Identification of Linear Dynamic Systems Using Dynamic Iterative Principal Component Analysis

Deepak Maurya, Arun K. Tangirala and **Shankar Narasimhan**

Department of Chemical Engineering

IIT Madras

June 08, 2016

Problem statement and setting

Objective

Identify the **dynamic (difference equation) model** from **measurements** of input and output using **principal component analysis** (PCA)

Problem statement and setting

Objective

Identify the **dynamic (difference equation) model** from **measurements** of input and output using **principal component analysis** (PCA)

Framework / Assumptions:

- Linear time-invariant **(LTI)** processes.

Problem statement and setting

Objective

Identify the **dynamic (difference equation) model** from **measurements** of input and output using **principal component analysis** (PCA)

Framework / Assumptions:

- Linear time-invariant **(LTI)** processes.
- **Input** and **output** are both known only with **errors** (the EIV case).

Problem statement and setting

Objective

Identify the **dynamic (difference equation) model** from **measurements** of input and output using **principal component analysis** (PCA)

Framework / Assumptions:

- Linear time-invariant **(LTI)** processes.
- **Input** and **output** are both known only with **errors** (the EIV case).
- **Order** and **delay** are unknown.

Problem statement and setting

Objective

Identify the **dynamic (difference equation) model** from **measurements** of input and output using **principal component analysis** (PCA)

Framework / Assumptions:

- Linear time-invariant **(LTI)** processes.
- **Input** and **output** are both known only with **errors** (the EIV case).
- **Order** and **delay** are unknown.
- **Noise covariance matrix** (of the errors in input and output) is unknown.

PCA for identification: Quick review

Suppose $\mathbf{x}[k] \in \mathbb{R}^{M\times 1}$ are **instantaneously** related through d **linear** constraints

$$\mathbf{A}\mathbf{x} = 0, \qquad \mathbf{A} \in \mathbb{R}^{d\times M},\ \text{rank}(\mathbf{A}) = d \tag{1}$$

Then, given N noise-free observations of $\mathbf{x}[k]$ in $\mathbf{X} = \{\mathbf{x}[k]\}_{k=0}^{N-1} \in \mathbb{R}^{N\times M}$, the following results fall out from the SVD (PCA) of $\mathbf{X}$ under $d < M$

1. rank$(\mathbf{X}) = M - d$, i.e., the last d singular values, $\sigma_{M-d+1} = \cdots = \sigma_M = 0$.
2. The **right singular vectors** corresponding to the d zero singular values provide **a basis for the null space** of $\mathbf{X}$, i.e.,

$$\bar{\mathbf{A}} \triangleq \bar{\mathbf{V}} = \begin{bmatrix} \mathbf{v}_{M-d+1} & \mathbf{v}_{M-d+2} & \cdots & \mathbf{v}_M \end{bmatrix}^T = \mathbf{T}\mathbf{A}, \qquad \det(\mathbf{T}) \neq 0 \tag{2}$$

PCA for identification ...contd.

Identification (regression):

Suppose $\mathbf{x} = \begin{bmatrix} \mathbf{x}_D \\ \mathbf{x}_I \end{bmatrix}$, $\mathbf{x}_D \in \mathbb{R}^{n_y \times 1}$, $\mathbf{x}_I \in \mathbb{R}^{n_u \times 1}$ and correspondingly $\mathbf{A} = \begin{bmatrix} \mathbf{A}_D & \mathbf{A}_I \end{bmatrix}$.

Then, the model

$$\mathbf{x}_D = \mathbf{B}\mathbf{x}_I, \qquad \mathbf{B} = -\mathbf{A}_D^{-1}\mathbf{A}_I \tag{3}$$

can be **exactly** recovered from an identical partitioning of $\tilde{\mathbf{A}}$ as

$$\mathbf{B} = -\bar{\mathbf{A}}_D^{-1}\bar{\mathbf{A}}_I \tag{4}$$

PCA for identification ...contd.

Identification (regression):

Suppose $\mathbf{x} = \begin{bmatrix} \mathbf{x}_D \\ \mathbf{x}_I \end{bmatrix}$, $\mathbf{x}_D \in \mathbb{R}^{n_y \times 1}$, $\mathbf{x}_I \in \mathbb{R}^{n_u \times 1}$ and correspondingly $\mathbf{A} = \begin{bmatrix} \mathbf{A}_D & \mathbf{A}_I \end{bmatrix}$.

