Robust and Adaptive Bayesian Network Soft Sensor Development for Multi-Rate and Noisy Data

by

Anudari Khosbayar

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in
Chemical Engineering

Department of Chemical and Materials Engineering
University of Alberta

Abstract

For efficient process control and monitoring, accurate real-time information of quality variables is essential. To predict these quality (or slow-rate) variables at a fast-rate, in the industry, inferential/soft sensors are often used. However, most of the conventional methods for soft sensors do not utilize prior process knowledge even if it is available. The prediction accuracy of these inferential sensors depends mainly on the quality of available data, which can be affected by significant noise, outliers, drift and possible sensor failures. To address these issues, in this work, soft sensors based on Bayesian network (BN) are developed. Compared to the existing soft sensors, the proposed approach will allow users to integrate prior knowledge into the BN structure. Due to the probabilistic nature of BNs, variances of measurement noises and disturbances between hidden states are simultaneously estimated. Moreover, BN based soft sensor can naturally handle multi-rate, missing data, outliers or the problem of drift, which usually arises during online soft sensor implementation stage. Performance of the proposed approach is demonstrated on a benchmark flow-network problem and an industrial process. It is observed that Bayesian network based soft sensors are able to give significantly better and more reliable estimates compared to the conventional approaches.

Acknowledgements

First and foremost, I would like to express my sincere gratitude and appreciation towards my supervisor, Prof. Biao Huang. It would have been impossible for me to be where I am today without his endless support, encouragement and advice during my M.Sc. program. He did not only gave me the great opportunity to learn and grow with the Computer Process Control (CPC) group, but also guided me through my program kindly and effortlessly. With all the thought provoking discussions, his critique and supervision, I was able to finish my degree successfully, grow so much as an individual and have a graduate experience that I will cherish forever. I must highlight professor Biao's care and kindness towards all his students. As a mother, who has a young child, it was not easy for me at times to continue my research as much as I wanted to. However, professor was always understanding, patient and flexible, which was the greatest motivation for me to lead my research confidently.

I consider myself very lucky to have this opportunity to work with all the talented and hard-working people of the CPC group. Particularly, I would like to especially thank two great members: Dr. Jayaram Valluru and Dr. Kirubakaran Velswamy. Dr. Jayaram Valluru has helped and supported me throughout my research journey by providing ideas, discussions and revisions. Dr. Kirubakaran Velswamy guided me through my industrial work and provided advice and feedbacks. I would like to thank all my colleagues for their help, listed but not limited as the follows: Aswathi Prabhakaran, Oguzhan Dogru, Jingyi Wang, Arun Senthil, Faraz Amjad, Nabil Magbool Jan, Rahul Raveendran. Lei Fan, Mengqi Fang, Yousef Alipouri, Hareem Shafi, Seraphina Kwak, Yashas Mohankumar and many else.

I am truly grateful to the industrial partners for giving me an opportunity to expose myself to a real-world problem and obtain invaluable knowledge about the oil sands industry. Especially, I would like to thank Amar Bin Halim, Emily Geba, Connor Davison, for their time and effort. Their constant support and explanations helped me to understand the particular process that I was working on. I am sending especial thanks to Connor Davison for his excellent work, genuine help that he has put into this project. I will remember him as the bright and kind young man, as he was.

I would like to acknowledge the financial support from Natural Science and Engineering Research Council of Canada.

Last but not least, I would like to express my special gratitude for my family for their constant love, support and encouragement. Especially my mother, Ariunaa Bekhchandar, to whom I will always look up to. Without her unconditional love, this perhaps was unachievable.

Table of Contents

Introduction 1

1.1 Background 1

1.2 Thesis Contribution 6

1.3 Thesis outline 8

Bayesian Network Soft Sensor Development for Multi-rate and Noisy Data 10

2.1 Introduction to Bayesian networks and Bayesian network soft sensor development 10

2.2 Modeling assumptions 13

2.3 Development of Bayesian network based soft sensor 14

2.3.1 Construction of Bayesian network structure 14

2.3.2 Parameter learning 16

2.3.3 Parameter learning for down-sampled data 17

2.3.4 Parameter learning for multi-rate/missing data 24

2.4 Inference in Bayesian networks 31

2.5 Simulation and industrial application 34

2.5.1 Flow Network 35

2.5.2 Industrial case study 41

2.6 Conclusion 49

Robust Bayesian Network Soft Sensor Development for Multi-Rate Data 51

3.1 Introduction 51

3.2 Modeling assumptions 53

3.3 Development of robust Bayesian network based soft sensor 54

3.3.1 Construction of Bayesian network structure 55

3.3.2 Robust parameter learning for down-sampled data 55

3.3.3 Robust parameter learning for multi-rate/missing data with outliers 63

3.4 Inference in Bayesian networks 69

3.5 Simulation and industrial application 70

3.5.1 Flow Network 71

3.5.2 Industrial case study 73

3.6 Conclusion 77
Adaptive Multi-Rate Bayesian Network Soft Sensor Development 79
4.1 Introduction 79
4.2 Adaptive MR-BN soft sensor development 82
4.2.1 Modeling assumptions for adaptive parameter learning 82
4.2.2 Construction of Bayesian network structure 83
4.2.3 Parameter learning of adaptive MR-BN soft sensor 83
4.2.4 Inference in adaptive MR-BN soft sensor 83
4.3 Conventional Bias update approach 92
4.4 Simulation study 93
4.4.1 Simulation study: flow-network problem 94
4.5 Conclusion 101
Conclusions 102
5.1 Summary of thesis research 102
5.2 Direction of future work 104
Appendix to Chapter 2 113
Appendix for Chapter 3 116

List of Tables

Table 2. 1: Steady state values of the process variables 36
Table 2. 2: Comparison of ARMSE values 38
Table 2. 3: Comparison of ARMSE values 40
Table 2. 4: Comparison of ARMSE values 41
Table 2. 5: Correlation coefficients 45
Table 2. 6: RMSE values 45
Table 2. 7: Correlation coefficients 46
Table 2. 8: RMSE values 47
Table 2. 9: Correlation coefficients 48
Table 2. 10: RMSE values 49
Table 2. 11: Comparison of noise variances true and estimated values 113
Table 2. 12: Comparison of true and estimated hidden noise variances 114
Table 2. 13: Comparison of true and estimated parameters 115
Table 3. 1: ARMSE of different approaches for increased output outliers……...72
Table 3. 2: ARMSE of different approaches for increased input outlier 73
Table 3. 3: Correlation coefficient for different approaches 75
Table 3. 4: RMSE of different soft sensors 75
Table 3. 5: Correlation coefficient values of different approaches 76
Table 3. 6: RMSE of different soft sensors 77
Table 3. 7: Comparison of noise variances true and estimated values 116
Table 3. 8: Comparison of true and estimated hidden noise variances 117
Table 3. 9: Comparison of true and estimated parameters 118
Table 4. 1: ARMSE values……..…….96
Table 4. 2: ARMSE values of the bias updated OLS with different forgetting factor 96
Table 4. 3: ARMSE values 99
Table 4. 4: ARMSE value estimates of the bias updated OLS with different forgetting factor ... 99
Table 4. 5: ARMSE values 101

List of Figures

Figure 2. 1: Common effect structure 11
Figure 2. 2: Chain structure 11
Figure 2. 3: Common cause structure 11
Figure 2. 4: Schematic representation of heat exchanger flow network 14
Figure 2.5: Bayesian network representation of flow-network 15
Figure 2. 6: Comparison of BN soft sensor predictions 38
Figure 2. 7: Convergences profile of the noise variances of measurements $Y5$ and $Y6$ 39
Figure 2. 8: Comparison of MR-BN soft sensor prediction 40
Figure 2. 9: Comparison of MR-BN soft sensor predictions to completely missing case 41
Figure 2. 10: Simplified process diagram of the industrial case study 42
Figure 2.11: Two-layered Bayesian network structure 43
Figure 2.12: Multi-layered Bayesian network structure 43
Figure 2.13: Comparison of different soft sensor performances 44
Figure 2.14: Comparison of different soft sensor predictions 46
Figure 2.15: Comparison of different soft sensor predictions 48
Figure 3. 1: Comparison of t robust soft sensor performances for 3% output outliers……..72
Figure 3. 2: Comparison of robust soft-sensor predictions for 10% input outliers 73
Figure 3. 3: Comparison of robust MR-BN soft sensor 74
Figure 3. 4: Comparison of robust MR-BN soft sensor for 10% input outliers 76
Figure 4. 1: Dynamic Bayesian network structure……85
Figure 4. 2: Added process drift 95
Figure 4. 3: Performance of different soft sensors 95
Figure 4. 4: Added process drift profile 97
Figure 4. 5: Performance of adaptive MR-BN soft sensor in fast-rate 98
Figure 4. 6: Performance of different soft sensors zoomed 98

Figure 4. 7: Added sensor drift 100
Figure 4. 8: Performance of different soft sensors 100

List of Abbreviations and Acronyms

ANN	*Artificial Neural Networks*
ARMSE	*Average Root Mean Squared Error*
BN	*Bayesian Network*
Corr	*Correlation Coefficient*
DS	*Down-Sampled*
DS-BN	*Down-Sampled Bayesian Network*
EM	*Expectation Maximization*
E-step	*Expectation Step*
HX	*Heat Exchanger*
MAP	*Maximum a Posteriori*
ML	*Maximum Likelihood*
MLP	*Multi- Layer Perception*
MR	*Multi-Rate*
MR-BN	*Multi-rate Bayesian Network*
M-step	*Maximization Step*
NP	*Non-Deterministic Polynomial-Time*
OLS	*Ordinary Least Squares*
PCA	*Principle Component Analysis*
PPCA	*Probabilistic Principle Component Analysis*
PPCR	*Probabilistic Principle Component Regression*
RBFN	*Radial Basis Function Networks*
RMSE	*Root Mean Squared Error*
SPL	*Splitter*
SS	*Soft Sensor*
SVM	*Support Vector Machines*
VAL	*Valve*
w.r.t	*with respect to*

Chapter 1

Introduction

1.1 Background

In the process industry, measurements of important quality variables, such as compositions, density and molecular weight are infrequently sampled, analyzed in the lab and the measurements are usually available after certain time delay. However, for efficient control and monitoring of the process, accurate and real-time information of these quality variables is essential. To overcome this challenge, in literature, inferential sensors or soft sensors are developed using the plant historical data, which can predict the slow rate variables or quality variables at a fast-rate. In the last few decades, soft sensors have become an emerging technology for advanced control applications allowing industrial users to improve productivity, save energy, reduce environmental impact and improve profitability by reducing the off-specification product[1,2,3,4].

Soft sensos that are developed from the energy, mass and material balance of the system are usually referred to as first principles model[5,6,7,8] (or *white box models*) based soft sensor. This approach requires expert knowledge, which is not always available. On the other hand, data-driven soft sensor models (or *black-box models*), which rely entirely on plant's historical data, have gained significant attention in the recent years[3,9,10,11]. Main advantage of data-driven soft sensor models is their relative ease to develop, which do not require many theoretical assumptions[1]. In between those two approaches there exist grey-box models, which integrate the first-principle models with the data-driven models[5,12] . This approach is often utilized to establish

model structures when certain process knowledge is available and the model parameters are estimated from process data.

The most popular deterministic data-driven soft sensor modeling methods are ordinary least squares (OLS), principle component analysis (PCA)[13], partial least squares (PLS)[14], artificial neural networks (ANN)[15] and support vector machines (SVM) [1]. OLS is the simplest data-driven approach that assumes target variable is a function of linear combination of input variables. PLS and PCA are dimensionality reduction algorithms used to address the issue of input data colinearity. ANN, on the other hand, is used to address the nonlinearity in the data, and it captures nonlinear behavior of the process[1,16,17]. Since process variables are contaminated with random noises, use of deterministic methods (i.e. OLS, PLS, PCA, ANN) can lead to inaccurate soft sensor models. The main drawbacks of the aforementioned non-probabilistic data-driven models are that they do not take into account the causal relation between process variables and most of them assume input variables to be noise-free. In terms of popular ANN approach, its success lies in building the network structure, which is generally a trial and error procedure. The main drawback of ANN is the inter-relation between input and output variables, which is completely black-box. Further, the learnt knowledge does not have any physical interpretation[1]. Therefore, ANN soft sensor performance mainly depends on the historical data quality[1]. Further, these approaches mitigate the issue of missing data mainly through case deletion [18], data augmentation[19] and multiple imputation[20]. Case deletion is a common approach for developing OLS based soft sensor models in the presence of missing data. In this approach, data of all other variables are deleted at instances when data of one variable is missing. This results in smaller number of samples to work with and thus loss of information.

In contrast, since the process measurements are contaminated with uncertainties, probabilistic methods are more appropriate to characterize the randomness in process data, which can be solved under probabilistic framework, such as maximum likelihood. Issues such as missing and multi-rate data, outliers and process drifts can be effectively handled by the probabilistic approaches using expectation maximization (EM) algorithm[21,22] and Bayesian inference. In literature, to account for the uncertainty in process data, such probabilistic approaches have been well developed, such as Probabilistic Principle Component Analysis (PPCA)[23], Probabilistic Principle Component Regression (PPCR)[24] and Probabilistic Partial Least Squares (PPLS)[25]. Although probabilistic models namely, PPCA and PPLS can handle multi-rate data and address the issue of uncertainty, they again suffer from the inability to utilize prior process knowledge, which is the major drawback.

Therefore, a probabilistic graphical model, namely Bayesian networks, is chosen as the base of this work. In literature, Bayesian networks have found applications in fault detection and diagnosis [26,27]. Its application to soft sensors has only gained recent attention with few industrial applications. Mohammadi et al. (2019) developed a soft sensor for quality variable prediction and fault detection, and applied it to a gas sweetening system in industrial process. The developed soft sensor is compared with PPCA for missing data scenario and it is observed that BN based soft sensor can estimate the states with better accuracy than PPCA. To address the time varying processes, Liu et al. (2018) developed adaptive prediction models (such as moving window, locally weighted, time-difference) under the Bayesian network framework and applied them to debutanizer and CO_2 absorption columns. It is observed that all the three adaptive models have a better estimation accuracy compared to PLS based adaptive models. Khatibisepehr & Huang (2008) proposed error-in-variables BN soft sensor, which accounts for

noise in the data and was illustrated on a simplified problem. From the above literature, it is evident that Bayesian network based soft sensor has shown a good potential. Therefore, in this work, Bayesian network based soft sensor is further developed to address the issues that arise during off-line soft sensor development and online implementation stages. Even though the aforementioned researches have proposed different algorithms under BNs modeling framework, this is a field that has not been well studied in depth. BN based soft sensors proposed by Mohammadi et al. (2019) and Liu et al. (2018) are developed under lenient conditions, where data is corrupted only with measurement noise and measurements of the key input variables are fully available. In reality, not only measurements, but also process states are affected by uncertainty and measurements of some key input variables may not be available due to sensor issue. Further, in literature, no general explicit analytical solution with considerations of all the above scenarios has been derived. Through this work, we are able to improve existing deterministic models by estimating the measurement and state uncertainties and predicting the true value of the target variable through probabilistic approach. Further, this work improves the existing probabilistic models by allowing users the possibility of incorporating prior information into the soft sensor modeling through application of Bayesian networks.

The proposed multi-rate Bayesian network soft sensor (MR-BN-SS) is developed under the assumption that the measurements are free of outliers i.e. noise effecting the measurements follows normal distribution with zero mean and some low variance. In general, process data is corrupted with outliers, which may be due to abrupt disturbances or sensor failures. Not accounting for these outliers in soft-sensor modeling stage can give inaccurate model. To address this issue, a robust multi-rate Bayesian network soft sensor (RMR-BN-SS) modeling approach is proposed, employing Student's t-distribution[28, 75].

Although Normal distribution is widely used in probabilistic modeling, due to its exponentially decaying tail, Normal distribution cannot handle outliers well. Just to compare these two distributions in the presence of outliers, Figure 1. 1 is illustrated. This graph compares the fit of Gaussian and Student's t-distributions to a given set of data without (top) and with (bottom) outliers. From this figure, it can be understood that as the number of outliers increases, Gaussian distribution is unable to capture the true mean and variance of the data (bottom figure) accurately. Instead it results in a slightly off mean and large variance value. t-distribution, in contrast, is capable of describing the data well, resulting in accurate statistics (bottom figure). For this reason, this work proposes a RMR-BN-SS through t-distribution.

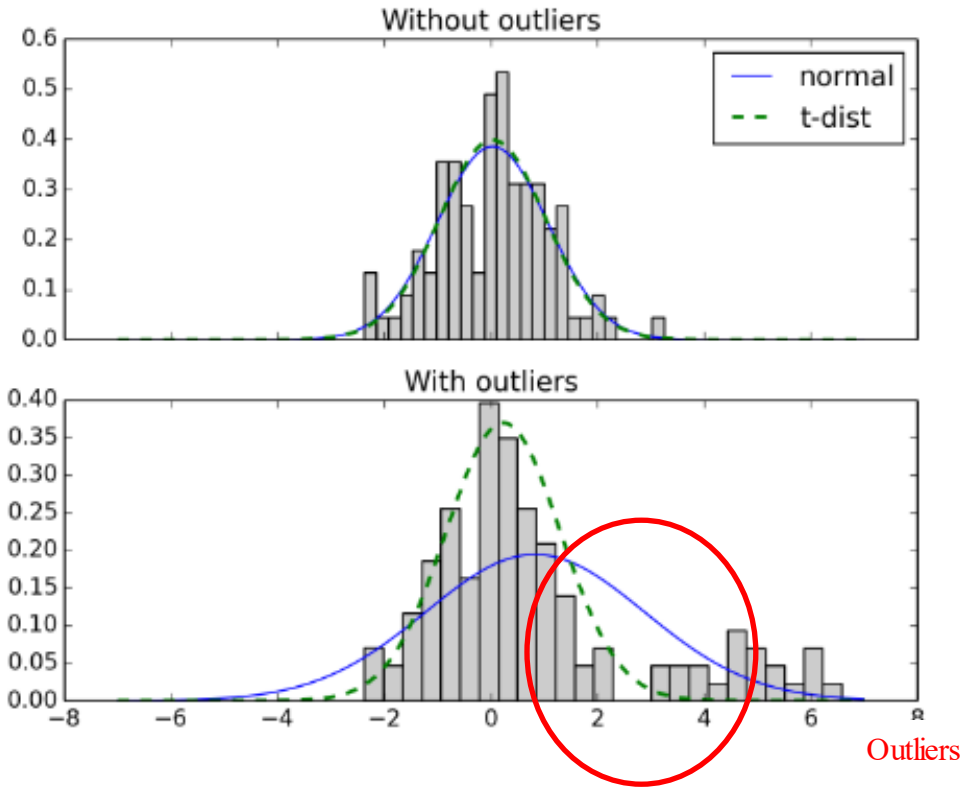

Figure 1. 1: Gaussian and Student's-t distribution fit

Soft sensor development and implementation is a multi-stage process, where the initial developed model has to be tested off-line. Once the off-line model performance is satisfactory, the performance is further validated by implementing it on open-loop real-time DCS platform. Upon satisfactory performance, predictions of the soft sensor are finally used for APC application. Due to the time dependent or drifting nature of chemical processes, real-time implementation of developed soft sensors faces the issue of process/sensor drift. Once developed soft senor is ready to be used, the process may slowly drift (i.e. may be due to catalyst deactivation, or fouling of heat exchanger) away from its initial operating conditions leading to inaccurate soft-sensor predictions. The conventional approach to address the drift is to carryout *bias update* when lab samples are available. The drawback of this approach is that, the bias update/correction to predictions is performed only when lab samples are available i.e. maybe once or twice a day. In between the availability of lab samples, the bias is kept constant as the old value leading to sub-optimal performance. To account for process/sensor drift, the inference step of the proposed multi-rate BN soft-sensor framework is extended to adaptive Bayesian inference and the soft sensor is named as adaptive multi-rate BN soft-sensor (AMR-BN-SS), where the process/sensor drift is captured through random-walk model or colored noise model, respectively. Therefore, through this proposed adaptive approach, we are motivated to adapt to drifting measurements and can achieve accurate output predictions compared to the conventional bias update approach.

1.2 Thesis Contribution

The contributions of this thesis to the existing literature are summarized below:

1. Development and analysis of two Bayesian network structures, namely two-layered and multi-layered structures. The former does not need any process information and the latter incorporates some process information.
2. A Bayesian network based soft sensor for down-sampled, multi-rate and noisy lab data is proposed, where parameter learning is carried out through EM algorithm and predictions are carried out through Bayesian inference. Analytical solutions are derived for both parameter learning and inference steps.
3. Complete missing input variable is considered in the proposed soft sensor structure.
4. Robust MR-BN-SS considering student-t distribution is proposed, where both input and output outliers are addressed. Analytical solutions are derived for both parameter learning and inference/ prediction steps.
5. Adaptive MR-BN-SS is proposed considering random walk and colored noise models, and analytical solutions are derived.
6. The performance of the proposed approaches is demonstrated on simulation of a benchmark flow-network system and on a set of industrial data.

Figure 1. 2 outline the major contributions made and common issues that this thesis have addressed through a simple flow-chart.

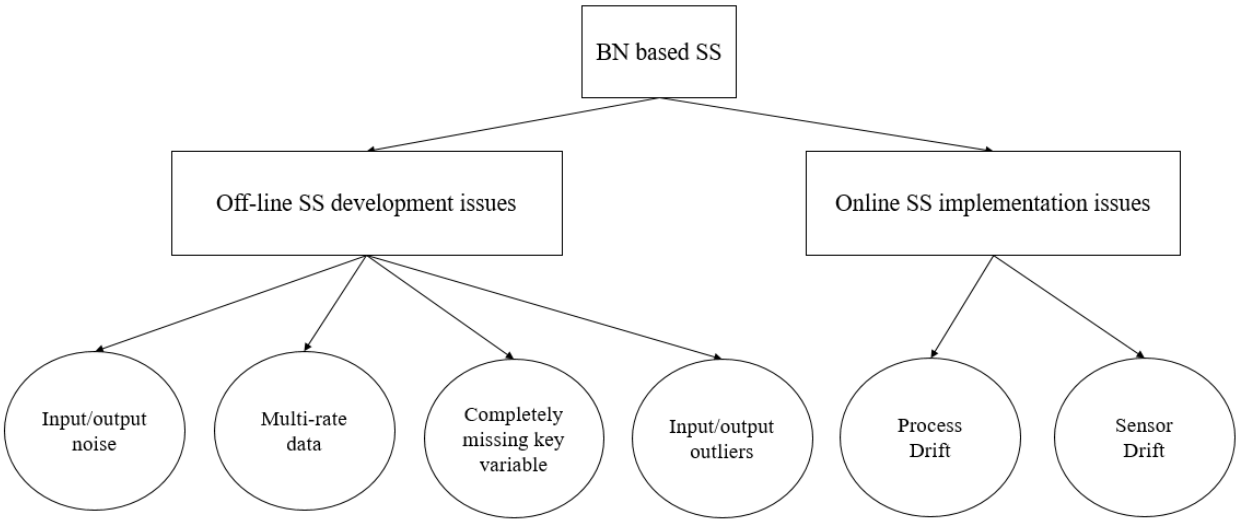

Figure 1. 2: Flow-chart of contributions

1.3 Thesis outline

The remainder of this thesis is organized as follows:

Chapter 2 discusses preliminaries and introduction to BNs. It outlines the systematic orders of BN based soft sensor development and includes two different formulations for down-sampled (DS) and multi-rate (MR) data. The proposed approach in this chapter can handle noisy, multi-rate and completely missing measurements without significantly deteriorating the soft sensor output prediction performance. Efficacy of the proposed approach was tested on simulation and industrial process data and compared to the popular OLS and PLS soft sensors.

Chapter 3 extends above formulation to addressing outliers and develops robust multi-rate Bayesian network soft sensor. Two separate formulations are included in this chapter, one for DS data and another for MR data, both corrupted with outliers. The performance of the proposed approach was tested on the same simulation and industrial case study discussed in the

previous chapter and different soft sensor performances were compared to the robust ordinary least squares (ROLS).

Chapter 4 proposes adaptive MR-BN-SS. This chapter tackles the issue of drift and provides two different formulations to address process and sensor drift separately. Effectiveness of the proposed approach is validated on the same simulation problem and was compared to bias updated OLS soft sensor.

Chapter 5 draws final conclusion of the thesis and provides a list of future work directions.

Chapter 2

Bayesian Network Soft Sensor Development for Multi-rate and Noisy Data

2.1 Introduction to Bayesian networks and Bayesian network soft sensor development

Bayesian networks are probabilistic graphical models representing random variables and their conditional dependencies via directed acyclic graph. In BN structure, nodes represent random variables and arcs represent cause and effect relationship between random variables[29,30,31]. Figure 2. 1 shows a simple common effect BN structure (or v-structure). In this structure, co-parents or the direct causes are $\{Y_{2\ldots}Y_m\}$ nodes and the child is the Y_1 node. From soft sensor development point of view, one can relate $\{Y_{2\ldots}Y_m\}$ as input variables, which affect the quality variable Y_1. Chain and common cause structures shown in Figure 2. 2 and Figure 2. 3 respectively are other important structures that are used in constructing BN structure of a process[29]. For a process with m random process variables $\{Y_1....Y_m\}$, there could be at least one or more source/ parentless node. Let's assume $\boldsymbol{Y_c}$ is the source node and c is number of source nodes in a given BN structure. For example, for the common effect structure shown in Figure 2. 1, $\boldsymbol{Y_c} = [Y_2, \ldots, Y_m]$ and $c = (m-1)$. Through the property of conditional independence i.e. given the evidence of parents of Y_i, ($P_a(Y_i)$), Y_i is dependent only on its parents (i.e. $p(Y_i|\ P_a(Y_i))$ and is independent of its ancestors. Using chain rule of probability, the joint probability distribution can be compactly expressed as follows:

$$p(Y_1 \ldots Y_m) = p(Y_c) \prod_{i=1}^{m-c} p(Y_i \mid p_a(Y_i)) \qquad (2.\ 1)$$

where, $p(\boldsymbol{Y_c})$ is the prior probability of the source nodes. Details regarding different common structures and properties of Bayesian network can be found in (Koller & Friedman (2009)).

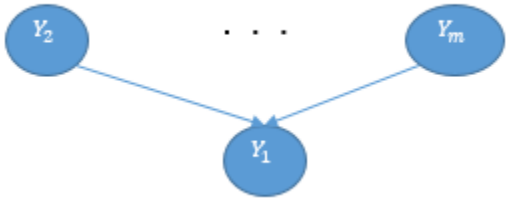

Figure 2. 1: Common effect structure

Figure 2. 2: Chain structure

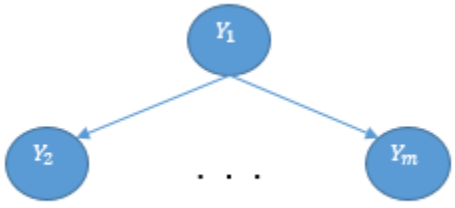

Figure 2. 3: Common cause structure

Initial step or building a Bayesian network structure is the most important yet challenging step for the development of BN based soft sensors. In BN structure, measured variables are treated as observed nodes and unmeasured variables/quality variables are considered as hidden variables of states. BN structure may be constructed through one of the following ways: (1) using process knowledge through process flow-sheets, (2) using historical process data and (3) by combination of process knowledge and process data[29,32,33]. In this work, the third approach is followed. Moreover, to estimate the noise-free state values, we introduce at least an equal number of hidden state nodes as the number of measurement nodes to the standard BN structure. This will be discussed in the following sections.

Once an optimal Bayesian network structure is constructed, the next critical step is to carry out parameter learning either through maximum-likelihood estimation or Bayesian inference. Due to the presence of hidden variables and missing data, expectation maximization (EM) algorithm[21,34,35,36] has to be used. The E-step, which involves computation of posterior probability of hidden variables, is usually computed through exact approaches, such as variable elimination and junction tree algorithms[37]. Although these approaches can obtain analytical solutions, the hidden nodes are assumed to be discrete and the run time of these inference algorithms is exponentially increasing, leading to NP hard problem[37,38,39]. In chemical plants, process variables are continuous in nature, so approximate inference approaches, such as Monte-Carlo sampling and Variational inference have been used to compute the posterior probability[37,40]. However, Monte-Carlo sampling is too computationally expensive for online prediction applications, while obtaining analytical solution is a difficult problem. Thus, in this work, Bayesian inference approach is used, which can provide analytical solutions.

Once parameters are estimated, the final step is output prediction. Given input data, posterior probability distribution of output is obtained through Bayesian inference. This approach also allows analytical solutions.

2.2 Modeling assumptions

Assumption 2. 1: The following measurement model is assumed between the measurement node (Y) and hidden state node (X) of j[th] variable:

$$Y_j = X_j + \varepsilon_j \tag{2.2}$$

where variable/node number is $j = [1,2,\dots,m]$ and $Y_j \in R^m$ is the measurement. $X_j \in R^m$ is hidden state corresponding to Y_j and follows Gaussian distribution. Also, ε_j is assumed to be a zero mean white noise signal and follows Gaussian distribution as:

$$\varepsilon_j \sim N(0, \sigma_{Y_j}^2) \tag{2.3}$$

Once the error is assumed to follow the Gaussian distribution, corresponding measurements will follow same distribution as shown below:

$$p(Y_j|X_j) \sim N(X_j, \sigma_{Y_j}^2) \tag{2.4}$$

Assumption 2. 2: In this formulation, there are at least same number of hidden states X as the number of the measurements Y. Between the hidden states themselves, the conditional distribution is assumed to follow a linear model as:

$$p\left(X_j \middle| Pa(X_j)\right) \sim N(\beta_{0,j} + \sum_{p=1}^{N_{Pa}} \beta_{0+p,j} Pa(X_j)_p, \sigma_{X_j}^2) \tag{2.5}$$

where, $Pa(X_j)$ includes all parent nodes of X_j and Npa is the number of parents.

