SLOVAK UNIVERSITY OF TECHNOLOGY
IN BRATISLAVA

FACULTY OF CHEMICAL AND FOOD TECHNOLOGY

Reg. No.: FCHPT-4622-86633

Multi-fidelity Modelling for Advanced Monitoring in Process Industries

DISSERTATION THESIS

Bratislava 2026 **Ing. Rastislav Fáber**

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

FACULTY OF CHEMICAL AND FOOD TECHNOLOGY

Reg. No.: FCHPT-4622-86633

Multi-fidelity Modelling for Advanced Monitoring in Process Industries

DISSERTATION THESIS

Study programme:	Process Control
Study field:	Cybernetics
Training workspace:	Department of Information Engineering and Process Control
Thesis supervisor:	doc. Ing. Radoslav Paulen, PhD.
Consultants:	Ing. Karol Ľubušký

Bratislava 2026 **Ing. Rastislav Fáber**

Slovak University of Technology in Bratislava
Department of Information Engineering and Process Control

Faculty of Chemical and Food Technology
Academic year: 2025/2026
Reg. No.: FCHPT-4622-86633

STU FCHPT

DISSERTATION THESIS TOPIC

Student:	**Ing. Rastislav Fáber**
Student's ID:	86633
Study programme:	Process Control
Study field:	Cybernetics
Thesis supervisor:	doc. Ing. Radoslav Paulen, PhD.
Head of department:	prof. Ing. Miroslav Fikar, DrSc.
Consultant:	Ing. Karol Ľubušký

Topic: **Multi-fidelity Modelling for Advanced Monitoring in Process Industries**

Language of thesis: English

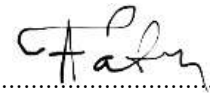
Deadline for submission of Dissertation thesis:	31. 05. 2026
Approval of assignment of Dissertation thesis:	22. 05. 2026
Assignment of Dissertation thesis approved by:	prof. Ing. Miroslav Fikar, DrSc. – Chairperson of Field of Study Board

Honour Declaration

I declare that the submitted dissertation thesis was completed on my own, in cooperation with my supervisor and co-authors, with the help of professional literature and other information sources, which are cited in the reference section of the thesis.

As the author of this dissertation thesis, I declare that I did not violate any third-party copyrights.

I further declare that large language models were used only as auxiliary tools for reviewing, and improving the clarity and language quality of the text. Their use did not replace my own scientific work, analysis, interpretation of results, or responsibility for the final form of the dissertation thesis.



Ing. Rastislav Fáber

Acknowledgment

I would like to express my sincere gratitude to Dr. Radoslav Paulen for his expert guidance, constructive feedback, and continuous support throughout this work. His advice helped me improve the quality of the thesis and stay focused on its main objectives.

I am also grateful to Ing. Karol Ľubušký for sharing practical industrial insights and for providing valuable consultations during my doctoral studies.

I would also like to thank Prof. Gabriele Pannocchia, Dr. Marco Vaccari, and Dr. Riccardo Bacci di Capaci from the University of Pisa for their guidance, valuable discussions, and collaboration during my research stay. Their support contributed significantly to the development of this work and to the preparation of our joint publications.

My thanks also go to my friends for their support, which helped me stay on the right track.

Finally, I would like to thank Dr. Martin Klaučo for sharing a LaTeX template that made the thesis writing process easier.

Abstract

Modern process industries use frequent online measurements for monitoring, control, and operational decision support. These measurements provide dense time-series data, but they often cannot capture important quality-related variables with laboratory-level accuracy. Online analyzers may contain noise, drift, calibration errors, bias, or local measurement faults. Laboratory analyses provide a more reliable reference, but they are sparse, delayed, and expensive. A model trained only on online analyzer data may therefore learn analyzer errors, while a model trained only on laboratory data usually has too few samples to describe the full process trajectory. In this thesis, we develop and evaluate a multi-fidelity framework for industrial process monitoring. We combine low-fidelity online measurements, high-fidelity laboratory samples, and low-fidelity simulation data from simplified process models. We train a low-fidelity predictor to capture the main process trend on the dense plant time grid, and then we use Gaussian Process Regression to model the difference between this prediction and the laboratory measurements. This gives a corrected soft sensor that keeps the time resolution of online data and moves the prediction closer to the laboratory reference. The proposed workflow includes data preprocessing, domain-knowledge-guided feature selection, training-testing separation, low-fidelity baseline modelling, residual correction, and fixed-threshold evaluation. In feature selection, we assess candidate variables not only by statistical performance, but also by physical meaning, measurement reliability, operational availability, and suitability for long-term deployment. We evaluate the framework on the Tennessee Eastman Process benchmark and on an industrial alkylation unit. In the data-based multi-fidelity soft-sensing study, the proposed correction gives the best results. It reduces the testing RMSE by 50.60 % for the Tennessee Eastman Process and by 23.59 % for the industrial alkylation unit, both relative to the online analyzer baseline. We also apply the same principle to simulation-supported soft sensing and anomaly detection. The proposed simulation-supported model reduces the testing RMSE at high-fidelity timestamps from 0.5165 to 0.2048. For anomaly detection, the proposed simulation-supported detector reaches sensitivity of 0.730, specificity of 0.722, accuracy of 0.7248, and a testing AUC of 0.750. The results show that multi-fidelity residual correction can combine heterogeneous industrial data sources in a practical way. The proposed framework does not replace

laboratory analysis or process expertise, but it provides a clear way to combine them for industrial soft sensing and anomaly detection.

Keywords: multi-fidelity modelling, soft sensors, residual correction, anomaly detection, process monitoring, industrial data analytics

Abstrakt

V priemyselných procesoch sa na monitorovanie a riadenie bežne využívajú online merania. Tie síce poskytujú nepretržitý tok dát v reálnom čase, no často nedosahujú presnosť laboratórnych analýz. Online analyzátory totiž môžu trpieť šumom, driftom, kalibračnými chybami či výpadkami. Laboratórne merania sú síce spoľahlivejšou referenciou, no vykonávajú sa zriedkavo, s časovým oneskorením a sú nákladné. Model postavený čisto na online dátach preto preberá ich chyby, zatiaľ čo model založený len na laboratórnych výsledkoch nemá dostatok vzoriek na zachytenie celého vývoja procesu v čase. Predložená práca navrhuje a vyhodnocuje kombinovaný (tzv. multi-fidelity) postup pre monitorovanie priemyselných procesov. Spája v ňom bežné online merania, presné laboratórne vzorky a dáta vygenerované zjednodušeným simulačným modelom procesu. Využíva pritom metódu korekcie rezíduí. Najskôr vytvoríme základný prediktor, ktorý zachytí hlavný trend procesu v čase, a následne pomocou Gaussovej procesnej regresie modeluje rozdiel medzi touto predikciou a skutočnými laboratórnymi meraniami. Výsledkom je soft-senzor, ktorý si zachováva rýchlu odozvu online dát, no presnosťou sa blíži k laboratórnej referencii. Navrhnutý postup pokrýva celý proces od spracovania dát, cez výber kľúčových veličín na základe fyzikálneho významu a prevádzkovej spoľahlivosti, až po samotné testovanie a detekciu anomálií. Metodiku sme overili na štandardnom benchmarku Tennessee Eastman Process a na reálnej priemyselnej alkylačnej jednotke. Výsledky ukázali, že korekcia modelov prináša výrazné zlepšenie. V porovnaní s čistým online analyzátorom sa predikčná chyba znížila o 50,60 % pri benchmarku a o 23,59 % pri alkylačnej jednotke. Rovnaký princíp sme úspešne aplikovali aj na modely založené na simulačných dátach pre detekciu anomálií. Po korekcii na laboratórne merania klesla predikčná chyba týchto modelov z 0,5165 na 0,2048. Pri detekcii anomálií dosiahol navrhovaný detektor porovnateľne najlepšie výsledky s hodnotou (AUC) 0,750. Výsledky potvrdzujú, že navrhovaná metodika dokáže efektívne prepojiť rôzne zdroje priemyselných dát. Tento postup nenahrádza prácu laboratórií ani expertov, ale ponúka spoľahlivý nástroj na ich prepojenie pri tvorbe soft-senzorov a hľadaní porúch.

Kľúčové slová: modelovanie s rôznou úrovňou dôveryhodnosti, soft-senzory, korekcia rezíduí, detekcia anomálií, monitorovanie procesov, priemyselná dátová analytika

Contents

Honour Declaration **iii**

Acknowledgment **v**

Abstract **vii**

Abstrakt **ix**

1 Introduction **1**

1.1 Motivation 5

1.2 General Objectives 6

1.3 Publications 7

2 Problem Formulation and Scope **11**

2.1 Industrial Data Characteristics 11

2.1.1 Multivariate Process Data 11

2.1.2 Measurement Frequency and Reliability 13

2.2 Multi-Fidelity Data Structure (MF) 14

2.2.1 Low-Fidelity Plant Data (LF) 15

2.2.2 Low-Fidelity Simulation Data (LFs) 15

2.2.3 High-Fidelity Laboratory Data (HF) 16
2.2.4 Temporal Misalignment and Pairing 17
2.3 Monitoring 18
2.3.1 Prediction Objective 18
2.3.2 Anomaly Detection Objective 18
2.3.3 Types of Anomalies 19
2.4 Ground Truth Definition (GT) 21
2.4.1 Reference Trajectory Concept 22
2.4.2 Threshold-Based Labeling 23

3 Foundations of Data-Driven Modelling and Prediction 25

3.1 Data Preprocessing 25
3.1.1 Missing Data Handling 26
3.1.2 Evident Outlier Treatment 26
3.1.3 Data Standardisation 29
3.2 Feature Engineering 30
3.2.1 Feature Subset Selection 31
3.2.2 Domain Knowledge (DK) Integration 32
3.2.3 Dimensionality Reduction 32
3.3 Model Training and Validation 34
3.3.1 Training, Validation, and Testing Splits 35
3.3.2 Cross-Validation (CV) Strategies 36
3.3.3 Performance and Optimisation 38
3.4 Regression Models 41
3.4.1 Linear and Nonlinear Model Classes 42

3.4.2 Static and Dynamic Model Structures 42
3.5 Threshold-Based Anomaly Classification 43
3.5.1 Decision Threshold Selection 44
3.5.2 Binary Detection Rule 45
3.5.3 Detector Evaluation 45

4 Approaches for Soft Sensing and Monitoring 47
4.1 First-Principles Modelling 47
4.1.1 Mass and Energy Balance Models 48
4.1.2 Advantages and Limitations 49
4.2 Data-Driven Soft Sensors 50
4.2.1 Linear Soft Sensors 50
4.2.2 Latent-Variable Soft Sensors 51
4.2.3 Sparse Regression Soft Sensors 52
4.2.4 Nonlinear Soft Sensors 52
4.3 Hybrid Modelling Approaches 55
4.3.1 Physics-Informed Models 55
4.3.2 Grey-Box Models 56
4.4 Multi-Fidelity Modelling 57
4.4.1 Co-Kriging and Gaussian-Process-Based Fusion 57
4.4.2 Residual-Correction Multi-Fidelity Models 58
4.5 Summary of Existing Approaches and Motivation for the Proposed Framework 60

5 Case Studies 63
5.1 Tennessee Eastman Process 63
5.1.1 Data Structure and Variable Definition 64

5.1.2 High-Fidelity Reference Construction 65
5.2 Alkylation Process 65
5.2.1 Data Structure and Variable Definition 66

6 Data-Based Multi-Fidelity Soft Sensing for Online Sensor Correction 67
6.1 Implementation Workflow 67
6.2 Tennessee Eastman Process Case Study 70
6.3 Industrial Alkylation Unit Case Study 76
6.4 Discussion 82

7 Physics-Guided Soft Sensing for Anomaly Detection 85
7.1 Implementation Workflow 86
7.2 Industrial Alkylation Data and Simulation Model 87
7.3 Discussion 99

8 Conclusions and Future Research 103

A Curriculum Vitae 107

B Author Publications 111

C Resumé 115

Bibliography 121

List of Figures

2.1 Illustrative example of a univariate time series, where one variable is observed over time. 12

2.2 Illustrative example of a multivariate time series, where several process variables are observed jointly over time. 12

2.3 Example of a point anomaly represented as an isolated spike. 20

2.4 Example of a contextual anomaly represented as a gradual drift. . . . 21

2.5 Example of a subsequence anomaly represented as a segment deviation. 22

3.1 Illustration of multivariate outlier detection based on Mahalanobis distance. 28

3.2 Effect of standardisation on two variables with different scales. 29

3.3 Illustration of principal component analysis (PCA). 33

3.4 Dense LF time series with overlaid sparse HF laboratory samples. . . . 35

3.5 Nested data-splitting scheme used in the multi-stage workflow. 36

3.6 Comparison of cross-validation strategies used in this work. 37

3.7 Illustration of the ROC curve and AUC. The red diagonal represents random classification. 41

4.1 Proposed multi-fidelity modelling workflow. 61

5.1 Simplified representation of the TEP benchmark. 64

5.2 Simplified alkylation process flowsheet. 66

6.1 Expanding-window CV splits for LF FS in the TEP case study. 71

6.2 Validation predictions $\hat{y}^{\text{stat}}_{\text{LF},i}$ on $\mathcal{T}_{\text{CV}}(\text{V}'')$ across CV splits for different FS methods in the TEP case study. 73

6.3 Validation RMSE distributions over repeated GP training runs for different kernels in the TEP case study. 75

6.4 Time-series comparison of LF data, HF samples, single-fidelity models, and MF models for the TEP case study. 76

6.5 Expanding-window CV splits used for LF FS in the industrial alkylation case study. 77

6.6 Validation predictions $\hat{y}^{\text{stat}}_{\text{LF},i}$ on $\mathcal{T}_{\text{CV}}(\text{V}'')$ across CV splits for different feature-selection methods in the industrial alkylation case study. . . . 78

6.7 Validation RMSE distributions over repeated GP training runs for different kernels in the industrial alkylation case study. 80

6.8 Time-series comparison of LF data, HF samples, single-fidelity models, and MF models for the industrial alkylation case study. 81

7.1 Simplified steady-state alkylation model implemented in gPROMS and used to generate the simulation dataset $\mathbf{X}^{\text{LF}_\text{s}}$ and $\mathbf{y}^{\text{LF}_\text{s}}$. The highlighted variables define the six-dimensional DK input space. 88

7.2 Equivalent simplified steady-state alkylation model implemented in AVEVA Process Simulation. 89

7.3 Histogram comparison of the six selected input variables in the industrial plant dataset $\mathbf{X}^{\text{LF}}_{\text{FS}}$ and the simulation dataset $\mathbf{X}^{\text{LF}_\text{s}}$. 90

7.4 Projection of simulation samples $\mathbf{X}^{\mathrm{LF_s}}$, industrial samples $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$, and paired HF samples $\mathbf{X}_{\mathrm{FS}}^{\mathrm{HF}}$ onto the first two principal components of the selected input space. 91

7.5 Training and testing split on the industrial alkylation time series. The dense LF analyzer signal y_i^{LF}, sparse HF laboratory values y_j^{HF}, and MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$ are shown on the plant time grid. 92

7.6 Tolerance band used for pseudo ground-truth construction around the MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$. Samples where $|y_i^{\mathrm{LF}} - \hat{y}_i^{\mathrm{ref}}| > \delta_{\mathrm{pGT}}$ are labelled as anomalous. 94

7.7 Parity plots on the TR subset. Predictions are evaluated against y_j^{HF} at the paired HF indices. 95

7.8 Parity plots on the TS subset. Predictions are evaluated against y_j^{HF} at the paired HF indices. 96

7.9 ROC curves on the TR subset for the selected anomaly detectors. The marked operating points correspond to the thresholds δ^* optimised on TR. 100

7.10 ROC curves on the TS subset for the selected anomaly detectors. The thresholds δ^* are transferred from TR without further tuning. 101

List of Tables

6.1 Average validation RMSE on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ for FS methods in the TEP case study. 72

6.2 RMSE comparison of LF data and single-fidelity models against HF data in the TEP case study. 74

6.3 Final RMSE performance against HF data in the TEP case study. Relative changes are computed with respect to the LF analyzer baseline. 75

6.4 Average validation RMSE on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ for feature-selection methods in the industrial alkylation case study. 79

6.5 RMSE comparison of LF data and single-fidelity models against HF data in the industrial alkylation case study. 79

6.6 Final RMSE performance against HF data in the industrial alkylation case study. Relative changes are computed with respect to the LF analyzer baseline. 82

7.1 Datasets used in the simulation-supported anomaly-detection case study. 89

7.2 Prediction RMSE values evaluated at paired HF indices in the simulation-supported anomaly-detection case study. 97

7.3 Confusion-matrix metrics for anomaly detection on TR and TS subsets. Sensitivity and specificity are weighted equally in the threshold optimisation, with $a = b = 1$. 98

Chapter 1

Introduction

Modern process industries operate under strict requirements on product quality, safety, energy efficiency, environmental impact, and economic performance [1]. Chemical and petrochemical plants are expected to run for long periods close to desired operating conditions, even when feed composition changes, equipment condition slowly degrades, catalysts age, maintenance actions interrupt normal operation, or disturbances arrive from upstream and downstream units [2]. These plants are therefore not static systems. Their behaviour changes over time, and the available measurements must be interpreted in this changing context. Under such conditions, reliable process monitoring is not an auxiliary task [3]. It is one of the basic requirements for stable, safe, and profitable operation [4].

The development of industrial digitalization has made large volumes of process data available. Distributed control systems, plant historians, online analyzers, laboratory information systems, and simulation environments describe the same process from different viewpoints [5]. Flow rates, temperatures, pressures, valve positions, compositions, and quality indicators are recorded over long operating periods and can be used for monitoring, control, optimization, and decision support [6, 7]. However, the availability of large datasets does not automatically lead to reliable knowledge about the plant. Industrial data are rarely clean, uniform, or complete. They contain missing values, sensor faults, calibration changes, operating-mode transitions, maintenance periods, and measurement errors [8]. For this reason, the main difficulty is not only to build a prediction model from available data. The more important difficulty is to decide how different data sources, each with different reliability and purpose, should be combined into a consistent monitoring framework [9].

This issue is especially important for quality-related process variables. In many industrial units, key quantities such as product composition, impurity concentration, recycle-stream composition, or product quality indicators cannot be measured continuously with laboratory-level accuracy. Online analyzers provide frequent information

and are useful for real-time operation, but their signals can suffer from noise, drift, bias, fouling, and maintenance-related faults [10, 11]. In this thesis, such frequent but less reliable measurements are referred to as low-fidelity (LF) data. Laboratory analyses are usually more reliable and are often treated by plant personnel as reference measurements. However, they are sparse, delayed, and expensive. They are therefore referred to as high-fidelity (HF) data. This creates a gap between the information needed for operation and the information that can be trusted. Real-time monitoring often depends on dense but imperfect LF signals, while the most reliable HF information is available only occasionally.

Soft sensors, also known as inferential sensors or virtual analyzers, address this gap by estimating hard-to-measure variables from correlated process measurements [10, 11]. A soft sensor uses available online variables, such as temperatures, pressures, flow rates, and other analyzer signals, to estimate a target variable that is unavailable, delayed, or unreliable. In industrial practice, this is useful because the soft sensor can provide a frequent estimate without installing a new physical analyzer. Classical soft sensors based on linear regression, principal component regression, and partial least squares remain attractive because they are simple, transparent, and relatively easy to maintain [12, 13]. More recent methods use nonlinear machine-learning models, including Gaussian process regression, neural networks, autoencoders, and recurrent architectures, to capture nonlinear and dynamic process behaviour [14, 15, 16]. These methods can improve prediction accuracy, but they also require more data, more tuning, and more careful validation. In industrial deployment, this additional complexity cannot be ignored.

Prediction accuracy alone is not sufficient for an industrial soft sensor. The model must also be understandable, maintainable, and based on variables that are physically meaningful and available in real operation. A model that depends on unreliable sensors, controller outputs, or variables affected by operating-policy changes may show good results on historical data but fail after maintenance, recalibration, or a change in controller configuration [17]. For this reason, process knowledge remains essential during model development. The selected inputs should not only reduce a statistical error criterion. They should also have a clear connection to the monitored variable, follow a reasonable process path, and remain trustworthy during long-term operation. A soft sensor that engineers cannot understand or maintain is unlikely to be useful, even if its offline error is low.

A related group of methods is based on prediction–correction estimation. In this setting, a process model predicts the current state or output, and new measurements are then used to correct this prediction [18]. Classical observer-based approaches, such

as Kalman filters and moving-horizon observers, follow this idea and are widely used when a suitable mechanistic model is available [19, 20]. These methods are attractive because they combine model-based prediction with measurement-based correction and can also provide uncertainty information. Their limitation is practical. For large-scale industrial units, developing, validating, and maintaining a sufficiently accurate first-principles model can be expensive, and parameter estimation under plant uncertainty is often difficult [21]. This motivates data-driven correction frameworks that preserve the prediction–correction logic, but do not require a complete mechanistic model.

A further limitation of many standard soft-sensing approaches is that they usually rely on one main data source. Models trained only on LF online analyzer data may reproduce the behaviour of the analyzer, but they may also reproduce its bias and drift. In such a case, the model does not correct the problem, instead, it learns it. Models trained only on HF laboratory data avoid this issue because they use trusted measurements, but HF samples are often too sparse to describe the full temporal behaviour of the process. Simulation models represent another possible information source. Even simplified first-principles or flowsheet models can encode mass-balance and unit-operation relationships. However, they are rarely accurate enough to reproduce the industrial plant without correction, because real units contain unmeasured disturbances, uncertain parameters, recycle effects, catalyst changes, and plant-specific operating practices. Each information source is therefore useful, but incomplete. The problem is therefore not only how to predict the monitored variable, but how to correct a dense but imperfect prediction using sparse trusted information.

This situation motivates the use of multi-fidelity (MF) modelling. In an MF setting, different information sources are combined because they do not provide the same accuracy, cost, resolution, or physical detail [22]. LF data provide temporal coverage and describe how the process evolves over time. HF data provide reference accuracy and indicate the trend that should be trusted. Simulation data can form an additional LF source because even a simplified process model may preserve the dominant physical relations between variables. The basic motivation of MF surrogate modelling is to improve prediction accuracy while limiting the need for expensive HF samples by combining many cheaper LF samples with a smaller number of HF samples [23].

In the context of this thesis, this idea is transferred from simulation-based surrogate modelling to industrial monitoring. Frequent plant measurements provide dense information about the process evolution, while sparse laboratory measurements provide the HF reference trend. The role of the LF source is to provide a trend that can later be calibrated or corrected. This interpretation is consistent with hierarchical MF modelling, where the LF model captures the general trend and the HF samples

are used for calibration [23]. For industrial applications, this distinction is important because it keeps the role of each source clear. The dense source provides continuity, while the trusted source defines the correction target.

Several MF modelling strategies have been proposed in the literature. Scaling-based approaches construct additive, multiplicative, or hybrid correction functions between LF and HF responses. Space-mapping approaches connect the input or output spaces of models with different fidelity levels. Co-kriging and Gaussian-process-based methods model the statistical relation between fidelities and can also include uncertainty in the prediction [23, 24, 25, 26, 27]. These methods show why MF modelling is attractive, but they also show why direct industrial deployment is not straightforward. They often assume a suitable relation between fidelity levels, require careful tuning, and may depend on whether LF and HF samples are available at matching or non-matching input locations.

This limitation is important for industrial data. In practice, laboratory measurements and online measurements are not always obtained under ideal sampling conditions, and simulation data rarely match plant data exactly. The relation between LF and HF sources can therefore be uncertain. If the LF source does not follow the HF trend sufficiently well, it can reduce prediction quality instead of improving it. Zhou et al. [23] show this effect in MF surrogate modelling. As the correlation between LF and HF models decreases, prediction performance generally deteriorates, and LF information may even degrade the MF model when the relation is very weak. This is directly relevant, as simulation data and online analyzer data should not be accepted blindly. They must be checked against plant behaviour and corrected using trusted measurements.

The central idea of this thesis is therefore to formulate industrial MF modelling as a practical residual-correction problem. The goal is not to replace plant measurements, and it is not to build a fully detailed mechanistic model of the plant. Instead, the proposed approach separates the monitoring task into two connected parts. First, an LF predictor captures the dominant process trend on the dense plant time grid. Second, sparse HF laboratory measurements are used to learn the difference between this prediction and the trusted reference. This structure uses LF information for trend description and HF information for correction, but it keeps the implementation simple enough for industrial monitoring. This residual-correction structure gives each data source a clear role. The correction model aligns the dense prediction with the trusted measurements and compensates for bias caused by analyzer drift, simplified simulation assumptions, or plant-model mismatch. The approach also avoids an unrestricted fusion of all available information. Only LF quantities that provide a useful trend are

allowed to enter the correction workflow. This makes the model easier to interpret and more suitable for cases where only a limited number of HF samples are available.

The same idea can support anomaly detection. Industrial anomaly detection is difficult because anomalous events are rarely labelled in historical datasets. Faults may be undocumented, operating logs may not contain exact time intervals, laboratory samples may be too sparse to define continuous labels, and online analyzer deviations may become visible only when they are compared with a trusted reference. Classical monitoring tools such as statistical process monitoring, control charts, robust covariance methods, clustering, and reconstruction-error methods can detect deviations from normal data patterns [28, 29, 30]. However, without a reliable reference, it is difficult to evaluate whether these deviations are relevant for product quality or only reflect negligible changes in operating conditions. An MF predictor can provide such a reference. If the corrected model follows the laboratory-quality trend on the plant time grid, then deviations between the online analyzer and this reference can indicate anomalous behaviour. This does not replace independently labelled fault data, and it should not be interpreted as absolute ground truth. It is a model-derived reference based on the best available information. Nevertheless, this reference is valuable in industrial situations where no manually labelled fault database exists. It allows different anomaly detectors to be compared under the same criterion and helps connect detection results with quality-related behaviour rather than only with statistical unusualness.

1.1 Motivation

The motivation for this thesis therefore comes from a realistic industrial setting. The plant has frequent LF online measurements, sparse HF laboratory references, and sometimes simplified simulation models, but these information sources are not automatically integrated. Operators often compare analyzer and laboratory values manually or through delayed correction procedures. Such corrections can be useful, but they usually arrive after the deviation has already affected operation. The aim of this thesis is to formalize this process into a structured modelling framework. The framework should improve soft sensing, support anomaly detection, and remain understandable enough for industrial use.

The main objective of this thesis is to develop and evaluate an MF framework for industrial process monitoring. The framework combines LF plant measurements, HF laboratory analyses, and simulation-based information when such information is available. The focus is on quality-related variables that are measured frequently but not always reliably online, and sparsely but more accurately in the laboratory. The

thesis does not aim to develop a universal black-box model for all industrial monitoring problems. Instead, it focuses on a practical class of problems where several imperfect information sources describe the same monitored variable and are used together.

1.2 General Objectives

The first objective is to establish a consistent problem formulation for industrial datasets with multiple fidelity levels. The thesis distinguishes between frequent LF plant data, sparse HF laboratory measurements, and simulation-generated data, and explains how their differences affect prediction and monitoring tasks. This formulation is necessary because the data sources differ not only in sampling frequency, but also in reliability, physical meaning, and use in model development. A clear formulation also prevents confusion between online analyzer values, laboratory references, simulation outputs, and corrected model predictions.

The second objective is to develop a residual-correction MF soft sensor. The framework uses a baseline predictor to describe the dominant process trend and then learns an HF correction from sparse laboratory samples. This produces a continuous estimate of the laboratory-quality output on the plant time grid. The correction is especially important because the LF predictor may capture the process trend but remain biased with respect to the laboratory reference. By modelling this discrepancy explicitly, the framework separates trend prediction from reference correction.

The third objective is to include process knowledge in the modelling workflow. Feature selection is not treated only as a statistical ranking problem, but also as an engineering task. Variables are assessed with respect to their physical relation to the monitored output, measurement reliability, availability in real operation, and suitability for long-term monitoring. This step is important because industrial models must remain useful after deployment, not only during offline testing. A compact and physically meaningful input set also improves interpretability and reduces maintenance effort.

The fourth objective is to examine how simplified simulation models can support soft sensing and anomaly detection. A reduced process model can provide physically consistent LF data and help construct surrogate predictors, even when it cannot fully reproduce the industrial unit. This objective is important for industrial practice because detailed first-principles models are often too expensive, uncertain, or difficult to maintain. A simplified model may still be useful if it captures the dominant process relationships and if its mismatch to the plant is corrected using HF information.

The fifth objective is to extend the MF framework toward anomaly detection. The corrected MF prediction is used as a reference trajectory for defining anomaly labels on the plant time grid, and different detector families are compared under this common reference. This objective addresses the common industrial situation where explicit fault labels are not available. The resulting evaluation does not claim to replace independent fault annotation, but it provides a practical way to study detector behaviour when laboratory-aligned reference information is available.

1.3 Publications

Journal articles

1. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Data-based multi-fidelity modeling for online sensors correction.* Computers & Chemical Engineering, Vol. 211, Article 109665, 2026.

2. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Physics-Guided Soft Sensing for Anomaly Detection: Simulation Surrogates in Industrial Processes.* Chemical Engineering Research and Design (ChERD), (submitted).

3. **M. Wadinger, R. Fáber, E. Pavlovičová, and R. Paulen**: *Carbon neutral greenhouse: Economic model predictive control framework for education.* European Journal of Control, Vol. 83, Article 101297, 2025.

Conference contributions

1. **R. Fáber, R. Valo, M. Roman, and R. Paulen**: *Towards Temperature Monitoring in Long-Term Grain Storage.* In 2022 Cybernetics & Informatics (K&I), pp. 1–6, 2022.

