Nonlinear systems identification with Gaussian process models

References:

J.Kocijan, A. Girard, B. Banko, R. Murray-Smith (2005), Dynamic systems identification with Gaussian processes. Mathematical and Computer Modelling of Dynamic Systems, Volume 11, Issue 4, Pages 411-424.

K. Ažman and J. Kocijan (2007), Application of Gaussian processes for black-box modelling of biosystems. ISA Transactions, Volume 46, Issue 4, Pages 443-457.

Methods of nonlinear systems identification explained up to now

- Multilayer perceptron

(approximation of nonlinearity with riged – sigmoid basis functions)

- Radial basis function network

(approximation of nonlinearity with radial basis functions – Gaussian bell functions)

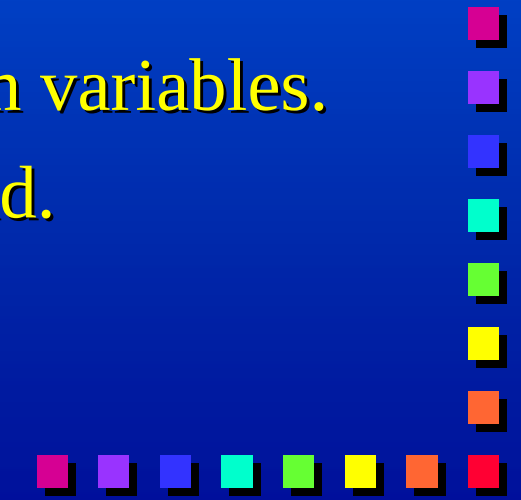
- Local model networks

(approximation of nonlinearity with linear local models as basis functions)

- Any different approaches?

Gaussian process models

- Let’s find distances between all input data and look for relation with output data: no basis functions, only relation between data.
- The method takes roots from statistics (Bayesian method).
- The method is based on vectors of random variables.
- Relatively complex theoretical background.
- Relatively easy use.



Some explanations

- Random variable: its value is random.
- Random process: realisation of more interconnected random variables.
- Gaussian process: random process, in which the random variables have normal - Gaussian distribution.

GP model

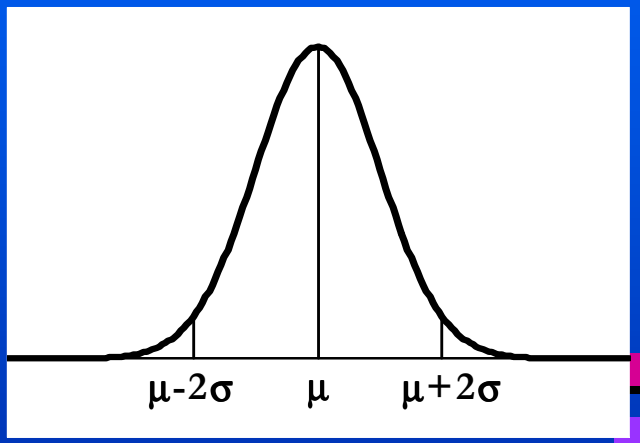
- Probabilistic (Bayes) model.
- Nonparametric model – no predetermined structure (basis functions) depending on system
- Determined by:
  - Input/output data $\mathbf{x}_i, y_i$ (data points, not signals) (learning data – identification data):
  - Covariance matrix:

$$K_{ij} = \operatorname{cov}(y_i, y_j)$$

$$\mathbf{K} = \begin{bmatrix} K_{11} & \dots & K_{1N} \\ \vdots & \ddots & \vdots \\ K_{N1} & \dots & K_{NN} \end{bmatrix}$$

GP model

- Prediction of the output based on similarity test input – training inputs
- Output: normal distribution
  - Predicted mean $\mu(\mathbf{x})$
  - Prediction variance $\sigma^2(\mathbf{x})$



$$\begin{aligned} \mu(\mathbf{x}^*) &= \mathbf{k}(\mathbf{x}^*)^T \mathbf{K}^{-1} \mathbf{y} \\ \sigma^2(\mathbf{x}^*) &= k(\mathbf{x}^*) - \mathbf{k}(\mathbf{x}^*)^T \mathbf{K}^{-1} \mathbf{k}(\mathbf{x}^*) \end{aligned}$$

$$\begin{aligned} \mathbf{k}(\mathbf{x}^*) &= [C(\mathbf{x}^1, \mathbf{x}^*) \dots C(\mathbf{x}^N, \mathbf{x}^*)] \\ k(\mathbf{x}^*) &= C(\mathbf{x}^*, \mathbf{x}^*) \end{aligned}$$

Covariance function

$$\mathbf{K} = \begin{bmatrix} K_{11} & \dots & K_{1N} \\ \vdots & \ddots & \vdots \\ K_{N1} & \dots & K_{NN} \end{bmatrix}$$

$$K_{ij} = \mathrm{cov}\,(y_i, y_j)$$

- Covariance function:

$$\mathrm{cov}\,(f(\mathbf{x}_i), f(\mathbf{x}_j)) = C(\mathbf{x}_i, \mathbf{x}_j)$$

  - functional part and noise part
  - stationary/unstationary, periodic/nonperiodic, etc.
  - Expreses prior knowledge about system properties,
  - frequently: Gaussian covariance function
    - Smooth function
    - Stationary function

$$C(\mathbf{x}_i, \mathbf{x}_j) = v \exp\left[-\frac{1}{2} \sum_{d=1}^{D} w_d (x_i^d - x_j^d)^2 \right] + \delta_{ij} v_0$$

Hyperparameters

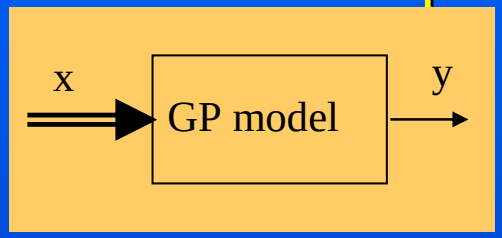
- Identification of GP model = optimisation of covariance function parameters

