CHEN4011
Advanced Modelling and COntrol

Curtin University

Dr. Ranjeet Utikar (RU)

Curtin University
Malaysia

Dr. Jobrun Nandong (JN)

Principal Component Analysis (PCA) and Partial Least Squared (PLS) Modelling

JN/CP Revised 2021
JN/RU Revised 2022

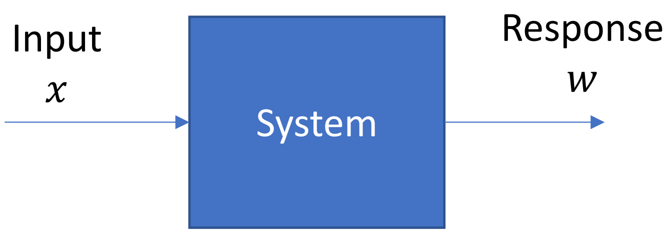
Outline

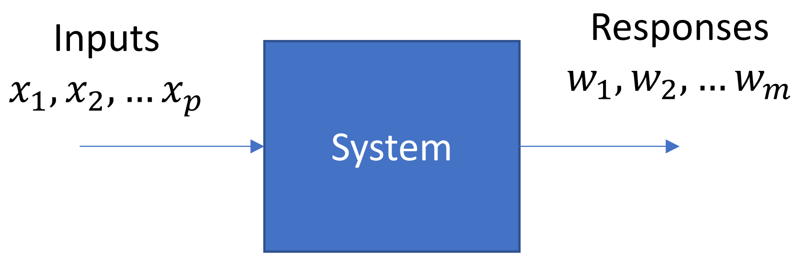
- Univariate and multivariate statistical analyses
- Principal Component Analysis (PCA)
- Fundamental of PCA
- Case study – application of PCA in fault detection
- Applications of PCA
- Introduction to Multivariate Regression
- Univariate Multiple Regression
- Multivariate Multiple Regression
- Partial Least Square Regression
- Some demonstrations using MATLAB *plsregress* function

Univariate vs Multivariate Analysis

- Univariate statistical analysis **– 1 input variable and 1 response variable**
- *E.g., input variable = reactor temperature; response variable = reactor conversion*
- Multivariate statistical analysis **– multiple variables** and **multiple responses**
- *E.g., input variables = reactor temperature, feed concentration; response variables = reactor conversion and product yield*
- **Multivariate analysis** attempts to **reveal the key information** from the correlated variables
- Widely used in the science and engineering applications – **data analysis**

Illustrations of Univariate and Multivariate





Univariate

Data set X consists of n rows of observations and 2 columns of variables:

$$X = \begin{bmatrix} x(1) & w(1) \\ x(2) & w(2) \\ \vdots & \vdots \\ x(n) & w(n) \end{bmatrix}$$

Multivariate

Data set X consists of n observations and $p + m$ columns of variables:

$$X = \begin{bmatrix} x_1(1) & \dots & x_p(1) & w_1(1) & \dots & w_m(1) \\ x_1(2) & \dots & x_p(2) & w_1(2) & \dots & w_m(2) \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ x_1(n) & \dots & x_p(n) & w_1(n) & \dots & w_m(n) \end{bmatrix}$$

Multivariate analysis

- Goal of many multivariate approaches is **simplification** – from large dimension to smaller or ***reduced dimension*** of datasets
- Such approaches are *exploratory*, e.g., generate hypotheses rather than for testing them
- Some approaches:
  i. **Discriminant Analysis** – identifying the relative contribution of p variables to separation of the groups
  ii. **Principal Component Analysis (PCA)** – reduces large dimension of a data set to smaller dimension
  iii. **Multivariate regression**, e.g., partial least square (PLS) regression

PCA approach

- **Maximize variance** of **a** **linear combination** of variables
- **Principal component 1, principal component 2**, …, principal component m
- Use the **first 2 or 3 principal components** $(z_1, z_2, \ldots)$ to explain majority of the total variances of original data set X
- Consider a data set consisting of 5 variables $(y_1, y_2, \ldots y_5)$ and its corresponding principal components $(z_1, z_2, \ldots z_5)$
  - Consider that z_1 and z_2 provide > 80% of the total variances in $y_1, y_2, \ldots y_5$
  - **5 variables** can be represented using only the **first two principal components (2 latent variables)**
- Basic assumption: $y_1, y_2, \ldots y_5$ are *correlated*
- Principal **components z_1 and z_2 are *uncorrelated*** *(orthonormal vectors)*

Multivariate analysis

- **Goal** of many multivariate approaches is **simplification** – from large dimension data to smaller data or ***reduced dimension*** of datasets
- Such approaches are *exploratory*, e.g., generate hypotheses rather than for testing them
- Some approaches:
  i. **Discriminant Analysis** – identifying the relative contribution of p variables to separation of the groups
  ii. **Principal Component Analysis (PCA)** – reduces large dimension of a data set to smaller dimension
  iii. **Multivariate regression**, e.g., partial least square (PLS) regression

PCA Analysis

- Maximize variance of **a linear combination** of variables
- Principal component 1, principal component 2, …, principal component m
- Use the **first 2 or 3 principal components** $(z_1, z_2, \dots)$ to explain majority of the total variances of original data set X
- Consider a data set consisting of 5 variables $(y_1, y_2, \dots y_5)$ and its corresponding principal components $(z_1, z_2, \dots z_5)$
  - Consider that z_1 and z_2 provide > 80% of the total variances in $y_1, y_2, \dots y_5$
  - **5 variables** can be represented using only the **first two principal components (2 latent variables)**
- Basic assumption: $y_1, y_2, \dots y_5$ are *correlated*
- Principal **components z_1 and z_2 are *uncorrelated*** *(orthonormal vectors)*

PCA Analysis

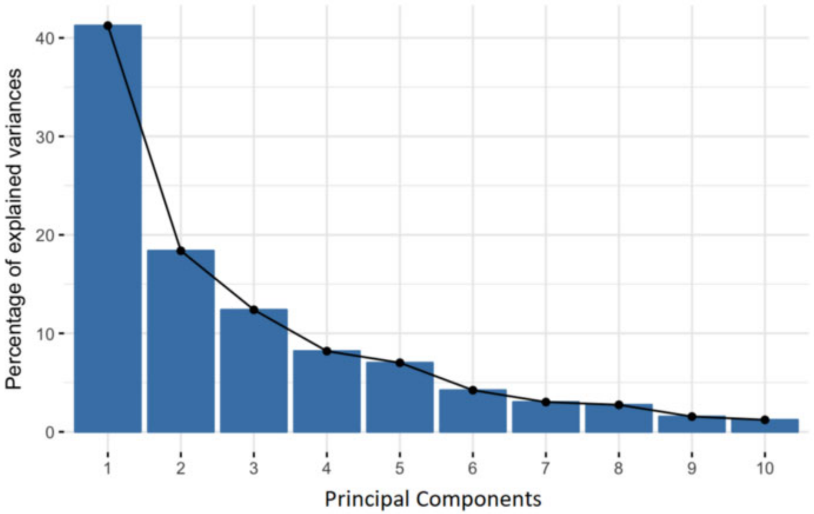
- PCA decomposes a data set X matrix (*n* observations, *k* variables) into:

$$\boldsymbol{X} = \boldsymbol{T}\boldsymbol{P}^{\boldsymbol{T}} + \boldsymbol{E} \quad where \quad X \in \mathcal{R}^{n\times k}, T \in \mathcal{R}^{n\times k}, P \in \mathcal{R}^{n\times k}, E \in \mathcal{R}^{n\times k}$$

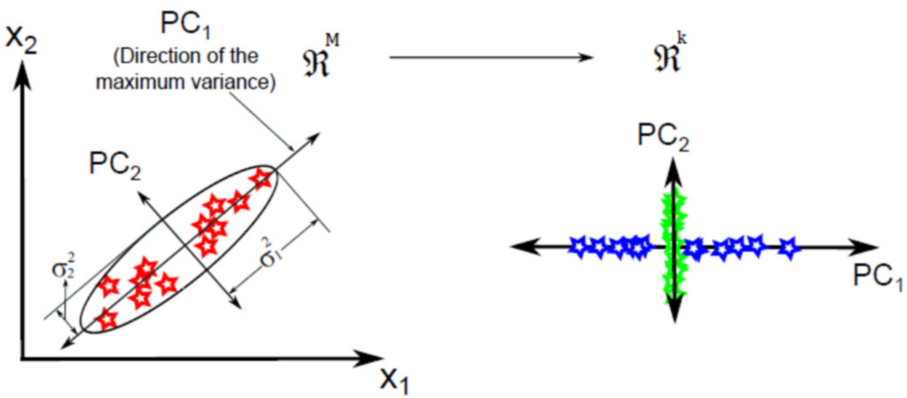
Where $\boldsymbol{T}$ is the matrix of principal component **scores** and $\boldsymbol{P}$ is the matrix of **loadings** that project X onto *principal component space* or ***latent variables***

E represents residual matrix

- Principal components are **orthogonal** to each other, i.e., they are uncorrelated
- P can be obtained using the **singular value decomposition** (SVD)



- **Principal components** show directions of the data that explain a **maximal amount of variance**
- **The larger the variance** carried by a line, the **larger the dispersion** of the data points along it