2. **R. Fáber, K. Ľubušký, M. Mojto, and R. Paulen**: *Enhancing Industrial Data Analysis through Machine Learning-based Classification of Petrochemical Datasets.* In 49th International Conference of the Slovak Society of Chemical Engineering SSCHE 2023, Slovak Society of Chemical Engineering, Bratislava, Slovakia, p. 160, 2023.

3. **R. Fáber, K. Ľubušký, and R. Paulen**: *Machine Learning-based Classification of Online Industrial Datasets.* In R. Paulen and M. Fikar (Eds.), Proceedings of the 2023 24th International Conference on Process Control, IEEE, Slovak University of Technology in Bratislava, pp. 132–137, 2023.

4. **R. Fáber, M. Mojto, K. Ľubušký, and R. Paulen**: *From Data to Alarms: Data-driven Anomaly Detection Techniques in Industrial Settings.* In Flavio Manenti and G. V. Rex Reklaitis (Eds.), 34th European Symposium on Computer Aided Process Engineering / 15th International Symposium on Process Systems Engineering, Vol. 53, pp. 2857–2862, 2024.

5. **R. Fáber, M. Mojto, K. Ľubušký, and R. Paulen**: *Integrated Data Analytics and Regression Techniques for Real-time Anomaly Detection in Industrial Processes.* In 12th IFAC Symposium on Advanced Control of Chemical Processes, pp. 320–325, 2024.

6. **R. Fáber, M. Klaučo, K. Ľubušký, and R. Paulen**: *Integrating Linear and Nonlinear Models for Enhanced Process Monitoring.* In R. Paulen, M. Fikar, and J. Oravec (Eds.), Proceedings of the 2025 25th International Conference on Process Control, IEEE, pp. 1–6, 2025.

7. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Multi-fidelity Modeling for Process Optimization in Refinery Operations.* In 51st International Conference of the Slovak Society of Chemical Engineering SSCHE 2025, Slovak Society of Chemical Engineering, Bratislava, Slovakia, p. 140, 2025.

8. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Improving Process Monitoring via Dynamic Multi-Fidelity Modeling.* In 14th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems, Elsevier, pp. 199–204, 2025.

9. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Soft-Sensor-Enhanced Monitoring of an Alkylation Unit via Multi-Fidelity Model Correction.* Systems and Control Transactions, Vol. 4, pp. 1389–1395, 2025.

10. **R. Fáber, K. Ľubušký, and R. Paulen**: *A Simplified Framework for Process Model Development in Industrial Simulation Environments.* In 52nd International Conference of the Slovak Society of Chemical Engineering SSCHE 2026, Slovak Society of Chemical Engineering. Bratislava, Slovakia, 2026. (accepted)

11. **R. Fáber, K. Ľubušký, and R. Paulen**: *A Hybrid Data-Driven Approach for the Optimization of an Industrial Alkylation Unit.* In *ESCAPE 36 – European Symposium on Computer Aided Process Engineering*, Sheffield, UK. June 21–24, 2026, *Syst. Control Trans.*. (accepted)

12. **R. Fáber, M. Vaccari, R. Bacci di Capaci, G. Pannocchia, and R. Paulen**: *A Multi-Fidelity Residual-Correction Framework for Industrial Soft Sensing.* In 24th European Control Conference (ECC) in Reykjavík, Iceland. July 07–10, 2026. (accepted)

13. **P. Arbetová, R. Fáber, K. Ľubušký, and R. Paulen**: *A Practical Framework for Process Anomaly Detection Analysis in Multivariate Time Series.* In 23rd IFAC World Congress. August 23–28, 2026. Busan, Republic of Korea. (accepted)

Chapter 2

Problem Formulation and Scope

2.1 Industrial Data Characteristics

Industrial process monitoring relies on time-series data collected from plant measurements. These data can be represented either as a single measured variable or as multiple variables observed simultaneously. Figure 2.1 shows a univariate time series, where a single variable is monitored over time. In contrast, Figure 2.2 shows a multivariate time series, where multiple variables are recorded simultaneously. These two representations form the basis for how industrial data are interpreted and modelled in practice. While univariate signals are useful for direct inspection of individual measurements, most industrial monitoring tasks rely on multivariate data structures.

2.1.1 Multivariate Process Data

Industrial processes are monitored through many measured variables recorded over time. These variables usually include temperatures, pressures, flow rates, valve positions, levels, and composition measurements. When collected together, they form a multivariate time series, where each timestamp is described by a vector of process measurements rather than by a single value. This means that the state of the process cannot be understood from one variable alone. Instead, it must be interpreted through the joint behaviour of several variables that are physically linked by material flow, energy transfer, reaction progress, recycle structure, and process control actions.

Let $\mathbf{x}_i \in \mathbb{R}^p$ denote the vector of p measured process variables at time t_i. The full process dataset can then be written as an input matrix

$$\mathbf{X} \in \mathbb{R}^{n \times p}, \tag{2.1}$$

where each row n corresponds to one timestamp and each column p corresponds to one measured variable. The monitored output variable is denoted by y. In the present work,

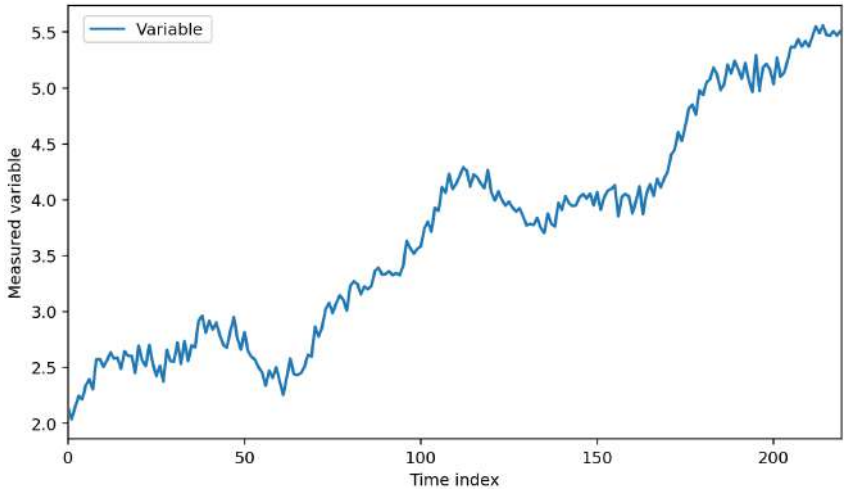


Figure 2.1: Illustrative example of a univariate time series, where one variable is observed over time.

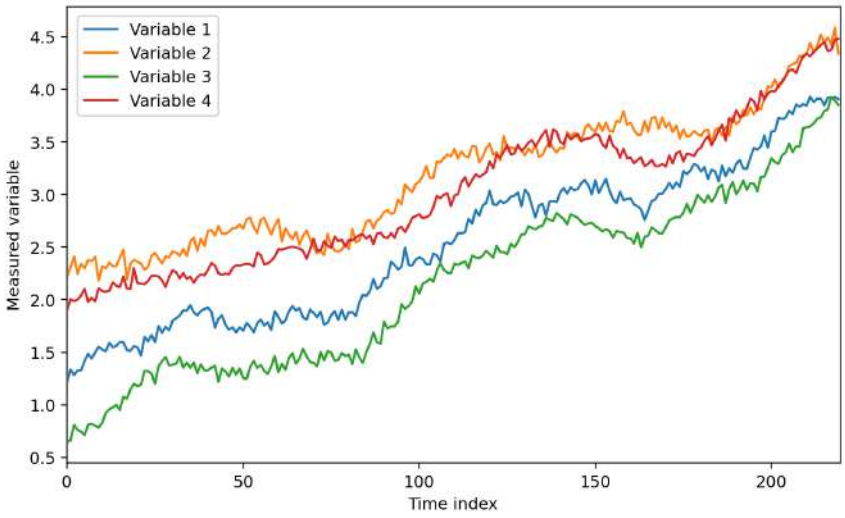


Figure 2.2: Illustrative example of a multivariate time series, where several process variables are observed jointly over time.

this output is a quality-related quantity that is measured online by a plant analyzer and only occasionally verified by a laboratory analysis. Based on this representation, different analysis perspectives can be considered.

A useful distinction must be made between univariate and multivariate data analysis. In a univariate setting, only one signal is studied, for example the analyzer signal of the monitored output. This can be useful when detecting abrupt jumps, drifts, or offsets directly in the measured output. However, it ignores the fact that the output is a consequence of other process variables. In a multivariate setting, the output is analysed together with the available process measurements. This is more informative because anomalous behaviour in the output may originate from changes in feed composition, recycle conditions, pressure, temperature, or flow rates, even if the output signal alone does not immediately appear extreme. For this reason, process monitoring in industrial systems is inherently multivariate.

Another important property of industrial multivariate data is correlation. Many process variables are not independent. Some are linked by physics, such as mass balances or separation behaviour, while others are linked by feedback control, where one signal reacts to another through controller action. As a result, several variables may carry overlapping information. This makes industrial datasets high-dimensional but not necessarily information-rich in every dimension. Careful feature selection or dimensionality reduction is therefore necessary before model training.

Finally, industrial data are not collected under ideal laboratory conditions. The measured time series may include missing values, flat segments, outliers, maintenance effects, controller interventions, and regime changes. Consequently, the data matrix $\mathbf{X}$ is not only large, but also heterogeneous in quality and relevance. This is one of the main reasons why the structure of the dataset must be defined clearly before any modelling or monitoring task is attempted. Additionally, industrial datasets also differ in how frequently and how reliably the measurements are obtained, which introduces an additional layer of complexity.

2.1.2 Measurement Frequency and Reliability

A central characteristic of industrial data is that not all measurements are collected with the same frequency or the same level of trust. In addition to their multivariate structure, industrial datasets are characterized by differences in measurement frequency and reliability. These differences are not a minor detail, rather, they fundamentally shape the monitoring problem.

Standard plant sensors and online analyzers usually provide frequent measurements. They may be recorded every few seconds, every minute, or at another regular process sampling interval. These measurements are immediately available to operators and control systems, which makes them useful for real-time monitoring and process control. However, frequent availability does not guarantee accuracy. Online measurements can be affected by noise, calibration drift, fouling, communication errors, maintenance issues, or slow bias development. For this reason, such measurements are treated here as low-fidelity data. They are dense in time, but their quality may be limited.

In contrast, laboratory analyses are typically much less frequent. A sample must be taken from the plant, transported, prepared, analysed, and reported. This procedure is slower and more expensive, so the resulting measurements are sparse in time. In many industrial applications, laboratory data are available only once per day, once per week, or even less frequent. Despite this low frequency, laboratory measurements are generally treated as the most reliable reference because they are performed under controlled conditions using validated methods. They are therefore regarded as high-fidelity data, and commonly used for low-fidelity data calibration.

The practical difficulty is that monitoring requires both frequency and accuracy, but these two properties often come from different sources. The plant has frequent measurements that may be wrong, and accurate measurements that arrive too late and too rarely. This mismatch creates the main challenge of the present work. If one uses only LF data, the model may learn fast but unreliable behaviour. If one uses only HF data, the model may be accurate at a few points but unable to capture process evolution over time. A meaningful monitoring framework should therefore combine both frequency and reliability instead of choosing only one of them. This observation naturally leads to a multi-fidelity view of industrial data, where measurements are categorized based on their frequency and reliability.

2.2 Multi-Fidelity Data Structure (MF)

Based on the differences in measurement frequency and reliability, industrial datasets can be organized into multiple fidelity levels. Each level provides complementary information about the same underlying process. In this work, three main data sources are considered: plant measurements, simulation data, and laboratory measurements.

2.2.1 Low-Fidelity Plant Data (LF)

The first data source is the plant dataset, denoted as low-fidelity plant data. It consists of online process measurements collected directly from the industrial unit. These measurements include the input variables used for prediction, together with the online analyzer signal of the monitored output.

Formally, let

$$\mathbf{X}^{\mathrm{LF}} \in \mathbb{R}^{n_{\mathrm{LF}} \times p} \tag{2.2}$$

denote the matrix of plant input measurements, and let

$$\mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{n_{\mathrm{LF}}} \tag{2.3}$$

denote the corresponding online measurements of the monitored output. The LF timestamp set is written as

$$\mathcal{T}_{\mathrm{LF}} = \{t_i\}_{i \in \mathcal{I}_{\mathrm{LF}}}, \qquad \mathcal{I}_{\mathrm{LF}} = \{1, \ldots, n_{\mathrm{LF}}\}. \tag{2.4}$$

Each row of $\mathbf{X}^{\mathrm{LF}}$ describes the measured state of the plant at one timestamp t_i.

The main strength of LF plant data is temporal density. Because the measurements are frequent, they describe short-term process evolution, transport effects, controller responses, and local disturbances. This makes them particularly useful for real-time monitoring and for the construction of dynamic predictors. Their weakness is reliability. The monitored output may deviate from the true process value because of sensor drift, bias, analyzer faults, or poor calibration. In other words, LF plant data provide the process trajectory that is available in real operation, but not necessarily the trajectory that should be trusted. This distinction is critical. In the present work, LF data are not discarded. They remain the main source of temporal information, but they must be interpreted with caution and, where possible, corrected using higher-quality information. While plant data provide real process trajectories, additional information may be obtained from simulation models.

2.2.2 Low-Fidelity Simulation Data (LFs)

A second low-fidelity source may be available from process simulation. In this thesis, simulation data are generated from a simplified first-principles model built to represent the dominant process behaviour over a selected operating region. These data are denoted as low-fidelity simulation data, or LFs.

The simulation dataset can be written as

$$\mathbf{X}^{\mathrm{LF_s}} \in \mathbb{R}^{n_{\mathrm{LF_s}} \times p_s}, \qquad \mathbf{y}^{\mathrm{LF_s}} \in \mathbb{R}^{n_{\mathrm{LF_s}}}. \tag{2.5}$$

Unlike plant data, simulation data are not recorded from real operation. We generate them by sampling the model over selected input ranges and solving the corresponding steady-state or dynamic model equations. This makes it possible to produce datasets that are much larger than the available high-fidelity laboratory dataset, often by one or several orders of magnitude.

$$n_{\mathrm{LF}} \ll n_{\mathrm{LF_s}}. \tag{2.6}$$

The purpose of these data is not to replace plant measurements, but to provide dense and physically consistent trends that may not be clearly visible in noisy industrial operation.

Simulation data have a different role from plant LF data. Plant LF data reflect what actually happened in the unit, including disturbances and sensor problems. Simulation data reflect what the process is expected to do according to the simplified physical model. This means that LFs data can help reveal dominant material-balance or process-structure trends, even if the simulation model does not reproduce the plant perfectly. At the same time, simulation data are also low-fidelity. The simplified model may omit kinetics, disturbances, unknown feedback effects, or plant-specific operating details. In the present context, they provide an additional information layer that can support surrogate modelling and, after suitable correction, improve monitoring performance. However, neither plant nor simulation data can fully guarantee accuracy, which motivates the use of laboratory measurements as a reference.

2.2.3 High-Fidelity Laboratory Data (HF)

The third data source is the set of laboratory measurements, treated as high-fidelity data. These measurements are assumed to provide the most reliable available representation of the monitored output.

Let

$$\mathbf{y}^{\mathrm{HF}} \in \mathbb{R}^{n_{\mathrm{HF}}} \tag{2.7}$$

denote the laboratory measurements of the same physical quantity measured online by the LF analyzer, where

$$n_{\mathrm{HF}} \ll n_{\mathrm{LF}}. \tag{2.8}$$

The corresponding timestamp set is

$$\mathcal{T}_{\mathrm{HF}} = \{\tau_j\}_{j\in\mathcal{I}_{\mathrm{HF}}}, \qquad \mathcal{I}_{\mathrm{HF}} = \{1, \ldots, n_{\mathrm{HF}}\}. \tag{2.9}$$

In many industrial cases, laboratory measurements provide only the output variable and not a complete set of synchronized input variables. Therefore, when building

HF-based predictors, the required input vector must usually be taken from plant LF measurements recorded at the nearest plant timestamp to the laboratory sample. This means that HF outputs are accurate, but their pairing with process inputs is indirect and depends on temporal matching.

The low number of HF samples makes model building difficult. A predictor trained only on laboratory data may capture the average output trend, but it usually cannot learn short-term process behaviour or local dynamics in a reliable way. This is particularly problematic when the laboratory sampling interval is much longer than the characteristic response time of the process. For this reason, HF data play a special role in this thesis as the main reference for ground truth. Their value lies in calibration, correction, validation, and final performance assessment. Because these three data sources differ not only in reliability but also in how they are collected over time, their temporal alignment must be addressed explicitly.

2.2.4 Temporal Misalignment and Pairing

The three aforementioned data sources generally do not share the same time grid. This temporal misalignment is one of the main technical difficulties of MF industrial modelling. Plant LF data are sampled frequently and almost regularly. Laboratory HF data are sampled sparsely, often at irregular intervals, and may also be delayed relative to the actual process state. Simulation data usually do not follow the plant clock at all, because they are generated offline by sampling model inputs over a chosen operating region rather than by tracking the real time evolution of the plant. Even when two datasets measure the same physical quantity, they are rarely aligned sample by sample. As a result, we cannot simply stack LF, LFs, and HF data into a single table without first defining a pairing rule.

For LF and HF data, the common practical assumption is that each laboratory sample is assigned to the nearest available plant timestamp, or to a short local time window around it. This creates matched pairs

$$\left(\mathbf{x}_{i(j)}^{\mathrm{LF}}, y_j^{\mathrm{HF}}\right), \tag{2.10}$$

where $i(j)$ denotes the plant index paired with the j-th laboratory sample. This pairing is simple, but it has limitations. If the process exhibits transport delay, dead time, mixing effects, or long residence time, the nearest timestamp may not reflect the true process state that produced the laboratory result. In such cases, interval averaging, lag compensation, or physically informed synchronization may be more appropriate.

For simulation data, the situation is different. Simulation samples are usually paired

through the input space rather than directly through time. If the simulator is built in the same selected feature space as the plant data, then the simulation predictor can be evaluated on plant inputs and used to generate a plant-aligned predicted trajectory. This predicted trajectory can then be corrected using the sparse HF samples. In this way, simulation data become indirectly aligned with the plant timeline through the surrogate model, not through online timestamps.

The main point is that MF industrial data do not naturally form one clean, synchronized dataset. They must be converted into a common analysis structure through explicit matching rules. The quality of this step affects all later stages, including model training, validation, anomaly labeling, and interpretation of results.

2.3 Monitoring

Once the MF data structure is defined, the monitoring task can be formulated in terms of prediction and anomaly detection.

2.3.1 Prediction Objective

The first objective of monitoring is prediction of the true output trajectory from the available process measurements. More precisely, the aim is to estimate, at each plant timestamp, the most reliable possible value of the monitored quality-related variable.

In a standard single-fidelity setting, this would mean training a predictor from one data source only. However, this approach is rarely sufficient in complex industrial processes. The prediction objective therefore combines all available fidelities. The model should also preserve the temporal richness of the plant data, while moving the predicted output toward the laboratory-quality trend. In practical terms, the desired predictor should provide a continuous estimate on the plant time grid and should behave like an online signal, but with a quality closer to HF measurements than to the LF analyzer. This is the role of the MF soft sensor developed in the related chapters. While prediction focuses on reconstructing the expected process behaviour, monitoring also requires identifying deviations from this behaviour.

2.3.2 Anomaly Detection Objective

The second objective is anomaly detection. Here, our aim is not only to estimate the monitored output, but also to identify when the observed plant signal no longer

follows the expected process behaviour. In this setting, an anomaly denotes a sample or a time interval where the measured signal deviates from the normal operating pattern. Such behaviour may result from sensor drift, miscalibration, short spikes, process disturbances, changes in operating conditions, or other local deviations from the expected trend. In industrial applications, anomaly labels usually come from confirmed fault records, operator logs, maintenance reports, or comparison with trusted HF measurements. For each evaluated sample, the anomaly detection task assigns a binary label. A normal sample belongs to the expected operating behaviour, while an anomalous sample indicates a deviation that requires attention. These labels allow the detector output to be compared with the reference classification of the data. The detector therefore does not only produce a prediction error or a continuous score, but also a decision on whether the sample should be treated as normal or anomalous.

This objective is narrower than general fault diagnosis. The goal is not to identify the physical cause of every abnormal event or to isolate the faulty unit. Instead, the goal is to detect when the monitored signal becomes inconsistent with the expected process trend. The anomaly detector therefore acts as a warning layer that highlights suspicious behaviour in the measured output and supports further inspection by process engineers.

2.3.3 Types of Anomalies

To formalize the anomaly detection objective, we first define the types of anomalous behaviour relevant to the case studies considered in this thesis. In industrial monitoring, anomalies may appear as isolated deviations, gradual shifts from the expected trend, or short intervals with a persistent offset from the reference. These patterns motivate the distinction between point outliers, contextual drift, and subsequence outliers. These three categories are not mutually exclusive, instead, they provide a general framework for describing common types of anomalous behaviour in industrial time series.

2.3.3.1 Point Anomalies

A point anomaly is an isolated observation that strongly deviates from the expected behaviour at a single timestamp, as illustrated in Figure 2.3. In practice, this may correspond to a sudden spike, a short communication glitch, a single wrong analyzer reading, or an abrupt offset introduced during maintenance or recalibration. Point anomalies are the simplest type to define because they are local in time. Their effect is concentrated in one sample or in a very short interval. In a univariate output

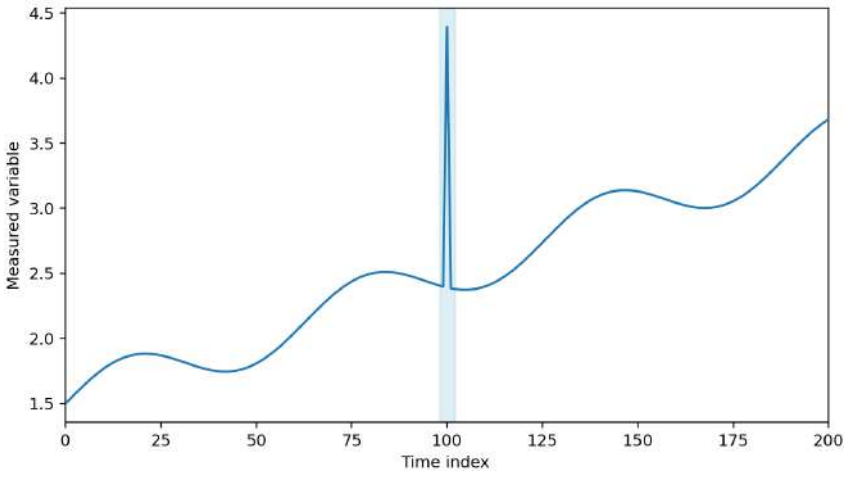


Figure 2.3: Example of a point anomaly represented as an isolated spike.

signal, they often appear as sharp peaks or drops. From an analytical perspective, point anomalies can be further distinguished as univariate or multivariate. Univariate anomalies are detected within a single signal and correspond to extreme deviations relative to its expected range. In contrast, multivariate anomalies arise from unusual combinations of variables, where individual measurements may appear normal when considered separately, but their joint configuration is inconsistent with typical process behaviour. Although point anomalies are easy to visualize, they should not be treated as the only relevant form of anomalous behaviour. Many important industrial problems do not appear as isolated spikes, but as longer and more structured deviations.

2.3.3.2 Contextual Anomalies

A contextual anomaly is an observation that may look acceptable when viewed alone, but becomes anomalous when interpreted in its process context. In industrial monitoring, this often appears as a gradual drift or persistent bias relative to the expected process trend, as illustrated in Figure 2.4. The measured value may remain numerically plausible, yet it is inconsistent with laboratory data, related process variables, or a trusted model-based reference. This type of anomaly is important in industrial sensing because a drifting analyzer may still produce smooth and seemingly reasonable measurements. The issue is not that the signal becomes physically impossible, but that it no longer represents the true process state. Contextual anomalies are therefore harder to detect from the output alone and usually require a multivariate or reference-based

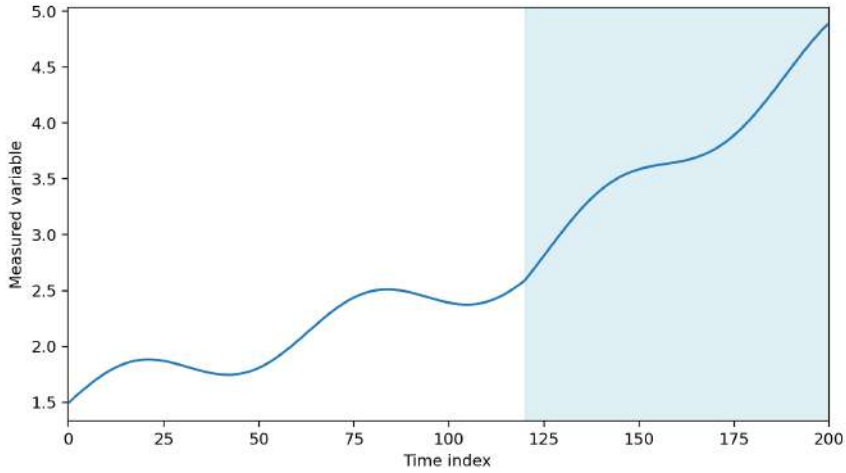


Figure 2.4: Example of a contextual anomaly represented as a gradual drift.

comparison.

2.3.3.3 Subsequence Anomalies

A subsequence anomaly is a short interval of consecutive samples whose collective behaviour is anomalous, even if the individual points are not extreme enough to be flagged separately. Examples include a temporary offset, an oscillatory segment, a short period of sensor miscalibration, or a local disturbance that pushes the signal away from the normal operating pattern for several minutes or hours, as illustrated in Figure 2.5. This class is especially relevant in continuous processes. Many anomalous events persist over several samples and should be interpreted as a segment rather than as a single isolated point. For this reason, anomaly detection methods in process monitoring must be able to capture not only magnitude deviations, but also time structure.

2.4 Ground Truth Definition (GT)

The anomaly detection objective defined above requires a reference against which the online signal can be compared. In this work, the ground truth for predictive modelling tasks is directly available from HF laboratory measurements, which provide reliable and physically meaningful reference values, although only at sparse time instants. This

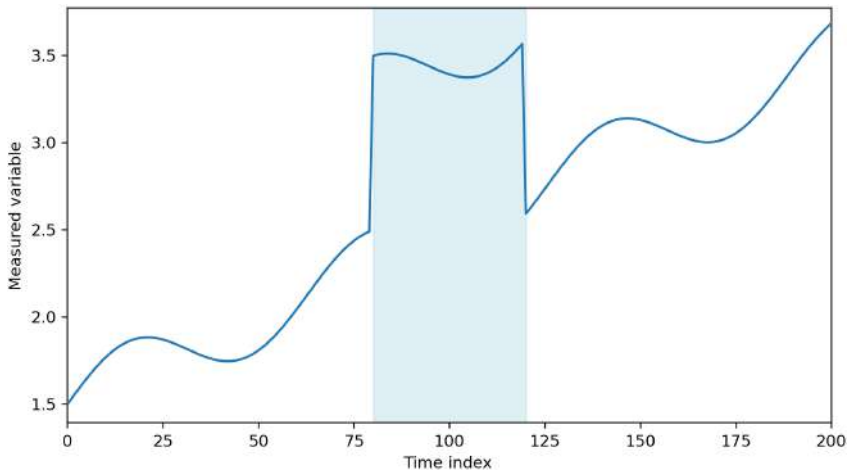


Figure 2.5: Example of a subsequence anomaly represented as a segment deviation.

makes supervised prediction tasks straightforward in terms of target definition.

The situation is fundamentally different for anomaly detection, where no complete and consistently available ground-truth labeling exists over the full plant time grid. In standard supervised anomaly detection, such a reference would come from manually validated labels, operator records, maintenance logs, or trusted high-fidelity measurements. In the considered industrial setting, these sources are not available in a form suitable for dense time-wise supervision. The laboratory data provide reliable information, but only at sparse timestamps, and therefore cannot label the full plant time grid directly.

2.4.1 Reference Trajectory Concept

We use a reference trajectory as an intermediate step between sparse laboratory measurements and full time-grid anomaly labels. The reference trajectory is a model-based estimate of the monitored output evaluated at all plant timestamps. Its role is to approximate the expected laboratory-quality behaviour of the output under normal operation. In this thesis, we construct this trajectory using a validated multi-fidelity predictor, which combines dense low-fidelity plant information with sparse high-fidelity laboratory samples [31]. The reference trajectory is provides a continuous estimate that follows the plant time grid while remaining corrected toward the laboratory reference. This makes it suitable for defining pseudo ground-truth labels (pGT) when manually validated anomaly labels are missing.

We do not treat the reference trajectory as a perfect description of the true process state. It is a practical and reproducible approximation of the expected output trend. Its quality depends on the validation performance of the underlying MF predictor and on the available laboratory samples. Therefore, the anomaly labels derived from this trajectory should be interpreted as pseudo labels, not as confirmed fault records. Their purpose is to support a consistent comparison of anomaly detectors under the same reference definition.

2.4.2 Threshold-Based Labeling

Once the reference trajectory is defined, anomaly labels can be assigned by comparing the measured LF plant output with this reference.