- Optimisation:
  - Cost function: maximum likelihood of data for learning

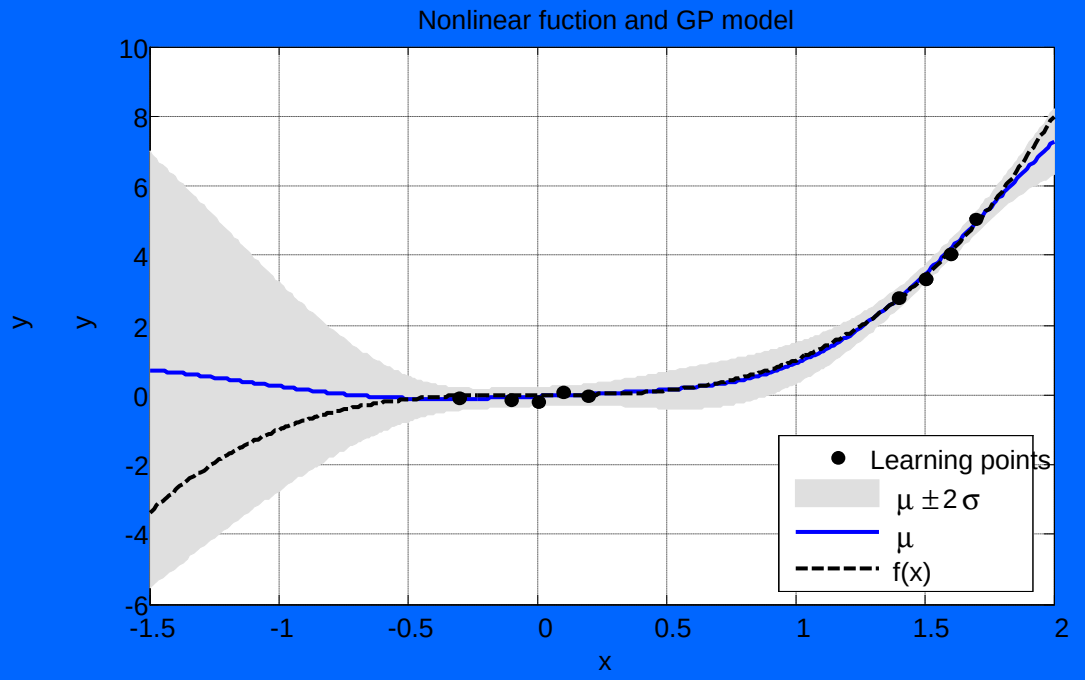
$$\mathcal{L}(\boldsymbol{\Theta}) = \log(p(\mathbf{y}|\mathbf{X}, \boldsymbol{\Theta})) = -\frac{1}{2}\log(|\mathbf{K}|) - \frac{1}{2}\mathbf{y}^T\mathbf{K}^{-1}\mathbf{y} - \frac{N}{2}\log(2\pi)$$

Static illustrative example

- Static example:



- $f(x) = x^3 + \epsilon$ $\mathcal{N}(0, 0.01)$
- 9 learning points: x_i, y_i
- Grey band: $\mu \pm 2\sigma$
- Rare data density → increased variance (higher uncertainty).



Dynamic systems

- Static vs. Dynamic
- Dynamic models:

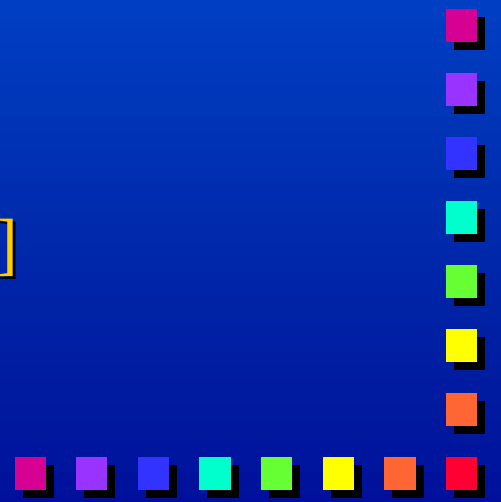
  conventional approach (ANN, fuzzy models, etc.) is delayed inputs and outputs as regressors
- Input/output training pairs $\mathbf{x}_i/y_i$

  $\mathbf{x}_i$ ... regressor values

  [u(k-1),..,u(k-L), y(k-1),..,y(k-L)]

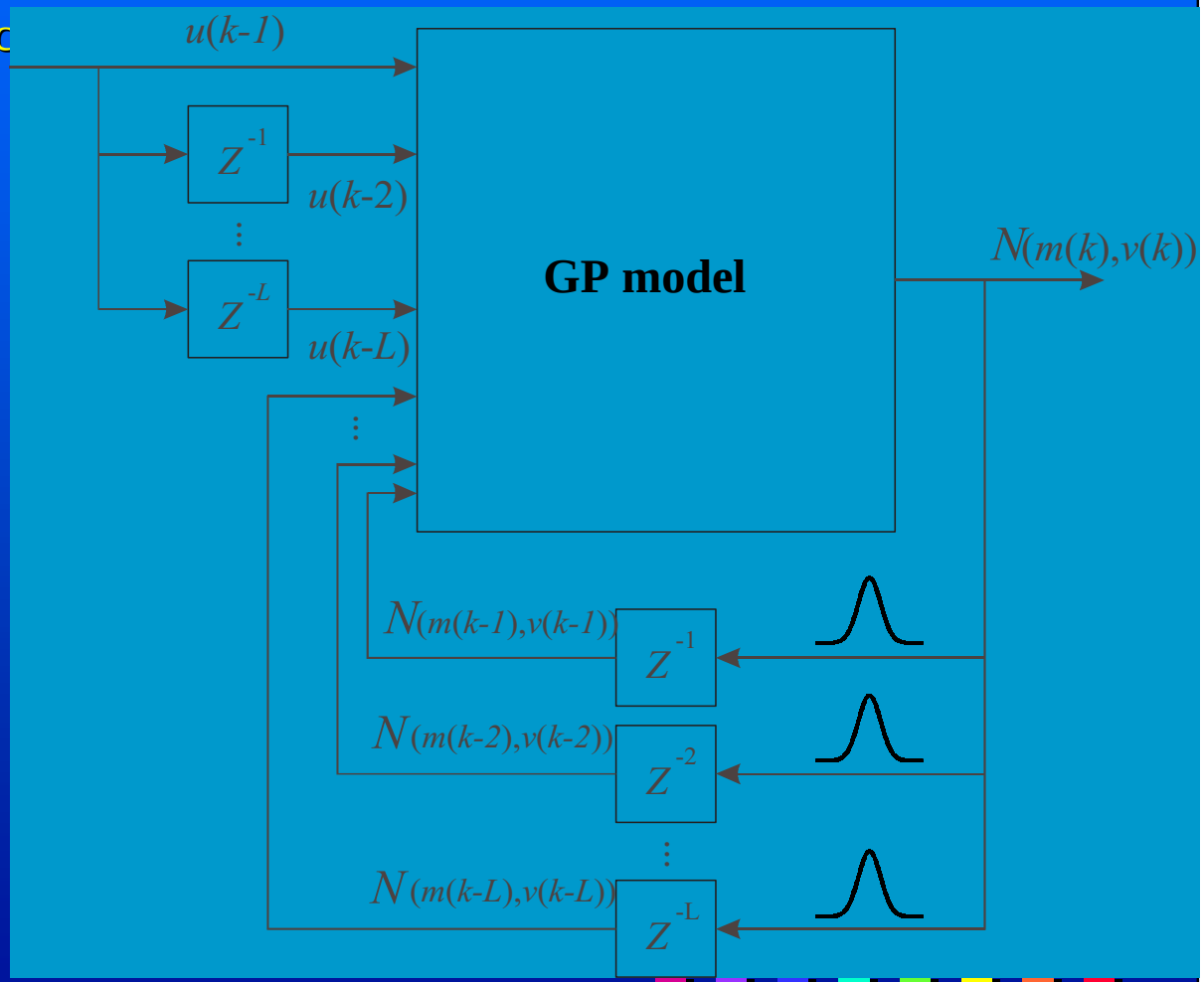
  y_i ... system output values

  y(k)



Dynamic system identification and model simulation

- Why does identification of dynamic systems seem more complex than modelling of static functions?
- Simulation
  - **“naive” ... m(k)**
  - with propagation m(k),v(k)
    - Analitic app.
      - **Taylor app.**
      - **“exact”**
    - MC Monte Carlo with mixtures



GP model attributes (vs. e.g. ANN)

