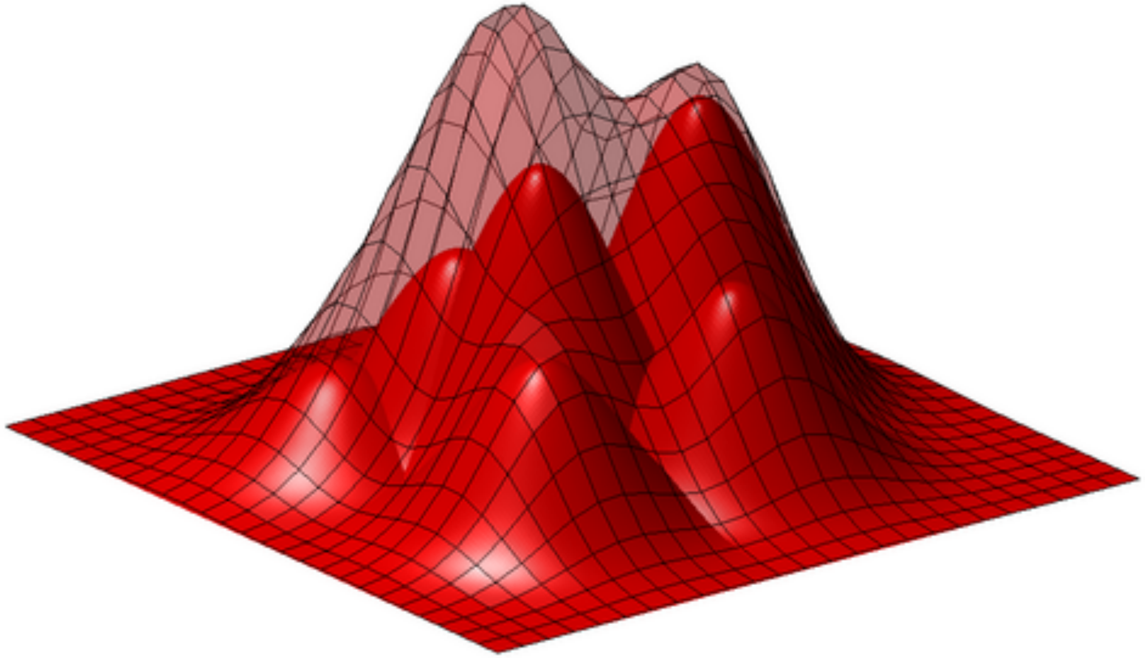
2.160 Identification, Estimation, and Learning

Part 4 Machine Learning and Nonlinear System Modeling

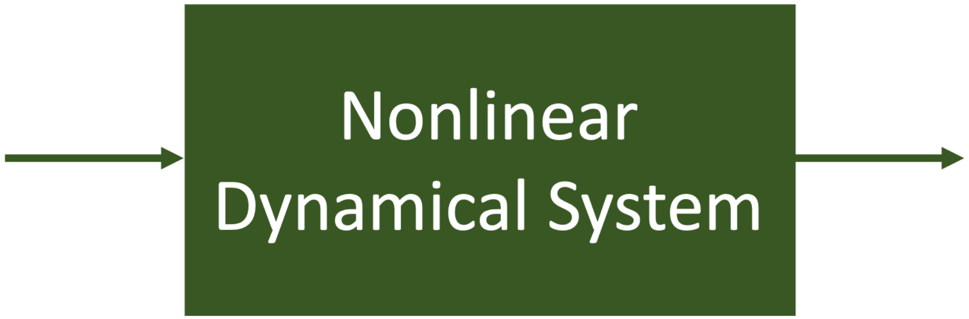
Lecture 19

Nonlinear Function Approximation



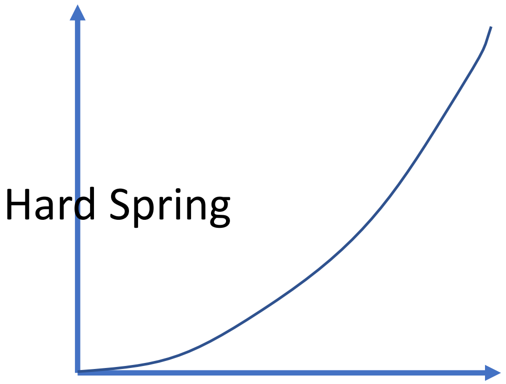
H. Harry Asada
Department of Mechanical Engineering
MIT

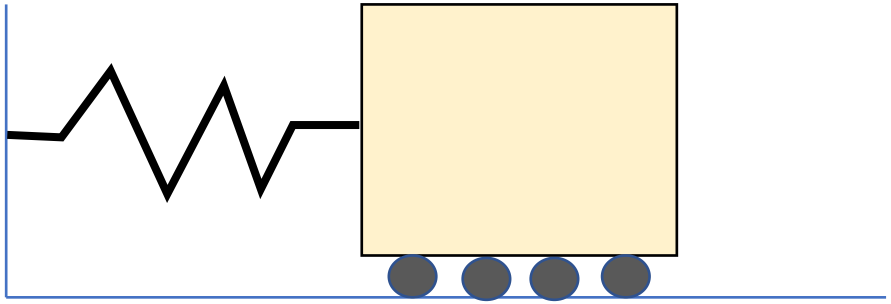
Nonlinear System Modeling

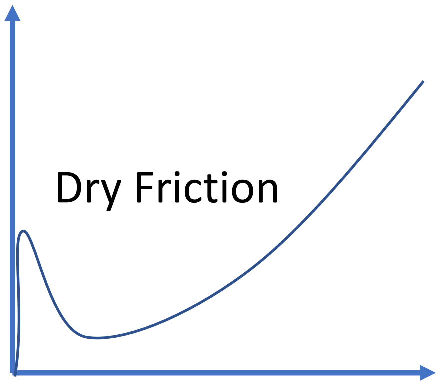


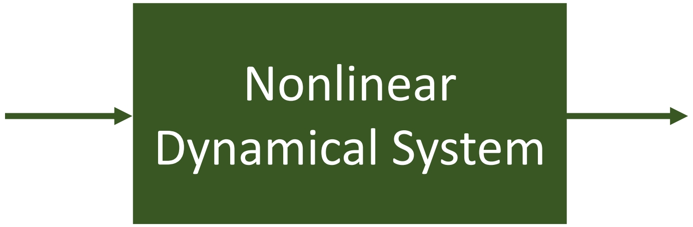
Nonlinear Dynamical System

- □ Practical systems are nonlinear to some extent.