Then, the model

$$\mathbf{x}_D = \mathbf{B}\mathbf{x}_I, \qquad \mathbf{B} = -\mathbf{A}_D^{-1}\mathbf{A}_I \tag{3}$$

can be **exactly** recovered from an identical partitioning of $\tilde{\mathbf{A}}$ as

$$\mathbf{B} = -\bar{\mathbf{A}}_D^{-1}\bar{\mathbf{A}}_I \tag{4}$$

Can we recover both d and $\mathbf{B}$ exactly when

1. Observations are **noisy**?

PCA for identification ...contd.

Identification (regression):

Suppose $\mathbf{x} = \begin{bmatrix} \mathbf{x}_D \\ \mathbf{x}_I \end{bmatrix}$, $\mathbf{x}_D \in \mathbb{R}^{n_y \times 1}$, $\mathbf{x}_I \in \mathbb{R}^{n_u \times 1}$ and correspondingly $\mathbf{A} = \begin{bmatrix} \mathbf{A}_D & \mathbf{A}_I \end{bmatrix}$.

Then, the model

$$\mathbf{x}_D = \mathbf{B}\mathbf{x}_I, \qquad \mathbf{B} = -\mathbf{A}_D^{-1}\mathbf{A}_I \tag{3}$$

can be **exactly** recovered from an identical partitioning of $\tilde{\mathbf{A}}$ as

$$\mathbf{B} = -\bar{\mathbf{A}}_D^{-1}\bar{\mathbf{A}}_I \tag{4}$$

Can we recover both d and $\mathbf{B}$ exactly when

1. Observations are **noisy**? NO, with vanilla PCA but YES with Iterative PCA (Narasimhan and Shah, 2008),and

PCA for identification ...contd.

Identification (regression):

Suppose $\mathbf{x} = \begin{bmatrix} \mathbf{x}_D \\ \mathbf{x}_I \end{bmatrix}$, $\mathbf{x}_D \in \mathbb{R}^{n_y \times 1}$, $\mathbf{x}_I \in \mathbb{R}^{n_u \times 1}$ and correspondingly $\mathbf{A} = \begin{bmatrix} \mathbf{A}_D & \mathbf{A}_I \end{bmatrix}$.

Then, the model

$$\mathbf{x}_D = \mathbf{B}\mathbf{x}_I, \qquad \mathbf{B} = -\mathbf{A}_D^{-1}\mathbf{A}_I \tag{3}$$

can be **exactly** recovered from an identical partitioning of $\tilde{\mathbf{A}}$ as

$$\mathbf{B} = -\bar{\mathbf{A}}_D^{-1}\bar{\mathbf{A}}_I \tag{4}$$

Can we recover both d and $\mathbf{B}$ exactly when

1. Observations are **noisy**? NO, with vanilla PCA but YES with Iterative PCA (Narasimhan and Shah, 2008),and
2. Lagged variables are related (**dynamic** model)? (focus of this work)

Motivating example

Consider the process excited with a $(N = 1023)$ **white** PRBS input:

$$y[k] - 0.5y[k-1] = 2u[k-1]$$

Measurements $u[k] = u[k] + e_u[k]$ and $y[k] = y[k] + e_y[k]$ are obtained by adding noise (**white**) such that (i) $\sigma^2_{e_y} \neq \sigma^2_{e_u}$ and (ii) SNR $= 10$.

Motivating example

Consider the process excited with a $(N = 1023)$ **white** PRBS input:

$$y[k] - 0.5y[k-1] = 2u[k-1]$$

Measurements $u[k] = u[k] + e_u[k]$ and $y[k] = y[k] + e_y[k]$ are obtained by adding noise (**white**) such that (i) $\sigma^2_{e_y} \neq \sigma^2_{e_u}$ and (ii) SNR $= 10$.

Assume order is known. Dynamic PCA (Ku, Storer, and Georgakis, 1995) yields:

$$\mathbf{z}[k] = \begin{bmatrix} y[k] & y[k-1] & u[k-1] \end{bmatrix}^T; \qquad \tilde{\mathbf{Z}} = \begin{bmatrix} \mathbf{z}[1] & \mathbf{z}[2] & \cdots & \mathbf{z}[N-1] \end{bmatrix}^T$$

yields $\Lambda = [8.5 \quad 3.9 \quad 0.1883];$ $\bar{\mathbf{A}} = [-0.3991 \quad 0.1904 \quad 0.8969]$

from which the model (in difference equation form) is recovered as:

$$y[k] - 0.4772y[k-1] = 2.2476u[k-1] \qquad \times \tag{5}$$

Scaling with Σ_e

Assume Σ_e is known. $\Sigma_e = \text{diag}[0.5758 \quad 0.5758 \quad 0.0918$

Scale data with $\Sigma_e^{-1/2}$, i.e., $\mathbf{z} \to \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

$$\tilde{\mathbf{z}}[k] = \begin{bmatrix}\tilde{y}[k] & \tilde{y}[k-1] & \tilde{u}[k-1]]\end{bmatrix}^T; \quad \tilde{\mathbf{Z}} = \begin{bmatrix}\tilde{\mathbf{z}}[1] & \tilde{\mathbf{z}}[2] & \cdots & \tilde{\mathbf{z}}[N-1]\end{bmatrix}^T$$