- Smaller number of parameters
- Measure of confidence in prediction, depending on data
- Incorporation of prior knowledge *
- Easy to use (practice)

- Computational cost increases with amount of data
- Recent method, still in development
- Nonparametrical model

* (also possible in some other models)

Example of identification of 1st order system

Mathematical model of the process with parameters

$$y(k) = y(k-1) - 0.5\tanh\left(y(k-1) + u(k-1)^3\right)$$

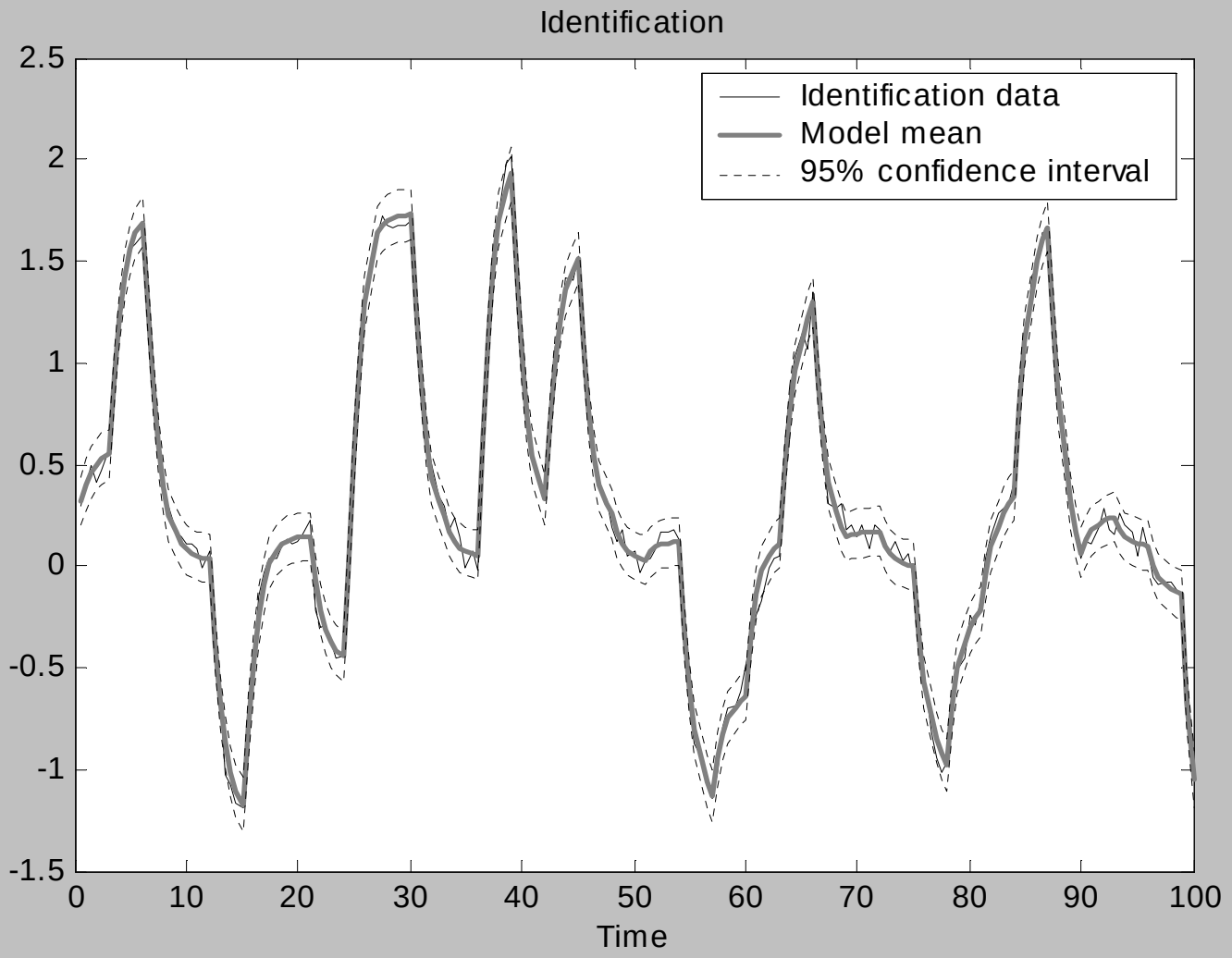
u – input signal

y – output signal

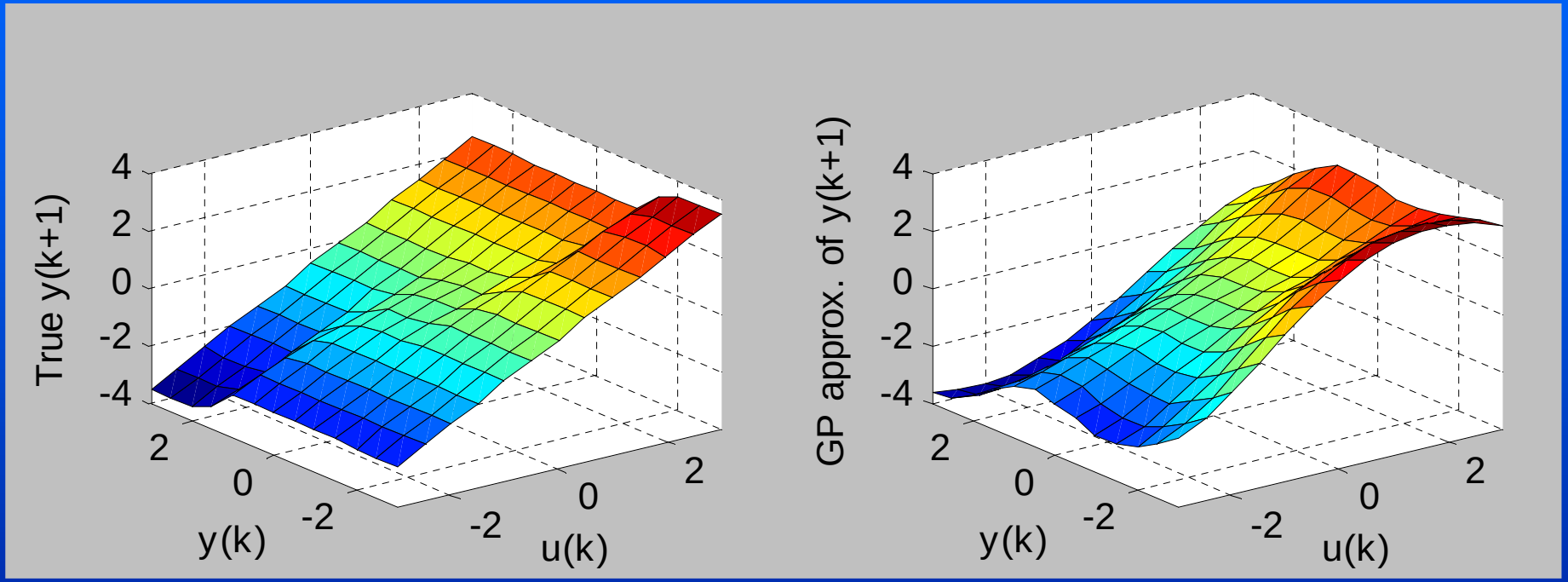
Input and output signals

- Input signal: [-1.3, 1.3]
- Sampling time 0.5 s
- Estimation data: input signal for identification (same as for ANN example).
- Validation data: two signals for validation (1st: same as for ANN example, 2nd: different region).

