Design of Soft Sensors Based on Slow Feature Analysis

Submitted in Partial Fulfillment for the BTech Course on

Capstone Project (CP 303)

By

Shivam Pandey (2019CHB1055)

Under the Guidance of Dr. V. Jayaram

Indian Institute of Technology, Ropar

Rupnagar, Punjab - 140001

May 2023

Acknowledgement

I would like to thank my project advisor Dr. V. Jayaram for his constant support and guidance provided throughout the course. His consistent input was really useful and valuable in getting the project done. He was receptive to hearing my opinions and making corrections when I erred, and also made sure I stayed on course with the task at hand.

Also, I would like to thank all those who directly or indirectly helped towards the completion of the project, mainly the authors of the articles used in completing the project. The articles were quite informative and helpful in providing the desired information.

Abstract

This report presents a comprehensive study on the design of soft sensors based on Slow Feature Analysis (SFA) for process industries. Soft sensors, advanced data-driven tools used for estimating unmeasured variables and predicting process behavior, offer significant advantages in process monitoring and optimization. The report begins by introducing the concept of soft sensors, discussing their various types and highlighting their importance in the process industry. Furthermore, it explores the challenges faced during the development of soft sensors and provides an overview of existing approaches. A generic framework for designing soft sensors is presented, covering data acquisition, preprocessing, feature extraction, and modeling stages.

The report then focuses on the application of Slow Feature Analysis as a powerful technique for feature extraction. It explains the working principle of SFA and demonstrates its benefits in enhancing signal representation and reducing noise. Additionally, the working principles of linear regression and random forest algorithms, commonly employed in soft sensor modeling, are discussed. Two case studies are conducted to validate the proposed soft sensor approach. The first case study involves fouling prediction in heat exchangers, showcasing the application of the developed soft sensor utilizing SFA-based feature extraction. The second case study revolves around the quadruple tank system, demonstrating the effectiveness of soft sensors in process monitoring and control. The results from both case studies are analyzed to evaluate the accuracy and reliability of the developed soft sensor models.

Overall, this report provides valuable insights into the design of soft sensors based on Slow Feature Analysis. It covers the various aspects of soft sensor development, including the conceptual understanding, challenges, approaches, and practical applications. The findings presented in this report contribute to the advancement of soft sensor technologies, facilitating their adoption in real-world process industries for improved monitoring, control, and optimization.

Table of Contents

1.Introduction

Soft sensors have revolutionized the field of process industries by offering a powerful solution for the prediction of values of expensive physical variables using software-based models. They provide real-time monitoring and control of various parameters such as temperature, pressure, and chemical composition, enabling efficient process optimization and enhanced operational performance. In process industries, the accurate prediction of physical variables is crucial for ensuring optimal process conditions and maximizing productivity. Soft sensors offer a diverse range of types, including model-based and data-driven approaches. Model-based soft sensors rely on mathematical models based on first principles, incorporating knowledge of the underlying physics and chemistry of the process. These models provide a mechanistic understanding of the system and facilitate the estimation of variables. For instance, in a chemical reactor, a model-based soft sensor can utilize reaction kinetics and mass balances to predict the concentration of key compounds[1].

On the other hand, data-driven soft sensors harness the power of machine learning algorithms to analyze historical data and derive patterns, correlations, and trends. These models learn from past observations and can make accurate predictions based on the learned relationships. For instance, in a wastewater treatment plant, a data-driven soft sensor can analyze historical data on influent characteristics, such as flow rate and pollutant concentrations, to predict the effluent quality in real-time.

The implementation of soft sensors in process industries addresses several critical needs. Experimental measurements of physical variables can be challenging, time-consuming, and prone to errors. Soft sensors offer a reliable and efficient alternative to expensive and cumbersome physical sensors. By providing real-time monitoring and control capabilities, soft sensors enable proactive decision-making and process optimization. They optimize the process, reduce waste, and enhance overall efficiency. For example, in a power plant, soft sensors can monitor turbine performance parameters, such as pressure and temperature, to optimize power output and minimize fuel consumption.

Moreover, soft sensors significantly reduce costs associated with installation and maintenance. Physical sensors often require extensive wiring, calibration, and regular maintenance, resulting in high upfront and operational expenses. Soft sensors, being software-based, eliminate these costs by leveraging existing data infrastructure and computational resources. They offer a cost-effective solution while providing accurate estimations of physical variables. For instance, in an oil refinery, soft sensors can estimate the properties of crude oil, such as density and viscosity, reducing the need for expensive online analyzers.

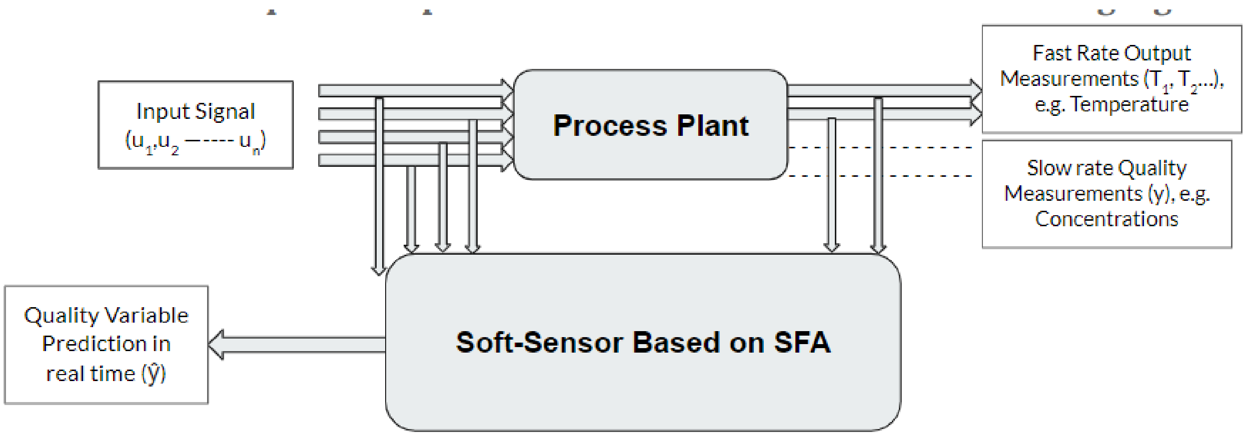


Fig.1: Flow Chart on Soft-Sensor Based on SFA

Why Data Driven Sensors?

Data-driven soft sensors have become the preferred choice in process industries due to several advantages over model-based approaches. Firstly, processes in these industries often exhibit high variability, making it challenging for model-based soft sensors that rely on accurate knowledge of system parameters to provide reliable results. In contrast, data-driven soft sensors leverage historical data to learn patterns and correlations, enabling them to adapt and deliver accurate estimations even in the presence of complex and variable process conditions. This flexibility makes them well-suited for processes with high variability[1].

