Radial Basis Functions
Radial Basis Functions (RBFs)

\[ Z = (\text{list of radial basis function evaluations}) \]

\[ \beta = (Z^T Z)^{-1} (Z^T y) \]

\[ y^{\text{est}} = \beta_0 + \beta_1 x_1 + \ldots \]
1-d RBFs

\[ y^{est} = \beta_1 \phi_1(x) + \beta_2 \phi_2(x) + \beta_3 \phi_3(x) \]

where

\[ \phi_i(x) = \text{KernelFunction}( | x - c_i | / KW) \]
Example

\[ y^{est} = 2\phi_1(x) + 0.05\phi_2(x) + 0.5\phi_3(x) \]

where

\[ \phi_i(x) = \text{KernelFunction}( | x - c_i | / KW) \]
RBFs with Linear Regression

\[ y_{\text{est}} = 2\phi_1(x) + 0.05\phi_2(x) + 0.5\phi_3(x) \]

where \( \phi_i(x) = \text{KernelFunction}(\|x - c_i\| / KW) \)

All \( c_i \)'s are held constant (initialized randomly or on a grid in m-dimensional input space)

\( KW \) also held constant (initialized to be large enough that there's decent overlap between basis functions*

*Usually much better than the crappy overlap on my diagram
RBFs with Linear Regression

\[ y_{\text{est}} = 2\phi_1(x) + 0.05\phi_2(x) + 0.5\phi_3(x) \]

where

\[ \phi_i(x) = \text{KernelFunction}(\| x - c_i \| / KW) \]

then given \( Q \) basis functions, define the matrix \( Z \) such that

\[ Z_{kj} = \text{KernelFunction}(\| x_k - c_i \| / KW) \]

where \( x_k \) is the kth vector of inputs

And as before, \( \beta = (Z^TZ)^{-1}(Z^Ty) \)

All \( c_i \)'s are held constant (initialized randomly or on a grid in m-dimensional input space)

\( KW \) also held constant (initialized to be large enough that there's decent overlap between basis functions

*Usually much better than the crappy overlap on my diagram
RBFs with NonLinear Regression

Allow the $c_i$'s to adapt to the data (initialized randomly or on a grid in m-dimensional input space)

$y_{est} = 2 \phi_1(x) + 0.05 \phi_2(x) + 0.5 \phi_3(x)$

where

$\phi_i(x) = \text{Kernel Function}(|x - c_i| / KW)$

$KW$ allowed to adapt to the data. (Some folks even let each basis function have its own $KW_i$, permitting fine detail in dense regions of input space)

But how do we now find all the $\beta_j$'s, $c_i$'s and $KW$?
RBFs with NonLinear Regression

Allow the $c_i$'s to adapt to the data (initialized randomly or on a grid in m-dimensional input space)

$y_{est} = 2\phi_1(x) + 0.05\phi_2(x) + 0.5\phi_3(x)$

where

$\phi_i(x) = \text{KernelFunction}( \left| \frac{x - c_i}{KW} \right| )$

But how do we now find all the $\beta_j$'s, $c_i$'s and $KW$?

$KW$ allowed to adapt to the data. (Some folks even let each basis function have its own $KW_j$, permitting fine detail in dense regions of input space)

Answer: Gradient Descent
RBFs with NonLinear Regression

Allow the $c_i$'s to adapt to the data (initialized randomly or on a grid in m-dimensional input space)

$y_{est} = 2\phi_1(x) + 0.05\phi_2(x) + 0.5\phi_3(x)$

where

$\phi_i(x) = \text{KernelFunction}\left( \frac{|x - c_i|}{KW} \right)$

But how do we now find all the $\beta_j$'s, $c_i$'s and $KW$?

$KW$ allowed to adapt to the data. (Some folks even let each basis function have its own $KW_j$, permitting fine detail in dense regions of input space)

Answer: Gradient Descent

(But I’d like to see, or hope someone’s already done, a hybrid, where the $c_i$'s and $KW$ are updated with gradient descent while the $\beta_j$'s use matrix inversion)
Radial Basis Functions in 2-d

Two inputs.
Outputs (heights sticking out of page) not shown.
Blue dots denote coordinates of input vectors

Sphere of significant influence of center

Happy RBFs in 2-d
Crabby RBFs in 2-d

What’s the problem in this example?

Blue dots denote coordinates of input vectors.

Center

Sphere of significant influence of center.
More crabby RBFs

Blue dots denote coordinates of input vectors

Center

Sphere of significant influence of center

And what’s the problem in this example?
Hopeless!

Even before seeing the data, you should understand that this is a disaster!

Center

Sphere of significant influence of center
Unhappy

Even before seeing the data, you should understand that this isn’t good either..

Sphere of significant influence of center

Center
So what do we do?
So what do we do?

Search to find the optimal size “width” for the Gaussians (on a test set, of course!)
What we have seen

• One can make nonlinear relations linear by transforming the features
  • Polynomial regression
  • Radial Basis Functions (RBF)
  • Kernel regression (more on this later)