GP model response on estimation data

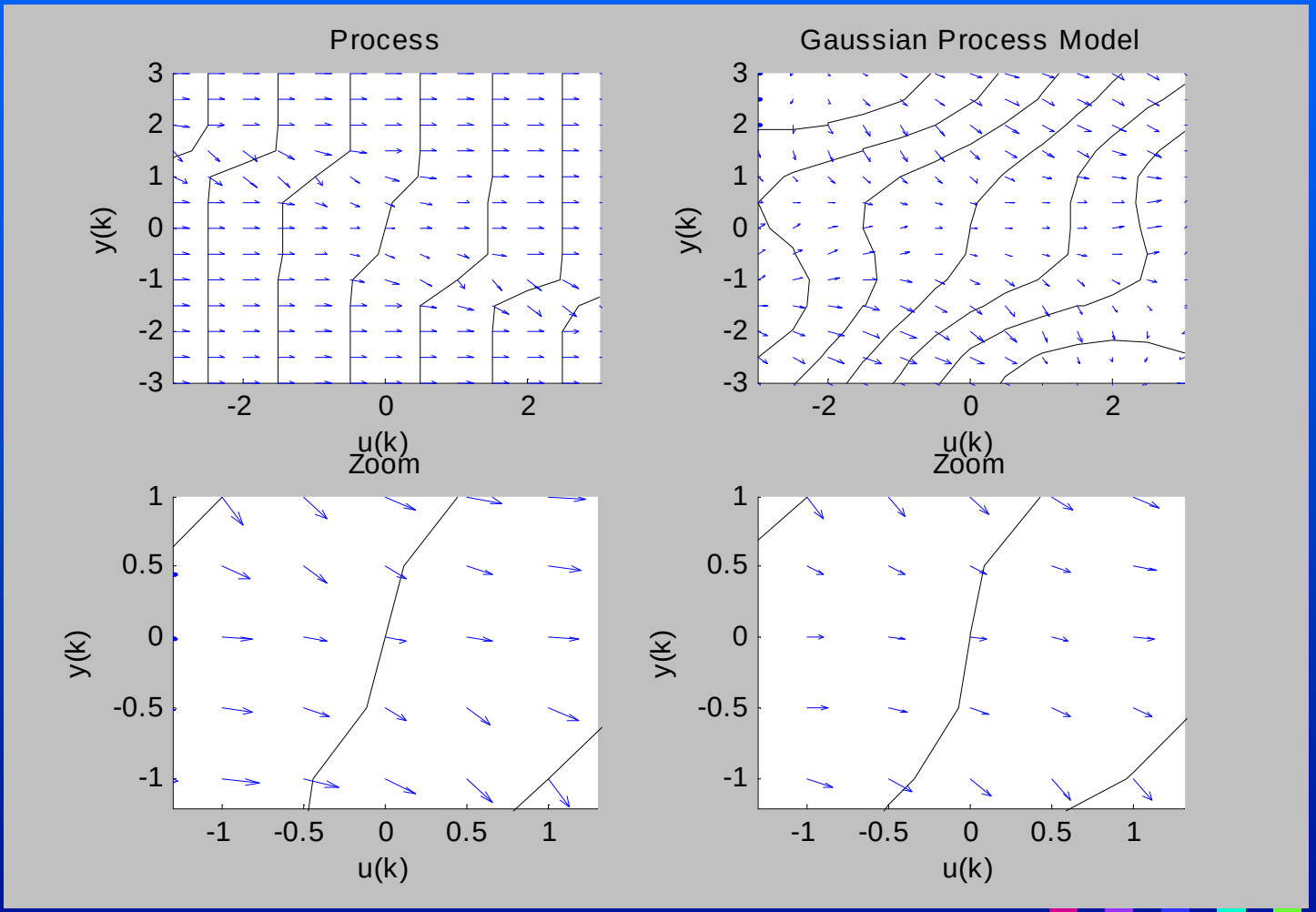


Model analysis

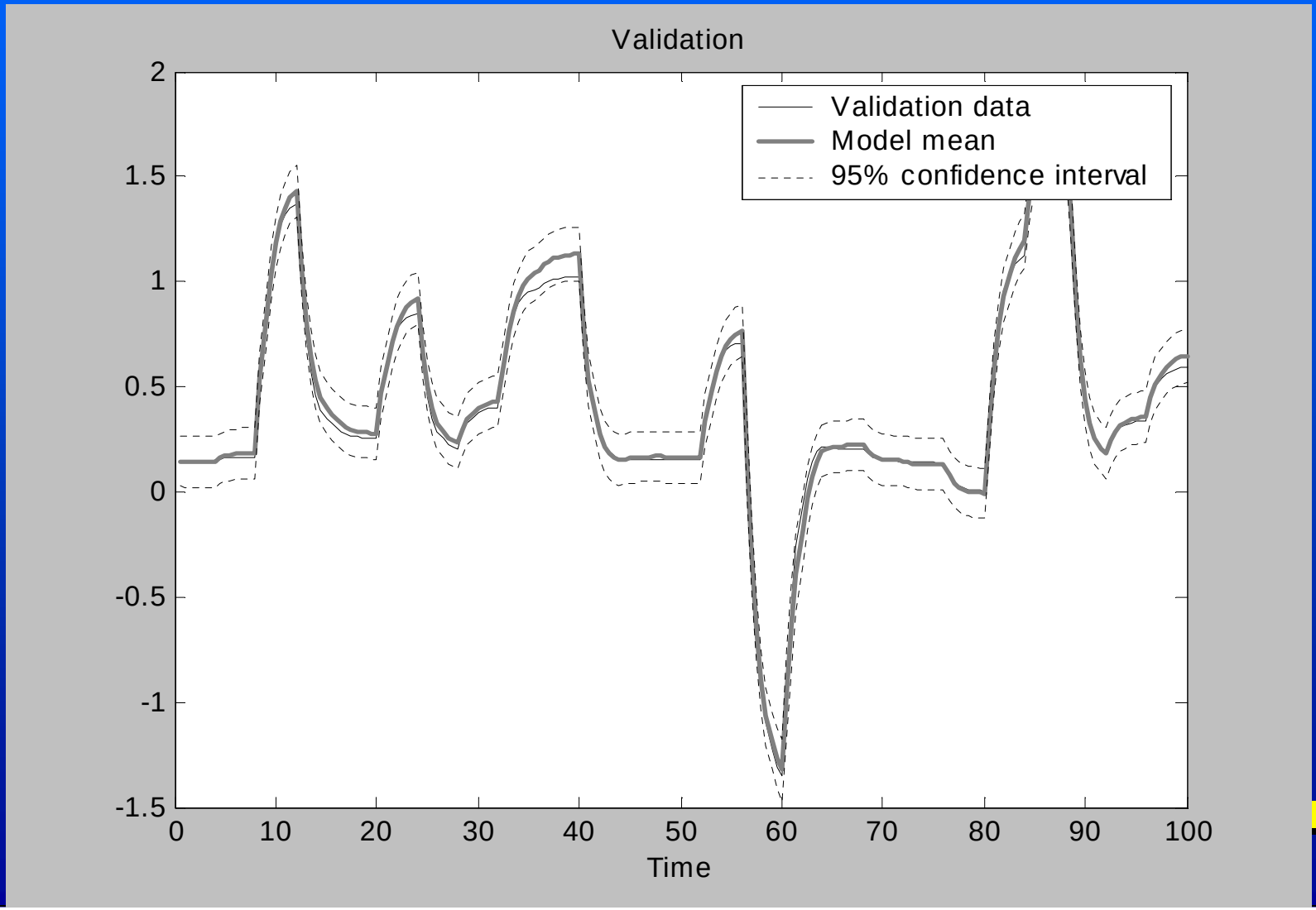


Nonlinearity of the original system y(k+1)=f(u(k),y(k)) (left figure) and GP model (right picture)