Let $\hat{y}_i^{\mathrm{ref}}$ denote the reference value at plant timestamp t_i, and let y_i^{LF} denote the corresponding online analyzer measurement. A sample is labeled as anomalous whenever the absolute deviation exceeds a selected decision threshold:

$$|y_i^{\mathrm{LF}} - \hat{y}_i^{\mathrm{ref}}| > \delta. \tag{2.11}$$

If the inequality is satisfied, the ground-truth label is set to one; otherwise, it is set to zero:

$$y_i^{\mathrm{pGT}} = \begin{cases} 1, & \text{if } |y_i^{\mathrm{LF}} - \hat{y}_i^{\mathrm{ref}}| > \delta, \\ 0, & \text{otherwise.} \end{cases} \tag{2.12}$$

The decision threshold δ defines what is still considered acceptable agreement between the online measurement and the expected process trend. In this thesis, the δ is chosen from the available HF–LF discrepancy, so that the tolerance band reflects the typical historical difference between sparse laboratory values and their paired online analyzer readings. This makes the labeling rule process-specific and grounded in the available data, rather than arbitrary.

Threshold-based labeling has two main advantages. First, it converts the monitoring problem into a clear binary decision problem. Second, it allows all candidate anomaly detectors to be evaluated against the same reference labels on the same plant timeline. Its limitation is that the labels depend on the quality of the reference trajectory and on the δ bounds definition. The resulting pGT should therefore be understood as systematic, data-based, and reproducible, but not identical to a complete set of validated plant fault records (true GT).

This formulation establishes a consistent framework for integrating MF data, defining monitoring objectives, and evaluating anomaly detection methods in the following chapters. The first step prepares the data for modelling. This includes the preprocessing steps from Chapter 3, the LF–HF pairing through the already defined index mapping $i(j)$, and the construction of a compact feature space. The second step trains a LF baseline model. Depending on the selected path, this baseline is obtained from plant data, from the online analyzer, or from a simulation surrogate. The third step computes the discrepancy between the LF baseline and the HF laboratory samples at the paired timestamps. The fourth step learns this discrepancy and evaluates the corrected prediction on the full LF plant time grid. The same workflow supports soft sensing and anomaly detection. For soft sensing, the corrected signal is evaluated against $\mathbf{y}^{\mathrm{HF}}$ at the available HF timestamps. For anomaly detection, the corrected signal is used as a reference trajectory on $\mathcal{T}_{\mathrm{LF}}$, and deviations from this reference are converted into pseudo ground-truth labels using the threshold-based definition.

Chapter 3

Foundations of Data-Driven Modelling and Prediction

The previous chapter established the structure of the industrial dataset, including the definition of low-fidelity plant data $\mathbf{X}^{\mathrm{LF}}, \mathbf{y}^{\mathrm{LF}}$, simulation data $\mathbf{X}^{\mathrm{LF_s}}, \mathbf{y}^{\mathrm{LF_s}}$, and high-fidelity laboratory measurements $\mathbf{y}^{\mathrm{HF}}$, together with their respective time grids. Based on that setting, this chapter introduces the main methodological concepts that support the later development of prediction and monitoring models.

We establish the main ideas that appear repeatedly in the later chapters and explains why these ideas are relevant in industrial applications. In such settings, the quality of a predictor depends not only on the selected regression method, but also on data quality, input definition, validation strategy, and the way anomalous behaviour is interpreted. For that reason, the chapter begins with data preprocessing, then moves to feature engineering, model training and validation, regression models, and finally basic anomaly-detection concepts.

3.1 Data Preprocessing

Industrial data are rarely ready for direct model development. In particular, the online LF plant dataset $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$ may distort feature selection, bias model parameters, and reduce the reliability of the final predictor. In this thesis, preprocessing refers to the set of steps used to prepare the online dataset for a later analysis. The goal is not to make the data artificially perfect, but to preserve the main process behaviour while reducing the effect of clearly corrupted or numerically problematic records. This distinction is important. A predictor trained on untreated industrial data may fit accidental measurement noise instead of actual process relationships, while excessively aggressive cleaning may remove meaningful variability and make the model unrealistically optimistic.

Preprocessing therefore serves two roles. First, it improves the numerical stability of later modelling steps. Second, it defines the operating region from which the predictor learns. In industrial applications, this second role is especially important, because later model performance depends strongly on whether the cleaned dataset still reflects the main variability of the real process.

3.1.1 Missing Data Handling

Missing values occur primarily in the LF plant dataset $\mathbf{X}^{\mathrm{LF}}$, where individual sensor readings may be unavailable at certain timestamps $t_i \in \mathcal{T}_{\mathrm{LF}}$. A predictor cannot be trained reliably if a large fraction of the input matrix is incomplete, because most regression methods assume that each sample is described by a complete regressor vector. The simplest solution is to remove all incomplete samples. This may be acceptable when only a very small part of the dataset is affected and when the deleted samples are isolated. In long industrial datasets, however, such deletion can remove useful information and break the continuity of the time series. This becomes even more important when dynamic models are used, because one missing value may affect several later regressors constructed from lagged samples.

For this reason, we handle missing data conservatively. We remove clearly unusable records when they belong to long corrupted segments or to non-representative operating periods, such as shutdowns, startups, or maintenance. For isolated missing samples, however, we avoid aggressive deletion and use interpolation or by another simple local approximation. Our goal is not to reconstruct the exact unknown value, because that is usually impossible. The practical aim is to avoid introducing artificial information into the data before model training.

The way missing data are filled should match the purpose of the model. In this work, we prefer simple and clear methods. These methods are easier to explain and easier to implement in practice. In an industrial setting, operators must be able to understand how the data were treated. If the imputation method is too complex, it can hide data problems instead of solving them. This can lead to results that look correct but are difficult to trust or interpret during operation.

3.1.2 Evident Outlier Treatment

We remove evident outliers from the LF input space $\mathbf{X}^{\mathrm{LF}}$, where anomalous samples correspond to operating conditions that deviate from the dominant process trend. If such points are used directly in model fitting, they can strongly affect the resulting

predictor, especially in least-squares-based regression where large residuals receive high weight. A common way to detect unusual multivariate samples is to compare each observation with the historical distribution of the dataset. In this thesis, this idea is represented by the Minimum Covariance Determinant (MCD) method [29]. MCD provides a robust estimate of the data centre and covariance structure. Unlike the ordinary sample mean and covariance, it is less sensitive to extreme points and therefore gives a more reliable description of the main operating space. Once these robust quantities are available, the distance of each sample from the centre can be measured by the Mahalanobis distance:

$$d_i = \sqrt{(\mathbf{x}_i^{\mathrm{LF}} - \boldsymbol{\mu})^\top \mathbf{S}^{-1} (\mathbf{x}_i^{\mathrm{LF}} - \boldsymbol{\mu})}, \tag{3.1}$$

where $\boldsymbol{\mu}$ is the robust mean vector and $\mathbf{S}$ is the robust covariance matrix.

If this distance is too large relative to the historical spread of the data, the sample can be treated as an outlier. In practice, this means that the point lies far from the main region of normal operation, not only in one variable, but in the joint multivariate space. This is important because a sample may look acceptable in each individual variable and still represent an unusual combination of measurements.

The treatment of evident outliers depends on the purpose of the analysis. During predictor development, the aim is usually to reduce the influence of clearly corrupted records so that the fitted model reflects the dominant process trend rather than accidental faults. During monitoring, however, anomalous behaviour becomes the quantity of interest. The same type of point that is removed during training may later be precisely what the monitoring system should detect. For this reason, outlier treatment must be described carefully and consistently, because it defines what the model learns as normal behaviour.

The aforementioned multivariate outlier detection is shown in Figure 3.1. The illustrated example forms a two-dimensional space where the main operating region is captured by a confidence ellipse around a robust centre. Points inside the ellipse represent normal operation. Points outside the ellipse are considered anomalous because they do not follow the overall data pattern. This example shows that detection is not based on simple limits for each variable. Instead, it depends on how variables change together. A point can lie within the range of each variable separately, but still be an outlier when the joint behaviour is taken into account.

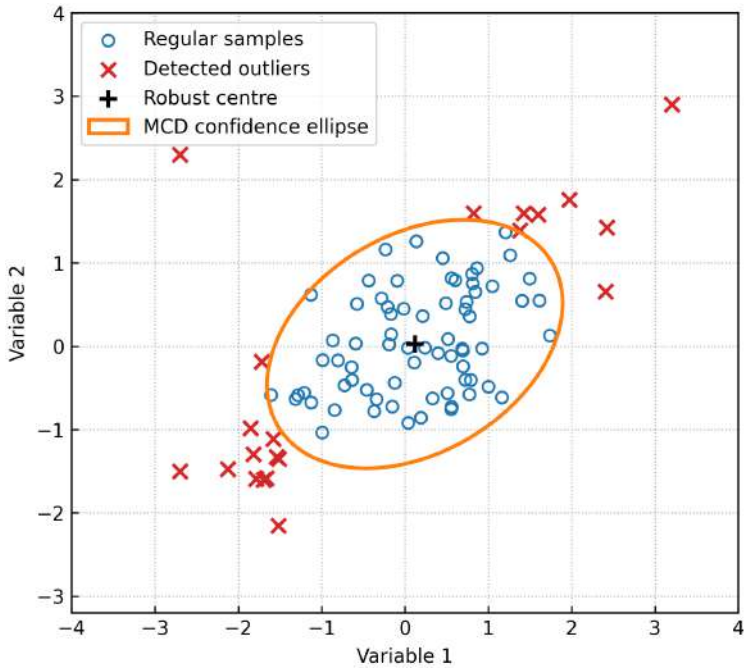


Figure 3.1: Illustration of multivariate outlier detection based on Mahalanobis distance.

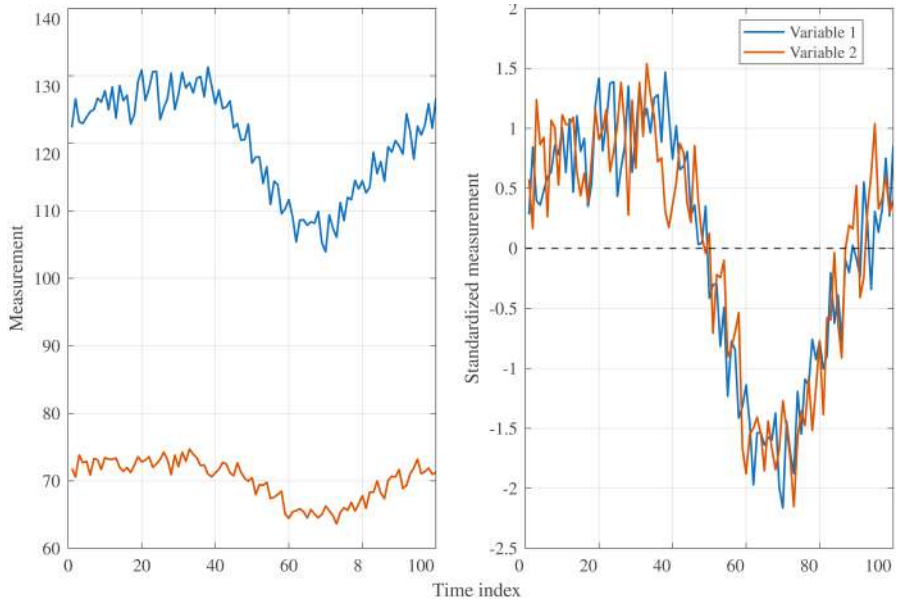


Figure 3.2: Effect of standardisation on two variables with different scales.

3.1.3 Data Standardisation

Standardisation is applied during preprocessing to both the input matrix $\mathbf{X}^{\mathrm{LF}}$ and the output variable $\mathbf{y}^{\mathrm{LF}}$ (and, where applicable, $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$) prior to model training. Available variables may differ strongly in magnitude. A flow rate, a pressure, and a composition measurement may have very different units and numerical ranges. If these online values are used directly, variables with large numerical magnitude may dominate the training criterion even if they are not more informative from a process perspective.

To avoid this effect, the variables are standardised before model fitting. Standardisation means that each variable is centred by its mean and scaled by its standard deviation. For a variable x, the standardised form is

$$x^{\mathrm{std}} = \frac{x - \bar{x}}{s_x}, \tag{3.2}$$

where $\bar{x}$ is the sample mean and s_x is the sample standard deviation.

The effect of this transformation is illustrated in Figure 3.2. In the original data, the variables have different offsets and numerical ranges. After standardisation, the underlying trend and relative structure of the data are preserved, meaning that

standardisation does not change the information content.

After this transformation, each variable is centred around zero and scaled to unit variance. This improves numerical conditioning, makes regression coefficients easier to compare, and allows dimensionality-reduction methods to reflect correlation structure rather than the original scale. Without scaling, latent-variable methods may simply follow the largest numerical variables instead of the most informative ones.

Standardisation is also important for model evaluation. When the output is standardised, evaluation criteria become dimensionless and can be compared more consistently across models within the same case study. This does not replace interpretation in physical units, but it provides a common numerical basis for model selection and comparison.

3.2 Feature Engineering

In the context of this thesis, feature selection (FS) is performed on the preprocessed LF input space $\mathbf{X}^{\mathrm{LF}}$ and results in a reduced subset of relevant variables used consistently across plant and simulation datasets. The available data matrix may contain many measured variables, but not all of them contribute equally to the monitored output. Some variables carry overlapping information, some are only weakly related to the output, and some may appear useful statistically while being unsuitable in practice. This step ensures that both plant data $\mathbf{X}^{\mathrm{LF}}$ and simulation data $\mathbf{X}^{\mathrm{LF_s}}$ are defined in a consistent feature space, which is required for the proposed framework introduced later.

Our objective is to construct a compact and informative input space that captures the dominant process behaviour. This improves predictive performance, reduces model complexity, and makes the final predictor easier to interpret and maintain in real-time operation. FS therefore aims to identify a subset of relevant variables. This is particularly important in industrial systems, where variables are often strongly correlated due to mass balances, energy balances, recycle structures, and control interactions. As a result, a large input set may increase model complexity without providing additional useful information. A suitable subset should remain stable across different data splits, should not rely on accidental correlations, and should preserve enough physical meaning for interpretation. If the selected variables change significantly when the data are slightly perturbed, the model is likely based on weak or unstable relationships. A predictor based on unstable input selection may perform well in an offline study but fail under changing operating conditions or when measurements

become unavailable.

We do not treat FS as a purely statistical task. Data-driven methods are used to rank candidate variables, but the final selection is constrained by process knowledge. This ensures that the resulting predictor is not only accurate on historical data, but also meaningful, reliable, and maintainable in operation. The following subsections describe the main components of this workflow.

3.2.1 Feature Subset Selection

This part focuses on methods that select variables directly from the original input space. These approaches aim to reduce the number of inputs while keeping predictive performance, without transforming the variables into a latent representation.

Least Absolute Shrinkage and Selection Operator (LASSO) introduces a penalty that shrinks weak coefficients toward zero:

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \|\boldsymbol{\theta}\|_1, \tag{3.3}$$

where λ controls the strength of penalization. As λ increases, the method removes more variables by forcing their coefficients to zero. This allows LASSO to perform prediction and variable selection at the same time. This method works directly with the original variables, which helps preserve interpretability. It is useful when the input space is moderately large and when a sparse predictor is required. However, strong collinearity limits its reliability. If several variables carry similar information, LASSO may select one of them without a clear physical reason, and the selected subset may change across different data splits.

Stepwise Regression (SR) follows a similar goal but uses an explicit search procedure. Forward selection starts with an empty input set and adds variables one by one. At each step, it selects the variable that gives the largest improvement in validation error. Backward elimination starts from the full set and removes variables that contribute the least. The procedure stops when further changes no longer improve the model. These methods are simple and easy to apply, and they provide a practical way to rank candidate variables. However, they show the same weakness as LASSO in the presence of strong correlations. Small changes in the data may lead to different selected subsets. For this reason, these approaches are used here as ranking tools rather than as final selectors.

3.2.2 Domain Knowledge (DK) Integration

A variable can appear statistically useful in historical data and still be a poor choice for practical deployment. This may happen when the signal is difficult to maintain, poorly calibrated, strongly influenced by operator actions, sensitive to control policy rather than process state, or only indirectly related to the monitored output. For this reason, statistical ranking alone is not sufficient enough.

Process knowledge provides the second filter. In the proposed workflow considered in this thesis, candidate variables are first ranked statistically. They are then checked from an engineering point of view. A process engineer assesses whether the variables have a clear physical relation to the monitored output, whether they are reliable in operation, whether they are available continuously, and whether they are suitable for long-term use. Variables that fail this check are replaced by more suitable candidates from the ranked list. This step is important because the goal is not to construct a black-box predictor from whatever variables happen to reduce historical error. The goal is to build a predictor whose inputs can be justified both statistically and physically. This is especially relevant in industrial systems, where a theoretically accurate model may still be unusable if it depends on a tag that is frequently recalibrated, missing, or strongly affected by controller tuning.

3.2.3 Dimensionality Reduction

Even after FS, the retained variables may still contain noticeable correlation. Dimensionality reduction addresses this by replacing the original variables with a smaller number of transformed variables. These transformed variables are easier to handle numerically and may capture the dominant structure of the data more compactly than the original measurements.

Principal Component Analysis (PCA) constructs new variables, called principal components (PCs), as linear combinations of the original inputs. For an input matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, PCA replaces these p original variables with k retained PCs, where $k < p$. The reduced representation can be written as

$$\mathbf{X} \approx \mathbf{T}\mathbf{P}^{\top}, \tag{3.4}$$

where $\mathbf{T} \in \mathbb{R}^{n \times k}$ contains the retained principal component scores, and $\mathbf{P} \in \mathbb{R}^{p \times k}$ contains the corresponding loadings. The columns of $\mathbf{T}$ represent the transformed variables used instead of the original inputs. These components are orthogonal to each other and are ordered according to the amount of input variance they explain.

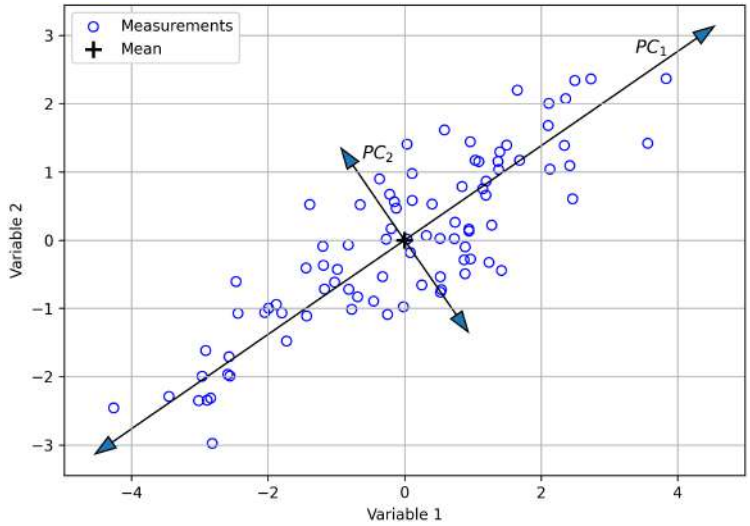


Figure 3.3: Illustration of principal component analysis (PCA).

The first component explains the largest part of the variance, the second explains the largest remaining part, and so on. If only the first few components are retained, the dimension of the problem is reduced while most of the dominant structure of the input data is preserved. This is useful when many variables carry overlapping information or when the original input space is poorly conditioned numerically.

The main limitation of PCA is that it is unsupervised. It does not consider the output variable when constructing the latent directions. A PC may explain a large portion of input variance and still be only weakly related to the prediction objective. PCA is therefore best understood as a compact representation of the input space rather than as a fully output-oriented feature construction method.

The geometric interpretation of PCA is illustrated in Figure 3.3. In this example, we consider two input variables, where each point represents one measurement, and their alignment shows the correlation between them. PCA transforms this representation into a new coordinate system defined by the PCs. The first PC_1 follows the direction of maximum variance in the data and aligns with the longest axis of the data cloud. The second PC_2 is orthogonal to PC_1 and captures the remaining variation. In this two-variable example, the number of PCs is equal to the number of original variables, so no dimensionality reduction is performed. The transformation only rotates the coordinate system to align with the dominant structure of the data. In general, for $\mathbf{X} \in \mathbb{R}^{n \times p}$ with $p > 2$, PCA allows the input space to be approximated using only the

first $k < p$ PCs. In that case, dimensionality reduction is achieved by retaining only the components that explain most of the variance.

Partial Least Squares (PLS) is similar to PCA, but with one important difference. PCA considers only variation in the input variables, whereas PLS also uses the relation between inputs and output. For an input matrix $\mathbf{X}$ and output vector $\mathbf{y}$, the basic decomposition can be written as

$$\mathbf{X} \approx \mathbf{T}\mathbf{P}^{\top}, \qquad \mathbf{y} \approx \mathbf{T}\mathbf{c}, \tag{3.5}$$

where $\mathbf{T}$ contains latent variables selected to be relevant for predicting $\mathbf{y}$, $\mathbf{P}$ contains the input loadings, and $\mathbf{c}$ contains the output regression weights. Because of this supervised character, PLS is often more directly useful for prediction than PCA. It is especially attractive when the number of candidate inputs is relatively large and when strong collinearity is present. In such cases, PLS can compress the input space while preserving the information that is most relevant for the regression task.

The main trade-off is interpretability. Both PCA and PLS reduce dimensionality, but each latent variable mixes information from several original measurements. As a result, the final predictor may be numerically efficient but less transparent than a predictor built directly on a small subset of physical variables.

3.3 Model Training and Validation

The availability of HF data $\mathbf{y}^{\mathrm{HF}}$ constrains model training, because these measurements exist only on the sparse set $\mathcal{T}_{\mathrm{HF}} \subset \mathcal{T}_{\mathrm{LF}}$. Therefore, we evaluate and compare models only at paired LF–HF timestamps, where reliable reference values are available. For each HF sample at time τ_j, we use the previously defined mapping $i(j)$ to identify the closest LF timestamp. To reduce the effect of short-term analyzer noise, we represent the LF value by a local average around this matched index, using the window $[i(j) - 5,\, i(j) + 5]$ when the neighbouring samples are available. This local averaging gives a more stable LF–HF comparison than a single online analyzer value. The relation between dense LF measurements and sparse HF laboratory samples is shown in Figure 3.4.

After defining the input space, we fit the model. However, the model is useful only if it generalizes to unseen data. Industrial datasets make this difficult because they contain strong temporal dependence, gradual changes in operating conditions, and only

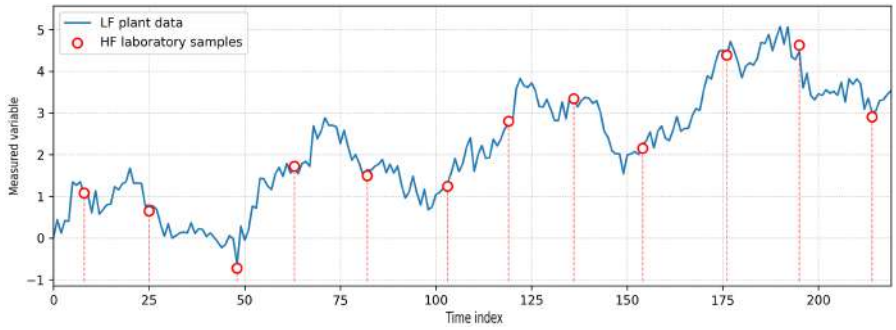


Figure 3.4: Dense LF time series with overlaid sparse HF laboratory samples.

a limited number of high-quality reference samples. For this reason, the way we divide the data into training and evaluation sets becomes a central part of the methodology. The validation strategy determines which conclusions we can trust. If model design or tuning uses information that would not be available at prediction time, the reported performance becomes optimistic and does not reflect real deployment conditions. This issue becomes more critical in multi-stage workflows, where we perform FS, single-fidelity modelling, and MF correction in sequence. Each stage must therefore use data that remain independent of later evaluation steps.

3.3.1 Training, Validation, and Testing Splits

Based on the validation requirements described above, the available data are divided into training (TR), validation (V), and testing (TS) subsets. The training set is used to estimate model parameters. The validation set is used to compare model structures, tune hyperparameters, and prevent overfitting. The testing set remains untouched until the final stage and serves only for the final performance evaluation.

In the workflow considered in this thesis, nested splits are introduced. One part of the data is reserved for final testing, another part is used for MF-level validation, and an inner part is used for FS and single-fidelity cross-validation. This structure enforces a strict separation of information. Each stage uses only the data that would be available at that point in practice. Without this separation, model performance may appear artificially high because of hidden information leakage between stages. The overall data-splitting procedure is illustrated in Figure 3.5.

- At Level 0, the full set of available LF timestamps $\mathcal{T}_{\mathrm{LF}}$ defines the preprocessed dataset used for model development.

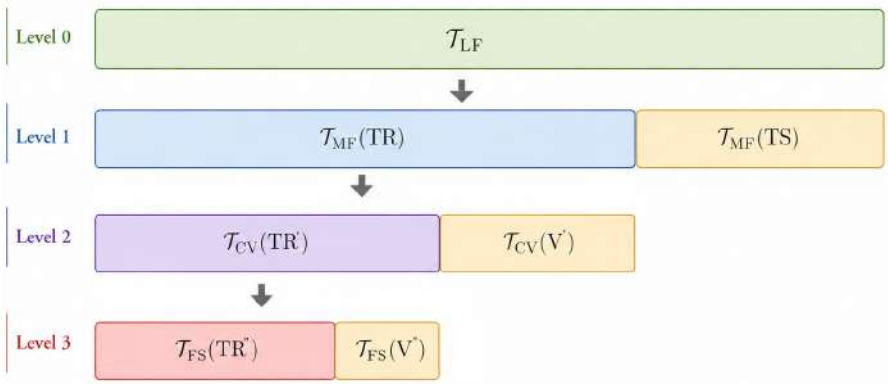


Figure 3.5: Nested data-splitting scheme used in the multi-stage workflow.

- At Level 1 (TR & TS), we define the outer split that separates the data into final MF training and testing subsets and final predictor evaluation. The MF testing set remains completely independent and is used only for the final performance evaluation of the proposed framework. All model design decisions are therefore restricted to the MF training set.
- At Level 2 (TR′ & TS′), we introduce a second split within the MF training set for MF-level cross-validation. This level supports the selection and tuning of the MF model while preserving independence from the final test set.
- At Level 3 (TR″ & TS″), we introduce an inner split within the MF training subset. This level is used for FS and single-fidelity model design, including cross-validation. The corresponding subsets are contained within the Level 2 training data, which ensures that no information from validation or testing sets is used during earlier modelling stages.

This hierarchical structure enforces a strict separation of information across all levels. Each stage uses only the data that would be available at that point in practice, which prevents information leakage and ensures that the final evaluation remains unbiased.

3.3.2 Cross-Validation (CV) Strategies

The nested structure defined above determines how validation is performed at each level of the workflow. In practice, a single validation split is often not sufficient to judge whether a model choice is stable. Instead, training and validation are repeated over several partitions of the available data. This reduces the dependence on one particular

Figure 3.6: Comparison of cross-validation strategies used in this work.

split and provides a more reliable comparison between candidate models. For industrial time series, the validation strategy must respect the chronological structure of the data. Random reshuffling mixes past and future observations and leads to overly optimistic results. For this reason, CV is performed using expanding-window schemes, where the TR set grows over time and validation is carried out on the immediately following segment. This setup reflects the real deployment scenario, in which only past data are available when predicting future operation.

At the MF level (Level 2 in Figure 3.5), CV is used to compare and tune MF model structures while preserving independence from the final test set. At the inner level (Level 3), CV supports FS and single-fidelity model design. In both cases, validation is performed only within the corresponding TR subset, which ensures that no information leaks from higher levels of the hierarchy. For sparse HF data, standard k-fold validation may still be applied, because the main limitation is the number of samples rather than their temporal density. In contrast, LF data require time-aware validation strategies due to their sequential nature. The use of different validation schemes for LF and HF data is therefore justified by their different roles and structures. These two validation strategies are illustrated in Figure 3.6. For LF time-series data, expanding-window CV (top) respects chronological order: the training window grows over time and validation is performed on the following segment. For sparse HF data, k-fold CV (bottom) can use the limited samples more efficiently by rotating the validation fold across the available observations.

3.3.3 Performance and Optimisation

A predictor should be evaluated by a quantitative measure. In the regression part of this thesis, the prediction error is defined through the Mean Squared Error (MSE):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2. \tag{3.6}$$

MSE measures the average squared deviation between predicted and observed values. Because the residuals are squared, larger errors receive a higher penalty, which is desirable in quality-related monitoring where large deviations are more critical than small ones.

For interpretation, we use the Root Mean Squared Error (RMSE), defined as

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}}. \tag{3.7}$$

RMSE has the same units as the output variable, which makes the magnitude of the prediction error easier to interpret in physical terms. This is particularly important in industrial applications, where the model output corresponds to a measurable process variable. In the MF framework considered later, RMSE is evaluated primarily at HF timestamps. This follows from the data structure introduced earlier, where HF measurements $\mathbf{y}^{\mathrm{HF}}$ represent the most reliable reference but are available only on the sparse set $\mathcal{T}_{\mathrm{HF}}$. A model may follow the dense LF signal closely and still deviate from the laboratory trend.

For anomaly-detection tasks, regression error alone is not sufficient. A detector usually produces a continuous deviation or score, denoted here by s_i. In the proposed framework, this score may correspond to the absolute deviation between a measured signal y_i and a reference trajectory $\hat{y}_i^{\mathrm{ref}}$,

$$s_i = |e_i|. \tag{3.8}$$

A decision threshold δ then determines whether the sample is treated as normal or anomalous. Smaller threshold values usually increase the number of detected anomalies, while larger threshold values make the detector more conservative. For a defined δ, the detector produces binary decisions that can be compared with reference labels. Let the reference label indicate whether the sample is truly normal or anomalous, and let the detector output indicate whether the sample is classified as normal or anomalous. The result can be summarized by a confusion matrix:

	Detected anomaly	Detected normal
Reference anomaly	TP	FN
Reference normal	FP	TN

(3.9)

where TP denotes true positives, FN denotes false negatives, FP denotes false positives, and TN denotes true negatives. True positives are correctly detected anomalies. False negatives are missed anomalies. False positives are normal samples incorrectly flagged as anomalous. True negatives are normal samples correctly left unflagged.