Secondly, data-driven soft sensors offer advantages in terms of development time and adaptability. Model-based soft sensors require significant time and effort to develop and validate accurate mathematical models, which can be particularly challenging in dynamic processes with rapidly changing system dynamics. In contrast, data-driven soft sensors can quickly train on available historical data and adapt to changing conditions, making them more efficient and effective for real-time monitoring and control. Additionally, data-driven soft sensors can overcome limitations in the knowledge of physical processes and data availability by leveraging available data to identify complex patterns and relationships.

Overall, the ease of implementation of data-driven soft sensors further contributes to their preference. Model-based approaches often require extensive domain knowledge and computational resources, while data-driven soft sensors primarily rely on machine learning algorithms and data analysis techniques, leveraging existing data infrastructure and computational resources. This makes data-driven soft sensors more accessible, cost-effective, and practical for implementation in process industries. By providing accurate estimations and valuable insights, data-driven soft sensors play a vital role in optimizing processes, reducing costs, and improving overall performance in process industries.

Challenges in Developing Data Driven Soft Sensors and Approaches Available

Developing data-driven soft sensors entails several challenges that need to be addressed for accurate and reliable models. One of the primary challenges is the optimal selection of input variables, as determining the most relevant inputs from a vast pool can be a complex task. Selecting the right set of variables that capture the essential information and dynamics of the process is crucial for building effective models. Data preprocessing poses another challenge, involving the removal of outliers, noise, and handling missing values. Ensuring the quality and integrity of the data is crucial for developing robust models. Techniques such as outlier detection, noise filtering, and imputation of missing values are commonly employed to enhance the quality of the data.

Process drift, which refers to changes in the process behavior over time, presents a significant challenge in developing data-driven soft sensors. Models should be adaptable to accommodate variations in the process dynamics and maintain accurate predictions even as the system evolves. Handling process drift is crucial for ensuring the reliability and effectiveness of soft sensor models. Dealing with data collinearity is another challenge that arises when input variables are highly correlated. Identifying and obtaining independent variables is important to avoid multicollinearity issues and improve the interpretability of the model. By reducing dependencies among input variables, installation and maintenance costs can be minimized.

The quantity of data available also plays a vital role in developing data-driven soft sensors. Larger datasets tend to provide more accurate results and better model generalization. Adequate data collection and storage strategies need to be implemented to ensure sufficient data availability for training and testing purposes. Choosing an appropriate model is crucial and depends on the complexity of the process and the relationships within the data. Linear models may be suitable for simple processes with linear relationships, while non-linear models, such as neural networks or support vector machines, may be required for more complex systems with non-linear dynamics.

In the literature, various approaches have been proposed and applied to address these challenges in developing data-driven soft sensors. Regression-based models, principal component analysis (PCA), slow feature analysis (SFA), deep learning techniques, ensemble methods like random forests and boosting algorithms, among others, have been utilized successfully in the development of data-driven soft sensors. These approaches offer different advantages and capabilities, allowing researchers and practitioners to choose the most appropriate method based on the specific requirements and characteristics of the process under consideration.

2. Method

Generic Approach in Developing Data-Driven Soft Sensors:

The development of data-driven soft sensors follows a generic approach encompassing key steps from data collection and preprocessing to model deployment and monitoring. These steps ensure the accuracy and reliability of the soft sensor models in process industries.

The first step is data collection, which involves gathering historical data related to the process variables of interest. This data serves as the foundation for training and validating the soft sensor models. Additionally, data preprocessing techniques are applied to enhance data quality. Outliers are identified and removed, and missing data is filled using appropriate methods to ensure the integrity and completeness of the dataset. Feature selection and engineering play a crucial role in developing effective data-driven soft sensors. Techniques like Principal Component Analysis (PCA) and Slow Feature Analysis (SFA) are commonly employed to identify and extract the most relevant features from the dataset. These techniques reduce dimensionality and capture essential information that represents the underlying process dynamics.

Model development involves selecting suitable regression-based models or machine learning algorithms to build the soft sensor models. Regression-based models such as linear regression or polynomial regression can be employed for simpler processes with linear relationships, while machine learning algorithms like support vector machines, random forests, or neural networks are utilized for more complex nonlinear processes. The models are trained on the preprocessed data to establish the mapping between the input variables and the desired output. Model optimization is crucial for fine-tuning the soft sensor models and improving their performance. This step involves tuning the hyperparameters of the chosen models and employing optimization techniques to achieve optimal model behavior and accuracy. The selection of appropriate hyperparameters and optimization techniques helps in achieving better model generalization and predictive capabilities.

To ensure the reliability and effectiveness of the soft sensor models, model validation is performed. The models are tested on unseen or validation data to assess their performance and generalization ability. Performance metrics such as mean squared error, root mean squared error, or coefficient of determination are calculated to evaluate the model's accuracy and predictive capabilities. Once the soft sensor models are validated, they are ready for deployment in the process industry. Continuous monitoring of the model's performance is essential to ensure accurate and reliable outputs. Monitoring mechanisms are established to track the model's performance over time and identify any deviations or degradation in performance. This monitoring process ensures that the soft sensor models remain effective and provide real-time monitoring and control of critical process variables.

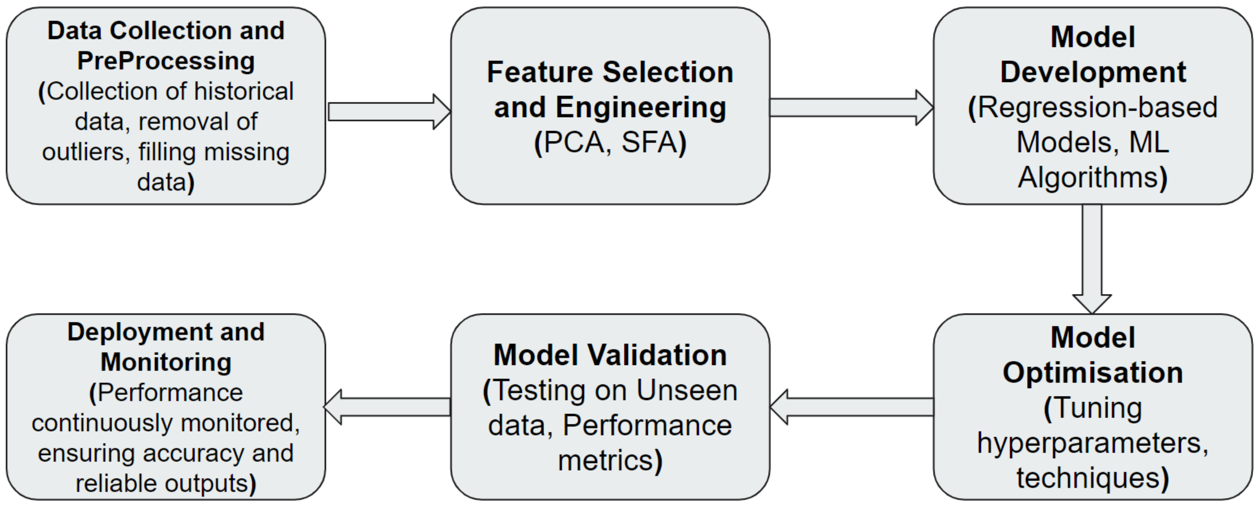


Fig.2 : Approach for Developing Soft Sensors

Feature Selection Methods - PCA and SFA:

Principal Component Analysis (PCA):

Principal Component Analysis (PCA) is a widely used method for feature analysis in data-driven soft sensor development. It aims to reduce the dimensionality of the dataset while preserving the most relevant information. The algorithm works by finding a set of orthogonal vectors called principal components that capture the maximum variance in the data.

