Universidade de São Paulo
Escola de Engenharia de São Carlos
Departamento de Engenharia Elétrica

**Alunos:** Luisa Helena Bartocci Liboni
Rodrigo de Toledo Caropreso

**Data de Entrega**: 18/06/2012

# Redes Neurais Artificiais

**(Prof. Ivan Nunes da Silva)**

## EPC-11

A previsão de potência elétrica é de suma importância para o planejamento operacional de um sistema de energia elétrica. A programação da manutenção do sistema, o planejamento de sua expansão e a análise de despacho é normalmente executado levando-se em consideração a potência elétrica prevista para o respectivo sistema.

A previsão para um dia específico pode ser efetuada levando-se em conta apenas a potência medida nas primeiras horas do dia. Baseado nessas informações, por intermédio da classificação de curvas, é possível traçar o perfil da potência que será necessária para todas as outras horas do dia.

Na tabela abaixo fornece-se a potência medida em 16 dias e que foram agrupadas em 4 classes (perfis) de demanda.

| Amostra | 7 horas | 8 horas | 9 horas | 10 horas | 11 horas | 12 horas | Classe |
|---|---|---|---|---|---|---|---|
| 1 | 2.3976 | 1.5328 | 1.9044 | 1.1937 | 2.4184 | 1.8649 | 1 |
| 2 | 2.3936 | 1.4804 | 1.9907 | 1.2732 | 2.2719 | 1.8110 | 1 |
| 3 | 2.2880 | 1.4585 | 1.9867 | 1.2451 | 2.3389 | 1.8099 | 1 |
| 4 | 2.2904 | 1.4766 | 1.8876 | 1.2706 | 2.2966 | 1.7744 | 1 |
| 5 | 1.1201 | 0.0587 | 1.3154 | 5.3783 | 3.1849 | 2.4276 | 2 |
| 6 | 0.9913 | 0.1524 | 1.2700 | 5.3808 | 3.0714 | 2.3331 | 2 |
| 7 | 1.0915 | 0.1881 | 1.1387 | 5.3701 | 3.2561 | 2.3383 | 2 |
| 8 | 1.0535 | 0.1229 | 1.2743 | 5.3226 | 3.0950 | 2.3193 | 2 |
| 9 | 1.4871 | 2.3448 | 0.9918 | 2.3160 | 1.6783 | 5.0850 | 3 |
| 10 | 1.3312 | 2.2553 | 0.9618 | 2.4702 | 1.7272 | 5.0645 | 3 |
| 11 | 1.3646 | 2.2945 | 1.0562 | 2.4763 | 1.8051 | 5.1470 | 3 |
| 12 | 1.4392 | 2.2296 | 1.1278 | 2.4230 | 1.7259 | 5.0876 | 3 |
| 13 | 2.9364 | 1.5233 | 4.6109 | 1.3160 | 4.2700 | 6.8749 | 4 |
| 14 | 2.9034 | 1.4640 | 4.6061 | 1.4598 | 4.2912 | 6.9142 | 4 |
| 15 | 3.0181 | 1.4918 | 4.7051 | 1.3521 | 4.2623 | 6.7966 | 4 |
| 16 | 2.9374 | 1.4896 | 4.7219 | 1.3977 | 4.1863 | 6.8336 | 4 |

Assim, implementar e treinar uma LVQ-1 que detecte as possíveis similaridades e regularidades entre todos os vetores pertencentes a cada uma das classes, objetivando-se a classificação futura de perfis de potência de outros dias (amostras). Considere para as simulações uma taxa de aprendizagem $\eta = 0.05$.

Após o treinamento da rede, utilize-a para a classificação do perfil de potência para os dias apresentados na tabela a seguir.

**Dados de Teste**

| Dia | 7 horas | 8 horas | 9 horas | 10 horas | 11 horas | 12 horas | Classe |
|---|---|---|---|---|---|---|---|
| 1 | 2.9817 | 1.5656 | 4.8391 | 1.4311 | 4.1916 | 6.9718 | 4 |
| 2 | 1.5537 | 2.2615 | 1.3169 | 2.5873 | 1.7570 | 5.0958 | 3 |
| 3 | 1.2240 | 0.2445 | 1.3595 | 5.4192 | 3.2027 | 2.5675 | 2 |
| 4 | 2.5828 | 1.5146 | 2.1119 | 1.2859 | 2.3414 | 1.8695 | 1 |
| 5 | 2.4168 | 1.4857 | 1.8959 | 1.3013 | 2.4500 | 1.7868 | 1 |
| 6 | 1.0604 | 0.2276 | 1.2806 | 5.4732 | 3.2133 | 2.4839 | 2 |
| 7 | 1.5246 | 2.4254 | 1.1353 | 2.5325 | 1.7569 | 5.2640 | 3 |
| 8 | 3.0565 | 1.6259 | 4.7743 | 1.3654 | 4.2904 | 6.9808 | 4 |