Several metrics can be computed from this matrix. Accuracy measures the overall fraction of correct decisions:

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{3.10}$$

Sensitivity, also called recall or true positive rate (TPR), measures how many reference anomalies are detected:

$$\mathrm{Sens} = \mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{3.11}$$

Specificity measures how many reference normal samples are correctly identified as normal:

$$\mathrm{Spec} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{3.12}$$

Precision measures how many raised alarms are correct:

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}. \tag{3.13}$$

The False Positive Rate (FPR) measures the fraction of normal samples that are incorrectly flagged:

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} = 1 - \mathrm{Spec}. \tag{3.14}$$

These metrics describe different aspects of detector behaviour. Sensitivity is important when missed anomalies are costly. Specificity is important when false alarms reduce operator trust. Precision is useful when the practical meaning of each alarm must be assessed. Accuracy gives a compact summary, but it can be misleading when normal and anomalous samples are strongly imbalanced. For this reason, the later anomaly-detection studies report several metrics instead of relying on accuracy alone.

The Receiver Operating Characteristic (ROC) curve evaluates the detector over all possible threshold values [32]. It plots TPR against FPR as δ changes. A detector with good separation between normal and anomalous samples reaches high TPR values while keeping FPR low. A detector with poor separation produces a curve close to the diagonal line, which corresponds to weak discrimination.

The Area Under the ROC Curve (AUC) summarizes the ROC curve into a single threshold-independent value:

$$\mathrm{AUC} = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\, d\mathrm{FPR}. \tag{3.15}$$

A value close to one indicates strong separation between normal and anomalous samples, while a value close to 0.5 indicates weak discrimination. In this thesis, ROC–AUC is used together with fixed-threshold metrics. ROC–AUC compares the overall ranking ability of detectors, while sensitivity, specificity, and accuracy describe the final detector behaviour after the threshold has been selected.

Threshold optimisation is needed because the final detector must operate at one selected threshold. In this thesis, the threshold is selected on the TR subset and then kept fixed for validation or testing. The threshold is selected by maximizing a weighted objective,

$$W(\delta) = a\,\mathrm{Sens}(\delta) + b\,\mathrm{Spec}(\delta), \tag{3.16}$$

where $\mathrm{Sens}(\delta)$ and $\mathrm{Spec}(\delta)$ are computed from the confusion matrix obtained at threshold δ. The weights a and b define the relative importance of detecting anomalies and avoiding false alarms. Equal weights give the same importance to sensitivity and specificity, while larger a values favour anomaly detection and larger b values favour false-alarm reduction. The optimal threshold can then be defined as

$$\delta^* = \arg\max_{\delta} W(\delta). \tag{3.17}$$

Later in the thesis, this threshold-selection problem is solved using the Nelder–Mead simplex method [33]. This method does not require gradient information, which is useful because $W(\delta)$ is computed from discrete classification counts and therefore changes in a non-smooth way as the threshold varies. After δ^* is selected on the training subset, it is applied without further tuning to the independent test subset. This prevents the detector performance from being overestimated by tuning the threshold directly on the test data.

Other metrics can also be considered, but RMSE remains the main choice in this thesis for regression-based soft sensing because it is consistent with least-squares-based model fitting and provides a directly interpretable measure of prediction error. For anomaly detection, ROC–AUC is used as a threshold-independent metric, as illustrated in Figure 3.7. It evaluates how well the detector separates normal and anomalous samples over all possible threshold values. TR and V errors are compared within the nested splitting structure introduced in Figure 3.5. A large gap between TR and

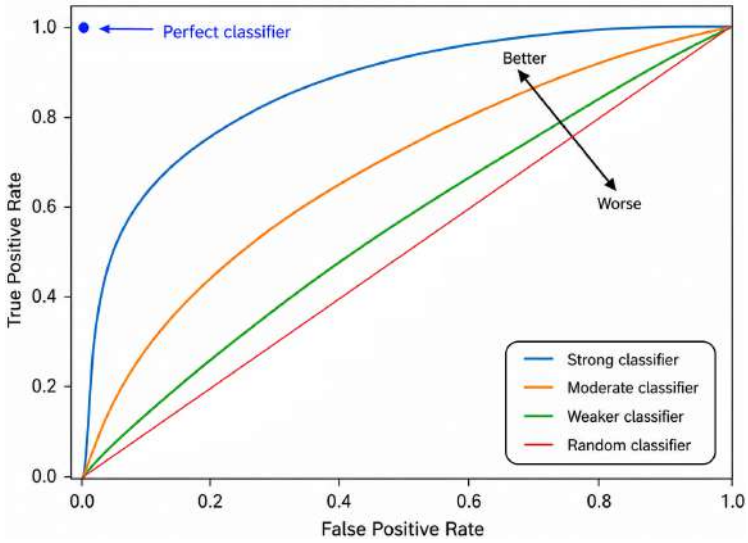


Figure 3.7: Illustration of the ROC curve and AUC. The red diagonal represents random classification.

V error indicates that the model is too flexible for the available data. Conversely, consistently high errors suggest that the model structure or selected features do not capture the process relationship sufficiently well.

3.4 Regression Models

After preprocessing, feature engineering, and data splitting, the relation between the selected inputs and the monitored output can be modelled explicitly. In general form, the prediction problem can be written as

$$y_i = F(\boldsymbol{\phi}_i) + \varepsilon_i, \tag{3.18}$$

where $\boldsymbol{\phi}_i$ denotes the selected feature vector at time t_i, $F(\cdot)$ is the unknown relation between inputs and output, and ε_i represents modelling error and measurement uncertainty.

The objective of regression modelling is to construct an approximation $\hat{F}$ such that

$$\hat{y}_i = \hat{F}(\boldsymbol{\phi}_i), \tag{3.19}$$

where $\hat{y}_i$ is the predicted value of the monitored output. Different regression methods

differ mainly in how they approximate $\hat{F}$, how much flexibility they allow, and how strongly they depend on data quality.

3.4.1 Linear and Nonlinear Model Classes

The simplest class is formed by linear models, where the prediction is expressed as a linear combination of selected regressors:

$$\hat{y}_i = \theta_0 + \sum_{q=1}^{r} \theta_q \phi_{i,q}. \tag{3.20}$$

Here, θ_0 is the intercept, θ_q are model coefficients, $\phi_{i,q}$ is the q-th regressor at sample i, and r is the number of regressors. Linear models are useful because they are transparent, stable, and easy to validate, especially when reliable reference data are limited.

Nonlinear models remove this linear restriction and allow a more flexible mapping,

$$\hat{y}_i = \hat{F}(\boldsymbol{\phi}_i). \tag{3.21}$$

This flexibility can improve prediction when the process contains nonlinear effects, but it also increases the risk of overfitting and usually makes interpretation more difficult. For this reason, nonlinear models should be used only when they provide a consistent improvement under proper validation.

3.4.2 Static and Dynamic Model Structures

Regression models can also be divided according to how they treat time. A static model uses only the current feature vector,

$$\boldsymbol{\phi}_i = \mathbf{x}_i. \tag{3.22}$$

A dynamic model also uses past inputs and, in some cases, past outputs:

$$\boldsymbol{\phi}_i = [\mathbf{x}_i, \mathbf{x}_{i-1}, \ldots, \mathbf{x}_{i-z}, y_{i-1}, \ldots, y_{i-l}], \tag{3.23}$$

where z and l denote the input and output lag orders. Dynamic structures are important in continuous industrial processes because the monitored output may depend on previous operating conditions, transport delays, recycle effects, and process memory.

The model order must be chosen carefully. Too few lags may miss important dynamics, while too many lags increase model complexity and sensitivity to noise. This trade-off is commonly addressed using information criteria that balance model fit and complexity.

One widely used measure is the Akaike Information Criterion (AIC), defined as

$$\mathrm{AIC} = 2n_\theta - 2\ln(\hat{L}), \tag{3.24}$$

where n_θ is the number of estimated parameters and $\hat{L}$ is the maximized likelihood of the model. The first term penalizes model complexity, while the second term rewards goodness of fit. Lower AIC values indicate a better trade-off.

For small sample sizes, a corrected version is often used:

$$\mathrm{AICc} = \mathrm{AIC} + \frac{2n_\theta(n_\theta + 1)}{n - n_\theta - 1}, \tag{3.25}$$

where n is the number of observations. This correction increases the penalty for complex models when data are limited, which is particularly relevant in industrial applications with sparse high-quality data.

Another commonly used criterion is the Bayesian Information Criterion (BIC), given by

$$\mathrm{BIC} = n_\theta \ln(n) - 2\ln(\hat{L}). \tag{3.26}$$

Compared to AIC, BIC imposes a stronger penalty on model complexity, especially as the dataset size increases. As a result, BIC tends to favour simpler models.

In practice, these criteria are evaluated for multiple candidate model orders, and the model with the lowest value is selected. This provides a systematic way to avoid both underfitting and overfitting while keeping the model structure interpretable.

In industrial applications, model selection is therefore not only a numerical optimisation problem. It is a trade-off between predictive accuracy, robustness, interpretability, and long-term maintainability. The specific model families that implement these ideas are reviewed in the following chapter.

3.5 Threshold-Based Anomaly Classification

Prediction alone does not complete the monitoring task. A regression model estimates the expected process behaviour, but monitoring also needs a decision rule that determines whether the measured signal remains consistent with this expectation. The deviation score s_i introduced in Eq. (3.8) provides this connection between prediction and detection. It measures the absolute difference between the observed signal and the selected reference trajectory $\hat{y}^{\mathrm{ref}}$ at sample i. Large values of s_i indicate that the measured signal differs from the expected trend.

In this thesis, we use threshold-based classification to convert the continuous score s_i into a binary monitoring decision. The decision threshold δ defines the maximum acceptable deviation from the reference trajectory $\hat{y}_i^{\text{ref}}$. This approach does not identify the physical cause of the deviation. It only detects that the measured signal y^{LF} no longer follows the expected behaviour closely enough.

3.5.1 Decision Threshold Selection

A simple statistical choice defines the threshold from the variability of the deviation signal. Let σ_e denote the standard deviation of the deviation e_i:

$$\sigma_e = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(e_i - \bar{e})^2}, \tag{3.27}$$

where $\bar{e}$ is the mean deviation. A common threshold then follows the $\kappa\sigma$ rule:

$$\delta = \kappa\sigma_e, \tag{3.28}$$

where κ determines the width of the tolerance band. Typical values are $\kappa = 1$, $\kappa = 2$, or $\kappa = 3$. Under Gaussian and stationary residuals, the 3σ rule gives a wide tolerance band and flags only large deviations. This statistical interpretation is useful, but industrial residuals rarely satisfy all required assumptions. In such cases, a variance-based threshold may not describe the real acceptable disagreement between the measured signal and the reference trajectory.

For this reason, a threshold can also be selected empirically from data. In MF monitoring, this can mean using the observed disagreement between LF online measurements and more reliable HF information. In this case, δ represents an acceptable discrepancy between information sources, not only a statistical confidence bound. This interpretation is more suitable when the deviation contains both process variability and measurement mismatch.

The threshold must be optimised without using the final test data. In this thesis, we optimise δ on the TR subset using The Nelder-Mead method, obtain the fixed threshold δ^* from Eq. (3.17), and then apply δ^* unchanged to the V or TS subset. This rule is important because threshold tuning directly affects detector performance. If the threshold were tuned on the test subset, the reported results would become optimistic and would not represent real deployment.

3.5.2 Binary Detection Rule

After selecting δ^*, the detector assigns a binary decision to each sample. Using the score from Eq. (3.8), the decision rule is

$$s_i > \delta^*. \tag{3.29}$$

Equivalently, since $s_i = |e_i|$, this can be written as

$$|e_i| > \delta^*. \tag{3.30}$$

If the condition is true, the detector flags the sample as anomalous, and normal otherwise. We denote the selected method detector output as

$$\hat{y}_i^{\text{det}} \in \{0, 1\}, \tag{3.31}$$

where $\hat{y}_i^{\text{det}} = 0$ denotes a normal detector decision and $\hat{y}_i^{\text{det}} = 1$ denotes an anomalous detector decision, while y_i^{pGT} represents the reference classification used for evaluation.

3.5.3 Detector Evaluation

We evaluate detector performance by comparing the detector output $\hat{y}_i^{\text{det}}$ with the reference label y_i^{pGT}. A TP occurs when the detector correctly flags an anomalous sample. A TN occurs when it correctly leaves a normal sample unflagged. A FP occurs when the detector raises an alarm for a normal sample, and a FN occurs when the detector misses an anomalous sample. These four outcomes form the confusion matrix used earlier in Eq. (3.9).

The selected threshold controls the balance between these outcomes. A lower threshold usually increases sensitivity because the detector flags more samples as anomalous. However, this can also increase false positives. A higher threshold usually increases specificity because the detector raises fewer alarms, but it can also increase false negatives. In industrial monitoring, this trade-off matters. Too many false alarms reduce operator trust, while missed anomalies may hide sensor drift, analyzer faults, or process disturbances.

This threshold-based formulation gives a direct link between prediction and monitoring. The prediction model defines the expected trajectory, the deviation score measures inconsistency with this trajectory, and the threshold converts the score into a binary decision. Later chapters use this structure to compare anomaly detectors under fixed training-testing separation and to evaluate whether the selected reference trajectory supports reliable anomaly detection.

Chapter 4

Approaches for Soft Sensing and Monitoring

The previous chapter introduced the main concepts required for data-driven prediction and monitoring, including preprocessing, feature engineering, validation, regression models, and anomaly-detection principles. This chapter builds on that foundation and reviews the main modelling approaches used for soft sensing and process monitoring.

The aim of this chapter is to explain the main modelling approaches that are relevant to this thesis. These include first-principles models, data-driven soft sensors, hybrid modelling approaches, and MF models. Each group provides useful ideas, but each also has limitations when applied to industrial datasets with dense LF measurements and sparse HF laboratory data. This review also clarifies the gap addressed in the later chapters.

4.1 First-Principles Modelling

First-principles modelling represents one of the oldest and most established approaches in process systems engineering. In this approach, the process is described using physical and chemical laws, such as conservation of mass, conservation of energy, phase equilibria, reaction kinetics, and transport phenomena [21, 34]. The model does not learn the process relation only from historical data. Instead, it starts from known process mechanisms and expresses them mathematically.

A first-principles model can be written in general form as

$$\mathcal{M}(\mathbf{x}_{\mathcal{M}}, \mathbf{u}_{\mathcal{M}}, \mathbf{p}_{\mathcal{M}}) = 0, \tag{4.1}$$

where $\mathbf{x}_{\mathcal{M}}$ denotes process states, $\mathbf{u}_{\mathcal{M}}$ denotes inputs or operating conditions, and $\mathbf{p}_{\mathcal{M}}$ denotes model parameters. Depending on the application, the model may be

steady-state or dynamic. A steady-state model describes the process after transients have disappeared, while a dynamic model describes how the process evolves in time.

In process monitoring and soft sensing, first-principles models can provide estimates of variables that are not measured continuously. For example, if a product composition is not available online, a process model may estimate it from flows, temperatures, pressures, and feed compositions. Such a model can also support optimisation, controller design, scenario analysis, and fault interpretation.

4.1.1 Mass and Energy Balance Models

Mass and energy balance models form the basic layer of most process models. A mass balance states that material cannot disappear or appear without reaction or accumulation. For a general process unit, the balance can be written as

$$\frac{\mathrm{d}M}{\mathrm{d}t} = \dot{M}_{\mathrm{in}} - \dot{M}_{\mathrm{out}} + \dot{M}_{\mathrm{gen}}, \tag{4.2}$$

where M is the amount of material in the unit, $\dot{M}_{\mathrm{in}}$ and $\dot{M}_{\mathrm{out}}$ are inlet and outlet material flows, and $\dot{M}_{\mathrm{gen}}$ represents generation or consumption due to reaction. Similarly, an energy balance relates internal energy or enthalpy to heat transfer, work, and energy carried by material flows. These equations are useful because they encode relationships that must hold independently of the available data. In industrial applications, this is important because plant data may contain noise, drift, or missing records, while physical conservation laws remain valid.

In a complete process model, mass and energy balances are combined with additional relations. These may include vapour-liquid equilibrium models, reaction-rate expressions, heat-transfer correlations, pressure-drop equations, and equipment-specific constraints [35, 36]. For example, a distillation column model may require tray balances, equilibrium relations, hydraulic constraints, and condenser and reboiler equations. A reactor model may require reaction kinetics, heat removal, residence time, and catalyst activity. However, full mechanistic detail is not always necessary. In many monitoring tasks, the goal is not to reproduce every internal state, but to capture the dominant input-output relation. For this reason, reduced first-principles models are often used. A reduced model may keep only the main unit operations, recycle paths, and balance relations that influence the monitored output. Such a model can still provide useful physical structure, even when it cannot reproduce the plant exactly. A simplified simulator should provide physically consistent trends.

In this thesis, reduced process models are implemented in industrial simulation envi-

ronments, mainly gPROMS ProcessBuilder [37] and AVEVA Process Simulation [38]. These tools allow the simplified flowsheet to be built from unit-operation blocks, material streams, recycle paths, and selected process specifications. The generated simulation data then serve as an additional LF information source for surrogate modelling and later HF residual correction. This information can later be corrected using plant and laboratory data.

4.1.2 Advantages and Limitations

First-principles models have several important advantages. First, they are physically interpretable. Each equation and parameter has a process meaning, which helps engineers understand why the model behaves in a certain way. Second, such models can extrapolate more safely than purely empirical models when the process moves outside the historical data range, provided that the physical assumptions remain valid. Third, they can simulate operating points that have not yet occurred in the plant. This makes them useful for design, optimisation, and sensitivity analysis.

They also provide a useful reference for monitoring. If a measured signal violates a mass balance or deviates strongly from a model-based estimate, this can indicate a sensor problem or an abnormal operating condition. This is why first-principles models remain important in advanced process control and process diagnostics.

However, their practical limitations are significant. A detailed industrial model requires many assumptions, parameters, and validation steps. Some parameters are difficult to measure directly. Reaction kinetics, catalyst activity, heat-transfer coefficients, phase-equilibrium parameters, and fouling effects may change over time. In large-scale units, unmeasured disturbances and recycle interactions further complicate the model.

Model maintenance is another problem. A model that works after commissioning may become less accurate after equipment changes, catalyst aging, sensor recalibration, or controller retuning. Updating the model then requires expert work. This limits the use of detailed first-principles models for routine soft sensing in plants where the monitored variable must be estimated continuously and reliably.

For these reasons, first-principles models are powerful but not always practical as standalone online soft sensors. They provide useful structure, but their direct deployment can be costly. This motivates methods that combine physical structure with data-driven correction.

4.2 Data-Driven Soft Sensors

Data-driven soft sensors estimate variables that are difficult, delayed, or unreliable to measure directly [10, 11]. In process industries, these variables are often product compositions, quality indicators, impurity levels, or internal stream properties. A practical soft sensor must provide accurate predictions, but it must also remain understandable and maintainable. This is important in industrial LF–HF settings, where online analyzers provide frequent but imperfect measurements, while laboratory analyses provide sparse but more reliable references. A model trained only on LF data may learn analyzer drift, whereas a model trained only on HF data may suffer from too few samples. The choice of model therefore depends not only on flexibility, but also on data quality, interpretability, and validation [19].

4.2.1 Linear Soft Sensors

Ordinary Least-Squares (OLS) Models are the simplest soft-sensor baseline. For a regressor vector $\mathbf{x}_i$, the prediction can be written as

$$\hat{y}_i = \theta_0 + \boldsymbol{\theta}^\top \mathbf{x}_i. \tag{4.3}$$

In practical soft-sensor applications, $\mathbf{x}_i$ may denote either the original selected process variables or an engineered input vector containing transformed, latent, or lagged variables. The parameters are estimated by minimizing the squared error:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \left(y_i - \theta_0 - \boldsymbol{\theta}^\top \mathbf{x}_i \right)^2. \tag{4.4}$$

OLS remains useful because it is transparent, fast to train, and easy to validate. If the input variables are selected well, a linear model can capture the dominant process trend. This is often sufficient when the plant operates in a limited operating region. The main weakness is sensitivity to collinearity and irrelevant variables, which are common in industrial datasets.

When process memory is relevant, the same linear modelling idea can be extended to dynamic input–output structures. In this thesis, dynamic linear LF predictors are identified using the SIPPY toolbox [39]. SIPPY is used as a practical system-identification tool for estimating candidate dynamic models from selected regressors and for comparing model orders. This is useful when the monitored output depends not only on the current operating point, but also on previous inputs or previous output values caused by recycle streams, residence time, transport delay, or analyzer delay.

The identified dynamic predictor can then serve as the LF baseline in the later MF residual-correction workflow.

4.2.2 Latent-Variable Soft Sensors

Dimensionality-reduction methods such as PCA and PLS were introduced in the previous chapter [12]. Here, we do not redefine their full mathematical background. Instead, we explain how they are used to construct soft sensors. The main idea is that the latent variables are not the final output of the model. They serve as transformed regressors for predicting the monitored variable.

Principal Component Regression (PCR) combines PCA with ordinary regression. First, PCA transforms the original input matrix into a smaller set of retained score variables as defined in Eq. (3.4). The soft sensor is then built by fitting a regression model between the retained scores and the output variable:

$$\hat{y}_i = \theta_0 + \boldsymbol{\theta}^\top \mathbf{t}_i, \tag{4.5}$$

where $\mathbf{t}_i$ is the i-th row of $\mathbf{T}_k$. PCR therefore uses PCA only as a preprocessing step. The regression model itself remains linear, but it operates on latent variables instead of the original measured variables. This can improve numerical stability when the original inputs are strongly correlated. The main limitation follows directly from PCA: the principal components are selected according to input variance, not according to their relation to the output. As a result, PCR may retain directions that explain large input variability but are only weakly useful for soft sensing.

PLS-Based Soft Sensors also construct latent variables, but they do so with the prediction objective in mind. As introduced earlier, PLS decomposes the input and output data, as defined in Eq. (3.5). The resulting soft sensor can be written as

$$\hat{y}_i = \mathbf{t}_i^\top \mathbf{c}, \tag{4.6}$$

where $\mathbf{t}_i$ is the latent score vector for sample i. Unlike PCR, PLS constructs the latent variables so that they are relevant for predicting the output [13]. This makes PLS suitable for soft sensing when the input variables are strongly collinear and when a compact predictive representation is needed. The trade-off remains interpretability, because each latent variable combines information from several physical measurements. Therefore, a PLS-based soft sensor can be accurate and stable, but less transparent than a model based directly on a small set of selected process variables.

4.2.3 Sparse Regression Soft Sensors

Methods such as LASSO and stepwise regression were also introduced earlier. Here, the focus is on how they define soft sensors. The key difference from latent-variable methods is that the model remains in the original variable space.

LASSO from the already defined formulation (Eq. (3.3)) has the following form for the resulting soft sensor

$$\hat{y}_i = \theta_0 + \sum_{j \in \mathcal{S}} \theta_j x_{i,j}, \tag{4.7}$$

where $\mathcal{S}$ is the subset of variables with nonzero coefficients. The main advantage is that the final model uses physical variables directly, which supports interpretation and deployment.

Stepwise regression produces the same model structure,

$$\hat{y}_i = \theta_0 + \sum_{j \in \mathcal{S}} \theta_j x_{i,j}, \tag{4.8}$$

but the subset $\mathcal{S}$ is obtained through a sequential search procedure.

In both LASSO and stepwise regression, the important point is not the selection algorithm itself, but the resulting compact model defined on physically meaningful variables. In the presence of strong collinearity, several variable combinations can provide similar prediction accuracy. As a result, the selected subset may not be unique and may change across datasets.

4.2.4 Nonlinear Soft Sensors

Gaussian Process Regression (GPR) is a nonlinear probabilistic regression method. It assumes that the unknown input-output relation follows a Gaussian process (GP):

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right), \tag{4.9}$$

where $m(\cdot)$ is the mean function and $k(\cdot, \cdot)$ is the covariance kernel [40]. The kernel defines the assumed similarity between two input points and therefore controls the smoothness, variability, periodicity, and noise sensitivity of the resulting model. For this reason, the kernel choice represents an important modelling decision and should not be selected manually without validation.

A commonly used kernel is the radial basis function (RBF) kernel,

$$k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\ell_{\mathrm{RBF}}^2}\right), \tag{4.10}$$

where σ_f^2 is the signal variance and ℓ_{RBF} is the length scale. The signal variance controls the typical variation of the function around its mean, while the length scale controls how quickly the function can change with respect to the inputs.

Besides the basic RBF kernel, this thesis considers several composite kernels. A scaled RBF kernel multiplies the RBF kernel by a constant factor,

$$k_{\mathrm{RBF},2}(\mathbf{x}_i, \mathbf{x}_j) = C\, k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j), \tag{4.11}$$

where C adjusts the overall output variability. Measurement uncertainty can be represented by adding a white-noise term,

$$k_3(\mathbf{x}_i, \mathbf{x}_j) = k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j) + \sigma_n^2 \delta_{\mathrm{K},ij}, \tag{4.12}$$

where σ_n^2 is the noise variance and $\delta_{\mathrm{K},ij}$ is the Kronecker delta. Periodic behaviour can be included using a periodic kernel component,

$$k_{\mathrm{PER}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 \exp\left(-\frac{2\sin^2\left(\pi\|\mathbf{x}_i - \mathbf{x}_j\|/d\right)}{\ell_{\mathrm{PER}}^2}\right), \tag{4.13}$$

where d defines the period and ℓ_{PER} controls the smoothness of the periodic component. This component can be combined with the RBF kernel either multiplicatively,

$$k_4(\mathbf{x}_i, \mathbf{x}_j) = C\, k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j) k_{\mathrm{PER}}(\mathbf{x}_i, \mathbf{x}_j) + \sigma_n^2 \delta_{ij}, \tag{4.14}$$

or additively,

$$k_5(\mathbf{x}_i, \mathbf{x}_j) = C\, k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j) + k_{\mathrm{PER}}(\mathbf{x}_i, \mathbf{x}_j) + \sigma_n^2 \delta_{ij}. \tag{4.15}$$

Additive kernel combinations allow the model to explain several effects independently, while multiplicative combinations require the effects to occur jointly.

In the proposed workflow, the kernel is selected by CV rather than fixed in advance. Candidate kernels are trained on the corresponding training subset and evaluated on the validation subset. The final kernel is selected according to the validation RMSE distribution, typically using the median RMSE and its variability across repeated training runs. This step reduces manual tuning and limits the risk that the GPR correction fits sparse HF samples too closely. In this thesis, GPR is especially relevant for residual correction, where the model learns the discrepancy between an LF prediction and sparse HF measurements. Its main advantage is that it can model smooth corrections from a small number of reliable samples and provide uncertainty estimates. Its main limitation is sensitivity to the selected kernel and possible overfitting when the number of HF samples is very small.

Neural-Network Soft Sensors. A feed-forward neural network represents a nonlinear mapping through several layers. For sample i, the input vector is denoted by $\mathbf{x}_i$. The vector $\mathbf{h}_i^{(L)}$ denotes the hidden representation, or activation vector, of sample i after layer L. The input layer is defined as

$$\mathbf{h}_i^{(0)} = \mathbf{x}_i. \tag{4.16}$$

For layer L, the hidden representation is computed as

$$\mathbf{h}_i^{(L)} = \sigma^{(L)}\left(\mathbf{W}^{(L)}\mathbf{h}_i^{(L-1)} + \mathbf{b}^{(L)}\right), \tag{4.17}$$

where $\mathbf{W}^{(L)}$ is the weight matrix of layer L, $\mathbf{b}^{(L)}$ is the bias vector of layer L, and $\sigma^{(L)}(\cdot)$ is the activation function used in layer L. The final prediction is obtained from the last hidden representation:

$$\hat{y}_i = \mathbf{w}^\top \mathbf{h}_i^{(L)} + b, \tag{4.18}$$

where $\mathbf{w}$ and b are the output-layer weight vector and bias.

Neural networks can represent nonlinear interactions between variables and are useful when large training datasets are available. In this thesis context, they can also serve as simulation surrogates, because simulation data can usually be generated in larger quantities than laboratory data. However, in sparse HF settings, neural networks can overfit easily. A neural network trained only on simulation data may also learn simulation-specific nonlinearities that do not transfer directly to the real plant.

Autoencoder-Based Soft Sensors learn a compressed latent representation of the input. For sample i, the encoder maps the input vector $\mathbf{x}_i$ into a latent vector:

$$\mathbf{z}_i = f_{\text{enc}}(\mathbf{x}_i), \tag{4.19}$$

and the decoder reconstructs the input:

$$\hat{\mathbf{x}}_i = f_{\text{dec}}(\mathbf{z}_i). \tag{4.20}$$

The autoencoder is trained by minimizing the reconstruction error:

$$\min \sum_{i=1}^{n} \left\|\mathbf{x}_i - \hat{\mathbf{x}}_i\right\|^2. \tag{4.21}$$

For soft sensing, the latent vector can be used as the regressor:

$$\hat{y}_i = \theta_0 + \boldsymbol{\theta}^\top \mathbf{z}_i. \tag{4.22}$$

Autoencoders act as nonlinear dimensionality-reduction tools. They may capture structure that PCA cannot represent. Their weakness is lower interpretability and higher data demand. They should therefore be used as nonlinear baselines rather than as the default industrial solution.