- **Original data** on the left with original coordinate x_1 and x_2
- Variance of each variable graphically represented
- **Direction of the maximum variance** i.e., principal component PC_1 and PC_2

Applications of PCA – Plant Monitoring

- ***Process monitoring using PCA – Fault detection***
- Construct $\boldsymbol{T^2}$ **and** $\boldsymbol{Q}$ **statistics**
- Consider an observation vector $x \in \mathcal{R}^{m\times 1}$, i.e., m measured variables
- T^2 statistic of the first k principal components (PCs):

$$\boldsymbol{T^2 = \ x^T P(\Lambda)^{-1} P^T x}$$

Where $\Lambda \in \mathcal{R}^{k\times k}$ is a diagonal matrix denoting estimated covariance matrix of **principal component scores**, and $P \in \mathcal{R}^{m\times k}$

- Note that $T^2 \in \mathcal{R}^{1\times 1}$, i.e., a scalar value

Plant Monitoring

- Q statistic can be calculated as follows:

$$\boldsymbol{Q = e^T e \quad where \quad e = \left(I - PP^T\right)x}$$

Where $I \in \mathcal{R}^{m \times m}$ is the identity matrix, i.e., with 1 on the diagonal and zero elsewhere

- **Matrix $\boldsymbol{e} \in \boldsymbol{\mathcal{R}^{m \times 1}}$** is the projection of observation x onto the **residual space**
- The **thresholds of the statistics** are calculated based on assumption that latent variables are **multivariate Gaussian distributed**
- For **non-Gaussian** the thresholds can be obtained using kernel density estimation
- $\boldsymbol{Q \in \mathcal{R}^{1 \times 1}, \text{i. e., a scalar value}}$

Example:- Case Study

Jiang and Yan (2014)

- $$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1.57 & 1.37 & 1.8 \\ 1.73 & 1.05 & 1.7 \\ 1.82 & 1.4 & 1.6 \\ 1.65 & 1.2 & 1.5 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

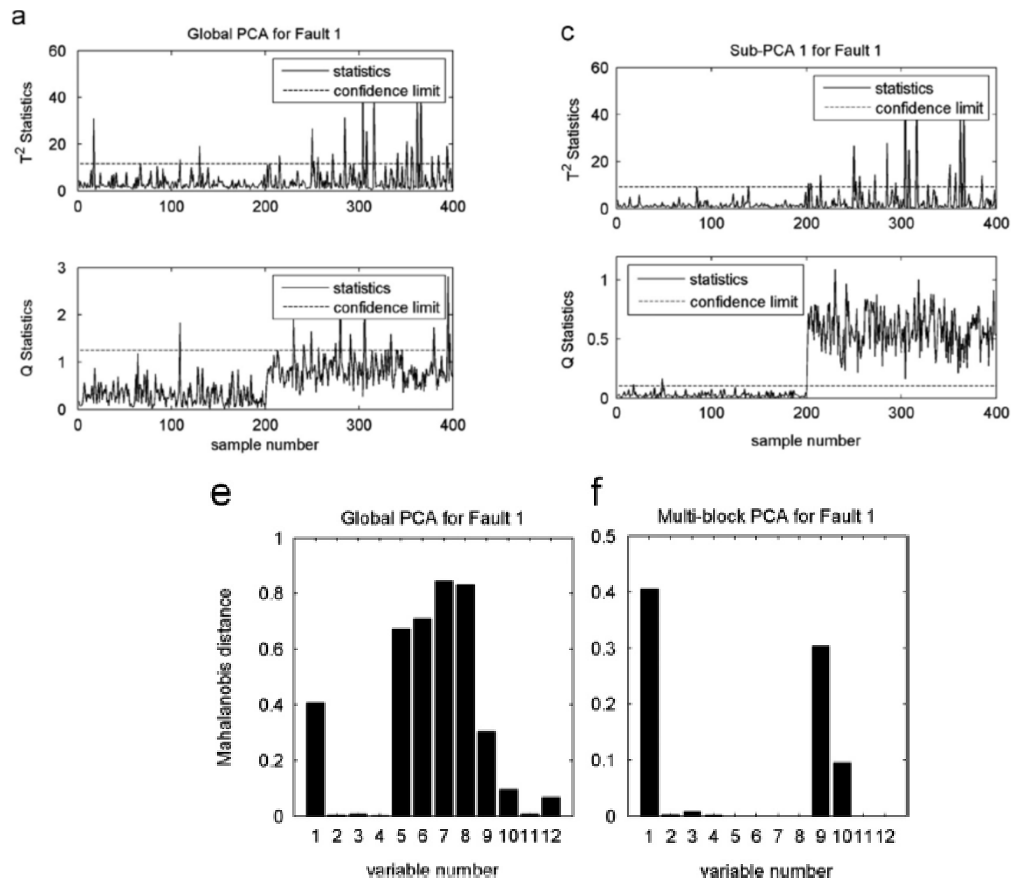
$$\begin{bmatrix} x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} 1.67 & 1.47 & 1.7 \\ 1.63 & 1.15 & 1.8 \\ 1.72 & 1.3 & 1.7 \\ 1.55 & 1.3 & 1.6 \end{bmatrix} \begin{bmatrix} s_3 \\ s_4 \\ s_5 \end{bmatrix} + \begin{bmatrix} e_5 \\ e_6 \\ e_7 \\ e_8 \end{bmatrix}$$

$$\begin{bmatrix} x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} = \begin{bmatrix} x_1^2 + x_2^2 \\ 2x_1^3 + x_3^2 \\ x_5^2 + x_6^2 \\ 2x_6^3 \end{bmatrix} + \begin{bmatrix} e_9 \\ e_{10} \\ e_{11} \\ e_{12} \end{bmatrix}$$

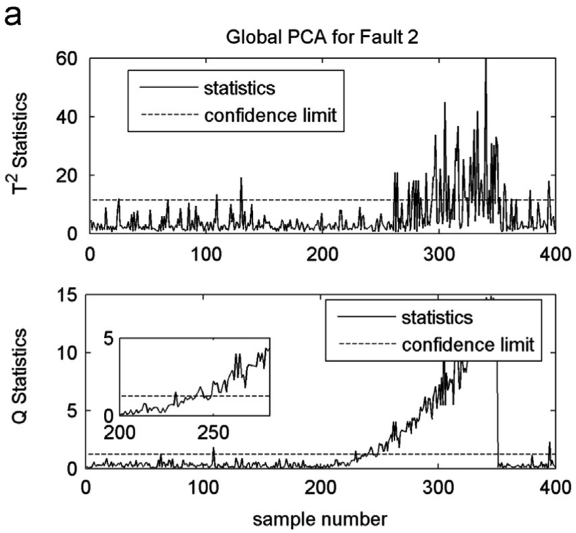
- Introduce faults:
- **Fault 1** – A step change of 0.25 is added to x_1 from sample 201
- **Fault 2** – A ramp change of $0.008(i - 200)$ is added to x_5 from sample 201 to 350, i denotes sample no.

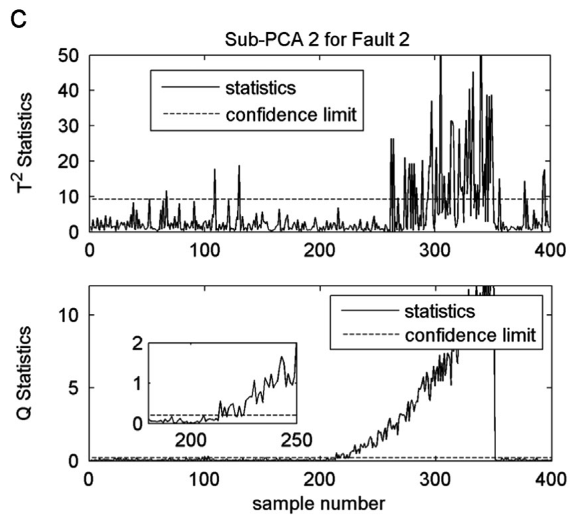
Case study cont..

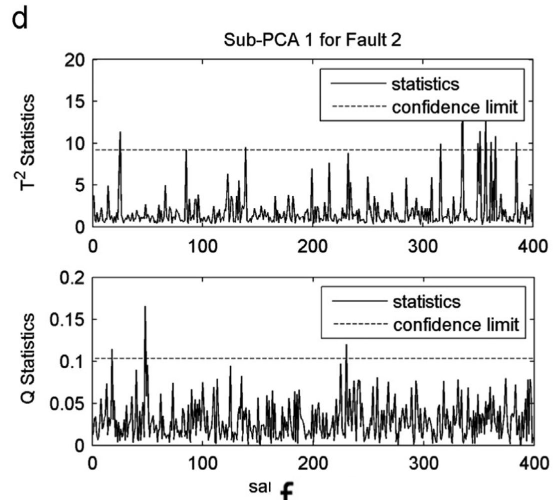
- Two approaches:
  1. **Global PCA** – treat X as a lumped sum dataset
  2. **Multi-block PCA** – divide X into 2 sub-datasets:
     - *i.* $X_1 = [x_1, x_2, x_3, x_4, x_9, x_{10}]$
     - *ii.* $X_2 = [x_5, x_6, x_7, x_8, x_{11}, x_{12}]$
- Global PCA failed to detect the fault 1
- Multi-block PCA can detect the fault 1
- Variables responsible for the fault 1 identified, i.e., x_1, x_9, x_{10}

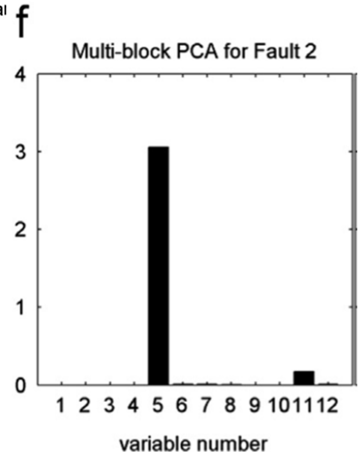


Case study cont..