The working principle of PCA involves transforming the original dataset into a new coordinate system represented by the principal components. The first principal component is the direction along which the data varies the most. Subsequent principal components are chosen to be orthogonal to the previous ones and capture the remaining variance in the data. Each principal component is a linear combination of the original variables, with coefficients determined through the eigenvalue decomposition of the covariance matrix.

The advantage of PCA lies in its ability to identify the most informative features and reduce the dimensionality of the dataset. By retaining the principal components that explain the majority of the variance, PCA helps eliminate redundant and irrelevant features, simplifying the modeling process and improving computational efficiency. Furthermore, PCA aids in visualizing the data by projecting it onto lower-dimensional spaces, facilitating data interpretation and analysis.

Slow Feature Analysis (SFA):

Slow Feature Analysis (SFA) is a powerful method used for feature extraction in data-driven soft sensor development. It aims to capture slowly varying components in the data, which are considered to be the most important and informative. SFA is particularly useful in handling noisy data caused by raw material fluctuations, environmental changes, and nominal disturbances in process industries.

The working principle of SFA involves extracting features that change slowly over time from time series data. It identifies the slowest varying components in the dataset, which represent the relevant and underlying trends in the process data. By focusing on the slowest features, SFA effectively filters out noise and removes the influence of fast-changing variables that may not contribute significantly to the soft sensor model.

One of the advantages of SFA is its ability to improve model performance by selecting the optimal input variables. It helps in reducing the dimensionality of the dataset, making the model less complex and computationally efficient. By extracting the slowly varying components, SFA aids in dimensional reduction while preserving the critical information required for accurate soft sensor modeling. In addition to removing collinearity like PCA, SFA goes further by specifically targeting slowly varying variables, ensuring that the extracted features capture the essential dynamics of the process. This approach allows for a better understanding of the underlying trends and patterns in the data, leading to improved model interpretability and predictive accuracy in data-driven soft sensor applications.

Working Principle:

Given an I dimensional input signal corrupted with noise and are correlated:

$$x(t) = [\ x_1(t)\ ,\ x_2(t),\ \ldots\ldots,\ x_I(t)\ ] \tag{1}$$

Objective: Find an input - output function

$$g(x) = [\ g_1(x),\ g_2(x), \ldots\ldots g_J(x)\ ]$$

To obtain slow features: (2)

$$s(t) = [\ s_1(t),\ s_2(t), \ldots\ldots s_J(t)\ ] \quad \text{where } s_J(t) = g_J(x(t))$$

Notations: (3)

For a signal X(t): quantitative measure of slowness or fastness is given by:

$$\Delta(\boldsymbol{X}(t)) \approx \left\langle \dot{\boldsymbol{X}}^2(t) \right\rangle_t$$

where $\langle . \rangle$ is the expectation operator.

$$\left\langle \mathbf{X}(t) \right\rangle_t = \frac{1}{t_1 - t_0} \int_{t_1}^{t_0} \mathbf{X}(t) dt \text{ (for continous)}$$

$$\approx \frac{1}{T} \sum_{t=1}^{T} \mathbf{X}(t) \text{ (for discrete)}$$

$$\dot{\mathbf{X}}(t) = \frac{d\mathbf{X}}{dt} \approx \mathbf{X}(t) - \mathbf{X}(t-1)$$

Our aim here is to remove noise and minimize the below expression:

$$\min_{g_j(.)} \Delta(s_j) = \left\langle \dot{s}_j^2 \right\rangle$$

such that (4)

$$\left\langle s_j \right\rangle = 0 \text{ (zero mean)}$$
$$\left\langle s_j^2 \right\rangle = 1 \text{ (unit variance)}$$
$$\left\langle s_{j'} s_j \right\rangle = 0 \text{ (Decorrelation)}$$

where j\` < j.

Directly solving the above optimization problem is a difficult task. Thus we approximate the input-output function g(x) as a linear combination of several nonlinear functions:

$$g_j(x(t)) = \sum_{k=1}^{K} w_{jk} h_k(x) = \mathbf{w}_j^T \mathbf{h}(x(t))$$

representing $\mathbf{z}(t) = \mathbf{h}(x(t))$. (5)

Then we have

$$s_j(x(t)) = g_j(x(t)) + w_j^T z(t)$$

$$\min_{\mathbf{w}_\mathbf{j}} \Delta(s_j) = \left\langle \dot{s}_j^2 \right\rangle = \mathbf{w}_\mathbf{j}^\mathbf{T} \left\langle \dot{\mathbf{z}} \dot{\mathbf{z}}^T \right\rangle \mathbf{w}_\mathbf{j}$$

Under the assumption that (6)

$$\left\langle \mathbf{z} \right\rangle = \mathbf{0}, \left\langle \mathbf{z}\mathbf{z}^T \right\rangle = \mathbf{I}$$

where w_j is an orthonormal vector.

On solving the above optimization problem (reduces to generalized eigen value problem), we finally can extract the slow features by integrating the same with PCA.

We can illustrate the working of Linear SFA through a simple numerical example. Consider the following four input signals with noises (v_1 to v_4)[5]:

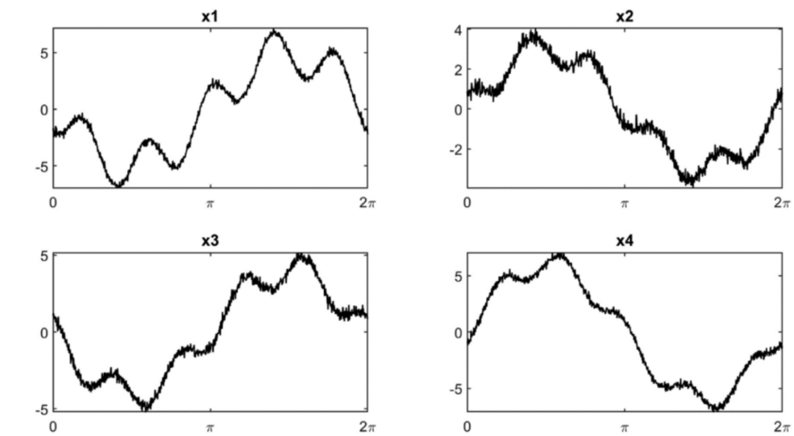
$$x_1(t) = -5\sin(t) - 2\cos(5t) + v_1$$

$$x_2(t) = 3\sin(t) + 0.75\cos(5t) + v_2$$

$$x_3(t) = -4\sin(t) + \cos(5t) + v_3$$

$$x_4(t) = 6\sin(t) - \cos(5t) + v_4 \qquad (7 \text{ to } 10)$$

The corresponding plot for these signals can be shown below:



