

## Lecture 1: Subspace methods for linear, time-invariant system identification

Lecturer: Nikolai Matni

Scribes: Duc Nguyen

**Disclaimer:** *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

## 1 System identification

Dynamical systems find applications in diverse settings from mechanical engineering, robotics to reinforcement learning. As a motivating example, imagine that you are operating a data center with many computers running concurrently. For these computers to function properly and not catch fire, you need to cool them with a cooling system. At the same time, for your business to make profits, you have to take into account the cost of cooling such a large space. Ideally, you want to spend just enough energy (and therefore electricity bill) in maintaining the computers at a reasonable temperature.

To solve the above problem, you have to take into account multiple groups of variables. Firstly, you need to control the state (the temperature of the machines) of the system. Secondly, you need to tune the input to inject into the system (energy to increase the speed of cooling fans, etc).

In addition, noises are also a possible component of the dynamical system. Process noises are fluctuations that arise from the dynamical system itself. Sensor noises are errors that arise from measurement inaccuracies (for example, your data center's thermostat might work perfectly all the time). The control inputs as well as the noises both affect the state of the system. Unfortunately, this relationship is often unknown.

System identification then is the task of learning, given input-output measurements, the relationship between the control inputs and the state of the system of interest. By estimating the underlying model of the dynamical system accurately, one can devise fine-tune control policies that keep the system stable or to operate the system "optimally".

### 1.1 Linear dynamical system

In system identification, we are given data of input and output measurements. However, the underlying model is ultimately a black box. A class of system identification approaches known as model based methods make assumptions about the parametric form of the underlying dynamics. One of these parametric models that has found great practical use and enjoyed a wealth of research effort is the linear dynamical system.

In linear dynamical systems, the relationship between the control input and the state of the system is represented with a linear equation and the dynamic is discretized over time. Although the true system dynamic might not be linear, linear dynamical systems are still widely used because using a finer discretization scheme and using more parameters often allow the model to approximate better the true underlying dynamic. Furthermore, many problems involving linear dynamical systems either have closed form solutions or can be solved efficiently.

### 1.2 Formulation

In linear dynamical systems, the relationship between the state, output and the input is represented using the following difference equations:

$$\begin{aligned}x_{t+1} &= Ax_t + Bu_t + w_t \\y_t &= Cx_t + Du_t + v_t\end{aligned}\tag{1}$$

Where, at time step  $t$ ,  $x_t \in \mathcal{R}^n$  is the state,  $u_t \in \mathcal{R}^m$  is the input,  $w_t \in \mathcal{R}^n$  is process noise,  $y_t \in \mathcal{R}^p$  is the state measurement and  $v_t \in \mathcal{R}^p$  is the sensor noise. Matrices  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times m}$  are the state parameters and matrices  $C \in \mathbb{R}^{p \times n}$ ,  $D \in \mathbb{R}^{p \times m}$  are the measurement parameters.

Another commonly seen formulation of linear dynamical system assumes that there is no process nor sensor noise. With such assumptions, the difference equations are as follows:

$$\begin{aligned}x_{t+1} &= Ax_t + Bu_t \\y_t &= Cx_t + Du_t\end{aligned}\tag{2}$$

At other times, by assuming that we have perfect measurement of the system state, the system is said to have fully observed state (in contrast to partially observed states as formulated above). The two difference equations as seen before can be further condensed into just one equation.

$$x_{t+1} = Ax_t + Bu_t + w_t\tag{3}$$

It is notable that linear dynamical systems with fully observed state are easier to solve than those with partially observed state. For example, systems with fully observed state can be solved via a least-square approach by minimizing the cost function  $\hat{A}, \hat{B} = \operatorname{argmin}_{A,B} \sum_{t=1}^T (x_{t+1} - Ax_t - Bu_t)^2$ . On the other hand, the corresponding least-square cost function for partially observed state is non-convex as  $\sum_{t=0}^T (y_{t+1} - Cx_{t+1} - Du_{t+1})^2 = \sum_{t=1}^T (y_{t+1} - CAx_t - CBu_t - Du_{t+1})^2$  contains both  $CA$  and  $CB$ .