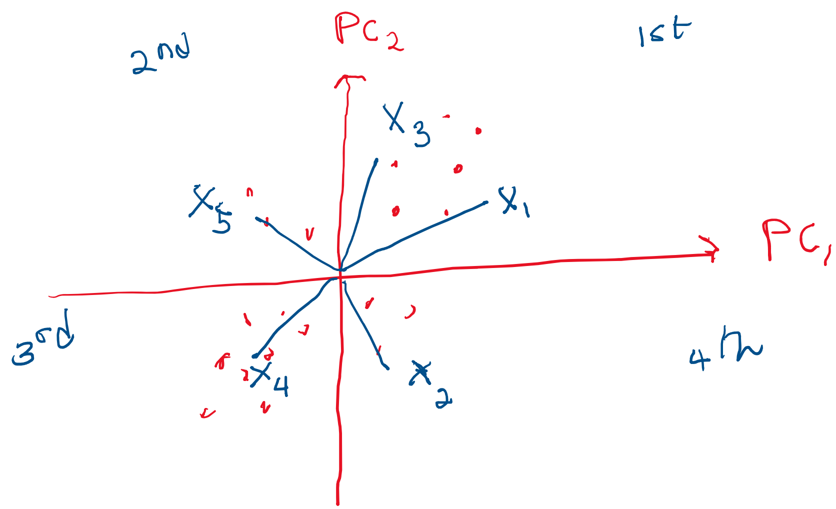




- Global PCA can detect the fault 2 but multi-block PCA **can detect it faster**
- Fault 2 is detected in the second block
- No detection in the first block

Division of dataset into blocks

- Consider a dataset
$$X = [x_1, x_2, x_3, x_4, x_5]$$
- Apply PCA to X
- Plot PC1 and PC2
- From the plot, variables in the same and opposite quadrants are related
  - Same quadrant – positively correlated
  - Opposite quadrant – negatively correlated
  - Variables in 1st and 3rd quadrants are related among each others, but not related to variables in 2nd and 4th quadrants
  - Variables in 2nd and 4th quadrants are related



Divide the dataset into two blocks:

i. $X_1 = [x_1, x_3, x_4]$

ii. $X_2 = [x_2, x_5]$

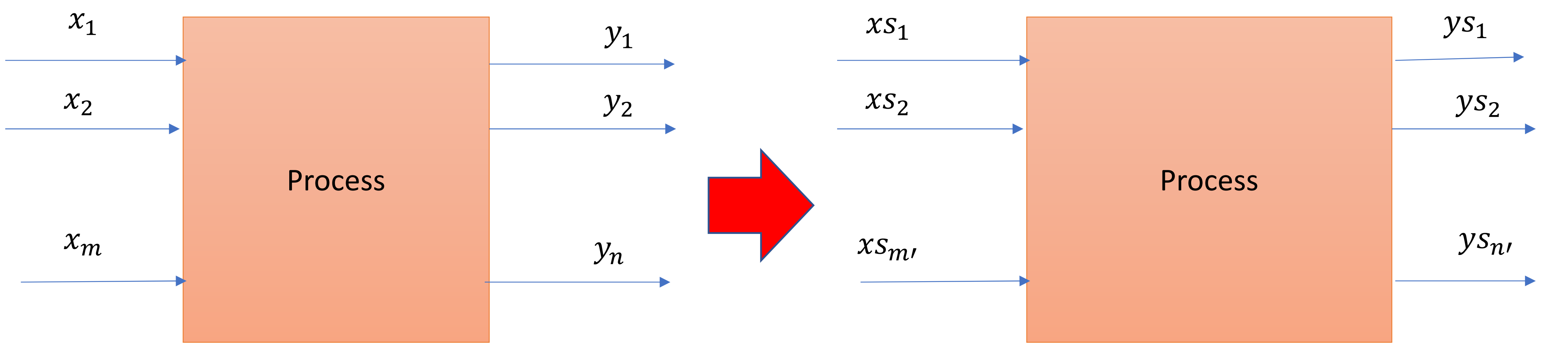
Applications of PCA

1. Fault detection, e.g., faulty sensor
2. Product quality monitoring
3. Data set reduction technique – help in data analysis
4. Soft sensor modelling
5. Detection and diagnosis process abnormalities

Partial Least Squared (PL) Model

- Projections of predictors (X) and responses (Y) to latent spaces (XS and YS)
- $X \Longrightarrow XS,\ Y \Longrightarrow YS$
- Find the model that correlate XS to YS

Partial Least Square (PLS) - Illustration



$\mathbf{X} = [x_1, x_2, \dots x_m]$ $\quad$ $\mathbf{Y} = [y_1, y_2, \dots y_n]$ $\quad$ $\mathbf{XS} = [xs_1, xs_2, \dots xs_{m'}]$ $\quad$ $\mathbf{YS} = [ys_1, ys_2, \dots ys_{n'}]$

- Original system on the (left) is reduced to system (right) with smaller numbers of variables
- $m' < m$ and $n' < n$

PLS Model

- Multivariate model

$$X = TP^T + E_X, \qquad Y = UQ^T + E_Y$$

Note:

- $X \in R^{n\times m}$ matrix of predictors, $T \in R^{n\times l}$ matrix of projections of X (scores), $P \in R^{m\times l}$ orthogonal loading matrix, $E_X \in R^{n\times m}$ error matrix
- $Y \in R^{n\times p}$ matrix of responses, $U \in R^{n\times l}$ matrix of projections of Y (scores), $Q \in R^{m\times l}$ orthogonal loading matrix, $E_Y \in R^{n\times p}$ error matrix

Note:

- Notation $R^{n\times p}$ denote a matrix with n number of rows (observations) and p number of columns (responses)

Multivariate Regression

- **Linear relationship** between one or more output or response variables (Y) with one or more input variables (X)
- 3 cases:
  1. Simple linear regression: one response $Y \in R^{n\times 1}$ and one predictor $X \in R^{n\times 1}$
  2. Multiple linear regression: one response $Y \in R^{n\times 1}$ and several predictor $X \in R^{n\times m}$
  3. **Multivariate multiple linear regression**: several $Y \in R^{n\times p}$ and several $X \in R^{n\times m}$

Multivariate regression usually refers to **Case 3**

Multiple Regression

- For the fixed-x regression model, each y in sample of $\boldsymbol{n}$ **observations** is a linear function of the $x's$ plus random error ε:

$$\left.\begin{array}{c} y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_q x_{1q} + \varepsilon_1 \\ y_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_q x_{2q} + \varepsilon_2 \\ \vdots \\ y_n = \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_q x_{nq} + \varepsilon_n \end{array}\right\} \Rightarrow \mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

- Where $\boldsymbol{\beta} = \left[\beta_0, \beta_1, \beta_2, \dots, \beta_q\right]^T$ are called **regression coefficients**
- Number of inputs used in the model is q
- **Main assumptions: Model is linear** and **error** terms are **uncorrelated**

Multivariate Multiple Regression

- Consider n observed values of the vector of $Y \in R^{n \times p}$ listed as rows:

$$\mathbf{Y} = \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1p} \\ y_{21} & y_{22} & \cdots & y_{2p} \\ \vdots & \vdots & & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{np} \end{pmatrix}$$

- Each **row** of $\mathbf{Y}$ contains the values of the p dependent variables measured at a given time or observation
- Each **column** of $\mathbf{Y}$ consists of the n observations on one of the p variables

Multivariate Multiple Regression

- The n values of $x_1, x_2, \ldots, x_q$ can be represented as follows

$$\mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1q} \\ 1 & x_{21} & x_{22} & \cdots & x_{2q} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nq} \end{pmatrix}$$

- Assume X is fixed from sample to sample i.e., for all observations
- Equation:

$$\therefore \ \mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

Illustrative example for $p = 2, q = 3$

$$\begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \\ \vdots & \vdots \\ y_{n1} & y_{n2} \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & x_{13} \\ 1 & x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & x_{n3} \end{pmatrix} \begin{pmatrix} \beta_{01} & \beta_{02} \\ \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \\ \beta_{31} & \beta_{32} \end{pmatrix} + \begin{pmatrix} \varepsilon_{11} & \varepsilon_{12} \\ \varepsilon_{21} & \varepsilon_{22} \\ \vdots & \vdots \\ \varepsilon_{n1} & \varepsilon_{n2} \end{pmatrix}$$