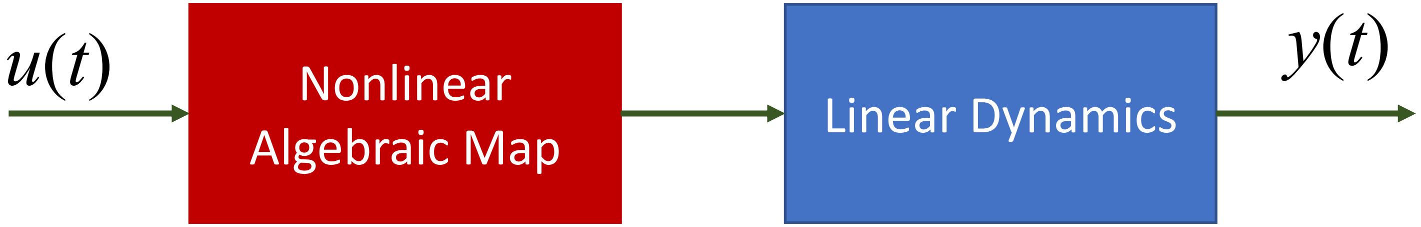




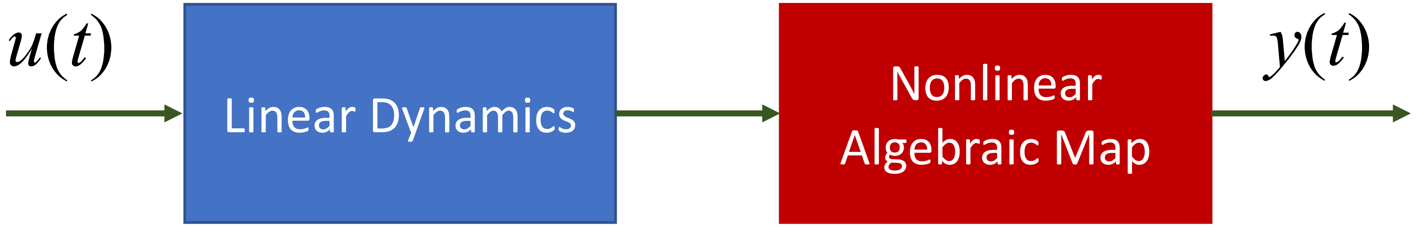
Hybrid Linear-Nonlinear Modeling

❑ Putting all nonlinear elements to either input side or output side, we can split a nonlinear dynamical system into a linear and nonlinear system.

Hammerstein Model



Wiener Model



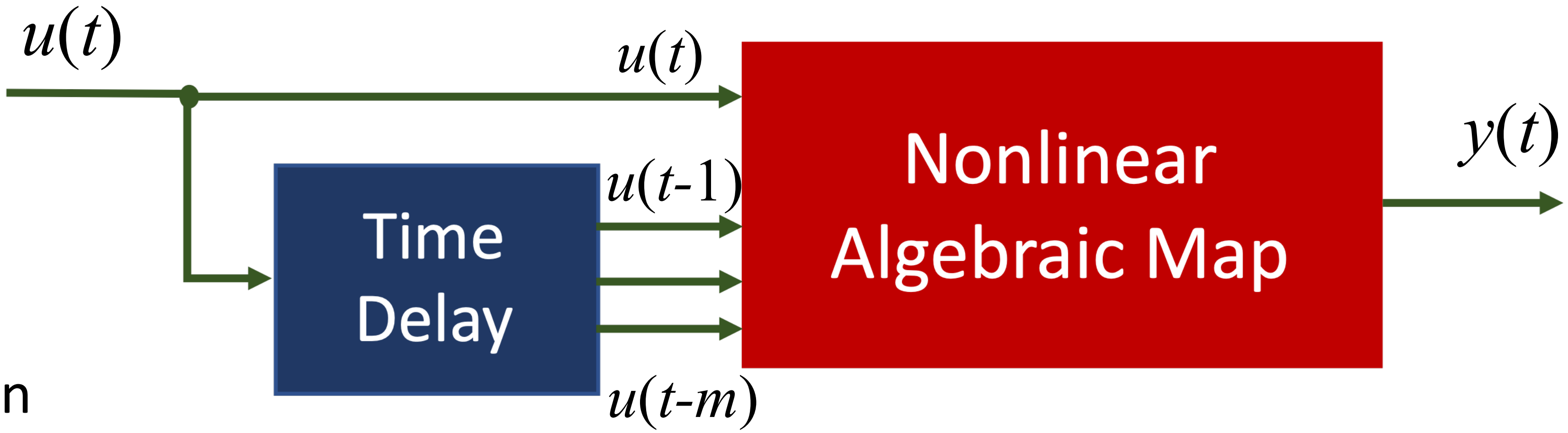
❑ Describing functions, too, apply to the above hybrid systems.

Nonlinear
Dynamical System

Use of Time Delay Outside an Algebraic Map

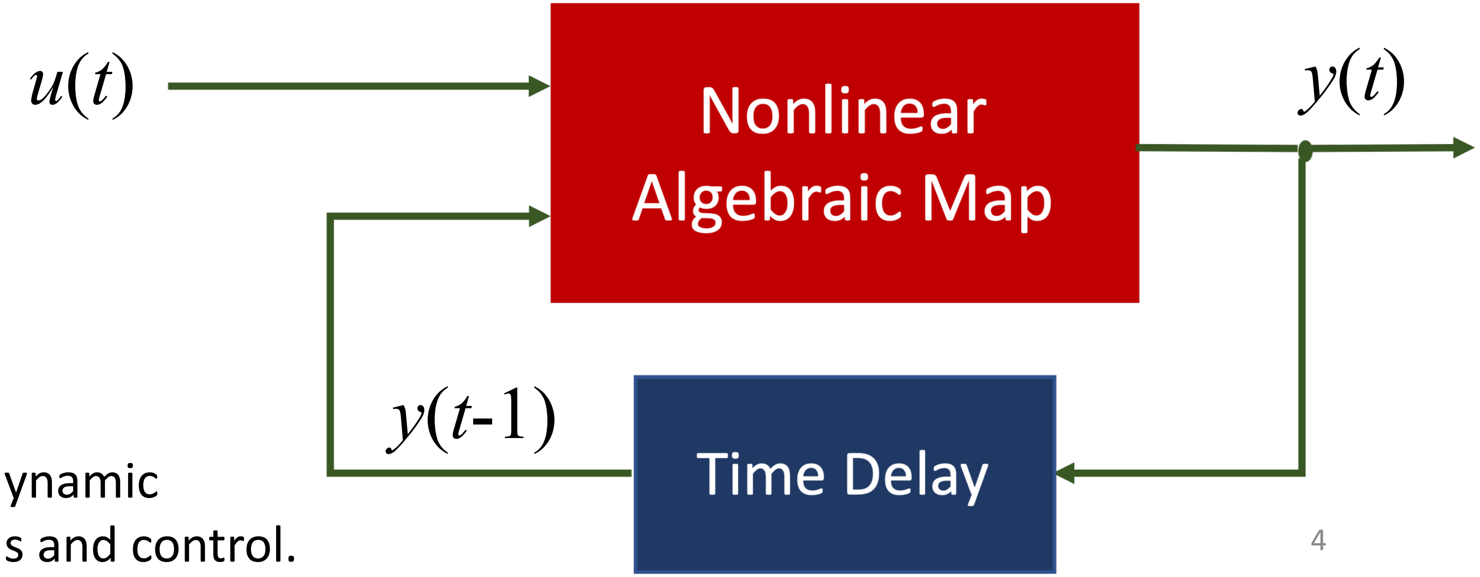
❑ Creating regressor at the input

Neural Network
Radial Basis Function



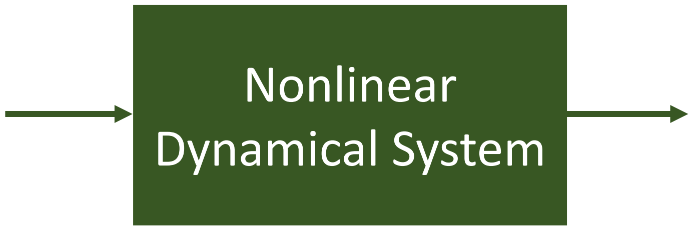
❑ Time Delay in the feedback loop

Recurrent Network



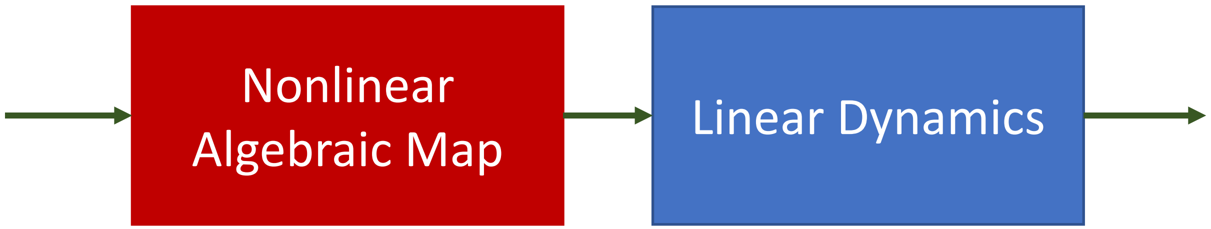
Directed Acyclic Graph can also capture dynamic behaviors, but are seldom used in systems and control.