### 1.3 Useful definitions

We have the following useful definitions. Note, however, that depending on the system model, each of these definition will have different mathematical significance. We will explicitly state what this significance is where ever appropriate.

**Definition 1.** *Time invariance: Time invariance is a property of the dynamical system where the function that maps from the input  $u_t$  to the state  $x_t$  of the system at any time does not depend on the time  $t$ . In linear dynamical systems, since the matrices  $A$  and  $B$  are assumed to constant, the time invariant property holds.*

**Definition 2.** *Transfer function: a function that theoretically models the system output for each control input.*

**Definition 3.** *State space model: a state-space representation is a mathematical model of a physical system as a set of input, output and state variables related by first-order differential equations or difference equations.*

**Definition 4.** *Controllability: A linear dynamical system such as one defined by (2) is controllable if given a start state  $x_0 \in \mathbb{R}^n$ , for every target state  $x_* \in \mathbb{R}^n$ , there exists a sequence of control inputs  $U = \{u_1, u_2, \dots, u_t\}$  that drives the state from  $x_0$  to  $x_*$  in some finite time  $t$*

**Definition 5.** *Observability: A linear dynamical system such as one defined by (2) is observable if given a trajectory of input-measurement pairs  $(U, Y) = \{(u_1, y_1), (u_2, y_2), \dots, (u_t, y_t)\}$ , it is possible to determine the state  $x_t$  for  $0 \leq t \leq t$ .*

**Definition 6.** *Persistence of excitation: intuitively, in order to obtain a good estimate of a (parametric) model, the input signal has to be "rich" enough to excite all "modes" of the system. Mathematically, an input  $u(t)$  is persistently exciting of order  $n$  if:*

*Firstly, the following must exists for all  $\tau$*

$$r_u(\tau) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=1}^{N-\tau} u(t+\tau)u^\top(t)$$

*Note that if  $u(t)$  is ergodic, then  $r_u(\tau) = \mathbb{E}[u(t+\tau)u^\top(t)]$*

*And define*

$$R_u(n) = \begin{bmatrix} r_u(0) & r_u(1) & \dots & r_u(n-1) \\ r_u(1) & r_u(0) & \dots & r_u(n-2) \\ \dots & \dots & \dots & \dots \\ r_u(n-1) & r_u(n-2) & \dots & r_u(0) \end{bmatrix}$$

Then secondly,  $R_u(n)$  must be positive definite

## 2 Alternative methods

Before discussing subspace methods however, it helps to briefly introduce the one popular alternative approach to system identification, that is prediction error methods. It is a class of methods that directly learn the mapping from control  $u$  to output  $y$ . One simple and well known method is the AR (auto-regressive) algorithm [1]. In the simple scalar output and scalar control case, we can write the relation between the output  $y$  and the control  $u$  as:

$$A(q^{-1})y_t = B(q^{-1})u_t + \epsilon_t \quad (4)$$

where  $A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a}$  and  $B(q^{-1}) = b_1q^{-1} + \dots + b_{n_b}q^{-n_b}$  with fixed but unknown coefficients  $\{a_1, \dots, a_{n_a}, b_1, \dots, b_{n_b}\}$ . And  $\epsilon_t$  is a random process noise. One intuitive way to understand this model is that the output at every time step  $t$  is a linear function of the previous inputs and outputs.