4.3 Hybrid Modelling Approaches

Hybrid modelling combines physical structure with data-driven correction. In the context of this thesis, the main objective is not to construct a fully mechanistic model, but to use available physical knowledge to guide data-driven prediction under limited and heterogeneous data. Simulation models may provide physically consistent trends but do not match the plant exactly. Online measurements are frequent but may be biased or drifting. Laboratory measurements are reliable but sparse. Hybrid modelling provides a way to combine these sources while preserving interpretability.

A hybrid model can be built in several ways. A data-driven model may correct the output of a physical model. A physical constraint may be included in the loss function of a neural network. A simulator may generate additional training data for a surrogate. Alternatively, a grey-box model may combine known equations with fitted empirical terms. Hybrid approaches are attractive in industrial monitoring because they reduce the weaknesses of purely mechanistic and purely data-driven models. They can exploit physical consistency without requiring a complete plant model, and they can use data to adapt to the real process.

4.3.1 Physics-Informed Models

Physics-informed models include physical laws directly in the modelling process. Instead of learning only from input-output pairs, the model is also constrained by equations such as balances, reaction relations, or known monotonicity [41]. This can be done by adding physical residuals to the loss function:

$$\mathcal{L} = \mathcal{L}_{\text{data}} + \lambda_{\text{phys}} \mathcal{L}_{\text{phys}}, \tag{4.23}$$

where $\mathcal{L}_{\text{data}}$ measures agreement with data and $\mathcal{L}_{\text{phys}}$ measures violation of physical constraints. The main advantage is improved consistency. A model that respects mass balance or energy balance is less likely to produce physically impossible predictions. This is important when data are sparse or noisy. Physical constraints can also improve extrapolation because they restrict the set of possible functions [42]. However, physics-informed models require reliable physical equations and suitable parameterization. In

industrial plants, not all relevant mechanisms are known or measurable. Simplified constraints may help, but incorrect constraints can bias the model. In addition, implementing physics-informed training may require more development effort than conventional regression.

In this thesis, the relevance of physics-informed modelling is not in the use of complex neural architectures, but in the principle that physical structure should constrain the model. This is implemented in a simpler way by selecting physically meaningful variables, by using simulation-based LF predictions, and by constructing residual corrections around a physically consistent baseline.

4.3.2 Grey-Box Models

Grey-box models combine known mechanistic equations with empirical components. The mechanistic part represents the known process structure, while the empirical part estimates uncertain relations or unknown parameters from data [43]. A general grey-box model can be written as

$$\hat{y}_i = f_{\text{phys}}(\mathbf{x}_i, \mathbf{p}) + f_{\text{emp}}(\mathbf{x}_i, \boldsymbol{\theta}), \tag{4.24}$$

where f_{phys} represents the physical model, f_{emp} represents the empirical correction, $\mathbf{p}$ denotes physical or mechanistic parameters, and $\boldsymbol{\theta}$ denotes empirical model parameters.

Grey-box modelling is useful when a full first-principles model is too complex but partial physical knowledge is available. For example, mass-balance relations may be known, while reaction rates or separation efficiencies may be uncertain. The empirical component can then compensate for the missing details. In soft sensing, grey-box models can estimate quality variables by combining process balances with data-driven calibration. This improves interpretability compared with a purely black-box model. Engineers can inspect the mechanistic part and understand how the correction behaves.

The limitation is that grey-box models still require a meaningful mechanistic structure. If the physical part is too inaccurate, the empirical correction may need to compensate for large systematic errors. This can reduce the benefit of the hybrid structure. Moreover, grey-box models often require repeated calibration when the plant changes. In this sense, residual-correction structures used later in this thesis can be interpreted as a simplified grey-box approach, where the baseline model represents the physical component and the correction model represents the empirical adjustment.

4.4 Multi-Fidelity Modelling

MF modelling deals with data sources that describe the same system with different levels of accuracy, cost, sampling frequency, or physical detail. In industrial process monitoring, this situation appears naturally because online measurements, laboratory analyses, and simulation data do not provide the same type of information. These sources should not be treated as equivalent, but they should also not be used in isolation. A MF model combines them so that each source contributes its strongest information: dense LF plant data provide temporal coverage, sparse HF laboratory data provide reference accuracy, and simulation data can provide physically consistent trends and additional operating scenarios. In this sense, MF modelling can be interpreted as a special case of hybrid modelling, where different information sources—data-driven or model-based—are systematically integrated without changing the underlying modelling structure.

In the notation used in this thesis, the LF plant dataset is denoted by $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$. The HF dataset is denoted by $\mathbf{y}^{\mathrm{HF}}$. An additional LF simulation data source is denoted by $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$. The modelling task is therefore not only to fit a predictor, but to combine these sources into a corrected estimate of the monitored output. This structure differs from ordinary supervised regression.

4.4.1 Co-Kriging and Gaussian-Process-Based Fusion

Co-kriging is one of the classical approaches for MF modelling. It assumes that low- and HF functions are correlated. A common formulation expresses the HF function as a scaled LF function plus a discrepancy term:

$$f_{\mathrm{HF}}(\mathbf{x}) = \rho f_{\mathrm{LF}}(\mathbf{x}) + \delta(\mathbf{x}), \tag{4.25}$$

where ρ is a scaling coefficient and $\delta(\mathbf{x})$ is a correction process. GPs are often used to model both the LF function and the discrepancy. This gives a probabilistic model that can interpolate sparse HF observations while using LF information to support the prediction. The approach is attractive because it provides both a mean prediction and uncertainty estimates.

In process applications, co-kriging can be useful when LF and HF data are strongly correlated and share a comparable input space. For example, a simulator may provide cheap approximate outputs, while laboratory or plant measurements provide accurate but sparse values. The co-kriging model can then combine both levels. However, classical co-kriging also has practical limitations. It can become computationally demanding when the LF dataset is large. It also requires careful covariance modelling

and can overfit when the number of HF samples is small. Therefore, co-kriging is an important reference method, but it is not always the most practical solution for deployment-oriented soft sensing. In many industrial cases, simpler residual-correction structures can provide similar or better robustness with less complexity.

4.4.2 Residual-Correction Multi-Fidelity Models

Residual-correction MF models form a practical class of methods for combining dense LF information with sparse HF reference data. Instead of learning the full HF output directly from the sparse laboratory samples, the model first constructs a LF baseline prediction and then learns the remaining discrepancy to the HF reference. This follows the same principle as bias correction in industrial practice, where online analyzer values are adjusted using laboratory results. The difference is that the correction is not limited to a manual offset. A data-driven correction model can learn how the discrepancy changes with process conditions.

Let $\hat{y}_i^{\mathrm{LF}}$ denote a LF baseline prediction available on the plant time grid. This baseline may come from the online analyzer, from a data-driven LF predictor, or from a simulation-based surrogate. At the HF timestamps, the residual between the trusted laboratory value and the matched LF baseline is defined as

$$\Delta_j = y_j^{\mathrm{HF}} - \hat{y}_{i(j)}^{\mathrm{LF}}, \tag{4.26}$$

where $i(j)$ denotes the LF index matched to the j-th HF sample. The residual Δ_j represents the part of the HF information that the LF baseline does not capture. It may include analyzer bias, drift, calibration mismatch, plant-model mismatch, or effects of process variables that are not fully represented in the baseline model.

The role of the residual correction can be seen directly from this definition. If the true residual were available at a matched HF timestamp, adding it to the LF baseline would give

$$\hat{y}_{i(j)}^{\mathrm{MF}} = \hat{y}_{i(j)}^{\mathrm{LF}} + \Delta_j = \hat{y}_{i(j)}^{\mathrm{LF}} + \left(y_j^{\mathrm{HF}} - \hat{y}_{i(j)}^{\mathrm{LF}}\right) = y_j^{\mathrm{HF}}. \tag{4.27}$$

This shows the main idea of the proposed MF approach. The correction is not only a numerical adjustment of the LF prediction. Its purpose is to move the LF baseline toward the HF laboratory reference. At the sparse HF timestamps, the ideal correction would exactly recover the laboratory value. Since y_j^{HF} is available only at these sparse timestamps, the true residual cannot be used directly over the full plant time grid.

For this reason, a correction model is trained on the available residuals. In the predictor-correction structure, the LF prediction is not used only as the signal to be

corrected. It is also included as an additional regressor in the MF correction model. The MF residual-correction regressor is written as

$$\boldsymbol{\phi}_i^{\mathrm{MF}} = \left[\hat{y}_i^{\mathrm{LF}}, \boldsymbol{\phi}_i^{\top}\right]^{\top}, \tag{4.28}$$

where $\boldsymbol{\phi}_i$ is the fueature-selected LF regressor vector available on the plant time grid. If the LF baseline is dynamic, then $\hat{y}_i^{\mathrm{LF}}$ is the dynamic LF prediction. In this way, the correction model receives both the selected process variables and the LF model output that approximates the process trend.

At the HF timestamps, the correction model is trained on the paired residual data

$$\left(\boldsymbol{\phi}_{i(j)}^{\mathrm{MF}}, \Delta_j\right), \qquad j \in \mathcal{I}_{\mathrm{HF}}. \tag{4.29}$$

The predicted correction on the full LF time grid is then

$$\hat{\Delta}_i = g\left(\boldsymbol{\phi}_i^{\mathrm{MF}}\right), \qquad i \in \mathcal{I}_{\mathrm{LF}}. \tag{4.30}$$

The final MF prediction is obtained by adding the predicted correction to the LF baseline:

$$\hat{y}_i^{\mathrm{MF}} = \hat{y}_i^{\mathrm{LF}} + \hat{\Delta}_i. \tag{4.31}$$

In this way, the model learns the LF–HF discrepancy at the sparse HF timestamps and applies the learned correction across the dense LF time grid. The resulting MF prediction can therefore be interpreted as an estimate of an HF-consistent trajectory over the full plant timeline.

This structure reduces the learning problem. The LF baseline captures the dominant process trend, while the correction model learns only the systematic discrepancy to the HF reference. If the baseline already follows the main process behaviour, the residual is usually smoother and easier to model than the full output. This is important in industrial MF soft sensing, because the number of HF laboratory samples is often too small for training a complete model from HF data alone.

Residual correction also improves interpretability. The baseline and the correction term can be inspected separately. If the correction remains small and smooth, the LF baseline is consistent with the HF reference. If the correction becomes large, unstable, or strongly dependent on a narrow operating region, this may indicate analyzer drift, calibration problems, plant-model mismatch, or operation outside the training region. This interpretation is useful for monitoring because the correction term itself provides information about the disagreement between fidelity levels.

Two practical variants are used in this thesis. The first variant corrects the online analyzer directly and is referred to as analyzer correction. In this case, the LF baseline is the online analyzer signal:

$$\hat{y}^{\mathrm{MF}}_{\Delta_{\mathrm{AC}},i} = y^{\mathrm{LF}}_i + \hat{\Delta}_i. \tag{4.32}$$

The corresponding correction is learned from the difference between the HF laboratory value and the matched LF analyzer value. This variant is simple and close to classical analyzer bias correction. However, it may inherit short-term analyzer noise, spikes, or local faults, because the corrected trajectory still starts from the online analyzer signal.

The second variant corrects a LF predictor and is referred to as predictor correction. In this case, the LF baseline is not the online analyzer value, but a prediction obtained from selected process variables:

$$\hat{y}^{\mathrm{MF}}_{\Delta_{\mathrm{PC}},i} = \hat{y}^{\mathrm{LF}}_i + \hat{\Delta}_i. \tag{4.33}$$

The LF predictor first captures the main process trend and can reduce the influence of short-term analyzer fluctuations, so the GPR correction does not need to reconstruct the LF trend from the available regressors alone. Instead, it learns how this LF trend must be shifted toward the HF laboratory reference. This variant is often more robust when the online analyzer contains drift, bias, short spikes, or local measurement faults.

Residual-correction MF models are well suited for the data structure considered in this thesis. They use dense LF data without treating them as fully reliable, and they use sparse HF data without forcing the complete output model to be trained only on a few laboratory samples. When simulation data are available, the same structure can also correct a simulation-based LF prediction toward the plant laboratory reference. The residual-correction form therefore provides a common modelling principle for data-based, analyzer-based, predictor-based, and simulation-supported MF soft sensors.

4.5 Summary of Existing Approaches and Motivation for the Proposed Framework

The reviewed methods show that a wide range of tools exists for soft sensing and process monitoring. However, existing approaches do not fully align with the combination of simplicity, interpretability, and deployability required in the considered industrial setting.

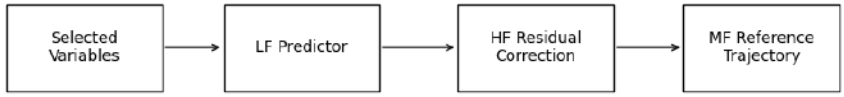


Figure 4.1: Proposed multi-fidelity modelling workflow.

First-principles models provide physical structure, but detailed models are expensive to build, calibrate, and maintain. Data-driven soft sensors are easy to train, but they depend strongly on data quality and may learn analyzer faults if trained on LF outputs. Nonlinear and deep-learning models increase flexibility, but this flexibility can become a weakness when HF data are sparse [14, 16]. Hybrid models improve consistency, but they still require careful design. Classical MF methods, such as co-kriging, provide a formal fusion framework but can become too complex or unstable for small HF datasets. The main limitation is not the absence of modelling tools, but the lack of a structured and deployable workflow that consistently exploits heterogeneous industrial data under reliability constraints.

This motivates the structured multi-fidelity framework proposed in the following chapters, which defines how heterogeneous data sources are consistently integrated into a unified soft-sensing workflow under industrial data constraints.

Chapter 5

Case Studies

This chapter introduces the two process systems used to evaluate the proposed methodology. The objective is not only to describe the systems themselves, but to establish a consistent representation of their data structure, modelling objectives, and associated challenges. Both case studies are formulated within the same multi-fidelity framework defined in the previous chapters.

Industrial process datasets rarely consist of measurements of uniform quality. Online sensors provide dense time series that reflect the instantaneous state of the process, but these signals are affected by noise, drift, calibration errors, and operational disturbances. In contrast, laboratory measurements provide accurate information about key quality variables, but they are available only at sparse time instants and are not synchronized with the plant timeline.

This mismatch between measurement frequency and reliability creates a fundamental limitation for standard modelling approaches. A predictor trained only on dense measurements may inherit their bias, while a predictor trained only on sparse reference values cannot capture process dynamics. The case studies considered here reflect this structure in two different settings. The Tennessee Eastman Process (TEP) provides a controlled benchmark where the data generation mechanism is known and can be manipulated. The alkylation unit represents a real industrial system, where process knowledge is incomplete and measurements are subject to uncertainty.

5.1 Tennessee Eastman Process

The Tennessee Eastman Process (TEP) is a well-established benchmark in process systems engineering, representing a complex chemical plant with interacting reaction, separation, and recycle subsystems. The process, shown in Figure 5.1, includes a

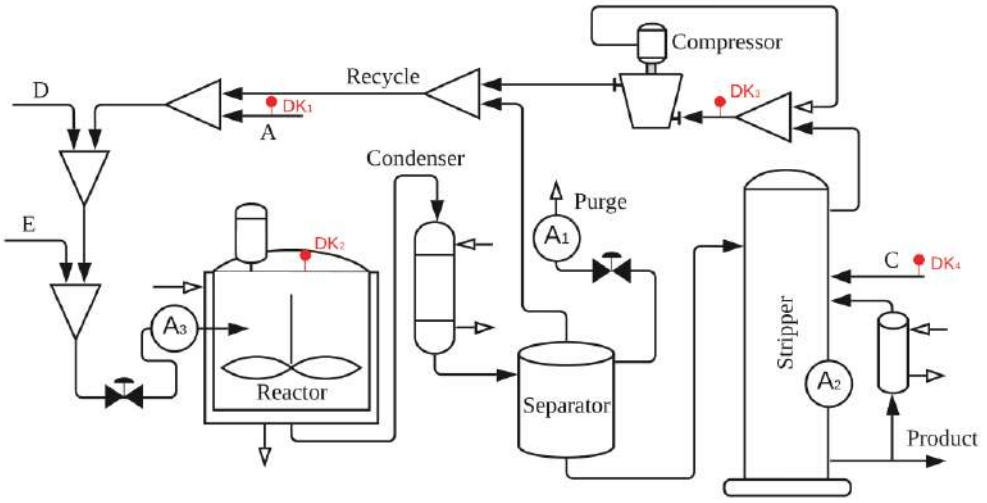


Figure 5.1: Simplified representation of the TEP benchmark.

reactor, condenser, separator, stripper, and recycle loop, forming a tightly coupled multivariable system. Multiple gaseous reactants undergo irreversible and exothermic reactions to produce liquid products, while recycle streams introduce strong feedback and dynamic interactions.

From a process perspective, TEP captures key characteristics of industrial systems, including nonlinear behaviour, multivariable coupling, and sensitivity to disturbances. At the same time, it differs from real industrial plants in one important aspect: the data are generated from a known model and are therefore free from sensor faults, calibration drift, and measurement inconsistencies. This makes TEP suitable as a controlled environment for evaluating modelling approaches under idealized conditions.

5.1.1 Data Structure and Variable Definition

The available dataset consists of $n_{\mathrm{LF}} = 1000$ samples of process measurements. Each sample contains both manipulated variables and measured process variables, resulting in a total of 52 variables, including 41 process measurements and 11 control inputs. These measurements form the LF dataset. In this study, the monitored output $\mathbf{y}^{\mathrm{LF}}$ is chosen as the concentration of component C in the purge stream. This variable reflects the combined effect of reaction and separation performance and is therefore suitable for evaluating soft-sensing models.

5.1.2 High-Fidelity Reference Construction

Since the original TEP dataset does not include laboratory measurements, a pseudo HF signal is constructed to emulate sparse and reliable reference data. The HF signal is obtained by applying a moving-average filter to the LF signal:

$$y^{\mathrm{HF}}(i) = \frac{1}{20} \sum_{k=0}^{19} y^{\mathrm{LF}}(i - k + B), \tag{5.1}$$

where B is a fixed offset introduced to prevent direct alignment between LF and HF samples. This transformation reduces high-frequency variability and produces a smoother signal representing the underlying process trend. The filtered signal is subsequently downsampled to obtain $n_{\mathrm{HF}} = 28$ samples, mimicking the sparse availability of laboratory measurements.

5.2 Alkylation Process

Alkylation is a refinery process used to produce high-octane gasoline blending components through the reaction of light olefins (C_3–C_4) with isobutane (i-C_4). The resulting product, referred to as alkylate, consists primarily of highly branched C_7–C_9 isoparaffins with favourable combustion properties.

The process is based on acid-catalyzed reaction mechanisms. The reaction sequence is initiated by protonation of the olefin in the presence of a strong acid catalyst:

$$\mathrm{C{-}C{=}C{-}C} \xrightarrow{\mathrm{H_2SO_4}} \mathrm{C{-}C{-}C^{+}{-}C} \tag{5.2}$$

This intermediate reacts with isobutane to form more stable carbonium ions:

$$\text{i-}\mathrm{C_4}^{+} + \text{olefin} \xrightarrow{\mathrm{H_2SO_4}} \mathrm{TMPs}^{+} \tag{5.3}$$

The reaction chain is sustained through hydride transfer reactions:

$$\mathrm{TMP}^{+} + \text{i-}\mathrm{C_4} \xrightarrow{\mathrm{H_2SO_4}} \mathrm{TMP} + \text{i-}\mathrm{C_4}^{+} \tag{5.4}$$

leading to the formation of trimethylpentanes, which dominate the alkylate composition.

A simplified representation of the alkylation unit is shown in Figure 5.2. The process consists of feed preparation, reaction, and separation stages. Olefins are mixed with fresh and recycled isobutane and fed into parallel reactors. Due to the exothermic nature of the reactions, the system is operated at low temperatures (5–10°C), maintained by a refrigeration cycle. The reaction mixture is then separated into acid and hydrocarbon

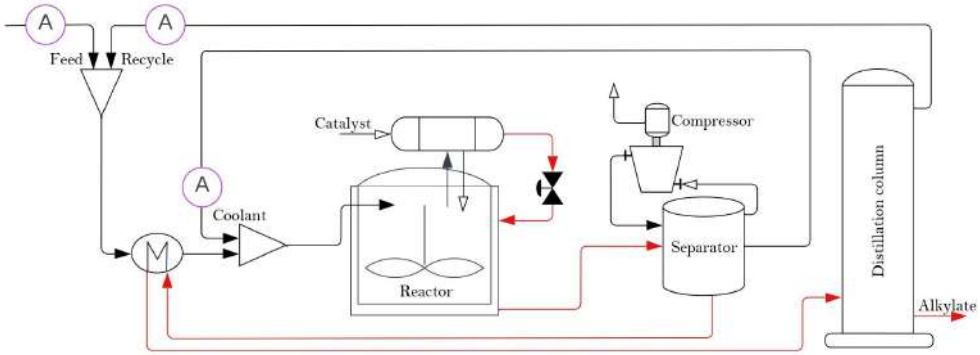


Figure 5.2: Simplified alkylation process flowsheet.

phases. The acid is recovered and recycled, while the hydrocarbon stream is processed in a fractionation system, where unreacted isobutane is separated and recycled. Recycle streams introduce strong coupling between process units. In particular, the amount of recycled isobutane affects both reaction conditions and downstream separation load.

A key operational parameter is the isobutane-to-olefin ratio. Maintaining this ratio within an optimal range is essential for suppressing undesired side reactions and ensuring product quality. In practice, this ratio is monitored using online analyzers. However, these analyzers often exhibit unreliable behaviour, including noise, drift, and abrupt deviations. As a result, their measurements cannot be consistently used for automatic control. Laboratory measurements provide accurate reference values but are available only at sparse intervals. Consequently, process operation relies on manual validation and frequent recalibration of analyzers. This limitation reduces the effectiveness of advanced control strategies and leads to suboptimal operation, increased recycle flows, and higher separation load.

5.2.1 Data Structure and Variable Definition

The available dataset consists of $n_{\mathrm{LF}} = 22,500$ samples of process measurements, representing a subset of a larger collection gathered over six months. Each sample contains measurements from online sensors, analyzers, and laboratory samples, resulting in a total of more than 1,085 variables. These measurements form the LF dataset. In this study, the monitored output $\mathbf{y}^{\mathrm{LF}}$ is chosen as the isobutane (i-C4) concentration in the coolant recycle stream.

Chapter 6

Data-Based Multi-Fidelity Soft Sensing for Online Sensor Correction

This chapter applies the MF workflow defined in previous chapters to online sensor correction. The aim is to improve the estimate of the monitored output by combining the dense LF analyzer signal $\mathbf{y}^{\mathrm{LF}}$ with sparse HF laboratory values $\mathbf{y}^{\mathrm{HF}}$. The general data structure, pairing rule $i(j)$, preprocessing, FS with DK, model validation, and residual-correction logic were already defined in previous sections, therefore, this chapter focuses on the implementation and results. This study is based on the methodology presented in our journal publication [31].

6.1 Implementation Workflow

Procedure 1 summarizes the complete implementation workflow. The workflow starts from the cleaned LF plant data $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$, together with the sparse HF reference $\mathbf{y}^{\mathrm{HF}}$ available on $\mathcal{T}^{\mathrm{HF}} \subset \mathcal{T}^{\mathrm{LF}}$. Feature selection with DK is carried out only on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$. The selected regressor vector $\boldsymbol{\phi}_i^{\mathrm{DK}}$ is then used to train LF and HF single-fidelity models. The LF model used in the MF correction is selected by RMSE evaluated at the paired HF indices $i(j)$, because the final objective is not to reproduce y_i^{LF}, but to approximate y_j^{HF}.

The nested data split follows the validation logic from Subsection 3.3. The outer split separates the LF timeline into $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ and $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$. The subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ is not used during FS, LF model selection, GP kernel selection, or residual-model training. Inside $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$, the split $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}') \cup \mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$ is used for MF model selection, while $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}') = \mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'') \cup \mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ is used for FS and single-fidelity model validation. Expanding-window CV is used for LF models on $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$, while k-fold CV is used for HF models because $\mathbf{y}^{\mathrm{HF}}$ is available only at sparse indices.

Static linear models, PCA, PLS, and GPR are implemented in `scikit-learn` [44].

Procedure 1 General workflow for model training and evaluation

Input: input matrix $\mathbf{X}$;
LF output $\mathbf{y}^{\mathrm{LF}}$ on $\mathcal{T}^{\mathrm{LF}}$ with index set $\mathcal{I}^{\mathrm{LF}}$;
HF output $\mathbf{y}^{\mathrm{HF}}$ on $\mathcal{T}^{\mathrm{HF}} \subset \mathcal{T}^{\mathrm{LF}}$ with index set $\mathcal{I}^{\mathrm{HF}}$.

1. **Preprocess data.**
   - Remove invalid samples and evident outliers. Standardize the dataset.
   - Apply the hierarchical data split.
2. **Select regressors.**
   - Apply the FS methods to columns of $\mathbf{X}$ using $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'')$.
   - Train linear single-fidelity predictors for the candidate subsets.
   - Select the subset with the lowest prediction error on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$.
3. **Train single-fidelity models.**
   - Train candidate single-fidelity predictors $\hat{y}$ with fidelity index $\{\mathrm{LF}, \mathrm{HF}\}$ on $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}')$.
   - Select the best-performing single-fidelity predictor using $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$.
   - Select the GP kernel structure (standard or composite) for the MF model using $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}')$ and $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$.
4. **Train multi-fidelity models.**
   - Use the selected single-fidelity predictor as the additional input.
   - Train the MF correction structures (AC and PC) on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$.
5. **Evaluate candidate models.**
   - Evaluate all trained models on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$.
   - Compute the prediction error using reference values at $i \in \mathcal{I}_{\mathrm{HF}}$.
   - Select the best-performing predictor $\hat{y}^{\star}$.

Output: predictor $\hat{y}^{\star}$ with the best performance against HF data.

Procedure 2 Proposed MF workflow: predictor correction (PC)

Input: preprocessed DK subset of columns of $\mathbf{X}$;
LF output $\mathbf{y}^{\mathrm{LF}}$ on $\mathcal{T}^{\mathrm{LF}}$with index set $\mathcal{I}^{\mathrm{LF}}$;
HF output $\mathbf{y}^{\mathrm{HF}}$ on $\mathcal{T}^{\mathrm{HF}} \subset \mathcal{T}^{\mathrm{LF}}$with index set $\mathcal{I}^{\mathrm{HF}}$.

1. **Train single-fidelity predictor.**
   - Identify the dynamic single-fidelity predictor $\hat{y}^{\mathrm{LF}}_{\mathrm{dyn}}$ on the preprocessed DK inputs.
   - Select the model order using BIC.
2. **Train MF correction model (PC).**
   - Add the dynamic single-fidelity prediction $\hat{y}^{\mathrm{LF}}_{\mathrm{dyn}}$ as an additional feature to the DK input subset.
   - Use the *Scaled RBF* kernel for the MF GP model.
   - Train the GPR model to predict the offset $\Delta_i = y^{\mathrm{HF}}_{,i} - \hat{y}^{\mathrm{LF}}_{\mathrm{dyn},i}$.
   - Correct the dynamic single-fidelity prediction using the GP-predicted offset $\hat{\Delta}$ to obtain the PC MF prediction.

Output: trained predictor-correction MF model $\hat{y}^{\mathrm{MF}}_{\Delta\mathrm{PC}}$.

Neural networks and autoencoders are implemented in Keras [45]. Dynamic LF models are identified using the `SIPPY` toolbox [39] with the PARSIM-K method. The dynamic model order is selected using information criteria. When AIC, AICc, and BIC suggest different orders, BIC is preferred because it keeps the model simpler and reduces the risk of fitting short-term analyzer noise.

Procedure 2 shows the predictor-correction implementation of $\hat{y}^{\mathrm{MF}}_{\Delta_{\mathrm{PC}},i}$ (Eq. (4.33)). After FS, the selected input vector $\boldsymbol{\phi}^{\mathrm{DK}}_i$ is used to train candidate LF models on $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'')$ and validate them on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$. The selected LF predictor gives $\hat{y}^{\mathrm{LF}}_i$ on the full LF grid $\mathcal{T}^{\mathrm{LF}}$. At the paired HF timestamps, the residuals $\Delta_j = y^{\mathrm{HF}}_j - \hat{y}^{\mathrm{LF}}_{i(j)}$ are computed for $j \in \mathcal{I}^{\mathrm{HF}}$. A GP model is then trained using $\boldsymbol{\phi}^{\mathrm{MF}}_i = [\hat{y}^{\mathrm{LF}}_i, \boldsymbol{\phi}^{\mathrm{DK},\top}_i]^\top$, and the final corrected signal is obtained as $\hat{y}^{\mathrm{MF}}_{\Delta_{\mathrm{PC}},i} = \hat{y}^{\mathrm{LF}}_i + \hat{\Delta}_i$.

For the GP residual model, five kernel structures from Subsection 4.2.4 are compared: RBF, Scaled RBF, RBF with white noise, multiplicative RBF-periodic with white noise, and additive RBF-periodic with white noise. Kernel selection is based on validation RMSE distributions from repeated GP training runs (100 iterations). After kernel selection, the final GP correction model is retrained on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ and evaluated on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$. All RMSE values in this chapter are computed on standardized signals. The final model comparison is performed only at the paired HF indices $i(j)$, because y_j^{HF} provides the trusted reference. Relative changes are reported with respect to the online analyzer baseline y^{LF}. A negative percentage therefore means lower RMSE and better agreement with the HF reference.

