

Contents

- Compare Matlab to our "homemade" RBF:
- Train the RBF using Matlab's built in routines:
- Extract the parameters and re-build the RBF from scratch:
- Compare our outputs:

Compare Matlab to our "homemade" RBF:

```
% Data for training:
P=linspace(-2,2,50);
T=sin(3*P)+0.2*randn(size(P));

% Data for testing
xx=linspace(-2,2);
NumPts=length(xx);
```

Train the RBF using Matlab's built in routines:

```
net=newrb(P,T,0.05,1);
```

NEWRB, neurons = 0, MSE = 0.549582

Extract the parameters and re-build the RBF from scratch:

```
Centers=net.IW{1,1}; %Centers
W=net.LW{2,1}; %Weight matrix
b1=net.b{1}; %Numcenters x 1- This is the scaling factor for the Gaussian
b2=net.b{2}; %Bias vector

%Now compute the network output "by hand":
A=edm(xx',Centers);
A1=A.*repmat(b1',NumPts,1); %Multiply by the scaling factor before computing phi

Phi=rbf1(A1,1,1);
```

Compare our outputs:

```
Yout=W*Phi'+b2;

%Get the output using Matlab's built in routine
Yout2=sim(net,xx);

% You should see Yout=Yout2:
plot(P,T,'k*',xx,Yout,xx,Yout2);

max(abs(Yout-Yout2))
```

ans =

2.8422e-14