By defining  $\theta = [a_1, \dots, a_{n_a}, b_1, \dots, b_{n_b}]^T$  and  $\phi_t^T = [-y_{t-1}, \dots, -y_{t-n_a}, u_{t-1}, \dots, u_{t-n_b}]^T$ , we can rewrite equation (4) as:

$$y(t) = \phi_t^T \theta + \epsilon_t \quad (5)$$

One viable way to solve the system identification problem then is via least square:

$$\hat{\theta} = \underset{\theta=\{a_1, \dots, a_{n_a}, b_1, \dots, b_{n_b}\}}{\operatorname{argmax}} \sum_{t=1}^T (y_t - \phi_t^T \theta)^2 \quad (6)$$

## 3 Subspace-based methods

The statistical properties of prediction error methods are well known and are connected to maximum likelihood estimation. However, large order systems and multivariate inputs/ outputs make the difference equations in prediction error methods more cumbersome to work with. This is because we have to learn a multivariate hidden variables for each time step in the difference equations analogous to (4). Furthermore, numerical reliability might be poor for large system order [3]. Further more, prediction error models methods with "black box" models where the input-output relation is captured by the difference equations and we don't make any assumptions about the parametric form of the dynamic. However, we have seen that a linear approximation of dynamical systems enjoys various benefits and can approximate well complex systems. Thus, in more complex systems, the preferred model should be a state-space model and subspace methods be the methods of choice for system identification.

### 3.1 Formulation

Note that in the following analysis, our linear dynamical system is represented by (1), reproduced here for convenience:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t \\ y_t &= Cx_t + Du_t + v_t \end{aligned}$$

### 3.2 Summary of approach

Recall that the goal of system identification is, with a pair of input and measurement sequence  $U = [u_0, \dots, u_T]$  and  $Y = [y_0, \dots, y_T]$ , to estimate  $\hat{A}, \hat{B}, \hat{C}, \hat{D}$  that approximate  $A, B, C, D$ . One classic approach in subspace methods, popularized as the Ho-Kalman algorithm, consists of the following steps:

1. Convert the difference equations into an algebraic formulation
2. Estimate the transfer function and then constructing the Hankel matrix
3. Recover the system parameters from the Hankel matrix through a process known as realization.

### 3.3 Algebraic formulation

Note that with (1), however, the difference equations are unwieldy. One trick is to transform these difference equations into algebraic equations that are easier for us to work with. This could be accomplished using the Z-transform. Applying the z-transform to the state difference equation of (1), gives us:

$$\mathbf{x} = (zI - A)^{-1}Bu + (zI - A)^{-1}w$$

Substituting this back into the measurement difference equation of (1) gives:

$$\mathbf{y} = [C(zI - A)^{-1}B + D]u + C(zI - A)^{-1}w + v$$

We can write the above equation compactly as  $\mathbf{y} = G(z)u + \delta(z)$ . Note that  $G(z) = C(zI - A)^{-1}B + D$  is called the transfer function and  $\delta(z) = C(zI - A)^{-1}w + v$ .

### 3.4 Estimate the transfer function

The transfer function defined previously is significant because by estimating  $G$ , one can recover the state and measurement parameters. For example, one can directly read off  $D$  from an estimate of  $G_0$ . To see why, note that:

$$\begin{aligned} G(z) &= C(zI - A)^{-1}B + D \\ &= \frac{1}{z}C\left(I - \frac{1}{z}A\right)^{-1}B + D \\ &= \frac{1}{z}C \sum_{i=0}^{\infty} \left(\frac{1}{z}A\right)^i B + D \\ &\equiv \sum_{t=0}^{\infty} \frac{1}{z^t} G_t \end{aligned}$$

Where the last line comes from the definition of the unilateral Z-transform with  $G_t$  denoting the **impulse response** at time  $t$ . By pattern matching, one can see that  $G_0 = D$  and  $G_t = CA^{t-1}B$  (Note that there is a  $\frac{1}{z}$  term in the 3rd equality that leads to the index being off by 1, or  $t - 1$ ). In addition, we have an alternative formulation. As we are dealing with a linear causal system, we can write  $y_t$  in terms of the inputs  $u_t$  via a convolution operation as follows:

$$\begin{aligned}
y_t &= \sum_{\tau=0}^t G_{\epsilon-\tau} u_\tau + \delta_\epsilon \\
&= [G_t \quad G_{t-1} \quad \dots \quad G_0] \begin{bmatrix} u_0 \\ u_1 \\ \dots \\ u_t \end{bmatrix} + \delta_t
\end{aligned} \tag{7}$$