Scaling with Σ_e

Assume Σ_e is known. $\Sigma_e = \text{diag}[0.5758 \quad 0.5758 \quad 0.0918$

Scale data with $\Sigma_e^{-1/2}$, i.e., $\mathbf{z} \rightarrow \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

$$\tilde{\mathbf{z}}[k] = \begin{bmatrix}\tilde{y}[k] & \tilde{y}[k-1] & \tilde{u}[k-1]\end{bmatrix}^T; \quad \tilde{\mathbf{Z}} = \begin{bmatrix}\tilde{\mathbf{z}}[1] & \tilde{\mathbf{z}}[2] & \cdots & \tilde{\mathbf{z}}[N-1]\end{bmatrix}^T$$

Eigenvalue, eigenvector and constraint vector from PCA of $\tilde{\mathbf{Z}}$:

$$\Lambda = [20.2 \quad 10.6 \quad \mathbf{0.9611}]; \qquad \tilde{\mathbf{A}} = [0.7235 \quad -0.3671 \quad -0.5846]$$

The constraint matrix for the "raw" data is obtained by re-scaling $\tilde{\mathbf{A}}$

$$\bar{\mathbf{A}} = [0.0298 \quad -0.0151 \quad -0.0603]$$

Scaling with Σ_e

Assume Σ_e is known. $\Sigma_e = \text{diag}[0.5758 \quad 0.5758 \quad 0.0918$

Scale data with $\Sigma_e^{-1/2}$, i.e., $\mathbf{z} \rightarrow \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

$$\tilde{\mathbf{z}}[k] = \begin{bmatrix}\tilde{y}[k] & \tilde{y}[k-1] & \tilde{u}[k-1]\end{bmatrix}^T; \quad \tilde{\mathbf{Z}} = \begin{bmatrix}\tilde{\mathbf{z}}[1] & \tilde{\mathbf{z}}[2] & \cdots & \tilde{\mathbf{z}}[N-1]\end{bmatrix}^T$$

Eigenvalue, eigenvector and constraint vector from PCA of $\tilde{\mathbf{Z}}$:

$$\Lambda = [20.2 \quad 10.6 \quad \mathbf{0.9611}]; \qquad \tilde{\mathbf{A}} = [0.7235 \quad -0.3671 \quad -0.5846]$$

The constraint matrix for the "raw" data is obtained by re-scaling $\tilde{\mathbf{A}}$

$$\bar{\mathbf{A}} = [0.0298 \quad -0.0151 \quad -0.0603]$$

The input-output model (in DE) is then recovered as

$$y[k] - 0.5073y[k-1] = 2.0232u[k-1] \quad \checkmark \qquad (6)$$

Iterative PCA (Narasimhan and Shah, 2008)

Core Idea

Perform PCA of scaled data with a scaling factor of $\Sigma_{\mathbf{e}}^{-1/2}$, i.e., $\mathbf{z} \rightarrow \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

Iterative PCA (Narasimhan and Shah, 2008)

Core Idea

Perform PCA of scaled data with a scaling factor of $\Sigma_{\mathbf{e}}^{-1/2}$, i.e., $\mathbf{z} \rightarrow \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

Result

Eigenvalues of the (sample) covariance matrices are shifted by unity!

$$\lambda(\mathbf{S}_{\tilde{\mathbf{z}}}) = \lambda(\mathbf{S}_{\tilde{\mathbf{x}}}) + 1 \tag{7}$$

Iterative PCA (Narasimhan and Shah, 2008)

Core Idea

Perform PCA of scaled data with a scaling factor of $\Sigma_{\mathbf{e}}^{-1/2}$, i.e., $\mathbf{z} \rightarrow \tilde{\mathbf{z}} \triangleq \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}$.

Result

Eigenvalues of the (sample) covariance matrices are shifted by unity!

$$\lambda(\mathbf{S}_{\tilde{\mathbf{z}}}) = \lambda(\mathbf{S}_{\tilde{\mathbf{x}}}) + 1 \tag{7}$$

Framework:

1. *Output-error model*: $\mathbf{z}[k] = \mathbf{x}[k] + \mathbf{e}[k], \;\; \mathbf{e}[k] \sim \text{GWN}(\mathbf{0}, \Sigma_{\mathbf{e}})$ (diagonal $\Sigma_{\mathbf{e}}$)
2. *Quasi-stationarity*: Noise-free signals $\{x_i[k]\}_{i=1}^{M}$ are quasi-stationary.
3. **Identifiability**: Number of constraints should satisfy

$$\boxed{\frac{d(d+1)}{2} > M} \tag{8}$$

Estimating $\Sigma_{\mathbf{e}}$ in IPCA

Iteratively estimate the noise covariance matrix and (a basis for) $\mathbf{A}$.