6.2 Tennessee Eastman Process Case Study

The TEP case study uses the already defined LF dataset $\mathbf{X}^{\mathrm{LF}} \in \mathbb{R}^{1000\times 52}$ and the corresponding analyzer output $\mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{1000}$. The monitored output is the concentration of component C in the purge stream, measured by Analyzer A1 (Figure 5.1). Since the benchmark does not provide laboratory measurements, the pseudo-HF reference $\mathbf{y}^{\mathrm{HF}}$ is constructed from $\mathbf{y}^{\mathrm{LF}}$ using Eq. (5.1). The resulting reference contains $n_{\mathrm{HF}} = 28$ samples on $\mathcal{T}^{\mathrm{HF}} \subset \mathcal{T}^{\mathrm{LF}}$. The subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ contains 700 LF samples and 18 paired HF samples, while $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ contains 300 LF samples and 10 paired HF samples. All feature selection, LF model selection, kernel tuning, and residual-model training are carried out only on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$. The subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ is used only for the final RMSE evaluation at the paired HF indices.

Feature selection is carried out only inside the MF training subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$. In this step, the available LF data are $\mathbf{X}^{\mathrm{LF}}_{\mathrm{MF(TR)}}$ and $\mathbf{y}^{\mathrm{LF}}_{\mathrm{MF(TR)}}$, while the paired HF values are available only at the indices $i(j)$, where $j \in \mathcal{I}^{\mathrm{HF}}_{\mathrm{MF(TR)}}$. The testing subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ is not used during feature selection.

The LF FS step uses the inner CV split, where $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}') = \mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'') \cup \mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$. Candidate input subsets are fitted on $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'')$ and evaluated on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ using the LF analyzer output $\mathbf{y}^{\mathrm{LF}}$. Figure 6.1 shows the expanding-window CV splits used for this step. The validation regions contain different signal behaviour, so the selected subset must generalize across several operating segments of $\mathcal{T}^{\mathrm{LF}}$ and not only describe one local part of the trajectory.

For each CV split, PCR, PLS, LASSO, SR, and DK selection construct a candidate regressor vector from the available LF input matrix. PCR and PLS form latent

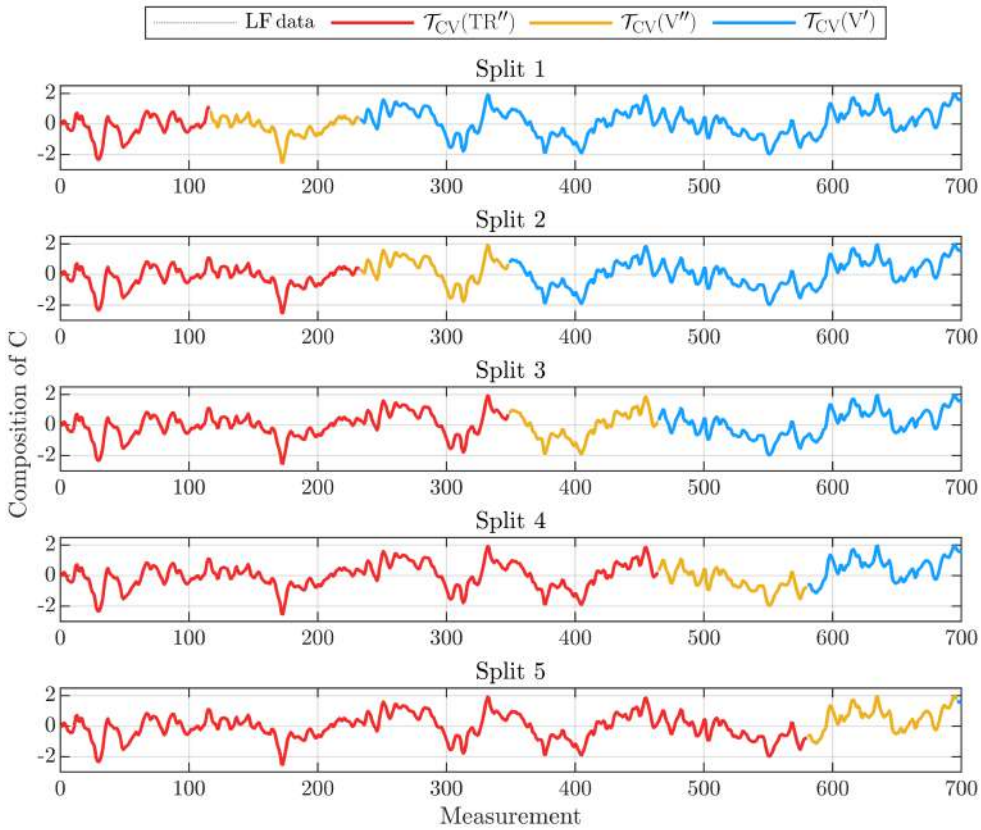


Figure 6.1: Expanding-window CV splits for LF FS in the TEP case study.

Table 6.1: Average validation RMSE on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ for FS methods in the TEP case study.

Feature-selection method	**Average** $\mathbf{RMSE}_{\mathrm{LF}}^{\mathrm{V}''}$
PCR	0.5865
PLS	0.7305
LASSO	0.5954
SR	0.7097
DK	0.5731

regressors from $\mathbf{X}_{\mathrm{CV(TR}'')}^{\mathrm{LF}}$, while LASSO and SR select original columns from $\mathbf{X}_{\mathrm{CV(TR}'')}^{\mathrm{LF}}$. The DK variant keeps a fixed six-variable subset and defines the regressor vector $\boldsymbol{\phi}_i^{\mathrm{DK}} \in \mathbb{R}^6$ for each $i \in \mathcal{I}^{\mathrm{LF}}$. Specifically, the selected variables include (i) feed composition and (ii) total fresh feed flow rate at the DK_1 tag, which represent the overall process throughput; (iii) reactor pressure and (iv) reactor liquid level at the DK_2 tag, which reflect the rate of material conversion in the reaction section; and, since direct measurements of product flow are not available in the dataset, (v) recycle flow at the DK_3 tag and (vi) total fresh C feed at the DK_4 tag, which are used instead to represent material accumulation and flow interactions. The corresponding process locations are shown in Figure 5.1.

Figure 6.2 compares the validation predictions $\hat{y}_{\mathrm{stat},i}^{\mathrm{LF}}$ on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ for the evaluated feature-selection methods. PCR and PLS require many latent components to describe $\mathbf{y}^{\mathrm{LF}}$. LASSO and SR use fewer original variables, but their selected subsets change more strongly between the CV splits. The DK approach keeps the same six regressors across all splits, which makes the resulting feature space more stable for the following LF, HF, and MF models.

Table 6.1 reports the average validation error $\mathrm{RMSE}_{\mathrm{LF}}^{\mathrm{V}''}$ computed on $\mathcal{I}_{\mathrm{CV(V}'')}^{\mathrm{LF}}$ for each feature-selection method. The DK subset gives the lowest average validation RMSE, equal to 0.5731. PCR and LASSO remain close, but PLS and stepwise regression give higher validation errors. Therefore, the DK subset is used in the next modelling steps as $\boldsymbol{\phi}_i^{\mathrm{DK}}$ for the LF, HF, and MF models.

The selected DK regressor vector $\boldsymbol{\phi}_i^{\mathrm{DK}}$ is used to train the single-fidelity models. Table 6.2 compares y_i^{LF} and the candidate predictions ξ_i against y_j^{HF} at the paired HF indices $i(j)$. The online analyzer gives $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{V}''}(y^{\mathrm{LF}}) = 0.5462$. The static LF prediction $\hat{y}_{\mathrm{stat},i}^{\mathrm{LF}}$ reduces this value to 0.4611, while the dynamic LF prediction $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ reduces it further to 0.3879. The nonlinear LF models do not give a clear advantage

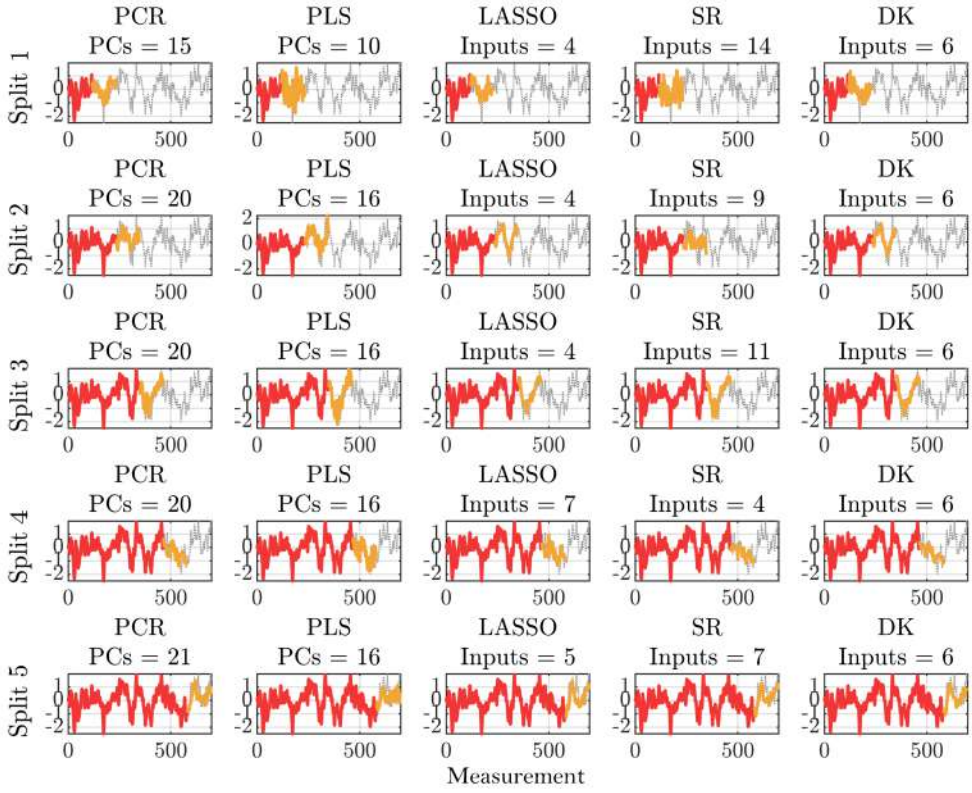


Figure 6.2: Validation predictions $\hat{y}^{\text{stat}}_{\text{LF},i}$ on $\mathcal{T}_{\text{CV}}(\text{V}'')$ across CV splits for different FS methods in the TEP case study.

Table 6.2: RMSE comparison of LF data and single-fidelity models against HF data in the TEP case study.

Model ξ	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TR}''}(\xi)$	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{V}''}(\xi)$
y^{LF}	0.9173	0.5462
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF}}$	0.6123	0.4611
$\hat{y}_{\mathrm{dyn}}^{\mathrm{LF}}$	0.6197	0.3879
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,GP}}$	0.6343	0.5129
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,AE}}$	0.5167	0.4453
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,NN}}$	0.7724	0.4290

over $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$. The dynamic LF model $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ is selected as the baseline predictor for the MF correction. It gives the lowest validation error against $\mathbf{y}^{\mathrm{HF}}$ and keeps the residual-correction task simple. This matters because the GP residual model is trained only at the 18 HF samples in $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$.

The GP correction model uses the dynamic LF prediction $\hat{y}_{\mathrm{dyn}}^{\mathrm{LF}}$ and the DK variables as inputs. The GP residual model is first trained on $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}')$ and validated on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$. Figure 6.3 shows the validation RMSE distributions for the candidate kernels. The Scaled RBF kernel gives the lowest median $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{V}'}$ and the narrowest variability, so it is selected for the final MF training on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$.

Figure 6.4 shows the final predictions on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ and $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$. The LF analyzer y_i^{LF} contains stronger local fluctuations than the constructed HF reference y_j^{HF}. The single-fidelity predictions $\hat{y}_{\mathrm{stat},i}^{\mathrm{LF}}$ and $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ smooth the signal, but the predictor-correction model $\hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}$ gives the closest testing agreement with y_j^{HF} at the paired HF indices.

Table 6.3 shows the final RMSE comparison. The LF analyzer baseline has $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}(y^{\mathrm{LF}}) = 0.9770$. The static and dynamic LF models reduce the testing error by about 30 %. The HF static model $\hat{y}_{\mathrm{stat},i}^{\mathrm{HF}}$ reduces it by 47.69 %. The direct MF model $\hat{y}_i^{\mathrm{MF}}$ gives the lowest training RMSE, but its weaker testing performance indicates overfitting. The analyzer-correction model $\hat{y}_{\Delta_{\mathrm{AC}},i}^{\mathrm{MF}}$ improves only slightly on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$. The predictor-correction model $\hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}$ gives the best result and reduces $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}$ from 0.9770 to 0.4827, which corresponds to a 50.60 % improvement.

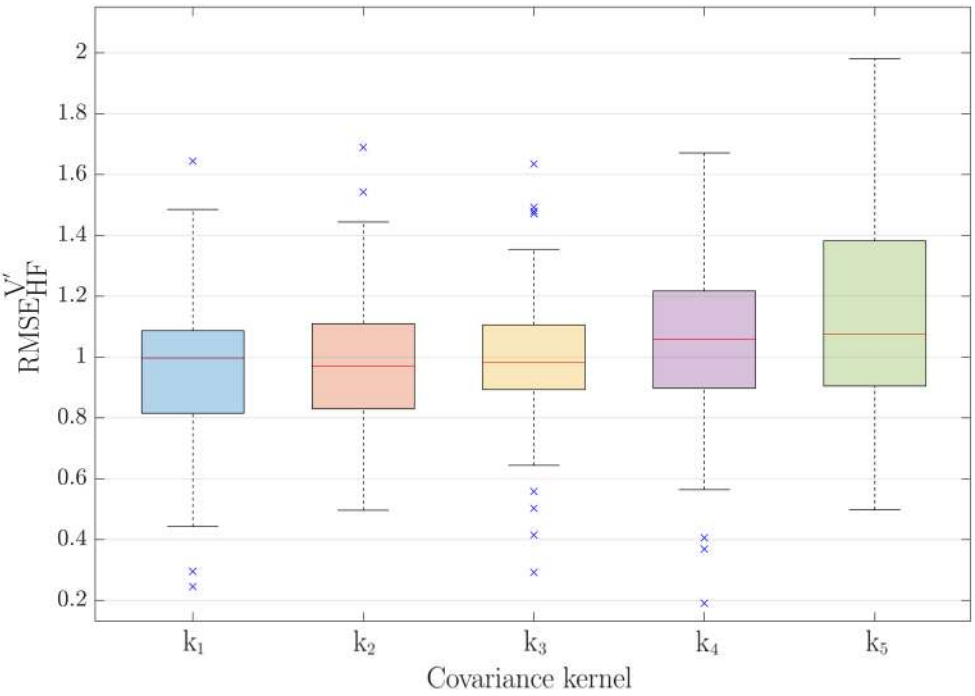


Figure 6.3: Validation RMSE distributions over repeated GP training runs for different kernels in the TEP case study.

Table 6.3: Final RMSE performance against HF data in the TEP case study. Relative changes are computed with respect to the LF analyzer baseline.

Model ξ	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TR}}(\xi)$	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}(\xi)$
y^{LF}	0.8354	0.9770
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF}}$	0.5580 (-33.20 %)	0.6784 (-30.56 %)
$\hat{y}_{\mathrm{dyn}}^{\mathrm{LF}}$	0.5673 (-32.09 %)	0.6810 (-30.30 %)
$\hat{y}_{\mathrm{stat}}^{\mathrm{HF}}$	0.4495 (-46.20 %)	0.5111 (-47.69 %)
$\hat{y}^{\mathrm{MF}}$	0.2463 (-70.52 %)	0.6189 (-36.66 %)
$\hat{y}_{\Delta_{\mathrm{AC}}}^{\mathrm{MF}}$	0.6402 (-23.37 %)	0.8891 (-9.00 %)
$\hat{y}_{\Delta_{\mathrm{PC}}}^{\mathrm{MF}}$	0.4090 (-51.05 %)	0.4827 (-50.60 %)

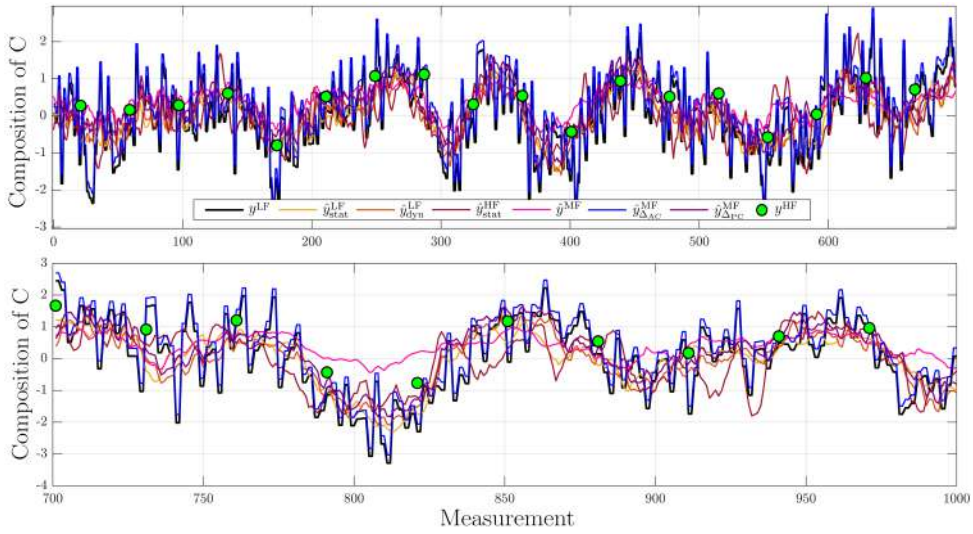


Figure 6.4: Time-series comparison of LF data, HF samples, single-fidelity models, and MF models for the TEP case study.

6.3 Industrial Alkylation Unit Case Study

The second case study uses data from a Stratco alkylation unit at the Slovnaft refinery 5.2. The industrial alkylation case study uses the LF plant dataset $\mathbf{X}^{\mathrm{LF}} \in \mathbb{R}^{25000\times p}$ with the corresponding online analyzer output $\mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{25000}$. The monitored output is the isobutane concentration measured by analyzer A_3. The HF dataset contains $n_{\mathrm{HF}} = 28$ laboratory measurements $\mathbf{y}^{\mathrm{HF}}$ of the same output. After MCD-based preprocessing, 2500 evident outlying LF samples are removed or replaced, which gives the working LF dataset $\mathbf{X}^{\mathrm{LF}} \in \mathbb{R}^{22500\times p}$ and $\mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{22500}$. The outer split follows the same MF partitioning rule as in the TEP case study. The subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ contains 13250 LF samples and 18 paired HF samples, while $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ contains 9250 LF samples and 10 paired HF samples. The final model comparison is evaluated only at the paired HF indices $i(j)$, because $\mathbf{y}^{\mathrm{HF}}$ provides the reference for RMSE calculation.

Feature selection is carried out only inside the MF TR subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$. In this step, the available LF data are $\mathbf{X}^{\mathrm{LF}}_{\mathrm{MF(TR)}}$ and $\mathbf{y}^{\mathrm{LF}}_{\mathrm{MF(TR)}}$, while the paired HF laboratory values are available only at the indices $i(j)$, where $j \in \mathcal{I}^{\mathrm{HF}}_{\mathrm{MF(TR)}}$. The TS subset $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$ is not used during feature selection. The LF FS step uses the inner CV split, where $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}') = \mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'') \cup \mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$. Candidate input subsets are fitted on $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}'')$ and evaluated on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ using the LF analyzer output $\mathbf{y}^{\mathrm{LF}}$. Figure 6.5 shows the expanding-window CV splits used for this step. Compared with the TEP

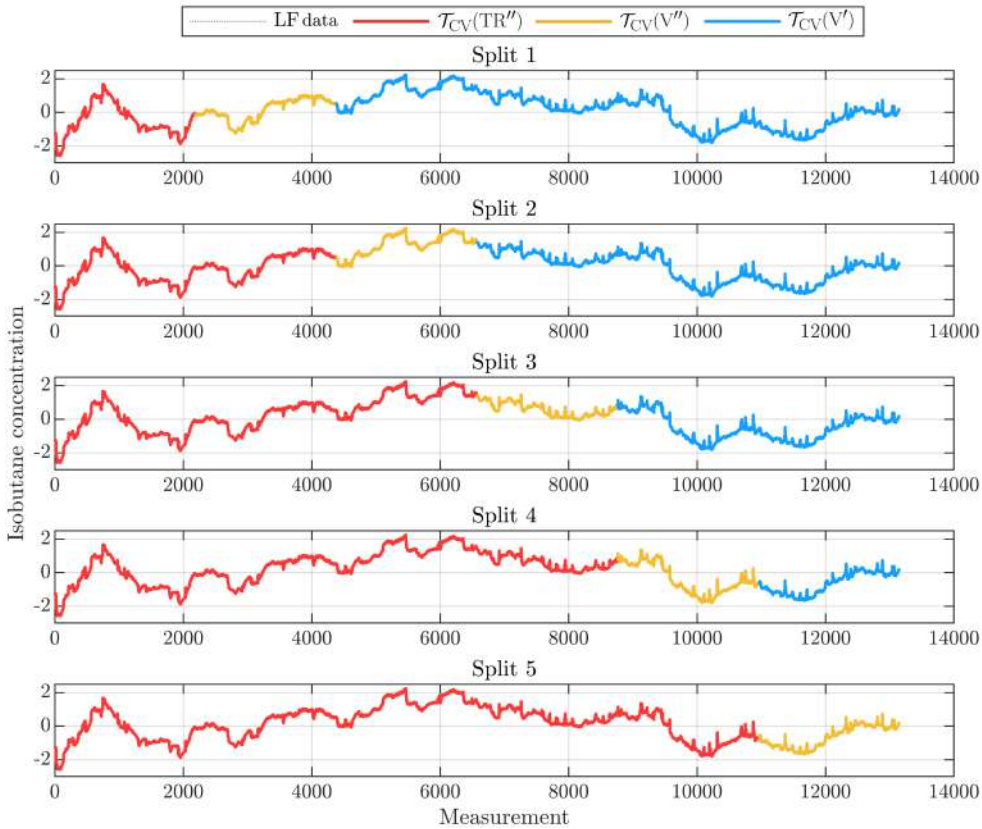


Figure 6.5: Expanding-window CV splits used for LF FS in the industrial alkylation case study.

case study, the validation regions contain more uneven excitation and visible analyzer disturbances. This makes the feature-selection problem more difficult and explains why purely statistical FS methods are less stable.

For each CV split, PCR, PLS, LASSO, SR, and DK selection construct a candidate regressor vector from the available LF input matrix. PCR and PLS form latent regressors from $\mathbf{X}^{\mathrm{LF}}_{\mathrm{CV}(\mathrm{TR}'')}$, while LASSO and SR select original columns from $\mathbf{X}^{\mathrm{LF}}_{\mathrm{CV}(\mathrm{TR}'')}$. The DK variant keeps a fixed six-variable subset and defines the regressor vector

$$\boldsymbol{\phi}_i^{\mathrm{DK}} = \left[x_i^{\mathrm{feed},1},\, x_i^{\mathrm{feed},2},\, x_i^{\mathrm{rec},1},\, x_i^{\mathrm{rec},2},\, x_i^{\mathrm{cat}},\, x_i^{\mathrm{alk}}\right]^{\top} \in \mathbb{R}^6, \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{6.1}$$

Specifically, $x_i^{\mathrm{feed},1}$ and $x_i^{\mathrm{feed},2}$ denote the two selected concentration variables from the

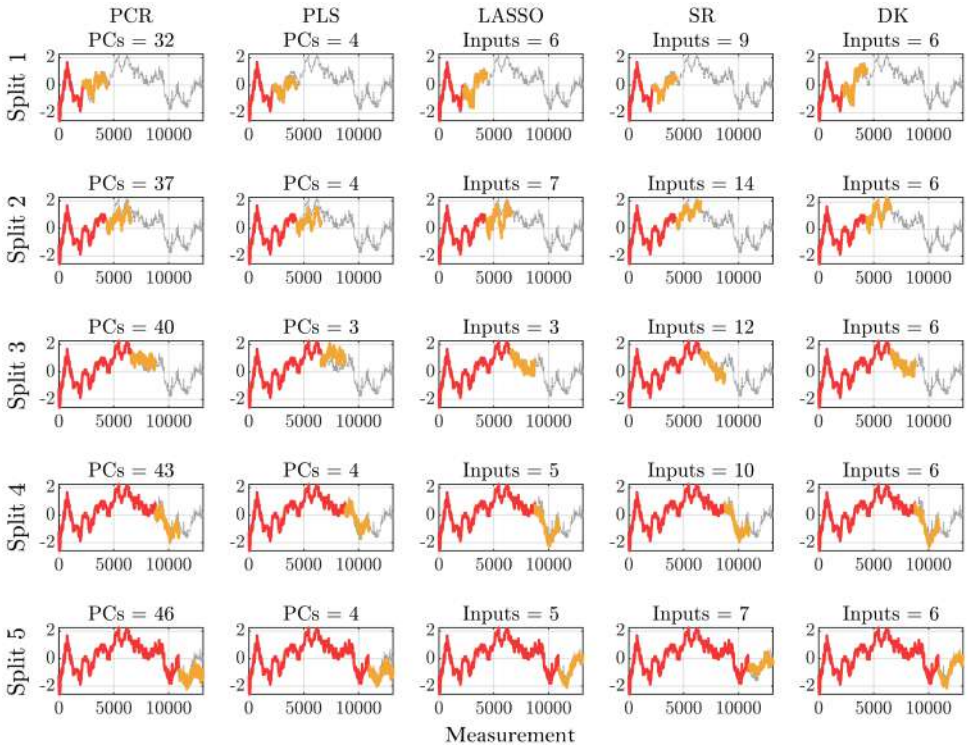


Figure 6.6: Validation predictions $\hat{y}^{\text{stat}}_{\text{LF},i}$ on $\mathcal{T}_{\text{CV}}(\text{V}'')$ across CV splits for different feature-selection methods in the industrial alkylation case study.

feed stream at the DK_1 tag, while $x_i^{\text{rec},1}$ and $x_i^{\text{rec},2}$ denote the two selected concentration variables from the recycle stream at the DK_2 tag. The variable x_i^{cat} denotes the recycled catalyst flow rate at the DK_3 tag, and x_i^{alk} denotes the alkylate product flow rate at the DK_4 tag. These variables correspond to the DK locations shown in Figure 5.2 and describe the main material path linked to the monitored isobutane concentration.

Figure 6.6 compares the validation predictions $\hat{y}^{\text{LF}}_{\text{stat},i}$ on $\mathcal{T}_{\text{CV}}(\text{V}'')$ for the evaluated FS methods. PCR requires 32–46 principal components, which is too large for a compact industrial soft sensor. PLS uses fewer components, but its validation error remains high. LASSO and SR give better prediction quality, but their selected variables change more strongly between the CV splits. The DK approach keeps the same six regressors across all splits, which makes the resulting feature space more stable for the following LF, HF, and MF models.

Table 6.4: Average validation RMSE on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}'')$ for feature-selection methods in the industrial alkylation case study.

Feature-selection method	Average $\mathbf{RMSE}_{\mathrm{LF}}^{\mathrm{V}''}$
PCR	0.5180
PLS	0.6050
LASSO	0.3314
SR	0.3543
DK	0.2851

Table 6.5: RMSE comparison of LF data and single-fidelity models against HF data in the industrial alkylation case study.

Model ξ	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TR}'}(\xi)$	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{V}'}(\xi)$
y^{LF}	0.5071	0.6898
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF}}$	0.2207	0.6275
$\hat{y}_{\mathrm{dyn}}^{\mathrm{LF}}$	0.4467	0.6152
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,GP}}$	0.5296	0.7007
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,AE}}$	0.5337	0.7045
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF,NN}}$	0.5144	0.6482

Table 6.4 reports the average validation error $\mathrm{RMSE}_{\mathrm{LF}}^{\mathrm{V}''}$ computed on $\mathcal{I}_{\mathrm{CV}(\mathrm{V}'')}^{\mathrm{LF}}$ for each FS method. The DK subset gives the lowest average validation RMSE, equal to 0.2851. LASSO is the best purely statistical method, with average RMSE equal to 0.3314, while PCR and PLS require larger latent spaces and still give higher validation errors. Therefore, the DK subset is used in the next modelling steps as $\boldsymbol{\phi}_i^{\mathrm{DK}}$ for the LF, HF, and MF models.

Table 6.5 compares y_i^{LF} and the candidate single-fidelity predictions ξ_i against y_j^{HF} at the paired HF indices $i(j)$. The online analyzer gives $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{V}'}(y^{\mathrm{LF}}) = 0.6898$. The static LF prediction $\hat{y}_{\mathrm{stat},i}^{\mathrm{LF}}$ reduces this value to 0.6275, and the dynamic LF prediction $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ reduces it to 0.6152. The nonlinear models do not improve the validation result. The GP and AE variants perform worse than y_i^{LF} on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$, while the NN provides only a moderate reduction.