Model for the first column:

$$\begin{pmatrix} y_{11} \\ y_{21} \\ \vdots \\ y_{n1} \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & x_{13} \\ 1 & x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & x_{n3} \end{pmatrix} \begin{pmatrix} \beta_{01} \\ \beta_{11} \\ \beta_{21} \\ \beta_{31} \end{pmatrix} + \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{n1} \end{pmatrix},$$

Least Squares Estimation

- Sum of Squares Error (SSE):

$$SSE = \sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} - \cdots - \hat{\beta}_q x_{iq}\right)^2$$

- Values of $\widehat{\boldsymbol{\beta}} = \left[\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_q\right]^T$ **that minimizes SSE can be calculated as:**

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$$

where $\mathbf{y} = [y_{i1}, y_{i2}, \ldots, y_{in}]^T$

Assumption: $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ is non-singular, i.e., it can have an inverse

Model based on deviated variables from their means ("center")

- The inputs $(x_{i1}, x_{i2}, \dots, x_{iq})$can be cantered by subtracting their means:

$$\bar{x}_1 = \sum_{i=1}^{n} x_{i1}/n\,,\ \bar{x}_2 = \sum_{i=1}^{n} x_{i2}/n\,,\ \dots,\ \bar{x}_q = \sum_{i=1}^{n} x_{iq}/n$$

- Model can be written as follows for the observation i

$$\boldsymbol{y_i = \alpha + \beta_1(x_{i1} - \bar{x}_1) + \beta_2(x_{i2} - \bar{x}_2) + \cdots + \beta_q\left(x_{iq} - \bar{x}_q\right)}$$

$$Input\ matrix\!:\ \ \mathbf{X_c} = \begin{pmatrix} x_{11} - \bar{x}_1 & x_{12} - \bar{x}_2 & \dots & x_{1q} - \bar{x}_q \\ x_{21} - \bar{x}_1 & x_{22} - \bar{x}_2 & \dots & x_{2q} - \bar{x}_q \\ \vdots & \vdots & & \vdots \\ x_{n1} - \bar{x}_1 & x_{n2} - \bar{x}_2 & \dots & x_{nq} - \bar{x}_q \end{pmatrix} = \begin{pmatrix} (\mathbf{x_1} - \bar{\mathbf{x}}_\mathbf{1})' \\ (\mathbf{x_2} - \bar{\mathbf{x}}_\mathbf{2})' \\ \vdots \\ (\mathbf{x_n} - \bar{\mathbf{x}}_\mathbf{n})' \end{pmatrix}$$

Model based on deviated variables from their means ("center")

- Minimization of SSE can be expressed as follows

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}_\mathbf{c}^\mathbf{T}\mathbf{X}_\mathbf{c}\right)^{-\mathbf{1}}\mathbf{X}_\mathbf{c}^\mathbf{T}\mathbf{y}$$

Where

$$\widehat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_q \end{pmatrix}, \qquad \widehat{\boldsymbol{\alpha}} = \bar{\mathbf{y}}$$

- Note $\widehat{\boldsymbol{\beta}}$, $\widehat{\boldsymbol{\alpha}}$ and $\bar{\mathbf{y}}$ are vectors of q rows and one column

Multivariate Multiple Regression

$$\mathbf{Y} = \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1p} \\ y_{21} & y_{22} & \cdots & y_{2p} \\ \vdots & \vdots & & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{np} \end{pmatrix} = \begin{pmatrix} \mathbf{y}'_1 \\ \mathbf{y}'_2 \\ \vdots \\ \mathbf{y}'_n \end{pmatrix}. \qquad \mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1q} \\ 1 & x_{21} & x_{22} & \cdots & x_{2q} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nq} \end{pmatrix}.$$

$$\widehat{\mathbf{B}} = \left(\mathbf{X}^{\mathbf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{Y}$$

For p response variables, q input variables and n observations or samples

$$\widehat{\mathbf{B}} = \begin{pmatrix} \hat{\beta}_{01} & \hat{\beta}_{02} & \cdots & \hat{\beta}_{0p} \\ \hat{\beta}_{11} & \hat{\beta}_{12} & \cdots & \hat{\beta}_{1p} \\ \vdots & \vdots & & \vdots \\ \hat{\beta}_{q1} & \hat{\beta}_{q2} & \cdots & \hat{\beta}_{qp} \end{pmatrix}$$

Example 1

- y_1 = percentage of unchanged starting material
- y_2 = percentage of converted to the desired product
- y_3 = percentage of unwanted by-product

Table 10.1. Chemical Reaction Data

Experiment Number	Yield Variables			Input Variables		
	y_1	y_2	y_3	x_1	x_2	x_3
1	41.5	45.9	11.2	162	23	3
2	33.8	53.3	11.2	162	23	8
3	27.7	57.5	12.7	162	30	5
4	21.7	58.8	16.0	162	30	8
5	19.9	60.6	16.2	172	25	5
6	15.0	58.0	22.6	172	25	8
7	12.2	58.6	24.5	172	30	5
8	4.3	52.4	38.0	172	30	8
9	19.3	56.9	21.3	167	27.5	6.5
10	6.4	55.4	30.8	177	27.5	6.5
11	37.6	46.9	14.7	157	27.5	6.5
12	18.0	57.3	22.2	167	32.5	6.5
13	26.3	55.0	18.3	167	22.5	6.5
14	9.9	58.9	28.0	167	27.5	9.5
15	25.0	50.3	22.1	167	27.5	3.5
16	14.1	61.1	23.0	177	20	6.5
17	15.2	62.9	20.7	177	20	6.5
18	15.9	60.0	22.1	160	34	7.5
19	19.6	60.6	19.3	160	34	7.5

Example 1 cont..

$$\hat{\mathbf{B}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

$$= \begin{pmatrix} 332.11 & -26.04 & -164.08 \\ -1.55 & .40 & .91 \\ -1.42 & .29 & .90 \\ -2.24 & 1.03 & 1.15 \end{pmatrix}.$$

```
>> [Xd,txt,raw] = xlsread('ChemReacData');
>> Y = Xd(:,2:4);
>> X = Xd(:,5:7);
>> I = ones(19,1);
>> Xi = [I X];
>> Beta = inv(Xi'*Xi)*Xi'*Y;
>> Beta
Beta =
 332.1110  -26.0353 -164.0789
  -1.5460    0.4046    0.9139
  -1.4246    0.2930    0.8995
  -2.2374    1.0338    1.1535
```

$$y_{i1} = 332.111 - 1.546x_{i1} - 1.4246x_{i2} - 2.2374x_{i3}$$

Partial Least Square Regression (PLSR)

- Method specifically to address the prediction in multivariate problems
- Very large p (no. of variables) and small n (no. of samples) can cause poor partial least square (PLS) regression results
- Need to find relevant sub-space in the p −dimensional variable space when the no. of variable increases
- Variable selection can help improve model interpretation and prediction performance (i.e., remove redundancy in the model)
- Assumption: Explanatory (input) variables X are linked to a response variable y, e.g., $y = \alpha + \mathbf{X\beta} + \ \ldots$

PLSR Algorithm in MATLAB

- MATLAB function called "plsregress"
- [XL,YL,XS,YS,BETA] = plsregress(X,Y,NCOMP,...) returns the PLS regression coefficients BETA.
- X is an n-by-p matrix of predictor variables (explanatory variables)
- Y is an n-by-m response matrix
- XL is a p-by-NCOMP matrix of predictor loadings
    - Each row of XL contains coefficients that define a linear combination of PLS components that approximate the original predictor variables
- YL is an m-by-NCOMP matrix of response loadings
    - Each row of YLOADINGS contains coefficients that define a linear combination of PLS components that approximate the original response variable

plsregress

Partial least-squares regression

collapse all in page

Syntax

```
[XL,YL] = plsregress(X,Y,ncomp)
[XL,YL,XS] = plsregress(X,Y,ncomp)
[XL,YL,XS,YS] = plsregress(X,Y,ncomp)
[XL,YL,XS,YS,BETA] = plsregress(X,Y,ncomp,...)
[XL,YL,XS,YS,BETA,PCTVAR] = plsregress(X,Y,ncomp)
[XL,YL,XS,YS,BETA,PCTVAR,MSE] = plsregress(X,Y,ncomp)
[XL,YL,XS,YS,BETA,PCTVAR,MSE] = plsregress(...,param1,val1,param2,val2,...)
[XL,YL,XS,YS,BETA,PCTVAR,MSE,stats] = plsregress(X,Y,ncomp,...)
```

Description

`[XL,YL] = plsregress(X,Y,ncomp)` computes a partial least-squares (PLS) regression of `Y` on `X`, using `ncomp` PLS components, and returns the predictor and response loadings in `XL` and `YL`, respectively. `X` is an n-by-p matrix of predictor variables, with rows corresponding to observations and columns to variables. `Y` is an n-by-m response matrix. `XL` is a p-by-`ncomp` matrix of predictor loadings, where each row contains coefficients that define a linear combination of PLS components that approximate the original predictor variables. `YL` is an m-by-`ncomp` matrix of response loadings, where each row contains coefficients that define a linear combination of PLS components that approximate the original response variables.