Corresponding extracted slow features can be shown below:

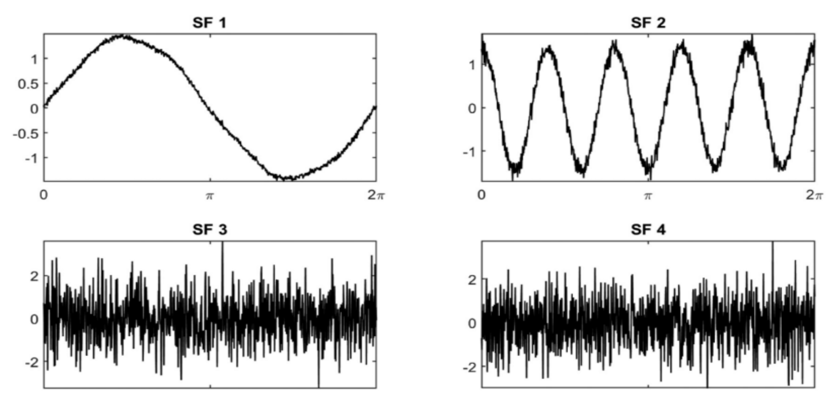


Fig. 3: Input signals and corresponding extracted slow feature signals.

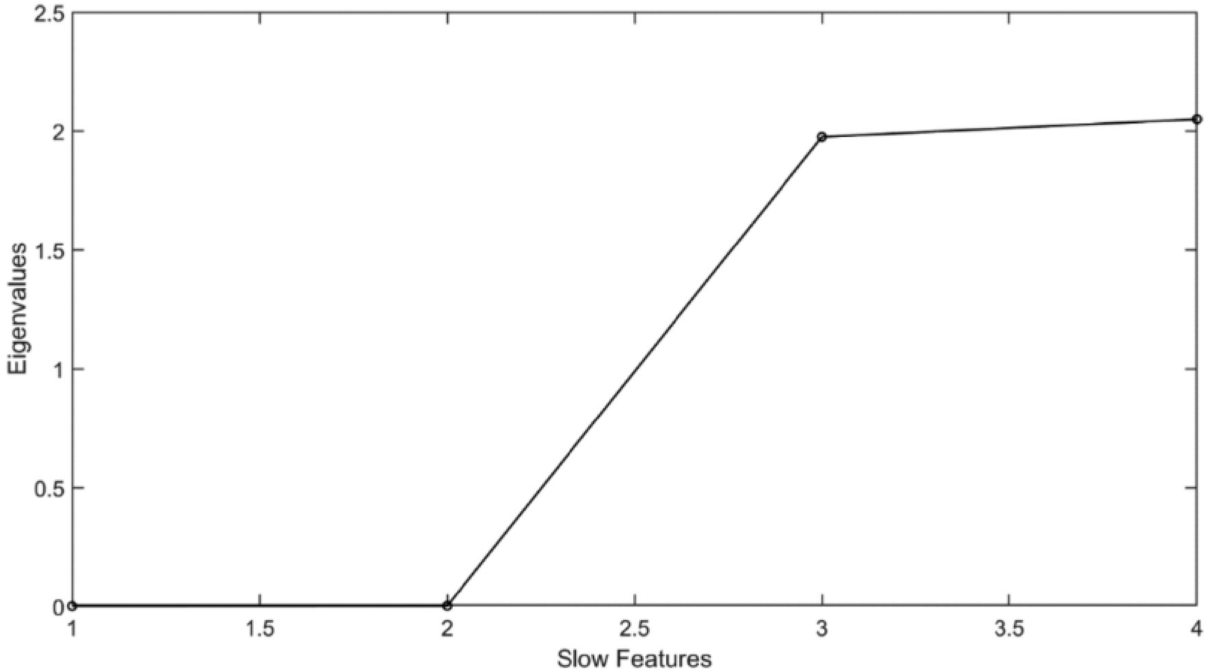


Fig.4: Eigenvalue Plot

We can see that the first two SFs are approximately sin(t) and cos(5t) respectively. Thus linear SFA can indeed extract the underlying driving forces behind a set of inputs. These are the two extracted slow feature signals in this case. The eigenvalue plot shows a clear elbow in the eigenvalues after two SFs, thus indicating that there are only two dominant SFs.

Model Development - Linear Regression and Random Forest

Linear Regression:

Linear regression is a fundamental and widely used modeling technique in data-driven soft sensor development. It is a supervised learning algorithm that establishes a linear relationship between the input variables and the output variable. The working principle of linear regression involves fitting a straight line to the data points by minimizing the sum of the squared differences between the observed and predicted values. Linear regression is used in various applications across different industries, including process industries. It is particularly effective when the relationship between the input and output variables can be approximated by a linear function. The model development process begins with the collection of relevant input and output data. The input variables are typically the process parameters, while the output variable represents the variable of interest.

In model development, the next step is to split the data into training and testing sets. The training set is used to estimate the model parameters through the ordinary least squares (OLS) method or other optimization techniques. The estimated parameters define the slope and intercept of the regression line. The testing set is then used to evaluate the model's performance and assess its ability to generalize to unseen data. Model evaluation is carried out using various performance metrics such as mean squared error (MSE), root mean squared error (RMSE), and coefficient of determination (R-squared). These metrics measure the accuracy and predictive capability of the linear regression model. The lower the MSE and RMSE values, and the higher the R-squared value, the better the model's fit and predictive performance.

Linear regression offers several advantages in soft sensor development. It provides simplicity and interpretability, making it easy to understand the relationship between the input and output variables. Additionally, linear regression models can be quickly trained and computationally efficient, allowing for real-time monitoring and control in process industries. However, linear regression assumes a linear relationship between the variables and may not capture complex nonlinear dynamics. In such cases, alternative approaches like nonlinear regression or machine learning algorithms may be more appropriate.

Working Principle[9]:

Let $\mathbf{x_i}$ be the independent and $\mathbf{y_i}$ be the dependent variable.

Hypothesis Function is represented as :

$$\mathbf{h_\theta(x_i) = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + \ldots\ldots \theta_j x_{ij} \ldots\ldots \theta_n x_{mn}}$$

where θ_j are parameters of hypothesis and x_{ij} (i^{th} training example of j^{th} feature). (11)

We have the cost function given as :

$$\mathbf{J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2}$$

$$\mathbf{J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2}$$

where $\hat{y}$ - predicted value, y - experimental value. (12)

Our goal here is to minimize the cost function:

$$\min_{\theta_0,\theta_1..\theta_n} \mathbf{J}(\theta_0, \theta_1 .. \theta_n)$$
$$\frac{\partial \mathbf{J}(\theta_j)}{\partial \theta_j} = \mathbf{0}$$

On solving the above optimization problem, we get hypothesis parameter set: (13)

$$\theta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Random Forest Algorithm:

The random forest model is a versatile and powerful machine learning algorithm widely used in data-driven soft sensor development. It is an ensemble learning method that combines multiple decision trees to make predictions. The working principle of the random forest model involves creating an ensemble of decision trees, each trained on a random subset of the data and features. The predictions from individual trees are then combined through voting or averaging to obtain the final prediction.