Suppose the **user-supplied** d **is correct** and at some iteration, $\hat{\mathbf{A}}^{(i)}$ is the solution. Then,

$$\hat{\mathbf{A}}^{(i)}\mathbf{z}[k] = \cancelto{0}{\hat{\mathbf{A}}^{(i)}\mathbf{x}[k]} + \hat{\mathbf{A}}^{(i)}\mathbf{e}[k] = \overbrace{\hat{\mathbf{A}}^{(i)}\mathbf{e}[k]}^{\mathbf{r}[k]} \tag{9}$$

Estimating $\Sigma_{\mathbf{e}}$ in IPCA

Iteratively estimate the noise covariance matrix and (a basis for) $\mathbf{A}$.

Suppose the **user-supplied** d **is correct** and at some iteration, $\hat{\mathbf{A}}^{(i)}$ is the solution. Then,

$$\hat{\mathbf{A}}^{(i)}\mathbf{z}[k] = \cancelto{0}{\hat{\mathbf{A}}^{(i)}\mathbf{x}[k]} + \hat{\mathbf{A}}^{(i)}\mathbf{e}[k] = \overbrace{\hat{\mathbf{A}}^{(i)}\mathbf{e}[k]}^{\mathbf{r}[k]} \tag{9}$$

The covariance of the residuals $\mathbf{r}[k]$ and that of the noise are related as

$$\Sigma_{\mathbf{r}} = \hat{\mathbf{A}}^{(0)}\Sigma_e(\hat{\mathbf{A}}^{(0)})^T \tag{10}$$

Estimating Σ_e in IPCA

Iteratively estimate the noise covariance matrix and (a basis for) $\mathbf{A}$.

Suppose the **user-supplied** d **is correct** and at some iteration, $\hat{\mathbf{A}}^{(i)}$ is the solution. Then,

$$\hat{\mathbf{A}}^{(i)}\mathbf{z}[k] = \cancelto{0}{\hat{\mathbf{A}}^{(i)}\mathbf{x}[k]} + \hat{\mathbf{A}}^{(i)}\mathbf{e}[k] = \overbrace{\hat{\mathbf{A}}^{(i)}\mathbf{e}[k]}^{\mathbf{r}[k]} \tag{9}$$

The covariance of the residuals $\mathbf{r}[k]$ and that of the noise are related as

$$\Sigma_{\mathbf{r}} = \hat{\mathbf{A}}^{(0)}\Sigma_e(\hat{\mathbf{A}}^{(0)})^T \tag{10}$$

Under these conditions, a (conditionally) optimal estimate of Σ_e can be generated by solving the following MLE problem:

$$\boxed{\min_{\Sigma_e} N \log\det \hat{\mathbf{A}}^{(i)}\Sigma_e(\hat{\mathbf{A}}^{(i)})^T + \sum_{k=0}^{N-1}(\mathbf{r}^T[k](\hat{\mathbf{A}}^{(i)}\Sigma_e(\hat{\mathbf{A}}^{(i)})^T)^{-1}\mathbf{r}[k])} \tag{11}$$

IPCA Algorithm

1. Stack the given N observations of M variables into a $N \times M$ matrix $\mathbf{Z}$.
2. Set counter $i = 0$ and $\Sigma_{\mathbf{e}} = \mathbf{I}$. Guess a value of d as dictated by the identifiability criterion.
3. Scale data as $\mathbf{z}[k] :\rightarrow \Sigma_{\mathbf{e}}^{-1/2}\mathbf{z}[k]$ and obtain estimate of constraint matrix, $\hat{\mathbf{A}}^{(k)}$ from PCA of scaled data.
4. Compute the estimate of noise covariance matrix $\Sigma_e^{(k)}$ from solving (11).
5. Increment $i :\rightarrow i + 1$ and repeat steps 3-4 until convergence.

- If $\dim$(unity eigenvalues) does not match the guessed value, repeat steps 2-4 with a refined guess of d.