Nonlinear System Modeling

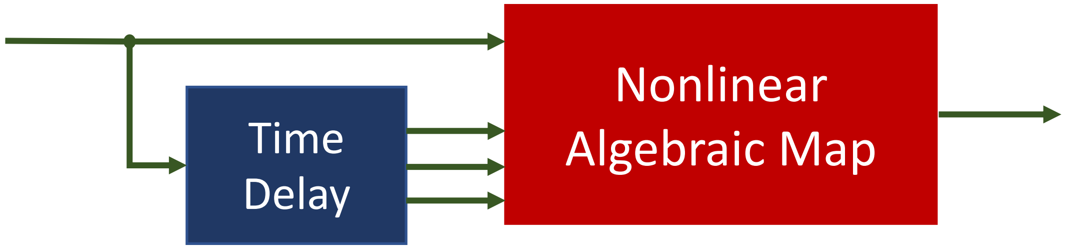

Nonlinear Dynamical System

Nonlinear Algebraic Map

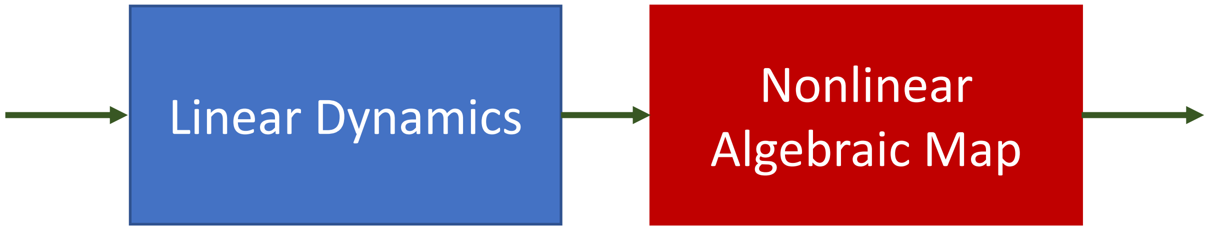
is involved in all models


Nonlinear Algebraic Map

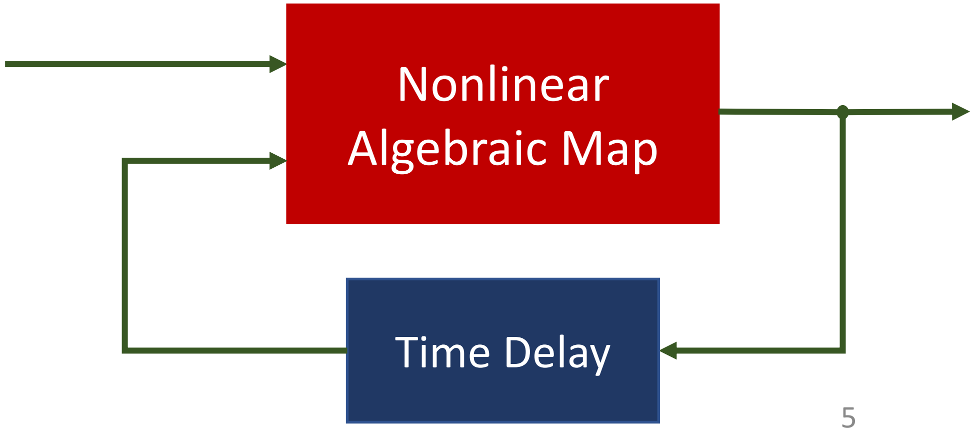
Linear Dynamics


Time Delay

Nonlinear Algebraic Map


Linear Dynamics

Nonlinear Algebraic Map

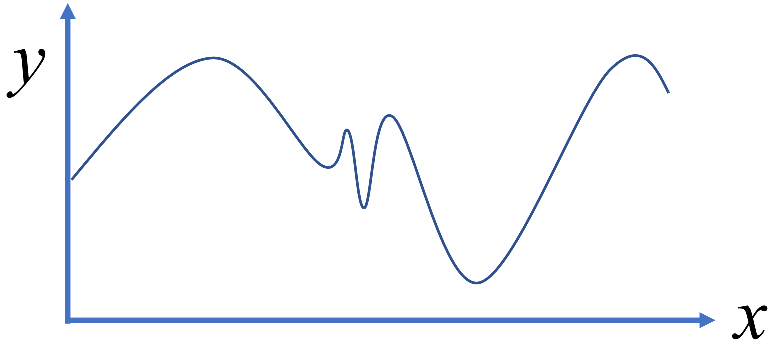

Nonlinear Algebraic Map

Time Delay

Nonlinear Algebraic Map

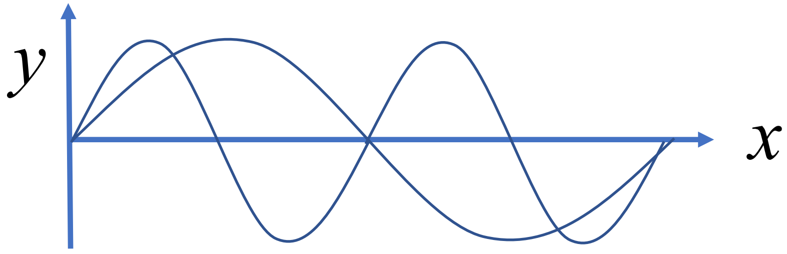
Expansion of a nonlinear function to a series of basis functions

$$y(x) \cong \sum_{k=1}^{m} \alpha_k g_k(x)$$



- □ Global basis functions
  - Trigonometric functions ----------- Fourier Series Expansion
  - Polynomials -------------------------- Volterra Series Expansion

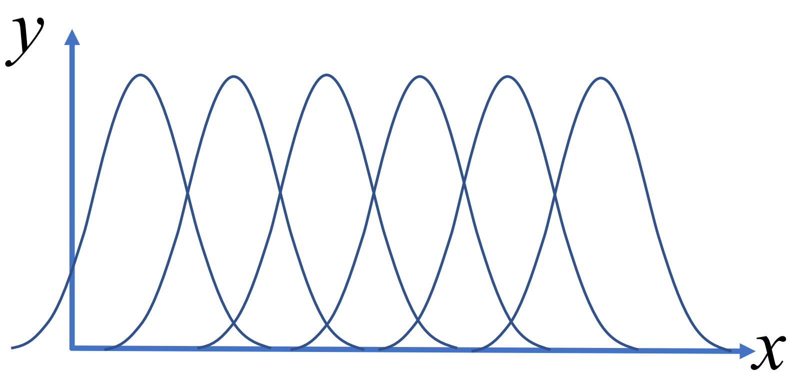
$$y = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_1 x_2 + \alpha_4 x_1^2 + \cdots$$



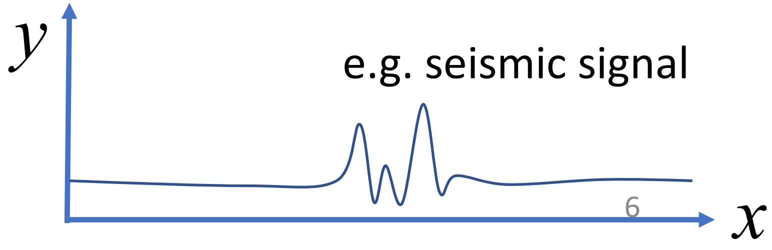
- □ Locally-tunable basis functions

  Capture local features of a nonlinear function, which would be averaged out if global basis functions are used.

