An RBF to try by hand

Homework for Friday

Suppose our model function maps $\mathbb{R}^3$ to $\mathbb{R}$, and suppose we have two centers: $[1, 0, -1]^T, [1, 1, 0]^T$. Given the transfer function $\phi(r) = r^3$, and a set of weights $[1, 2]$ and a bias (constant) of -1, use a calculator to compute the output of the RBF given the point $[-1, 1, 2]^T$ (You may use Matlab as your calculator, but be sure you can perform the necessary operations).

Matlab and the RBF

Here is a quick example. Matlab likes to use P, T for "patterns" and "targets"- For us, these were X and Y, respectively.

```
P=linspace(-2,2,50);
T=sin(3*P)+0.2*randn(size(P));
eg=0.05;  %eg is error goal
sc=1;       %sc is scaling for the RBF
net=newrb(P,T,eg,sc);
```

Network Structure for the RBF

- The transfer function is the Gaussian (we'll need to discuss the actual width used).
- The matrix of centers is located in: `C=net.IW{1,1}` (note the curly braces).

  In this example, the matrix had dimensions 6×1 (that's 6 centers with dimension 1). We
- The scaling of the Gaussian is contained in the vector $\mathbf{b}_1$ =`net.b{1}`

  In this example, the vector is 6×1.

  There is also a constant vector `b2=net.b{2}`

  In this example, this vector is a scalar.

If we track the sequence of steps we perform to transform a set of vectors in the domain, x, into the function output,

```
% Here is some data:
P=linspace(-2,2,50);
T=sin(3*P)+0.2*randn(size(P));

%Train the RBF
net=newrb(P,T,0.05,1);


xx=linspace(-2,2);  %New data in the domain
NumPts=length(xx);  %Used below in computing A1
```

```
%Here are the relevent parameters from the network structure.
Centers=net.IW{1,1};
W=net.LW{2,1};
b1=net.b{1};  %Numcenters x 1- This is the scaling factor for the Gaussian
b2=net.b{2};  %Bias term (See below)

%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);
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);
```