`[XL,YL,XS] = plsregress(X,Y,ncomp)` returns the predictor scores `XS`, that is, the PLS components that are linear combinations of the variables in `X`. `XS` is an n-by-`ncomp` orthonormal matrix with rows corresponding to observations and columns to components.

`[XL,YL,XS,YS] = plsregress(X,Y,ncomp)` returns the response scores `YS`, that is, the linear combinations of the responses with which the PLS components `XS` have maximum covariance. `YS` is an n-by-`ncomp` matrix with rows corresponding to observations and columns to components. `YS` is neither orthogonal nor normalized.

`plsregress` uses the SIMPLS algorithm, first centering `X` and `Y` by subtracting off column means to get centered variables `X0` and `Y0`. However, it does not rescale the columns. To perform PLS with standardized variables, use `zscore` to normalize `X` and `Y`.

If `ncomp` is omitted, its default value is `min(size(X,1)-1,size(X,2))`.

The relationships between the scores, loadings, and centered variables `X0` and `Y0` are:

`XL = (XS\X0)' = X0'*XS`,

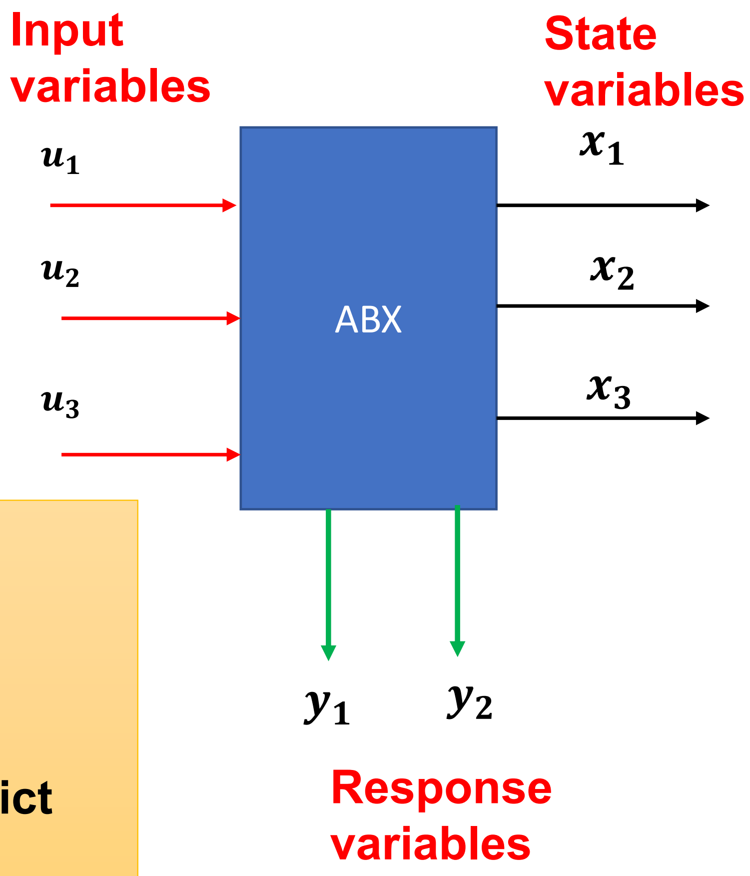
`YL = (XS\Y0)' = Y0'*XS`,

MATLAB PLS - plsregress

Illustrative Example – Process ABX

Table 1. Sample data

	Predictors						Responses	
Observation	u1	u2	u3	x1	x2	x3	y1	y2
1	-0.562	-0.562	-0.562	0.000	0.000	0.000	1.000	-0.006
2	-0.562	-0.562	-0.562	-0.177	-0.196	0.459	-0.610	0.053
3	-0.906	-0.906	-0.906	-0.056	-0.336	0.768	-1.179	-0.487
4	-0.906	-0.906	-0.906	0.052	-0.505	1.247	-2.050	-0.728
5	0.358	0.358	0.358	0.338	-0.563	1.544	-1.564	0.063
6	0.358	0.358	0.358	1.030	-0.078	0.672	1.867	-0.363
7	0.359	0.359	0.359	0.987	0.342	0.032	3.573	-0.036
8	0.359	0.359	0.359	0.640	0.579	-0.432	4.484	0.511
9	0.869	0.869	0.869	0.219	0.635	-0.729	4.137	1.474
10	0.869	0.869	0.869	0.009	0.739	-1.298	4.760	1.789

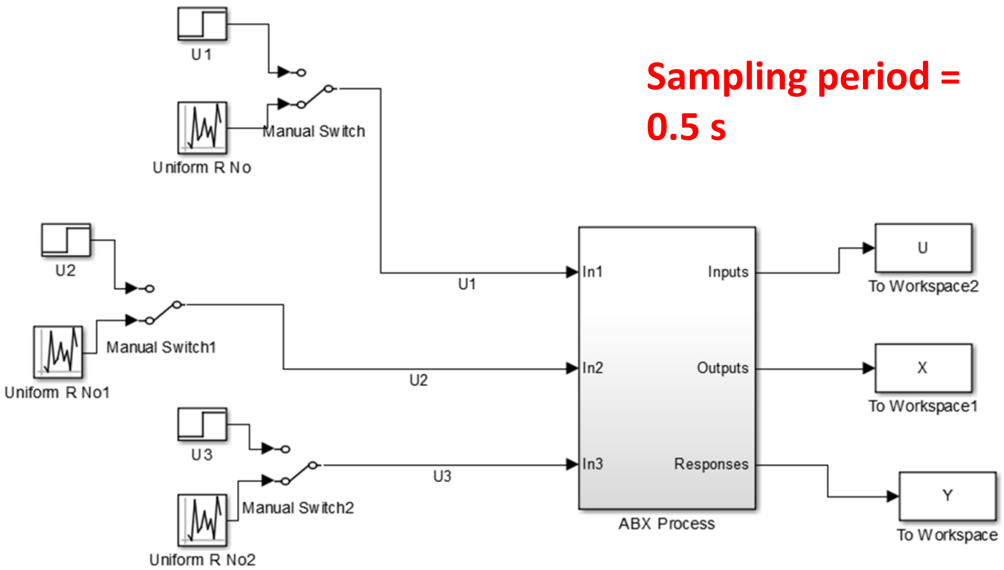


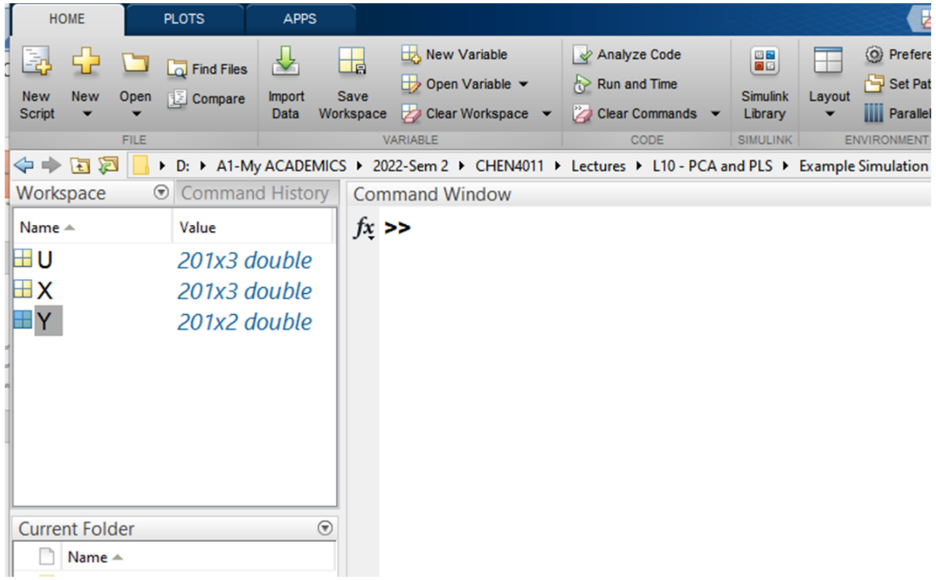
- Predictors: $X = [u_1, u_2, u_3, x_1, x_2, x_3]$
- Responses: $Y = [y_1, y_2]$
- Sample data shown in Table 1, i.e., up to 10 observations
- The dataset has 201 observations in total
- **Thus, $X \in R^{201\times 6}$ and $Y \in R^{201\times 2}$**
- **Use *plsregress* function to obtain a PLS model to predict the responses**

MATLAB - plsregress function

- Load the data to MATLAB workspace
- If the data is in excel, copy each column of the data and paste to Command Window, e.g.,
- >> x1 = [   ]; % paste the data inside the brackets
- Then, after all data has been uploaded to workspace, we can combine the data together to form the predictor matrix, and response matrix.
- If the system is in Simulink, run the system to generate the data
- In this example, data is generated by running a Simulink model