  - Radial basis functions ----------- We will discuss this further here.
  - Wavelets ------------------------- Spatiotemporal functions



- □ Hybrid local-global basis functions
  - Neural networks



Radial Basis Functions

□ Radial Basis Functions are one of the most widely used local basis functions in control, machine learning; kernel, Gaussian processes, Koopman observables, etc.

□ Local basis functions are powerful tools for capturing local features and representing a nonlinear function with locally-tunable resolution and accuracy.

□ A Radial Basis Function $g_k : \Re^n \to \Re,\ k = 1, \cdots, m$ is a real-valued function that depends only on the distance between an input $x \in \Re^n$ and a center point $\gamma_k \in \Re^n$.

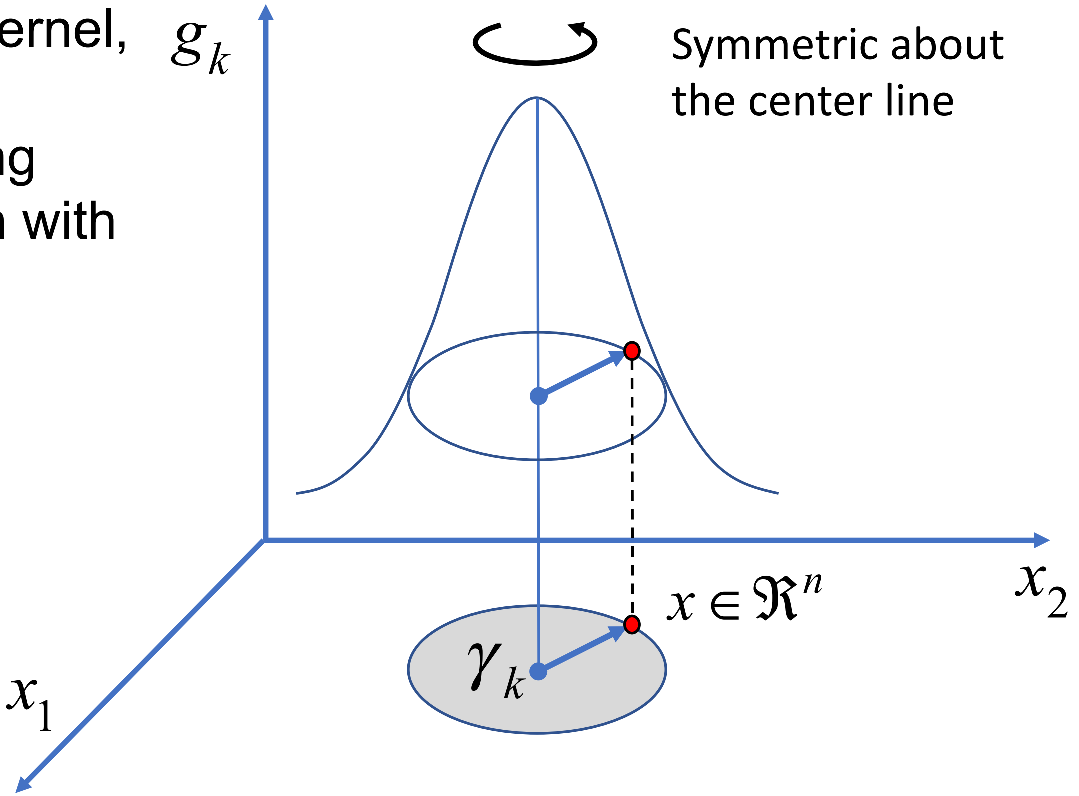
$$y = g_k(r_k) \quad k = 1, 2, \cdots, m$$

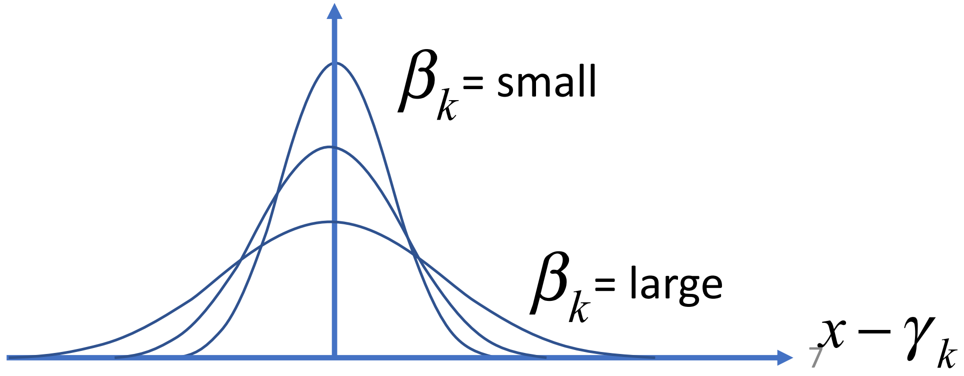
□ The distance r_k is scaled with a dilation parameter β_k:

Center point

$$r_k = \frac{|x - \gamma_k|}{\beta_k}, \quad k = 1, 2, \cdots, m$$

Dilation parameter





Radial Basis Functions - 2

❑ Among many radial basis functions, Gaussian function is the most prevailing.