Note that the measurement and state noise variance terms ($\sigma_{Y_j}^2$, $\sigma_{X_j}^2$) are unknown and need to be estimated along with the unknown model parameters between the hidden states, $\boldsymbol{\beta}_j = \left[\beta_{0,j},\left[\beta_{1,j} \ldots \beta_{N_{Pa},j}\right]\right]$. For any variable j, all unknown parameters can be represented as:

$$\Theta_j = \left[\boldsymbol{\beta}_j, \sigma_{Y_j}^2, \sigma_{X_j}^2\right] \tag{2.6}$$

2.3 Development of Bayesian network based soft sensor

Under these assumptions Bayesian network soft sensor is developed through the following three steps: (1) construction of Bayesian network structure, (2) parameter learning and (3) output prediction. The details will be discussed in the following sections.

2.3.1 Construction of Bayesian network structure

In this sub-section, Bayesian network structure is developed from the process flow diagram. To illustrate the procedure, a benchmark heat exchanger flow network problem[41] given in Figure 2. 4 is considered.

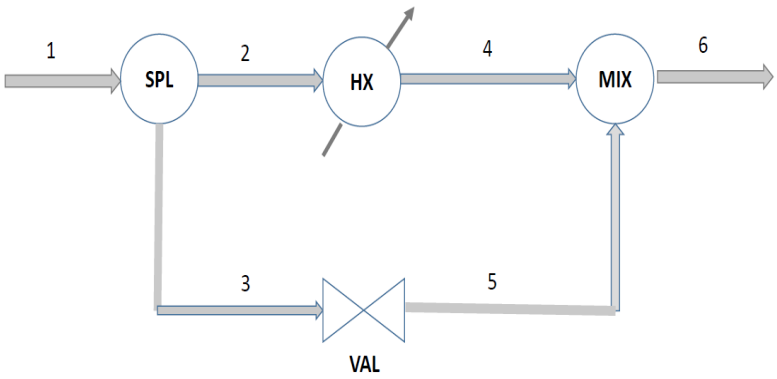

Figure 2. 4: Schematic representation of heat exchanger flow network

Here, the numbers 1 to 6 corresponds to the flow-variables, where flow-4 is the desired quality variable with slow-rate measurements. This system consists of 6 streams. The source flow, marked as flow-1, passes through a splitter (SPL) to become flow-2 and 3. Flow-2 further passes through a heat exchanger (HX) resulting in flow-4, while flow-3 passes through a valve (VAL) resulting in flow-5. Finally, flow-5 and flow-4 pass through mixer resulting in the final product flow-6. Considering this cause and effect relationship between the flow variables, Bayesian network structure given in Figure 2.5 can be developed.

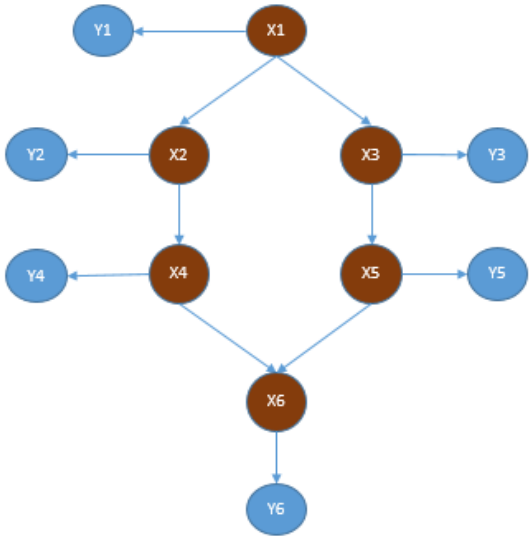

Figure 2.5: Bayesian network representation of flow-network

In Figure 2. 5, $\{Y_1 \ldots Y_6\}$ correspond to the measurements nodes and $\{X_1 \ldots X_6\}$ correspond to hidden states (noise-free). For any BN structure with m random process variables, by using the property of D-separation principle [29] and conditional independency between the nodes, the joint probability density function can be written as:

$$p(Y_1 \ldots Y_m, X_1 \ldots X_m) = p(X_c) \prod_{i=1}^{m} p(Y_i \mid X_i) \prod_{i=1}^{m-c} p(X_i \mid p_a(X_i)) \qquad (2.7)$$

where $\boldsymbol{X_c}$ is a variable without parent node, and c is the total number of source node. In this BN structure, given in Figure 2.5, $\boldsymbol{X_s} = X_1$ and $c = 1$.

2.3.2 Parameter learning

Consider a batch of N independent data points of m random process variables generated by perturbing the mean of the source node with certain variance, given by

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_{11} & .. & .. & .. & \mathbf{y}_{m1} \\ \mathbf{y}_{12} & .. & .. & .. & \mathbf{y}_{m2} \\ .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. \\ \mathbf{y}_{1N} & .. & .. & .. & \mathbf{y}_{mN} \end{bmatrix}_{N \times m} \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}_{11} & .. & .. & .. & \mathbf{x}_{m1} \\ \mathbf{x}_{12} & .. & .. & .. & \mathbf{x}_{m2} \\ .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. \\ \mathbf{x}_{1N} & .. & .. & .. & \mathbf{x}_{mN} \end{bmatrix}_{N \times m}$$

where each column in the data matrix **Y** (measurements) and **X** (hidden states) represents N independent samples of a random node. Each rows represents a sample of random nodes $[1, \ldots, m]$ at a particular sampling instant. Further, for simplicity **Y** and **X** can be written as **Y** = $\{Y_1 \ldots . Y_m\}$ and **X** = $\{X_1 \ldots . X_m\}$. In the absence of hidden variables, parameter estimation problem for any general Bayesian network structure (i.e. given in Figure 2. 1) can be posed as maximizing the log-likelihood function w.r.t unknown parameters i.e.

$$\Theta^* = arg \max_{\Theta} log\, p(\boldsymbol{Y}|\Theta) \qquad (2.8)$$

However, in the presence of hidden nodes, such as the Bayesian network structure given in Figure 2.5, unknown parameters (Θ) are obtained by maximizing the joint log-likelihood as shown below.

$$\max_{\Theta} log p(\boldsymbol{X},\boldsymbol{Y}) = \max_{\Theta}\left[\sum_{j=1}^{m}\log p\left(Y_j|X_j,\Theta\right)+\sum_{j=1}^{m-c}\log p\left(X_j|Pa\left(X_j\right),\Theta\right)+\log p(\boldsymbol{X_c}|I)\right] \quad (2.\ 9)$$

where $p(\boldsymbol{X_c}|I)$ refers to prior information of the source nodes $\boldsymbol{X_c}$. Since the joint density function given in Equation (2. 9) contains hidden nodes, direct numerical maximization of the above joint density may result in sub-optimal solutions, and explicit solution for the parameters is intractable. Thus, here we resort to expectation maximization (EM) approach[21].

2.3.3 Parameter learning for down-sampled data

EM algorithm is a popular iterative approach for obtaining maximum likelihood estimates of parameters in the presence of missing data or hidden states. In EM approach, unknown parameters are estimated by maximizing the lower bound of the likelihood function, which is known as the Q function [21]. EM algorithm consists of two steps, Expectation-step (E-step) and Maximization-step (M-step), which will be explained in detail in the subsequent sections.

E-step:

Given the observations (**Y**) and parameters (Θ_r) at the r^{th} iteration, expectation of the complete log-likelihood function w.r.t all the hidden variables **X** is calculated (as Q function).

$$Q(\Theta|\ \Theta_r) = E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}[\log p(\boldsymbol{Y},\boldsymbol{X}|\Theta)] \quad (2.\ 10)$$

Using the property of D-separation principle and conditional dependencies (or independencies) among the m random variables, the Q function given in Equation (2. 10) can be expressed as follows:

$$\mathrm{Q}(\Theta|\Theta_r) = \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\sum_{j=1}^{m}\log p\left(Y_j|X_j,\Theta\right)+\sum_{j=1}^{m-c}\log p\left(X_j|Pa\left(X_j\right),\Theta\right)+\log p(\boldsymbol{X_c}|I)\right] \quad (2.\ 11)$$

Assumption 2. 3: The prior distribution for the source node, $p(\boldsymbol{X_c}|I)$, can be utilized if the information is available. However, in the absence of prior knowledge, it can be assumed to follow uniform distribution and this assumption will be adopted for the rest of the derivations. Under Assumption 2. 3, Equation (2. 11) can be approximated as follows:

$$\mathrm{Q}(\Theta|\Theta_r) = \mathrm{E}_{X|Y,\Theta_r}\left[\sum_{j=1}^{m} \log p\left(Y_j|X_j,\Theta\right) + \sum_{j=1}^{m-c} \log p\left(X_j|Pa\left(X_j\right),\Theta\right)\right] \tag{2. 12}$$

For a batch of data with size *N*, and under Assumption 2. 1 that the measurements noises are mutually independent, Equation (2. 12) can be further expressed as:

$$\mathrm{Q}(\Theta|\Theta_r) = \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\sum_{j=1}^{m}\sum_{i=1}^{N} \log p\left(y_{i,j}|x_{i,j},\Theta\right) + \sum_{j=1}^{m-c}\sum_{i=1}^{N} \log p\left(x_{i,j}|Pa\left(x_{i,j}\right),\Theta\right)\right] \tag{2. 13}$$

From modeling Assumption 2. 1 & Assumption 2. 2, the expressions for $p\left(y_{i,j}|x_{i,j},\Theta\right)$ and $p\left(x_{i,j}|Pa\left(x_{i,j}\right),\Theta\right)$ of any i^{th} sample and j^{th} variable are written as

$$p\left(y_{i,j}|x_{i,j},\Theta\right) = \frac{1}{\sqrt{2\pi\sigma_{y_j}^2}} exp\left(-\frac{\left(y_{ij}-x_{ij}\right)^2}{2\sigma_{y_j}^2}\right) \tag{2. 14}$$

$$p\left(x_{i,j}|Pa\left(x_{i,j}\right),\Theta\right) = \frac{1}{\sqrt{2\pi\sigma_{x_j}^2}} \exp\left(-\frac{\left(x_{ij}-\beta_{0,j}-\sum_{p=1}^{N_{Pa}} Pa(x_{ij})_p\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2}\right) \tag{2. 15}$$

Using the terms shown in Equations (2. 14) & (2. 15) and Equation (2. 13) results in the following equation.

$$\mathrm{Q}(\Theta|\Theta_r) = \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\sum_{j=1}^{m}\sum_{i=1}^{N}\left[\frac{1}{\sqrt{2\pi}}\log(\sigma_{y_j}^2) - \frac{(y_{ij}-x_{ij})^2}{2\sigma_{y_j}^2}\right] + \sum_{j=1}^{m-c}\sum_{i=1}^{N}\left[\frac{1}{\sqrt{2\pi}}\log(\sigma_{x_j}^2) - \frac{\left(x_{ij}-\beta_{0,j}-\sum_{p=1}^{NPa} Pa(x_{ij})_p\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2}\right]\right] \quad (2.\ 16)$$

Using the linear property of Expectation, this equation can be further expressed as:

$$\mathrm{Q}(\Theta|\Theta_r) = \left[\sum_{j=1}^{m}\sum_{i=1}^{N}\left[\frac{1}{\sqrt{2\pi}}\log(\sigma_{y_j}^2) - \frac{\left(y_{ij}^2+\mathrm{E}_{\mathrm{X}|\mathrm{Y},\Theta_r}\left[x_{ij}^2\right]-2y_{ij}\,\mathrm{E}_{\mathrm{X}|\mathrm{Y},\Theta_r}\left[x_{ij}\right]\right)}{2\sigma_{y_j}^2}\right] + \sum_{j=1}^{m-c}\sum_{i=1}^{N}\left[\frac{1}{\sqrt{2\pi}}\log(\sigma_{x_j}^2) - \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\frac{\left(x_{ij}-\beta_{0,j}-\sum_{p=1}^{NPa} Pa(x_{ij})_p\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2}\right]\right]\right] \quad (2.\ 17)$$

Thus, from Equation (2. 17), it can be observed that the following statistics are required in order to evaluate the Q function.

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}^2\right],\ \ \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}\right],\ \ \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}Pa(x_{ij})\right],\ \ \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\,Pa(x_{ij})_{N_{pa}}\,Pa_{N_{pa+1}}(x_{ij})\right] \quad (2.\ 18)$$

where these statistics can be further expanded as:

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}^2\right] = var\left(x_{ij}\right) + \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}\right]\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_k}\left[x_{ij}\right] \quad (2.\ 19)$$

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}\,Pa(x_{ij})\right] = cov\left(x_{ij}, Pa(x_{ij})\right) + \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[x_{ij}\right]\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[Pa(x_{ij})\right] \quad (2.\ 20)$$

For a node with multiple parents, additional statistics concerning the relationship between the parent nodes will be necessary. Those statistics can be expanded as:

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[Pa(x_{ij})_{N_{pa}} Pa(x_{ij})_{N_{pa+1}}\right] = cov\left(Pa(x_{ij})_{N_{pa}}, Pa(x_{ij})_{N_{pa+1}}\right) + \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[Pa(x_{ij})_{N_{pa}}\right] \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[Pa(x_{ij})_{N_{pa+1}}\right] \quad (2.\ 21)$$

Thus, from Equations (2. 19) - (2. 21), it is evident that the following statistics will be required.

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}[x_{ij}],\ var\left(x_{ij}\right), cov\left(x_{ij}, Pa(x_{ij})\right),\ \mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[Pa(x_{ij})\right], cov\left(Pa(x_{ij})_{Npa}, Pa(x_{ij})_{Npa+1}\right) \quad (2.\ 22)$$

These statistics are obtained by evaluating the posterior distribution of the hidden variables, and the variance and covariance terms are obtained from the covariance matrix, which is explained in the following sections.

Computing the statistics from posterior distribution

A full posterior distribution via Bayesian rule can be expressed as:

$$p(\boldsymbol{X}|\boldsymbol{Y}, \Theta_{old}) = \frac{p(\boldsymbol{X}, \boldsymbol{Y}|\Theta_{old})}{p(\boldsymbol{Y})} \quad (2.\ 23)$$

Thus, for a batch of data, posterior distribution of the hidden state $\boldsymbol{X}$ is obtained from computation of the below formulation, where γ is a normalizing constant that equals $p(\boldsymbol{Y})^{-1}$.

$$p(\boldsymbol{X}|\boldsymbol{Y}, \Theta_r) = \gamma\left[\prod_{j=1}^{m} p\left(Y_j|X_j, \Theta_r\right) \prod_{j=1}^{m-c} p\left(X_j|Pa\left(X_j\right), \Theta_r\right) p(\boldsymbol{X_c}|I)\right] \quad (2.\ 24)$$

Maximizing logarithmic of posterior distribution function w.r.t each hidden state will result in set of simultaneous equations, from which mode of the hidden states can be computed as:

$$\begin{aligned}\widehat{\boldsymbol{X}} &= \max_{\boldsymbol{X}} \log p(\boldsymbol{X}|\boldsymbol{Y}, \Theta_r) \\ &= \max_{X} \left[\sum_{j=1}^{m} \log p\left(Y_j \middle| X_j, \Theta_r\right) + \sum_{j=1}^{m-c} \log p\left(X_j \middle| Pa\left(X_j\right), \Theta_r\right) + \log p(\boldsymbol{X_c}|I) \right. \\ &\quad \left. + \log(\gamma) \right]\end{aligned} \tag{2. 25}$$

Based on the Assumption 2. 1-Assumption 2. 3, Equation (2. 25) can be expressed as follows:

$$\begin{aligned}&\max_{\boldsymbol{X}} \log p(\boldsymbol{X}|\boldsymbol{Y}, \Theta_r) \\ &= \max_{X} \left[\sum_{i=1}^{N} \sum_{j=1}^{m} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma_{y_{j,r}}^2) - \frac{\left(y_{ij} - x_{ij}\right)^2}{2\sigma_{y_{j,r}}^2} \right] \right. \\ &\quad + \sum_{i=1}^{N} \sum_{j=1}^{m-c} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma_{x_{j,r}}^2) - \frac{\left(x_{ij} - \beta_{0,j,r} - \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij}) \beta_{0+p,j,r}\right)^2}{2\sigma_{x_{j,r}}^2} \right] \\ &\quad \left. + \log(\gamma) \right]\end{aligned} \tag{2. 26}$$

Considering the first order optimality conditions for Equation (2. 26) i.e. $\frac{\partial\, logP\,(X|Y,\Theta_r)}{\partial X} = 0$, for any i^{th} sample and j^{th} variable, the estimates of hidden states can be represented in analytical form as given by Equations (2. 27) - (2. *28*). Here N_{ch} stands for the number of children that the

target node (x_j) has in the Bayesian network structure and $Ch(x_j)$ indicates the child node of the variable x_j.

$$\alpha\left(x_j\right) = \left(\frac{1}{\sigma^2_{y_{j,r}}} + \frac{1}{\sigma^2_{x_j|Pa(x_j),r}} + \sum_{c=1}^{N_{ch}} \frac{\left(\beta^2_{c,j,r}\right)}{\sigma^2_{Ch_c(x_j)|x_{j,r}}}\right) \tag{2. 27}$$

$$\hat{x}_{ij} = \frac{\frac{y_{ij}}{\sigma^2_{y_{j,r}}} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j,r}\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{Ch(x_j)|x_{j,k,r}}} + \frac{\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{x_j|Pa(x_j),r}}}{\alpha(x_j)} \tag{2. 28}$$

For $j = 1 \ldots m$ variables, this will result in m simultaneous linear equations, which can be solved by using any linear equation solver.

By Gaussian distribution property and from linear relation among the variables, the covariance matrix of the hidden states, $X = [X_1, \ldots X_m]$, can be computed by rearranging the posterior probability distribution Equation i.e. Equation (2. 24) as:

$$\boldsymbol{p}(\boldsymbol{X}|\boldsymbol{Y}) \propto exp(-\tfrac{1}{2}(X - \hat{X})^T S(X - \hat{X})) \tag{2. 29}$$

where, $\hat{X}$ corresponds to the estimated state vector for i^{th} sampling instant whose elements are obtained from Equation (2. 29), and D is the inverse covariance matrix of the hidden states. For any j^{th} variable, the diagonal elements of the D matrix are obtained as:

$$D_{jj} = \left(\frac{1}{\sigma^2_{y_{j,r}}} + \frac{1}{\sigma^2_{x_j|Pa(x_j),r}} + \sum_{c=1}^{N_{ch}} \frac{\left(\beta^2_{c,j,r}\right)}{\sigma^2_{Ch_c(x_j)|x_{j,r}}}\right) \tag{2. 30}$$

and the off-diagonal elements, which are interaction between the j^{th} node and its child or parent nodes, can be expressed as:

$$D_{zj} = \frac{\beta_{z,j,r}}{\sigma^2_{Ch_c(x_j)|x_{j,r}}} \; if \; z = child \; node \; (z \in 1 \dots N_{ch}) \tag{2. 31}$$

$$D_{zj} = \frac{\beta_{z,j,r}}{\sigma^2_{(x_j)|pa(x_{j,r})}} \; if \; z = parent \; node \; \left(z \in 1 \dots N_{pa}\right) \tag{2. 32}$$

where $$D_{zj} = 0 \;\; if \; z = neither \; chid \; nor \; parent \tag{2. 33}$$

Thus, for i^{th} sampling instant, all the elements of D matrix are obtained from the above Equations. For a batch of data N, the D matrix results in a block-diagonal matrix $\boldsymbol{D}$. The covariance matrix of the hidden states (i.e. $\Sigma_x = D^{-1}$) for i^{th} sample or for entire batch of data (N) can be obtained by computing the inverse of D matrix or full block diagonal matrix ($\boldsymbol{D}$), respectively.

M-step:

In the M-step, Q function is maximized w.r.t all the parameters Θ_r as shown in Equation (2. 34) as:

$$\Theta_{r+1} = arg \; \max_{\Theta} Q[\Theta|\Theta_r] \tag{2. 34}$$

In the context of this problem, maximization can be expanded as:

$$\frac{\partial Q}{\partial \sigma^2_{y_j}} = 0, \qquad \frac{\partial Q}{\partial \sigma^2_{x_j}} = 0, \qquad \frac{\partial Q}{\partial \beta_j} = 0 \tag{2. 35}$$

Closed form solutions for all the parameters can be derived as follows:

$$\sigma^2_{y_j,r+1} = \frac{E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N}\left(y_{ji} - x_{ji}\right)^2\right)}{N} \tag{2. 36}$$

$$\sigma_{x_j,r+1}^2 = \frac{E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N}\left(x_{ji} - \beta_{0j} - \sum_{p=1}^{N_{Pa}} \beta_{0+p,j}\, Pa_p(x_{ij})\right)^2\right)}{N} \tag{2. 37}$$

$$\beta_{0,j,r+1} = \frac{E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N}\left(x_{ij} - \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij})\beta_{0+p,j}\right)\right)}{N} \tag{2. 38}$$

$$\beta_{0j+p,r+1}$$

$$= \frac{E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N} Pa_p(x_{ij})\right) E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N}\left(x_{ij} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{ij})\beta_{0+p,j}\right)\right)}{E_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left(\sum_{i=1}^{N} Pa_p(x_{ij})\right)^2} \tag{2. 39}$$

2.3.4 Parameter learning for multi-rate/missing data

Until now the derivation of parameter learning has assumed that all measurements have the same sampling rate and are fully available. In reality, quality variables that are typically obtained through lab analysis are available only at a slow-rate, resulting the multi-rate problem. Further, due to possible sensor problems some key variables can be completely missing. Thus, this section formulates parameter learning of data with missing variables.

Assumption 2. 4: To account for missing data in parameter learning, the batch of measurement data ($\boldsymbol{Y} \in R^{m\times N}$) and corresponding hidden states ($\boldsymbol{X} \in R^{m\times N}$) of m variables with N samples can be expressed as:

$$\boldsymbol{Y} = [\boldsymbol{Y}_{obs}\, \boldsymbol{Y}_{mis}], \qquad \boldsymbol{X} = [\boldsymbol{X}_{obs}\, \boldsymbol{X}_{mis}] \tag{2. 40}$$

where, $Y_{obs,} \in R^{m \times O}$ is the available measurements in the training data set, $Y_{mis}, \in R^{m \times M}$ refers to the missing measurements, X_{obs} corresponds to the hidden states and X_{mis} corresponds to the hidden states of the corresponding measurements. O, M refer to the total number of observed and missing data samples in the training data set, respectively. Thus, based on Assumption 2. 4, the unknown parameters can be obtained by maximizing the logarithm of complete likelihood function given as:

$$\max_{\Theta} \log p(\boldsymbol{Y}_{obs}, \boldsymbol{Y}_{mis}, \boldsymbol{X}|\Theta) \tag{2. 41}$$

Using the property of independency of measurements and from Bayesian network principles, the joint likelihood function can be factored as:

$$p(\boldsymbol{Y}_{obs}, \boldsymbol{Y}_{mis}, \boldsymbol{X}|\Theta) \tag{2. 42}$$
$$= \prod_{j=1}^{m} p\left(Y_{obs,j}|X_{obs,j}, \Theta\right) \prod_{j=1}^{m} p\left(Y_{mis,j}|X_{mis,j}, \Theta\right) \prod_{j=1}^{m-c} p\left(X_j|Pa\left(X_j\right), \Theta\right) p(X_c|I)$$

where, $p(Y_{obs}|X_{obs}, \Theta), P(Y_{mis}|X_{mis}, \Theta)$ and $p(X|Pa(X), \Theta)$ follow Gaussian distribution from Assumption 2. 1 - Assumption 2. 3, joint log-likelihood function can be written as:

$$\log p(\boldsymbol{Y}_{obs}, \boldsymbol{Y}_{mis}, \boldsymbol{X}|, \Theta)$$
$$= \sum_{j=1}^{m} \log p\left(Y_{obs,j}|X_{obs,j}, \Theta\right) + \sum_{j=1}^{m} log p\left(Y_{mis,j}|X_{mis,j}, \Theta\right) \tag{2. 43}$$
$$+ \sum_{j=1}^{m-c} \log p\left(X_j|Pa\left(X_j\right), \Theta\right)$$

Expectation Maximization (EM) Algorithm

E-step:

Expectation of the complete log-likelihood function w.r.t. all the missing data ($\boldsymbol{Y}_{mis}$) and hidden variables ($\boldsymbol{X}$) is evaluated as:

$$Q(\Theta|\ \Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}[\log p(\boldsymbol{Y}_{obs},\boldsymbol{Y}_{mis},\boldsymbol{X}|\Theta)] \tag{2. 44}$$

Using the property of independency of measurement and conditional dependency of the nodes, the Q function can be expressed as:

$$Q(\Theta|\Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[\sum_{j=1}^{m}\log p\left(Y_{obs,j}\middle|X_{obs,j},\Theta\right) + \sum_{j=1}^{m} logp\left(Y_{mis,j}\middle|X_{mis,j},\Theta\right) + \sum_{j=1}^{m-c}\log p\left(X_j\middle|Pa\left(X_j\right),\Theta\right)\right] \tag{2. 45}$$

As per Assumption 2.3, prior term is assumed to be uniform. For ease of presentation, above equation can be separated as:

$$Q(\Theta|\Theta_r) = Q_1(\Theta|\ \Theta_r) + Q_2(\Theta|\ \Theta_r) \tag{2. 46}$$

where

$$Q_1(\Theta|\Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\sum_{j=1}^{m}\log p\left(Y_{obs,j}\middle|X_{obs,j},\Theta\right) + \sum_{j=1}^{m} logp\left(Y_{mis,j}\middle|X_{mis,j},\Theta\right)\right] \tag{2. 47}$$

$$Q_2(\Theta|\Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\sum_{j=1}^{m-c}\log p\left(X_j\middle|Pa\left(X_j\right),\Theta\right)\right] \tag{2. 48}$$

For a batch of data with size N, $Q_1(\Theta|\Theta_r)$ from Equation (2. 47) is evaluated as follows.

$$Q_1(\Theta|\Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\log\prod_{i=1}^{O}\prod_{j=1}^{m} p(y_{obs,ij}|x_{obs,ij},\Theta)\right] + E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\log\prod_{i=O+1}^{N}\prod_{j=1}^{m} p(y_{mis,ij}|x_{mis,ij},\Theta)\right] \quad (2.\ 49)$$

Based on the Assumption 2. 1 and Assumption 2. 2 that conditional distributions follow Gaussian distribution, Equation (2. 49) can be further expressed as:

$$Q_1(\Theta|\Theta_r) = E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\sum_{i=1}^{O}\sum_{j=1}^{m}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_j^2}}\right) - \frac{\left(y_{obs,ij}-x_{obs,ij}\right)^2}{2\sigma_j^2}\right]\right] + E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|Y_{obs},\Theta_r}\left[\sum_{i=O+1}^{N}\sum_{j=1}^{m}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_j^2}}\right) - \frac{\left(y_{mis,ij}-x_{mis,ij}\right)^2}{2\sigma_j^2}\right]\right] \quad (2.\ 50)$$

For evaluating $Q_1(\Theta|\Theta_r)$ function, in addition to the required statistics of hidden variables corresponding to the observed measurements i.e. the following statistics related to the missing measurements and their corresponding hidden states are required.

$$E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[x_{obs,ij}^2\right],\ E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[x_{obs,ij}\right] \quad (2.\ 51)$$

$$\mathrm{E}_{\mathbf{X},\mathbf{Y}_{\mathrm{mis}}|\mathbf{Y}_{\mathrm{obs}},\Theta_r}\left[\mathrm{y}_{\mathrm{mis,ij}}^2\right], \mathrm{E}_{\mathbf{X},\mathbf{Y}_{\mathrm{mis}}|\mathbf{Y}_{\mathrm{obs}},\Theta_r}\left[\mathrm{x}_{\mathrm{mis,ij}}^2\right], \mathrm{E}_{\mathbf{X},\mathbf{Y}_{\mathrm{mis}}|\mathbf{Y}_{\mathrm{obs}},\Theta_r}\left[\mathrm{x}_{\mathrm{mis,ij}}\,\mathrm{y}_{\mathrm{mis,ij}}\right] \quad (2.\ 52)$$

where these statistics can be further expanded as follows:

$$E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[x_{mis,ij}y_{mis,ij}\right] = cov\left(x_{mis,ij},y_{mis,ij}\right) + \quad (2.\ 53)$$

$$E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[x_{mis,ij}\right] E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r} E\left[\, y_{mis,ij}\right]$$

$$E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[y_{mis,ij}^2\right] = var\left(y_{mis,ij}\right) + \; E_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left[y_{mis,ij}\right]^2 \tag{2. 54}$$

It is noted that, evaluating $Q_2(\Theta|\Theta_r)$ function from Equation (2. 48), is similar to evaluation of the following term in Equation (2. 17) i.e.

$$\mathrm{E}_{\boldsymbol{X}|\boldsymbol{Y},\Theta_r}\left[\frac{\left(x_{ij} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}} Pa_p\left(x_{ij}\right)\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2}\right] \tag{2. 55}$$

and identical statistics illustrated in Equation (2. 22) are required. Therefore, all the statistics will be derived in the following section.