Example: Flow mixing

Two flows mixing at three nodes of a flow network to produce three other flows

$$x_3[k] = 2x_1[k] + 3x_2[k] \quad (12a)$$

$$x_4[k] = x_1[k] + x_2[k] \quad (12b)$$

$$x_5[k] = x_1[k] - 2x_2[k] \quad (12c)$$

Observe: Sufficient redundancy is available, i.e., the identifiability requirement is satisfied since $d_0 = 3$ and $d_0(d_0+1)/2 = 6 > M = 5$.

Flow mixing example: Remarks

- Two flows $x_1[k]$ and $x_2[k]$ are generated randomly. Flows x_3 to x_5 generated as per (12).
- Measurements: $z_i[k] = x_i[k] + e_i[k], \quad e_i[k] \sim \text{GWN}(0, \sigma_i^2), \quad i = 1, \cdots, 5$. SNR is set to 10. The true noise covariance matrix is,

$$\Sigma_{\mathbf{e},0} = \text{diag}(0.1, 0.1, 1.3, 2, 0.5) \tag{13}$$

The true constraint and regressor matrices are:

$$\mathbf{A}_0 = \begin{bmatrix} 2 & 3 & -1 & 0 & 0 \\ 1 & 1 & 0 & -1 & 0 \\ 1 & -2 & 0 & 0 & -1 \end{bmatrix} \qquad \mathbf{B}_0 = \begin{bmatrix} 2 & 3 \\ 1 & 1 \\ 1 & -2 \end{bmatrix} \tag{14}$$

Flow mixing example: Results from IPCA

Guess $d = 3$ (the minimum identifiable constraints) and initialize with estimates from PCA.

Singular values (σ_i)	6.0307, 4.2764, 1.0053, 0.9987, 0.9956 (IPCA) 55.5795, 40.5724, 9.5545, 9.2713, 9.0856 (PCA)
Constraint matrix $\hat{\hat{\mathbf{A}}}$	$\begin{bmatrix} 0.9217 & -1.5157 & 0.4334 & -1.1246 & -0.6706 \\ -1.9351 & -1.4441 & 0.4065 & 0.8119 & 0.2760 \\ -0.4021 & 1.4974 & 0.4115 & -1.2332 & 0.7663 \end{bmatrix}$
Covariance matrix $\hat{\Sigma}_e$	0.1123, 0.0993, 1.3163, 0.1846, 0.4352 (diagonal)
Regressor matrices $\hat{\mathbf{B}}_{\text{PCA}}$, $\hat{\mathbf{B}}_{\text{IPCA}}$ and $\mathbf{B}_0$,	$\begin{bmatrix} 0.536 & 0.82 \\ 0.689 & 0.693 \\ 0.461 & -0.913 \end{bmatrix}, \begin{bmatrix} 2.063 & 2.956 \\ 1.001 & 0.971 \\ 1.028 & -1.979 \end{bmatrix}, \begin{bmatrix} 2 & 3 \\ 1 & 1 \\ 1 & -2 \end{bmatrix}$

Extending IPCA to the dynamic case

Stacking with lags up to **true order and delay** (if known) does not provide adequate redundancy to estimate both model and noise covariance matrix!

Example: For the first-order, unit-delay example, stacking exactly with $d = 1$ and unit delay, for **diagonal** $\Sigma_{\mathbf{e}}$ would require estimation of $M = 3$ $(y[k], y[k-1], u[k-1])$ variances. However, $d(d+1)/2 = 1 \ngeq M = 3$.

Extending IPCA to the dynamic case

Stacking with lags up to **true order and delay** (if known) does not provide adequate redundancy to estimate both model and noise covariance matrix!

Example: For the first-order, unit-delay example, stacking exactly with $d = 1$ and unit delay, for **diagonal** $\Sigma_{\mathbf{e}}$ would require estimation of $M = 3$ $(y[k], y[k-1], u[k-1])$ variances. However, $d(d+1)/2 = 1 \ngeq M = 3$.

Proposition

1. Exploit the fact that $\text{diag}(\Sigma_{\mathbf{e}}) = \begin{bmatrix} \sigma_y^2 \mathbf{1}_{m+2} & \sigma_u^2 \mathbf{1}_{m+2} \end{bmatrix}$ and **modify** IPCA.
2. Stack variables with "sufficiently" excess lags

$$\mathbf{z}_L[k] = \begin{bmatrix} y[k] & y[k-1] & \cdots & y[k-L] & u[k] & u[k-1] & \cdots & u[k-L] \end{bmatrix} \quad (15)$$

such that $L > (n_y + n_u)$ (in practice $L \gg (n_y + n_u)$ is better).

Recovering the model: The idea of rotation

Challenge: Stacking lagged variables in excess (of true order and delay) produces multiple relations, i.e., constraints in excess of the true number, are obtained!