Random Forest finds applications in various domains, including process industries, where it is particularly useful in handling complex and nonlinear relationships between input and output variables. In the model development process, data collection is the first step, where both input and output variables are gathered. The dataset is then split into training and testing sets to train and evaluate the model, respectively. During the training phase, multiple decision trees are built, each considering a random subset of the features. This randomness helps in reducing overfitting and improving the model's generalization ability. The decision trees are constructed by recursively partitioning the data based on different features and their thresholds, aiming to minimize impurity or maximize information gain at each split.

Hyperparameter tuning is an essential step in the model development process. Hyperparameters such as the number of trees, maximum depth of trees, and minimum number of samples required for a split significantly impact the model's accuracy and robustness. Proper tuning of these hyperparameters is crucial to achieve optimal performance. Once the random forest model is trained, it is evaluated using the testing set to assess its performance. Common evaluation metrics include accuracy, precision, recall, and F1 score. These metrics provide insights into the model's ability to correctly predict the output variable based on the input variables.

Random Forest offers several advantages in soft sensor development. It can handle high-dimensional data and capture complex relationships between variables effectively. The model is resistant to overfitting and performs well even with noisy data or missing values. Additionally, random forests provide feature importance measures, allowing for variable selection and understanding the relative importance of different inputs in the soft sensor model.

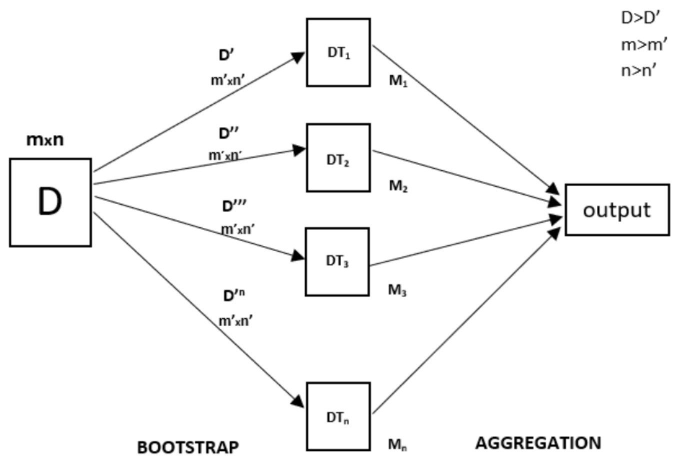


Fig.5: Random Forest Algorithm.

3. Case Studies

I . Fouling in Heat Exchangers:

Fouling in heat exchangers is a common problem encountered in process industries, which significantly impacts heat transfer efficiency and increases energy consumption. It refers to the deposition and accumulation of unwanted substances on the heat transfer surfaces, leading to reduced performance and operational issues. Fouling can occur due to various factors such as impurities in the process fluid, chemical reactions, scaling, or biological growth. Understanding and mitigating fouling is essential for maintaining the optimal performance of heat exchangers. Extensive research has been conducted to develop strategies for fouling prevention, detection, and cleaning. Predictive models have been developed to estimate fouling rates and predict fouling behavior based on parameters such as fluid properties, flow conditions, and surface characteristics. These models enable engineers to anticipate fouling effects and implement preventive measures.

Fouling monitoring systems play a crucial role in real-time fouling detection. These systems employ sensors and instrumentation to measure parameters such as pressure drop, temperature, and heat transfer coefficient. By monitoring these parameters, early signs of fouling can be detected, enabling timely intervention and preventive actions. Effective cleaning methods are employed to remove fouling deposits from heat exchanger surfaces. Mechanical cleaning, chemical cleaning, and innovative techniques like ultrasonic cleaning and high-pressure water jetting are utilized to restore heat transfer efficiency and prolong the equipment's lifespan.

Fig.6: Fouling in Heat Exchanger

Information about the dataset:

DataSet Reference: Asomaning, S., 1990. The role of olefins in fouling of heat exchangers. University of British Columbia, Vancouver [4].

Independent Variables (Features and their Correlation) : Density (0.244), Time (0.308), Surface temperature (0.91), fluid temperature (-0.386), fluid velocity(0), equivalent diameter(0), dissolved oxygen (0.344). [Used Pearson's Correlation] [2].

Dependent Variable: Fouling Factor.

II . Quadruple Tank System:

The quadruple tank system is a well-known benchmark system widely used in the field of process control and automation. It consists of four interconnected tanks, each with its own inlet and outlet valves, allowing for controlled fluid flow and level regulation. The system's complexity arises from the interaction between the tanks and the need to maintain desired liquid levels. The quadruple tank system serves as a valuable testbed for studying and evaluating different control strategies and algorithms. It provides a controlled environment to investigate various aspects of process control, including level control, disturbance rejection, stability, and robustness. Researchers and engineers use this system to develop and validate control algorithms, compare different control techniques, and assess the performance of advanced control strategies.

The primary objective of the quadruple tank system is to maintain desired liquid levels in each tank despite disturbances and uncertainties. This requires implementing control actions to adjust the flow rates through the inlet and outlet valves. By continuously monitoring the liquid levels and comparing them to the desired setpoints, control algorithms can compute the appropriate control signals to regulate the valves and achieve the desired level control. The quadruple tank system represents a simplified version of real-world industrial processes and serves as a stepping stone for understanding and implementing process control in more complex systems. It provides a platform for studying fundamental control concepts such as feedback control, cascade control, and advanced control techniques like model predictive control (MPC) or fuzzy logic control.

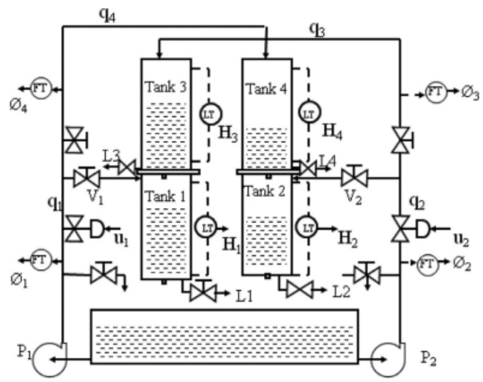


Fig.7: Quadruple Tank Setup[8]

Information about the dataset:

DataSet Reference: Jayaram V, Piyush L, Sachin C, Lorenz T,2017 . Development of moving window state and parameter estimators under maximum likelihood and Bayesian frameworks, Department of Chemical Engineering, Indian Institute of Technology Bombay, India [8].

Independent Variables (Features and their Correlation) : Input Variables (manipulated variables) : Flow1 (0.881), Flow2 (0.249), Flow3 (0.251), Flow4 (0.877). Measured Variables : H1 (0.701), H2 (0.847), H3 (0.309).