Now suppose that we've run an experiment and generated a trajectory pair  $(u_t, y_t)_{t=0}^{T+N}$  and that we have picked  $T$  large enough to capture most of the effect of past inputs on the current state. This works because if  $A$  is stable (maximum eigenvalue of  $A$  is less than 1), then such a  $T$  exists. Aggregating (7) for  $t = T, \dots, T + N$  gives:

$$[y_T \quad y_{T+1} \quad \dots \quad y_{T+N}] \approx [G_0 \quad G_1 \quad \dots \quad G_T] \begin{bmatrix} u_T & u_{T+1} & \dots & u_{T+N} \\ u_{T-1} & u_T & \dots & u_{T+N-1} \\ \dots & \dots & \dots & \dots \\ u_0 & u_1 & \dots & u_T \end{bmatrix}$$

As we are trying to estimate  $G$ , one viable solution is to solve the following optimization problem:

$$\hat{G} = \arg \min_G \|Y(z) - G(z)U(z)\|_F^2 = U^\dagger Y$$

Where  $U^\dagger = U^\top(UU^\top)^{-1}$  is the pseudo inverse. Of course, this assumes that  $(UU^\top)^{-1}$  exists or that  $U$  has full row rank. And if  $(UU^\top)^{-1}$  exists, the inputs are also called persistently exciting.

Unfortunately, even if  $U$  has full row rank, stability issues still exist since we are not solving directly for  $G$  but we are doing a "truncated" approach (only account for time steps from  $T$  to  $T + N$ ). We can expose the problem by noting that:

$$Y_{T,N} = G_{T,N}U_{T,N} + \delta + T$$

where  $\delta$  is the error term and  $T$  is the tail term that accounts for the dynamic for time  $t = T + N$  to  $\infty$ . Then:

$$\hat{G} = Y_{T,N}U_{T,N}^\top(U_{T,N}U_{T,N}^\top)^{-1} = G_{T,N} + (\delta + T)U_{T,N}^\top(U_{T,N}U_{T,N}^\top)^{-1}$$

If  $U_{T,N}^\top(U_{T,N}U_{T,N}^\top)^{-1}$  is ill conditioned, we will magnify the error in the estimation.

### 3.5 Recovering the model parameters

Suppose that we have estimated  $\hat{G} \approx G$ , the question is then how do we recover  $A, B, C$ ? The first issue is that we can only hope to recover the state parameters  $A, B$  up to some similarity transformation. To see why, suppose we have an invertible transformation matrix  $T \in \mathcal{R}^{n \times n}$  and consider the parameters  $(T^{-1}AT, T^{-1}B, CT, D)$  in place of  $(A, B, C, D)$ , we get an equivalent system:

$$\hat{y} = CT(zI - T^{-1}AT)^{-1}T^{-1}B + D = C(zI - A)^{-1}B + D = y$$

The second issue is the choice of the dimension of the state space or how does one decide on the order of the system?

As we will see, in the simple noiseless case, we can recover all model parameters exactly. Firstly, define the Hankel matrix from the impulse response estimates  $G_t$ , truncated by two time boundaries  $T_1, T_2$  (one can think of  $T_2 = T_1 + N$ , similar to the notations in the previous subsection) as follows:

$$\begin{aligned}
H_{T_1, T_2}(G) &= \begin{bmatrix} G_1 & G_2 & \dots & G_{T_2} \\ G_2 & G_3 & \dots & G_{T_2+1} \\ \dots & \dots & \dots & \dots \\ G_{T_1} & G_{T_1+1} & \dots & G_{T_1+T_2-1} \end{bmatrix} \\
&= \begin{bmatrix} CB & CAB & \dots & CA^{T_2-1}B \\ CAB & CA^2B & \dots & CA^{T_2}B \\ \dots & \dots & \dots & \dots \\ CA^{T_1-1}B & CA^{T_1}B & \dots & CA^{T_1+T_2-2}B \end{bmatrix} \\
&= \begin{bmatrix} C \\ CA \\ \dots \\ CA^{T_1-1} \end{bmatrix} [B \quad AB \quad \dots \quad A^{T_2-1}B]
\end{aligned}$$