Computing the statistics from posterior distribution

From Bayes' rule, the full posterior probability distribution is:

$$p(\boldsymbol{X}, \boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs}, \Theta_r) = \frac{p(\boldsymbol{Y}_{obs}|\boldsymbol{X}_{obs}, \Theta_r)p(\boldsymbol{X}, \boldsymbol{Y}_{mis}|I)}{p(\boldsymbol{Y}_{obs})} \tag{2. 56}$$

Since in the above Equation, the denominator acts as normalizing constant, its inverse is marked as γ and the posterior distribution can be written as:

$$p(\boldsymbol{X}, \boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs}, \Theta_r) = \gamma p(\boldsymbol{Y}_{obs}|\boldsymbol{X}_{obs}, \Theta_r)p(\boldsymbol{X}, \boldsymbol{Y}_{mis}|I) \tag{2. 57}$$

By using the property of conditional dependence between the nodes, above equation can be further decomposed as:

$$p(\boldsymbol{X}, \boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs}, \Theta_r) = \gamma p(\boldsymbol{Y}_{obs}|\boldsymbol{X}_{obs}, \Theta_r) p(\boldsymbol{Y}_{mis}|\boldsymbol{X}_{mis}, \Theta_r) p(\boldsymbol{X}|Pa(\boldsymbol{X}), \Theta_r) p(Pa(\boldsymbol{X})|I) \quad (2.58)$$

It is to be noted that in the above formulation, $p(Pa(\boldsymbol{X})|I)$ is the prior information of the source node/parentless nodes. Thus, given the observed measurements (Y_{obs}) and parameter vector (Θ_r), estimates of hidden states (X) and missing measurements (Y_{mis}) are obtained by maximizing the log of posterior distribution function w.r.t. these variable as:

$$\begin{aligned} \max_{\boldsymbol{X}, \boldsymbol{Y}_{mis}} & \; logp(\boldsymbol{X}, \boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs}, \Theta_r) \\ &= \max_{\boldsymbol{X}, \boldsymbol{Y}_{mis}} [logp(\boldsymbol{Y}_{obs}|\boldsymbol{X}_{obs}, \Theta_r) + logp(\boldsymbol{Y}_{mis}|\boldsymbol{X}_{mis}, \Theta_r) \\ &+ logp(\boldsymbol{X}|Pa(\boldsymbol{X}), \Theta_r) + logp(Pa(\boldsymbol{X})|I) + \log(\gamma)] \end{aligned} \quad (2.59)$$

Based on all the assumptions made, Equation (2. 59) can be expressed as follows:

$$\begin{aligned} \max_{\boldsymbol{X}, \boldsymbol{Y}_{mis}} & \; logp(\boldsymbol{X}, \boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs}, \Theta_r) \\ &= \max_{\boldsymbol{X}, \boldsymbol{Y}_{mis}} \left[\sum_{i=1}^{O} \sum_{j=1}^{m} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma_{y_j,r}^2) - \frac{\left(y_{obs,ij} - x_{obs,ij}\right)^2}{2\sigma_{y_j,r}^2} \right] \right. \\ &+ \sum_{i=O+1}^{N} \sum_{j=1}^{m} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma_{y_j,r}^2) - \frac{\left(y_{mis,ij} - x_{mis,ij}\right)^2}{2\sigma_{y_j,r}^2} \right] \\ &\left. + \sum_{i=1}^{N} \sum_{j=1}^{m-c} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma_{x_j,r}^2) - \frac{\left(x_{ij} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij}) \beta_{0+p,j}\right)^2}{2\sigma_{x_j,r}^2} \right] \right] \end{aligned} \quad (2.60)$$

Considering the first order optimality conditions for Equation (2. 60) (equating first order derivatives w.r.t. hidden states and missing measurement to zero), for any i^{th} sample and j^{th}

variable, the estimates of hidden states and missing measurements can be derived in analytical form as follows, where the denominator term is given in Equation (2. 27).

$$\hat{x}_{ij} = \frac{\frac{y_{obs\,ij}}{\sigma^2_{y_j,r}} + \sum_{c=1}^{C} \frac{\beta_{c,j,r}\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{Ch(x_j)|x_j,r}} + \frac{\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{x_j|Pa(x_j),r}}}{\alpha(x_j)} \tag{2. 61}$$

At the instants where the slow-rate samples, $\hat{y}_{mis,ij}$, are not available, by using the posterior probability given in Equation (2. 62), the hidden state value of the corresponding measurement $\hat{y}_{mis,ij}$ can be expressed as in Equation (2. 63), where the denominator for the j^{th} missing variable is given as Equation (2. 64).

$$\hat{y}_{mis,ij} = \hat{x}_{ij} \tag{2. 62}$$

$$\hat{x}_{ij} = \frac{\sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j,r}\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{Ch(x_j)|x_j,r}} + \frac{\left(\beta_{0,j,r} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij})\beta_{0+p,j,r}\right)}{\sigma^2_{x_j|Pa(x_j),r}}}{\alpha_{mis}(x_j)} \tag{2. 63}$$

$$\alpha_{mis}(x_j) = \left(\frac{1}{\sigma^2_{x_j|Pa(x_j),r}} + \frac{1}{\sigma^2_{x_{j-prior}}} + \sum_{c=1}^{N_{ch}} \frac{\left(\beta^2_{c,j}\right)}{\sigma^2_{Ch_c(x_j)|x_j}}\right) \tag{2. 64}$$

The covariance between the hidden states X can be computed as discussed in section 0, Equations (2. 30)-(2. 33).

For variables $j = 1 \ldots m$, Equations (2. *61*)-(2. *63*) result in a set of m simultaneous linear equations, which can be solved simultaneously by any linear solver.

M-step:

In the M-step, Q function is maximized w.r.t all the parameters Θ as shown in Equation (2. 34). Solving the updated measurement noise variance term with multi-rate data will result in the following parameter update equation.

$$\sigma^2_{y_j,r+1} = \frac{\mathrm{E}_{\boldsymbol{X},\boldsymbol{Y}_{mis}|\boldsymbol{Y}_{obs},\Theta_r}\left(\sum_{i=1}^{O}\left(y_{obs,ji} - x_{ji}\right)^2 + \sum_{i=O+1}^{N}\left(y_{mis,ji} - x_{ji}\right)^2\right)}{N} \tag{2. 65}$$

The rest of the model parameters are updated through Equations (2. 36) - (2. 39).

Convergence Check

Once E-step and M-step of the algorithm are completed, newly estimated parameter set is checked for the convergence i.e.

$$\Theta_{i+1} - \Theta_i \leq tolerance \tag{2. 66}$$

and the EM algorithm will iterate until the tolerance is reached.

2.4 Inference in Bayesian networks

Once the parameter learning is carried out, and model and variance parameters are estimated, the next step is to predict output using the newly available measurements. Here, Bayesian inference is utilized to predict the corresponding hidden state node of the quality variable, e.g. X_4 in the illustrative example.

Considering the k^{th} sampling instant, measurements of all the input variables are available as $\boldsymbol{y}_k = [y_{1,k} \quad \cdots \quad y_{m-1,k}]^T$, only the measurements of quality variable $y_{j=f,k}$ ($where\, j \in$

$1 \ldots . m$) are unavailable, and the hidden state vector of all the process variables to be inferred is given as $\boldsymbol{x}_k = [x_{1,k} \quad \cdots \quad x_{m,k}]^T$.

From Bayes' theorem given in Equation (2. 23), the posterior probability of the hidden states is the following, where γ is a normalizing constant that equals $P(\boldsymbol{y_k})^{-1}$.

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) = \gamma p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{I}) \tag{2. 67}$$

Using the property of independency of measurements and conditional dependency of the nodes, the posterior probability distribution function can be further decomposed, resulting in

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) = \gamma \prod_{j=1, j\neq f}^{m-1} p(y_{j,k}|x_{j,k}) \prod_{j=1}^{m-c} p\left(x_{j,k}\middle|Pa(x_{j,k})\right) p(x_{c,k}|I) \tag{2. 68}$$

where $p(x_{c,k}|I)$ is prior probability of the source node x_c at the k^{th} instant. Thus, point estimates of the hidden state variables at the k^{th} sampling instant can be obtained by minimizing the negative logarithmic of posterior distribution function i.e.

$$\begin{aligned} \hat{\boldsymbol{x}}_k &= \min_{\boldsymbol{x}_k} -[\log p(\boldsymbol{x}_k|\boldsymbol{y}_k)] \\ &= \min_{\boldsymbol{x}_k} -\left[\sum_{j=1, j\neq f}^{m-1} \log p(y_{j,k}|x_{j,k}) + \sum_{j=1}^{m-c} \log p\left(x_{j,k}\middle|Pa(x_{j,k})\right)\right. \\ &\quad \left. + log\, p(x_{c,k}|I) + \log(\gamma)\right] \end{aligned} \tag{2. 69}$$

Since the prior probability distribution of the source node i.e. $p(x_{c,k}|I)$ is assumed to follow uniform distribution, namely a constant, this term can be omitted without affecting the maximization operation. Thus, Equation (2. 69) is expanded to the following expression:

$$\min_{\boldsymbol{x}_k} -[\log p(\boldsymbol{x}_k|\boldsymbol{y}_k)]$$

$$= \min_{\boldsymbol{x}_k} -\left[\sum_{j=1,j\neq f}^{m-1} \frac{\left(y_{j,k}-x_{j,k}\right)^2}{2\sigma_{y_j}^2} + \sum_{j=1}^{m-c} \frac{\left(x_{j,k}-\beta_{0,j}-\sum_{p=1}^{N_{Pa}} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2} + \log(\gamma)\right] \tag{2. 70}$$

Maximizing the objective function w.r.t. each hidden states gives a set of simultaneous linear Equations as:

$$f(\hat{\boldsymbol{x}}_k, \Theta) = B \tag{2. 71}$$

which can be transformed into $A\hat{\boldsymbol{x}}_k = B$, and the estimates of all the hidden states are obtained by

$$\hat{\boldsymbol{x}}_k = A^{-1}B \tag{2. 72}$$

Thus, the analytical expression of estimate of hidden state variable j $(where, j \in 1 \ldots m)$ at k^{th} instant is given by Equation (2. 73) and Equation (2. 74), where the denominators are given in Equation (2. 27) and Equation (2. 64) respectively.

$$\hat{x}_{j\neq f,k} = \frac{\frac{y_{j,k}}{\sigma_{y_j}^2} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j}\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}-1} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)}{\sigma_{Ch(x_j)|x_j}^2} + \frac{\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)}{\sigma_{x_j|Pa(x_j)}^2}}{\alpha_{obs}\left(x_j\right)} \tag{2. 73}$$

$$\hat{x}_{j=f,k} = \frac{\sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j}\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}-1} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)}{\sigma^2_{Ch(x_j)|x_j}} + \frac{\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)}{\sigma^2_{x_j|Pa(x_j)}}}{\alpha_{mis}\left(x_j\right)} \quad (2.\ 74)$$

Thus, through solving above simultaneous linear equations, estimates of hidden states are obtained at every sampling instant.

2.5 Simulation and industrial application

The performance of the proposed approach is demonstrated on following benchmark simulation [41] example and an industrial case study. In both cases, we have analyzed following three scenarios:

i. Multi-rate and noisy lab data

ii. Noisy input

iii. Completely missing key input + noisy input

In simulation studies, to compare the efficacy of the proposed approach, average root mean squared error (ARMSE) is computed based on 10 realizations, with 100 data samples each. This equation for the j^{th} variable is:

$$(ARMSE)_j = \frac{1}{N_r}\sqrt{\frac{\sum_{i=1}^{N}\left(x_{i,j} - \hat{x}_{i,j}\right)^2}{N}} \quad (2.\ 75)$$

where N_r corresponds to the number of Monte-Carlo simulations and N represents number of samples. In the industrial case study, all the real-time predictions have to be down-sampled

according to the sampling rate of available lab data in order to compute RMSE and correlation coefficient since lab data are only available at slow-rate. For the j^{th} variable,

$$(RMSE)_j = \sqrt{\frac{\sum_{i=1}^{N}(y_{i,j} - \hat{x}_{i,j})^2}{N}} \tag{2. 76}$$

and correlation coefficient, $Corr$ (or Pearson's product-moment correlation coefficient), between the lab measurement and estimated state is calculated as:

$$(Corr)_j = \frac{\sum_{i=1}^{N}(y_{i,j} - \bar{y_j})(\hat{x}_{i,j} - \bar{\hat{x}_j})}{\sqrt{\sum_{i=1}^{N}(y_{i,j} - \bar{y_j})^2}\sqrt{\sum_{i=1}^{N}(\hat{x}_{i,j} - \bar{\hat{x}_j})^2}} \tag{2. 77}$$

where the arithmetic average of a measurement is:

$$\bar{y_j} = \frac{1}{N}\sum_{i=1}^{N} y_{ij} \tag{2. 78}$$

2.5.1 Flow Network

Schematic of the flow network system is given in Figure 2. 4 and process model Equations are given in Equation (2. 79). Corresponding Bayesian network structure is illustrated in Figure 2. 5.

$$\begin{aligned} X_2 &= split\ \ ratio * X_1 + \varepsilon_{x_2} \\ X_3 &= (1 - split\ \ ratio) * X_1 + \varepsilon_{x_3} \\ X_4 &= X_2 + \varepsilon_{x_4} \\ X_5 &= X_3 + \varepsilon_{x_5} \end{aligned} \tag{2. 79}$$

$$X_6 = X_4 + X_5 + \varepsilon_{x_6}$$

The uncertainty in the state ε_{x_i} is a white noise signal and follows Gaussian distribution i.e.

$$\varepsilon_{x_i} \sim N(0, \sigma_{x_i}) \quad (2.80)$$

where

$$\sigma_{x_i} = 0.01 * \sigma(X_{i,true}) \quad (2.81)$$

As discussed in section 2.4, inference was conducted for X_4, which is the flow-4 hidden state node. Corresponding measurement, Y_4, is available only at every 30 sampling instances. Steady state mean values of the process flow variables are given in Table 2. 1. For the multi-rate Bayesian network soft sensor (MR-BN-SS) scenario, 2500 data samples were used for training and 500 samples were used for validation. For the down-sampled Bayesian network soft sensor (DS-BN-SS), namely OLS and PLS approaches, previously mentioned 2500 data samples were down-sampled according to the slow sampling rate.

Process Variable	Mean values (m/s)
Y_1	100
Y_2	64
Y_3	36
Y_4	64
Y_5	36
Y_6	100

Table 2. 1: Steady state values of the process variables

Multi-rate and noisy lab data

In this simulation, Y_4 is corrupted with noise and is multi-rate. Under these conditions, the efficacy of the proposed down-sampled Bayesian network soft sensor (DS-BN-SS) and multi-rate Bayesian network soft sensor (MR-BN-SS) are compared to the popular OLS and PLS soft sensors. Figure 2. 6 is the graphical result. Average RMSE values for the target hidden state are reported in Table 2. 2. From this table, it can be observed that MR-BN-SS has the lowest ARMSE value. Moreover, the proposed DS-BN-SS also performs better than the conventional approaches, but not as good as MR-BN-SS. The reason is that MR-BN-SS is capable of using fast-rate samples, constituting a semi-supervised learning. To illustrate the goodness of BN based soft sensors, noise variance convergence profiles of two measurements (Y_5 and Y_6) are reported only in Figure 2. 7. Rest of the estimated parameters follow similar performance and hence are not reported here. Estimated noise variances and model parameters are reported in Table 2. 11 - Table 2. 13 given in Appendix to Chapter 2. It can be observed from these tables that BN based soft sensors can accurately estimate the unknown noise variances and the model parameters; thus, output predictions are much more accurate.

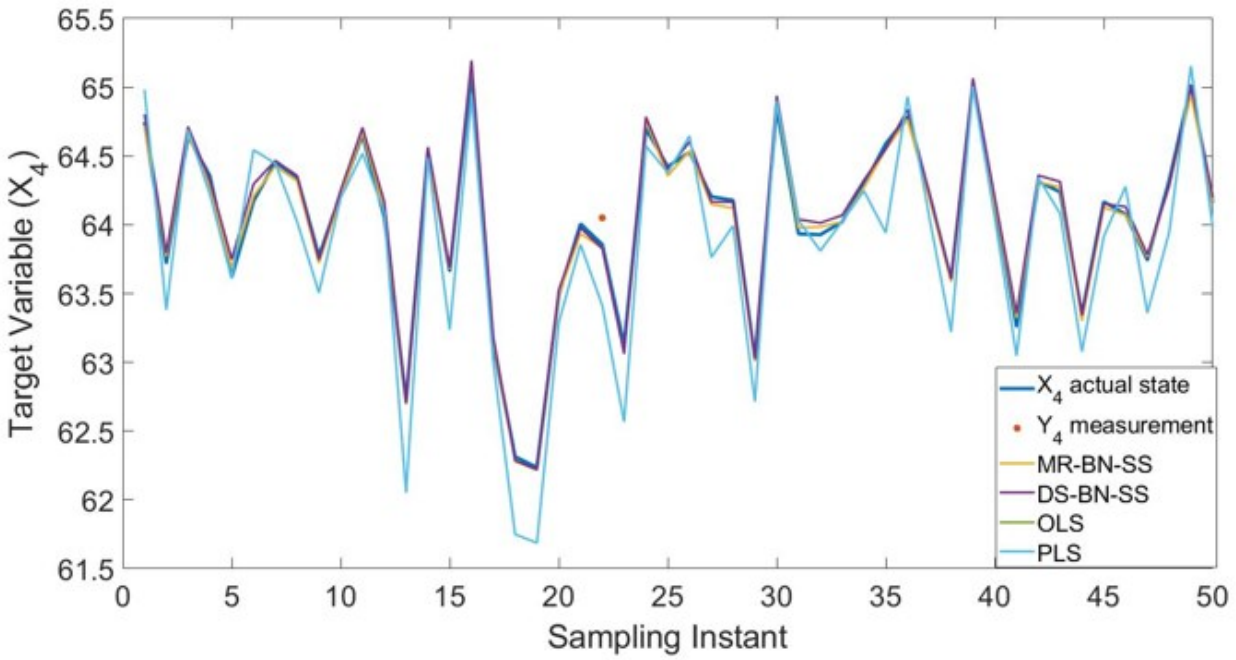

Figure 2. 6: Comparison of BN soft sensor predictions

Variables	True and Noisy measurements	DS- BN-SS	MR-BN-SS	OLS	PLS
X_4	0.6472	0.0734	0.0485	0.2516	0.2514

Table 2. 2: Comparison of ARMSE values

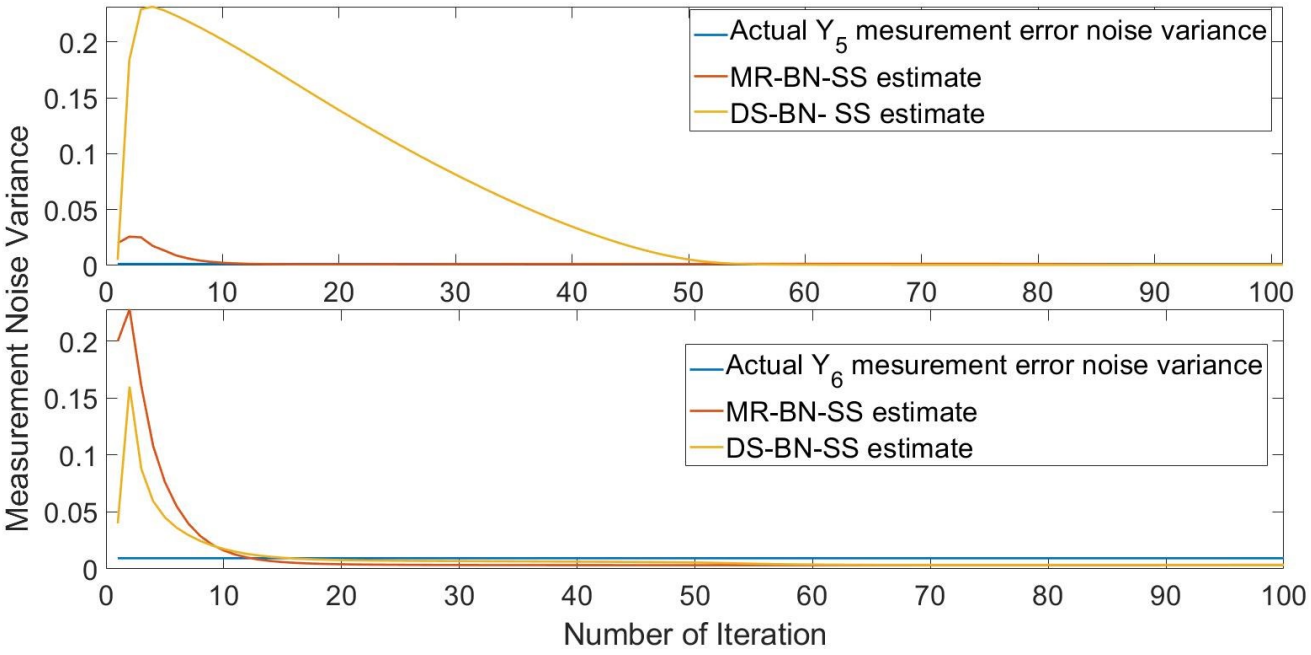

Figure 2. 7: Convergences profile of the noise variances of measurements Y_5 and Y_6

Noisy input data with multi-rate lab data

In this sub-section, one of the key input variable (i.e. Y_2) is assumed to be corrupted with strong noise. Figure 2.8 shows the prediction comparisons of different soft sensors. From this figure and the ARMSE values shown in Table 2. 3, it can be seen that the MR-BN-SS is able to predict the target state with much lower ARMSE than the popular OLS and PLS soft sensors.

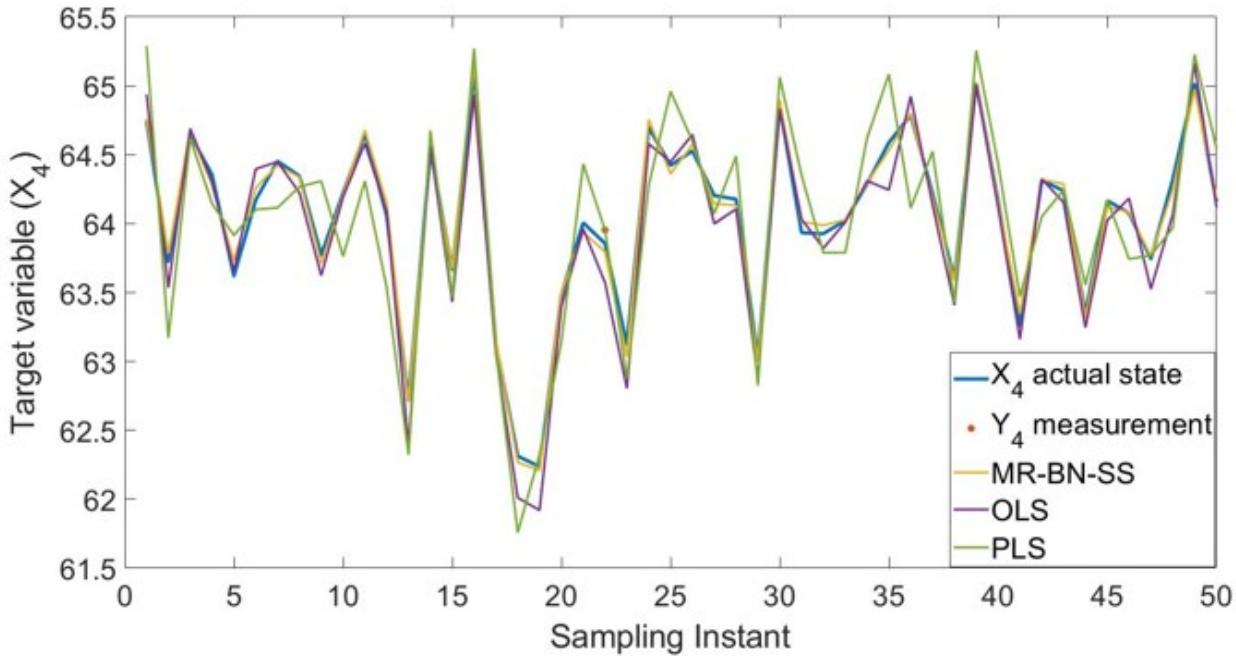

Figure 2. 8: Comparison of MR-BN soft sensor prediction

Variables	True and Noisy measurements	MR-BN-SS	OLS	PLS
X_4	0.3236	0.0588	0.1378	0.3077

Table 2. 3: Comparison of ARMSE values

Completely missing/sensor failure of key input variable

In this scenario, in addition to noisy Y_2 input, measurement of Y_6 is assumed to be completely missing. Figure 2. 9 compares the predictions of the proposed MR-BN-SS under this situation. From the reported ARMSE values in Table 2.4, it can be observed that performance of the proposed approach still outperforms the conventional approaches.

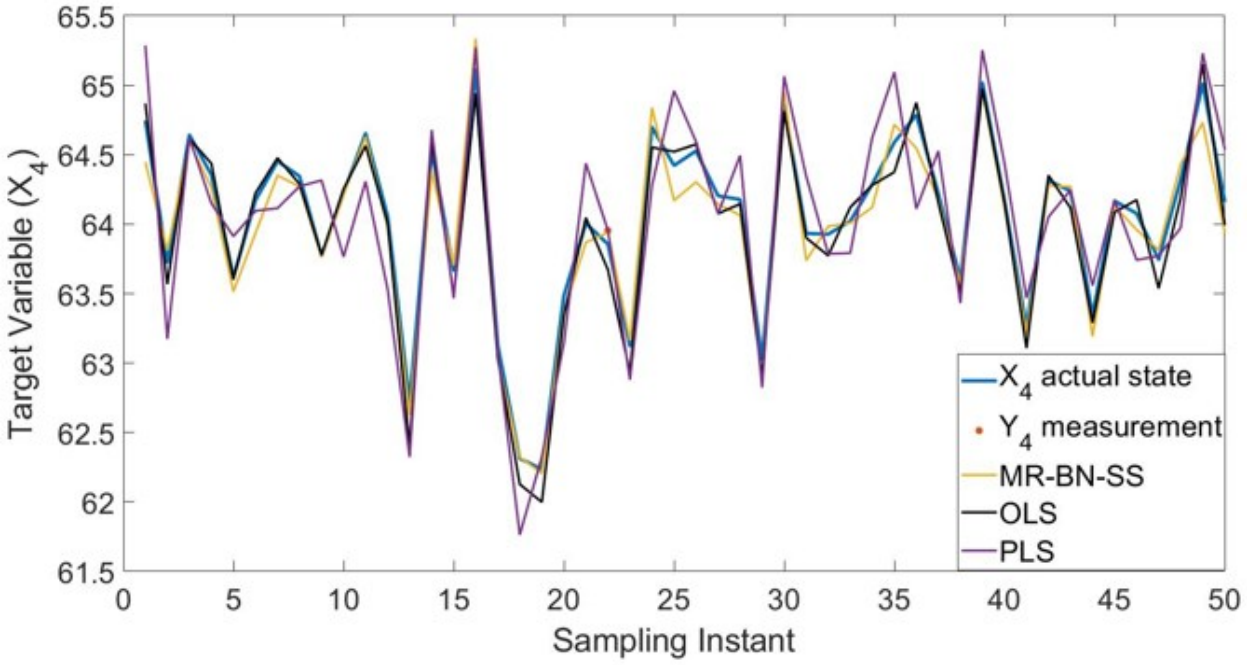

Figure 2. 9: Comparison of MR-BN soft sensor predictions to completely missing case

Variables	True and Noisy measurements	MR-BN-SS	OLS	PLS
X_4	0.3314	0.1175	0.1267	0.3080

Table 2. 4: Comparison of ARMSE values

2.5.2 Industrial case study

In this sub-section, performance of the proposed Bayesian network soft sensor is demonstrated on a real industrial data set. In oil sands industry, upgrading unit is an energy intensive process, where bitumen extracted is hydro-processed and fractionated to convert into lighter components, which are further blended and transferred to refinery for further treatment. Figure 2. 10 shows a schematic representation of a typical process unit in the upgrading section. The liquid feed is sent

to the fractionator and depending on the boiling points, light products are recovered and further processed in the stripping unit to obtain the desired product, (Y_8). For propriety reasons, detailed process is not shown and all data are normalized.

