Radial Basis Functions

- In most feedforward networks, new information necessitates total re-training.
- This re-training's cost becomes prohibitive when there are many data points and many weights.
- One way to avoid re-training particularly when large nets are involved is to allow hidden units to be sensitive only to a limited range of input patterns.
- One means of accomplishing this is by using radial basis functions as the activation functions of the hidden units.
- The learning set $\{v, y\}$ is considered to be a sampled version of some continuous function $y(v)$ that is not known.
- The goal is to construct an approximation from the samples which comes close to the true function $y(v)$.
- In the one-dimensional case, the continuous function $y(v)$ is sampled at regular intervals and the task is to produce the reconstruction $d(v)$ from the support points.
- If a high enough sampling rate is used then the solution is to apply an appropriate low-pass filter to the sequence of points, or spikes.
- The low-pass filter changes every spike into a point spread function given by the impulse response function of the low-pass filter and renders the approximating smooth function $d(v)$ as the additive superposition of all the smoothed spikes.
- The approximation of an unknown probability function $prob(v)$ from a given learning set can be viewed as the combination of histogramming and smoothing.
- The histogram is established by
    1. partitioning the whole measurement space into the required number of bins
    2. counting how many points in the training set fall into each bin

- Vector quantization clustering can be used to perform both of these functions.
- This results in a sampled representation of the distribution function $prob(v)$
  - samples are the cluster cell centroids v_l with cluster cell counts h_l
  - the sampled representation is the set of pairs $[v_l, h_l]$ where $l = 1, \ldots, L$.
- The sampled representation must be smoothed to produce a continuous approximation. This requires a kernel function, usually the normal distribution function.
- The simplest approach is the use of the uniform width rotational Gaussians.
- A better approach is to apply individual kernel functions to each of the cluster cells, making use of the cluster cell covariance matrix K_l.
- If each of the L cluster cells is represented by a normal distribution function $prob(v|l)$ then it has its probability mass centred at μ_l but distributed over all the measurement space.
- The cluster count, $(h_l/total\ number\ of\ samples)$ is an estimate of the a priori probability $prob(l) = P_l$ of the respective component distribution.
- Summation over all L clusters gives a legitimate estimation for $prob(v)$

$$prob(v) = \sum_{l=1}^{L} P_l prob(V|l)$$

- The resulting approximations can be directly used for pattern classification applying the Bayes decision rule.
- Any least mean-square approximation $d(v)$ targeted to y is an estimation for the vector p of a posteriori probabilities.
- This is valid for a linear combination of radial basis functions.
- Each of the L radial basis functions consists of three parts

- the reference vector v_l
- a distance-measuring function $g(\cdot)$
- the kernel function $f(\cdot)$

which combine to give

$$x_l = f(g(v, v_l))$$

- The usual choices are
  - Euclidean distance for $g(\cdot)$
  - negative exponential function for $f(\cdot)$

  This results in

$$x_l = exp(-\eta|v - v_l|^2)$$

  which is called the Gaussian kernel.

- The kernel width is controlled by η instead of the standard deviation σ of the normal distribution

$$\eta = \frac{1}{2\sigma^2}$$

- The individual kernel function has $N + 1$ parameters
  - N components of the reference point v_l
  - width parameter η – in the simplest case this is kept constant for all the L kernel functions

- The linear combination of the L radial basis functions x_l, $l = 1, \ldots, L$ from the L reference points with weights c_l to be optimized gives the estimating function

$$d(v) = \sum_{l=1}^{L} c_l x_l(v)$$

  or

$$d(v) = A^T x(v)$$

- For purposes of pattern classification, we introduce a vector-valued polynomial function $d(v)$ consisting of K scalar polynomials

$$d_k(v) = a_k^T x(v) \; responsible \; for \; class \; k = 1, \ldots, K$$

- The K class-specific coefficient vectors a_k are combined into a coefficient matrix
$$d(v) = A^T x(v)$$
- Only A will be adjusted during the optimization (learning) procedure.
- Thus radial basis functions are combined with least mean-square optimization for adjusting the coefficient matrix A from the training set
$$S^2 = E\{|d(v) - y|^2\} = E\{|A^T x(v) - y|^2\} = min_A$$
- The optimizing criterion S^2 is called the residual variance
  - describes the remaining error after y has been approximated by $d(v) = A^T x(v)$
  - function of A and depends quadratically on A
  - the goal is to find the minimum of a quadratic function of A
  - if A is a scalar, finding the minimum involves
    - compute the first derivative of S^2 with respect to A
    - set the result to zero
    - the result is a linear expression for the optimum parameter A
- Procedure for finding the optimum coefficient matrix A
  - assume that A is the optimum solution to S^2
  - any deviation δA from the optimum A results in an increase of S^2
$$S^2(A + \delta A) \geq S^2(A) \;\; \forall \delta A \neq 0$$
  - the right side of this equation can be transformed to
$$S^2(A) = E\{|A^T x - y|^2\} = E\{[A^T x - y]^T [A^T x - y]$$
  - in general, the estimation $d(v)$ does not fit the target y exactly so that an error vector Δd remains
$$\Delta d(v) = d(v) - y = A^T x(v) - y$$
- the optimum coefficient matrix A is simply computed by solving a system of linear equations that consist of the two moment matrices $E\{xx^T\}$ and $E\{xy^T\}$ which describe the properties of the pattern source from which the patterns $[v, y]$ come.

  - the moment matrices must be computed from a given set of learning samples $\{v, y\}$

- Learning from examples consists of collecting moment matrices from the training set and solving a linear matrix equation.
- given an input vector x_i, the output of a hidden unit is given by

$$F(x) = e - \sum_i ((x_i - k_i)/\sigma)^2$$

  where k_i is the centre of the radial basis function and σ is the width of the Gaussian distribution specified by the function.
- This activation function dictates that the hidden unit's largest response occurs when the input matches k_i and falls off rapidly according to the degree of mismatch.
- In the 2-D case, this is like the response curve of a sensory cell.
- Main advantage is the limited effect of any input on the network's hidden units and weights.
- In the simplest case,
  - L reference vectors v_l and the width parameter η are discovered from vector clustering
  - the only free parameters are the coefficients in A
  - least mean-square technique is used to solve for A
- More complex variant,
  - use of individual width parameters η_l
  - these are included in the optimization procedure which must be a comprehensive gradient descent procedure
- Even more complex,
  - including the L reference vectors v_l into the optimization procedure
  - the Euclidean metric can be replaced by the Mahalanobis distance

$$(v - v_l)^T K_l^{-1} (v - v_l)$$

  - the matrix K_l of this metric can be included in the set of adjustable parameters