```
function [J,grad] = costeRN(params_rn,num_entradas,num_ocultas,...
    num_etiquetas,X,y,lambda)
    %costeRN calcula el coste y el gradiente de una red neuronal de
 dos
    %capas

    Theta1 = reshape(params_rn(1:num_ocultas*(num_entradas+1)),...
      num_ocultas,(num_entradas+1));
    Theta2 = reshape(params_rn((1 + (num_ocultas * (num_entradas +
 1))):end), ...
                num_etiquetas, (num_ocultas + 1));

  %el numero de ejemplos de entrenamiento
    m = size(X,1);
    J = 0;
    Theta1_grad = zeros(size(Theta1));
    Theta2_grad = zeros(size(Theta2));


    a1 = [ones(m, 1) X];

    z2 = a1 * Theta1';
    a2 = sigmoide(z2);
    a2 = [ones(size(a2,1), 1) a2];

    z3 = a2 * Theta2';
    a3 = sigmoide(z3);
    hThetaX = a3;

    yVec = zeros(m,num_etiquetas);

    for i = 1:m
        yVec(i,y(i)) = 1;
    end

    J = 1/m * sum(sum(-1 * yVec .* log(hThetaX)-(1-yVec) .* log(1-
hThetaX)));

    regularator = (sum(sum(Theta1(:,2:end).^2)) +
 sum(sum(Theta2(:,2:end).^2))) * (lambda/(2*m));

    J = J + regularator;

    for t = 1:m


        a1 = [1; X(t,:)'];
        z2 = Theta1 * a1;
        a2 = [1; sigmoide(z2)];

        z3 = Theta2 * a2;
        a3 = sigmoide(z3);
```

```
        yy = ([1:num_etiquetas]==y(t))';

        delta_3 = a3 - yy;

        delta_2 = (Theta2' * delta_3) .* [1; sigmoideGradiente(z2)];
        delta_2 = delta_2(2:end);


        Theta1_grad = Theta1_grad + delta_2 * a1';
        Theta2_grad = Theta2_grad + delta_3 * a2';
    end

    Theta1_grad = (1/m) * Theta1_grad + (lambda/m) *
 [zeros(size(Theta1, 1), 1) Theta1(:,2:end)];
    Theta2_grad = (1/m) * Theta2_grad + (lambda/m) *
 [zeros(size(Theta2, 1), 1) Theta2(:,2:end)];



    grad = [Theta1_grad(:) ; Theta2_grad(:)];


end
```

```
Not enough input arguments.

Error in costeRN (line 6)
    Theta1 = reshape(params_rn(1:num_ocultas*(num_entradas+1)),...
```

Published with MATLAB® R2015b