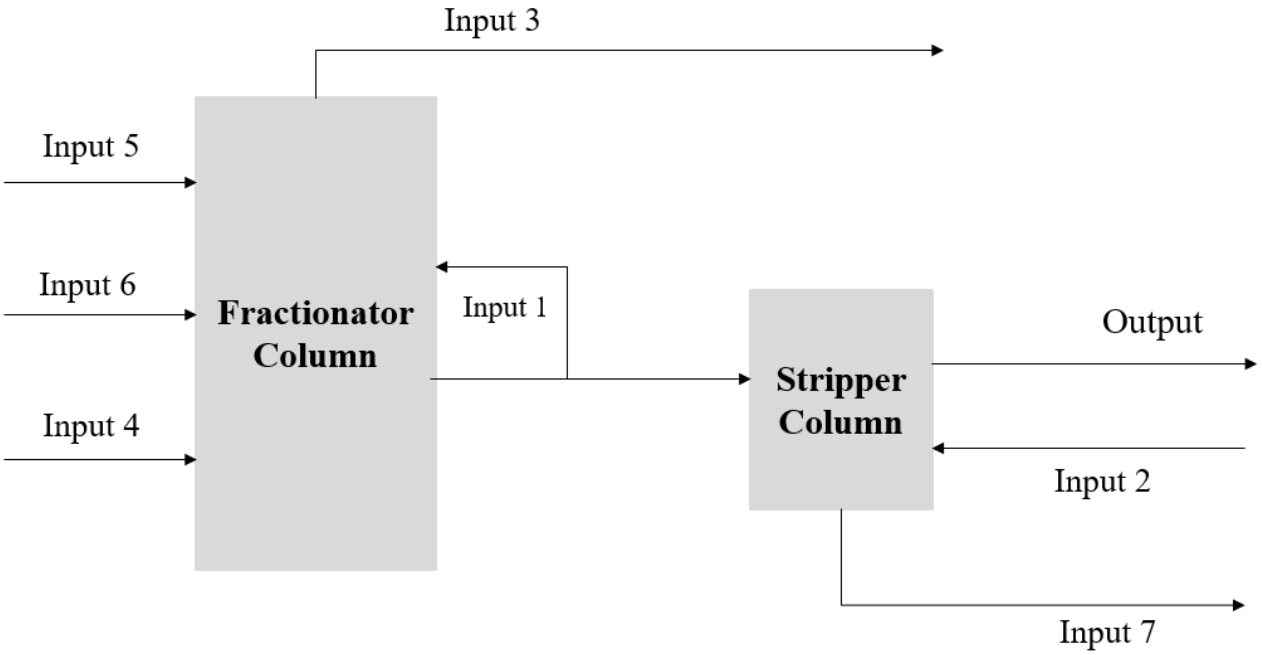

Figure 2. 10: Simplified process diagram of the industrial case study

Here, lab measurements of quality variable (Y_8) are available every 24 hours. Through correlation analysis, 7 inputs, which have higher correlation with the lab data are selected for developing a Bayesian network soft sensor. Further, two BN structures are considered, one is two-layered, second one is multi-layered. When process information is unavailable, the two-layer structure shown in Figure 2. 11 is used. This structure is similar to the conventional ordinary least-squares regression model structure, except that additional hidden layers representing true values of the variables are introduced.

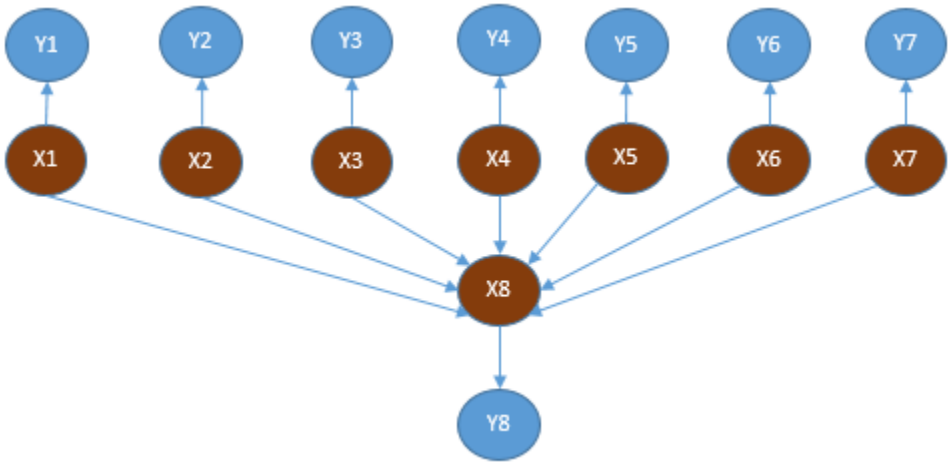

Figure 2.11: Two-layered Bayesian network structure

The multi-layered structure given in Figure 2. 12, utilizes prior process knowledge such as the flow diagram. When it is difficult to separate two or more variables, the variables having higher correlation with the target variable precede the ones having lower correlations.

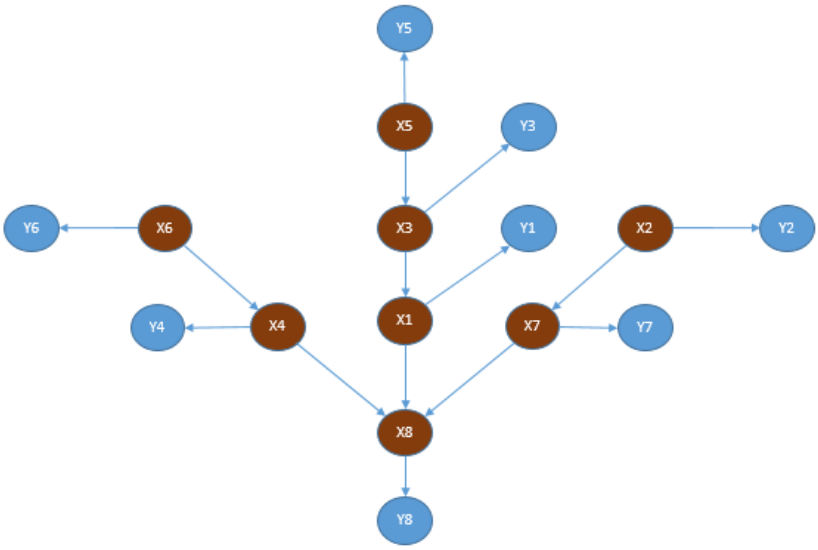

Figure 2.12: Multi-layered Bayesian network structure

Multi-rate and down-sampled BN soft sensors

In this sub-section, Bayesian network based soft sensors are developed considering both down-sampled and multi-rate data, and the performances are compared with OLS and PLS. For parameter learning of DS-BN-SS, a total of 240 and for MR-BN-SS, a total of 4130 samples are used respectively. Trained models are validated on the same set of validation data with 91 lab samples. Figure 2. 13 is the graphical result of different proposed approaches. From this result, it can be observed that the multi-layered MR-BN-SS has a superior performance to the other approaches. Form Table 2. 5 & Table 2. 6, it is evident that BN based soft sensors have better accuracy compared to OLS and PLS soft sensors. It can be understood that through multi-layered structure, much better predictions can be made both in the DS-BN-SS and MR-BN-SS scenarios. On the other hand, data-driven two-layered approach is still better than the conventional soft sensors. This is due to the probabilistic nature of BNs and its ability to estimate noise statistics.

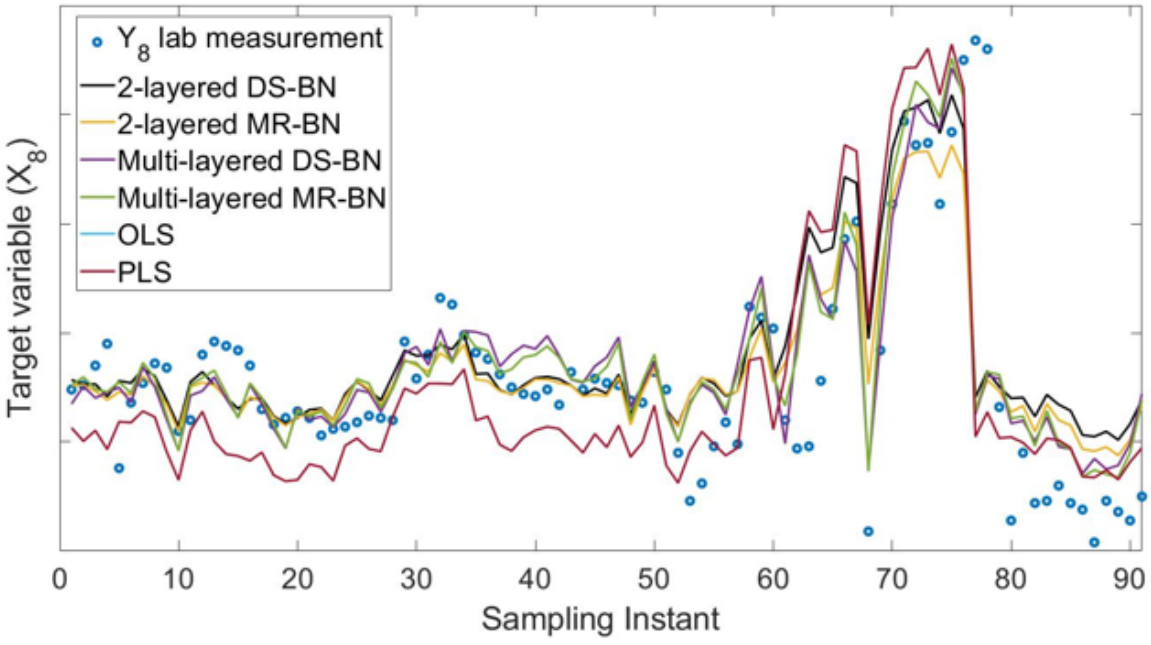

Figure 2.13: Comparison of different soft sensor performances

Approach	Correlation coefficient **(2-layered structure)**	Correlation coefficient **(multi-layered structure)**
MR-BN SS	**0.7200**	**0.7601**
DS-BN SS	0.6957	0.7476
OLS	0.6673	
PLS	0.6698	

Table 2. 5: Correlation coefficients

Approach	RMSE **(2-layered structure)**	RMSE **(multi-layered structure)**
MR-BN SS	**3.3865**	**3.2194**
DS-BN SS	3.6408	3.2613
OLS	4.1703	
PLS	4.1458	

Table 2. 6: RMSE values

MR-BN-SS with noisy input

In this sub-section, key input variable Y_1 is artificially injected with strong sensor noise. Figure 2. 14 shows the performance of MR-BN-SS compared to the conventional OLS and PLS

methods. From this figure, one can see the great ability of MR-BN-SS to handle input noise. It has higher correlation coefficient (reported in Table 2. 7) and lower RMSE (reported in Table 2. 8). Once again, multi-layered MR-BN-SS has the best performance.

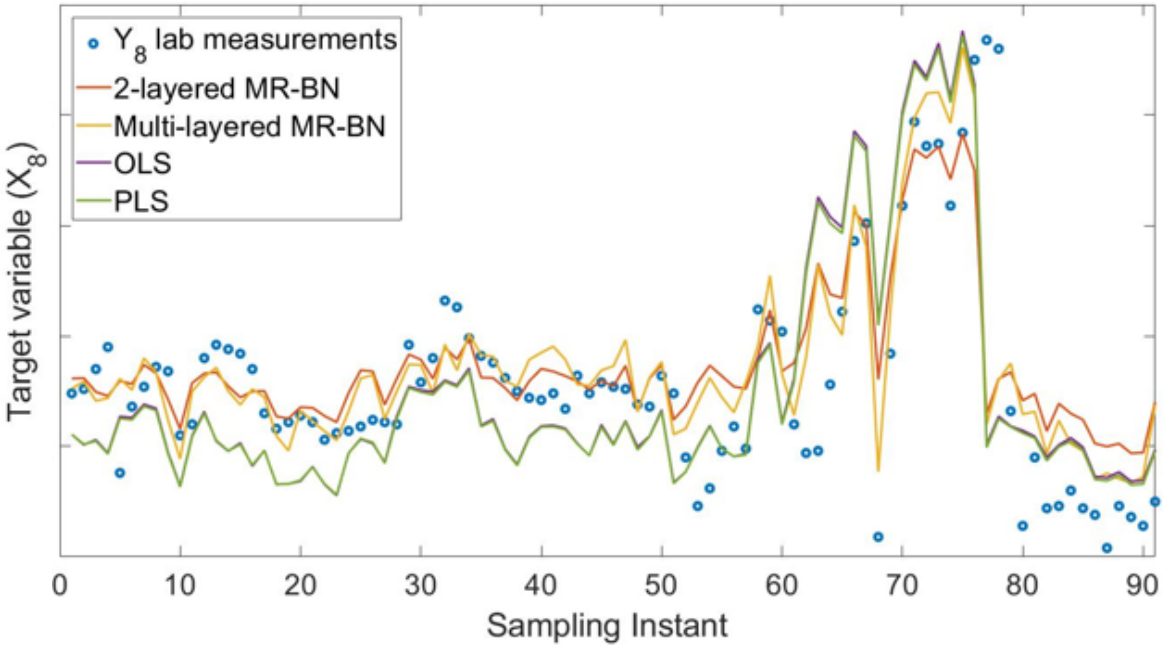

Figure 2.14: Comparison of different soft sensor predictions

Approach	Correlation coefficient **(2-layered structure)**	Correlation coefficient **(multi-layered structure)**
MR-BN-SS	0.7158	0.7510
OLS	0.6602	
PLS	0.6623	

Table 2. 7: Correlation coefficients

Approach	RMSE **(2-layered structure)**	RMSE **(multi - layered structure)**
MR-BN-SS	3.4792	3.2815
OLS	4.2091	
PLS	4.1789	

Table 2. 8: RMSE values

MR-BN-SS with input noise and missing measurement

In this case, on top of all the assumptions made in the previous trial in section 0, measurement of input variable Y_4 is completely missing. From Figure 2. 15, it can be seen that even with a missing key input variable, MR-BN-SS is able to capture the trend of the lab measurements well. Compared to the other soft sensors, as reported in Table 2. 9 and Table 2. 10, multi-layered MR-BN–SS prediction has higher correlation and smaller RMSE.

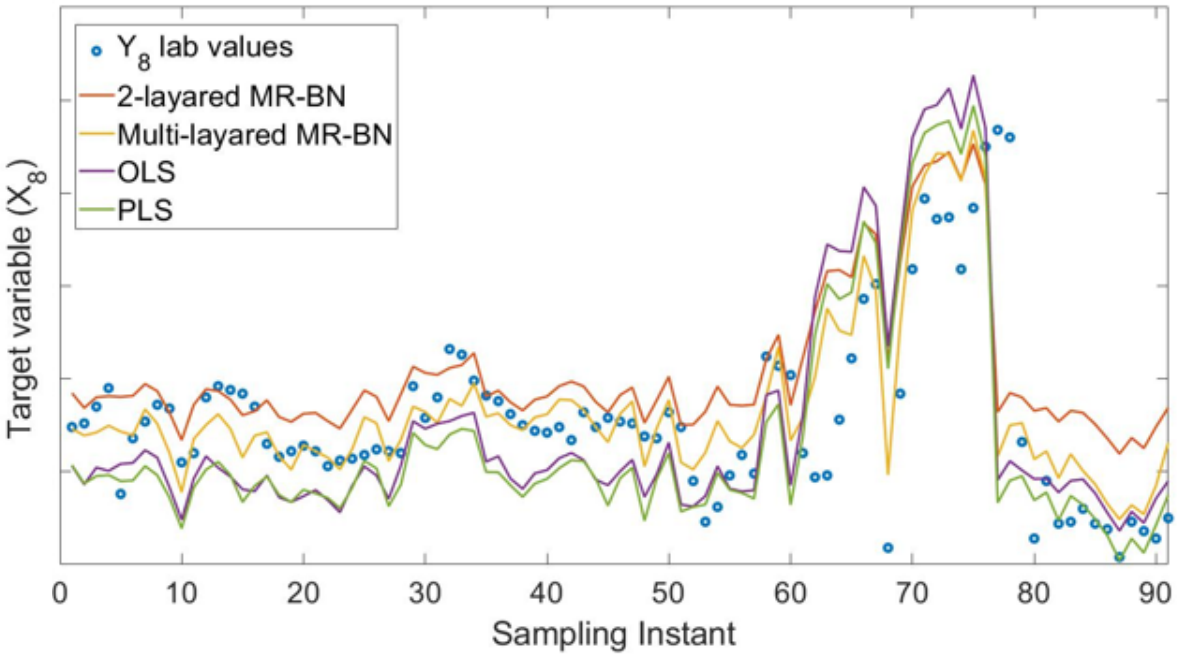

Figure 2.15: Comparison of different soft sensor predictions

Approach	Correlation coefficient **(2-layered structure)**	Correlation coefficient **(multi -layered structure)**
MR-BN SS	0.6898	0.7316
OLS	0.6597	
PLS	0.6735	

Table 2. 9: Correlation coefficients

Approach	RMSE **(2-layered structure)**	RMSE **(multi -layered structure)**
MR-BN SS	4.2534	3.4933
OLS	4.7251	
PLS	4.5846	

Table 2. 10: RMSE values

2.6 Conclusion

The conventional soft sensors such as OLS, PLS and ANN, which are widely used in industry, do not incorporate the causal relationship between the process variables while developing a soft sensor. Moreover, these approaches do not always consider noises for the input variables. To account for these issues, a novel Bayesian network based soft sensor is proposed, which has a flexibility of incorporating prior process knowledge into the soft sensor development stage. This approach accounts for noisy input/output and missing data in probabilistic framework along with Bayesian inference. Analytical solutions are derived under the proposed framework.

Developed multi-rate Bayesian network soft sensor is validated on simulation and industrial cases. From these results, it is observed that for all scenarios considered, the proposed approach has demonstrated superior performance compared to the popular OLS and PLS. The importance of utilizing prior process knowledge is further demonstrated by considering two different BN structures. It is observed that BN soft sensor built through prior knowledge has out-performed the data-driven BN soft sensor. Thus, obtaining optimal

Bayesian network structure from process data is a challenging problem and will be an interesting direction to explore further.

Chapter 3

Robust Bayesian Network Soft Sensor Development for Multi-Rate Data

3.1 Introduction

Outliers are data points appearing noticeably away from the normal operating region. Not accounting for the outliers during soft sensor development process can lead to poor soft sensor predictions . Thus, it is vital to account for outliers in the soft sensor development. Chapter 2 deals with developing a BN based soft sensor under the assumption that data is free of outliers. Thus, current chapter extends previous chapter to robust BN soft senor development. Outliers in input measurements may occur due to large disturbances, instrument failures or sudden changes in operational modes. Outliers in the output variable (lab data) can occur due to human error in recording, collecting samples or mis-calibration of lab equipment.

In literature, researchers proposed different frameworks to robust modeling methods, which can be classified into deterministic and probabilistic approaches. In deterministic modeling, robust regression by Huber's M-estimator [42] is widely used, where the weighted least squared function or the likelihood function is replaced by a robust function that reduces the effect of outliers by assigning appropriate weight to each data point. This weight can be calculated through different objective functions; one example could be inverse of a distance measure. As the data point moves away from the normal operating region, its effect on objective function will be smaller. Robust deterministic approaches belong to this class includes R-estimates [43], which minimizes sum of score of ranked residuals, and S-estimates [44] that

minimizes variance of the residuals. More details of these approaches can be found in (Kodamana et al. 2018). Major drawback of this approach is in selecting appropriate tuning parameters for the robust functions.

On the other hand, probabilistic robust modeling approaches address the outlier issue systematically [45]. Typically, first step is to select model structure. This can be either non-parametric model (fully data-driven) or parametric model such as Auto Regressive Exogenous (ARX) or state space models [46]. For high dimensional data, dimensionality reduction approaches are used, such as PCA and PLS. Next step would be selecting appropriate noise model to account for the outliers. Common distributions used to model the outliers are mixture of Gaussian [47], flat topped t [48] and Student's t-distributions [49]. Once appropriate noise model is chosen, the final step is in selecting suitable method for parameter learning, which can be carried out through maximum likelihood (ML) or Bayesian frameworks. Under the ML framework, EM algorithm is applied when data contains hidden variables [50,51,52,53]. Also, in recent literature, EM algorithm is successfully used for robust identification of linear variables [54]. Meanwhile, Bayesian methods calculate the posterior probability distribution of parameters given the prior information [45]. Examples of such methods are Variational Bayesian (VB) [55], Expectation Propagation (EP) [56] and Markov Chain Monte Carlo (MCMC) [57] approaches. Although these methods are popular in literature, MCMC approach suffers in severe computational intensity, while obtaining explicit analytical solution through VB is a difficult task. Further, selecting appropriate prior probability for VB approach is challenging as well.

A few literature on probabilistic robust identification methods include, robust identification of ARX models by Kodamana et al. (2018), mixture PPCA[58], and more recent work by Fan et al. (2018) on robust probabilistic slow feature analysis (PSFA). Fan et al. (2018)

proposed a robust probabilistic slow feature analysis PSFA that is used to extract temporally correlated dynamic features from high-dimensional data corrupted with outliers. In their work, student t-distribution is applied to model the outliers and the problem is formulated in ML framework. From this work, authors show that Student's t-distribution is able to handle outliers well due to its heavier tail, size of which is controlled by the degree of freedom variable v [59].

From the above discussion it is evident that robust probabilistic approaches, specifically student t-distribution, handle outliers in a systematic way and possess several advantages compared to robust deterministic modeling. The current chapter focusses on developing robust multi-rate Bayesian network soft sensor (RMR-BN-SS) by modeling outliers through student t-distribution.

3.2 Modeling assumptions

Assumption 3. 1: For measurements of any j^{th} variable (i.e. input or quality variable) corrupted with outliers, the noise term (ε_j) of the linear measurement model given by Equation (2.2) is assumed to follow student's-t distribution with zero mean, variance (σ_{y_j}) and degree of freedom (v_j) given as follows:

$$\varepsilon_j \sim t(0, \sigma_{y_j}^2, v_j) \tag{3. 1}$$

So, corresponding measurements will follow the same distribution as:

$$p(Y_j|X_j) \sim t(X_j, \sigma_{y_j}^2, v_j) \tag{3. 2}$$

Mathematically, t-distribution can be seen as scaled Gaussian distribution of unknown variance scale r, which follows a Gamma distribution. This can be expressed as the following:

$$p\left(Y_j\middle|X_j, \sigma_{y_j}^2, r_j\right) = \int p\left(Y_j\middle|X_j, \sigma_{y_j}^2, r_j\right) p(r_j|v_j)\, dr \tag{3. 3}$$

where

$$p(r_j|v_j) \sim \Gamma\left(\frac{1}{2}v_j, \frac{1}{2}v_j\right) \tag{3. 4}$$

$$p\left(Y_j \middle| X_j, \sigma_{y_j}^2, r_j\right) \sim N\left(X_j, \sigma_{y_j}^2 / r_j\right) \tag{3. 5}$$

where unknown latent variance scale variable r_j is considered as hidden random variable and need to be estimated along with hidden states.

For relations between the hidden states, the linear conditional distribution model discussed in Equation (2.5) is considered. For input or quality variables not corrupted with outliers, the linear measurement models given by Equations (2.3) - (2.5) are considered.

Note that the measurement and state noise variances, and degree of freedom variable (i.e. $\sigma_{y_j}^2, \sigma_{x_j}^2, v_j$) are unknown and need to be estimated along with the unknown model parameters between the hidden states $\boldsymbol{\beta}_j = [\beta_{0,j}, \beta_{1,j} \ldots \beta_{N_{Pa},j}]$. For any variable j, all the unknown parameters can be represented in vector form as:

$$\Theta_j = [\theta_m \ \theta_e \ \theta_o] \tag{3. 6}$$

where:

θ_m is the parameters vector representing the model parameters, $\boldsymbol{\beta}_j$

θ_e is the parameters vector representing noise variance terms, $\sigma_{y_j}^2, \sigma_{x_j}^2$

θ_o is the degree of freedom v_j coming from the assumed t-distribution

3.3 Development of robust Bayesian network based soft sensor

By considering the modeling assumptions made in Section 3.2, objective of this study is to develop robust BN soft sensor, which is insensitive to the outliers. Thus, robust BN soft sensors

for down-sampled/ multi-rate data are developed through the following three steps: (1) construction of BN structure, (2) parameter learning and (3) inference. These are discussed individually in the following sections.

3.3.1 Construction of Bayesian network structure

In this chapter, Bayesian network structure shown in Figure 2. 5 is considered for simulation studies, and both the two-layered and multi-layered structures shown in Figure 2. 11 and Figure 2. 12 are considered for the industrial problem.

3.3.2 Robust parameter learning for down-sampled data

Consider for a batch of data with size N, the observed variables $Y = \{Y_1, \dots, Y_m\}$ and hidden variables $X = \{X_1, \dots, X_m\}, r = \{r_1, \dots, r_m\}$. Unknown parameters are shown in Equation (3. 6). The unknown parameters Θ can be estimated by maximizing the joint log-likelihood function as follows:

$$\Theta^* = arg\max_{\Theta} \log p(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{r}|\Theta) \tag{3. 7}$$

For simple BN structure of flow-network problem, considered in Figure 2. 5, using the conditional independence properties, Equation (3. 7) can be further decomposed as given in Equation (3. 8). In this Equation $j = K$ is the node that is corrupted with outliers and $p(\boldsymbol{X_c}|I)$ refers to prior information of the source node X_c.

$$\begin{aligned} &\max_{\Theta} log\, p(X, Y, r) \\ &= \max_{\Theta} \Bigg[\log p\left(Y_{j=K} \middle| X_K, r_K\right) + \log p(r_K) \\ &+ \sum_{j=1, j\neq K}^{m-1} \log p\left(Y_j \middle| X_j\right) + \sum_{j=1}^{m-c} \log p\left(X_j \middle| Pa\left(X_j\right)\right) + \log p(X_c|I) \Bigg] \end{aligned} \tag{3. 8}$$

where $p(Y|X,r)$ and $p\big(X\big|Pa(X)\big)$ both follow Gaussian distribution, while $p(r)$ follows a Gamma distribution.

Assumption 3. 2: The prior distribution for the source node, $p(X_c|I)$, can be utilized if the information is available. However, in this work, it is assumed to follow uniform distribution and this assumption is considered for remaining part of the derivation.

EM algorithm for down-sampled data

Since the joint density function given in Equation (3. 8) contains hidden nodes (X,r), direct maximization is difficult and may result in sub-optimal solutions. Thus, here we resort to expectation maximization (EM) approach to solve joint likelihood function given in Equation (3. 8).

E-step:

In the E-step, given the observations (**Y**) and parameters (Θ_r) at the r^{th} iteration, expectation of the complete log-likelihood function w.r.t all the hidden variables **X** and $\boldsymbol{r}$ is calculated i.e. Q function, given as:

$$Q(\Theta|\ \Theta_r) = E_{X,r|Y,\Theta_r}[\log p(\boldsymbol{Y},\boldsymbol{X},\boldsymbol{r}|\Theta)] \tag{3. 9}$$

Using the property of D-separation principle and conditional dependencies (or independencies) among the *m* random variables, the Q function can be expressed as follows:

$$Q(\Theta,\Theta_r) = E_{X,r|Y,\Theta_r}\left[\log p\big(Y_{j=K}\big|X_K,r_K,\Theta\ \big) + \sum_{j=1,j\neq K}^{m-1}\log p\big(Y_j\big|X_j,\Theta\big) + \sum_{j=1}^{m-c}\log p\big(X_j\big|Pa\big(X_j\big),\Theta\big) + \log p(r_K|\Theta)\right] \tag{3. 10}$$

For a batch of data with size N, and under the assumption that measurements are independent, Equation (3. 10) can be expanded as:

$$\mathrm{Q}(\Theta|\Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{j=1,j\neq K}^{m-1}\sum_{i=1}^{N}\log p\left(y_{i,j}\middle|x_{i,j},\Theta\right) + \sum_{i=1}^{N}\log p\left(y_{i,K}\middle|x_{i,K},r_{i,K},\Theta\right) + \sum_{j=1}^{m-c}\sum_{i=1}^{N}\log p\left(x_{i,j}\middle|Pa\left(x_{i,j}\right),\Theta\right) + \sum_{i=1}^{N} logp(r_{i,K}|\Theta)\right] \quad (3.\ 11)$$

where c refers to number of source nodes. To further simplify the computation of E-step, the Q function can be expressed as follows:

$$Q(\Theta,\Theta_r) = Q_1(\Theta,\Theta_r) + Q_2(\Theta,\Theta_r) + Q_3(\Theta,\Theta_r) \quad (3.\ 12)$$

where:

$$Q_1(\Theta,\Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{i=1}^{N}\log p\left(y_{i,K}\middle|x_{i,K},r_{i,K},\Theta\right) + \sum_{j=1,j\neq K}^{m-1}\sum_{i=1}^{N}\log p\left(y_{i,j}\middle|x_{i,j},\Theta\right)\right] \quad (3.\ 13)$$

$$Q_2(\Theta,\Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{j=1}^{m-c}\sum_{i=1}^{N}\log p\left(x_{i,j}\middle|Pa\left(x_{i,j}\right),\Theta\right)\right] \quad (3.\ 14)$$

$$Q_3(\Theta,\Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{i=1}^{N} logp(r_{i,K}|\Theta)\right] \quad (3.\ 15)$$

From Assumption 3. 1, for any i^{th} sample and j^{th} variable corrupted with outliers, the conditional distributions $p\left(y_{i,j}\middle|x_{i,j},r_{i,j},\Theta\right)$ and $p\left(r_{ij}\middle|\Theta\right)$ can be expressed as:

Measurement with outliers

$$p\left(y_{i,j}\middle|x_{i,j},r_{i,j},\Theta\right) = \frac{1}{\sqrt{2\pi\sigma_{y_j}^2/r_j}}\exp\left(-\frac{\left(y_{ij}-x_{ij}\right)^2}{2\sigma_{y_j}^2/r_j}\right) \quad (3.\ 16)$$

Gamma distribution

$$p(r_{ij}|\Theta) = -\log\Gamma\left(\frac{v_j}{2}\right) + \frac{v_j}{2}\log\left(\frac{v_j}{2}\right) + \left(\frac{v_j}{2} - 1\right)\log(r_{ij}) - \frac{v_j}{2}r_{ij} \quad (3.\ 17)$$

Thus, expanding $Q_1(\Theta, \Theta_r)$ using Equation (3. 16) gives:

$$Q_1(\Theta, \Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{i=1}^{N}\sum_{j=1,,j\neq K}^{m-1}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_j^2}}\right) - \frac{(y_{ij} - x_{ij})^2}{2\sigma_j^2}\right] + \sum_{i=1}^{N}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_j^2/r_j}}\right) - \frac{(y_{iK} - x_{iK})^2}{2\sigma_j^2/r_j}\right]\right] \quad (3.\ 18)$$

Using Equation (2. *15*), $Q_2(\Theta, \Theta_r)$ can be extended as:

$$Q_2(\Theta, \Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{i=1}^{N}\sum_{j=1}^{m-c}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_{x_j}^2}}\right) - \frac{\left(x_{ij} - \beta_{j\,0} - \beta_{j\,0+P}Pa(x_{ij})\right)^2}{2\sigma_{x_j}^2}\right]\right] \quad (3.\ 19)$$

Using linear property of expectation, the following statistics are required to compute Q_1 and Q_2.

$$\mathrm{E}_{\mathrm{X,r|Y},\Theta_r}[x_{ij}^2], \qquad \mathrm{E}_{\mathrm{X,r\,|Y},\Theta_r}[x_{ij}\, r_{ij}], \qquad \mathrm{E}_{\mathrm{X,r\,|Y},\Theta_r}[x_{ij}^2\, r_{ij}] \quad (3.\ 20)$$

$$\mathrm{E}_{\mathrm{X,r|Y},\Theta_r}[x_{ij}Pa(x_{ij})], \qquad \mathrm{E}_{\mathrm{X,r|Y},\Theta_r}[\,Pa_p(x_{ij})\,Pa_{p+1}(x_{ij})] \quad (3.\ 21)$$

Assumption 3. 3 The hidden variable introduced form the t-distribution r_j is dependent on its corresponding hidden state x_j as well as its corresponding measurements y_j. However, due to complex dependencies between these variables, computing the expectation terms i.e. $E_{X,r\,|Y,\Theta_r}[x_{ij}\ r_{ij}]$ will be difficult. Therefore, in this formulation we have assumed that the hidden variable r_j is constant w.r.t variable x_j and y_j[59].

