SysEng 5212 /EE 5370
Introduction to Neural Networks and Applications

Lecture 8: Radial Basis Functions II and Regularization Theory

Cihan H Dagli, PhD

Professor of Engineering Management and Systems Engineering
Professor of Electrical and Computer Engineering
Founder and Director of Systems Engineering Graduate Program

dagli@mst..edu

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY
Rolla, Missouri, U.S.A.

SYSTEMS ENGINEERING
Research Center
Department of Defense UARC

Smart Engineering Systems Lab
Engineering Management and Systems Engineering Department

Lecture outline

- Exam Review
- Engineering Project Progress
- Interpolation Problem
- RBFN Solution
- Interpolation Matrices
- Deriving RBFN Learning
- Regularization Theory
- Generalized RBFN Example
- RBF Learning Approaches

Engineering Problem Volunteers Tally

- Murat Aslan
- Tatiana Cardona Sepulveda
- Prince Codjoe
- Xiongming Dai
- Jeffrey Dierker

Engineering Problem Volunteers Tally

- Venkata Sai Abhishek Dwivadula
- Brian Guenther
- Anthony Guertin
- Timothy Guertin
- Seth Kitchen

Engineering Problem Volunteers Tally

- Gregory Leach
- Yu Li
- John Nganga
- Igor Povarich
- Jack Savage

Engineering Problem Volunteers Tally

- William Symolon
- Wayne Viers III
- Tao Wang
- Kari Ward
- Julia White
- Jun Xu

Cover's Theorem Recap

1. Cover's theorem states that a nonlinearly separable pattern classification problem can be solved by mapping the input space into a new space of higher dimension.

2. A nonlinear mapping is used to transform the pattern classification problem into a linearly separable one.

MATLAB: y = purelin(v)

Input-Output Mapping as a Hypersurface

- Consider a feedforward network with a single hidden layer designed to,
  - perform a nonlinear mapping from the input space to the hidden space
  - followed by a linear mapping from the hidden space to the output space
  - the network represents a map from an m0-dimensional input space to a m2-dimensional output space.
    - $s : R^{m0} \rightarrow R^{m2}$
- The input-output mapping can be considered a multidimension plot (*hypersurface*) of the output as a function of the input.

Phases of the Learning Process

- Training phase: We use training data in the form of input-output patterns to fit the hypersurface. Training is the process of optimizing this fitting procedure to find the optimum approximation to the true surface.
- Generalization phase: Find the output for unknown inputs by interpolating between the data points along the fitted surface.
- The interpolation problem can be mathematically stated,

Given a set of N points $\{x_i \in \mathbb{R}^{m_0} | i = 1, 2, \ldots, N\}$ and a corresponding set of N real numbers $\{d_i \in \mathbb{R}^{m_2} | i = 1, 2, \ldots, N\}$, find a function $F : \mathbb{R}^{m_0} \rightarrow \mathbb{R}^{m_2}$, that satisfies the condition,

$$F(x_i) = d_i, \;\; i = 1, 2, \ldots, N$$

Without loss of generality, we can assume $m_2 = 1$.

Using RBF Network to Find the Interpolating Surface

- For RBF networks the interpolating surface i.e., function F is given by,

$$F(\mathbf{x}) = \sum_{i=1}^{N} w_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) \tag{1}$$

  where $\{\phi(\|\mathbf{x} - \mathbf{x}_i\|) | i = 1, 2, \ldots, N\}$ is a matrix of nonlinear functions known as radial-basis functions
- $\|\cdot\|$ is the Euclidean norm
- $x_i \in \mathbb{R}^{m_0}, i = 1, 2, \ldots, N$ are the centers of the RBFs

Solution of the RBFN

To solve for the weights, we expand equation (1),

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \dots & \phi_{1N} \\ \phi_{21} & \phi_{22} & \dots & \phi_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N1} & \phi_{N2} & \dots & \phi_{NN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix}$$

where $\phi(\|\mathbf{x} - \mathbf{x}_i\|)|(j,i) = 1, 2, \dots, N$; $\mathbf{d}$ and $\mathbf{w}$ are the desired response and the weight vector respectively. N is the number of training samples.