**Código -Fonte**

```
clc;
clear;

Carrega_Tabela_Treino;

eta = 0.05;
x0 = [DB_X1 DB_X2 DB_X3 DB_X4 DB_X5 DB_X6]';
d  = [DB_D]'; %saidas para treinamento supervisionado
N1 = 4; %neuronios (na LVQ o numero de neurons == quantidade de classes)
N  = 6; %entradas
epson = 1e-4;
max_epocas = 500;
N_Amostras = length(DB_X1);

%Normaliza o vetor de entrada
for k=1:N_Amostras
       x( :, k ) = x0( :, k ) / norm(x0( :, k ));
end;

%Inicializacao da matriz de pesos (cada vetor ou neuron deve ser associado
%a uma classe de dados)
w( :, 1 ) = x( :, 1 );
w( :, 2 ) = x( :, 5 );
w( :, 3 ) = x( :, 9 );
w( :, 4 ) = x( :, 13 );

%Treinamento
epoca = 0;
stop = 0;

w_anterior = w;
maior_mudanca_anterior = 0;

while(~stop)
    for k=1:N_Amostras
        for j = 1: N1
                                D(j, k) = sqrt( sum( ( x( :, k ) - w( :, j ) ).^2 ) );
        end;

            [value, winner] = min(D(:, k)); %menor distancia, determina o vencedor

        %Atualiza pesos do vencedor
```

```
        if( winner == d(k) ) %vencedor pertence a classe do treinamento -> aproxima ele do cluster
                w( :, winner ) = w( :, winner ) + eta * ( x( :, k ) -  w( :, winner ) );
         else %vencedor NAO pertence a classe do treinamento -> afasta ele do cluster
                w( :, winner ) = w( :, winner ) - eta * ( x( :, k ) -  w( :, winner ) );
        end;

        %Normaliza
         w( :, winner ) = w( :, winner ) / norm( w( :, winner ) );
    end;
    epoca = epoca + 1;

    %Criterio de parada
%               maior_mudanca_atual    = max(max(abs(w-w_anterior)));

    w_change = w-w_anterior;
    for hh=1:size(w_change,2)
        norma(hh) = norm( w_change( :, hh ) ); %calcula a norma de cada coluna
    end;

                maior_mudanca_atual    = max(   norma   );     %norm(max(abs((w-w_anterior)), [],2));
     maior_mudanca_atual
     if( maior_mudanca_atual < epson ) %||     (     maior_mudanca_atual     - maior_mudanca_anterior ) < epson )
         stop = 1;
     end;
%
%            maior_mudanca_anterior   = maior_mudanca_atual;

    w_anterior = w;
end;

disp 'Fim do Treinamento: Numero de Epocas'
epoca
pause

%OPERACAO
Carrega_Tabela_Operacao;
xOp = [DB_X1 DB_X2 DB_X3 DB_X4 DB_X5 DB_X6]';

N_Amostras = length(DB_X1);

%Normaliza o vetor de entrada
for k=1:N_Amostras
       x( :, k ) = xOp( :, k ) / norm(xOp( :, k ));
end;

%Verificação
for k=1:N_Amostras
    for j = 1: N1
        D(j, k) = sqrt( sum( ( x( :, k ) - w( :, j ) ).^2 ) );
    end;

      [value, winner] = min(D(:, k)); %menor distancia, determina o vencedor

     disp( sprintf( 'Neuronio vencedor (classe): %d', winner ) );
end;

[N_Amostra  DB_X1  DB_X2  DB_X3  DB_X4  DB_X5       DB_X6       DB_D]       = textread(  'Dados_Treino.dat',  '%d  %f  %f  %f  %f  %f  %d', 'headerlines', 1);

[N_Amostra  DB_X1  DB_X2  DB_X3  DB_X4  DB_X5            DB_X6]            = textread(  'Dados_Operacao.dat', '%d %f  %f  %f  %f  %f  %f', 'headerlines', 1);
```