Utilizing Assumption 3. 3, these statistics can be further simplified as:

$$\mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[x_{ij}r_{ij}] = \mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[x_{ij}]\ \mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[\,r_{ij}] \tag{3. 22}$$

$$\mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[x_{ij}^2\ r_{ij}] = \mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[x_{ij}^2]\ \mathrm{E}_{\mathrm{X,r\,|Y,}\Theta_r}[\,r_{ij}] \tag{3. 23}$$

$$\mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}[x_{ij}^2] = var\ (x_{ij}) + \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}[x_{ij}]\ \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [x_{ij}] \tag{3. 24}$$

$$\mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [x_{ij}\ Pa(x_{ij})] = cov\left(x_{ij}, Pa(x_{ij})\right) +\ \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [x_{ij}]\ \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [Pa(x_{ij})] \tag{3. 25}$$

For a node with multiple parents, additional statistics concerning relationship between the parent nodes will be necessary. These statistics can be obtained as:

$$\begin{aligned}\mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [Pa_p(x_{ij})\ Pa_{p+1}(x_{ij})] = cov\big(Pa_p(x_{ij}), Pa_{p+1}(x_{ij})\big) + \\ \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [Pa_p(x_{ij})]\ \mathrm{E}_{\mathrm{X,r|Y,}\Theta_r}\ [Pa_{p+1}(x_{ij})]\end{aligned} \tag{3. 26}$$

Now, lastly expanding Equation (3. 15), using gamma distribution in Equation (3. 17) gives:

$$Q_3(\Theta, \Theta_r) = E_{X,r|Y,\Theta_r}\left[\sum_{i=1}^{N} -\log\Gamma\left(\frac{v_j}{2}\right) + \frac{v_j}{2}\log\left(\frac{v_j}{2}\right) + \left(\frac{v_j}{2} - 1\right)\log(r_{ij}) - \frac{v_j}{2}r_{ij}\right] \tag{3. 27}$$

Evaluating Equation (3. 27) would result in need of below statistics.

$$\mathrm{E}_{\mathrm{X,r}\,|\,\mathrm{Y},\Theta_r}[r_{ij}] \tag{3. 28}$$

$$\mathrm{E}_{\mathrm{X,r}\,|\,\mathrm{Y},\Theta_r}\left[\ln(r_{ij})\right] \tag{3. 29}$$

Thus, the required statistics (i.e. from Equations (3. 22) - (3. 26) and Equations (3. 28) - (3. 29) are obtained evaluating posterior distribution of the hidden variables (i.e. $p(\boldsymbol{X}|\boldsymbol{r},\boldsymbol{Y},\Theta_r)$, $p(\boldsymbol{r}|\boldsymbol{Y},\boldsymbol{X},\Theta_r)$) as illustrated in the following section.

Computing the statistics from posterior distribution

A full posterior distribution for the hidden variable $\boldsymbol{X}$ via Bayesian rule can be expressed as:

$$p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{r},\Theta_r) = \frac{p(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{r}|\Theta_r)}{p(\boldsymbol{Y})} \tag{3. 30}$$

Thus, for a batch of data, posterior distribution of the hidden state $\boldsymbol{X}$ is obtained from computation of below formulation, where γ is a normalizing constant that equals $p(\boldsymbol{Y})^{-1}$.

$$p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{r},\Theta_r)$$

$$= \gamma\left[p\left(Y_{j=K}|X_K,r_K,\Theta_r\right)\prod_{j=1,j\neq K}^{m-1} p\left(Y_j|X_j,\Theta_r\right)\prod_{j=1}^{m-c} p\left(X_j|Pa\left(X_j\right),\Theta_r\right)p(r_K|\Theta_r\,)p(X_c|I)\right] \tag{3. 31}$$

Maximizing logarithmic of posterior distribution function w.r.t each hidden state will result in set of simultaneous Equations, from which mode of hidden states can be computed. Using Assumption 3.2, prior probability term is omitted.

$$\hat{\boldsymbol{X}} = \max_{\boldsymbol{X}} \log p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{r},\Theta_r) = \max_{X}\left[logp\left(Y_{j=K}|X_K,r_K,\Theta_r\right) + \sum_{j=1,j\neq K}^{m-1}\log p\left(Y_j|X_j,\Theta_r\right) +\right.$$

$$\sum_{j=1}^{m-c} \log p\left(X_j \middle| Pa\left(X_j\right), \Theta_r\right) + logp\left(r_K \middle| \Theta_r\right) + \log(\gamma)\Big] \tag{3. 32}$$

For a batch of data with size N, Equation (3. 32) can be expressed as:

$$\hat{\boldsymbol{X}} = \max_{\boldsymbol{X}} \log p(\boldsymbol{X}|\boldsymbol{Y}, \boldsymbol{r}, \Theta_r) = \max_{X}\Big[\sum_{i=1}^{N} logp\left(y_{ij}\middle|x_{iK}, r_{iK}, \Theta_r\right) + \sum_{j=1,j\neq K}^{m-1}\sum_{i=1}^{N} \log p\left(y_{ij}\middle|x_{ij}, \Theta_r\right) +$$

$$\sum_{j=1}^{m-c}\sum_{i=1}^{N} \log p\left(x_{ij}\middle|Pa\left(x_{ij}\right), \Theta_r\right) + \sum_{i=1}^{N} logp\left(r_{iK}\middle|\Theta_r\right) + \log(\gamma)\Big] \tag{3. 33}$$

Considering the first order optimality conditions for Equation (3. 33) i.e. $\frac{\partial\, logP\,(X,r|Y,\Theta_r)}{\partial X} = 0$, for any i^{th} sample and j^{th} variable, the estimates of hidden states can be represented in analytical form as given by Equations (3. 34) - (3. 35), where N_{ch} stands for the number of children that the target node has in the Bayesian network structure and $Ch(x_j)$ is the child of variable x_j.

$$\alpha\left(x_{j=K}\right) = \left(\frac{\widehat{r_{ij}}}{\sigma_{y_j}^2} + \frac{1}{\sigma_{x_j|Pa(x_j)}^2} + \sum_{c=1}^{N_{Ch}} \frac{\left(\beta_{c,j}^2\right)}{\sigma_{Ch_c(x_j)|x_j}^2}\right) \tag{3. 34}$$

$$\hat{x}_{i,j=K} = \frac{\frac{\widehat{r_{ij}}\, y_{ij}}{\sigma_{yj}^2} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j}\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{ij})\beta_{0+p,j}\right)}{\sigma_{Ch(x_j)|x_j}^2} + \frac{\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij})\beta_{0+p,j}\right)}{\sigma_{x_j|Pa(x_j)}^2}}{\alpha\left(x_{j=K}\right)} \tag{3. 35}$$

Note that for $j = 1, \ldots, m$ and $j \neq K$ (i.e. variables not corrupted with outliers), $\widehat{r_{ij}} = 1$. The covariance between hidden states is calculated as discussed in section 0.

Here posterior distribution of r_j can be calculated from Bayes' theorem as follows.

$$p(\boldsymbol{r}|\boldsymbol{Y},\boldsymbol{X},\Theta_r) = \frac{p(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{r},\Theta_r)p(\boldsymbol{r}|\Theta_r)}{p(\boldsymbol{Y})} \tag{3. 36}$$

Since x_{ij} and r_{ij} are independent[59], $logp(X|Pa(X),\Theta_k)$ term is omitted from the above equation and it is expanded as follows, where γ equals $p(\boldsymbol{Y})^{-1}$and is a normalizing constant.

$$p\left(r_{j=K}|Y_j,X_j,\Theta_r\right) = \gamma\left(\prod_{i=1}^{N} Normal\, p(y_{iK}|r_{iK},x_{iK},\Theta_r)\prod_{i=1}^{N} Gamma\, p(r_{iK}|\frac{v_K}{2},\frac{v_K}{2})\right) \tag{3. 37}$$

Since Gamma distribution is a conjugate prior of Normal distribution, the posterior is expressed as:

$$p\left(r_{j=K}|Y_j,X_j,\Theta_r\right) = \prod_{i=1}^{N} Gamma(r_{ij}|a_{ij},b_{ij}) \tag{3. 38}$$

The standard expressions for the a_{ij},b_{ij} are given as:

$$a_{ij} = \frac{v_j+1}{2} \tag{3. 39}$$

$$b_{ij} = \frac{v_j}{2} + \frac{1}{2\sigma_j^2}\left(y_{ij}-x_{ij}\right)^2 \tag{3. 40}$$

Therefore, it is known that the statistics concerning posterior probability of unknown scale variance r_{i_j} can be obtained from the following standard equations[60].

$$\mathrm{E}_{\mathrm{X,r}\,|\mathrm{Y},\Theta_r}\left[r_{ij}\right] = \frac{a_{ij}}{b_{ij}} \tag{3. 41}$$

$$\mathrm{E}_{\mathrm{X,r}\,|\mathrm{Y},\Theta_r}\left[\log(r_{ij})\right] = \psi\left(a_{ij}\right) - \log(b_{ij}) \tag{3. 42}$$

where:

$$\psi(a_{ij}) = \frac{\Gamma'(r_{ij})}{\Gamma(r_{ij})} \tag{3. 43}$$

M-Step

In the **M-step**, Q function is maximized w.r.t all the unknown parameters. In context of this chapter, maximization can be expanded as:

$$\frac{\partial\, Q}{\partial \theta_m} = 0 \quad \frac{\partial\, Q}{\partial \theta_e} = 0 \tag{3. 44}$$

Due to the non-linear dependencies, obtaining a closed form solution for the degree of freedom variable v_j is not possible. Therefore, it is obtained by directly maximizing the Q function as given in Equation (3. 46). Closed form solutions for the model parameters and noise variances of the variables free of outliers are identical as given in Equations (2. 36) - (2. 39). Only this time, noise variance of the measurement corrupted with outliers is given as follows:

$$\sigma^2_{y_K,r+1} = \frac{E_{X,r|Y,\Theta_r}\left(\sum_{i=1}^{N}(r_{iK})(y_{iK} - x_{iK})^2\right)}{N} \tag{3. 45}$$

$$v_{j,r+1} = \max_{v_j} Q_3\,(\Theta, \Theta_r) \tag{3. 46}$$

3.3.3 Robust parameter learning for multi-rate/missing data with outliers

In the previous section, robust parameter learning for BN soft sensor is performed under the assumption that all the measurements are down-sampled to the sampling instant at which

samples of quality variable are available. In reality, lab samples of quality variable and information of some key variables may be missing due to sensor issues. Accounting for this information in model development stage will help in obtaining accurate model. Therefore, this section formulates robust parameter learning for multi-rate data.

Consider the quality variable measurements consist of observed and missing data as $Y = [Y_{obs}\ Y_{mis}]$ and hidden states as $X = [X_{obs}\ X_{mis}]$ and O and M stands for number of observed and missing data in the training data set respectively.

Based on this assumption, unknown parameters (Θ) can be obtained by maximizing the logarithm of complete likelihood function given as:

$$\max_{\Theta} \log p(\boldsymbol{Y_{obs}}, \boldsymbol{Y_{mis}}, \boldsymbol{X}, \boldsymbol{r}|\Theta) \tag{3. 47}$$

Using the property of independency of measurements and from Bayesian network principles under the assumption that output variable Y_K is assumed to be slow-rate and corrupted with outliers, the joint likelihood function can be factored as follows. One can also formulate for outliers in input variable, following similar steps.

$$p(\boldsymbol{Y_{obs}}, \boldsymbol{Y_{mis}}, \boldsymbol{X}, \boldsymbol{r}|\Theta) = p\left(Y_{K_{obs}}|X_{K_{obs}}, r_{K_{obs}}, \Theta\right)p\left(Y_{K_{mis}}|X_{K_{mis}}, r_{K_{mis}}\Theta\right)$$
$$\prod_{j=1,\ j\neq K}^{m-1} p\left(Y_j|X_j, r_j, \Theta\right)\prod_{j=1}^{m-c} p\left(X_j|Pa\left(X_j\right), \Theta\right)p(r_K|\ \Theta) \tag{3. 48}$$

EM algorithm for robust parameter learning (multi-rate data)

E-Step

Expectation of the complete log-likelihood function w.r.t. all the missing data ($\boldsymbol{Y}_{mis}$), hidden variables ($\boldsymbol{X}$) and the unknown variance scale variable $\boldsymbol{r}$ is evaluated as:

$$Q(\Theta, \Theta_r) = E_{\boldsymbol{X,r,Y_{mis}}|\boldsymbol{Y_{obs}},\Theta_r}[\log p(\boldsymbol{Y_{obs}}, \boldsymbol{Y_{mis}}, \boldsymbol{X}, \boldsymbol{r}|\Theta)] \tag{3. 49}$$

From the joint density function given by Equation (3. 48), Q function can be expressed as:

$$\begin{aligned} Q(\Theta, \Theta_r) = E_{X,r,Y_{mis}|Y_{obs},\Theta_r}\Bigg[& p\left(Y_{K_{obs}}\middle|X_{K_{obs}}, r_{K_{obs}}, \Theta\right) + p\left(Y_{K_{mis}}\middle|X_{K_{mis}}, r_{K_{mis}}\Theta\right) \\ & + \sum_{j=1, j\neq K}^{m-1} p\left(Y_j\middle|X_j, r_j, \Theta\right) + \sum_{j=1}^{m-c} p\left(X_j\middle|Pa\left(X_j\right), \Theta\right) + p(r_K|\ \Theta)\Bigg] \end{aligned} \tag{3. 50}$$

For simplicity, above Equation is assumed to consist of following three parts.

$$\begin{aligned} Q_1(\Theta, \Theta_r) = E_{X,r,Y_{mis}|Y_{obs},\Theta_r}\big[& logp\left(Y_{K_{obs}}\middle|X_{K_{obs}}, r_{K_{obs}}, \Theta\right) \\ & + logp\left(Y_{K_{mis}}\middle|X_{K_{mis}}, r_{K_{mis}}, \Theta\right)\big] \end{aligned} \tag{3. 51}$$

$$Q_2(\Theta, \Theta_r) = E_{X,r,Y_{mis}|Y_{obs},\Theta_r}\left[\sum_{j=1, j\neq K}^{m-1} p\left(Y_j\middle|X_j, r_j, \Theta\right) + \sum_{j=1}^{m-c} p\left(X_j\middle|Pa\left(X_j\right), \Theta\right)\right] \tag{3. 52}$$

$$Q_3(\Theta, \Theta_r) = E_{X,r,Y_{mis}|Y_{obs},\Theta_r}[p(r_K|\ \Theta)] \tag{3. 53}$$

For notation simplicity, $E_{X,r,Y_{mis}|Y_{obs},\Theta_r}$ is denoted as E_{Θ_r}. Based on previous assumptions, for a batch of data with size N, Equation (3. 51) can be further expanded as:

$$\begin{aligned} Q_1(\Theta, \Theta_r) = & E_{\Theta_r}\left[\sum_{i=1}^{O}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_K^2/r_K}}\right) - \frac{\left(y_{K_{obs},i} - x_{K_{obs},i}\right)^2}{2\sigma_K^2/r_K^2}\right]\right] \\ & + E_{\Theta_r}\left[\sum_{i=O+1}^{N}\left[\log\left(\frac{1}{\sqrt{2\pi\sigma_K^2/r_K}}\right) - \frac{\left(y_{K_{mis},i} - x_{K_{mis},i}\right)^2}{2\sigma_K^2/r_K}\right]\right] \end{aligned} \tag{3. 54}$$

As per Assumption 3. 3 , the unknown scale variable r_K is constant[57] in terms of $X_{K_{mis}}$, $Y_{K_{mis}}$. Thus, expectation terms can be further computed as the following:

$$E_{\Theta_r}[x_{j_{mis},i}\, y_{j_{mis},i}] = cov(x_j y_j) + E_{\Theta_r}[x_{j_{mis},i}]E_{\Theta_r}[\, y_{j_{mis},i}] \tag{3. 55}$$

$$E_{\Theta_r}[y_{j_{mis},i}\, r_{j_{mis},i}] = E_{\Theta_r}[y_{j_{mis},i}]E_{\Theta_r}[\, r_{j_{mis},i}] \tag{3. 56}$$

$$E_{\Theta_r}[y^2_{j_{mis},i}] = var(y_{j_{mis},i}) + E_{\Theta_r}[y_{j_{mis},i}]E_{\Theta_r}[\, y_{j_{mis},i}] \tag{3. 57}$$

Note that computation of $Q_2(\Theta|\Theta_r)$ and $Q_3(\Theta|\Theta_r)$ will result in need of same statistics discussed in Equations (3. 20) - (3. 21) and Equations (3. 28) - (3. 29), respectively. The above statistics are obtained from evaluation of posterior distribution discussed in following section.

Posterior distribution computation

Posterior distribution of the hidden states: X (= $[X_{obs}, X_{mis}]$), r and missing Y_{mis} measurement, are obtained through following Bayesian formulation.

From Bayes' rule, the full posterior probability distribution is:

$$p(\boldsymbol{X}, \boldsymbol{Y_{mis}}|\boldsymbol{Y_{obs}}, \boldsymbol{r}, \Theta_r) = \frac{p(\boldsymbol{Y_{obs}}|\boldsymbol{X_{obs}}, r, \Theta_r)p(\boldsymbol{r}|\Theta_r)p(\boldsymbol{X}, \boldsymbol{Y_{mis}}|I)}{p(\boldsymbol{Y_{obs}})} \tag{3. 58}$$

Since denominator of the above Equation acts as a normalizing constant, its inverse is marked as γ and above posterior distribution is rewritten as:

$$p(\boldsymbol{X}, \boldsymbol{Y_{mis}}|\boldsymbol{Y_{obs}},\ \boldsymbol{r}, \Theta_r) = \gamma p(\boldsymbol{Y_{obs}}|\boldsymbol{X_{obs}}, \boldsymbol{r}, \Theta_r)p(\boldsymbol{r}|\Theta_r)p(\boldsymbol{X}, \boldsymbol{Y_{mis}}|I) \tag{3. 59}$$

By using the property of conditional dependence between the nodes, above equation can be further decomposed as:

$$p(\boldsymbol{X}, \boldsymbol{Y_{mis}} | \boldsymbol{Y_{obs}}, \boldsymbol{r}, \Theta_r) \tag{3. 60}$$
$$= \gamma p(\boldsymbol{Y_{obs}} | \boldsymbol{X_{obs}}, \boldsymbol{r_{obs}}, \Theta_r) p(\boldsymbol{Y_{mis}} | \boldsymbol{X_{mis}}, \boldsymbol{r_{mis}}, \Theta_r) p(\boldsymbol{r} | \Theta_r) p(\boldsymbol{X} | Pa(\boldsymbol{X}), \Theta_r) p(Pa(\boldsymbol{X}) | I)$$

Note that in the above formulation, $p(Pa(X)|I)$ is prior information of the source node. Thus, given the observed measurements (Y_{obs}) and parameter vector (Θ_r), estimates of hidden states (X) and missing measurements (Y_{mis}) are obtained by maximizing the log of posterior distribution function w.r.t. these variable i.e.

$$\max_{X, Y_{mis}} logp(\boldsymbol{X}, \boldsymbol{Y}_{mis} | \boldsymbol{Y}_{obs}, \boldsymbol{r}, \Theta_r)$$
$$= \max_{X, Y_{mis}} [logp(\boldsymbol{Y}_{obs} | \boldsymbol{X}_{obs}, \boldsymbol{r_{obs}}, \Theta_r) + logp(\boldsymbol{Y}_{mis} | \boldsymbol{X}_{mis}, \boldsymbol{r}_{mis}, \Theta_r) \tag{3. 61}$$
$$+ logp(\boldsymbol{r} | \Theta_r) + logp(\boldsymbol{X} | Pa(\boldsymbol{X}), \Theta_r) + logp(Pa(\boldsymbol{X}) | I) + \log(\gamma)]$$

Since $p(\boldsymbol{r}|\Theta_r)$ term is independent of the variables X and Y_{mis}, this term is omitted. Based on the Assumptions 2.1 & 2.2 and Assumptions 3.2, Equation (3. 61) can be expanded as:

$$\max_{X, Y_{mis}} logp(\boldsymbol{X}, \boldsymbol{Y}_{mis} | \boldsymbol{Y}_{obs}, \boldsymbol{r}, \Theta_r) = \max_{X, Y_{mis}} \left[\sum_{i=1}^{O} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma^2_{y_{j=K}}) - \right. \right.$$
$$\left. \frac{r_{K_{obs},i} \left(y_{K_{obs},i} - x_{K_{obs},i} \right)^2}{2\sigma^2_{y_K}} \right] + \sum_{i=O+1}^{N} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma^2_{y_K}) - \frac{r_{K_{mis},i} \left(y_{K_{mis},i} - x_{K_{mis},i} \right)^2}{2\sigma^2_{y_K}} \right] +$$
$$\sum_{j=1,\ j \neq K}^{m-1} \sum_{i=1}^{N} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma^2_{y_j}) - \frac{(y_{ij} - x_{ij})^2}{2\sigma^2_{y_j}} \right] + \sum_{i=1}^{N} \sum_{j=1, j \neq K}^{m-c} \left[\frac{1}{\sqrt{2\pi}} \log(\sigma^2_{x_j}) - \right. \tag{3. 62}$$
$$\left. \left. \frac{\left(x_{ij} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}} Pa_p(x_{ij}) \beta_{0+p,j} \right)^2}{2\sigma^2_{x_j}} \right] + \log(\gamma) \right]$$

Considering the first order optimality conditions for Equation (3. 62) (first order derivatives w.r.t. hidden states and missing measurement equal to zero), for any i^{th} sample and j^{th} variable, the estimates of hidden states and missing measurements can be represented in analytical form as follows:

If the output measurement (containing outliers) is observed, $y_{K_{obs}}$:

$$\hat{x}_{K_{obs},i} = \frac{\frac{\hat{r}_{K_{obs},i}\, y_{K_{obs},i}}{\sigma^2_{y_K}} + \sum_{c=1}^{N_{ch}} \frac{\beta_{c,K}\left(\beta_{0,K} + \sum_{p=1}^{N_{pa}-1} Pa_p(x_{K,i})\beta_{0+p,K}\right)}{\sigma^2_{Ch(x_K)|x_K}} + \frac{(\beta_{0,K} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{K,i})\beta_{0+p,K})}{\sigma^2_{x_K|Pa(x_K)}}}{\alpha_{K_{obs}}(x_K)} \tag{3. 63}$$

where denominator $\alpha_{K_{obs}}(x_K)$ is as given in Equation (3. 34) with $j = K$.

If the measurement is missing, $y_{K_{mis}}$:

$$\hat{x}_{K_{mis},i} = \frac{\sum_{c=1}^{N_{Ch}} \frac{\beta_{c,K}(\beta_{0,K} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{K,i})\beta_{0+p,K})}{\sigma^2_{Ch(x_K)|x_K}} + \frac{(\beta_{0,K} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{K,i})\beta_{0+p,K})}{\sigma^2_{x_j|Pa(x_j)}}}{\alpha_{K_{mis}}(x_K)} \tag{3. 64}$$

where the denominator is given as follows:

$$\alpha_{K_{mis}}(x_K) = \left(\frac{1}{\sigma^2_{x_K|Pa(x_K)}} + \sum_{c=1}^{N_{ch}} \frac{\left(\beta^2_{c,K}\right)}{\sigma^2_{Ch_c(x_K)|x_K}}\right) \tag{3. 65}$$

The covariance between hidden states is calculated as discussed in section 0 through Equation (2. 30).

The posterior probability of $r_{K,i}$ is calculated from Bayes' theorem given in (3. 66) and required statistics to compute this posterior probability can be computed from the standard equations given in Equations (3. 41) - (3. 42).

$$p(\boldsymbol{r}|\boldsymbol{Y},\boldsymbol{X},\Theta_r) = \frac{p(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{r},\Theta_r)p(\boldsymbol{r}|\Theta_r)}{p(\boldsymbol{Y})} \tag{3. 66}$$

Note that the rest of the measurements without outliers are computed from the analytical solutions discussed in Equations (2. 27) - (2. 28). Additionally, the statistics related to the posterior probability of the unknown scale variable r_{ij} is obtained from the standard equation given in Equations (3. 41) - (3. 42).

M-step

In the M-step, Q function is maximized w.r.t all the parameters Θ as shown in Equation (2. 34). This time, the variance of outlier-corrupted output measurement variable is updated as:

$$\sigma^2_{y_K,(r+1)} = \frac{E_{\Theta_r}\left(\sum_{i=1}^{O}(r_{K,i})\left(y_{K_{obs},i} - x_{K_{obs},i}\right)^2 + \sum_{i=O+1}^{N}(r_{K,i})\left(y_{K_{mis},i} - x_{K_{mis},i}\right)^2\right)}{N} \tag{3. 67}$$

Rest of the model parameters are updated through the previous expressions in Equations (2. 36) - (2. 39).

3.4 Inference in Bayesian networks

Once parameter learning is carried out, and parameters are estimated, next step is output prediction, using the newly available measurements. Here, Bayesian inference is utilized to predict the corresponding hidden state node of the quality variable, e.g. $X_{j=4}$ in the illustrative example.

Consider at k^{th} sampling instant, measurements of all the input variables are available as $\boldsymbol{y}_k = [y_{1,k} \quad \cdots \quad y_{m-1,k}]^T$, only the measurements of quality variable $y_{j=f,k}$ $(where\ j \in 1 \ldots . m)$ is unavailable, and the hidden state vector of all the process variables to be inferred is given as $\boldsymbol{x}_k = [x_{1,k} \quad \cdots \quad x_{m,k}]^T$. In this case, inference is carried out through Bayesian inference and output can be predicted through analytical solutions given in Equation (2. 73) and Equation (2. 74).

On the other hand, if one of the input variables is contaminated with outliers, the analytical solution for this variable. $j = l$, will be of the following form. The rest of the hidden state expressions will remain same as discussed in Section 2.4.

$$\hat{x}_{j=l,k} = \frac{\frac{y_{j,k}\hat{r}_{j,k}}{\sigma_{y_j}^2} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,j}\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_{j,k})\beta_{0+p,j}\right)}{\sigma_{Ch(x_j)|x_j}^2} + \frac{\left(\beta_{0,j} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{j,k})\beta_{0+p,j}\right)}{\sigma_{x_j|Pa(x_j)}^2}}{\alpha\ (x_j)} \tag{3. 68}$$

where

$$\alpha\ (x_j) = \left(\frac{\hat{r}_{j,k}}{\sigma_{y_j}^2} + \frac{1}{\sigma_{x_j|Pa(x_j)}^2} + \sum_{c=1}^{N_{Ch}} \frac{\left(\beta_{c,j}^2\right)}{\sigma_{Ch_c(x_j)|x_j}^2} \right) \tag{3. 69}$$

Thus, the developed soft sensor can give predictions of the desired hidden states.

