```
% Define number of inputs/outputs
n_inp = 2;
n_hn = 2;
n_out = 1;
```

```
% Define Activations
syms logsig(x)
logsig(x) = 1 ./ (1 + exp(-x));

syms tansig(x)
tansig(x) = 2 ./ (1 + exp(-2*x)) - 1;

syms sca_le(x)
sca_le(x) = (x - 0).* 2 + -1;
```

XOR/AND Network

```
X  = sym('X',      [n_inp 1],    'real');

IW = sym('IW',     [n_hn n_inp], 'real');
BI = sym('BI',     [n_inp 1],    'real');

LW = sym('LW',     [n_out n_hn], 'real');
BL = sym('BL',     [n_out 1],    'real');

tan_sig = tansig(IW*sca_le(X)+BI);
log_sig = logsig(LW*tan_sig+BL);

T = sym('T',  [n_out 1],         'real');
E = (log_sig - T).^2;
```

```
% Add Allias (names)
LOGSIG = sym('LOGSIG',  [n_out 1], 'real');
TANSIG = sym('TANSIG',  [n_hn  1], 'real');
X_IN  = sym('X_IN',     [n_inp 1], 'real');
T_OUT = sym('X_OUT',    [n_out 1], 'real');
```

```
% Estimate Jacobean Matrices (Signal Approximation Neural Network)
[JIW] = cell(n_hn,1);
vars = {log_sig, tan_sig, X, T};
alias = {LOGSIG, TANSIG, X_IN, T_OUT};

for i = 1:n_hn
    JIW{i} = sub_vars(jacobian(E, IW(:,i)'),...
                      vars, alias);
end
disp('JIW[1]'); JIW{1}
```

```
JIW[1]
ans =
```

$$\left(\frac{4\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_1\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TANSIG}_1-\mathrm{LW}_2\,\mathrm{TANSIG}_2}\,\sigma_2\,(4\,X_{\mathrm{IN1}}-2)\,(\mathrm{LOGSIG}_1-X_{\mathrm{OUT1}})}{(\sigma_2+1)^2}\quad\frac{4\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_2\,\mathrm{e}}{}\right.$$

where

$$\sigma_1 = \mathrm{e}^{-2\,\mathrm{BI}_2-2\,\mathrm{IW}_{2,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{2,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

$$\sigma_2 = \mathrm{e}^{-2\,\mathrm{BI}_1-2\,\mathrm{IW}_{1,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{1,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

```
disp('JIW[2]'); JIW{2}
```

```
JIW[2]
ans =
```

$$\left(\frac{4\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_1\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TANSIG}_1-\mathrm{LW}_2\,\mathrm{TANSIG}_2}\,\sigma_2\,(4\,X_{\mathrm{IN2}}-2)\,(\mathrm{LOGSIG}_1-X_{\mathrm{OUT1}})}{(\sigma_2+1)^2}\quad\frac{4\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_2\,\mathrm{e}}{}\right.$$

where

$$\sigma_1 = \mathrm{e}^{-2\,\mathrm{BI}_2-2\,\mathrm{IW}_{2,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{2,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

$$\sigma_2 = \mathrm{e}^{-2\,\mathrm{BI}_1-2\,\mathrm{IW}_{1,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{1,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

```

disp('JBI')
```

```
JBI
```

```
JBI = sub_vars(jacobian(E, BI'),...
           vars, alias)
```

```
JBI =
```

$$\left(\frac{8\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_1\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TANSIG}_1-\mathrm{LW}_2\,\mathrm{TANSIG}_2}\,\sigma_2\,(\mathrm{LOGSIG}_1-X_{\mathrm{OUT1}})}{(\sigma_2+1)^2}\quad\frac{8\,\mathrm{LOGSIG}_1{}^2\,\mathrm{LW}_2\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TAN}}}{}\right.$$

where

$$\sigma_1 = \mathrm{e}^{-2\,\mathrm{BI}_2-2\,\mathrm{IW}_{2,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{2,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

$$\sigma_2 = \mathrm{e}^{-2\,\mathrm{BI}_1-2\,\mathrm{IW}_{1,1}\,(2\,X_{\mathrm{IN1}}-1)-2\,\mathrm{IW}_{1,2}\,(2\,X_{\mathrm{IN2}}-1)}$$

```

[JLW] = cell(n_out,1);
for i = 1:n_out
    JLW{i} = sub_vars(jacobian(E, LW(:,i)'),...
                      vars, alias);
```

```
end
disp('JLW')
```

```
JLW
```

```
JLW{1}
```

ans = $2\,\mathrm{LOGSIG}_1{}^2\,\mathrm{TANSIG}_1\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TANSIG}_1-\mathrm{LW}_2\,\mathrm{TANSIG}_2}\,(\mathrm{LOGSIG}_1-X_{\mathrm{OUT1}})$

```
JBL = sub_vars(jacobian(E, BL'),...
               vars, alias)
```

JBL = $2\,\mathrm{LOGSIG}_1{}^2\,\mathrm{e}^{-\mathrm{BL}_1-\mathrm{LW}_1\,\mathrm{TANSIG}_1-\mathrm{LW}_2\,\mathrm{TANSIG}_2}\,(\mathrm{LOGSIG}_1-X_{\mathrm{OUT1}})$

```
disp('JBL')
```

```
JBL
```