Lab: Math 350, Dec 3

We'll need to download some files from the class website:

- `edm.m`

  Computes the distance matrix between two sets of points organized in two arrays, W (s points in $\mathbb{R}^n$) and P (t points in $\mathbb{R}^n$), and returns the $s \times t$ matrix E:

  ```
  E=edm(W,P);
  ```

- `rbf1.m`

  Given a matrix of distances A (computed from using `edm.m`), returns $\phi(A)$, where ϕ is a nonlinear function chosen from 1 of 6 functions (see the help file or the `rbf1.m` file for the choices). We'll look at some examples in class.

- `edmInterp1.m`

  Performs interpolation using the Euclidean Distance matrix (with $\phi(r) = r$).

- `edmInterp2.m`

  Changes the function to use $\phi(r) = r^3$.

- `rbfError2d.m`

  Uses some of Matlab's built-in functions to produce the RBF regression model.

Recall that in all cases (either the EDM alone or as the RBF), we are given some data,

$$\{(\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), (\mathbf{x}_3, \mathbf{t}_3), \cdots\}$$

and our goal is to build a (nonlinear) function F so that either $F(\mathbf{x}_i) = \mathbf{t}_i$ (the interpolation problem) or F will minimize some error function. Typically, this is the Mean Square Error (MSE):

$$E = \frac{1}{p} \sum_{i=1}^{p} \|F(\mathbf{x}_i) - \mathbf{t}_i\|^2$$

(so E depends on the parameters in F). Think of the function F as being built in layers:

$$\mathbf{x}_i \longrightarrow \begin{bmatrix} \|\mathbf{x}_i - \mathbf{c}_1\| \\ \|\mathbf{x}_i - \mathbf{c}_2\| \\ \vdots \\ \|\mathbf{x}_i - \mathbf{c}_k\| \end{bmatrix} \longrightarrow \begin{bmatrix} \phi(\|\mathbf{x}_i - \mathbf{c}_1\|) \\ \phi(\|\mathbf{x}_i - \mathbf{c}_2\|) \\ \vdots \\ \phi(\|\mathbf{x}_i - \mathbf{c}_k\|) \end{bmatrix} = \Phi \longrightarrow W\Phi = Y$$

The first layer is implemented by `edm.m`, the second layer is implemented by `rbf1.m`, and the third layer is either matrix multiplication (if W is known) or the pseudoinverse (to compute W using Φ and Y).

Questions

Before working through these questions, run `edmInterp1.m` and `edmInterp2.m`. The goal of today's lab is to both understand the mathematics of the RBF, and be able to run (and use) the code that we'll download from the class website.

1. Run `rbfError2d.m`, then modify it so that $\phi(r) = \frac{1}{1+r}$.

2. (Revised version of Exercise 1, p. 183) Find the radial basis function F with the following specifications:

   - 5 centers randomly chosen in the plane, like `C=2*randn(5,2);`
   - Use a Gaussian function with $\sigma = 1$

   Find the weights (α's) so that we minimize the mean square error on the data in X and Y below (arranged so that `edm` and `rbf1.m` work without change):

   $$X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ -1 & 0 \end{bmatrix} \qquad Y = \begin{bmatrix} 2 & 1 & -1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

   so that F takes each row of X and maps it to the corresponding row of Y. *Side Note 1:* What are you solving for (and give the size). *Side Note 2:* Before you solve this, would you expect to get an error, or will the error probably be zero?

   Have your script file output the function using the rows of X to verify your earlier guess.

3. Using the same data, look at the help file for `newrb` then do the example listed,

   ```
   P=[1 2 3];
   T=[2.0 4.1 5.9];
   net=newrb(P,T);
   tt=linspace(1,3);
   yy=sim(net,tt);
   plot(P,T,'k*',tt,yy);
   ```

   Find the weights and the centers inside the Matlab data structure `net`, and verify your answer by constructing the function yourself and plotting the result (We'll discuss more later).