Let $\mathbf{\Phi}$ denote the matrix of elements ϕ_{ji},

$$\mathbf{\Phi}\mathbf{w} = \mathbf{x} \tag{2}$$

$$\mathbf{w} = \mathbf{\Phi}^{-1}\mathbf{x} \tag{3}$$

For $\mathbf{\Phi}^{-1}$ to exist the interpolation matrix must be square and nonsingular.

Micchelli's Theorem

Micchelli (1986) proved the following,

Theorem

Let $\{\mathbf{x}_i\}_{i=1}^{N}$ be a set of distinct points in $\mathbb{R}^{m_0}$. Then the N by N interpolation matrix $\mathbf{\Phi}$, whose ji-th element is $\phi(\|\mathbf{x} - \mathbf{x}_i\|)$, is nonsingular.

- This theorem is applicable to a large class of radial-basis functions.
- The condition for singularity of the interpolation matrix $\boldsymbol{\Phi}$ is that$\{x_i\}_{i=1\ to\ N}$ must all be different.
- Whatever the value of N, the centers must be distinct.

RBFNs with Non-square Interpolation Matrices

- When the number of hidden units, m_1 is less than the number of input samples N, $\boldsymbol{\Phi}$ is a rectangular matrix.
- We can solve for the weight vector w by using the pseudo inverse of $\boldsymbol{\Phi}$, $\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{d}$
- Centers may be chosen randomly from the input vector or can be learned using a learning algorithm.
  – K-means: unsupervised learning
  – stochastic gradient: supervised learning

Examples of RBFs

Micchelli's theorem is applicable to a large class of radial-basis functions which includes,

- Gaussian functions

$$\phi(r) = exp\left(- \frac{r^2}{2\sigma^2} \right), \text{ for some } \sigma > 0 \text{ and } r \in \mathbb{R}$$

- Multiquadrics

$$\phi(r) = \frac{1}{(r^2 + c^2)^{1/2}}, \text{ for some } c > 0 \text{ and } r \in \mathbb{R}$$

- Inverse multiquadrics

$$\phi(r) = (r^2 + c^2)^{1/2}, \text{ for some } \sigma > 0 \text{ and } r \in \mathbb{R}$$

Deriving the RBF Learning Expression

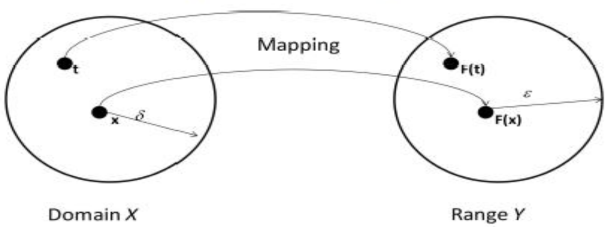
- The output of the RBF network with m1 hidden units is given by,

$$F(x) = \sum_{k=1}^{m_1} w_k \phi_k(\mathbf{x})$$

- This expression was originally proposed based on intuitive reasoning.
- We can derive this expression based a theoretical point-of-view.

Ill-posed versus Well-posed Problems

- Suppose we have a set of numbers X related to another set of numbers Y by some function f.
- The problem of reconstructing the function F is considered *well-posed* if it satisfies three-conditions,
  - Existence: for each input $x \in \mathcal{X}$ there is an output $y = f(x)$
  - Uniqueness: a pair of inputs x_1 and x_2 have the same output, $f(x_1) = f(x_2)$, if only if $x_1 = x_2$
  - Continuity: for any $\epsilon > 0$ there exists $\delta = \delta(\epsilon)$ such that $\rho_x(x, t) < \delta \rightarrow \rho_y(f(x), f(t)) < \epsilon$, where $\rho(\cdot, \cdot)$ is the distance between x and t in their respective spaces.



