Laboratory 5

Nonparametric system identification of house temperature forecasting

System and data description. We consider again the domotic system of Laboratory 4 and the corresponding data set stored in `house.mat`.

Data plotting. Plot the data y^N and u^N.

Data detrending. Detrend the data.

Model structure design. We perform the identification of the system using a black-box modeling approach. More precisely, we consider the following four ARX model structures (approximations of nonparametric BJ models):

Model structure	Degrees
$\mathcal{M}_1(\theta): \quad A(z)\boldsymbol{y}(t) = B(z)u(t-1) + \boldsymbol{e}(t)$	$n_A = 50,\ n_B = 50$ $K_A = \lambda_A K_{DI,\beta_A}$ $K_B = \lambda_B K_{DI,\beta_B}$ $\tilde{\eta} = [\,\lambda_A\ \beta_A\ \lambda_B\ \beta_B\ \sigma^2\,]^T$
$\mathcal{M}_2(\theta): \quad A(z)\boldsymbol{y}(t) = B(z)u(t-1) + \boldsymbol{e}(t)$	$n_A = 50,\ n_B = 50$ $K_A = \lambda_A K_{TC,\beta_A}$ $K_B = \lambda_B K_{TC,\beta_B}$ $\tilde{\eta} = [\,\lambda_A\ \beta_A\ \lambda_B\ \beta_B\ \sigma^2\,]^T$
$\mathcal{M}_3(\theta): \quad A(z)\boldsymbol{y}(t) = B(z)u(t-1) + \boldsymbol{e}(t)$	$n_A = 50,\ n_B = 50$ $K_A = \lambda_A K_{SS,\beta_A}$ $K_B = \lambda_B K_{SS,\beta_B}$ $\tilde{\eta} = [\,\lambda_A\ \beta_A\ \lambda_B\ \beta_B\ \sigma^2\,]^T$
$\mathcal{M}_4(\theta): \quad A(z)\boldsymbol{y}(t) = B(z)u(t-1) + \boldsymbol{e}(t)$	$n_A = 50,\ n_B = 50$ $K_A = \lambda_A B_{\beta_A} B_{\beta_A}^T$ $K_B = \lambda_B B_{\beta_B} B_{\beta_B}^T$ $\tilde{\eta} = [\,\lambda_A\ \lambda_B\ \sigma^2\,]^T$

where we made explicit the dependence of K_{DI}, K_{TC} and K_{SS} on β. Matrix $B_\beta = [\,\theta^{(1)} \dots \theta^{(m)}\,]$ contains the random features of Example approximating the SS kernel. Choose $m = 5$.

Hyperparameters estimation. Adopting the Bayesian perspective, we estimate the optimal kernels K_A, K_B (i.e. the optimal augmented hyperparameters vector $\tilde{\eta}_i^{OPT}$, $i = 1, 2, 3, 4$) minimizing the negative marginal log-likelihood. Use the given (modified) function `arxRegul.m` for $i = 1, 2, 3$:

the third output is the negative marginal log-likelihood evaluated for the optimal augmented hyperparameters vector, i.e. the value $\ell(y^N; \tilde{\eta}_i^{OPT})$; the fourth output are the corresponding degrees of freedom, i.e. $d_f(\tau, \tilde{\eta}_i^{OPT})$. Use the given function `arxRegulRF2.m` for $i = 4$: the input are the data, the orders of the polynomials and the number of random features m; the output is the same of the one in `arxRegul.m`.

Training step. Compute the kernel-based PEM estimate using $\mathcal{M}_i$, $i = 1, 2, 3, 4$. Let $\mathcal{M}_i(\hat{\theta}_K)$, $i = 1, 2, 3, 4$, be the estimated models, respectively. Plot the corresponding impulse responses of $A(z)$ and $B(z)$.

Validation step

Marginal likelihood evaluation. Compare the different values for the negative marginal log-likelihood:

$$\ell(y^N; \tilde{\eta}_i^{OPT}) \ \ i = 1, 2, 3, 4.$$

Hold-out cross-validation. Let $\hat{\theta}_K^{CV}$ be the kernel-based PEM estimate using $\mathcal{M}_i$, with $i = 1, 2, 3, 4$, and the data:

$$u_T = [\ u(1) \quad \dots \quad u(\tfrac{N}{2})\ ]^T, \ \ y_T = [\ y(1) \quad \dots \quad y(\tfrac{N}{2})\ ]^T.$$

Compute the fit percent of the h-step ahead prediction and using the data

$$\begin{aligned} u_V &= [\ u(\tfrac{N}{2}+1) \quad \dots \quad u(N)\ ]^T, \\ y_V &= [\ y(\tfrac{N}{2}+1) \quad \dots \quad y(N)\ ]^T, \end{aligned}$$

with $h = 1, 2, 3, 4$.

Residual analysis. According to the Fisherian viewpoint, perform the residual analysis for the models using the autocorrelation test and the cross-correlation test.

SURE criterium. According to the Fisherian viewpoint, compute the SURE index (use the given Matlab function `surek.m`).

Computation time. Compute the amount of time required to estimate $\tilde{\eta}_i^{OPT}$ for $i = 3, 4$, i.e. the model structure corresponding to the SS kernel and its approximation using the random features, respectively. In doing that, use the Matlab functions `tic`, `toc`, e.g.

```
tic
 instruction;
toc
```

which measure the time required in order to execute `instruction`. In order to obtain a more realistic estimate of the computational time: for each method, write a for-loop of 20 iterations and in each iteration compute the time required to estimate $\tilde{\eta}_i^{OPT}$. Then, compute the mean and the standard deviation of such execution time.

Question 1: In view of the results found in the validation step, which is the best model describing the house temperature? Motivate your choice.

Question 2: In view of the results found in this laboratory and in Laboratory 4, which is the best model describing the house temperature? Motivate your choice.