Lesson 2: Static Nonlinearities

Nonlinear Signal Processing – SS 2019

Christian Knoll
Signal Processing and Speech Communication Laboratory
Graz University of Technology

5. April 2019

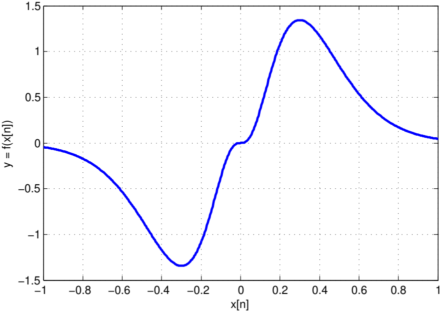
Session contents

- Static nonlinearities
- Numerical approximation
- Radial basis functions (RBFs), Fourier and sigmoid kernels
- Polynomial fit
- Influence of approximation parameters

Static Nonlinearities

- Output at time n is a deterministic function of the input at time n: $y[n] = f(x[n])$

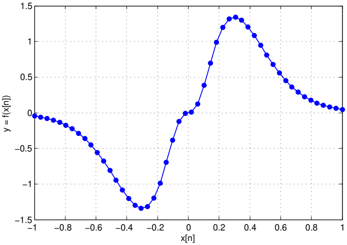
$$x[n] \longrightarrow \boxed{f(x[n])} \longrightarrow y[n]$$



Three Basic Problems

$$x[n] \longrightarrow \boxed{f(x[n])} \longrightarrow y[n]$$

1. $x[n], f(\cdot)$ known: Compute the output
2. $f(\cdot), y[n]$ known: Inverse modelling
3. $x[n], y[n]$ known: System identification



- We will deal with problem 3
- Given: Points $\boldsymbol{x} = [x_1, \ldots, x_N]$ and associated outputs $\boldsymbol{y} = [f(x_1), \ldots, f(x_N)]$

$\rightarrow$ Goal: Find a representation of NL $f(x[n])$

Representation of Static NL

- One possibility: Weighted sum of parametrized functions:

$$y[n] \approx \hat{f}(x(n)) = \sum_{k=1}^{K} \alpha_k \phi_k(x[n])$$

- *K basis functions* $\phi_k(x[n])$, popular example: *Radial Basis Functions* (depends on distance from a center), e.g. Gauss

$$\phi_k(x[n]) = \exp\left\{-\frac{(x[n]-\mu_k)^2}{2\sigma_k^2}\right\}$$

$\rightarrow$ How to find K, widths σ_k, centers μ_k and weights α_k?

Parameter Optimization (1)

- Here: Choose K, σ_k and μ_k *by hand*, i.e.
  - Examine plot of NL, make an educated guess
  - Grid search for optimal parameters
  - There are also formal, systematic ways (e.g. Bayesian techniques)
- Choice of weights α_k: Model is linear in the weights, easy optimization

$$y_i = \sum_k \alpha_k \phi_k(x_i) \quad \text{There are N such equations}$$

- Arrange in equation system

$$\underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} \phi_1(x_1) & \cdots & \phi_K(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_N) & \cdots & \phi_K(x_N) \end{bmatrix}}_{\boldsymbol{\Phi},\ (N \times K),\ N \gg K} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_K \end{bmatrix}$$

Parameter Optimization (2)

- Overdetermined system, least-squares solution
$$\boldsymbol{\alpha} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}$$
- This happens in `rbf.m` (with Gaussian $\phi_k(x)$)
- Also in `sigmoid.m` and `fourier.m`, just for different kernels
- For certain parameter choices the problem above can be ill-conditioned!

Evaluation of approximation

- How do we evaluate the quality of the approximation?
- True and approximated output available, calculate error

$$\boldsymbol{e} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}$$

- Can be arbitrarily small iff $f(x)$ is in signal space spanned by basis functions $\phi_k(x)$
- We use the Q_2 value to evaluate different approximations

$$Q_2 = 10\log_{10}\frac{\sum_n |y_{orig}[n] - y_{reconst}[n]|^2}{\sum_n |y_{orig}[n]|^2}$$

Polynomial approximation

- Another possibility for the approximation is given by polynomials

$$f(x[n]) = \sum_{k=0}^{\infty} c_k x^k[n]$$

- Model also linear in coefficients $\rightarrow$ Least squares fit analoguously possible (Matlab: `help polyfit()`)
- If $f(x)$ is differentiable, also Taylor series expansion possible

$$f(x) = \sum_{k=0}^{\infty} \frac{1}{k!} \frac{d^k f(a)}{dx^k} (x-a)^k = \sum_{k=0}^{\infty} c_k (x-a)^k$$

$\rightarrow$ See next session!

Next meeting . . .

- Next meeting after easter break
- $\rightarrow$ May, 5.
- Will deal with analytical techniques for static NLs