Recovering the model: The idea of rotation

Challenge: Stacking lagged variables in excess (of true order and delay) produces multiple relations, i.e., constraints in excess of the true number, are obtained!

Proposed method

Consider the **SISO case** with true order d_0.

- Assume that IPCA of the stacked matrix $\mathbf{Z}_L$ identifies p constraints, $p > d_0$.
- Then the identified constraint matrix $\tilde{\mathbf{A}}$ is of size $(p \times (2L+2))$. Partitioning $\tilde{\mathbf{A}}$ as earlier, we have

$$\tilde{\mathbf{A}}_D \mathbf{y}_L = -\tilde{\mathbf{A}}_I \mathbf{u}_L \qquad (16)$$

- Rotate $\mathbf{A}_D$ through a rotation matrix $\mathbf{R}_D$ such that

$$\text{structure}(\mathbf{R}_D \bar{\mathbf{A}}_D) = \text{structure}(\mathbf{A}_{D,0}) \qquad (17)$$

  where $\mathbf{A}_{D,0}$ is the **true** constraint matrix for the given stacking. Rotate $\bar{\mathbf{A}}_I$ through the same matrix, i.e., $\bar{\mathbf{A}}_I :\to \mathbf{R}_D \bar{\mathbf{A}}_I$.

Determining the true structure

The **key** to recovering the model is in determining the structure of $\mathbf{A}_{D,0}$.

1. **Known order:** Locations of zero and non-zero entries in $\mathbf{A}_{D,0}$ are known $\implies$ structure$(\mathbf{A}_{D,0})$ is known, by virtue of the shift property of lagged relations.
   For a first-order with unit-delay,

$$\mathbf{A}_{D,0} = \begin{bmatrix} 1 & a_1 & 0 \\ 0 & 1 & a_1 \end{bmatrix} \tag{18}$$

Determining the true structure

The **key** to recovering the model is in determining the structure of $\mathbf{A}_{D,0}$.

1. **Known order:** Locations of zero and non-zero entries in $\mathbf{A}_{D,0}$ are known $\Longrightarrow$ structure$(\mathbf{A}_{D,0})$ is known, by virtue of the shift property of lagged relations.
   For a first-order with unit-delay,
   $$\mathbf{A}_{D,0} = \begin{bmatrix} 1 & a_1 & 0 \\ 0 & 1 & a_1 \end{bmatrix} \tag{18}$$
2. **Unknown order:** First determine the order using the relation
   $$\boxed{p = L - d + 1 \;\text{ where }\; d = \max(\text{input, output order})} \tag{19}$$
   Subsequently, follow the known-order route.

Arriving at the model

Estimating the rotation matrix

The rotation matrix $\mathbf{R}_D \in \mathbb{R}^{p \times p}$ can be estimated in two different ways

1. *Exact route:* Estimate $\mathbf{R}_D$ by exactly matching the zero- and unity-values.
2. *Overdetermined route:* Estimate $\mathbf{R}_D$ by additionally forcing the non-zero entries to match across rows.

- The overdetermined method is preferred in presence of noise.

Finally, the nearly identical multiple relations (from $\mathbf{R}_D\bar{\mathbf{A}}_D$ and $\mathbf{R}_D\bar{\mathbf{A}}_I$) thus identified can be averaged to obtain a single relation.

Motivational example (first-order, unit-delay)

Assume order is **known**. Stack up to lag $L = 2$ (minimum requirement).

1. Construct $\mathbf{z_2}[k]$ and $\mathbf{Z_2}$ as

$$\mathbf{z_2}[k] = \begin{bmatrix} y[k] & y[k-1] & y[k-2] & u[k] & u[k-1] & u[k-2] \end{bmatrix}^T$$
$$\mathbf{Z_2} = \begin{bmatrix} \mathbf{z_2}[2] & \mathbf{z_2}[3] & \cdots & \mathbf{z_2}[N] \end{bmatrix}^T$$

2. Eigenvalues are found to be

$$\Lambda = [10.4 \quad 5.09 \quad 3.6 \quad 1.079 \quad 0.2086 \quad 0.1607]$$

3. Eigenvalues after two iterations of modified IPCA

$$\Lambda = [25.74 \quad 16.26 \quad 11.41 \quad 10.88 \quad 1.0005 \quad 0.9995]$$

