Work Packages for Bachelor and Master Thesis

Tim Oliver Heinz
Department of Mechanical Engineering
University of Siegen
D-57068 Siegen, Germany
tim.heinz@uni-siegen.de

April 29, 2015

Abstract

This document gives some introducing work packages to get familiar with system identification

1 Least squares, training and test error

1.1 Description

To get familiar with system identification it is necessary to get some theoretic introduction. For this read and understand Chapter 3.1 out of [1] about the least squares (LS) method. Chapter 7 including subsection 7.3.1 gives an introduction to optimal model complexity.

Write a Matlab script, to get familiar with the LS-method:

1. generate $N = 20$ input data points in the range $[0, ..., 1]$
2. take a parabolic function ($y(u) = 4 \cdot (u - 0.5)^2$) and calculate the output for the input data. This represents the process data.
3. add some noise to the process output.
4. Create a Regressor X of the order n
5. Calculate the optimal Parameters Θ of the n-th order polynomial using the LS equation
6. calculate the model output using the Regressor X
7. calculate the normalized root mean squared error (NRMSE) between the process and the model output
8. generate new input data for validation with $N_{\text{val}} = 20000$.
9. get the process output using the function in 3
10. Create a Regressor X_{val} of the order n
11. Calculate the model output with Θ out of 6
12. Calculate the NRMSE of the validation data.
13. Loop steps 3 to 13 for different model order n
14. plot the two NRMSE over n
15. plot the model output of the validation data sets.

1.2 Matlab functions

- `linspace(`a`,`b`,`N`)`
  `creates a row vector with` N `equal distance data points bounded by` a `and` b
- `randn(`d_1`,`d_2`,...)`
  `creates a` $d_1 \times d_2 \times \ldots$ `matrix with normal distributed random numbers`
- `mldivide(X,y) Calculates the LS solution for the Regressor` X `an the output` y
- `plot(u,y) plots` y `over` u

1.3 Estimated Workload

20h-30h

References

[1] Oliver Nelles. Nonlinear system identification, 2002.

2 Weighted least squares

2.1 Description

In ch. 3.1.6 out of [1] the weigthed LS (WLS) method is presented. Use the process out of sec. 1 ($y = (0.5 - u)^2$) and add noise depending of the input value. For example: No noise at $u = 0$ and maximum noise at $u = 1$. Choose the maximum, so that the underlying process can still be observed.

Define a weighting matrix, which weights the disturbed data less than the undisturbed data.

Calculate the LS and WLS solution for a model of 0-th, first and second order. Compare the model output and the validation error and test error of the two estimation methods.

2.2 Matlab functions

- `diag(`q`)`
  `creates a diagonal matrix out of the vector` q
- `pinv(`X`)`
  `calculates the pseudo inverse of a matrix` X `(recommended in this context)`

2.3 Estimated Workload

$\approx$ 20h

References

[1] Oliver Nelles. Nonlinear system identification, 2002.

3 RBF/NRBF Net

3.1 Description

In contrast to the last sections, this section gives a short introduction about a radial base function (RBF) net / normalized radial base function(NRBF) net.

Generate process data according to sec. 1. Calculate three functions($g_1(u)$, $g_2(u)$ and $g_3(u)$) for 1 dimensional Gaussian placed on $\mu_1 = 0.3$, $\mu_1 = 0.5$ and $\mu_1 = 0.7$. and $\sigma = 0.3$

1. Start with a RBF net

$$g_i(u) = e^{-\frac{1}{2}\left(\frac{u-\mu_i}{\sigma}\right)^2}$$

   Create a regressor of the three functions:

$$X = \begin{bmatrix} g_1(u_1) & g_2(u_1) & g_3(u_1) \\ g_1(u_2) & g_2(u_2) & g_3(u_2) \\ \vdots & \vdots & \vdots \\ g_1(u_N) & g_2(u_N) & g_3(u_N) \end{bmatrix}$$

   Calculate the least squares solution and visualize the results.

2. normalize the g_i to get the NRBFs.

$$n_i(u) = \frac{g_i(u)}{\sum\limits_i g_i(u)}$$

   Create a regressor of the three functions:

$$X = \begin{bmatrix} n_1(u_1) & n_2(u_1) & n_3(u_1) \\ n_1(u_2) & n_2(u_2) & n_3(u_2) \\ \vdots & \vdots & \vdots \\ n_1(u_N) & n_2(u_N) & n_3(u_N) \end{bmatrix}$$

   Calculate the least squares solution and visualize the results.

3. Now take the functions out of step 2 as weighting function for estimating a polynomial of 1st order (regression line). For each weighting function $n_i(u)$ calculate a separate WLS solution. To calculate the output, the three LS solutions have to be aggregated.

3.2 Matlab functions

3.3 Estimated Workload

30h

References

[1] Oliver Nelles. Nonlinear system identification, 2002.