Comparison of gradients plots



GP model response on validation data – case 1



Validation of residuals

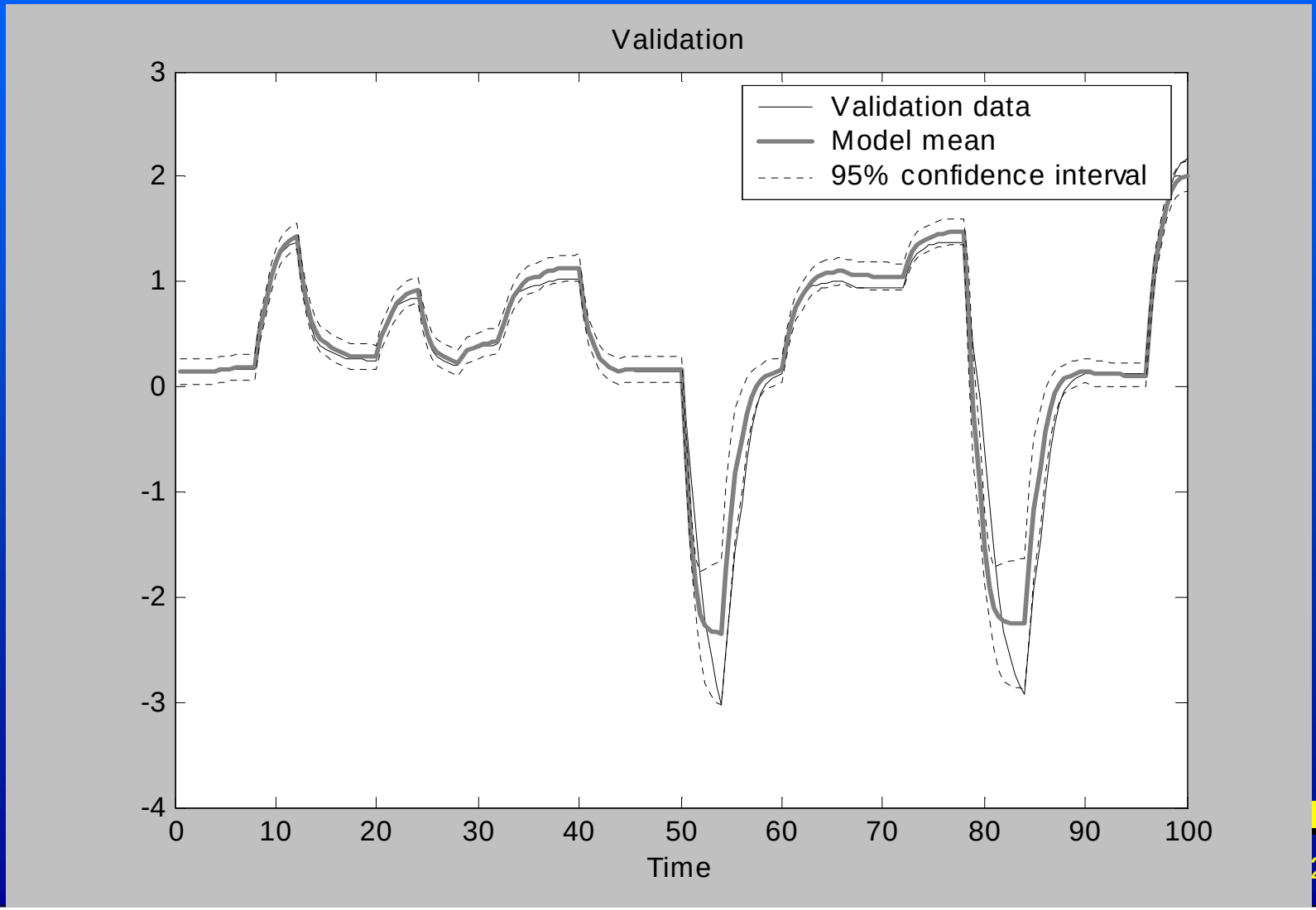
$$MAE = \frac{1}{N}\sum|\hat{\mathbf{y}} - \mathbf{y}| = 0.028$$

$$MSE = \frac{1}{N}\sum(\hat{\mathbf{y}} - \mathbf{y})^2 = 0.0016$$

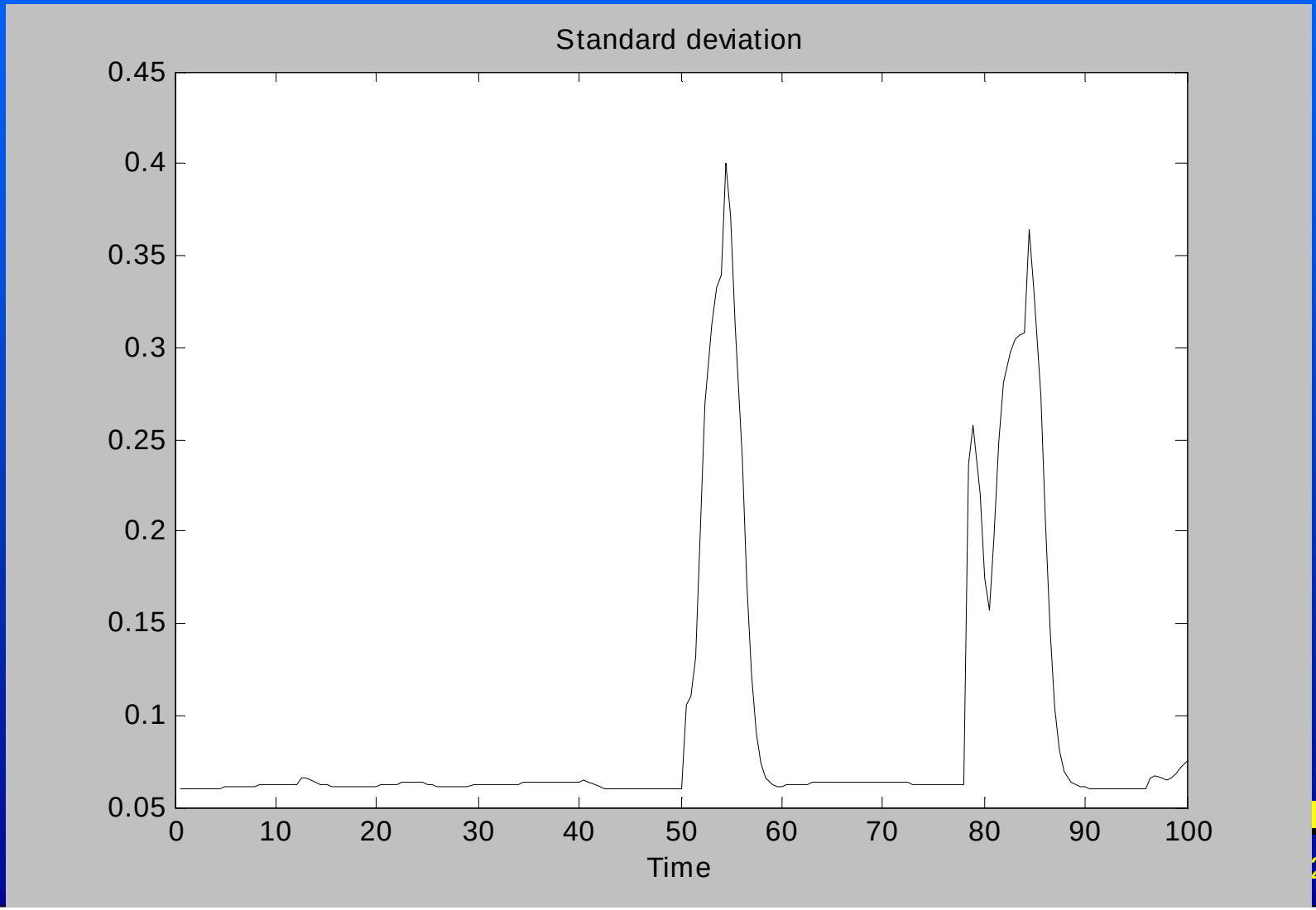
$$LD = \frac{1}{2N}\sum(\log(2\pi) + \log(\sigma^2) + \frac{(\hat{\mathbf{y}} - \mathbf{y})^2}{\sigma^2} = -1.6992$$