3.5 Simulation and industrial application

The performance of the proposed RMR-BN-SS is demonstrated on similar benchmark[41] simulation case study and industrial data as discussed in Section 2.5. Here, following two scenarios are considered

i. Multi-rate data with **output** outliers
ii. Multi-rate data with **input** outliers

In the simulation studies, total of 3000 samples were generated, of which 2200 was used in parameter learning step. To compare the efficacy of the proposed robust soft sensor, average root mean squared error (ARMSE) is computed through Equation (2.75), based on 10 realizations, with 100 data samples for each realization. For industrial case study, all the real-time predictions are down-sampled according to the sampling rate of available lab data to compute RMSE (Equation (2.76)) and correlation coefficient (Equation (2.77)).

3.5.1 Flow Network

Bayesian network structure of the flow network problem in Figure 2. 5 of Chapter 2 is considered in this sub-section. Detailed description of the system and noise variances chosen for data generation can be found in section 2.5.1.

Output outliers

In this simulation study, multi-rate lab measurement Y_4 is assumed to be corrupted with different percentages of outliers i.e. 3%, 5%, 8% and 10%. Figure 3. 1 shows comparison of the proposed robust BN soft sensor with the conventional robust ordinary least squares (ROLS) soft sensor predictions for the scenario when training data is corrupted with 3% outliers. From this figure, it can be seen that predictions of RMR-BN-SS are closer to the true value compared to ROLS predictions. From ARMSE values reported in Table 3. 1, it is evident that from different percentage of outliers considered in the training data, proposed approach has significant improvement in ARMSE values compared to the popular conventional approach. Further, from Table 3. 7- Table 3. 9 (refer to Appendix for Chapter 3), all the noise statistics and parameters are able to converge closer to the actual values.

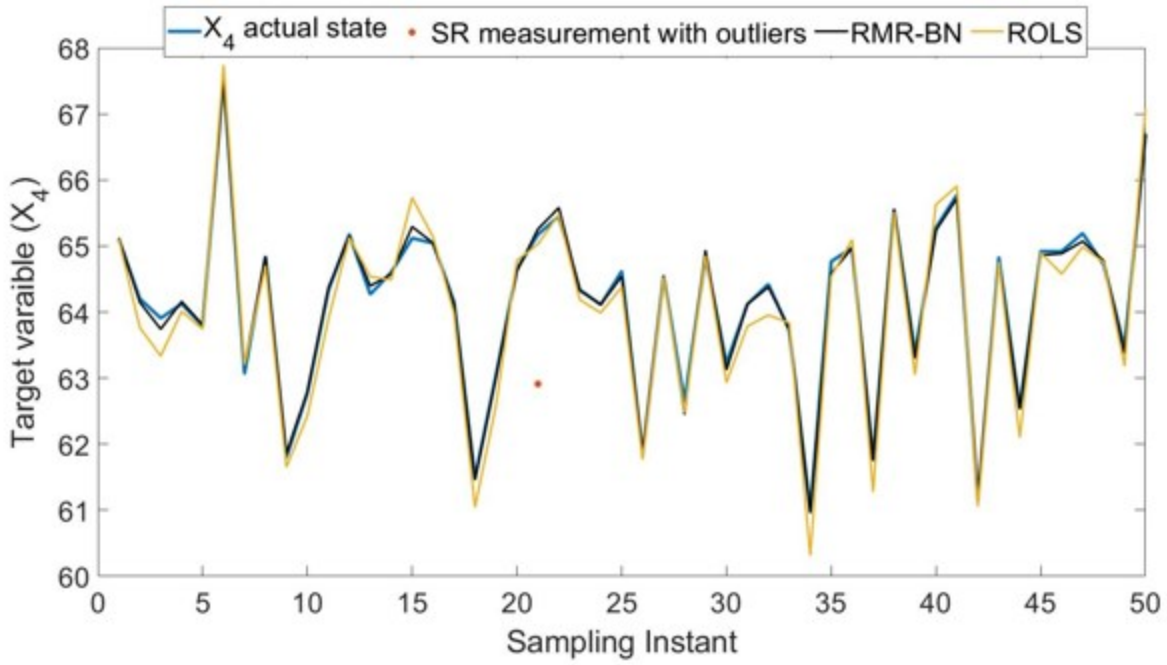

Figure 3. 1: Comparison of t robust soft sensor performances for 3% output outliers

Percentage of outliers	3%	5%	8%	10%
ARMSE RMR-BN	0.0848	0.0991	0.0842	0.0843
ARMSE ROLS	0.3035	0.3237	2.3979	1.7515

Table 3. 1: ARMSE of different approaches for increased output outliers

Input outliers

In this scenario, measurements of input variable Y_2 is assumed to contain different percentages of outliers i.e. 10%, 15%, 20% and 25%. Figure 3. 2 shows comparison of different robust soft sensor predictions for 10% outliers in the training data. From this figure, it is evident that the proposed approach has better performance compared to the conventional ROLS approach. Table 3. 2 reports ARMSE values, where RMR-BN-SS has lower ARMSE value compared to ROLS approach for different percentages of outliers considered in the data.

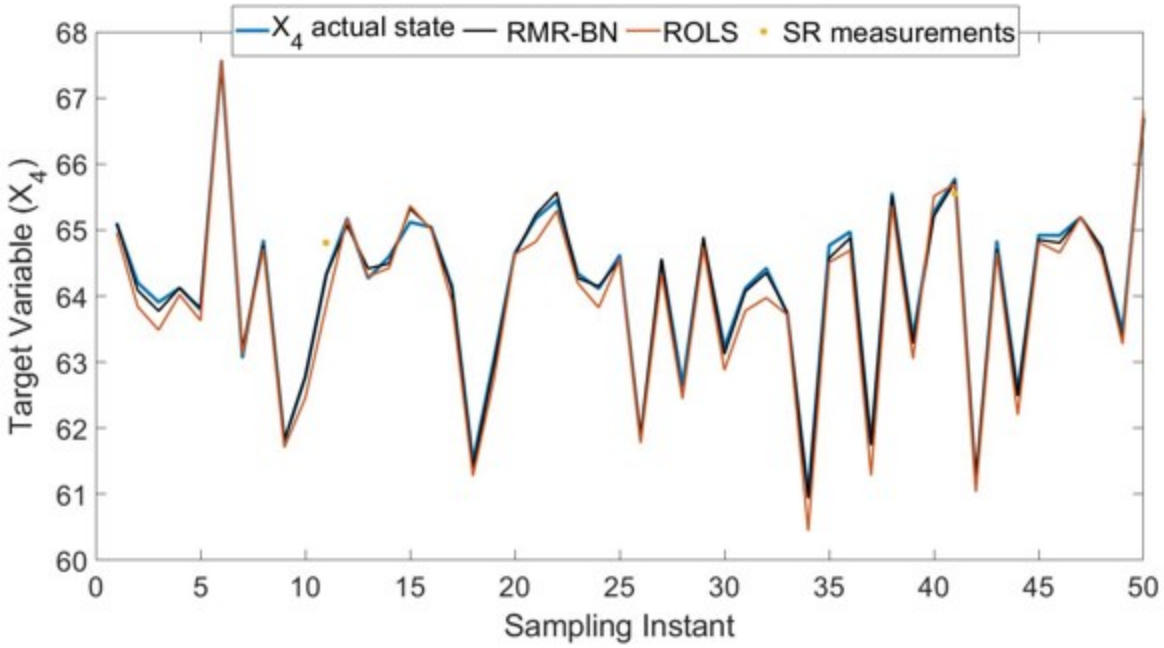

Figure 3. 2: Comparison of robust soft-sensor predictions for 10% input outliers

Percentage of outliers	10%	15%	20%	25%
ARMSE RMR-BN	0.1020	0.1099	0.1153	0.1304
ARMSE ROLS	0.2685	0.2324	0.2331	0.2430

Table 3. 2: ARMSE of different approaches for increased input outlier

3.5.2 Industrial case study

Industrial process considered in Section 2.5.2 is further analyzed in this section for the scenario when training data is corrupted with outliers. Two Bayesian network structures discussed in Chapter 2, namely two-layered and multi-layered are utilized for developing RMR-BN-SS and the results are compared in the subsequent section.

Output outliers

In this scenario, measurements of multi-rate output variable Y_8 are assumed to contains 3% outliers, where these outliers were artificially added to the true lab data. Under this assumption RMR-BN-SS is developed for both two-layered and multi-layered structures and the performance is compared to OLS and ROLS approaches. It can be observed from Figure 3. 3 that the predictions of proposed robust soft sensor are closer to the true lab estimates compared to ROLS and OLS approaches. Further, from Table 3. 3 & Table 3. 4, RMR-BN-SS has better correlation with the lab data and smaller RMSE value compared to the ROLS approach. In addition to that, between the different BN structures, multi-layered RMR-BN-SS is able to give the best result by capturing the process dynamics well.

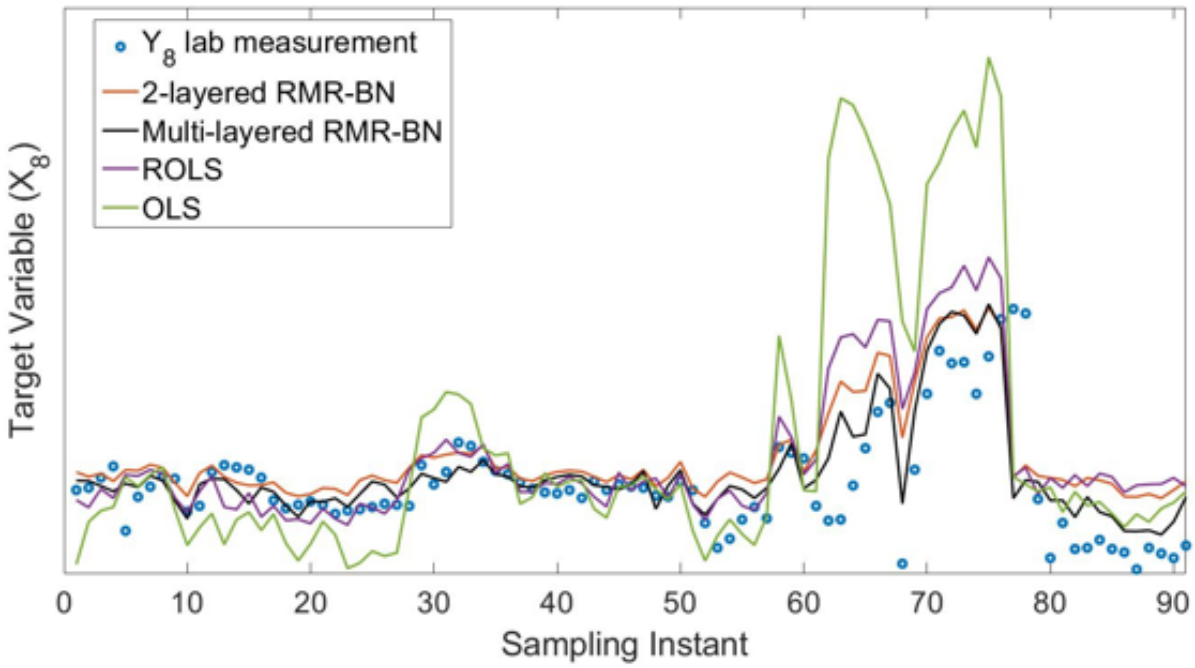

Figure 3. 3: Comparison of robust MR-BN soft sensor

Approach	Correlation
RMR-BN (**multi-layered structure**)	**0.7432**
RMR-BN (**2-layered structure**)	0.6801
ROLS	0.6208
OLS	0.5797

Table 3. 3: Correlation coefficient for different approaches

Approach	RMSE
RMR-BN (**multi-layered structure**)	**3.4820**
RMR-BN (**2-layered structure**)	4.3724
ROLS	5.2506
OLS	10.5608

Table 3. 4: RMSE of different soft sensors

Input outliers

To account for outliers in the input variable, in this sub-section it is assumed that measurements of input variable Y_5 are corrupted with different percentages of outliers (i.e. 10%, 15%, 20% and 25%). As mentioned earlier these outliers were artificially introduced into the sensor measurements. Performance of the proposed approach was compared with the ROLS approach and the graphical result is presented in Figure 3. 4 (10% outliers in training data). From this figure, predictions of proposed RMR-BN-SS are accurate compared to the conventional ROLS approach. One can observe superior performance of RMR-BN-SS at different percentages of

input outliers from Table 3. 5 and Table 3. 6. Further, between the two different BN structures, multi-layered RMR-BN-SS has the best correlation and captured the lab-measurement trend accurately.

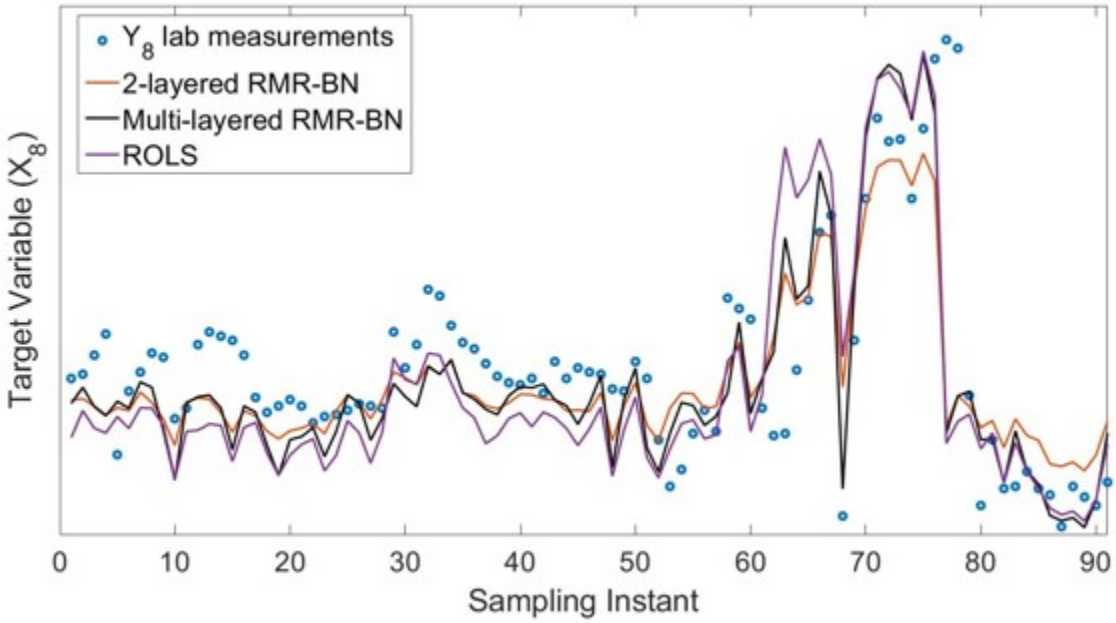

Figure 3. 4: Comparison of robust MR-BN soft sensor for 10% input outliers

Percentage of Outliers	Correlation (**2-layered structure**) RMR-BN	Correlation (**multi-layered structure**) RMR-BN	Correlation ROLS
3%	0.7154	0.7463	0.6783
5%	0.7154	0.7463	0.6758
10%	0.7148	0.7464	0.6749
15%	0.7145	0.7463	0.6757

Table 3. 5: Correlation coefficient values of different approaches

Percentage of Outliers	RMSE (**2-layered structure**) RMR-BN	RMSE (**multi-layered structure**) RMR-BN	RMSE ROLS
3%	3.4420	3.5494	4.2154
5%	3.4429	3.5493	4.2467
10%	3.4680	3.5491	4.2546
15%	3.4841	3.5504	4.2558

Table 3. 6: RMSE of different soft sensors

3.6 Conclusion

BN based soft sensors proposed in the literature do not account for presence of outliers in the measurements while carrying out parameter learning. Not accounting to outliers during model development stage can significantly affect the performance of model predictions. Thus, this work accounts for developing a robust BN soft-sensor, which is insensitive to outliers in the measurements. Since BNs are probabilistic in nature, student's-t distribution, which is a popular probabilistic approach for modeling outliers, is chosen in this work for developing a robust BN soft sensor. The proposed soft sensor is able to handle both missing data and down-sampled data. The performance of proposed RMR-BN SS is demonstrated on benchmark simulation and industrial data sets. Through simulation studies and industrial application, it is evident that it is evident that the proposed approach is able to mitigate the effect of different percentages of outliers contained in input and output, while handling multi-rate data. Further from simulation and industrial application results, one can observe that incorporation of prior process knowledge (i.e. process model or process flow-sheet) in constructing BN structure can significantly improve

soft sensor performance compared to the conventional approaches. In the absence of prior process knowledge, it has been shown that robust soft sensor developed through two-layered BN structure has improved performance compared to the conventional approaches. Thus, to conclude performance of BN soft sensors depend on the chosen BN structure. Soft sensor based on optimal BN structure will have much better prediction performance compared to the popular conventional approaches.

Chapter 4

Adaptive Multi-Rate Bayesian Network Soft Sensor Development

4.1 Introduction

BN based soft sensor development is a multi-stage process, where, initial step is to develop soft sensor model and then to test the model performance off-line. Once the off-line validation of model performance is satisfactory, the model performance is further tested online by implementing in open-loop real time DCS platform. Finally, upon satisfactory performance in open-loop real time DCS platform, soft-sensor predictions are used for closed loop control and process monitoring. Chapter 2 and Chapter 3 addresses common data pre-processing issues (which occur at offline soft-sensor development stage) such as multi-rate noisy data and outliers. Current chapter focuses on addressing issues concerning online soft sensor implementation stage i.e. process and sensor drift.

Due to time dependency or drifting nature of process, i.e. where the process may slowly drift away from its initial operating conditions, soft-sensor validated in offline using the data generated from initial operating conditions may give biased predictions when implemented online. The main reasons for drift in the data could be due to a sensor drift, a process drift or their combination. Drift in the sensor can be due to sensor malfunction, miscalibration of lab equipment or any human error in collecting and recording lab measurements. Process drift on the

other hand can be due to fouling of heat exchangers, catalyst deactivation or process feed quality change and interplay of other important factors. Therefore, this chapter aims to develop adaptive multi-rate BN soft sensor (AMR-BN-SS), which accounts for drift in the data and thereby gives unbiased model predictions.

In the literature, adaptive methods are well studied. Popular adaptive data-driven approaches are generally categorized under the following three umbrella terms: (1) moving windows techniques, (2) recursive approaches [61] and (3) classical bias update approach [62, 63, 64].

In machine learning literature, one of the research branch developing adaptive data-driven techniques is known as concept drift theory [61]. To deal with drifts, first drift has to be detected through symptoms, such as degrading model performance. Once drift is detected, then it is handled [65] using either through instance selection (moving window techniques) [66], instance weighting (recursive adaptation techniques [67] or ensemble methods [68]. Examples of existing moving window based adaptive approaches include: block-wise and sample-wise moving widow[69] techniques. In the moving window technique, the model is trained using most recent batch of data, where the size of data depends on pre-determined window size. User can chose to train the model as soon as a new sample point enters the window and oldest one gets deleted (sample-wise) [70] or after accumulating a certain number of data points (block-wise approach [71]). Sample-wise recalculation of model can be an effective approach and further can be combined with any existing soft sensor models, such as ANN, PCR or PLS. Regardless of which training method is used, both the approaches require to train the model frequently for data of chosen window size, which could be computationally intensive. Moreover, successful parameter estimation depends on the choice of window and step sizes (adaptation intervals between the

updates) [61]. If these two critical parameters are chosen inappropriately, performance of the developed soft sensors can be poor.

On the other hand, recursive adaptation method is other popular approach to address drift in process data. In this approach, the old model predictions in combination with the new input measurements are used to retrain the model, where the amount of information carried from the past to the present is controlled through a forgetting factor λ. This method is often used in adaptive soft-sensors developed through ordinary least squares [71], PCA [72] and PLS [73] approaches. Drawback of this approach is that there is no suitable method to select the forgetting factor for adaptation purposes. Since role of λ is to determine the rate of forgetting old information (the speed of the temporal decay of the samples [61]), it is critical to select this variable for accurately capturing true drift. If this variable is not selected accurately, the prediction performance of the developed soft sensor will degrade over a period.

Lastly, due to its light computational load and ease of implementation, bias update method is widely used to adapt data driven soft sensors to account for drift in the data. Whenever a new lab measurement is available, soft-sensor predictions are corrected by adding a bias term, where bias is computed as difference between soft-sensor prediction and its actual lab measurement. Even though this adaptation method is very common in the industry, due to slow-rate availability of the quality variable i.e. maybe once or twice in a day, bias update is performed only when measurement of the quality variable is available. In the absence of measurement of quality variable, calculated previous bias is used for correcting soft-sensor predictions until a new lab sample is available, which could lead to inaccurate soft-sensor predictions. Thus, for practical and efficient adaptation method, issues of process drift and sensor drift are tackled separately in this chapter.

Since BNs are probabilistic graphical models, process drift is assumed to be a random variable and is modeled through a random walk model. Thus, with the static model representing relation between hidden variables and a dynamic model representing drift in the process, Bayesian inference is performed for carrying out predictions of the quality variable. Through this approach, soft-sensor predictions are corrected at every time instant by simultaneously estimating hidden states and drifting random variable. While sensor drift, which is a slow drift in the measurements of quality variable, is modeled through colored noise. Further, these proposed approaches do not require re-estimation of all the model parameters at every time instant, unlike certain moving window based adaptive models.

4.2 Adaptive MR-BN soft sensor development

As discussed in previous two chapters, the proposed AMR-BN-SS development includes following three main steps: (1) Construction of Bayesian network structure, (2) Parameter learning and (3) Inference. The BN structure construction and parameter learning steps demonstrated in Section 2.3.1 and Section 2.3.2 respectively are considered. However, in this chapter, the modeling assumptions of parameter learning step will be changed to accommodate the drift introduced in the validation data set. The details are discussed in the following sections.

4.2.1 Modeling assumptions for adaptive parameter learning

The modeling assumptions considered in Section 2.2 are utilized in parameter learning step of this chapter. The measurement model between measurement node (Y) and hidden node (X) of j^{th} variable are given as in Equation (2.2) and between the hidden states, the model is linear Gaussian and as given in Equation (2.5).

4.2.2 Construction of Bayesian network structure

If the drift is caused by a process drift, parameter learning and output prediction steps have to be formulated under two different BN structures. The static BN structure constructed for the parameter learning step, for example BN structure shown in Figure 2. 5 for the flow-network problem, has to be converted into a dynamic BN structure by introducing additional time-dependent variable. However, if the reason of drift is due to sensor problem, the same static BN structure can be considered to carry out the two steps.

4.2.3 Parameter learning of adaptive MR-BN soft sensor

Parameter learning for AMR-BN-SS is similar to the parameter learning for MR-BN-SS development, discussed in Section 2.3.4. As it was concluded, parameters can be estimated through the analytical solutions given in Equations (2. 36) and in Equation (2. 65).

4.2.4 Inference in adaptive MR-BN soft sensor

The previous two chapters discuss extensively on the off-line development of BN based soft sensors. In this chapter, the framework of proposed multi-rate BN based soft-sensor is extended to account for drift in the process data. This is achieved by modifying the modeling assumptions in the output inference step, depending on the type of drift. In the case of process drift, true state as well as its corresponding measurement are assumed to be drifting. On the other hand, in the case of a sensor drift, only the measurement is assumed to be drifting, while the true state is assumed to be at the initial nominal conditions. Therefore, in the following sections, two separate formulations for inference step are discussed for addressing process and sensor drift.

Inference for process drift

In this case, the target state, $x_{j=f}$, is assumed to be drifting and its measurement $y_{j=f}$ is slow-rate and is assumed to be available without any delay. Drift in the process is assumed to be a random variable v and is modeled through a random walk model as follows:

$$v_{j,k} = v_{j,k-1} + \mathrm{e_{j,k}} \tag{4. 1}$$

where

$$\mathrm{e_{j,k}} \sim N(0, \sigma_{v_k}) \tag{4. 2}$$

Thus, the linear model assumed between the j^{th} measurement y_j and its corresponding hidden variable x_j, given in Equation (2.5), is modified as given below.

$$y_{j,k} = x_{j,k} + \mathrm{w_{j,k}} \tag{4. 3}$$

where

$$\mathrm{w_{j,k}} \sim N(0, \sigma_{w_k}) \tag{4. 4}$$

By assuming that the drift is modeled through random variable (*v)* effecting the hidden quality variable, linear conditional Gaussian distribution model of quality variable is given as

$$p\left(x_f \middle| Pa(x_f)\right) \sim N(v_f + \beta_{0,f} + \sum_{p=1}^{N_{Pa}} \beta_{0+p,j} Pa(x_f)_p, \sigma_{X_f}^2) \tag{4. 5}$$

With the drifting dynamic model given by Equation (4. 1) and static model Equations (4. 3), the BN structure for inference step can be presented as shown in Figure 4. 1. In this figure, common cause BN structure shown in Figure 2. 1 is extended by introducing time-dependent variable v and graphed through time.

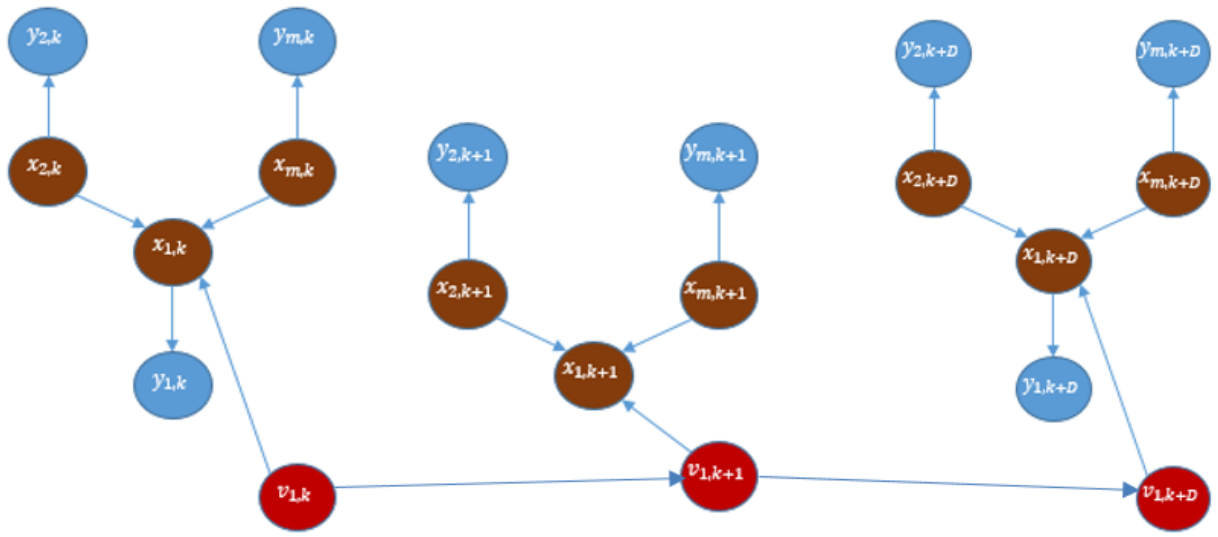

Figure 4. 1: Dynamic Bayesian network structure

From this figure, it can be observed that by considering drift v as a time dependent random variable, BN structure is no longer static and can be represented in dynamic form. Here, multi-rate dynamic BN structure for the process variables $X = [X_1, X_2, \ldots, X_m]$ is illustrated across three sampling instances, namely, k, $k+1$ and $k+D$, where k and $k+D$ are the sampling instances at which lab measurements are available, while $k+1$ is the sampling instant at which measurement of quality variable (y_1) is not available. D corresponds to the time gap between availability of two lab samples. For simplicity, D is assumed to be equal to 1 in this figure although in practice, this wait is much longer i.e. may be 12 hr or 24 hr. The measurements of lab values (slow rate measurements) therefore can be referred as $y_{s,k}$ and $y_{s,k+D}$.

From Bayes' theorem at sampling instant k, posterior probability of the hidden variables can be obtained by Equation (4. 6), where γ equals $p(\mathbf{y}_k)^{-1}$ and is a normalizing constant.

