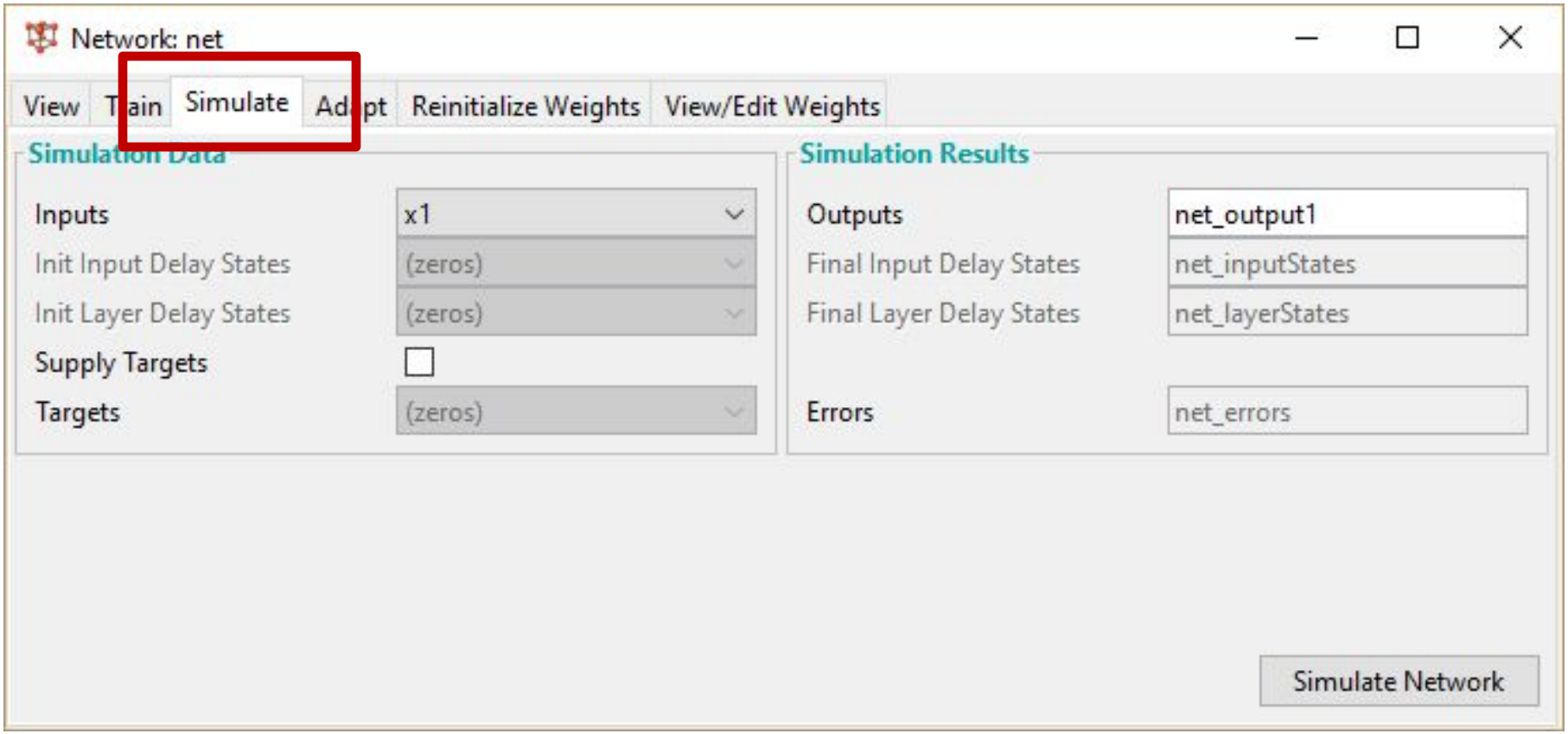
Работа в пакете MATLAB. Построение графиков.