- MAE – mean absolute error
- SE – Mean squared error
- LD – log predictive density

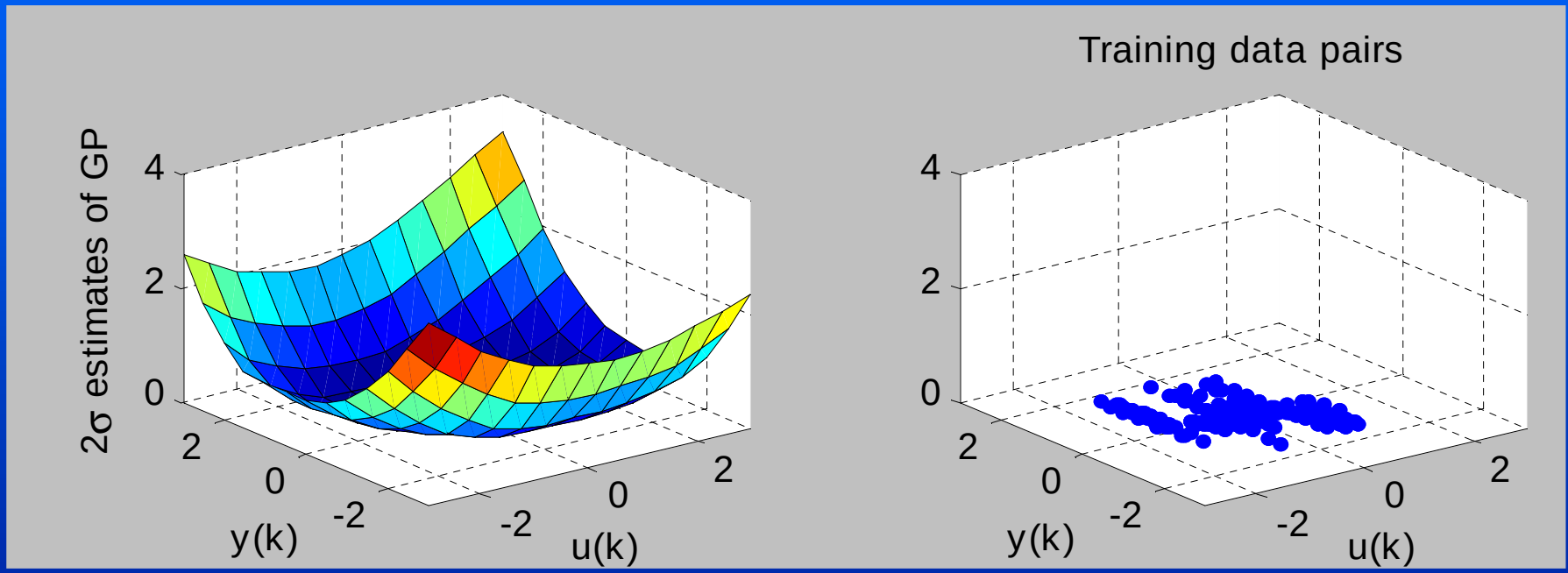
GP model response on validation data – case 2



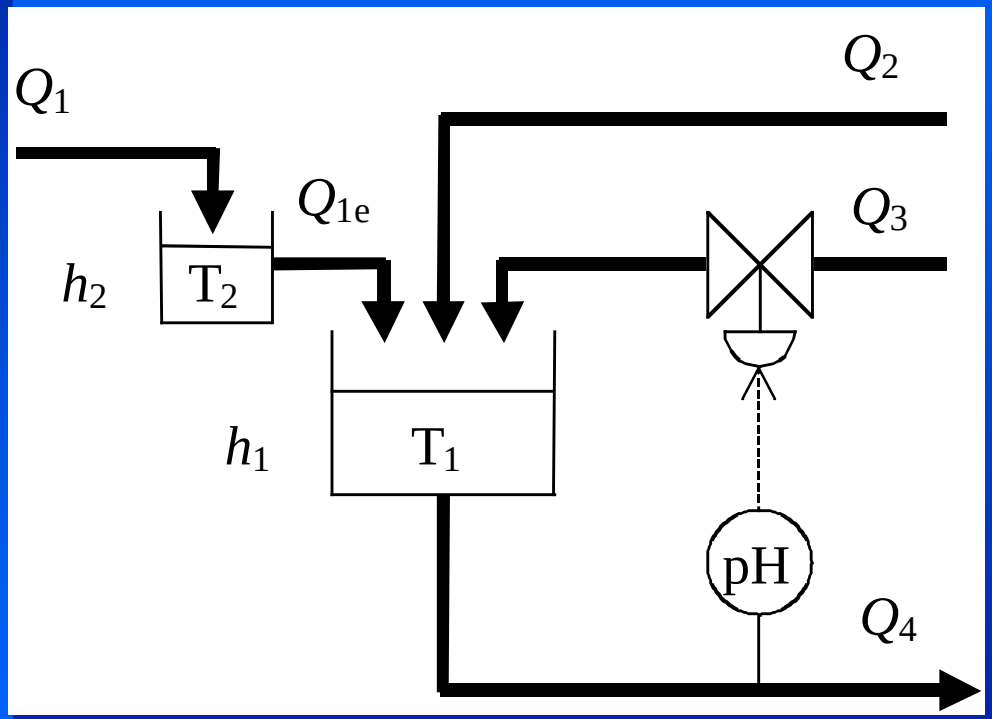
GP model response on validation data – case 2