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) = \gamma p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{I}) \tag{4. 6}$$

For Equation (4. *6*), hidden state vector is $\boldsymbol{x_k} = [\boldsymbol{x_{j=1,k}}, \dots, \boldsymbol{x_{j=m,k}}, \boldsymbol{v_{j,k}}]$ and measurement vector is $\boldsymbol{y_k} = [\boldsymbol{y_{j=1,k}}, \dots, \boldsymbol{y_{j=m,k}}]$. For notation clarity, measurement vector can be written as in Equation (4. *7*), consisting of slow-rate measurement denoted as $j = s$ and all the other fast rate inputs denoted as $j = f$.

$$\boldsymbol{y_k} = [\boldsymbol{y_{s,k}}, \boldsymbol{y_{f,k}}] \tag{4. 7}$$

Here, $\boldsymbol{y_{s,k}}$ is the slow-rate lab values. The posterior probability distribution function (given by Equation (4.6)) can be further decomposed resulting in the following expression:

$$p(\boldsymbol{x}_k, v_k|\boldsymbol{y}_k) = \gamma p\big(y_{s,\triangle k}\big|x_{s,\triangle k}\big)p\big(x_{s,k}\big|Pa\big(x_{s,k}\big), v_{s,k}\big)$$
$$\prod_{j=1, j\neq s}^{m-1} p\big(y_{f,k}\big|x_{f,k}\big) \prod_{j=1, j\neq s}^{m-(c+1)} P\left(x_{f,k}\middle|Pa\big(x_{f,k}\big)\right)p\big(v_{f,k}|v_{f,k-1}\big) \tag{4. 8}$$

In this Equation (4. 8), $\triangle k \in \{k, k+D, k+2D \dots.\}$ represents the sampling interval at which measurements of quality variable are available, and $k-1$ is the immediate past sample of the fast rate measurements. Thus, estimates of the hidden state variables at the k^{th} sampling instant can be obtained by minimizing the negative logarithmic of posterior distribution function i.e.

$$\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{v}}_k = \min_{\boldsymbol{x}_k, v_k} -[\log p(\boldsymbol{x}_k, v_k|\boldsymbol{y}_k)]$$

$$= \min_{\boldsymbol{x}_k, v_k} -\Bigg[\log p\left(y_{s,\triangle k}\middle|x_{s,\triangle k}\right) + \log p\left(x_{s,k}\middle|Pa\left(x_{s,k}\right), v_{s,k}\right)$$

$$+ \sum_{j=1, j\neq s}^{m-1} \log p\left(y_{f,k}\middle|x_{f,k}\right) + \sum_{j=1, j\neq s}^{m-(c+1)} \log p\left(x_{f,k}\middle|Pa\left(x_{f,k}\right)\right) \tag{4. 9}$$

$$+ \log p(v_{f,k}|v_{f,k-1}) + \log(\gamma)\Bigg]$$

where the prior probability distribution of the source node is assumed to be uniform and is omitted from Equation (4. 9). Moreover, as per Assumption 2.3 and Equation (2. 41), the lab measurements $\boldsymbol{y_s}$ is assumed to contain an observed and missing counterparts as $\boldsymbol{y_s} = [\boldsymbol{y_{s_{obs}}}\ \boldsymbol{y_{s_{mis}}}]$. Therefore, Equation (4. 9) can be further expanded as follows:

$$\min_{\boldsymbol{x}_k, v_k} -[\log p(\boldsymbol{x}_k, v_k|\boldsymbol{y}_k)] = \hat{\boldsymbol{x}}_k, \hat{\boldsymbol{v}}_k$$

$$= \min_{x_k, v_k} -\Bigg[\frac{\left(y_{s_{obs},\triangle k} - x_{s_{obs},\triangle k}\right)^2}{2\sigma_{y_s}^2} + \frac{\left(x_{s,k} - v_{s,k} - \beta_{0,s} - \sum_{p=1}^{N_{Pa}} Pa\left(x_{s,k}\right)\beta_{0+p,s}\right)^2}{2\sigma_{x_s}^2}$$

$$+ \sum_{j=1, j\neq s}^{m-1} \frac{\left(y_{j,k} - x_{j,k}\right)^2}{2\sigma_{y_j}^2} + \sum_{j=1, j\neq s}^{m-(c+1)} \frac{\left(x_{j,k} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}} Pa\left(x_{j,k}\right)\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2} \tag{4. 10}$$

$$+ \frac{\left(v_{f,k} - v_{f,k-1}\right)^2}{\sigma_{v_f}^2}\Bigg]$$

In this Equation (4.10), the past value of the newly introduced hidden state, $v_{s,k-1}$, can be seen as an arrival cost-value, carrying all the past information of the drift to the current time instance, so the drifting state can be estimated accurately at present. Thus, analytical expressions to compute hidden state value of the slow-rate target node $x_{s,k}$ is different, depending on the availability of lab data. Using first order optimality conditions (first order derivatives w.r.t. hidden states and drifting variable to be equal to zero), estimates of j^{th} variable at k^{th} sampling instant are obtained through the following analytical expressions given in Equation (4. 11) & (4. *12*). If the quality variable $y_{s,}$ is observed i.e. $\triangle k \in \{k, k+D, k+2D \ldots.\}$, fast rate estimates of the quality variable is given by the following analytical expression

$$\hat{x}_{s_{obs},k} = \frac{\frac{y_{s_{obs},\triangle k}}{\sigma^2_{y_s}} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,s}\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_s)\beta_{0+p,s}\right)}{\sigma^2_{Ch(x_s)|x_s}} + \frac{\left(\boldsymbol{v}_{s,k} + \beta_{0,s} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{s,k})\beta_{0+p,s}\right)}{\sigma^2_{x_s|Pa(x_s)}}}{\alpha_{s_{obs}}} \quad (4.\ 11)$$

If the quality variable y_s is missing i.e. no lab sample is available, fast rate estimates of the quality variable can be obtained as follows

$$\hat{x}_{s_{mis},k} = \frac{\sum_{c=1}^{N_{Ch}} \frac{\beta_{c,s}\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_f)\beta_{0+p,s}\right)}{\sigma^2_{Ch(x_s)|x_s}} + \frac{\left(\boldsymbol{v}_{s,k} + \beta_{0,s} + \sum_{p=1}^{N_{Pa}} Pa_p(x_{s,k})\beta_{0+p,s}\right)}{\sigma^2_{x_s|Pa(x_s)}}}{\alpha_{s_{mis}}} \quad (4.\ 12)$$

Estimates of drifting random variable can be obtained from the following expression

$$\boldsymbol{v}_{s,k} = \frac{\frac{v_{s,k-1}}{\sigma^2_{v_s}} + \frac{\left(\hat{x}_{s,k} - \beta_{0,s} - \sum_{p=1}^{N_{Pa}} Pa_p(x_{s,k})\beta_{0+p,s}\right)}{\sigma^2_{x_s|Pa(x_s)}}}{\alpha_{v_s}} \quad (4.\ 13)$$

where

$$\alpha_{s_{obs}} = \left(\frac{1}{\sigma_{y_s}^2} + \frac{1}{\sigma_{x_s|Pa(x_s)}^2} + \sum_{c=1}^{N_{Ch}} \frac{\left(\beta_{c,s}^2\right)}{\sigma_{Ch_c(x_s)|x_s}^2} \right) \tag{4. 14}$$

$$\alpha_{s_{mis}} = \left(\frac{1}{\sigma_{x_s|Pa(x_s)}^2} + \sum_{c=1}^{N_{Ch}} \frac{\left(\beta_{c,s}^2\right)}{\sigma_{Ch_c(x_s)|x_s}^2} \right) \tag{4. 15}$$

$$\alpha_{v_s} = \left(\frac{1}{\sigma_{v_s}^2} + \frac{1}{\sigma_{x_s|Pa(x_s)}^2} \right) \tag{4. 16}$$

The rest of the drift-free hidden states with a fast-rate measurements will be computed through the analytical solutions in Equations (2. 73) & (2. 74).

Remark 4.1. The variance $\sigma_{v_s}^2$ of dynamic node can either be estimated from the validation data set or can be tuned. In this work, optimal value of $\sigma_{v_s}^2$ is chosen by trial and error approach.

Inference for sensor drift

Since quality variable sample collection and further laboratory processing involves uncertainty in the final reported value, use of such measurements for bias update may result in inaccurate predictions. Thus, it is vital to account for drift in the lab measurements to improve the output prediction accuracy. For this scenario, measurement of quality variable is assumed to be drifting due to sensor issue, while its true state is assumed to have no effect of drift. Therefore, this section aims at developing a Bayesian inference framework for estimating true hidden quality variable in the presence of drift in the sensor/lab measurements ($\boldsymbol{y_s}$).

The regular measurement model between j^{th} measurement y_j and its corresponding hidden variable x_j ($i.e.$ Equation (2.2)) is now assumed to be corrupted with a colored noise (not a white noise) as given in Equation (4. 17), where k is the current time stamp.

$$y_{s,k} = x_{s,k} + \frac{1}{1-q^{-1}} e_{s_y,k} \tag{4. 17}$$

Upon further algebraic simplification, Equation (4. 17) can be further written as:

$$y_{s,k} = y_{s,k-\triangle k} + x_{s,k} - x_{s,k-\triangle k} + e_{s_y,k} \tag{4. 18}$$

where $e_{s_y,k} \sim N(0, \sigma_{s_y})$

For notation simplicity, Equation (4. 18) can be further expressed as:

$$y_{s,k} = \boldsymbol{H_{s,k}} + e_{s_y,k} \tag{4. 19}$$

where $$\boldsymbol{H_{s,k}} = [x_{s,k} \;\; x_{s,k-\triangle k} \;\; y_{s,k-\triangle k}] \tag{4. 20}$$

Also, the linear Gaussian model between hidden states for slow rate quality variable is expressed as:

$$X_{s,k} = f\left(Pa\left(X_{s,k}\right)\right) + e_{s_x,k} \tag{4. 21}$$

where $e_{s_x,k} \sim N(0, \sigma_{s_x})$

where the error terms, e_{s_y} and e_{s_x} follow Gaussian distributions with zero mean and corresponding variances. Thus, using the updated measurement model of quality variable shown in Equation (4. 20), posterior probability of hidden states can be obtained from Bayes' theorem as follows.

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) = \gamma p\left(y_{s,k}\middle|x_{s,k}, y_{s,k-\triangle k}, x_{s,k-\triangle k}\right) \prod_{j=1, j\neq s}^{m-1} p\left(y_{j,k}\middle|x_{j,k}\right) \prod_{j=1}^{m-c} p\left(x_{j,k}\middle|Pa\left(x_{j,k}\right)\right) \quad (4.\ 22)$$

Thus, estimates of the hidden state variables at the k^{th} sampling instant can be obtained by minimizing the negative logarithmic of posterior distribution function i.e.

$$\begin{aligned}\hat{\boldsymbol{x}}_k &= \min_{\boldsymbol{x}_k} -[\log p(\boldsymbol{x}_k|\boldsymbol{y}_k)] \\ &= \min_{\boldsymbol{x}_k} -\left[\log p\left(y_{s,k}\middle|x_{s,k}, y_{s,k-\triangle k}, x_{s,k-\triangle k}\right)\right. \\ &\left. + \sum_{j=1, j\neq s}^{m-1} \log p\left(y_{j,k}\middle|x_{j,k}\right) + \sum_{j=1}^{m-c} \log p\left(x_{j,k}\middle|Pa\left(x_{j,k}\right)\right) + \log(\gamma)\right]\end{aligned} \quad (4.\ 23)$$

Note: prior probability distribution term i.e. $p\left(x_{c,k}\middle|I\right)$ is assumed to be uniform and omitted from Equation (4. 23). As it was assumed in previous sections, the lab measurements $\boldsymbol{y_s}$ contains observed and missing counterparts and written as:

$$\boldsymbol{y_s} = [\boldsymbol{y_{s_{obs}}}\ \ \boldsymbol{y_{s_{mis}}}] \quad (4.\ 24)$$

Thus, Equation (4. 23) can be further expanded to the following expression,

$$\begin{aligned}\min_{\boldsymbol{x}_k} -[\log p(\boldsymbol{x}_k|\boldsymbol{y}_k)] &= \hat{\boldsymbol{x}}_k \\ &= \min_{x_k} -\left[\frac{\left(y_{s_{obs},\triangle k} - y_{s,k-\triangle k} - x_{s_{obs},\triangle k} + x_{s,k-\triangle k}\right)^2}{2\sigma_{y_s}^2}\right. \\ &\left. + \sum_{j=1, j\neq s}^{m-1} \frac{\left(y_{j,k} - x_{j,k}\right)^2}{2\sigma_{y_j}^2} + \sum_{j=1}^{m-c} \frac{\left(x_{j,k} - \beta_{0,j} - \sum_{p=1}^{N_{Pa}} Pa_p\left(x_{j,k}\right)\beta_{0+p,j}\right)^2}{2\sigma_{x_j}^2}\right]\end{aligned} \quad (4.\ 25)$$

Similar to previous formulation, $\triangle k$ represents the slow-rate sampling interval and $k-\triangle k$ represents the previous slow-rate sampling interval. Thus, using first order optimality (first order derivatives w.r.t. hidden states and missing measurement equal to zero), estimates of j^{th} variable at k^{th} instance are obtained through the following analytical solutions. If the quality variable y_s is observed, $y_{s_{obs}}$:

$$\hat{x}_{s_{obs},k} = \frac{\frac{y_{s_{obs},\triangle k} - y_{s,k-\triangle k} + x_{s,k-\triangle k}}{\sigma_{y_s}^2} + \sum_{c=1}^{N_{Ch}} \frac{\beta_{c,s}\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_s)\beta_{0+p,s}\right)}{\sigma_{Ch(x_s)|x_s}^2} + \frac{\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}} Pa_p(x_s)\beta_{0+p,s}\right)}{\sigma_{x_s|Pa(x_s)}^2}}{\alpha_{s_{obs}}} \tag{4. 26}$$

If the quality variable $y_{s,}$ is missing, $y_{s_{mis}}$:

$$\hat{x}_{s_{mis},k} = \frac{\sum_{c=1}^{N_{Ch}} \frac{\beta_{c,s}\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}-1} Pa_p(x_s)\beta_{0+p,s}\right)}{\sigma_{Ch(x_s)|x_s}^2} + \frac{\left(\beta_{0,s} + \sum_{p=1}^{N_{Pa}} Pa_p(x_s)\beta_{0+p,s}\right)}{\sigma_{x_s|Pa(x_s)}^2}}{\alpha_{s_{mis}}} \tag{4. 27}$$

where the denominators are given in Equation (4. 14) & (4. 15) respectively. Remaining fast rate nodes without any drift can be obtained from the analytical solutions given in Equations (2. 73) & (2. 74).

4.3 Conventional Bias update approach

Adaptation of conventional soft sensors through bias correction is widely used approach. To explain this approach, let's assume that y is our quality variable to be predicted, and the estimated value, using any predictive modeling approach is denoted as $\hat{y}$. Due to process drift, time dependency and nonlinearity of the process, estimated value may contain some degree of uncertainty and can be corrected through bias correction as given in Equation (4. 28).

$$\hat{y} = f(\boldsymbol{x}, \alpha) + \boldsymbol{bias} \tag{4. 28}$$

where $\boldsymbol{x}$ is the input variable vector, α is the estimated parameter set [72] and model structures are assumed to be known. In Equation (4. 28), every time a new measurement y is available, the adaptation technique can be used by updating the $\boldsymbol{bias}$ [64, 74] as given in Equation (4. 29).

$$\boldsymbol{bias} = y - f(\boldsymbol{x}, \alpha) \tag{4. 29}$$

For this particular chapter, predictions of OLS soft sensor model are bias updated through Equation (4. 30). In this Equation, $\hat{y}_{k_{bu}}$ is bias updated OLS output predictions, $\hat{y}_k$ is the OLS estimate before bias update, b_k is current bias, b_{k-1} is bias computed at the previous at which lab samples are available and y_k is slow-rate measurement of quality variable.

$$\begin{aligned} \hat{y}_{k_{bu}} &= \hat{y}_k + b_k \\ b_k &= \alpha(b_{k-1}) + (1-\alpha)(y_k - \hat{y}_k) \end{aligned} \tag{4. 30}$$

The drawback of this approach is that the output predictions are corrected only when lab measurements are available and when the lab measurements are absent, old bias is used to correct current predictions. Additionally, the forgetting factor α needs to be tuned effectively. Tuning α is a critical step, since the output prediction accuracy mainly depends on this parameter. Effectiveness of this approach is further demonstrated through simulation study in section 4.4 of this chapter.

4.4 Simulation study

Performance of the proposed adaptive approach is demonstrated on the same flow-network problem [41] in section 2.5.1. To compare the efficacy of AMR-BN-SS to bias updated OLS, for the simulation studies, average root mean squared error (ARMSE) is computed as given in

Equation (2. 75), based on 10 realizations with 800 samples for each realization. OLS model is bias updated through Equation (4. 30) for fair comparison with the proposed approach, which also utilizes available lab measurements to make better predictions.

4.4.1 Simulation study: flow-network problem

Description of the flow network system and its operating conditions can be referred from section 2.5.1. In this section, given all the input variables, quality variable X_4 will be predicted. Following three scenarios are considered:

i. Multi-rate lab data in the presence of process drift
   a. Process drift generated through random walk model
   b. Process drift generated through sudden changes i.e. step changes
ii. Multi-rate lab data in the presence of sensor drift
   a. Sensor drift generated through slow monotone change

Process drift through random walk model

Initially for carrying out parameter learning, 2200 process samples were generated under steady state conditions without any drift in the quality variable (X_4). Since the problem was formulated through random walk model, drift was generated first through the same model in the 800 samples of validation data set for variable X_4. Figure 4. 2 shows the drifting variable X_4 (in blue) and its corresponding measurement Y_4 (in red). Figure 4. 3 illustrates comparison of different adaptive approaches. Note that for bias updated OLS approach, forgetting factor is chosen to be $\alpha = 0.4$. From Figure 4. 3 it can be observed that the predictions obtained by AMR-BN-SS accurately tracks the drifting quality variable (X_4) compared to bias updated OLS. Also, from ARMSE values shown in Table 4. 1, it is evident that the proposed approach has lower ARMSE. Further,

Table 4. 2 demonstrates the sensitiveness of bias updated OLS approach to the choice of forgetting factor α; thus, efficient tuning of this parameter is essential.

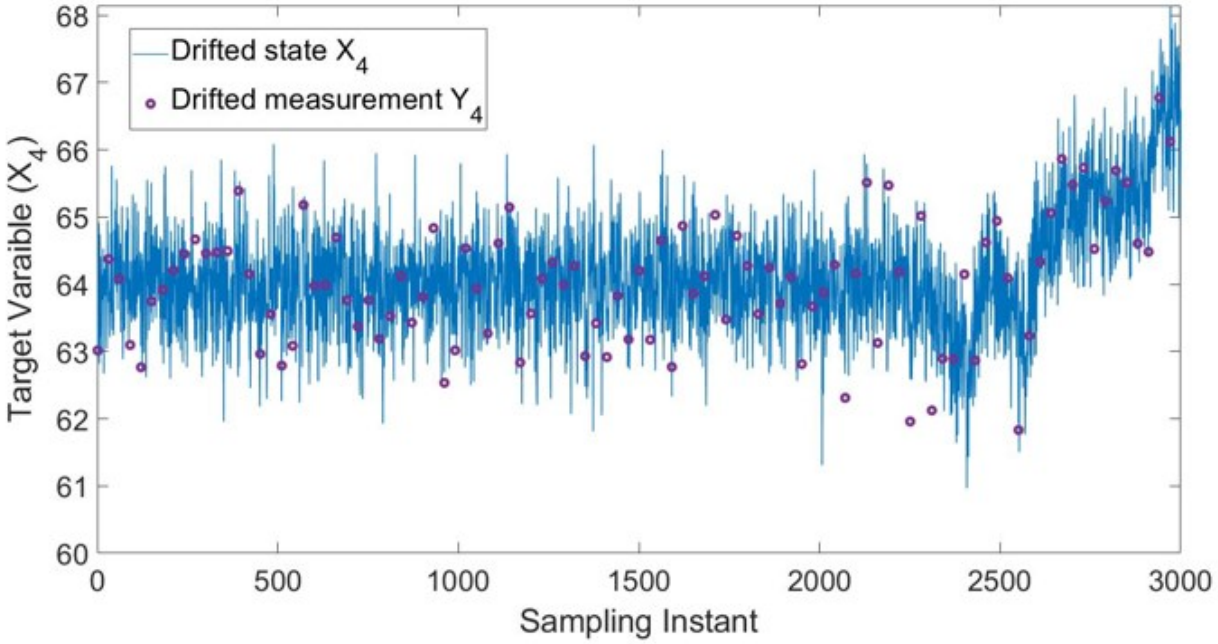

Figure 4. 2: Added process drift

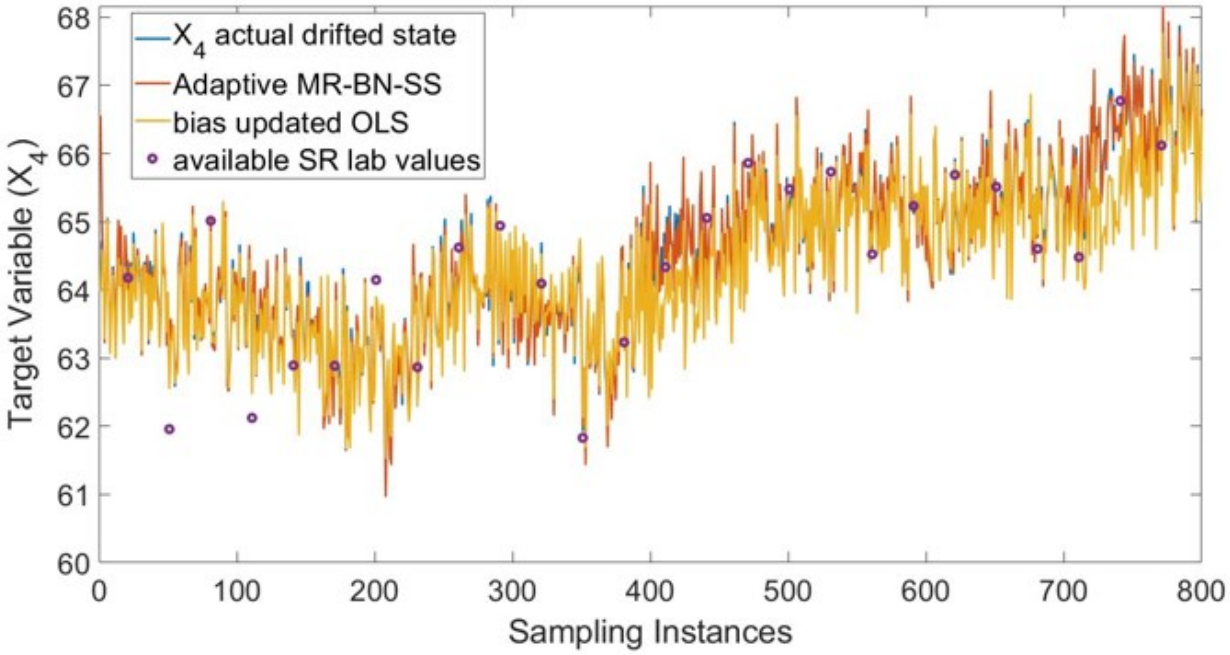

Figure 4. 3: Performance of different soft sensors

Target variable	ARMSE adaptive MR-BN-SS	ARMSE bias updated OLS
X_4	0.1424	0.3725

Table 4. 1: ARMSE values

α	0.0001	0.005	0.01	0.1	0.2	0.3	**0.4**	0.5	0.6	0.99
RMSE of bias updated OLS	0.4520	0.4506	0.4492	0.4252	0.4018	0.3834	**0.3725**	0.3729	0.3906	1.0310

Table 4. 2: ARMSE values of the bias updated OLS with different forgetting factor

Process drift through step changes

In this sub-section, the same 2200 training data set for the previous trial was considered and sudden drift or step changes were introduced in the 800 samples of X_4 validation data set. Figure 4. 4 shows the drifting variable X_4 (in blue) and its corresponding measurement Y_4 (in red). Figure 4. 5 compares performances of different adaptive approaches. In this simulation example, forgetting factor of bias updated OLS approach is $\alpha = 0.5$. From Figure 4. 5, one can see the advantage of the proposed approach, where it is much faster to catch up the sudden changes in the process, compared to the bias updated OLS predictions. This is due to the slow-rate lab measurements for OLS model. OLS predictions are corrected only when new lab measurement arrives. On the other hand, the proposed approach does not only depend on the

slow-rate lab measurements, it also takes the remaining input variables and the covariance of the hidden states into account, which causes the proposed approach to give accurate predictions faster or not long after sudden process change. Figure 4. 6 zoomed into the different adaptive soft sensor performances graph and illustrates the profile over the first step change. As it can be observed from this figure, AMR-BN-SS is adapting to the change faster, compared to the bias-updated OLS, which is taking some time to finally come close to the real value. Also, from ARMSE values reported in Table 4. 3, it is evident that the proposed adaptive approach has lower ARMSE. Further, Table 4. 4 demonstrates the effect of different values of forgetting factor α to the bias updated OLS soft sensor predictions.

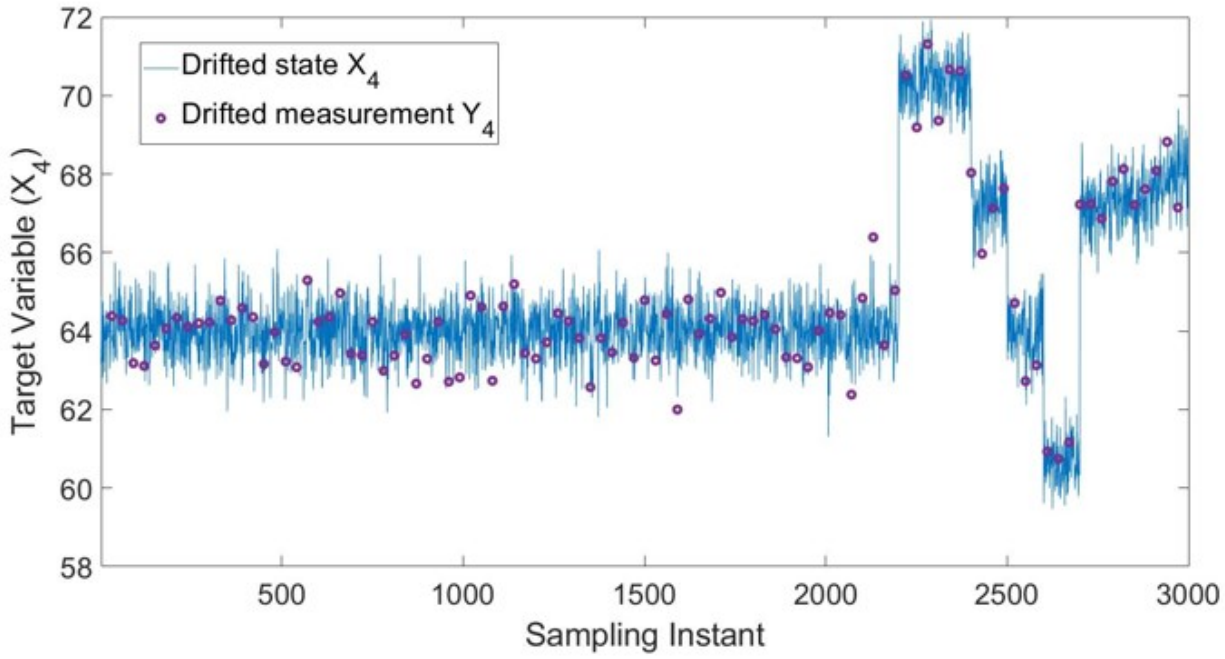

Figure 4. 4: Added process drift profile

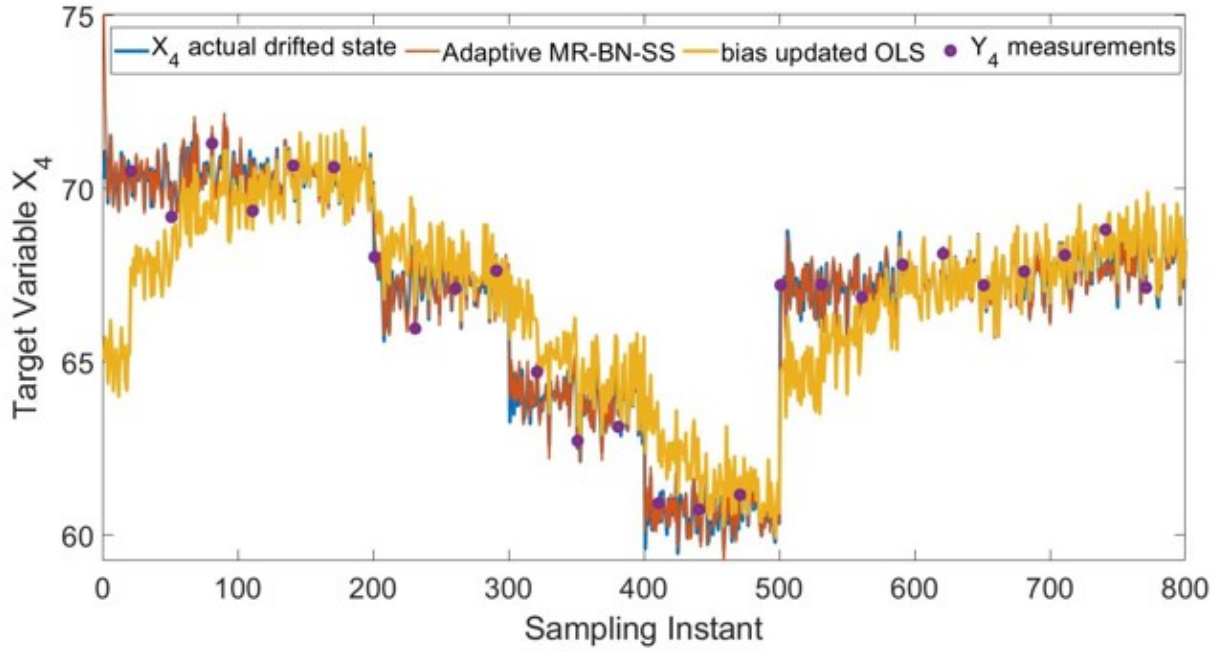

Figure 4. 5: Performance of adaptive MR-BN soft sensor in fast-rate

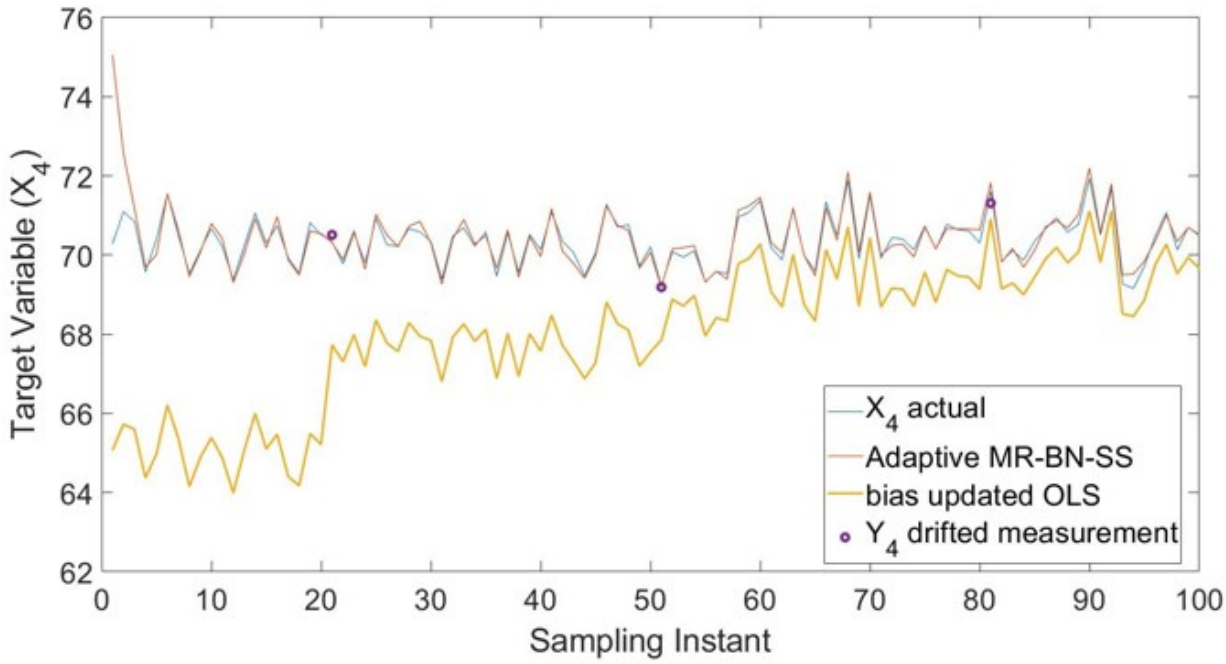

Figure 4. 6: Performance of different soft sensors zoomed

Target variable	ARMSE adaptive MR-BN-SS	ARMSE bias updated OLS
X_4	0.3083	1.3919

Table 4. 3: ARMSE values

α	0.005	**0.01**	0.1	0.2	0.3	0.4	0.5	0.6
RMSE of bias updated OLS	1.0139	**1.0136**	1.0205	1.0586	1.1306	1.2397	1.3919	1.5975

Table 4. 4: ARMSE value estimates of the bias updated OLS with different forgetting factor

Sensor drift

In this scenario total of 2200 samples were generated without drift and artificial slow change or monotone drift shown in Figure 4. 7 is added to the validation data set (samples from 2200 to 3000) of the measurement Y_4. From this figure, one can distinguish between the drifting measurement (in red) and the true drift-free state (in blue). Since it is assumed that drift is only in the measurement, the corresponding hidden state X_4 is within its normal operating range. From Figure 4. 8, the performance comparison of different adaptive modeling approaches, and Table 4. 5 (calculated ARMSE), it is observed that if sensor-drifted measurements are utilized to bias update OLS predictions, it can lead to poor performance. Whereas, with the proposed adaptive approach, sensor drift is able to be captured by the newly introduced colored noise model resulting in accurate output predictions with much lower ARMSE.