- The problem becomes ill-posed if even one of these conditions is not met.

Supervised Learning as an Ill-Posed Problem

- We may not have a training data set with all possible inputs.
- Every available training input may not have a unique output.
- Noise in the dataset or uncertainty in the measurements can lead to an imprecise mapping. For a given input, we might produce an output that is not within the output range.
- Ill-posed problems do not have unique solutions. In order to narrow down the range of solutions we need some kind of prior knowledge. This is called regularization.
- Using regularization we can make an ill-posed problem into a well-posed one, and derive the RBF formalism.

Regularization Theory

- Proposed by Tikhonov in 1963 for solving ill-posed problems
- The basic idea is to introduce prior information about the solution into the learning process to compensate for incomplete or noisy data.
- Most common form of prior information is that similar inputs correspond to similar outputs -that the mapping is smooth.

Tikhonov's Regularization Theory

Consider the training data, inputs: $\mathbf{x}_i \in \mathbb{R}^{m_0}$ and targets: $\mathbf{d}_i \in \mathbb{R}^1$, where $i = 1, 2, \ldots, N$ is the number of data samples. The task is to approximate the function $F(\mathbf{x})$

Tikhonov's regularization theory involves two terms:

1. Standard error term: difference between desired and actual response

$$\mathcal{E}_s(F) = \frac{1}{2} \sum_{i=1} N(d_i - y_i)^2 = \frac{1}{2} \sum_{i=1} N(d_i - F(\mathbf{x}_i))^2$$

2. Regularizing term: depends on the geometric properties of F

$$\mathcal{E}_c(F) = \frac{1}{2} \|\mathbf{D}F\|^2$$

where D is a linear differential operator, and $\|\cdot\|$ is the norm.

Prior information about the solution is embedded in $\mathbf{D}$. Choice of $\mathbf{D}$ is problem dependent.

Tikhonov Functional

- The objective in regularization theory is to minimize the sum of the standard error term and regularizing term.

$$\mathcal{E}(F) = \mathcal{E}_s(F) + \lambda\mathcal{E}_c(F) \tag{4}$$

$$= \frac{1}{2}\sum_{i=1} N(d_i - F(\mathbf{x}_i))^2 + \frac{1}{2}\lambda\|\mathbf{D}F\|^2 \tag{5}$$

λ is the regularization parameter; usually a positive real number

- $\mathcal{E}(F)$ is known as the *Tikhonov functional* and its solution is denoted $F_\lambda(\mathbf{x})$
- The regularization parameters λ can be seen as an indicator of the sufficiency of the dataset.

$\lambda \rightarrow 0$ implies an unconstrained problem

$\lambda \rightarrow \infty$ implies a constrained problem

Solution to the Tikhonov Functional

- For the Tikhonov functional to have a minimum at $F_\lambda(x)$, the solution must satisfy the *Euler-Lagrange* equation give below,

$$\tilde{\mathbf{D}}\mathbf{D}F_\lambda(\mathbf{x}) - \frac{1}{\lambda}[d_i - F(\mathbf{x}_i)]\delta(\mathbf{x} - \mathbf{x}_i) = 0$$

where $\tilde{\mathbf{D}}$ is the adjoint operator of $\mathbf{D}$ and $\delta(\cdot)$ is the *Dirac delta* function.