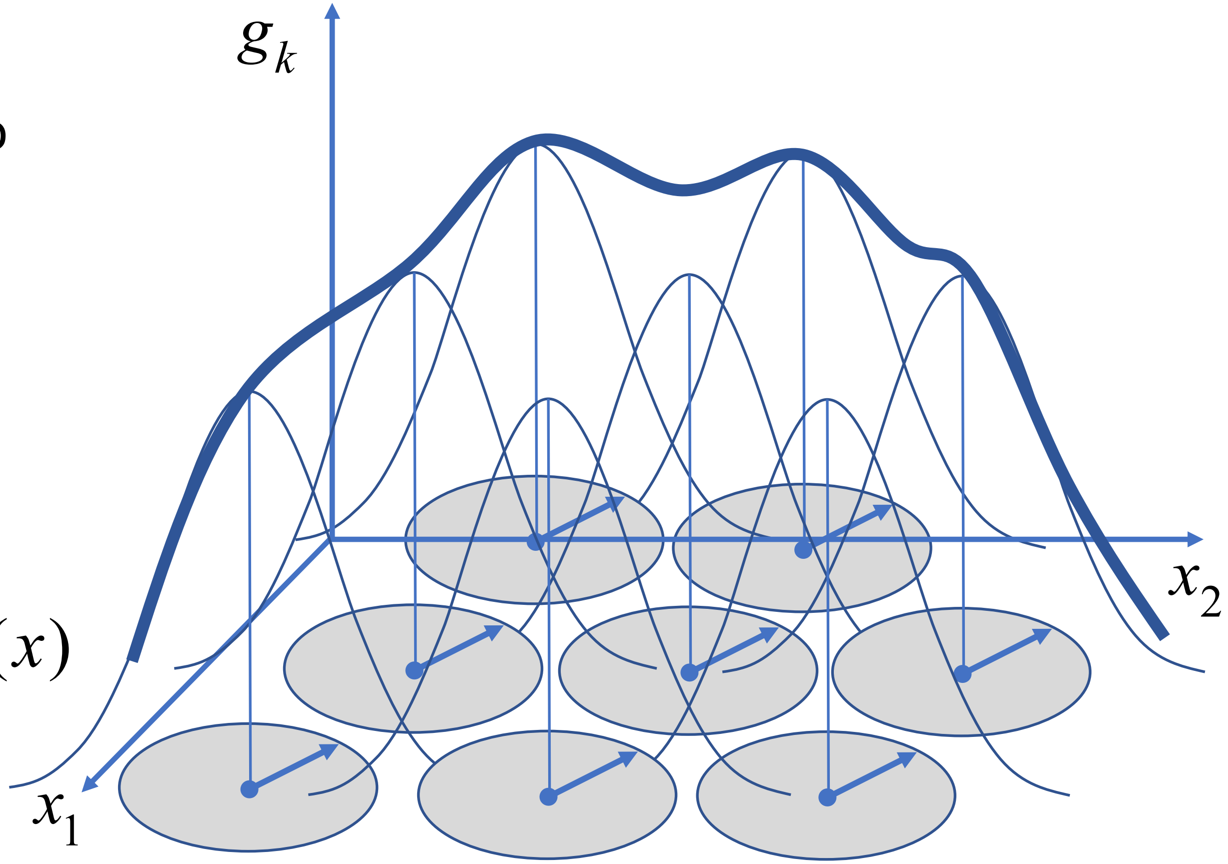
$$g_k(r_k) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} r_k^2 \right), \quad k = 1, 2, \cdots, m$$

❑ Expanding a nonlinear function, $g_0(x)$, to a series of radial basis functions:

$$g_0(x) = \sum_{k=1}^{m} \alpha_k g_k(r_k; \beta_k, \gamma_k)$$

❑ Placing m radial basis functions at center points γ_k with dilation parameters β_k, we approximate a given nonlinear function $g_0(x)$ by tuning weights α_k

$$r_k = \frac{|x - \gamma_k|}{\beta_k}, \quad k = 1, 2, \cdots, m$$



$$g_0(x) = \sum_{k=1}^{m} \alpha_k g_k(r_k;\beta_k,\gamma_k)$$

- ❑ This class of basis functions (and many other basis functions, including a neural network) can approximate a broad class of nonlinear functions to any accuracy.

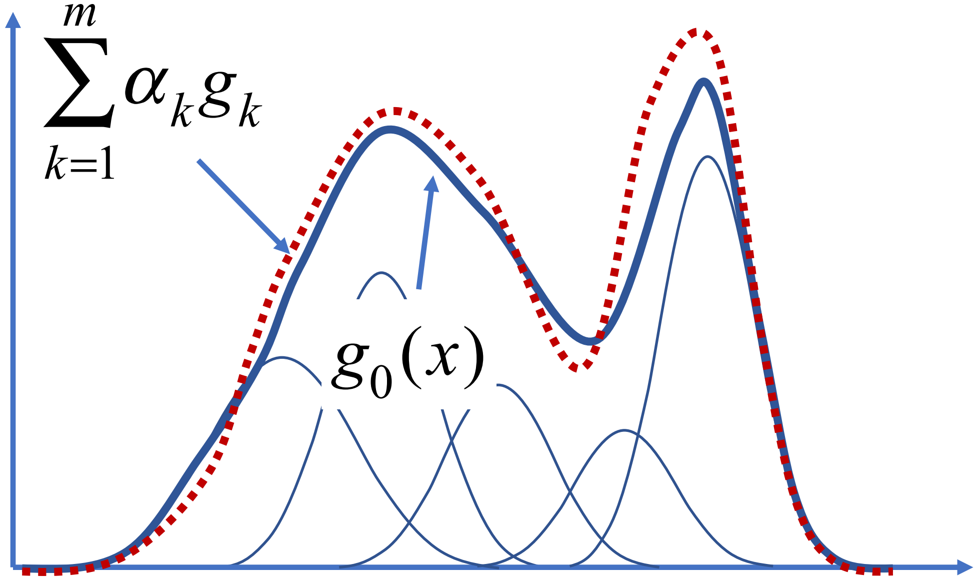
Function Approximation Theorem

For any (small) positive $\varepsilon > 0$, there exists a finite integer $m < \infty$ such that

$$\left| g_0(x) - \sum_{k=1}^{m} \alpha_k g_k(r_k;\beta_k,\gamma_k) \right| < \varepsilon,$$

$$\forall x \in D$$

where D is a compact subset.



This function approximation problem was studied by Kolmogorov, etc.

Tuning of a RBF network

❑ The objective of system identification is to fit a series of RBFs, or a network of RBFs, to a given set of input-output data.

$$\left\{(x(i), y(i)) \mid i = 1, \cdots, N\right\}$$

1). Non-Adaptive Grid Method

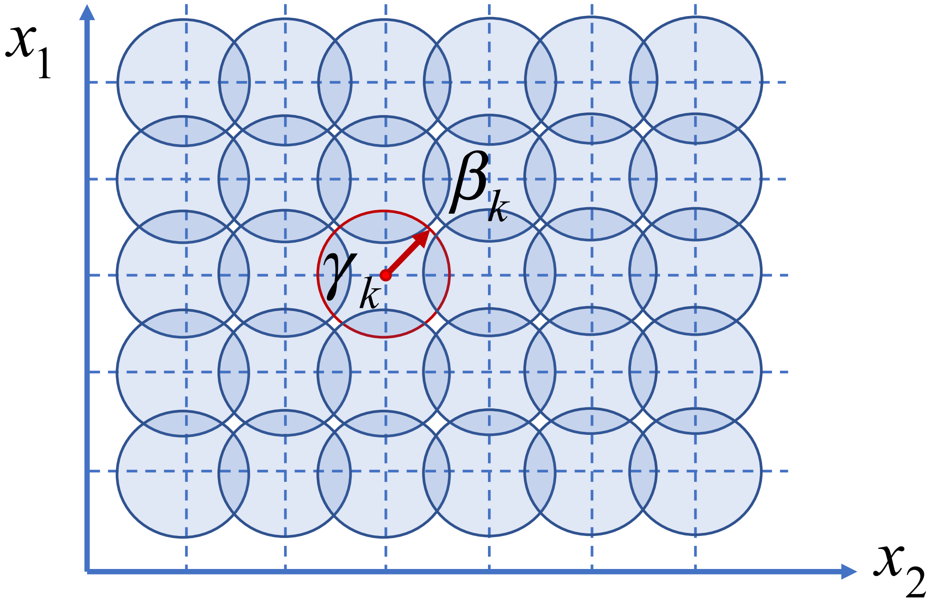
❑ Place center points of m RBFs at grid points uniformly across the input space : $x \in \Re^{n\times 1}$

$$\gamma_k \in \Re^{n\times 1}, k = 1, \cdots, m$$

❑ Dilation parameters β_k, too, are pre-determined.