The block matrix on the left is called the observability  $\mathcal{O}$  matrix and the block matrix on the right is called the controllability matrix  $\mathcal{C}$ .

Recall the definition of controllability and observability in section 1.3. Focusing on linear dynamical systems with fully observed state and no noise, a system is controllable if and only if  $\mathcal{O}$  is full-rank or  $\text{rank}(\mathcal{O}) = n$ . On the other hand, a linear system with fully observed state and no noise is controllable if and only if  $\mathcal{C}$  is full-rank or  $\text{rank}(\mathcal{C}) = n$ . As we will see, one can recover  $A, B, C$  via factorizing the Hankel matrix. The following sub-sections will look at the noiseless case, followed by the imperfect data case.

### 3.5.1 The noiseless case

Recall that besides recovering the model parameters, we also wish to learn the system order  $n$ . Furthermore, we also want to estimate parameters that represent a system of the lowest possible order  $n$ . This is also known as **minimal realization**.

In fact, minimal realization has some connection to controllability and observability: given any transfer function, any state-space model that is both controllable and observable, and has the same input-output behaviour as the transfer function is said to be a minimal realization of the transfer function.

Assume that  $(A, B)$  is controllable and  $(A, C)$  is observable and  $T_1, T_2 \geq n$ , then  $\text{rank}(H_{T_1, T_2}(G)) = \min(\text{rank}(\mathcal{O}), \text{rank}(\mathcal{C})) = n$ . In other words, via SVD, we can recover exactly the system order by excluding the subspace associated with zero singular values. Note that in the original subspace based method proposed by Kalman and Ho, the authors actually proposed factorizing the Transfer function matrix  $G$  with QR factorization. However, SVD was later proposed as a tool to reduce the sensitivity to errors in the measured impulse responses:

$$\begin{aligned}
H_{T_1, T_2} &= U\Sigma V^\top \\
&= (U\Sigma^{1/2})(\Sigma^{1/2}V^\top) \\
&\equiv \mathcal{O}\mathcal{C}
\end{aligned}$$

And recall that by definition,  $\mathcal{O} \equiv \begin{bmatrix} C \\ CA \\ \dots \\ CA^{T_1-1} \end{bmatrix}$  and  $\mathcal{C} \equiv [B \quad AB \quad \dots \quad A^{T_2-1}B]$ . Therefore, we can read

off  $C$  and  $B$  by inspecting the first row block and column block of  $\mathcal{O}$  and  $\mathcal{C}$  respectively.

Lastly, to recover  $A$ , we notice that  $\mathcal{O}$  has a shift property where  $\mathcal{O}_{1:T_1-2}A = \mathcal{O}_{2:T_1-1}$  where the  $i : j$  indices indicate the submatrix of that span from the  $i^{\text{th}}$  to  $j^{\text{th}}$  block of  $\mathcal{O}$ . Solving the above equation gives us  $A$ .