- The solution to the Euler-Lagrange equation is given in terms of *Green's functions*,

$$F_\lambda(\mathbf{x}) = \frac{1}{\lambda}\sum_{i=1}^{N}[d_i - F(\mathbf{x}_i)]G(\mathbf{x}, \mathbf{x}_i) \tag{6}$$

where $G(\cdot, \cdot)$ is a Green's function whose form is determined by the choice of $\mathbf{D}$

Determining the Weights

- The solution is expressed as linear superposition of N Green's functions, where $\mathbf{x}_i$ are the centers of the expansion and the weights $\frac{1}{\lambda}[d_i - F(\mathbf{x}_i)]$ are the coefficients of the expansion.

$$w_i = \frac{1}{\lambda}[d_i - F(\mathbf{x}_i)] \tag{7}$$

- Recasting equation (6) as

$$F_\lambda(\mathbf{x}) = \sum_{i=1}^{N} w_i G(\mathbf{x}, \mathbf{x}_i) \tag{8}$$

- Evaluating the above for input $x_j, j = 1, 2, \ldots, N$

$$F_\lambda(\mathbf{x_j}) = \sum_{i=1}^{N} w_i G(\mathbf{x_j}, \mathbf{x}_i) \tag{9}$$

Determining the Weights (Contd.)

Using matrix notation

$$\mathbf{F}_\lambda = [F_\lambda(\mathbf{x}_1), F_\lambda(\mathbf{x}_2), \ldots, F_\lambda(\mathbf{x}_1)]^T$$

$$\mathbf{d} = [d_1, d_2, \ldots, d_N]^T$$

$$\mathbf{G} = \begin{bmatrix} G(\mathbf{x}_1, \mathbf{x}_1) & G(\mathbf{x}_1, \mathbf{x}_2) & \ldots & G(\mathbf{x}_1, \mathbf{x}_N) \\ G(\mathbf{x}_2, \mathbf{x}_1) & G(\mathbf{x}_2, \mathbf{x}_2) & \ldots & G(\mathbf{x}_2, \mathbf{x}_N) \\ \vdots & \vdots & \ldots & \vdots \\ G(\mathbf{x}_N, \mathbf{x}_1) & G(\mathbf{x}_N, \mathbf{x}_2) & \ldots & G(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix}$$

$$\mathbf{w} = [w_1, w_2, \ldots, w_N]^T$$

Equations (7) and (9) can be rewritten as,

$$\mathbf{w} = \frac{1}{\lambda}(\mathbf{d} - \mathbf{F}_\lambda) \quad (10)$$

$$\mathbf{F}_\lambda = \mathbf{G}\mathbf{w} \quad (11)$$

Determining the Weights (Contd.)

Eliminating $\mathbf{F}_\lambda$,

$$(\mathbf{G} + \lambda\mathbf{I})\mathbf{w} = \mathbf{d}$$

Given that $G(\mathbf{x}_j, \mathbf{x}_i) = G(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathbf{G}^T = \mathbf{G}$

$$\mathbf{w} = (\mathbf{G} + \lambda\mathbf{I})^{-1}\mathbf{d}$$

In conclusion, the solution to the regularization problem is given by,

$$F_\lambda(\mathbf{x}) = \sum_{i=1}^{N} w_i G(\mathbf{x}, \mathbf{x}_i)$$

where $G(\cdot, \cdot)$ is the Green's function for the self-adjoint operator $\tilde{\mathbf{D}}\mathbf{D}$

Regularization Theory and RBF

- When the stabilizer D is *translationally* and *rotationally* invariant, the Green's function becomes a radial-basis function.
- One example of a Green's function that corresponds to a *translationally* and *rotationally* invariant D is the multivariate Gaussian function.

$$G(\mathbf{x}, \mathbf{x}_i) = G(\|\mathbf{x} - \mathbf{x}_i\|)$$

$$G(\mathbf{x}, \mathbf{x}_i) = exp\left(-\frac{1}{2\sigma_i^2}\|\mathbf{x} - \mathbf{x}_i\|^2\right)$$

- The regularized solution becomes

$$F_\lambda(\mathbf{x}) = \sum_{i=1}^{N} w_i exp\left(-\frac{1}{2\sigma_i^2}\|\mathbf{x} - \mathbf{x}_i\|^2\right)$$

- This is a RBF network! It is known to be a *universal approximator.*