4. Estimated $\Sigma_{\mathbf{e}}$: $[0.5151 \quad 0.0989]$, **true values** $[0.5758 \quad 0.0918]$.

Rotation Matrix Estimation

1. Partition the (2×6) from two IPCA iterations into submatrices corresponding to dependent and independent variables

$$\hat{\mathbf{A}}_D \mathbf{y_2}[k] = -\hat{\mathbf{A}}_I \mathbf{u_2}$$

$$\begin{bmatrix} 0.0312 & -0.0116 & -0.0023 \\ 0.0041 & 0.0291 & -0.0158 \end{bmatrix} \begin{bmatrix} y[k] \\ y[k-1] \\ y[k-2] \end{bmatrix} = - \begin{bmatrix} 0.0021 & -0.0635 & -0.0080 \\ 0.0006 & -0.0063 & -0.0637 \end{bmatrix} \begin{bmatrix} u[k] \\ u[k-1] \\ u[k-2] \end{bmatrix}$$

2. Construct the structure of $\mathbf{A}_{D,0}$ from known order and shift property.

$$\mathbf{A}_{D,0} = \begin{bmatrix} 1 & a_{11} & 0 \\ 0 & 1 & a_{11} \end{bmatrix}$$

3. Estimate rotation matrix $\mathbf{R}$ (2×2) using overdetermined approach

Model Parameters Estimation

1. Pre-multiply the estimated constraint matrix using $\mathbf{R}$.

$$\mathbf{R}_D\hat{\mathbf{A}}_D = -\mathbf{R}_D\hat{\mathbf{A}}_I$$

$$\begin{bmatrix} 0.9983 & -0.5083 & -0.0063 \\ 0.0008 & 1.0016 & -0.5051 \end{bmatrix} \mathbf{y}_2 = \begin{bmatrix} -0.0653 & 2.0428 & -0.0226 \\ -0.0116 & -0.0642 & 2.0496 \end{bmatrix} \mathbf{u}_2$$

The estimated model is averaged to obtain:

$$y[k] - 0.5067y[k-1] = 2.0462u[k-1] \tag{20}$$

Model Parameters Estimation

1. Pre-multiply the estimated constraint matrix using $\mathbf{R}$.

$$\mathbf{R}_D\hat{\mathbf{A}}_D = -\mathbf{R}_D\hat{\mathbf{A}}_I$$

$$\begin{bmatrix} 0.9983 & -0.5083 & -0.0063 \\ 0.0008 & 1.0016 & -0.5051 \end{bmatrix} \mathbf{y}_2 = \begin{bmatrix} -0.0653 & 2.0428 & -0.0226 \\ -0.0116 & -0.0642 & 2.0496 \end{bmatrix} \mathbf{u}_2$$

The estimated model is averaged to obtain:

$$y[k] - 0.5067y[k-1] = 2.0462u[k-1] \tag{20}$$

2. MC simulations are performed for SNR $=$ 10 and lag order 2. Parameter estimates are found to follow a Gaussian distribution. The average model with 95% CIs are.

$$y[k] - \underset{(\pm 0.0255)}{0.5003}\, y[k-1] = -\underset{(\pm 0.068)}{0.001}\, u[k] + \underset{(\pm 0.1008)}{2.0022}\, u[k-1]$$

Example 2: DIPCA

Measurements ($N = 1023$) from a simulation process with full band PRBS input.

1. Construct augmented data matrix for up to lag order 6

$$\mathbf{z_6}[k] = \begin{bmatrix} y[k] & y[k-1] \ldots y[k-6] & u[k] & u[k-1] \ldots u[k-6] \end{bmatrix}^T$$

$$\mathbf{Z_6} = \begin{bmatrix} \mathbf{z_6}[6] & \mathbf{z_6}[7] & \cdots & \mathbf{z_6}[N-1] \end{bmatrix}^T$$

Example 2: DIPCA

Measurements ($N = 1023$) from a simulation process with full band PRBS input.

1. Construct augmented data matrix for up to lag order 6

$$\mathbf{z_6}[k] = \begin{bmatrix} y[k] & y[k-1] \ldots y[k-6] & u[k] & u[k-1] \ldots u[k-6] \end{bmatrix}^T$$

$$\mathbf{Z_6} = \begin{bmatrix} \mathbf{z_6}[6] & \mathbf{z_6}[7] & \cdots & \mathbf{z_6}[N-1] \end{bmatrix}^T$$