❑ Writing $g_k(r_k; \beta_k, \gamma_k)$ as $g_k(x)$

$$r_k = \frac{|x - \gamma_k|}{\beta_k}$$



$$\hat{y}(x) = \alpha_1 g_1(x) + \cdots + \alpha_k g_k(x) + \cdots + \alpha_m g_m(x)$$

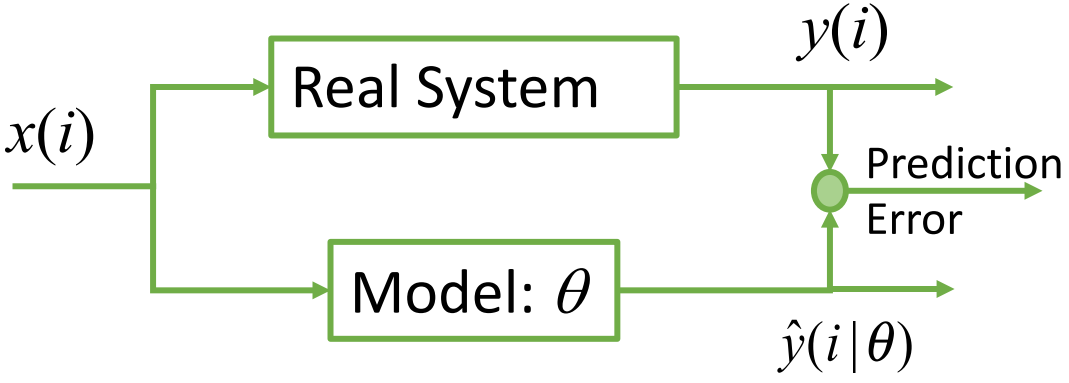
❑ Determine weights $\alpha_1 \cdots \alpha_m$ to minimize the prediction error.

Non-Adaptive Grid Method

- ❑ This is a standard Least Squares Estimate problem.

$$\hat{\theta}^{LS} = \arg\min \sum_{i=1}^{N} \left(y(i) - \hat{y}(x(i)) \right)^2$$

$$\left\{ (x(i), y(i)) \mid i = 1, \cdots, N \right\}$$



$$\hat{y}(x) = \alpha_1 g_1(x) + \cdots + \alpha_k g_k(x) + \cdots + \alpha_m g_m(x)$$

- ❑ Treat $g_1(x), \cdots, g_m(x)$ as components of a regressor,

$$\varphi = \begin{pmatrix} g_1 \\ \vdots \\ g_m \end{pmatrix}, \quad \theta = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{pmatrix}$$

We can write the predictor as a linear regression.

$$\hat{y}(x) = \theta^T \varphi(x)$$

Non-Adaptive Grid Method

❑The solution is given by

$$\hat{\theta}^{LS} = \arg\min \sum_{i=1}^{N} \left(y(i) - \hat{y}(x(i)) \right)^2 \qquad \hat{y}(x) = \alpha_1 g_1(x) + \cdots + \alpha_k g_k(x) + \cdots + \alpha_m g_m(x)$$

$$\hat{\theta}^{LS} = \begin{pmatrix} \hat{\alpha}_1 \\ \vdots \\ \hat{\alpha}_m \end{pmatrix} = \left[\sum_{i=1}^{N} \begin{pmatrix} g_1(x(i)) \\ \vdots \\ g_m(x(i)) \end{pmatrix} \begin{pmatrix} g_1(x(i)) & \cdots & g_m(x(i)) \end{pmatrix} \right]^{-1} \sum_{i=1}^{N} y(i) \begin{pmatrix} g_1(x(i)) \\ \vdots \\ g_m(x(i)) \end{pmatrix}$$

❑This simple LSE can be computed in real time. It has been applied to adaptive control, where the linear regression $\hat{y}(x) = \theta^T \varphi(x)$ is incorporated into a Lyapunov function, or computed with recursive least squares.

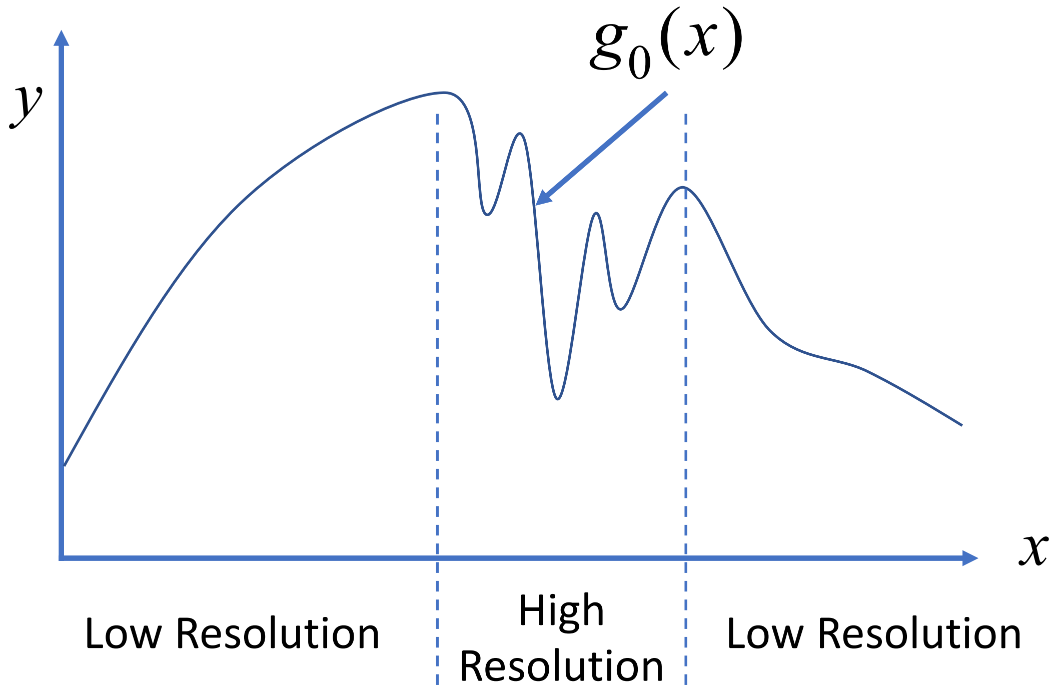
❑This, however, requires offline tuning of dilation parameters and grid points.

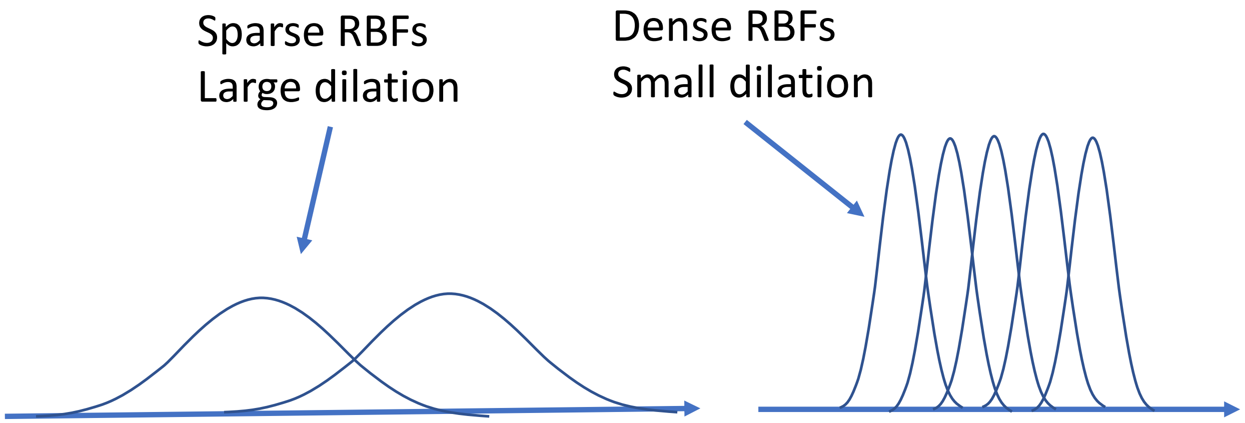
❑One drawback is the Curse of Dimensionality. As the input dimension increases, a number of RBF are required.