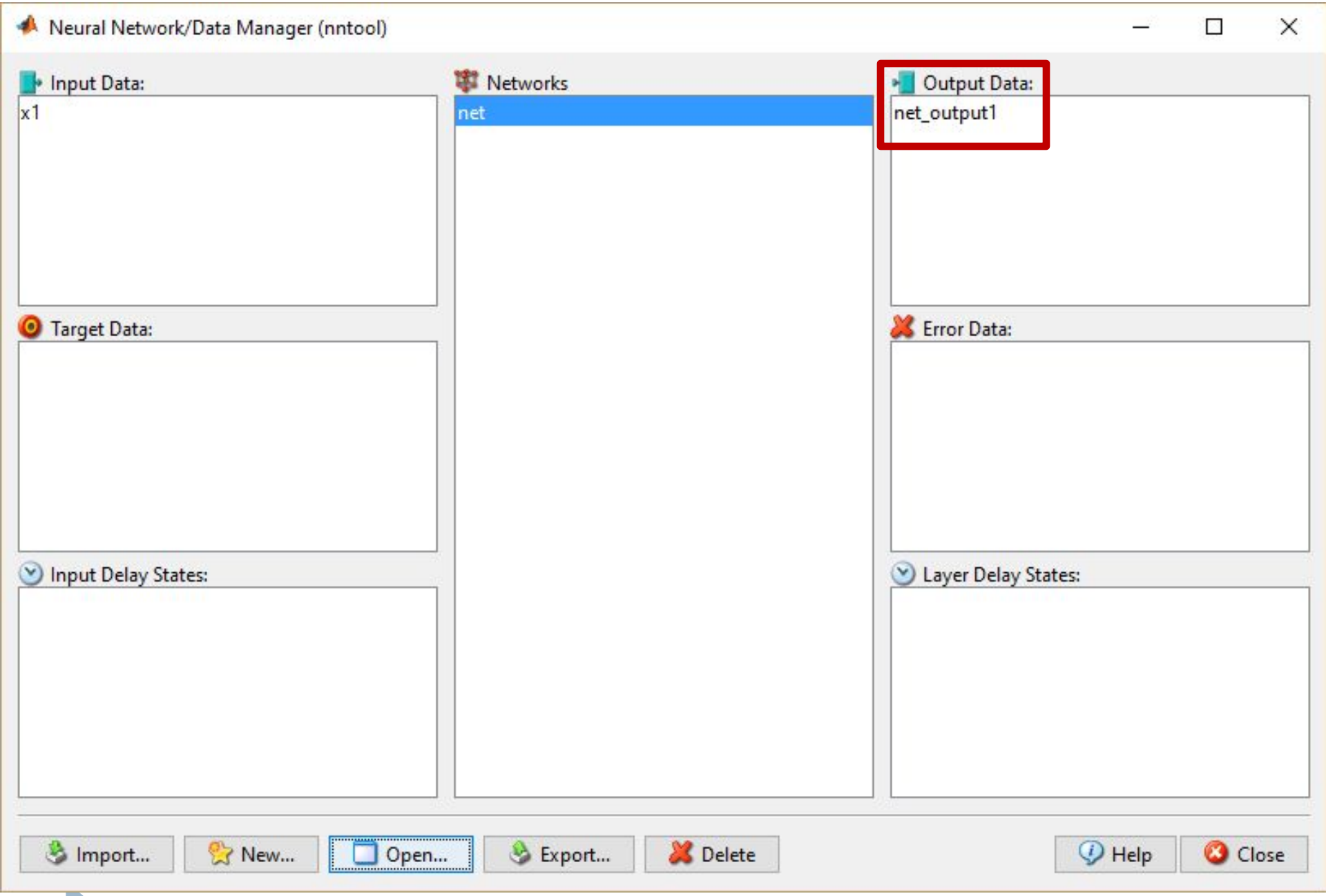
- plot (x, y);
- hold on;
- plot (x, net_output, ‘o’);


1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

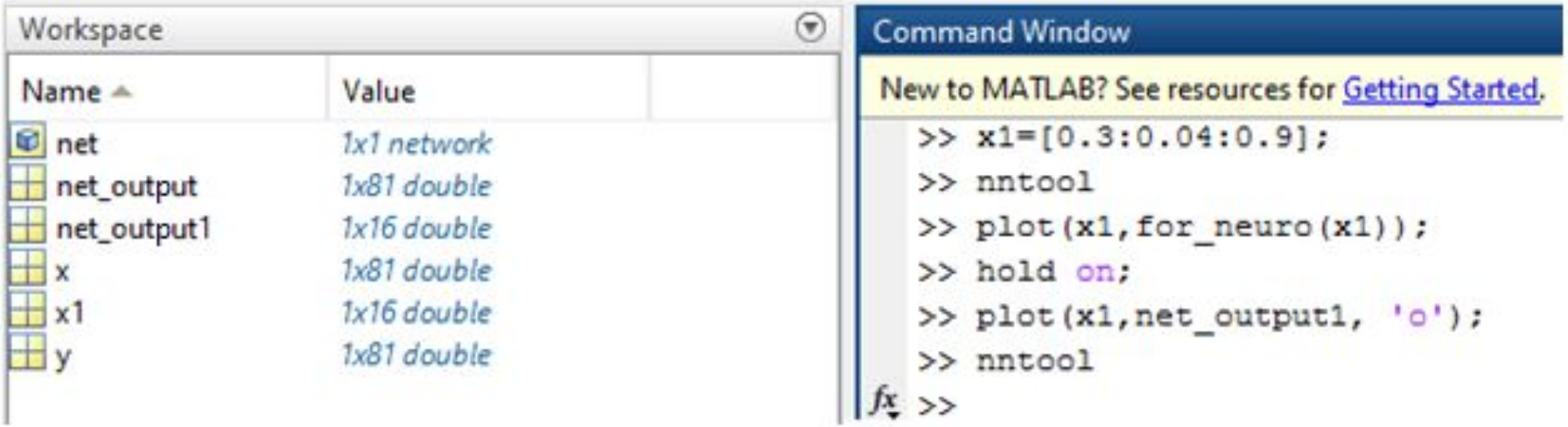

Neural Network/Data Manager (nntool)
Input Data:
x1
Networks
net
Output Data:
Target Data:
Error Data:
Input Delay States:
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close