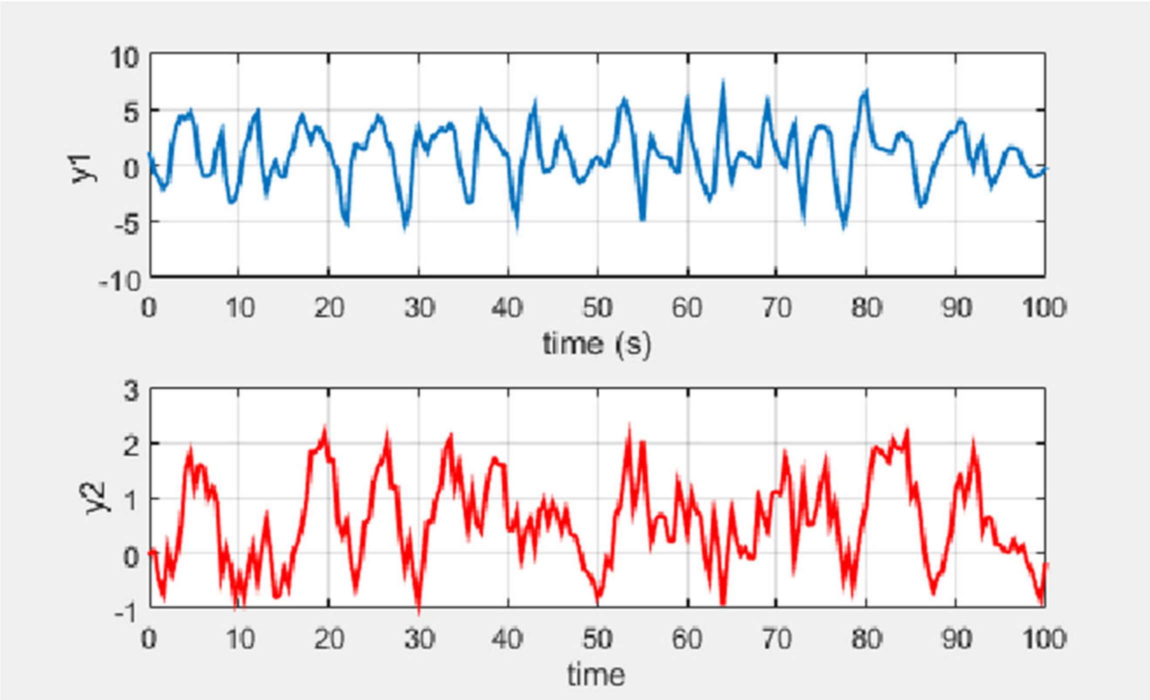
Sampling period = 0.5 s

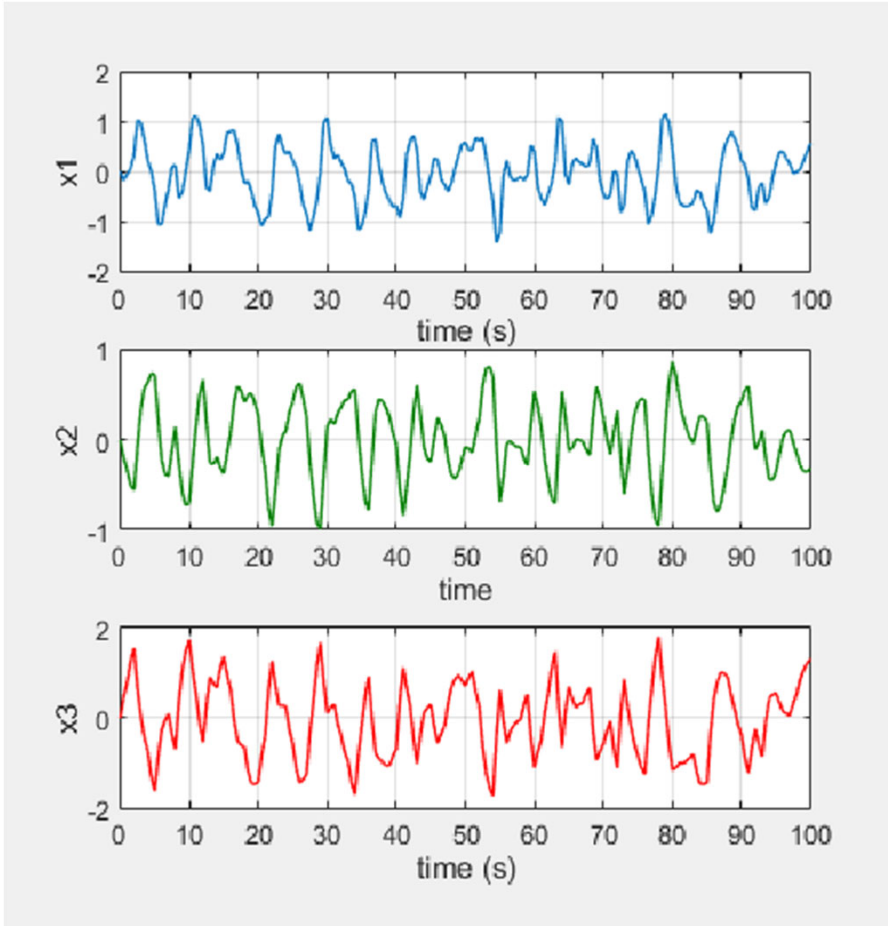




- Data quality is important to develop a data-driven model, e.g., ANN, PLS and many others
- Data quality depends on it is generated, sampling period, number of observations, missing data, etc.
- Sampling period cannot be too large as this can lead a significant loss of information
- Short sampling period can capture most of the information but this can lead dimensionality issue, e.g., storage capacity.
- In practice, trade-off between information preservation and storage/processing capacity is required

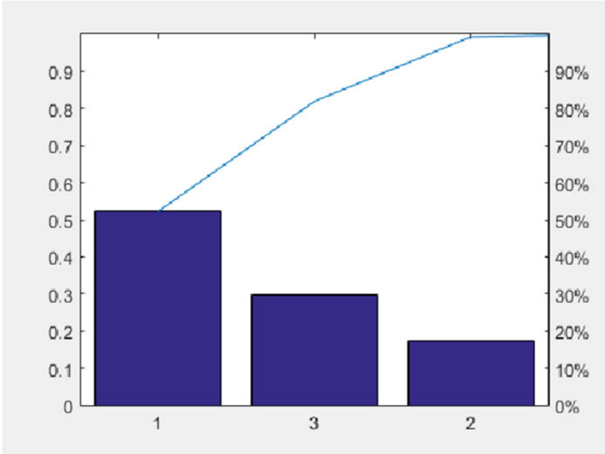
Illustrative example - data pattern





Example - plsregress

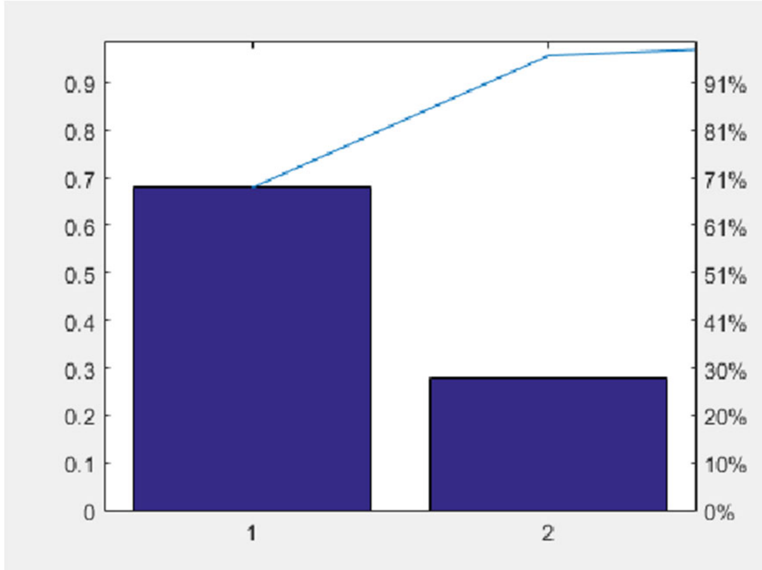
- Let’s try 4 PLS components
- >> [XL,YL,XS,YS,BETA,PCTVAR,MSE] = plsregress(Xd,Y,4);
- Variances of principal components
  - >> PCTVAR
  - PCTVAR =
  - 0.5227 0.1735 0.2962 0.0077
  - 0.6792 0.2776 0.0257 0.0021
- First row, variances for the predictors
- Second row, variances for the responses
- Use pareto plot, e.g.:
- >>pareto(PCTVAR(1,:)); % pareto plot of the predictor variance



- The sum of variances of the first 3 PLS components > 90%
- 3 PLS components are sufficient

Example – plsregress - MSE

- Pareto plot of the response variances
- >> pareto(PCTVAR(2,:));
- Sum of variances of 2 PLS components > 90%
- Check Mean Squared Error (MSE)
  - >> MSE
  - MSE =
  - 2.1694 1.0356 0.6592 0.0167 0.0000
  - 6.5541 2.1028 0.2834 0.1150 0.1011
  - First row for the predictors, second row for the responses
  - MSE getting smaller from left to right, 1^{st} PLS component (column 2), 2^{nd} PLS component (column 3), etc.
  - 1^{st} column in MSE are constants, e.g., steady-state values



Example – plsregress – XL, YL, XS, YS and BETA

- Predicted $X = XL * YS$ (loading matrix and score matrix)
- Predicted $Y = YL * YS$
- Regression coefficients, BETA
  - >> BETA
  - BETA =
  - 0.8561 0.5421
  - 0.5860 0.1522
  - 0.5860 0.1522
  - 0.5860 0.1522
  - 0.5215 0.1354
  - 0.8835 0.2295
  - -1.3988 -0.3633
  - No. rows the same as number of predictors
  - No. columns the same as number of responses