Model uncertainty



pH process identification example



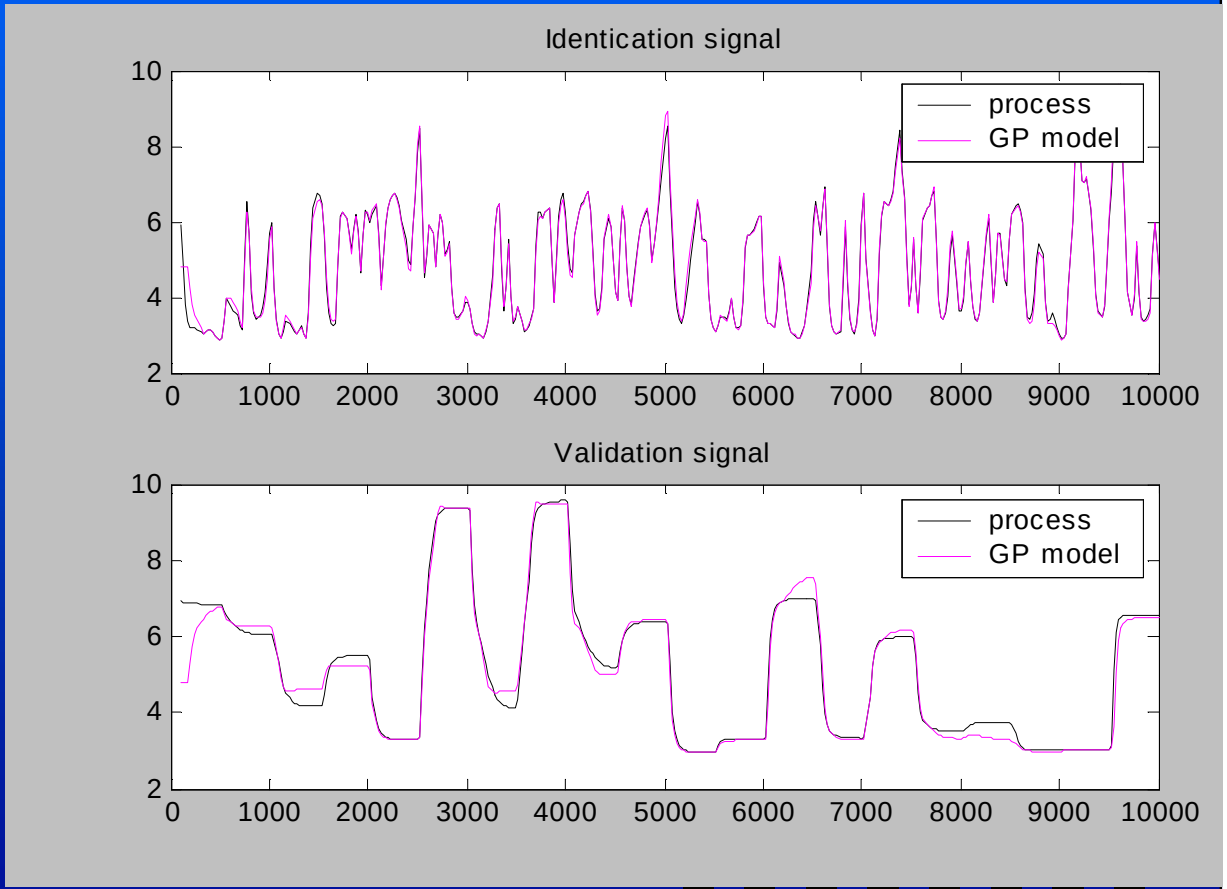
- Benchmark of Henson and Seeborg
- input: base flow rate Q_3
- output: value of pH in outflow

GP model identification

- Covariance function:

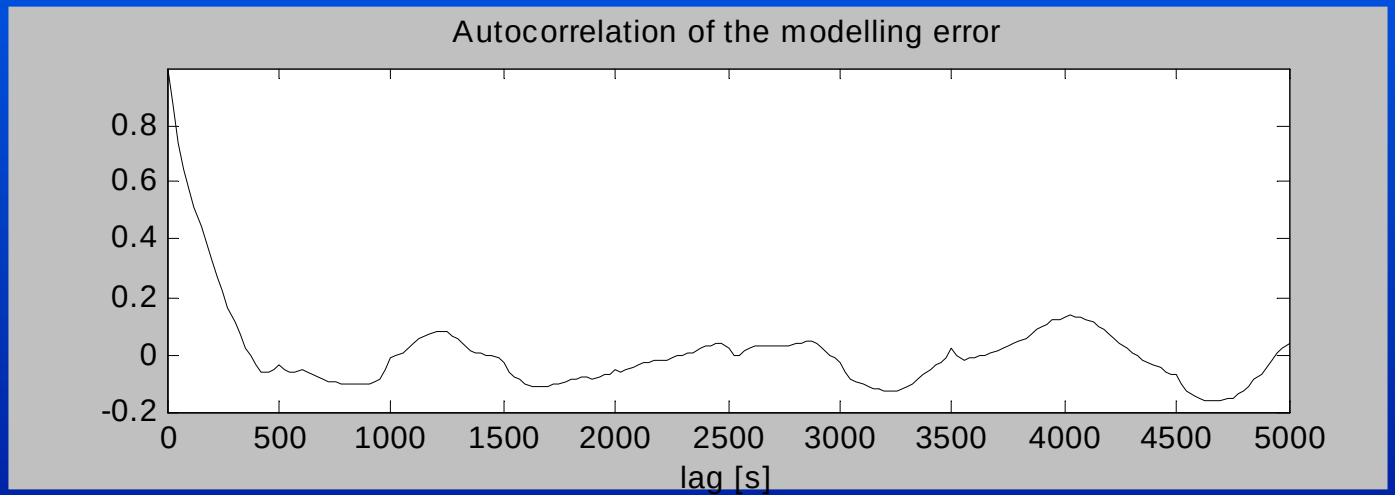
$$C(X,X') = v_1 e^{-\frac{1}{2}\sum_{d=I}^{D} w_d \left(X_d - X'_d\right)^2} + v_0$$

- Regressors: *y(k-1),...,y(k-4), u(k-1),...,u(k-4);*
- 10 hyperparameters for optimisation
- considerable computational load (cca.110 times bigger than in previous 1st order example)