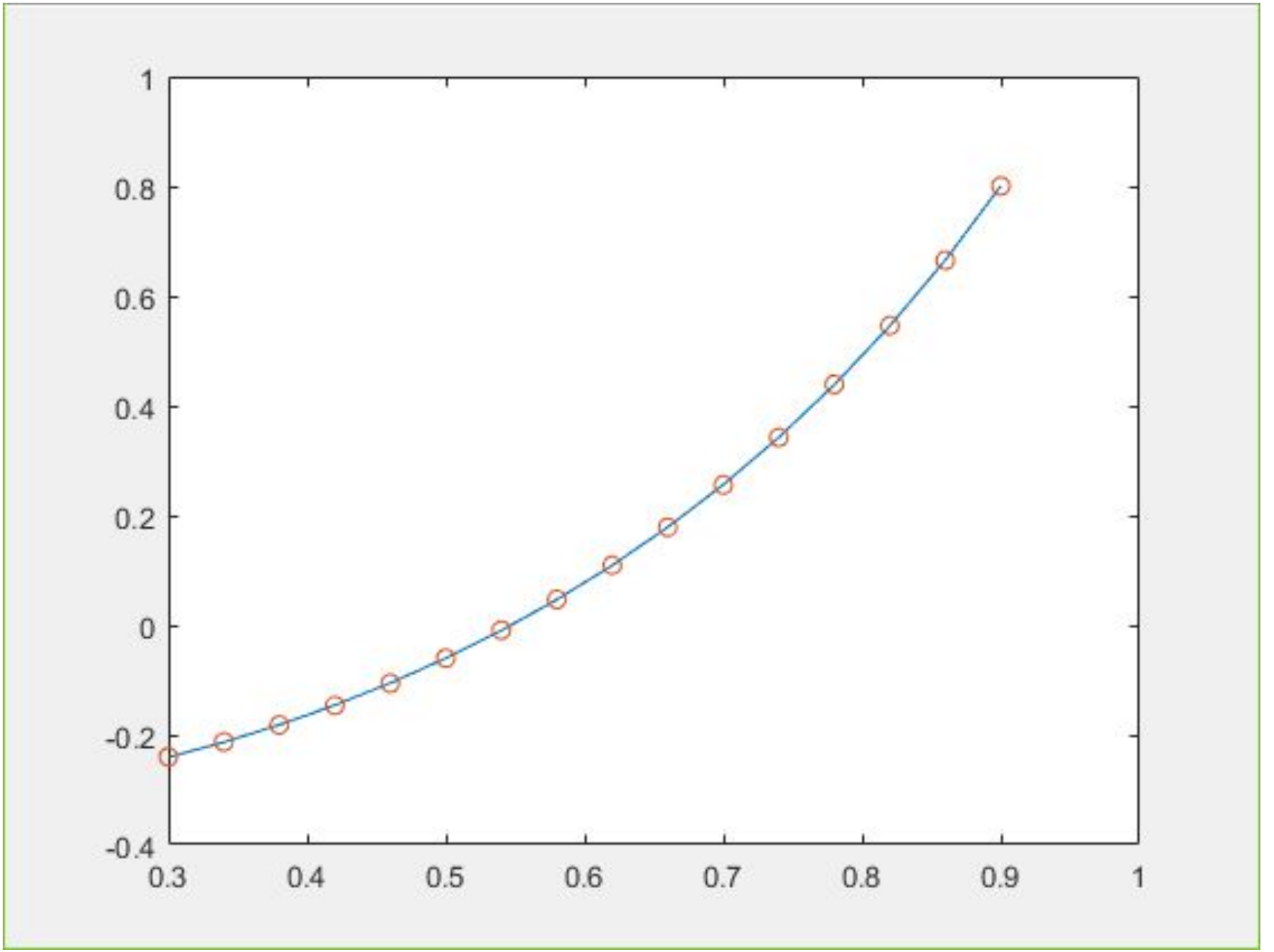
Работа в пакете MATLAB. Проверка нейронной сети.




Neural Network/Data Manager (nntool)
Input Data:
x1
Networks
net
Output Data:
net_output1
Target Data:
Error Data:
Input Delay States:
Layer Delay States:
Import...
New...
Open...
Export...
Delete
Help
Close

Работа в пакете MATLAB. Проверка нейронной сети.




1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1