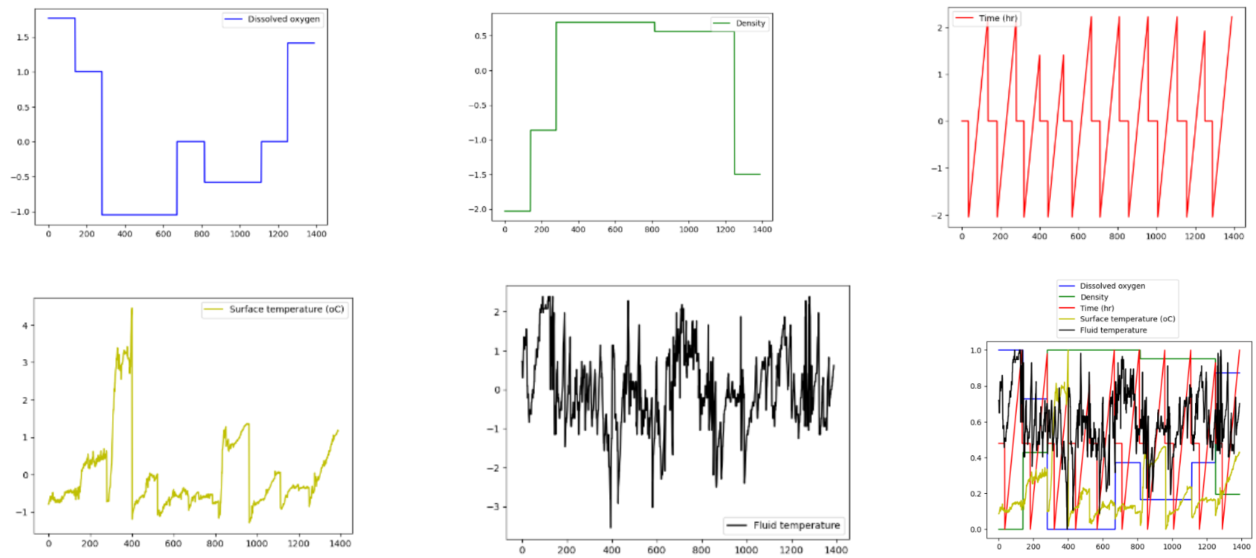
Dependent Variable: Measured Variable - Level 4

4. Results and Discussion

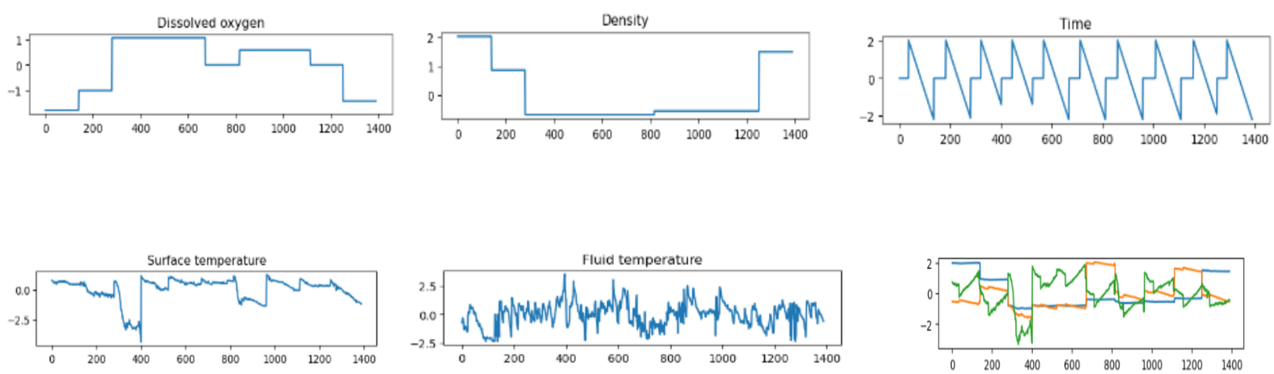
Case Study 1:

PCA Results: Features Include Density, Time, Surface Temperature, Fluid Temperature, Dissolved Oxygen.

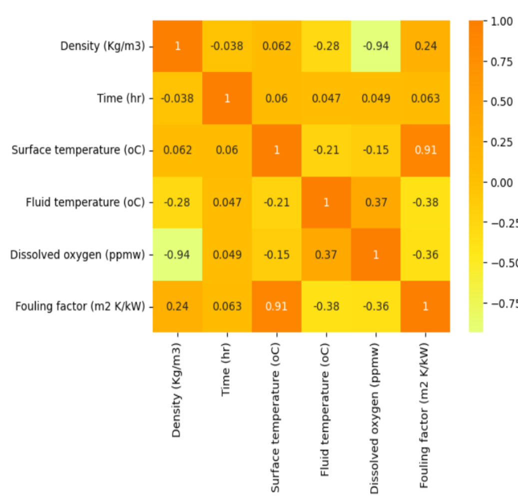
Input Signals:



Extracted Slow Feature Signals:



Correlation Matrix:



Values of Mean Square Error (MSE) and R^2:

$R^{2} = 1 - A/B$ (Closeness to the fitted regression line) (14)

where $A = \sum_{i=1}^{m}(\hat{y}_i - y_i)^2$ and $B = \sum_{i=1}^{m}(y_i - \bar{y}_i)^2$ $\hat{y}$ - predicted value, $\bar{y}$ - mean value