### 3.5.2 The noisy case

Unfortunately, the previous analysis applies only to the case where we can estimate  $G$  exactly. Suppose we can only obtain  $\hat{G}$  which is a noisy estimate of  $G$ , that means our constructed Hankel matrix will be a noisy estimate of the true Hankel matrix as well.

$$H_{T_1, T_2}(G) = \mathcal{O}\mathcal{C} + E$$

Where  $E$  is the error term. If  $E$  is white noise, which is a reasonable expectation, then  $H_{T_1, T_2}(G)$  is going to be full rank. In that case, SVD will not automatically recover  $n$ . There are several viable solutions to obtain a better estimate  $\hat{G}$ :

1. One can use SVD and use some sort of thresholding to remove the "small" singular values. However, there is no clear way to select the "optimal" number of significant singular values.
2. Solving the following regularized optimization problem:

$$\min_G \|Y - GU\|_F^2 + \lambda \text{rank}(H)$$

Intuitively, this means finding the lowest rank approximation of the Hankel matrix. Unfortunately, in its original form, this optimization problem is highly non-convex. Fortunately, convex relaxation techniques [2] give us an easier convex optimization problem:

$$\min_G \|Y - GU\|_F^2 + \lambda \|H_{T_1, T_2}(G)\|_*$$

where the nuclear norm is defined as:  $\|X\|_* = \sum_{i=1}^r \sqrt{\lambda_i(XX^T)}$  where  $\text{rank}(X) = r$

## 4 Pre-lecture reading

[3] is a survey of system identification methods for learning linear, time-invariant causal dynamical system. These techniques can be broadly classified into two classes: realization based techniques and direct techniques.

### 4.1 Realization based methods

It is notable that the previous sections cover up to section 3.1 in [3]. The author would classify the approach outlined above as a realization based approach. That is, via estimating the impulse responses and forming the Hankel matrix  $H$ , one learns the model parameters  $A, B, C, D$ .

### 4.2 Direct methods

Another class of algorithms explained in the paper is called direct methods. Whereas in the realization based approach, one has to form the Hankel matrix (that involves estimating the impulse responses,  $G_t$ ), direct methods bypass this step and directly learns the mapping from the input to the responses. The dynamical system is captured by the following equation:

$$Y_N = \Gamma_N X + \Phi_N U \quad (8)$$

where  $Y_N = [y_1, y_2, \dots, y_N]$ ,  $X = [x_1, x_2, \dots, x_N]$  and  $U = [u_1, \dots, u_N]$ ,  $\Gamma_T$  is the controllability matrix as seen before and  $\Phi_T$  is a block lower diagonal matrix defined as follows:

$$\Phi_N = \begin{bmatrix} D & 0 & 0 & 0 \\ CB & D & 0 & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}B & \dots & CB & D \end{bmatrix}$$

First, we can estimate  $\Phi_N$  by solving for  $\hat{\Phi}_N = \min_{\Phi} \|Y - \Phi U\|_F^2$ . Then  $\Gamma_N X \approx Y - \hat{\Phi}_N U$ . We then perform SVD on  $\Gamma_N X$  and separate the resulting decomposition into the 'signal' space and the 'noise' components:

$$Y - \hat{\Phi}_N U = \hat{Q}_s \hat{S}_s \hat{V}_s^\top + \hat{Q}_{ns} \hat{S}_{ns} \hat{V}_{ns}^\top$$

where  $s$  denotes the signal and  $ns$  denotes the noise component respectively. In the absence of noise, similar to the SVD step in realization based methods, we can automatically recover the system order  $n$  as  $Q_{ns} = 0$ . In the presence of noise, heuristics such as removing "small" singular values can be used to separate the signal space from the noise space.

The estimated observability matrix is taken as  $\Gamma_N = \hat{Q}_s$ . At this point we can follow similar approach as before to obtain  $C$  and  $A$ .

Obtaining  $B$  and  $D$  is slightly more complicated. By pre-multiplying (8) with  $\hat{Q}_s^\top$  and post-multiplying it by  $U^\dagger$ , one obtain  $\hat{Q}_{ns}^\top Y U^\dagger = \hat{Q}_{ns}^\top \Phi_N$ . Let  $L = \hat{Q}_{ns}^\top Y U^\dagger$  and  $R = \hat{Q}_{ns}^\top$ , and recall the structure of  $\Phi_N$  where  $B$  and  $D$  always appear as a linear function. As such solving for  $B, D$  from  $L = R\Phi_N$  can be solved via a least square approach.