The dynamic LF model $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ is selected for the correction stage. Although its numerical gain over $\hat{y}_{\mathrm{stat},i}^{\mathrm{LF}}$ is not large, it gives smoother predictions and better represents delayed responses caused by recycle streams and residence time between the reaction and

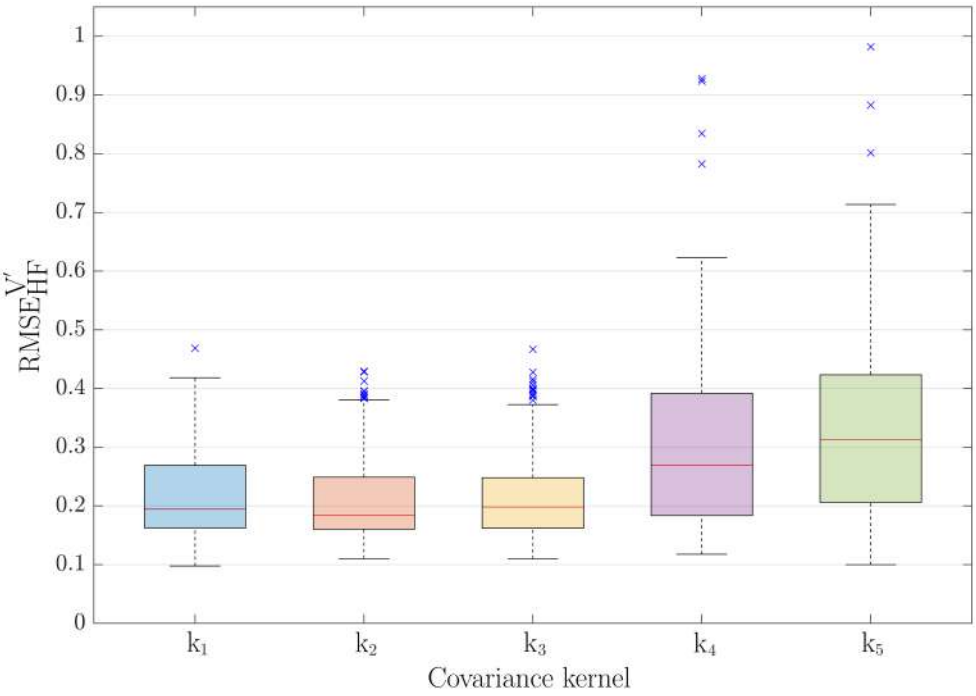


Figure 6.7: Validation RMSE distributions over repeated GP training runs for different kernels in the industrial alkylation case study.

separation sections.

The GP residual model uses the same predictor-correction regressor structure as in the TEP case:

$$\boldsymbol{\phi}_i^{\mathrm{MF}} = \left[\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}, \boldsymbol{\phi}_i^{\mathrm{DK},\top}\right]^{\top}. \tag{6.2}$$

The GP model is trained on the residuals $\Delta_j = y_j^{\mathrm{HF}} - \hat{y}_{\mathrm{dyn},i(j)}^{\mathrm{LF}}$ using $\mathcal{T}_{\mathrm{CV}}(\mathrm{TR}')$ and validated on $\mathcal{T}_{\mathrm{CV}}(\mathrm{V}')$. Figure 6.7 shows the validation RMSE distributions for the five candidate kernels. The Scaled RBF kernel again gives the lowest median $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{V}'}$ and the narrowest interquartile range, so it is selected for the final GP correction model.

Figure 6.8 shows the final predictions on $\mathcal{T}_{\mathrm{MF}}(\mathrm{TR})$ and $\mathcal{T}_{\mathrm{MF}}(\mathrm{TS})$. The analyzer signal y_i^{LF} follows the broad process trend, but it shows local deviations from the laboratory samples y_j^{HF}. The LF models reproduce part of this behaviour, but they do not correct the bias to the laboratory reference. The HF-only model improves the prediction at the paired HF indices, but it remains limited by the small number of laboratory samples.

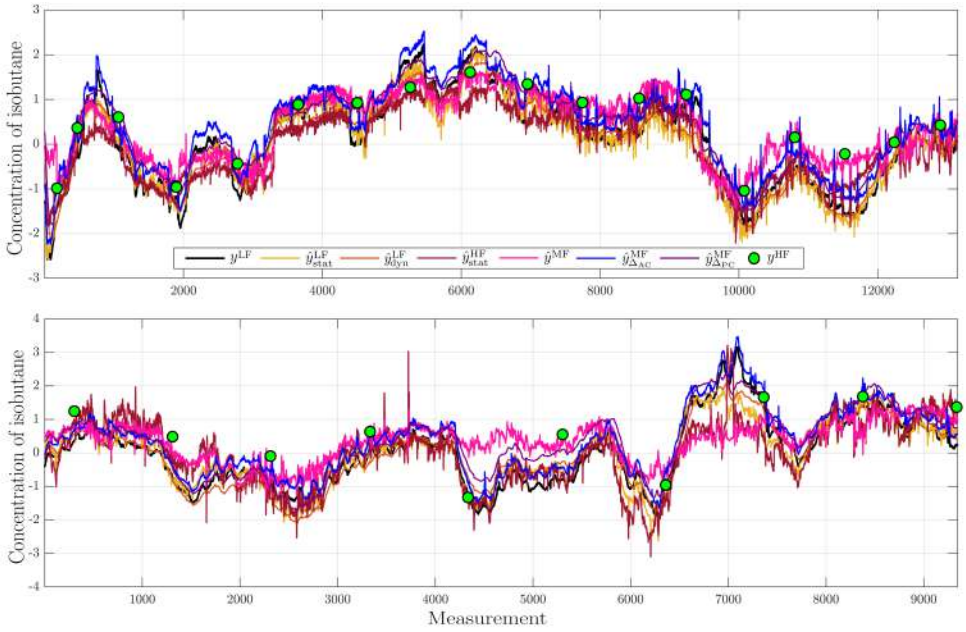


Figure 6.8: Time-series comparison of LF data, HF samples, single-fidelity models, and MF models for the industrial alkylation case study.

The residual-correction models provide the clearest improvement.

Table 6.6 shows the final RMSE values. The LF analyzer baseline has $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}(y^{\mathrm{LF}}) = 0.7945$. The static and dynamic LF models increase the testing RMSE to 0.8534 and 0.8222, respectively, which means that LF-only modelling reproduces the analyzer behaviour but does not improve agreement with $\mathbf{y}^{\mathrm{HF}}$. The HF static model $\hat{y}_{\mathrm{stat},i}^{\mathrm{HF}}$ reduces the testing RMSE to 0.7447. The direct MF model $\hat{y}_i^{\mathrm{MF}}$ gives very low training RMSE, equal to 0.0722, but testing RMSE remains 0.7715, which indicates overfitting. The analyzer-correction model $\hat{y}_{\Delta_{\mathrm{AC}},i}^{\mathrm{MF}}$ reduces the testing RMSE to 0.6136, while the predictor-correction model $\hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}$ gives the best testing RMSE, equal to 0.6071. This corresponds to a 23.59 % improvement relative to y_i^{LF}.

The industrial case confirms the same ranking as the benchmark case. The best result is achieved when $\hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}}$ first extracts the dense process trend and the GP model then corrects its residual to the sparse laboratory samples y_j^{HF}. Direct correction of y_i^{LF} is less reliable because the residual still contains local analyzer fluctuations. Direct MF regression is also less reliable because $n_{\mathrm{HF}} = 28$ is too small for a flexible model.

Table 6.6: Final RMSE performance against HF data in the industrial alkylation case study. Relative changes are computed with respect to the LF analyzer baseline.

Model ξ	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TR}}(\xi)$	$\mathbf{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}(\xi)$
y^{LF}	0.6351	0.7945
$\hat{y}_{\mathrm{stat}}^{\mathrm{LF}}$	0.6280 (-1.10 %)	0.8534 (+7.41 %)
$\hat{y}_{\mathrm{dyn}}^{\mathrm{LF}}$	0.6125 (-3.56 %)	0.8222 (+3.48 %)
$\hat{y}_{\mathrm{stat}}^{\mathrm{HF}}$	0.4441 (-30.07 %)	0.7447 (-6.28 %)
$\hat{y}^{\mathrm{MF}}$	0.0722 (-88.63 %)	0.7715 (-2.90 %)
$\hat{y}_{\Delta_{\mathrm{AC}}}^{\mathrm{MF}}$	0.4301 (-32.27 %)	0.6136 (-22.77 %)
$\hat{y}_{\Delta_{\mathrm{PC}}}^{\mathrm{MF}}$	0.3698 (-41.77 %)	0.6071 (-23.59 %)

6.4 Discussion

The two case studies show the same main result, but under different levels of difficulty. In both systems, $\mathbf{y}^{\mathrm{LF}}$ provides the dense process trajectory, but it does not always follow $\mathbf{y}^{\mathrm{HF}}$. A HF-only model uses the more reliable values y_j^{HF}, but the sparse set $\mathcal{T}^{\mathrm{HF}}$ does not contain enough samples to describe the full trajectory on $\mathcal{T}^{\mathrm{LF}}$. The predictor-correction MF model combines both sources by first extracting the dense trend through $\hat{y}_i^{\mathrm{LF}}$ and then correcting this trend using the GP residual model trained on $\Delta_j = y_j^{\mathrm{HF}} - \hat{y}_{i(j)}^{\mathrm{LF}}$. The improvement is larger in the TEP benchmark, where $\hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}$ reduces $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}$ from 0.9770 to 0.4827, which corresponds to a 50.60 % improvement relative to y_i^{LF}. This result is expected because $\mathbf{y}^{\mathrm{HF}}$ is constructed from $\mathbf{y}^{\mathrm{LF}}$ by smoothing and downsampling. The main task is therefore to remove short-term fluctuations and reconstruct the smoother trend. The industrial alkylation case is more demanding because $\mathbf{y}^{\mathrm{HF}}$ comes from real laboratory measurements and the mismatch contains analyzer drift, bias, and local faulty intervals. In this case, the same predictor-correction structure reduces $\mathrm{RMSE}_{\mathrm{HF}}^{\mathrm{TS}}$ from 0.7945 to 0.6071, which corresponds to a 23.59 % improvement. The model comparison also shows why the correction must be constrained. Nonlinear single-fidelity models do not consistently improve the validation error at the paired HF indices. Direct MF regression can strongly overfit when n_{HF} is small. This behaviour appears in both case studies: $\hat{y}_i^{\mathrm{MF}}$ reaches very low training RMSE, but its testing performance does not follow the same trend. The useful part of the MF structure is therefore not flexibility alone, but the construction of $\boldsymbol{\phi}_i^{\mathrm{MF}} = [\hat{y}_i^{\mathrm{LF}}, \boldsymbol{\phi}_i^{\mathrm{DK},\top}]^\top$, where the LF prediction carries the dense process trend and the GP correction learns only the remaining HF residual. Feature selection plays the same role in both applications. The DK regressor vector

$\boldsymbol{\phi}_i^{\mathrm{DK}}$ gives the most consistent validation behaviour and keeps the model interpretable. This matters because the selected columns of $\mathbf{X}^{\mathrm{LF}}$ must not only reduce RMSE, but also remain available, meaningful, and maintainable during plant operation. Purely statistical methods can select useful variables, but they may also select unstable, redundant, or less suitable signals for long-term deployment. The DK feature set therefore acts as a practical constraint on the model and helps avoid unnecessary complexity.

The final implementation recommendation from this chapter is

$$\boldsymbol{\phi}_i^{\mathrm{DK}} \rightarrow \hat{y}_{\mathrm{dyn},i}^{\mathrm{LF}} \rightarrow \Delta_j = y_j^{\mathrm{HF}} - \hat{y}_{\mathrm{dyn},i(j)}^{\mathrm{LF}} \rightarrow \hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}.$$

This sequence gives the best trade-off between accuracy, interpretability, and stability. The LF predictor should not be treated only as an intermediate result. It should enter the MF correction model as part of $\boldsymbol{\phi}_i^{\mathrm{MF}}$, because it carries information from the dense LF timeline that the sparse HF data cannot provide alone. Based on the TEP and industrial alkylation results, $\hat{y}_{\Delta_{\mathrm{PC}},i}^{\mathrm{MF}}$ is selected as the preferred data-based MF soft sensor for the following parts of the thesis.

Chapter 7

Physics-Guided Soft Sensing for Anomaly Detection: Simulation Surrogates in Industrial Processes

This chapter extends the MF monitoring workflow from the previous chapter toward anomaly detection. The objective is no longer only to improve the estimate of the monitored output at the sparse HF timestamps. The objective is to train a predictor on the LF plant data to identify anomalous deviations in the online analyzer signal. The chapter uses the industrial alkylation unit introduced in Chapter 5. In contrast to Chapter 6, where the LF baseline is learned only from plant data, this chapter also uses a simplified process simulator as an additional LF information source. Its role is to provide a physically consistent trend based on the selected DK variables. This trend is then transferred to the industrial dataset and corrected using the sparse HF laboratory samples.

The implementation therefore combines three data sources: the simulation dataset $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$, the dense LF plant dataset $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$, and the sparse HF laboratory values $\mathbf{y}^{\mathrm{HF}}$. The workflow follows the same residual-correction principle as the previous chapter, but the LF prediction now comes from a simulation-trained surrogate. The corrected simulation-supported predictor is then used both for soft sensing and for anomaly detection. The methodology in this chapter is based on the submitted manuscript by Fáber et al., "Physics-Guided Soft Sensing for Anomaly Detection: Simulation Surrogates in Industrial Processes", submitted to Chemical Engineering Research and Design.

7.1 Implementation Workflow

The simulation-supported workflow starts from the DK input subset ($\boldsymbol{\phi}_i^{\mathrm{DK}} \in \mathbb{R}^6$) selected for the industrial alkylation unit. This subset defines the common feature space shared by the industrial plant data and the simulation data. The reduced plant input matrix is denoted by $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$, while the simulation input matrix is denoted by $\mathbf{X}^{\mathrm{LF_s}}$. Both matrices contain the same six selected variables. The corresponding plant analyzer output is $\mathbf{y}^{\mathrm{LF}}$, the simulation output is $\mathbf{y}^{\mathrm{LF_s}}$, and the laboratory reference is $\mathbf{y}^{\mathrm{HF}}$.

A simulation surrogate is first trained on the simulation dataset,

$$\hat{y}_k^{\mathrm{s}} = f_{\mathrm{s}}\left(\mathbf{x}_k^{\mathrm{LF_s}}, \boldsymbol{\theta}_{\mathrm{s}}\right), \qquad k \in \mathcal{I}^{\mathrm{LF_s}}. \tag{7.1}$$

After training, the same surrogate is evaluated on the industrial plant inputs,

$$\hat{y}_i^{\mathrm{s}} = f_{\mathrm{s}}\left(\mathbf{x}_{\mathrm{FS},i}^{\mathrm{LF}}, \boldsymbol{\theta}_{\mathrm{s}}\right), \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{7.2}$$

This produces a dense simulation-supported prediction on the plant time grid $\mathcal{T}^{\mathrm{LF}}$. Since the simplified simulator does not reproduce the industrial plant exactly, the prediction $\hat{y}_i^{\mathrm{s}}$ can contain systematic bias. The sparse HF laboratory samples are therefore used to correct this mismatch. At the paired HF indices, the simulation-to-laboratory residual is defined as

$$\Delta_j^{\mathrm{s}} = y_j^{\mathrm{HF}} - \hat{y}_{i(j)}^{\mathrm{s}}, \qquad j \in \mathcal{I}^{\mathrm{HF}}. \tag{7.3}$$

A GP residual model is then trained on the paired data $\left(\mathbf{x}_{\mathrm{FS},i(j)}^{\mathrm{LF}}, \Delta_j^{\mathrm{s}}\right)$. The predicted residual on the full plant grid is

$$\hat{\Delta}_i^{\mathrm{s}} = g_{\mathrm{s}}\left(\mathbf{x}_{\mathrm{FS},i}^{\mathrm{LF}}\right), \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{7.4}$$

The bias-corrected simulation-supported prediction is then

$$\hat{y}_i^{\mathrm{b}} = \hat{y}_i^{\mathrm{s}} + \hat{\Delta}_i^{\mathrm{s}}, \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{7.5}$$

Here, the superscript s denotes the simulation-supported surrogate prediction, while the superscript b denotes the bias-corrected prediction. This notation is used in the following sections for both the OLS and NN simulation surrogates.

For anomaly detection, this chapter uses the reference-trajectory and threshold-based pGT construction already defined in Section 2.4.2. The reference trajectory is denoted by $\hat{y}_i^{\mathrm{ref}}$ and is obtained from the MF residual-correction soft sensor from Chapter 6. The

case-specific tolerance δ_{pGT} is computed from the mean absolute LF–HF disagreement at the paired HF timestamps, using the already defined pairing rule $i(j)$. The resulting pGT labels y_i^{pGT} are then assigned on the LF plant grid $\mathcal{T}^{\mathrm{LF}}$ by comparing y_i^{LF} with $\hat{y}_i^{\mathrm{ref}}$.

Each candidate predictor ξ_i is converted into an anomaly detector using the threshold-based classification rule from Section 3.5. The detector score is the absolute deviation between the online analyzer and the candidate reference,

$$s_i(\xi) = \left| y_i^{\mathrm{LF}} - \xi_i \right|, \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{7.6}$$

The detector threshold δ^* is optimised only on the TR subset using Eq. (3.16) with $a = b = 1$, and the selected threshold is applied unchanged to the TS subset.

7.2 Industrial Alkylation Data and Simulation Model

The industrial plant dataset initially contains 10000 LF samples and 731 measured variables. The corresponding input matrix is

$$\mathbf{X}^{\mathrm{LF}} \in \mathbb{R}^{10000\times 731}, \qquad \mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{10000}. \tag{7.7}$$

The monitored output is the recycled isobutane concentration measured by the online analyzer. The HF laboratory dataset contains

$$\mathbf{y}^{\mathrm{HF}} \in \mathbb{R}^{11}, \tag{7.8}$$

with laboratory samples available only at sparse timestamps $\mathcal{T}^{\mathrm{HF}}$. Since the laboratory data provide only sparse output values, each laboratory sample is paired with the closest LF plant timestamp through the mapping $i(j)$.

The analysis uses the same six-variable DK feature space, shown in Figure 5.2, as the simplified simulation model. The selected feature set is

$$\boldsymbol{\phi}_i^{\mathrm{DK}} = \left[x_i^{\mathrm{feed},1},\, x_i^{\mathrm{feed},2},\, x_i^{\mathrm{rec},1},\, x_i^{\mathrm{rec},2},\, x_i^{\mathrm{cat}},\, x_i^{\mathrm{alk}} \right]^{\top} \in \mathbb{R}^6, \qquad i \in \mathcal{I}^{\mathrm{LF}}, \tag{7.9}$$

denoting analyzer-based composition measurements, and flow-rate measurements. Restricting the original plant input matrix to this subset gives

$$\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}} \in \mathbb{R}^{10000\times 6}. \tag{7.10}$$

The paired HF input matrix is obtained by selecting the rows $\mathbf{x}_{\mathrm{FS},i(j)}^{\mathrm{LF}}$ at the matched HF timestamps, which gives

$$\mathbf{X}_{\mathrm{FS}}^{\mathrm{HF}} \in \mathbb{R}^{11\times 6}. \tag{7.11}$$

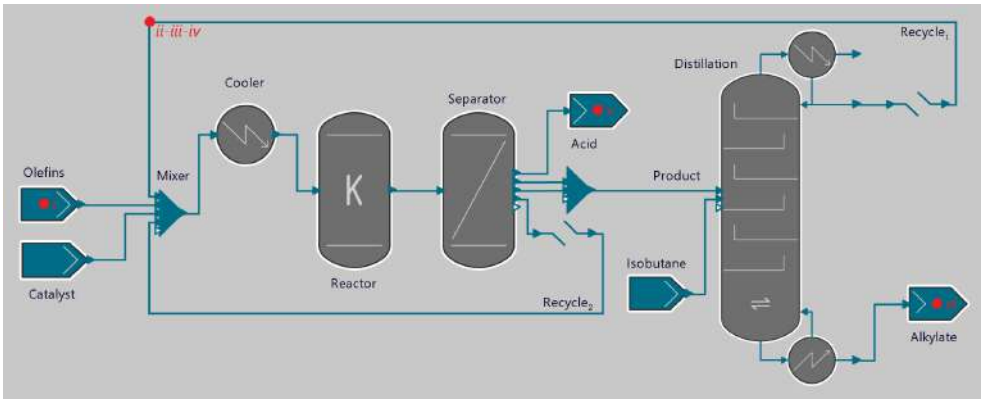


Figure 7.1: Simplified steady-state alkylation model implemented in gPROMS and used to generate the simulation dataset $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$. The highlighted variables define the six-dimensional DK input space.

The simulation dataset is generated from a simplified steady-state model of the alkylation unit. The model is implemented in gPROMS ProcessBuilder and represents the main material-flow structure of the unit 7.1. It contains the mixing section, reactor block, separator, distillation column, and recycle paths. The purpose of this model is not to reproduce all dynamic and kinetic details of the plant. Its purpose is to generate physically consistent data in the same six-variable DK input space.

To verify the correctness of the implementation, an equivalent steady-state model is independently modelled in AVEVA Process Simulation, shown in Figure 7.2. The two implementations were compared under identical operating conditions and were found to converge to comparable steady-state solutions. Since the focus of this work is large-scale data generation and analysis, the gPROMS implementation was selected for further use due to its more convenient workflow for simulation management and data export. The resulting simulation dataset is

$$\mathbf{X}^{\mathrm{LF_s}} \in \mathbb{R}^{5000\times 6}, \qquad \mathbf{y}^{\mathrm{LF_s}} \in \mathbb{R}^{5000}. \tag{7.12}$$

Table 7.1 summarizes the data sources used in this chapter. The important point is that the plant and simulation datasets are both represented in the same six-dimensional feature space, while the HF data provide only sparse reference values at the paired plant timestamps.

Before training the simulation surrogates, the simulation input space is compared with the industrial plant input space. This step checks whether the simplified simulator

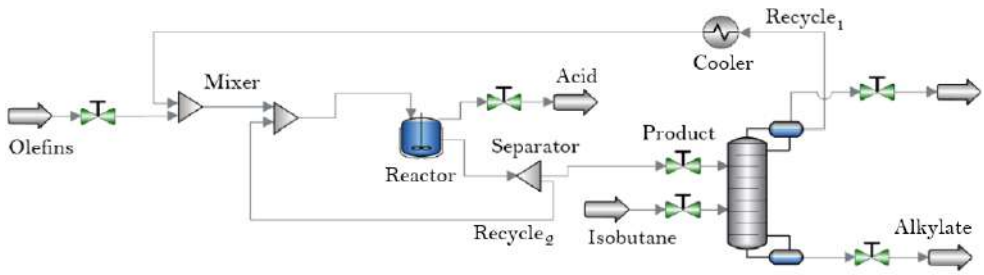


Figure 7.2: Equivalent simplified steady-state alkylation model implemented in AVEVA Process Simulation.

Table 7.1: Datasets used in the simulation-supported anomaly-detection case study.

Data source	Input matrix	Output vector
LF simulation data	$\mathbf{X}^{\mathrm{LF_s}} \in \mathbb{R}^{5000\times 6}$	$\mathbf{y}^{\mathrm{LF_s}} \in \mathbb{R}^{5000}$
LF plant data	$\mathbf{X}^{\mathrm{LF}}_{\mathrm{FS}} \in \mathbb{R}^{10000\times 6}$	$\mathbf{y}^{\mathrm{LF}} \in \mathbb{R}^{10000}$
HF laboratory data	$\mathbf{X}^{\mathrm{HF}}_{\mathrm{FS}} \in \mathbb{R}^{11\times 6}$	$\mathbf{y}^{\mathrm{HF}} \in \mathbb{R}^{11}$

covers the relevant operating region of the plant. If the simulation data were located far away from the plant data, the surrogate would mainly extrapolate when evaluated on $\mathbf{X}^{\mathrm{LF}}_{\mathrm{FS}}$, and the following bias correction would become unreliable.

Figure 7.3 compares the empirical distributions of the six selected DK variables. The industrial plant samples from $\mathbf{X}^{\mathrm{LF}}_{\mathrm{FS}}$ and the simulation samples from $\mathbf{X}^{\mathrm{LF_s}}$ show a reasonable overlap for most variables. Some variables show narrower simulation ranges, which is expected because the simplified model is sampled within prescribed steady-state operating limits and does not include all plant disturbances. Nevertheless, the histograms indicate that the simulation dataset covers the main plant operating region sufficiently for surrogate training.

The joint input-space coverage is further checked using PCA. The PCA projection is computed from the standardised six-dimensional feature space. Figure 7.4 shows the projection of the simulation samples, industrial plant samples, and paired HF samples onto the first two principal components. The HF samples lie inside the operating region covered by the plant and simulation data. This confirms that the sparse laboratory samples do not represent an isolated operating region. The simulation samples also cover the same main region as the industrial data, although the spread differs because the simulator covers a slightly broader operating range than the industrial data..

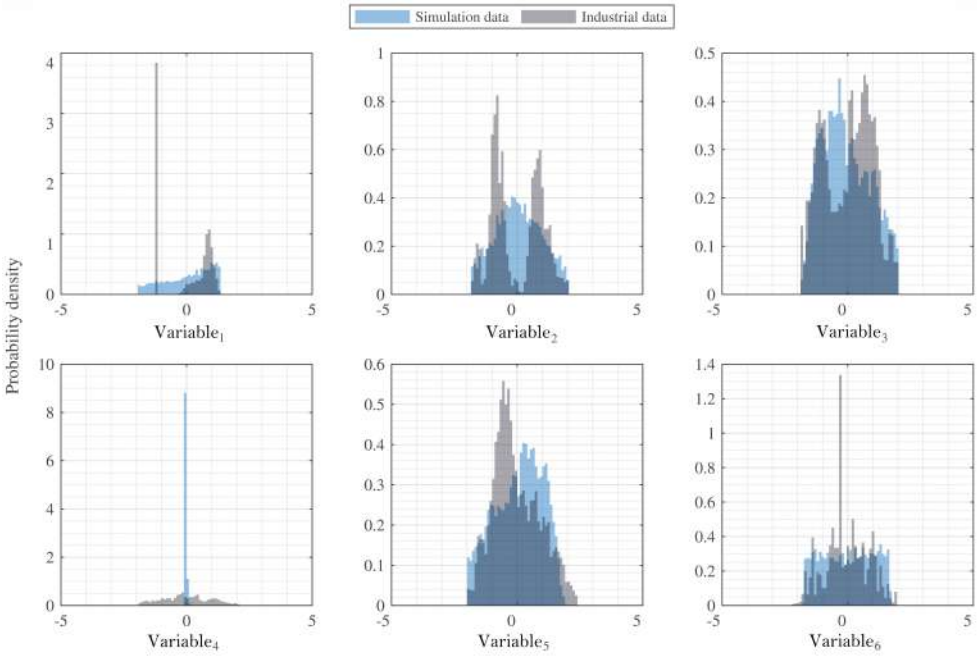


Figure 7.3: Histogram comparison of the six selected input variables in the industrial plant dataset $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$ and the simulation dataset $\mathbf{X}^{\mathrm{LF_s}}$.

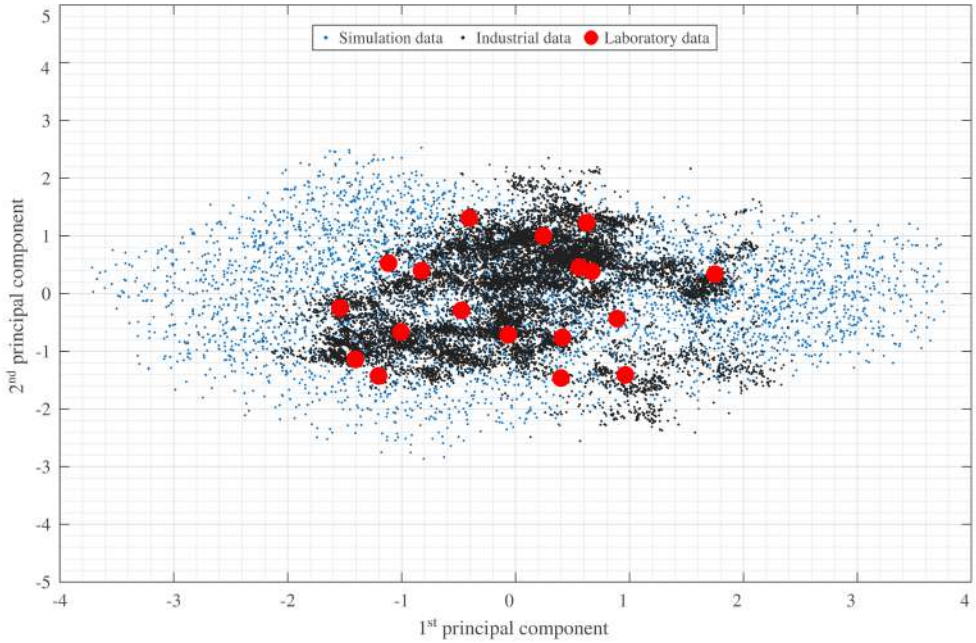


Figure 7.4: Projection of simulation samples $\mathbf{X}^{\mathrm{LF}_s}$, industrial samples $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$, and paired HF samples $\mathbf{X}_{\mathrm{FS}}^{\mathrm{HF}}$ onto the first two principal components of the selected input space.