Without SFA, MSE = 0.0022 , R^2 = 0.913

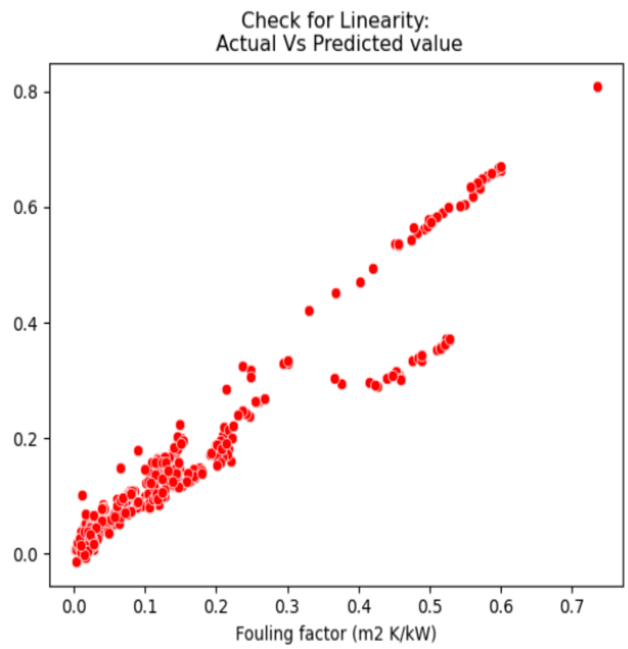


Fig.8: Plot between actual and predicted values

With SFA,

For n = 3, MSE = 0.0079, R^2 = 0.8238

For n = 4, MSE = 0.0041, R^2 = 0.9072

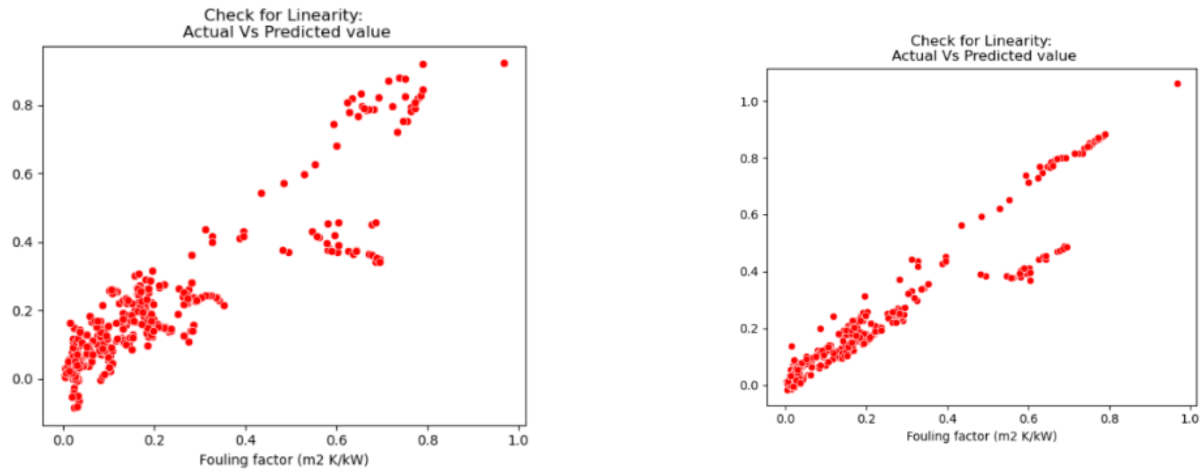
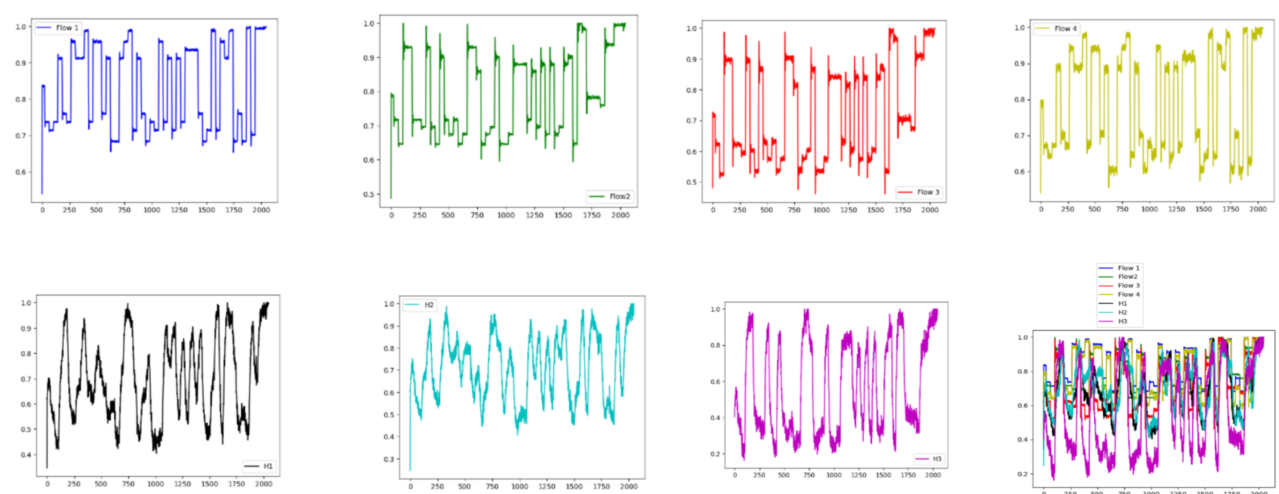


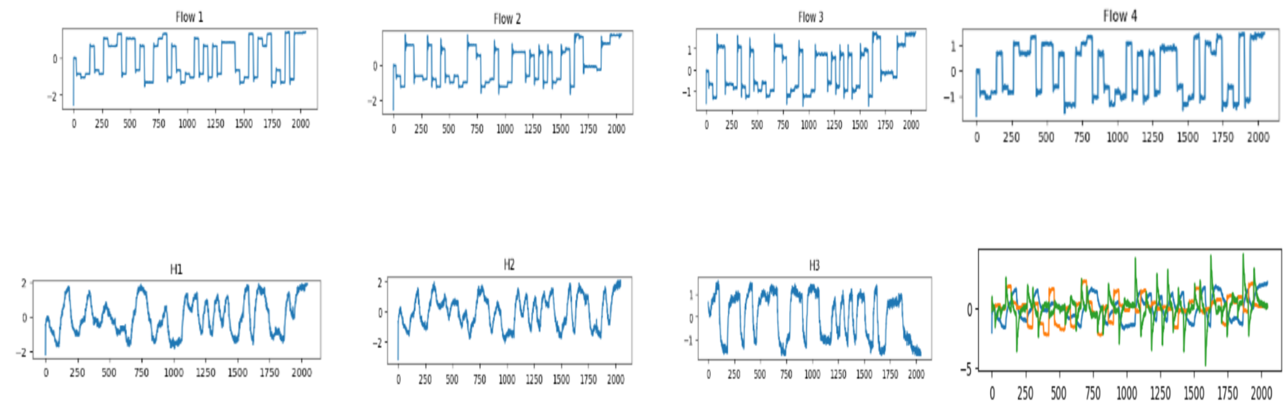
Fig.9: Plot between actual and predicted values for n=3 and n=4 respectively

Case Study 2:

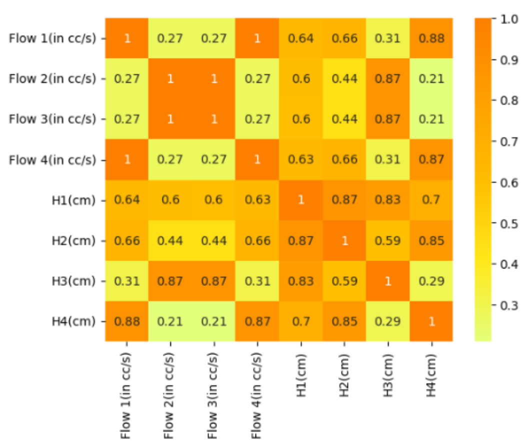
Input Signals:



Extracted Slow Feature Signals:



Correlation Matrix:



Values of Mean Square Error (MSE) and R^2:

Without SFA,

Linear Regression, MSE = 0.0031, R^2 = 0.9453.