PLS models

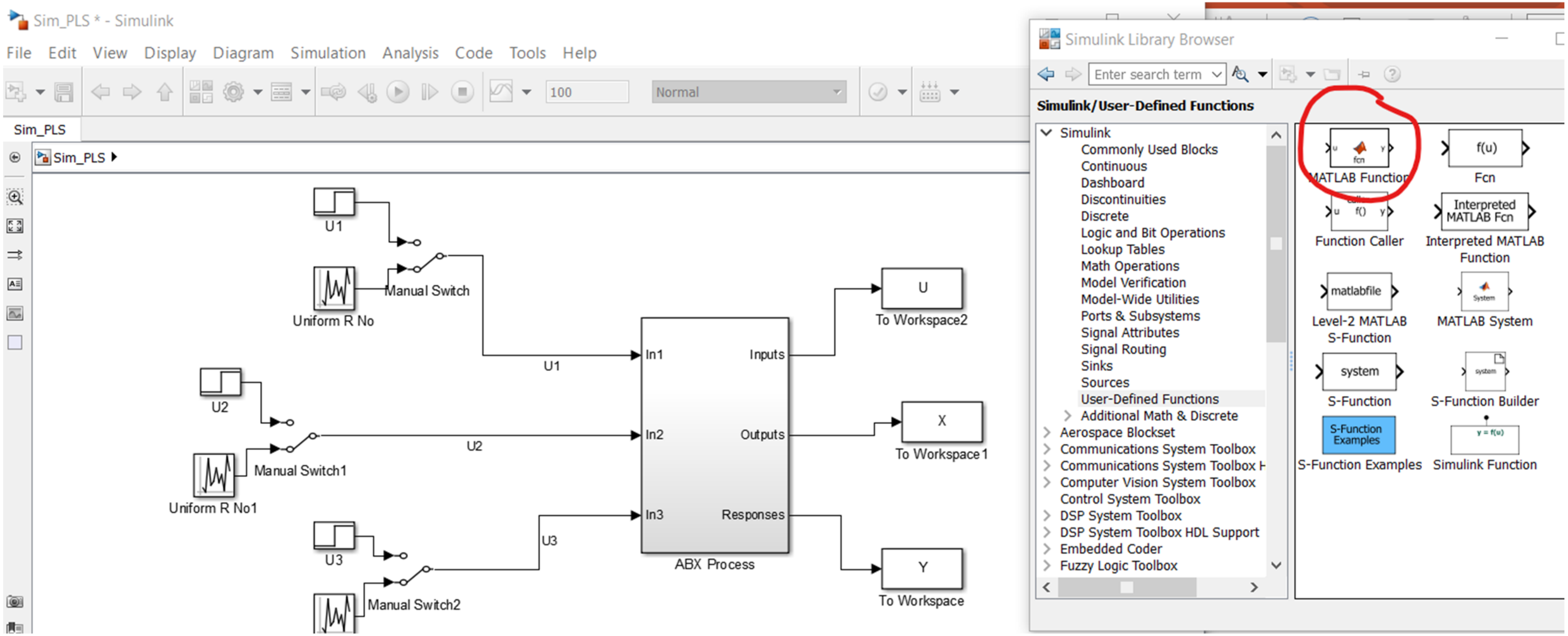
- The PLS model to predict the first response y_1

$$y_1 = 0.8561 + 0.5860u_1 + 0.5860u_2 + 0.5860u_3 + 0.5215x_1 + 0.8835x_2 - 1.3988x_3$$

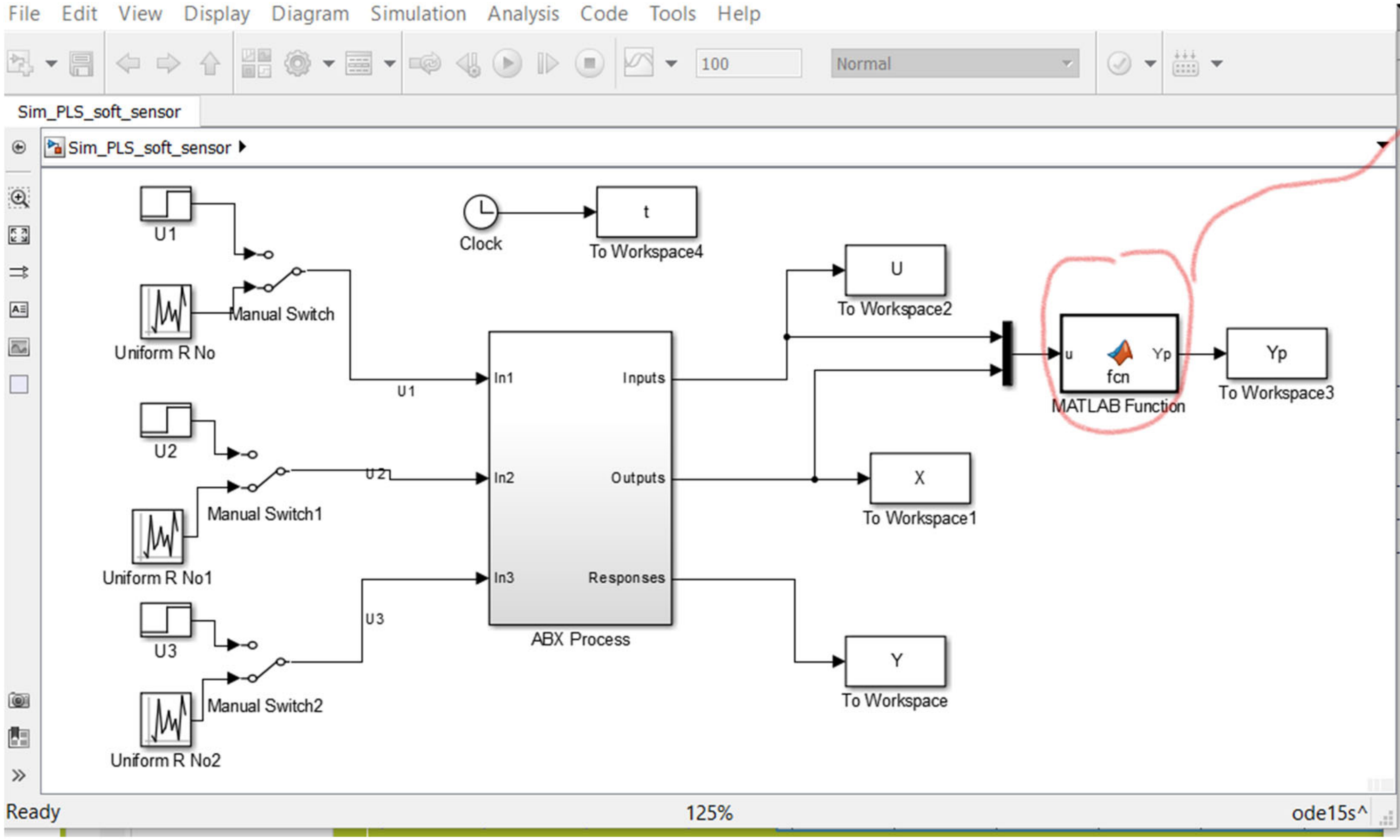
- The PLS model to predict the second response y_2

$$y_2 = 0.5421 + 0.1522u_1 + 0.1522u_2 + 0.1522u_3 + 0.1354x_1 + 0.2295x_2 - 0.3633x_3$$