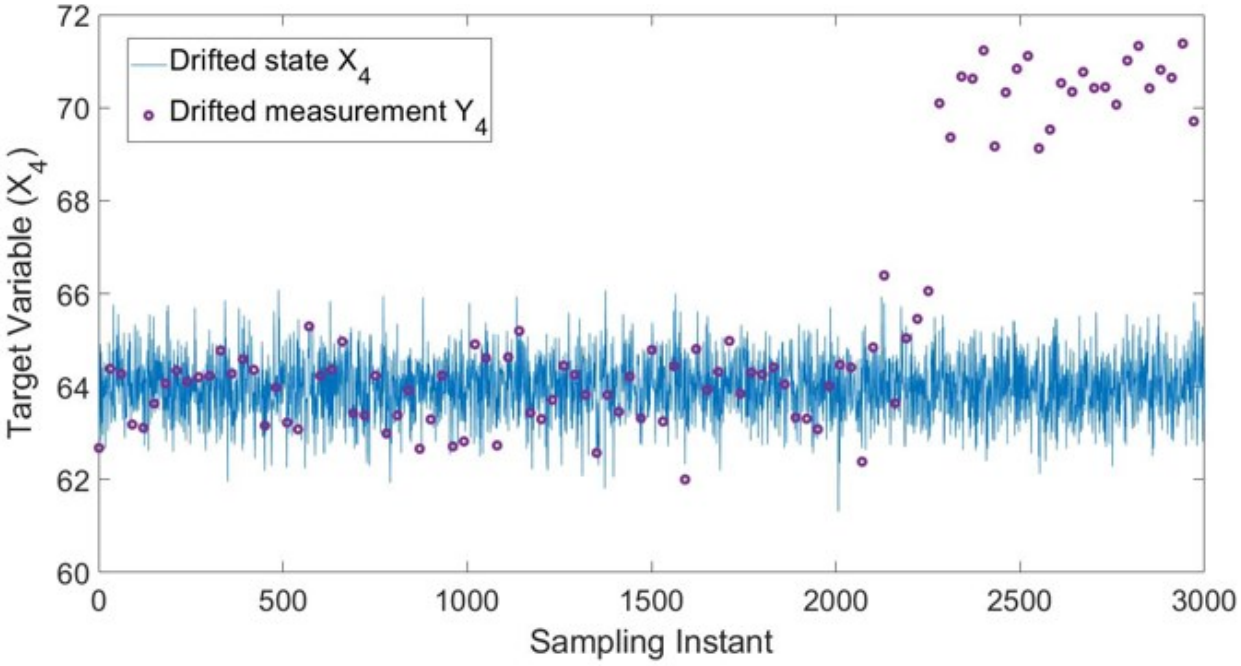

Figure 4. 7: Added sensor drift

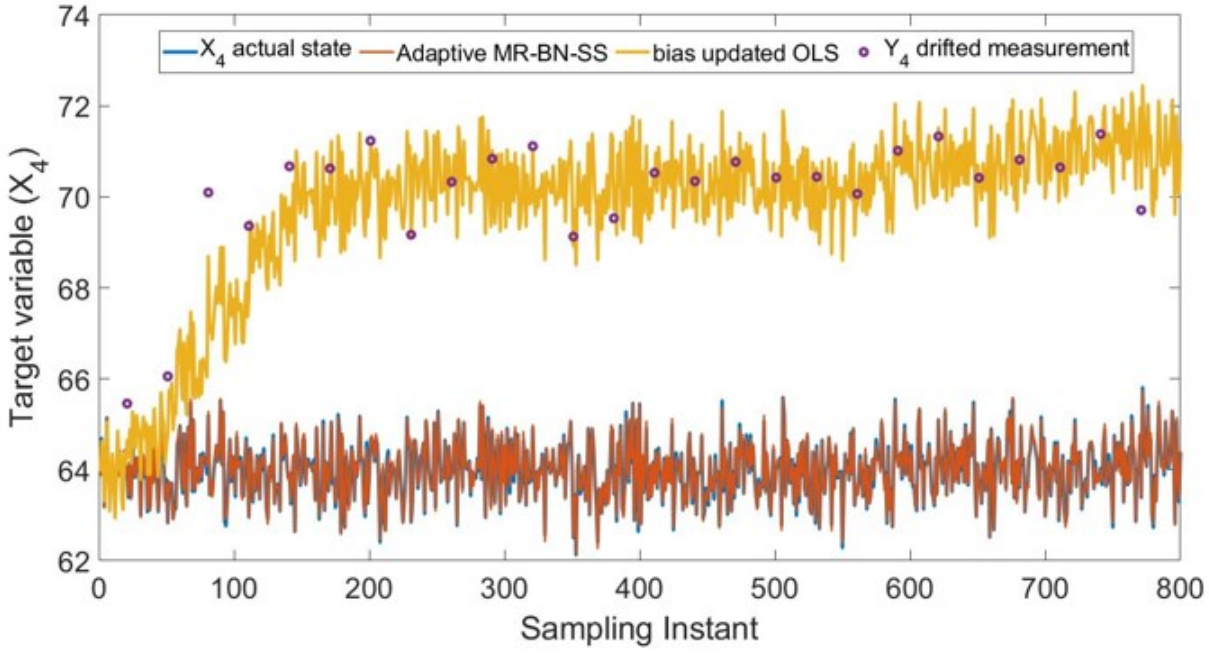

Figure 4. 8: Performance of different soft sensors

Target variables	ARMSE adaptive MR-BN-SS	ARMSE bias updated OLS
X_4	0.2452	2.7951

Table 4. 5: ARMSE values

4.5 Conclusion

Adaptive approaches are necessary when the developed soft sensor is ready for online implementation. This could be due to two reasons, a process drift or a sensor drift. In the literature, moving window based, recursive and classical bias update approaches are utilized to adapt to the drifting process. However, some of these methods are computationally intensive or the bias update approach for OLS is only updated when there is a new lab measurement available. Through the simulation studies, it was shown the accurate selection of forgetting factor is essential for this approach. To overcome these challenges, two separate formulations are proposed in this chapter, assuming that the cause of drift is already known. Efficacy of the proposed approaches are demonstrated through simulations and proven to be performing better than the conventional bias updated OLS under different types of drift. It will be interesting to further apply the process drift formulation on actual industrial application and test efficacy of the proposed approach on industrial data.

Chapter 5

Conclusions

5.1 Summary of thesis research

This thesis developed a new framework for soft sensing based on popular probabilistic graphical model, namely Bayesian networks, which aims to solve common data pre-processing issues as well as problems that arise during soft sensor online implementation stage and motivated to deliver a complete solution. Summary and conclusions of the three objectives considered in the work are listed in detail as follows:

- **Multi-rate Bayesian network soft sensor for noisy data (Chapter 2)**: This Chapter addresses common issues that are faced during off-line soft sensor development stage such as noisy, multi-rate data and completely missing data. A Bayesian network soft sensor problem is formulated for down-sampled and multi-rate data, and analytical solutions are provided. In addition to this, we investigated the possibilities of developing different Bayesian network structures for the same batch of data. From the analysis of simulation and industrial data set results, we have observed that multi-layered BN structure based soft sensor are able to perform the best due to incorporation of prior process knowledge. Meanwhile, BN based soft sensors can be constructed based on a two-layered BN structure. In this case, no prior process knowledge is necessary and utilization of this approach will be as simple as constructing OLS soft sensors. Yet, two-layered BN based soft sensors are able to perform better than the popular OLS or PLS approaches due to its probabilistic nature to estimate measurement noise statistics.

- **Robust Multi-rate Bayesian network soft sensor (Chapter 3):** We further extended our work and looked at the input and output outliers in Chapter 3. This is a common problem that we face during off-line soft sensor development stage. Robust MR-BN-SS is capable of handling not only outliers, but also data pre-pressing issues addressed in previous chapter and analytical solutions are also derived in this chapter. Outliers are modeled in this chapter through the use of t-distribution and handled through the probabilistic framework systematically. From the simulation and industrial applications, it is observed that the proposed RMR-BN-SS is able to outperform the existing popular ROLS approach at different percentages of input/ output outliers.
- **Adaptive multi-rate Bayesian network soft sensor (Chapter 4):** In this, we addressed the issue of drift, which is a common problem during online soft sensor implementation stage. Here, only the output prediction step of MR-BN-SS was reformulated to account for the process or sample drift. A process drift, which reflects time dependency of the variables, is formulated through dynamic BN structure by introducing additional time-dependent node into the static BN structure of parameter learning step. On the other hand, sensor drift was formulated simply through static BN structure with assumed colored noise model. Analytical solutions are provided for the Bayesian inference step. From the simulation studies, it is evident that the proposed formulations are appropriate for adapting BN based soft sensor to a drifting data. Due to the probabilistic framework AMR-BN-SSs handle both sensor and process drift well, resulting in accurate predictions compared to the conventional bias updated OLS soft sensors.

From the simulations and industrial applications presented in this thesis, we can clearly observe the potential of probabilistic framework for developing soft sensors. The proposed approaches were always able to outperform the compared conventional soft sensors. Moreover, the flexibility of BN based soft sensors were demonstrated through two different BN structures. In which, multi-layered BN structured based soft sensors showed great potential to predict with greater accuracy. Exploring the possibilities of constructing different BN structures for a given problem and further analyzing their performances will be an interesting direction to go from this point.

5.2 Direction of future work

1) One of the interesting directions to take from here is to explore different BN structures and propose a systematic approach to construct an optimal BN structure for a given problem. Surely, core of the problem will be a tradeoff between computational complexity and prediction accuracy of developed BN based soft sensors. Therefore, one can extend this work into exploring different possibilities.
2) The assumption of this thesis is that all the variables are continuous and the problem is a single mode process. Practically, chemical process is usually multi-model; thus, one can extend this work to develop bank of BN based soft sensors for multi-mode problem. In this case, there will be addition of a discrete node, a scheduling variable, through which different modes will be identified. For the flow-network problem discussed in the simulation, the split ratio is kept constant, making the problem single mode. If we introduce a discrete node for this variable and assume it to be varying, the problem will be multi-model.

3) The BN structures are static in this work (except for the adaptive MR-BN-SS for process drift). One can make this BN structure dynamic, assuming all the variables are time dependent. In reality, process variables are time dependent; thus, this change will make the problem more realistic.
4) Moreover, the prior information of the source node was assumed to be uniform throughout the derivation, this information can also be utilized to improve output predictions.
5) Parameter learning was done by using EM algorithm. One can try different algorithms such as VB. This will assign distributions to the parameters and one can also utilize prior information of parameters.
6) In this thesis, output predictions are instantaneous and bias update was done whenever the newly available lab data is available. The assumption is that these values are arriving without any delay. However, in practice, newly arriving lab values have a time delay, meaning it is the information of the past; therefore, these are not utilized to make current instantaneous predictions. So, to account for this lab delay, one can attempt to implement window based inference with window size of N, which is bigger than the known lab delay. This way, newly coming lab measurements can be placed in the current window and this information can be utilized to make better prediction at the current instance.

Bibliography

1. P. Kadlec, B. Gabrys, S.Strandt, Data-driven soft sensors in the process industry. Computers and Chemical Engineering 33 (2009) 795-814.
2. P. Kadlec, R.Grbic', B.Gabrys, Review of adaptation mechanism for data driven soft-sensors, Computers and chemical engineering 35 (2011) 1-24.
3. S. Khatibisepehr, B.Huang, F.Xu, A.Espejo, A Bayesian approach to design of adaptive multi-model inferential soft sensors with application in oil sands industry, Journal of Process Control 22 (2012) 1913-1929.
4. Weiming Shao, Zhiqiang Ge, Zhihuan Song, Kai Wang, Nonlinear industrial soft sensor development based on semi-supervised probabilistic mixture of extreme learning machines, Control Engineering Practice, 91 (2019) 104098.
5. S. Khatibisepehr, B. Huang. Dealing with irregular data in soft sensors: Bayesian method and comparative study. Industrial & Engineering Chemistry Research 47 (2008) 8713-8723.
6. E. Radvar, G. Practical Issues in Non-Linear System Identification, M.Sc. Thesis, Department of Chemical and Materials Engineering, University of Alberta, Spring 2002.
7. Y.Z.Friedman, E.A .Neto, C.R.Porfirion, First Principles Distillation Inference Model for Product Quality Prediction. Hydrocarbon Process. 81 (2) (2002) 53-58.
8. S.D. Grantham, L.H. Ungar, A First Principles Approach to Automated Troubleshooting of Chemical Plants. Comput. Chem. Eng. 14 (1990) 783-798.
9. P. Kadlec, B. Gabrys, Soft Sensors: Where are we and what are the current and future challenges? IFAC Proceedings Volumes 42 (2009) 572-577.
10. R.A.M Noor, Z. Ahmad, M.M. Don, and M.H. Uzir, Modelling and Control of different types of polymerization process using neural network technique: A review. The Canadian Journal of Chemical Engineering, 88 (6) (2010) 1065-1084.
11. F. Guo, B. Huang, Output-relevant Variational Autoencoder for Just In Time Soft Sensor Modeling with Missing Data, Journal of Process Control, 92 (2020) 90-97.
12. A. Cozad, N.V. Sahinidis, and D.C. Miller, A combined first principles and data driven approach to model building, Computers and Chemical Engineering, 73 (2015) 116-127.

13. H. Abdi, LJ. Williams, Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics 2, (2010) 433-459.
14. P. Geladi, BR. Kowalski, Partial least-squares regression: A tutorial. Analytica Chimica Acta 185 (1986) 1-17.
15. R. Xie, N. Magbool Jan, K. Hao, L. Chen, B. Huang, Supervised Variational Autoencoders for Soft Sensor Modeling with Missing Data, IEEE Transactions on Industrial Informatics, 16 (4) (2020) 2820 - 2828.
16. R.B. Gopaluni, A. Tulsyan, B. Chachuat, B. Huang, J.M. Lee, F. Amjad, S.K.Damarla, J.W. Kim, N.P. Lawrence, Modern Machine Learning Tools for Monitoring and Control of Industrial Processes: A Survey. In Proceedings of IFAC World Congress Germany 2020.
17. Bingbing Shen, Le Yao, Zhiqiang Ge, Nonlinear probabilistic latent variable regression models for soft sensor application: From shallow to deep structure Control Engineering Practice, Volume 94, (2020) 104198.
18. D.Goeij, Moniek C. M, van Diepen M, Jager KJ, Tripepi G, Zoccali C, Dekker FW. Multiple imputation: Dealing with missing data. Nephrology, dialysis, transplantation 28 (2013) 2415-2420.
19. M.A. Tanner, W. H. Wong, The Calculation of Posterior Distribution by Data Augmentation, J. Am. Statist. Assoc. 82 (398) (1990) 528-540.
20. S.A. Imtiaz, S.L. Shah, Treatment of Missing Values in Process Data Analysis, The Canadian Journal of Chemical Engineering 86(5) (2008).
21. A.P. Dempster, N. M. Laird and D. B. Rubin, Maximum Likelihood from Incomplete Data via EM Algorithm, J.R. Statist. Soc. Ser. B (Methodol.) 39(1) (1977) 1-38.
22. N. Sammaknejad, Y. Zhao, B. Huang., A review of the expectation maximization algorithm in data-driven process identification. Journal of Process Control 73 (2019) 123-136.
23. M.E. Tipping & C.M. Bishop, Probabilistic Principal Component Analysis, J. R. Statist. Soc. B 61 (1999) 611-622.
24. Z. Ge, F. Gao, Z. Song, Mixture probabilistic PCR model for soft sensing of multimode processes. Chemometrics and Intelligent Laboratory Systems 105 (2011) 91-105.

25. M. G. Gustafsson, A Probabilistic Derivation of the Partial Least-Squares Algorithm, J. Chem. Inf. Comput. Sci. 41 (2) (2001) 288-294.
26. A. Mohammadi, R.Zarghami , D. Lefebvre, S. Golshan, N. Mostoufi, Soft sensor design and fault detection using Bayesian network and probabilistic principal component analysis. Journal of Advanced Manufacturing and Processing (2019).
27. Z. Liu, Z. Ge, G. Chen, Z. Song, Adaptive soft sensors for quality prediction under the framework of Bayesian network, Control Engineering Practice 72 (2018) 19-28.
28. S. Shoham, M.R. Fellows, R.A. Norman, Robust automatic spike sorting using mixtures of multivariate t-distributions, J. Neurosci. Methods 127 (2) (2003) 111-122.
29. D. Koller, N. Friedman, Probabilistic graphical models: principles and techniques, The MIT press (2009).
30. Z. Gharamani, Learning Bayesian networks. Lecture notes in Artificial Intelligence, Department of Computer Science, University of Canada, Toronto, 1997.
31. D. Heckerman, A Tutorial on learning Bayesian networks. Technical Report MSR-TR-95-06, Microsoft research, November, 1996.
32. D. Margaritis, Learning Bayesian Network Model Structure from Data, Ph.D. Thesis Dissertation (2003).
33. D. M. Chickeringn, Learning Bayesian networks is NP-Complete, New-york, NY: Springer-Verlag (1996).
34. T. K. Moon, The Expectation- Maximization Algorithm, IEEE Signal Processing Magazine, (1996) 47-60.
35. Friedman, N., Learning Belief Networks in the presence of Missing values and hidden variables.
36. D. Heckerman, A Tutorial on Learning with Bayesian networks, Technical report.
37. K.P. Murphy, An introduction to graphical models, Web, 2001. URL http://www.ai. mit.edu/~murphyk/Papers/intro_gm.pdf.
38. P. Dagum, M. Luby, Approximating probabilistic inference in Bayesian belief networks is NP-hard. Artificial Intelligence 60 (1993) 141-153.
39. G.F. Cooper, The computational complexity of probabilistic inference using Bayesian belief networks. Artificial Intelligence 42 (1990) 393-405.

40. M.I. Jordon, Z. Ghahramani, T.S. Jaakkola and L.K. Saul. An introduction to variational methods for graphical models. In M. Jordon, editor, Learning in Graphical Models. MIT Press, 1998.
41. Data reconciliation and gross error detection; an intelligent use of process data. Scitech Book News 1 (2000) 24.
42. P.Huber, Robust Statistics, John Wiley & Sons, NY, 1981.
43. L.A. Jaeckel, Estimating regression coefficients by minimizing the dispersion of the residuals, Ann. Math.Stat. 43 (5) (1972) 1449-1458.
44. P. Rousseeuw, Least median of squares regression, J. Am. Stat. Assoc. 79 (388) (1984) 242-272.
45. R.A.M Noor, Z. Ahmad, M.M. Don, and M.H. Uzir, Modeling and Control of different types of polymerization process using neural network technique: A review. The Canadian Journal of Chemical Engineering, 88 (6) (2010) 1065-1084.
46. L.Ljung, System Identification: Theory for the User, Prentice Hall PTR, Upper Saddle River, HJ,
47. M.Svensen, C. Bishop, Robust Bayesian mixture modeling, Neurocomputing 64 (2005) 235-252.
48. . R.Tan, B.Huang, Z.Li, Estimation for flat-topped Gaussian distribution with application in system identification, J.Chemom. 30 (12) (2016) 726-738.
49. S. Shoham, M.R. Fellows, R.A. Norman, Robust automatic spike sorting using mixtures of multivariate t-distributions, J. Neurosci. Methods 127 (2) (2003) 111-122.
50. G. McLachlan , T. Krishnan, The EM algorithm and extensions, in: WILEY Series in Probability and Statistics, Wiley, Hoboken, NJ, 2008.
51. K. Lange, The EM algorithm, in: Numerical Analysis for Statisticians, Springer New York, New York, NY (2010) 223-247.

52. M.R.Gupta, Y. Chen, Theory and Use of the EM Algorithm, Now Publishers Inc, 2011.
53. S. Gibson, B. Ninnes, Robust maximum-likelihood estimation of multivariable dynamic systems, Automatica 41 (10) (2005) 1667-1682.
54. X. Yang, Y. Lu, Z. Yan, Robust global identification of linear parameter varying systems with generalized expectation-maximization algorithm, IET Control Theory Appl.9 (7) (2015) 1103-1110.
55. M.J. Beal, Variational algorithms for approximate Bayesian inference (PhD thesis), University of London, 2003.
56. T. Manika, Expectation propagation for approximate Bayesian inference, in: Proceedings of the Seventeenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-01), Morgan Kaufmann, San Francisco, CA, (2001) 62-369.
57. W.K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrika 57 (1) (1970) 97-109.
58. Zhu J, Ge Z, Song Z. Robust semi-supervised mixture probabilistic principal component regression model development and application to soft sensors. Journal of Process Control 32 (2015) 25-37.
59. Fan L, Kodamana H, Huang B. Identification of robust probabilistic slow feature regression model for process data contaminated with outliers. Chemometrics and Intelligent Laboratory Systems 173 (2018) 1-13.
60. C. Scheffler, A derivation of the EM updates for finding the maximum likelihood parameter estimates of the Student's t-distribution, First draft (2008)
61. P. Kadlec, R. Grbic, B.Gabrys, Review of adaptation mechanisms for data-driven soft sensors. Computers and Chemical Engineering 35 (2011) 1-24.

62. A.D. Quelhas, J.C.Pinto, Soft sensor models: Bias updating revisited,
63. Singh, A. Modeling and model updating in the real-time optimization of gasoline blending, Master thesis, University of Toronto (1997)
64. R. Sharmin, U. Sundararaj, S. Shah, L.V. Griend, Y. Sun, Inferential sensors for estimation of polymer quality parameters – Industrial application of PLS- based soft sensor for LDPE plant, Chemical Engineering Science, 61 (2006) 6372-6348
65. J. Gama, P. Medas, G. Castillo, P. Rodrigues, Learning with drift detection. In proceedings of the 17th Brazilian symposium on artificial intelligence advances in artificial intelligence (SBIA 2004) (2004) 78-83
66. M. A. Maloof , R. S. Michalski, selecting examples for partial memory learning. Machine learning, 41 (1) (2000) 27-52
67. R. Klinkenberg, Learning drifting concept: Example selection vs. example weighing. Intelligent Data Analysis, 8 (3) (2004) 281-300
68. M. Scholz, R. Klinkenberg, Boosting classifiers for drifting concept. Intelligent Data Analysis, 11(1) (2007) 3-28
69. X. Wang, U. Kruger, G.W. Irwin, Process monitoring approach using fast moving window PCA. Industrial and Engineering Chemistry Research, 44 (15) (2005) 5691-5702
70. X.Liu, U. Kruger, T. Littler, L. Xie, S. Wang, Moving window kernel PCA for adaptive monitoring of nonlinear processes. Chemometrics and Intelligent Laboratory Systems, 96 (2) (2009) 132-143
71. B.S. Dayal, J. F. MacGregor, Recursive exponentially weighted PLS and its applications to adaptive control and prediction. Journal of Process Control, 7(3) (1997) 169-179

72. W. Li, H. H. Yue, S. Valle- Cervantes, S.J. Qin, Recursive PCA for adaptive process monitoring. Journal of Process Control, 10 (5) (2005) 471-486

73. S.J. Qin, Recursive PLS algorithm for adaptive data modeling. Computers and Chemical Engineering, 22-(4-5) (1998) 503-514

74. S. Mu, Y. Zeng, R. Liu, P. Wu, H. Su, J. Chu, Online dual updating with recursive PLS model and its application in predicting crystal size of purified terephthalic acid (PTA) process. Journal of Process Control, 16 (6) (2006) 557-566

75. C. Liu, MI estimation of the multivariate distribution and the EM algorithm, J. Multivariate Anal. 63 (2) (1997) 296-312.

Appendix A

Appendix to Chapter 2

Approach	True	down-sampled BN-SS	multi-rate BN-SS
σ_{y_1}	0.010	0.015	0.0091
σ_{y_2}	0.004	0.117	0.0032
σ_{y_3}	0.001	0.074	0.0012
σ_{y_4}	0.420	0.554	0.566
σ_{y_5}	0.001	0.00061	0.00097
σ_{y_6}	0.01	0.0036	0.0034

Table 2. 11: Comparison of noise variances true and estimated values

Approach	True	down-sampled BN-SS	multi-rate BN-SS
σ_{x_2}	0.00004	0.0012	0.0012
σ_{x_3}	0.00001	0.0017	0.00032
σ_{x_4}	0.00005	0.0012	0.0037
σ_{x_5}	0.00001	0.00041	0.00034
σ_{x_6}	0.0001	0.00041	0.0033

Table 2. 12: Comparison of true and estimated hidden noise variances

Approach	True	down-sampled BN-SS	multi-rate BN-SS
β_1	0	-0.29	0.0034
β_2	1	1	1
β_3	0	-0.05	0.0018
β_4	1	1	1

β_5	1	1	1
β_6	0	-0.66	-0.50
β_7	0.64	0.65	0.65
β_8	0	0.28	1.01
β_9	1	1	1
β_{10}	0	0.73	0.5
β_{11}	0.36	0.35	0.35

Table 2. 13: Comparison of true and estimated parameters

Appendix B

Appendix for Chapter 3

Approach	True value	RMR-BN estimates
σ_{y_1}	0.0409	0.0384
σ_{y_2}	0.0164	0.0142
σ_{y_3}	0.0052	0.0050
σ_{y_4}	0.4117	11.3031
σ_{y_5}	0.0052	0.0044
σ_{y_6}	0.0376	0.0187

Table 3. 7: Comparison of noise variances true and estimated values

Approach	True value	R- MR-BN estimates
σ_{X_2}	0.0002	0.0030
σ_{X_3}	0.0001	0.0006
σ_{X_4}	0.0002	0.0116
σ_{X_5}	0.0001	0.0009
σ_{X_6}	0.0004	0.0105

Table 3. 8: Comparison of true and estimated hidden noise variances

Approach	True value	RMR-BN estimates
β_1	0	0.0028
β_2	1	0.9999
β_3	0	0.0072
β_4	1	1.0000
β_5	1	1.0000

β_6	0	-0.4965
β_7	0.64	0.6449
β_8	0	-0.0029
β_9	1	1.0000
β_{10}	0	0.4987
β_{11}	0.36	0.3550
r	-	0.8907

Table 3. 9: Comparison of true and estimated parameters