Random Forest Regression, MSE = 0.0026, R^2 = 0.9551

With SFA,

For n = 3,

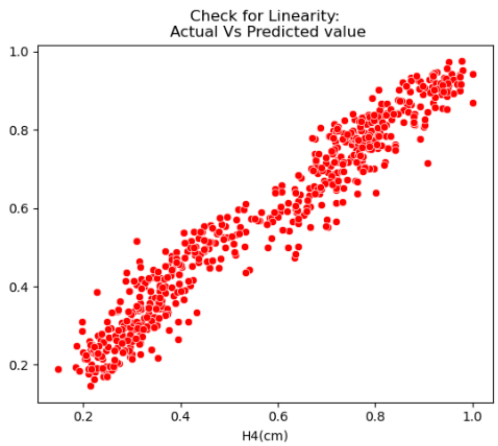
Linear Regression, MSE = 0.0065, R^2 = 0.88

Random Forest Regression, MSE = 0.005, R^2 = 0.915

For n = 4,

Linear Regression, MSE = 0.0034, R^2 = 0.94

Random Forest Regression, MSE = 0.0027, R^2 = 0.9535



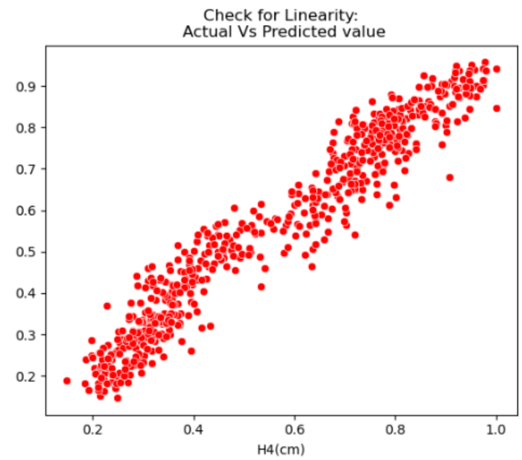


Fig.10: Plot for without and with SFA respectively.

We can see that even after reducing the number of input features from 5 to 3 and 7 to 4 in case study 1 and 2 respectively, our accuracy of prediction was better. This shows the significance of using SFA and PCA.

5. Conclusion

This report has focused on the development and implementation of soft sensors based on Slow Feature Analysis (SFA) for the prediction and monitoring of key variables in process industries. The objective was to overcome the limitations of physical sensors and harness the power of data-driven models for accurate estimation and control. To achieve this, we employed various algorithms and techniques in the model development process. Linear regression, a widely used algorithm in regression-based modeling, was utilized to establish relationships between input variables and desired outputs. The simplicity and interpretability of linear regression made it suitable for modeling certain aspects of the process.

Additionally, the Random Forest algorithm was employed as a powerful ensemble method to handle nonlinearities and complex interactions within the data. Its ability to handle high-dimensional datasets and capture complex relationships made it valuable in the development of accurate and robust soft sensors. Furthermore, Principal Component Analysis (PCA) was applied for feature selection and dimensionality reduction. By extracting the most important and uncorrelated features from the data, PCA aided in improving the efficiency and performance of the soft sensor models.

The developed soft sensor models were then implemented and evaluated in two case studies: fouling prediction in heat exchangers and the quadruple tank system. In both cases, the models demonstrated promising results, effectively estimating key variables and providing valuable insights for process monitoring and control. For the fouling prediction in heat exchangers, the soft sensor model utilizing SFA, linear regression, and PCA successfully predicted the fouling levels with a high degree of accuracy. This allowed for timely maintenance and improved efficiency of the heat exchanger system. Similarly, in the quadruple tank system, the implemented soft sensor models utilizing SFA and Random Forest demonstrated excellent performance in estimating the fluid levels and controlling the system. The models effectively handled disturbances and uncertainties, leading to stable and accurate level control.

Overall, the development and implementation of soft sensors based on SFA, linear regression, Random Forest, and PCA proved to be a valuable approach for accurate estimation and control of key variables in process industries. The successful application of these models in the case studies highlights their potential for enhancing process monitoring, optimization, and efficiency. Further research and development in this area could lead to even more advanced and robust soft sensor models for a wide range of industrial applications.

6. Future Work

In the future, there are several avenues to explore for the development and improvement of soft sensor models based on Slow Feature Analysis (SFA) in process industries. One potential direction is to investigate non-linear SFA approaches, which can capture complex relationships and dynamics within the data. By incorporating non-linear transformations and kernel-based techniques, the soft sensor models can potentially achieve higher accuracy and performance.

Another area of interest is the application of dynamic SFA, which considers the temporal dynamics of the process variables. By capturing the time-evolving patterns and trends in the data, dynamic SFA can provide a more comprehensive understanding of the process behavior and enable more accurate predictions and control. Additionally, future work can involve exploring advanced machine learning techniques, such as Artificial Neural Networks (ANN), for model development. ANN-based models have shown great potential in capturing complex relationships and non-linearities in data. Comparing the performance of ANN-based models with the existing linear regression and Random Forest models can provide insights into the effectiveness of different modeling approaches and identify the most suitable method for specific process applications.

Furthermore, incorporating additional process knowledge, such as physics-based models or expert rules, into the soft sensor models can enhance their accuracy and robustness. Hybrid models that combine data-driven approaches with first principles-based models can leverage the strengths of both methods and lead to improved predictions and control.

7. References:

1. Lin B., Knudsen J, 2006. *A systematic approach for soft sensor development* [online]. ScienceDirect.[Accessed on 15 January 2023].
2. Ehsan D, Behzad V, 2018. *Applying artificial neural networks for systematic estimation of degree of fouling in heat exchangers*[online]. ScienceDirect. [Accessed on 18 January 2023].
3. Saleh H, Amith K, Muhammad C, 2022. *Novel and robust machine learning approach for estimating the fouling factor in heat exchangers*[online]. ScienceDirect. [Accessed on 21 January 2023].
4. Asomaning, S., 1990. *The role of olefins in fouling of heat exchangers*[online]. University of British Columbia, Vancouver. [Accessed on 28 January 2023].
5. Zhang J, Corrigan J, 2020. *Integrating dynamic slow feature analysis with neural networks for enhancing soft sensor performance*[online]. ScienceDirect. [Accessed on 12 February 2023].
6. C. Shang, B. Huang, F. Yang, D. Huang, 2015. *Probabilistic Slow Feature Analysis based representation learning from massive process data for soft-sensor modeling* [online]. AIChE. [Accessed on 16 March 2023].
7. W. Laurentz, T. J. Sejnowski, 2002. *Slow feature analysis: unsupervised learning of invariances* [online]. Neural computation 14, 715-770. [Accessed on 18 February 2023].
8. Jayaram V, Piyush L, Sachin C, Lorenz T,2017 . *Development of moving window state and parameter estimators under maximum likelihood and Bayesian frameworks* [online]. Department of Chemical Engineering, Indian Institute of Technology Bombay, India. [Accessed on 26 April 2023].
9. Sudhir K, 2020. *Linear Regression Tutorial* [online]. Available from: https://www.kaggle.com/code/sudhirnl7/linear-regression-tutorial [Accessed 10 March 2023].