Tuning of a RBF network

2). Data-Adaptable Method

- □ The Grid Method places RBFs uniformly across the input space and, thereby, requires a lot of RBFs.
- □ The Grid Method also uses a uniform dilation parameter β_k for all the RBFs.
- □ Nonlinear functions may have diverse spatial frequencies, depending on the region of the function.
- □ Higher resolution may be needed in a particular region, while low resolution is acceptable for other regions.
- □ If the density of RBFs and dilation parameters can be tuned to specific local properties and needs, a limited number of RBFs can be used effectively for approximating a given functional relationship.





Data-Adaptable Method

- ❑ Depending on the spatial frequencies of nonlinear function $g_0(x)$ and required approximation accuracy, the density of RBFs and dilation parameters can be adjusted.

$$g_k(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} \frac{\left| x - \gamma_k \right|^2}{\beta_k{}^2} \right), \quad k = 1, 2, \cdots, m$$

- ❑ Challenge: Dilation β_k and center point γ_k are nonlinearly involved in each RBF.
- ❑ In principle, the three parameters, α_k , β_k and γ_k should be optimized simultaneously to best fit a given data set, but it is difficult. A practical approach is a sequential tuning:

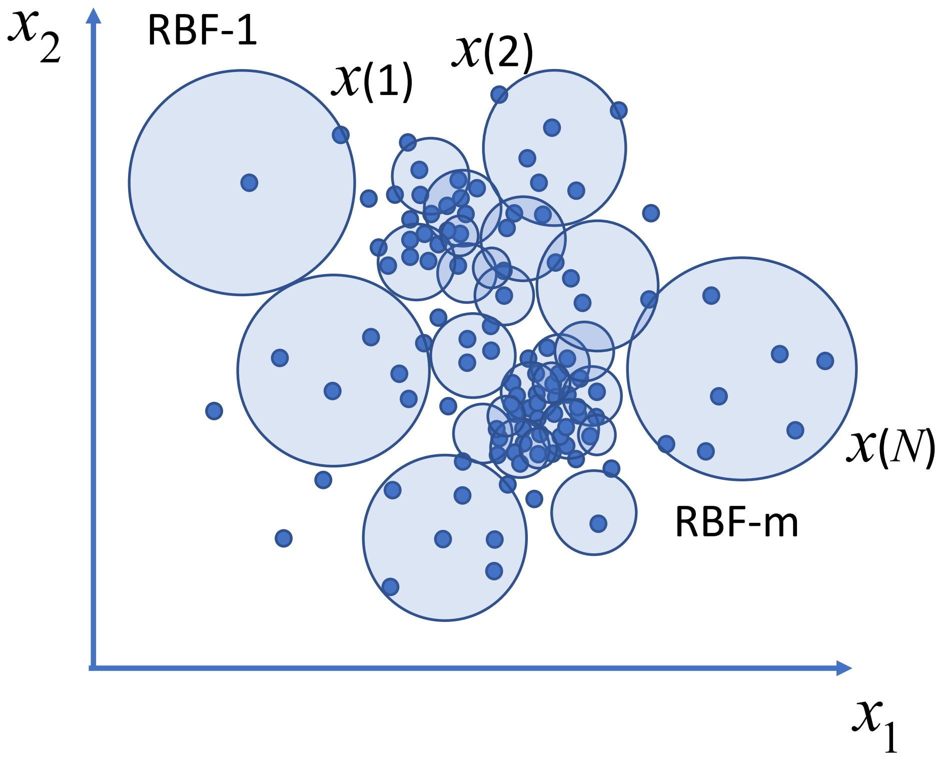
$$\gamma_k \to \beta_k \to \alpha_k$$

  - 1st Step: Placement of center points;
  - 2nd Step : Determination of dilation based on variance of nearby data points; and
  - 3rd Step : Least Squares Estimate

Data-Adaptable Method

Step 1 Center Point Placement

- ❑ More RBFs are placed in an area having a higher density of data points.
- ❑ Suppose that a given number of RBFs, m, are to be placed in an input space of N data points. How can we find m center points, so that the density of data points is approximately proportional to the density of center points?
- ❑ This problem can be solved as a Clustering Problem (Vector Quantization).



Given:

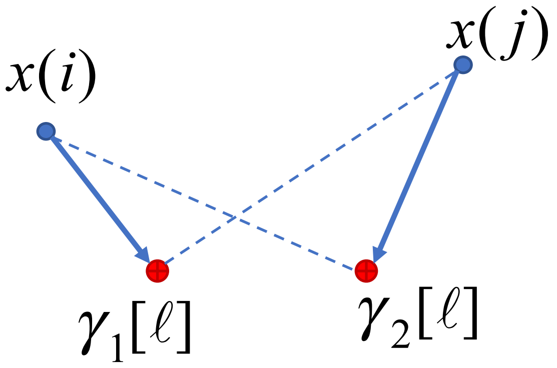
- ❑ N input data: $x(1), x(2), \cdots, x(N)$
- ❑ The number of RBFs: m;
- ❑ Initial locations of m center points: $\gamma_1[0], \gamma_2[0], \cdots, \gamma_m[0]$,

k – Means Clustering Algorithm

Set iteration index $\ell = 1$

Step 1. Given $\gamma_1[\ell], \gamma_2[\ell], \cdots, \gamma_m[\ell],$ find the nearest center point for each data point

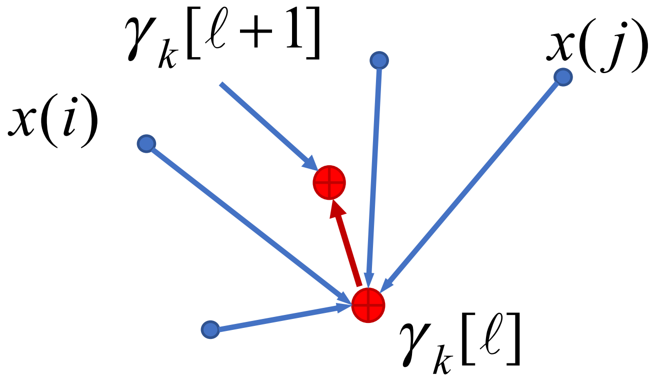
$$x(i),\ i = 1, \cdots, N$$



Step 2. Compute the center of mass, i.e. 1st order moment, of the data points that have been classified to the same cluster in Step 1.

$$\gamma_k[\ell] \rightarrow \gamma_k[\ell + 1], \quad k = 1, \cdots, m$$

Revised center point



Step 3. Set $\ell = \ell + 1$ and repeat Steps 1 and 2 until the within-cluster sum of squares converges to a local minimum.

$$J[\ell] = \frac{1}{N} \sum_{j=1}^{m} \sum_{i=1}^{N} \left| x(i) - \gamma_j[\ell] \right|^2 q_{ij}[\ell]$$

where

$$q_{ij}[\ell] = \begin{cases} 1: \text{ if data point } i \text{ belongs to cluster } j \\ 0: \text{else} \end{cases}$$

Membership Function $Q = \{q_{ij}\}$

$$J[\ell] = \frac{1}{N}\sum_{j=1}^{m}\sum_{i=1}^{N}\left|x(i)-\gamma_j[\ell]\right|^2 q_{ij}[\ell] \qquad q_{ij}[\ell] = \begin{cases} 1: \text{ if data point } i \text{ belongs to cluster } j \\ 0: \text{else} \end{cases}$$

$$Q = \{q_{ij}\} = \begin{array}{c|ccccc|} & 1 & \cdots & j & \cdots & m \\ \hline 1 & & & & & \\ \vdots & & & & & \\ i & 0 & 0 & 1 & 0 & 0 \\ \vdots & & & & & \\ N & & & & & \\ \hline \end{array} \qquad Q = \{q_{ij}\} = \begin{array}{c|ccccc|} & 1 & \cdots & j & \cdots & m \\ \hline 1 & & & 0 & & \\ \vdots & & & 1 & & \\ i & & & 1 & & \\ \vdots & & & 0 & & \\ N & & & 1 & & \\ \hline \end{array}$$

Only one 1 in each row.