### 4.3 Statistical frameworks

In system identification, we not only care about learning the system parameters, we are also interested in quantifying the estimation error in the process. This is useful because in a robust control system, with accurate error bounds and estimation, we can guarantee the robustness of the overall system.

In order to formalize the statistical treatment, we first start by writing down the following difference equations for the dynamical system:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + Ke_t \\ y_t &= Cx_t + Du_t + e_t \end{aligned}$$

where  $e_t$ , a temporal white noise, is called the innovation process

If we make the following assumptions:

1. The innovation process is a stationary, ergodic white random process with zero mean and positive definite covariance matrix.  $\mathbb{E}[e_s e_t^\top] = R_{ee} \delta_{t,s}$ .
2. All eigenvalues of  $A$  are less than 1,  $(A, C)$  are observable,  $(A, B, K)$  are controllable.
3. Inputs  $u_0, u_1, \dots$  are persistently exciting and ergodic.

Then we obtain the following consistency results:

1. For direct methods, assuming that  $K = 0, R_{ee} = \sigma^2 I$  then the method returns a consistent estimate of the transfer function.

**Remarks:** When  $K = 0$ , it means the output is corrupted by measurement noise only and not by process noise. Such model is also known as **output error** model (as opposed to a **full noise model** where  $K \neq 0$ ).

2. For realization based methods, consistency of the estimate of the transfer function depends on the consistency of the estimation of the impulse responses. The resulting model from realization based methods is always in output-error form.

Thus realization based methods are recommended when an output-error model is expected. In this write up and [3], there is no rigorous analysis of the error bounds of system parameter estimates. However, in subsequent lectures, we will look at the error bounds of system parameter estimates, starting with ordinary least squares for multiple trajectory cases, followed by single trajectory cases.



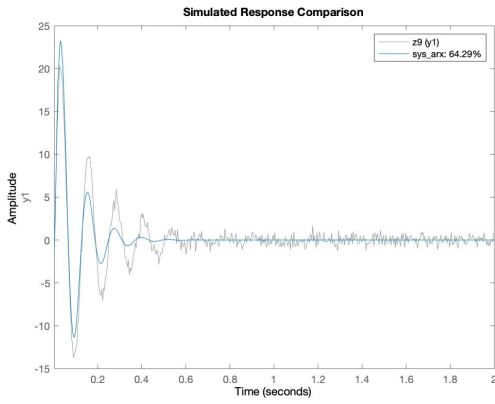
## 5 Numerical experiments

In this section, we show some numerical results comparing between prediction error methods and subspace methods. For this experiment, we use Matlab's system identification toolbox which is available for downloads online. The toolbox also comes with system response data. The dataset that we use is denoted as `z9`. The following code compares between ARX, a well known prediction error methods with `n4SID`, a subspace-based algorithm. In both cases, the system order is chosen to be at 5, 10 and 20. We then compare between the simulated responses of the two methods against the true system.

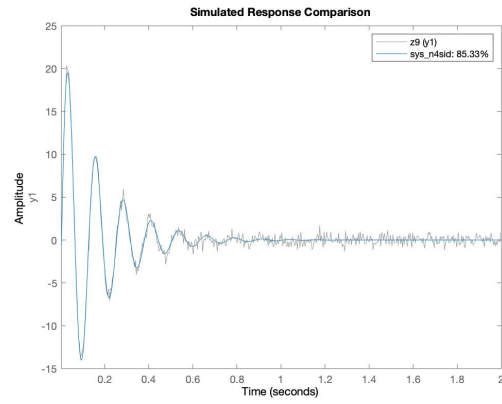
```
load iddata9 z9
nx = 5;
sys_arx = arx(z9, nx);
sys_n4sid = n4sid(z9, nx);

compare(z9, sys_arx, 10)
compare(z9, sys_n4sid, 10)
```