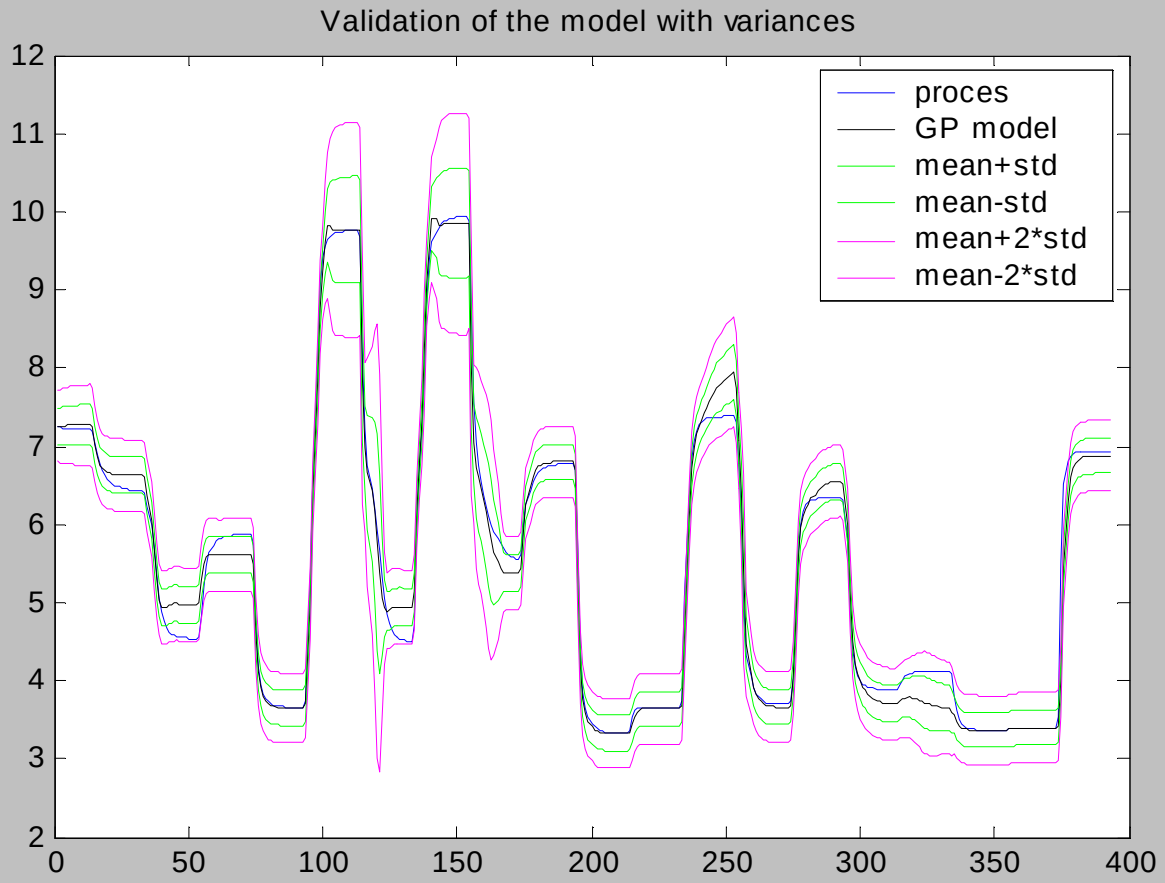


GP model validaton

- MAE=0.1494
- MSE=0.0512

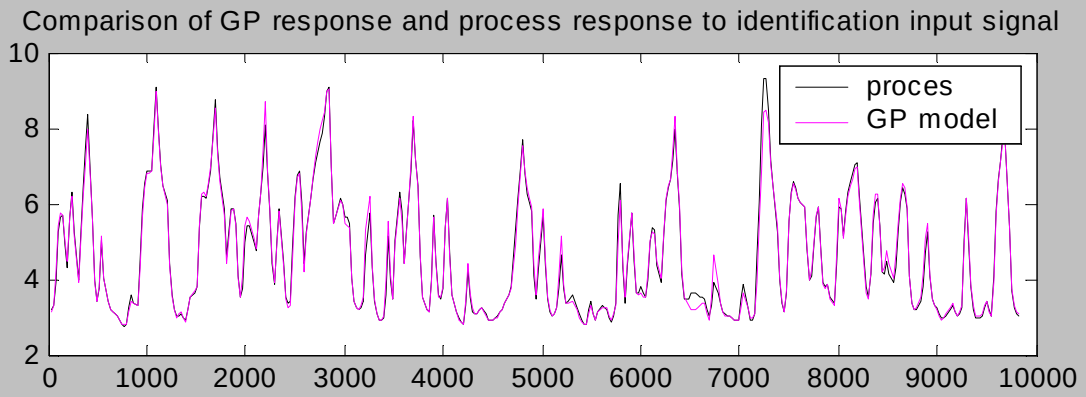


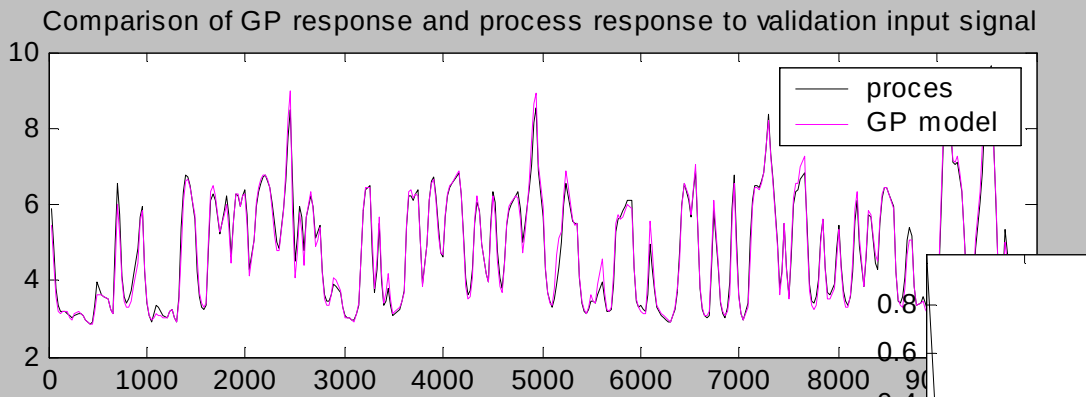
Variances of validation response



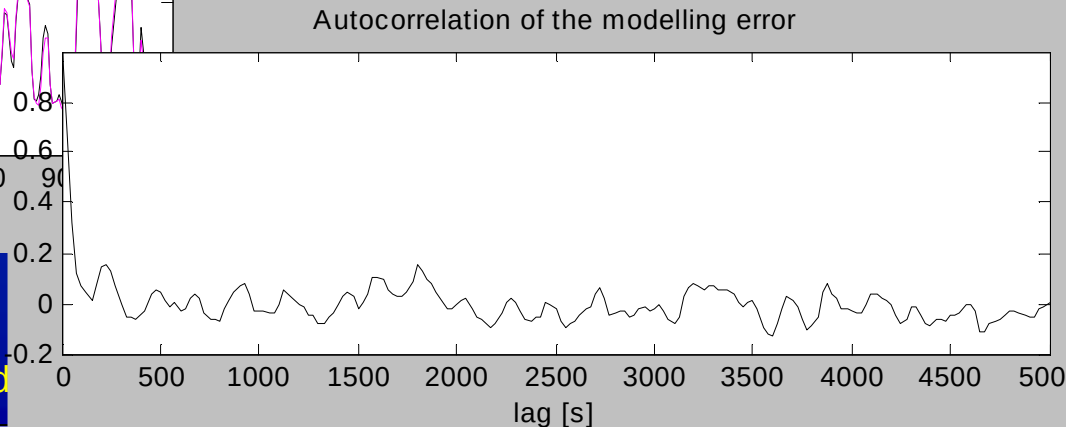
Higher variance means less confident prediction.

Validation results – case 2





Same GP model



Practical aspects of GP model identification

- Nonparametric methods are traditionally more popular with engineers who design control.
- Variances give information about model prediction confidence and trust that we can put in model.
- Computational load – covariance matrix inverse.
- What does model consist of? (input data, output data, hyperparameters, covariance function)