Data point i belongs to center j.

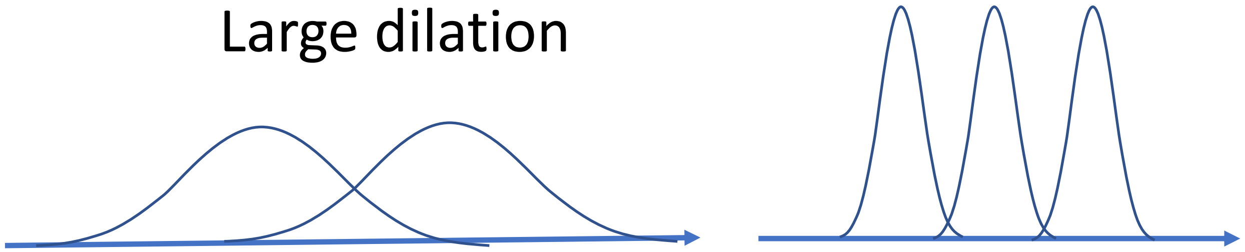
Multiple data points belong to center j.

Determining Dilation β_k

Properties of Dilation Parameter

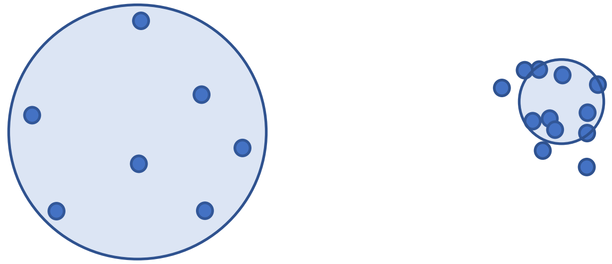
Large dilation

Small dilation



Sparse

Dense



Large overlap → Smooth curve

Low resolution

High resolution

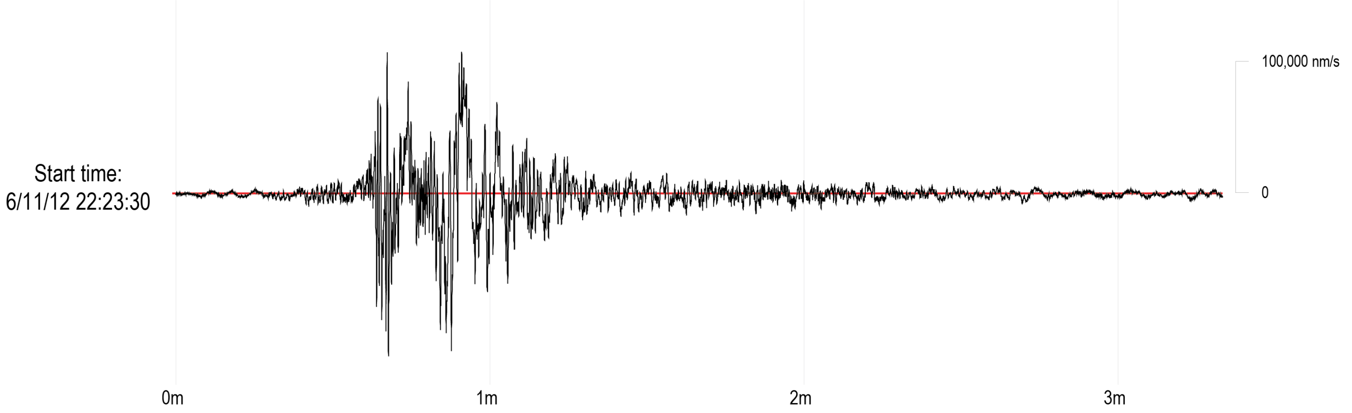
Determine Dilation based on the variance of data points belonging to the same cluster, i.e. within-cluster variance.

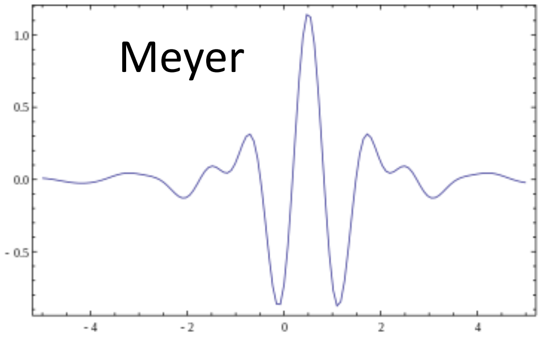
$$\beta_k = \frac{\sum_{i=1}^{N} \left| x(i) - \gamma_k \right|^2 q_{ik}}{\sum_{i=1}^{N} q_{ik}}, \quad k = 1, \cdots, m$$

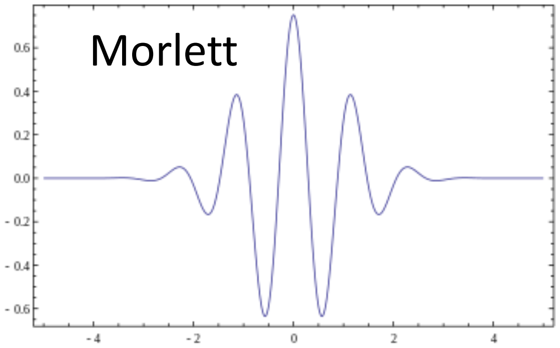
Discussion: Is this the right method?

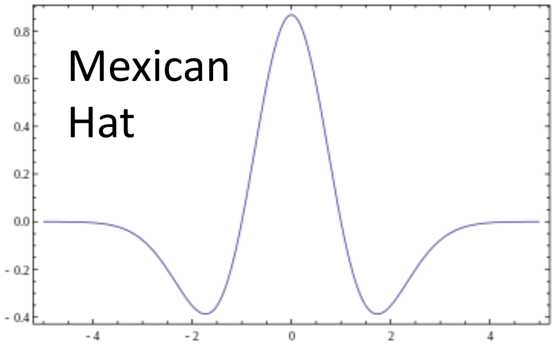
Wavelets

- Wavelets are a collection of brief oscillation functions, each having a specific frequency and features.
- When each function is convolved with an unknown signal, the correlation between the wavelet and the signal reveals where/when that component with the particular frequency and features has occurred.
- Applications: Seismograph, voice recognition, heart monitoring, etc.


Start time:
6/11/12 22:23:30
100,000 nm/s
0m
1m
2m
3m


Meyer


Morlett


Mexican
Hat