- Let us apply the models for online predictions of y_1 and y_2



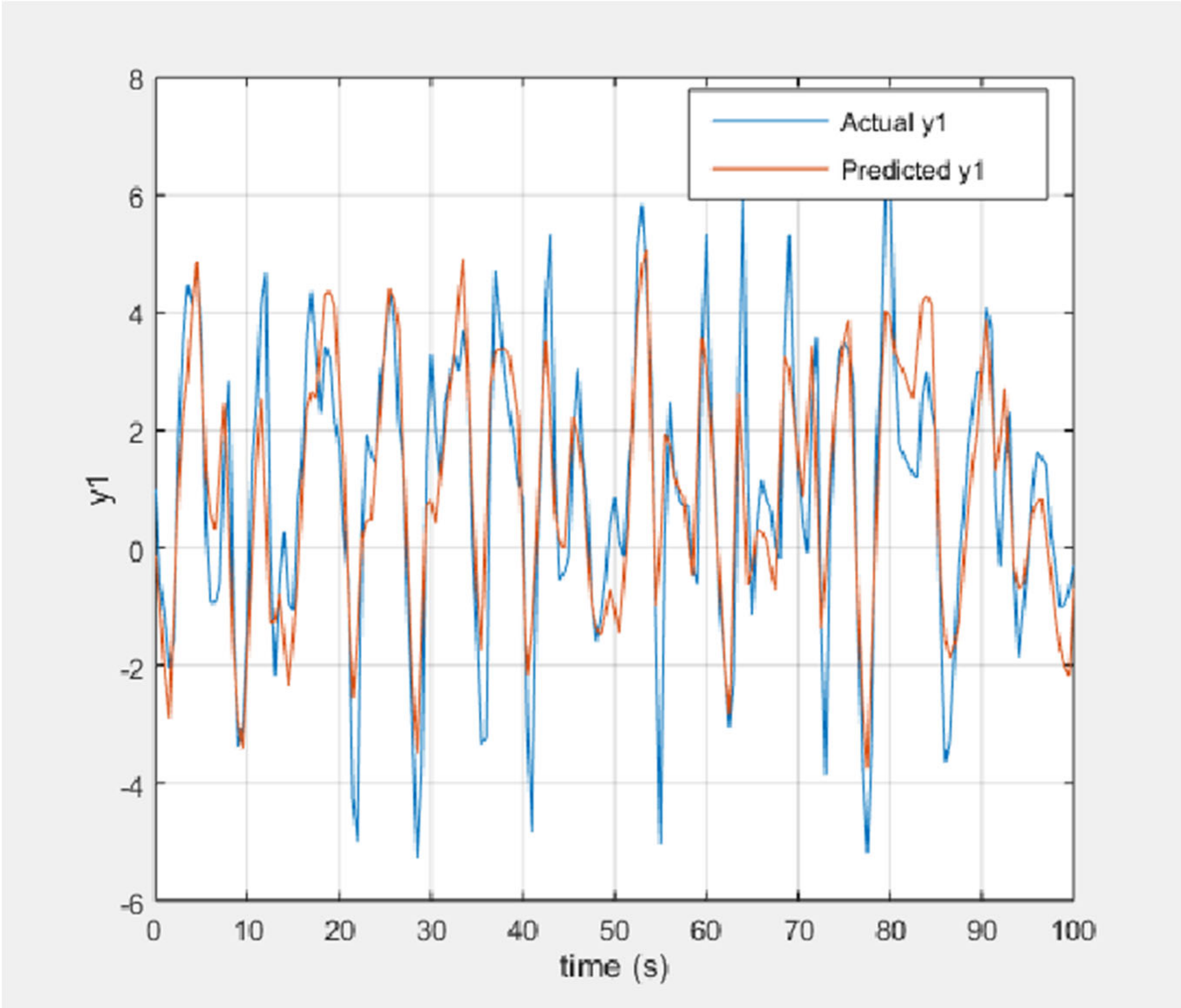
Sim_PLS * - Simulink
File Edit View Display Diagram Simulation Analysis Code Tools Help
100
Normal
Sim_PLS
U1
Manual Switch
Uniform R No
U1
U2
U2
Manual Switch1
Uniform R No1
U3
U3
Manual Switch2
In1
In2
In3
Inputs
Outputs
Responses
ABX Process
U
To Workspace2
X
To Workspace1
Y
To Workspace
Simulink Library Browser
Enter search term
Simulink/User-Defined Functions
Simulink
Commonly Used Blocks
Continuous
Dashboard
Discontinuities
Discrete
Logic and Bit Operations
Lookup Tables
Math Operations
Model Verification
Model-Wide Utilities
Ports & Subsystems
Signal Attributes
Signal Routing
Sinks
Sources
User-Defined Functions
Additional Math & Discrete
Aerospace Blockset
Communications System Toolbox
Computer Vision System Toolbox
Control System Toolbox
DSP System Toolbox
DSP System Toolbox HDL Support
Embedded Coder
Fuzzy Logic Toolbox
MATLAB Function
f(u)
Fcn
Function Caller
Interpreted MATLAB Fcn
Interpreted MATLAB Function
matlabfile
Level-2 MATLAB S-Function
MATLAB System
system
S-Function
S-Function Builder
S-Function Examples
y = f(u)
Simulink Function



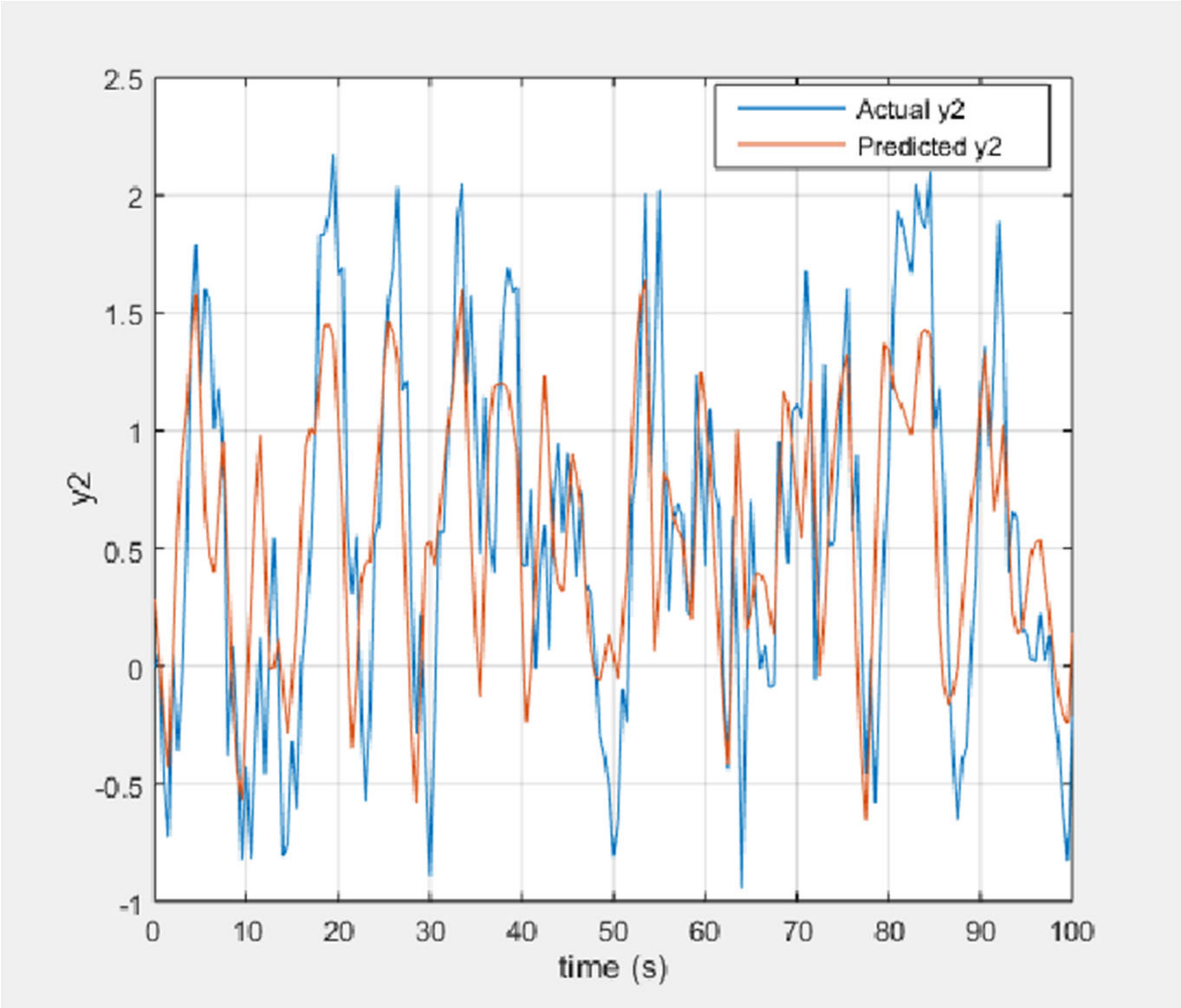
MATLAB Function

```
function Yp = fcn(u)
%#codegen
u1  = u(1);
u2  = u(2);
u3  = u(3);
x1  = u(4);
x2  = u(5);
x3  = u(6);
% PLS models
y1 = 0.8561 + 0.5860*u1 + 0.5860*u2 + 0.5860*u3 +...
        0.5215*x1 + 0.8835*x2 - 1.3988*x3;
y2 = 0.5421 + 0.1522*u1 + 0.1522*u2 + 0.1522*u3 +...
        0.1354*x1 + 0.2295*x2 - 0.3633*x3;

Yp = [y1; y2];
```



Actual y1
Predicted y1
y1
time (s)
8
6
4
2
0
-2
-4
-6
0
10
20
30
40
50
60
70
80
90
100


Actual y2
Predicted y2
y2
time (s)
2.5
2
1.5
1
0.5
0
-0.5
-1
0
10
20
30
40
50
60
70
80
90
100

Summary

- Process plant monitoring is crucial to ensure safe and profitable operation
- Early detection of faulty sensor, or process abnormalities can improve safety and profit
- Principal Component Analysis (PCA) – reduce dataset dimensionality for data analysis
- PCA projects the original dataset X onto principal component space, i.e., latent variables
- PCA has many applications in process industry
- Technological advances have reduced the data acquisition
- Huge data is available – including irrelevant information
- Modelling of the system using the data help in the predictions of key variables or responses, and interpret the system
- Multivariate Multiple Regression – to model the relationships between several input (explanatory) variables and several response variables
- Partial Least Square – projects the explanatory (predictor) variables and response variables to laten space
- PLS model - multidimensional direction in the X space correspond to maximum multidimensional variance direction in the Y space