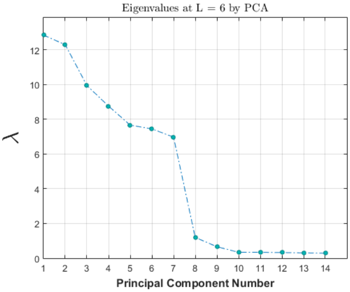

Figure 1: From DPCA

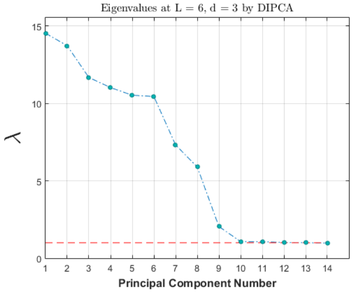

Figure 2: From DIPCA with guess $d = 3$

Eigenvalues

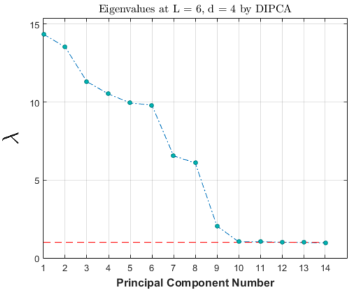

Figure 3: From DIPCA with guess $d = 4$

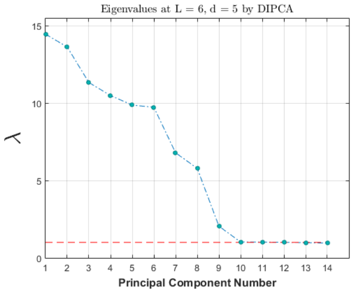

Figure 4: From DIPCA with guess $d = 5$

Figure 3 shows even though $d = 4$ eigenvalues were expected to be unity but last 5 turned out to be unity. Figure 4 shows last 5 eigenvalues to be unity as expected. So order is derived as $d = 6 - 5 + 1 = 2$.

Model Parameters

1. The model is estimated to be a second-order DE:

$$y[k] - 0.972y[k-1] + 0.199y[k-2] = -0.028u[k] + 2.596u[k-1] - 2.472u[k-2]$$

Noise variance is estimated to be $[1.2567 \quad 0.2067]$; **true values:** $[1.3949 \quad 0.1926]$.

2. DIPCA gives **consistent** estimates.
Check: sample size is increased to $N = 12000$ for SNR = 5. Model is determined as second-order:

$$y[k] - 1.005y[k-1] + 0.241y[k-2] = 0.029u[k] + 2.486u[k-1] - 2.358u[k-2]$$

Noise variance is estimated to be $[1.4059 \quad 0.1984]$; **true values:** $[1.3746 \quad 0.1984]$.

Confidence Intervals

Sample size is fixed at $N = 1023$ and SNR 5 while performing MC simulations. Confidence interval with the proposed approach:

$$y[k] - \underset{(\pm 0.0654)}{0.9981}\, y[k-1] + \underset{(\pm 0.0582)}{0.2368}\, y[k-2] =$$
$$- \underset{(\pm 0.1708)}{0.0082}\, u[k] + \underset{(\pm 0.3910)}{2.5161}\, u[k-1] - \underset{(\pm 0.4093)}{2.3874}\, u[k-1]$$

Confidence Intervals

Sample size is fixed at $N = 1023$ and SNR 5 while performing MC simulations. Confidence interval with the proposed approach:

$$y[k] - \underset{(\pm 0.0654)}{0.9981}\, y[k-1] + \underset{(\pm 0.0582)}{0.2368}\, y[k-2] =$$
$$- \underset{(\pm 0.1708)}{0.0082}\, u[k] + \underset{(\pm 0.3910)}{2.5161}\, u[k-1] - \underset{(\pm 0.4093)}{2.3874}\, u[k-1]$$

Data generating process:

$$y[k] - y[k-1] + 0.24y[k-2] = 2.5u[k-1] - 2.375u[k-2]$$

Data was generated for full length ($N = 1023$), full band PRBS input and noise with SNR $= 5$ was added to both input and output signals.

Concluding remarks

- A systematic, rigorous method to accurately estimate the dynamic model for the EIV case using dynamic, iterative PCA has been presented.
  - Minimal user intervention (maximum stacking lag to be supplied).
  - Determines the order
  - Noise covariance estimate is also provided.
  - Estimator is observed to be consistent.

Concluding remarks

- A systematic, rigorous method to accurately estimate the dynamic model for the EIV case using dynamic, iterative PCA has been presented.
  - Minimal user intervention (maximum stacking lag to be supplied).
  - Determines the order
  - Noise covariance estimate is also provided.
  - Estimator is observed to be consistent.
- Extensions to MISO and MIMO case:
  - Break up the given system into individual SISO systems using signal conditioning, i.e., work with *partial* covariance matrices.

Bibliography

- Ku, W., R. Storer, and C. Georgakis (1995). Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 30, pp. 179–196.
- Narasimhan, S. and S. Shah (2008). Model identification and error covariance matrix estimation from noisy data using PCA. *Control Engineering Practice*, 16, pp. 146–155.
- Tangirala, A. K. (2014). *Principles of System Identification: Theory and Practice*. Boca Raton, FL, USA: CRC Press, Taylor & Francis Group.