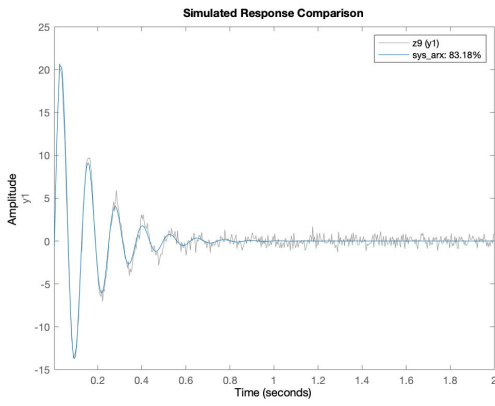
The following figure shows the results for different choices of system order  $nx$ . It is notable that at lower system order  $nx = 5$ , N4SID performs much better than ARX. However, as the complexity of the model grows, the performance of ARX catches up. At  $nx = 10$ , the performance of ARX and that of N4SID are very much similar.



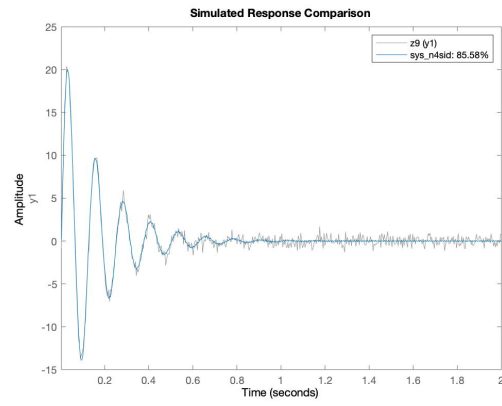
(a)  $n_x = 5$



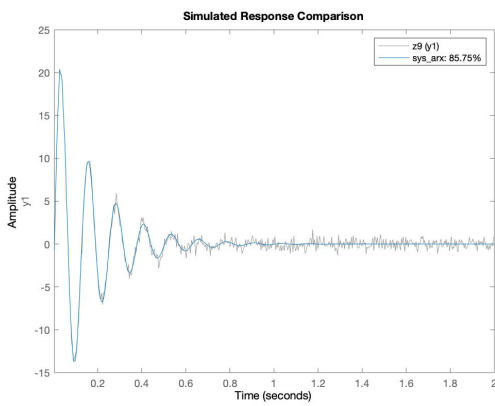
(b)  $n_x = 5$



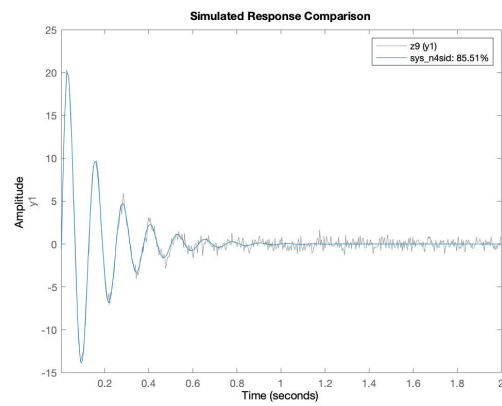
(c)  $n_x = 10$



(d)  $n_x = 10$



(e)  $n_x = 20$



(f)  $n_x = 20$

## References

- [1] Lennart Ljung. *System Identification: Theory for the User*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1986. ISBN: 0-138-81640-9.
- [2] Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *arXiv e-prints*, arXiv:0706.4138 (2007), arXiv:0706.4138. arXiv: 0706.4138 [math.OC].
- [3] Mats Viberg. “Subspace-based Methods for the Identification of Linear Time-invariant Systems”. In: *Automatica* 31.12 (Dec. 1995), pp. 1835–1851. ISSN: 0005-1098. DOI: 10.1016/0005-1098(95)00107-5. URL: [http://dx.doi.org/10.1016/0005-1098\(95\)00107-5](http://dx.doi.org/10.1016/0005-1098(95)00107-5).