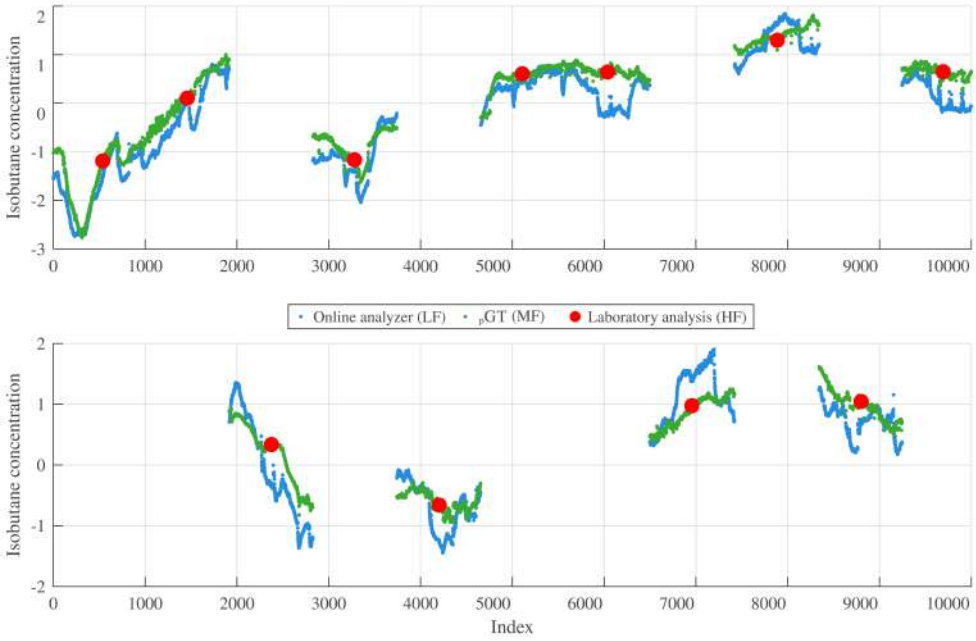


Figure 7.5: Training and testing split on the industrial alkylation time series. The dense LF analyzer signal y_i^{LF}, sparse HF laboratory values y_j^{HF}, and MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$ are shown on the plant time grid.

Training and testing are defined from the available HF samples. Since only $n_{\mathrm{HF}} = 11$ laboratory values are available, the split is driven by the HF indices. The HF samples are assigned to TR and TS using an alternating pattern that targets an approximate 60/40 division. This gives 7 HF samples in the TR subset and 4 HF samples in the TS subset:

$$\mathcal{I}^{\mathrm{HF}} = \mathcal{I}_{\mathrm{TR}}^{\mathrm{HF}} \cup \mathcal{I}_{\mathrm{TS}}^{\mathrm{HF}}, \qquad \mathrm{card}\left(\mathcal{I}_{\mathrm{TR}}^{\mathrm{HF}}\right) = 7, \qquad \mathrm{card}\left(\mathcal{I}_{\mathrm{TS}}^{\mathrm{HF}}\right) = 4. \tag{7.13}$$

The full LF plant timeline is then partitioned into intervals around the HF indices. Each plant sample inherits the TR or TS label of the nearest HF sample. This gives a plant-grid split that is consistent with the sparse laboratory sampling structure.

Figure 7.5 shows the resulting split on the monitored isobutane concentration. The online analyzer signal y_i^{LF} is shown on the dense plant grid, the HF laboratory samples y_j^{HF} are shown at the paired timestamps, and the MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$ is shown on the full LF grid. The reference trajectory is used later for pseudo ground-truth construction.

The first group of predictors is trained directly from plant data. The plant-data baseline $\hat{y}_i^{\mathrm{LF,OLS}}$ is obtained by fitting an OLS model from $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$ to the LF analyzer output $\mathbf{y}^{\mathrm{LF}}$ on the TR subset. This represents the usual data-driven soft-sensor approach where the model learns from available online plant measurements.

Two additional predictors are trained only from the paired HF samples. The HF linear model $\hat{y}_i^{\mathrm{HF,OLS}}$ and the HF nonlinear model $\hat{y}_i^{\mathrm{HF,GPR}}$ are fitted using $\mathbf{X}_{\mathrm{FS}}^{\mathrm{HF}}$ and $\mathbf{y}^{\mathrm{HF}}$ on the TR subset. These models use the most reliable output values, but only 7 HF samples are available for training. They therefore represent a useful reference, but they are expected to overfit.

The second group of predictors is trained from simulation data. Two simulation surrogates are considered. The first surrogate is an OLS model trained on $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$. After training, it is evaluated on $\mathbf{X}_{\mathrm{FS}}^{\mathrm{LF}}$ to produce the plant-grid prediction $\hat{y}_i^{\mathrm{s,OLS}}$. The second surrogate is a feed-forward neural network trained on the same simulation data. The network uses four dense layers with widths 6–3–2–1 and activation functions `selu`, `tanh`, `selu`, and linear. After training, it is evaluated on the industrial plant inputs to produce $\hat{y}_i^{\mathrm{s,NN}}$.

The third group contains the bias-corrected simulation surrogates. For the OLS simulation surrogate, the HF residuals are

$$\Delta_j^{\mathrm{OLS}} = y_j^{\mathrm{HF}} - \hat{y}_{i(j)}^{\mathrm{s,OLS}}, \qquad j \in \mathcal{I}_{\mathrm{TR}}^{\mathrm{HF}}. \tag{7.14}$$

A GP residual model is trained on these residuals and evaluated on the LF plant grid. The corrected OLS simulation-supported prediction is

$$\hat{y}_i^{\mathrm{b,OLS}} = \hat{y}_i^{\mathrm{s,OLS}} + \hat{\Delta}_i^{\mathrm{OLS}}, \qquad i \in \mathcal{I}^{\mathrm{LF}}. \tag{7.15}$$

The same procedure is used for the NN simulation surrogate, producing $\hat{y}_i^{\mathrm{b,NN}}$. The final set of evaluated predictors is therefore

$$\xi_i \in \left\{ y_i^{\mathrm{LF}},\, \hat{y}_i^{\mathrm{LF,OLS}},\, \hat{y}_i^{\mathrm{HF,OLS}},\, \hat{y}_i^{\mathrm{HF,GPR}},\, \hat{y}_i^{\mathrm{s,OLS}},\, \hat{y}_i^{\mathrm{s,NN}},\, \hat{y}_i^{\mathrm{b,OLS}},\, \hat{y}_i^{\mathrm{b,NN}} \right\}. \tag{7.16}$$

All predictors are evaluated against $\mathbf{y}^{\mathrm{HF}}$ only at the paired HF indices.

The plant dataset does not contain manually validated anomaly labels. Therefore, pseudo ground-truth labels are constructed using the MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$ and the tolerance threshold δ_{pGT} from Section. (2.4.2). The reference trajectory approximates the laboratory-quality trend on the LF plant grid. It does not represent a perfect fault record, but it provides a reproducible reference for comparing detectors under the same definition. Figure 7.6 shows the tolerance band around $\hat{y}_i^{\mathrm{ref}}$. A sample

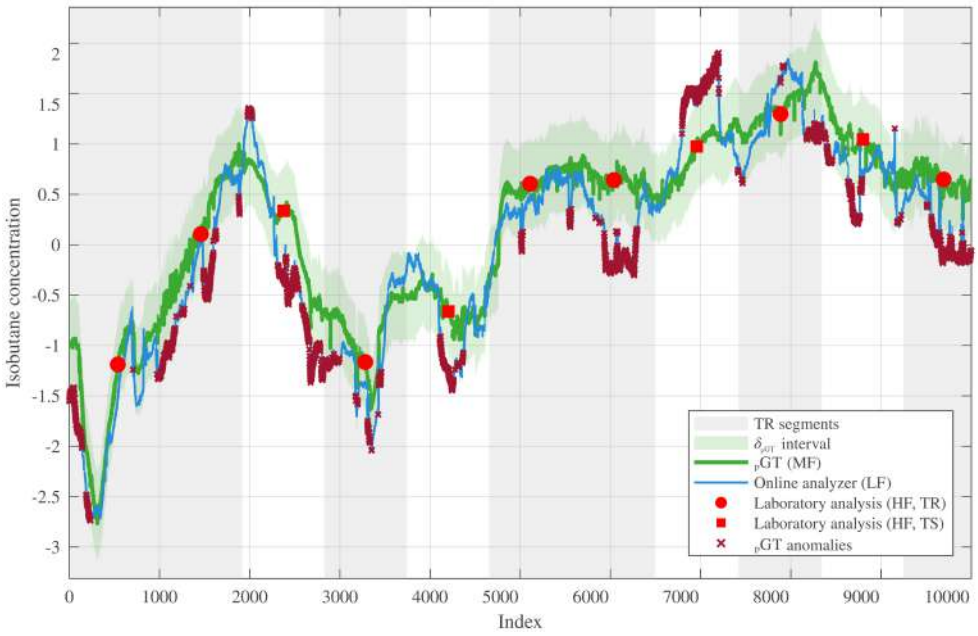


Figure 7.6: Tolerance band used for pseudo ground-truth construction around the MF reference trajectory $\hat{y}_i^{\mathrm{ref}}$. Samples where $|y_i^{\mathrm{LF}} - \hat{y}_i^{\mathrm{ref}}| > \delta_{\mathrm{pGT}}$ are labelled as anomalous.

is treated as anomalous when the LF analyzer value y_i^{LF} leaves this band. The resulting labels capture several types of behaviour. Isolated point outliers are visible around indices 1900, 5000, and 5500. Gradual deviations from the reference appear near indices 2000 and 10000. Consecutive anomalous segments appear around indices 6000 and 7000. These patterns correspond to the point, contextual, and subsequence anomalies introduced earlier in the thesis.

Each candidate predictor ξ_i is converted into a detector by forming a deviation score on the plant grid, as defined in Eq. (7.6). For each predictor, the detector threshold δ is optimised on the TR subset using the objective

$$W(\delta) = a\,\mathrm{Sens}(\delta) + b\,\mathrm{Spec}(\delta), \qquad a = b = 1. \tag{7.17}$$

The optimised threshold δ^* is then fixed and applied to the TS subset. The detector decision is

$$y_i^{\mathrm{det}}(\xi) = \begin{cases} 1, & s_i(\xi) > \delta^*, \\ 0, & s_i(\xi) \le \delta^*. \end{cases} \tag{7.18}$$

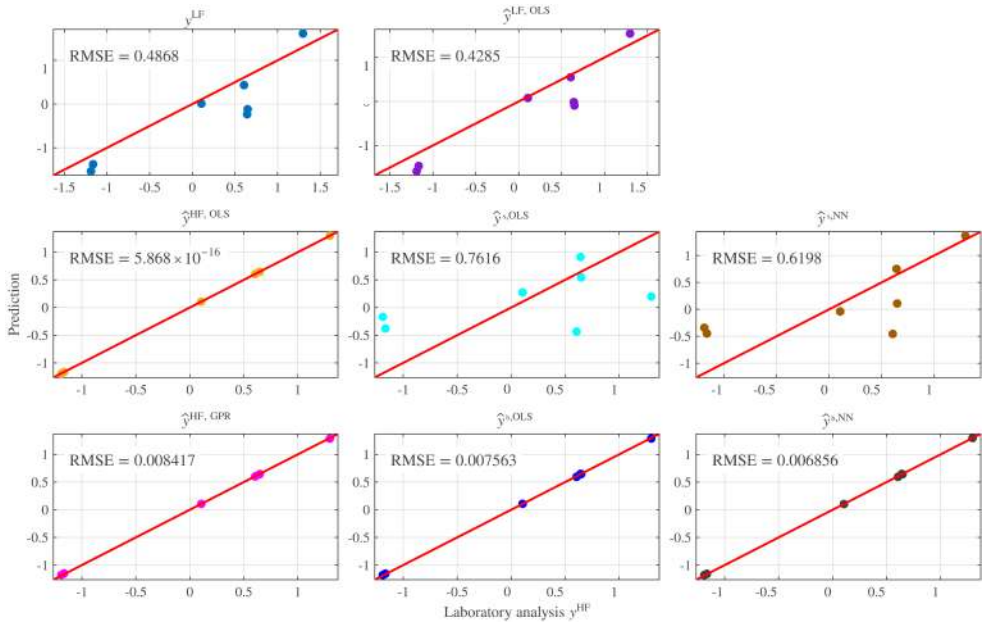


Figure 7.7: Parity plots on the TR subset. Predictions are evaluated against y_j^{HF} at the paired HF indices.

The detected labels y_i^{det} are compared with y_i^{pGT} using sensitivity, specificity, and accuracy.

Figures 7.7 and 7.8 show the parity plots for the TR and TS subsets. The comparison is made only at the paired HF indices, because $\mathbf{y}^{\mathrm{HF}}$ is the trusted reference. The red diagonal represents perfect agreement with the laboratory values.

The online analyzer baseline y_i^{LF} gives RMSE equal to 0.4868 on TR and 0.5165 on TS. This confirms that the analyzer does not consistently follow the laboratory reference. The plant-data OLS predictor $\hat{y}_i^{\mathrm{LF,OLS}}$ reduces the error to 0.4285 on TR and 0.3845 on TS. This shows that the selected DK inputs contain useful information about the monitored output, but the plant-only predictor still does not fully reproduce the laboratory trend. The HF-only predictors reach almost zero training RMSE because only 7 HF samples are available in TR. This does not indicate a robust model. It mainly shows that $\hat{y}_i^{\mathrm{HF,OLS}}$ and $\hat{y}_i^{\mathrm{HF,GPR}}$ can interpolate the small HF training set. On TS, the HF GPR model gives RMSE equal to 0.2928, while the HF OLS model gives RMSE equal to 0.3382. Both improve over the online analyzer, but their training behaviour shows that the small number of HF samples makes them

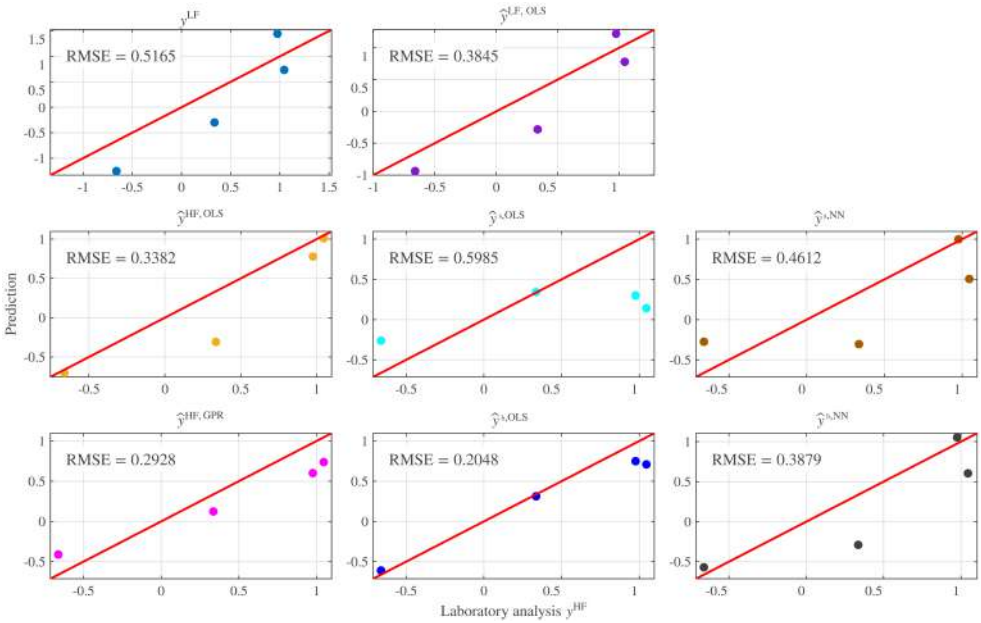


Figure 7.8: Parity plots on the TS subset. Predictions are evaluated against y_j^{HF} at the paired HF indices.

Table 7.2: Prediction RMSE values evaluated at paired HF indices in the simulation-supported anomaly-detection case study.

Model ξ	$\mathbf{RMSE}_{\mathrm{TR}}(\xi)$	$\mathbf{RMSE}_{\mathrm{TS}}(\xi)$
y^{LF}	0.4868	0.5165
$\hat{y}^{\mathrm{LF,OLS}}$	0.4285	0.3845
$\hat{y}^{\mathrm{HF,OLS}}$	≈ 0	0.3382
$\hat{y}^{\mathrm{HF,GPR}}$	≈ 0	0.2928
$\hat{y}^{\mathrm{s,OLS}}$	0.7616	0.5985
$\hat{y}^{\mathrm{s,NN}}$	0.6198	0.4612
$\hat{y}^{\mathrm{b,OLS}}$	≈ 0	0.2048
$\hat{y}^{\mathrm{b,NN}}$	≈ 0	0.3879

sensitive to overfitting. The simulation-only predictors show a different behaviour. The OLS simulation surrogate $\hat{y}_i^{\mathrm{s,OLS}}$ gives RMSE values 0.7616 on TR and 0.5985 on TS. The NN simulation surrogate $\hat{y}_i^{\mathrm{s,NN}}$ gives RMSE values 0.6198 on TR and 0.4612 on TS. These errors are higher than the corrected models, which indicates a systematic simulation-to-plant mismatch. The simplified simulator captures the main mass-balance trend, but it does not include all plant disturbances, dynamic effects, and unmodelled operating variability. The bias-corrected simulation surrogate gives the best testing result. The corrected OLS surrogate $\hat{y}_i^{\mathrm{b,OLS}}$ reduces the TS RMSE to 0.2048, which is the lowest value among the evaluated predictors. The corrected NN surrogate $\hat{y}_i^{\mathrm{b,NN}}$ reaches TS RMSE equal to 0.3879, which does not improve over the corrected OLS variant. This indicates that the additional nonlinear flexibility of the NN surrogate does not transfer reliably from the simplified simulation domain to the industrial plant domain.

Table 7.2 summarizes the reported RMSE values. Since the HF subset contains only 4 TS points, the exact numerical ranking should be interpreted carefully. However, the main pattern is clear: simulation-only models are biased, HF-only models are sensitive to sparse data, and the bias-corrected OLS simulation surrogate gives the strongest testing agreement with the HF reference.

Based on the TS prediction results and the need to compare detector behaviour, five predictors are selected for anomaly detection:

$$\left\{\hat{y}_i^{\mathrm{LF,OLS}},\ \hat{y}_i^{\mathrm{HF,OLS}},\ \hat{y}_i^{\mathrm{HF,GPR}},\ \hat{y}_i^{\mathrm{s,OLS}},\ \hat{y}_i^{\mathrm{b,OLS}}\right\}. \tag{7.19}$$

The NN-based simulation variants are not kept for the final detector comparison because they do not improve the testing prediction error relative to the OLS corrected

Table 7.3: Confusion-matrix metrics for anomaly detection on TR and TS subsets. Sensitivity and specificity are weighted equally in the threshold optimisation, with $a = b = 1$.

			TR			TS		
Detector	δ^*	$W(\delta^*)$	**Sens**	**Spec**	**Acc**	**Sens**	**Spec**	**Acc**
$\hat{y}^{\mathrm{LF,OLS}}$	0.137	1.114	0.5445	0.5691	0.5627	0.604	0.474	0.5189
$\hat{y}^{\mathrm{HF,OLS}}$	0.489	1.453	0.6739	0.7793	0.7517	0.621	0.679	0.6591
$\hat{y}^{\mathrm{HF,GPR}}$	0.447	1.392	0.8520	0.5401	0.6218	0.728	0.564	0.6205
$\hat{y}^{\mathrm{s,OLS}}$	0.555	1.206	0.7966	0.4095	0.5109	0.715	0.567	0.6180
$\hat{y}^{\mathrm{b,OLS}}$	0.502	1.428	0.7714	0.6571	0.6870	0.730	0.722	0.7248

surrogate.

After pseudo ground-truth construction, each selected predictor is converted into a detector. Table 7.3 reports the confusion-matrix metrics. The plant-data predictor $\hat{y}_i^{\mathrm{LF,OLS}}$ gives weak testing performance, with sensitivity 0.604, specificity 0.474, and accuracy 0.5189. This means that its residuals do not separate the pseudo-GT anomalies from normal samples well. The HF OLS predictor improves the balance, with testing sensitivity 0.621, specificity 0.679, and accuracy 0.6591. The HF GPR predictor increases testing sensitivity to 0.728, but specificity drops to 0.564, which means that it detects more anomalous samples but also raises more false alarms.

The simulation-only OLS surrogate $\hat{y}_i^{\mathrm{s,OLS}}$ gives testing sensitivity 0.715, specificity 0.567, and accuracy 0.6180. This shows that the simulation trend contains useful structure for detecting deviations, but its systematic bias limits the final detector quality. The best balance is obtained by the bias-corrected OLS simulation surrogate $\hat{y}_i^{\mathrm{b,OLS}}$. It reaches testing sensitivity 0.730, specificity 0.722, and accuracy 0.7248. This is the most balanced result among the compared detectors.

The TR results show that a high training score does not guarantee the best testing detector. The HF GPR detector reaches high TR sensitivity, equal to 0.8520, but its TR specificity is only 0.5401. This indicates that the model raises many alarms and does not separate normal and anomalous samples cleanly. The HF OLS detector gives the highest TR value of $W(\delta^*)$, but the testing results show that the bias-corrected simulation surrogate transfers better to TS. The result for $\hat{y}_i^{\mathrm{b,OLS}}$ is important because it shows that the simplified simulator is useful only after alignment with the plant. The onli simulation surrogate provides a physically meaningful trend, but the GP correction is needed to remove the simulation-to-plant bias. Once this correction is

included, the detector becomes more balanced and gives the best testing accuracy.

The fixed-threshold metrics in Table 7.3 describe detector behaviour at the optimised threshold δ^*. To evaluate detector behaviour independently of one threshold, ROC curves are also used. Figure 7.9 shows the ROC curves on TR, and Figure 7.10 shows the ROC curves on TS. The ROC curve is obtained by varying δ and computing the corresponding true positive rate and false positive rate.

On TR, the HF OLS detector performs strongly because it is trained and tuned close to the same sparse HF information used to construct the reference behaviour. The bias-corrected simulation surrogate remains close to this performance, but it does not rely only on interpolation of the HF samples. This difference becomes more important on TS, where the optimised threshold is transferred from TR. The marked operating points on TS do not always coincide with the best possible points on the ROC curves, because the residual distributions differ between TR and TS.

The TS ROC result confirms the same conclusion as the confusion-matrix metrics. The bias-corrected simulation surrogate $\hat{y}_i^{\mathrm{b,OLS}}$ maintains the most stable trade-off between sensitivity and false alarms. It reaches the highest testing AUC, equal to 0.750. This confirms that the corrected simulation-supported predictor improves not only pointwise prediction at HF timestamps, but also the ranking of normal and anomalous samples over a range of thresholds.

7.3 Discussion

The results show that simulation-supported monitoring is useful only when each data source keeps its correct role. The simulation data $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$ provide a physically consistent trend from the simplified process model, but this trend is shifted from the real plant because the simulator does not include all disturbances, dynamic effects, and plant-specific behaviour. The LF plant data $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$ describe the real operating trajectory, but the online analyzer may contain drift, bias, and local faults. The HF laboratory values $\mathbf{y}^{\mathrm{HF}}$ provide the trusted reference, but their sparse sampling does not support a reliable detector by itself. The best result is obtained when the simulation trend is corrected toward the laboratory reference. The OLS simulation surrogate $\hat{y}_i^{\mathrm{s,OLS}}$ captures the dominant mass-balance relation from the simplified model, but its direct plant prediction remains biased. The GP residual correction removes part of this simulation-to-plant mismatch by learning the difference between $\hat{y}_{i(j)}^{\mathrm{s,OLS}}$ and y_j^{HF} at the paired HF indices. The corrected predictor $\hat{y}_i^{\mathrm{b,OLS}}$ therefore combines the

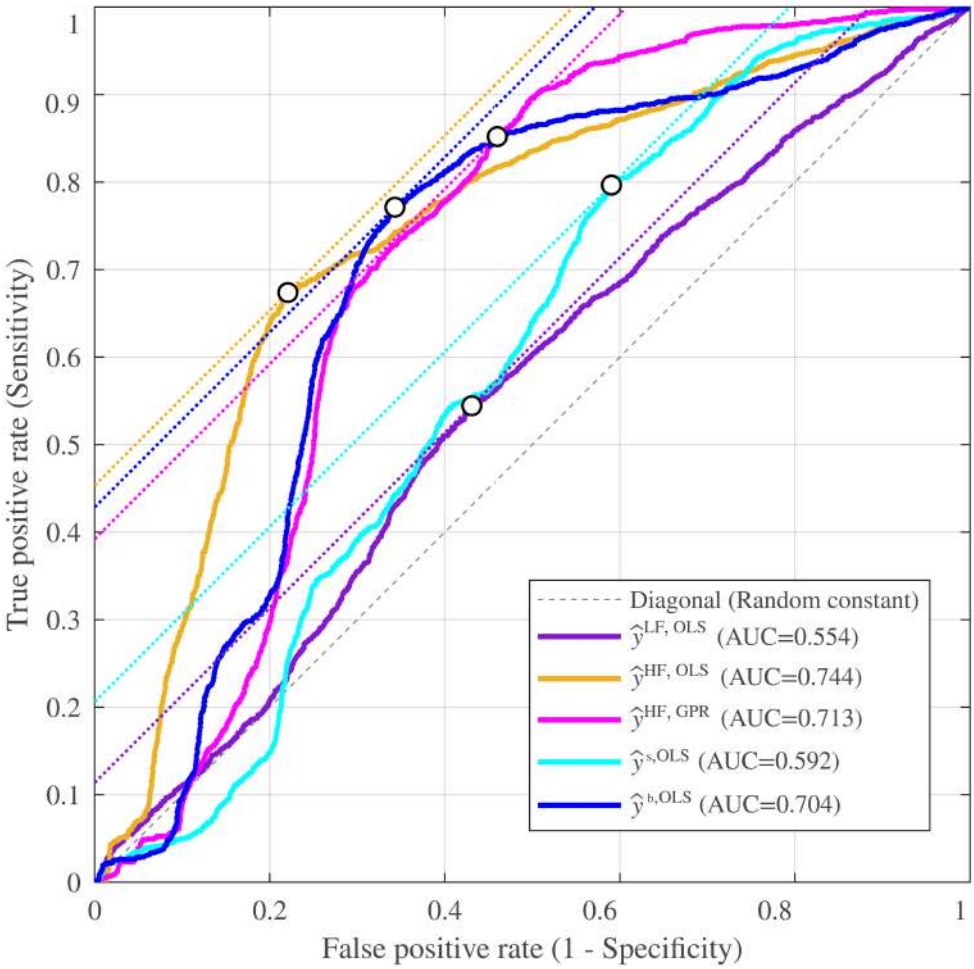


Figure 7.9: ROC curves on the TR subset for the selected anomaly detectors. The marked operating points correspond to the thresholds δ^* optimised on TR.

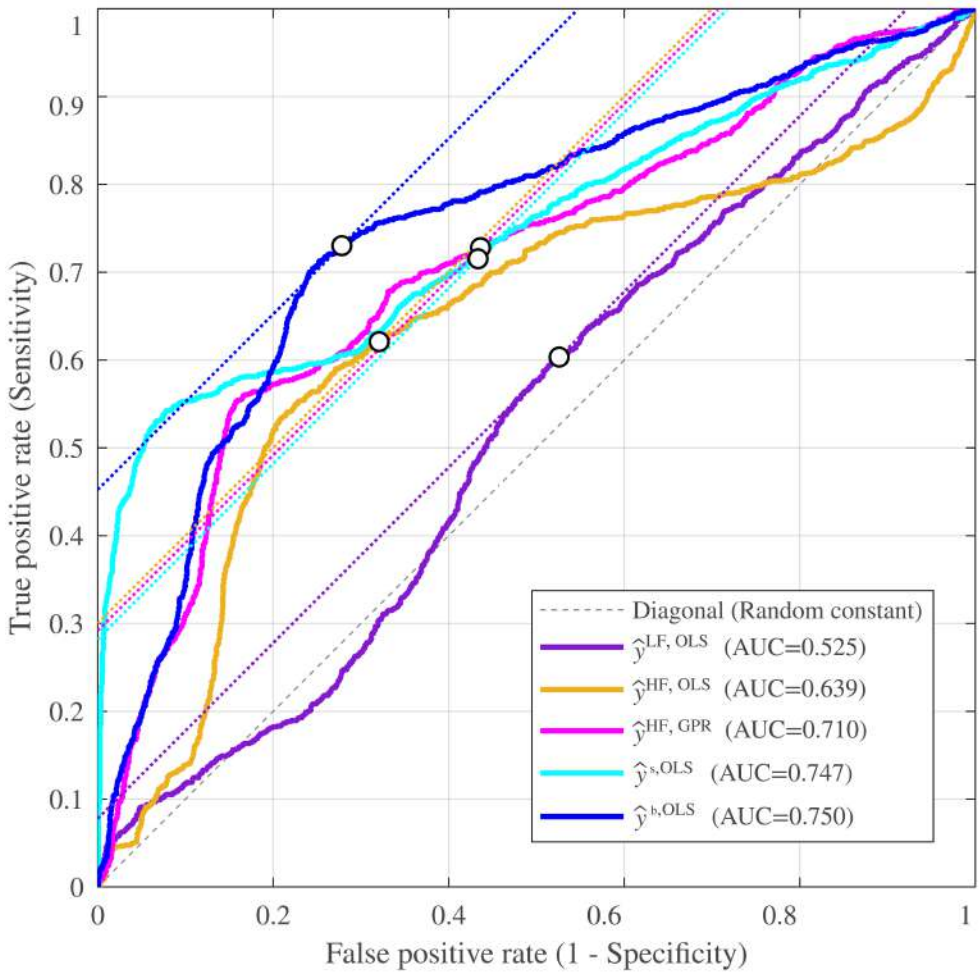


Figure 7.10: ROC curves on the TS subset for the selected anomaly detectors. The thresholds δ^* are transferred from TR without further tuning.

physical structure of the simulator with the reliability of the laboratory samples. This gives the best testing prediction result, reducing the RMSE at HF timestamps from 0.5165 for the online analyzer to 0.2048. The same corrected predictor also gives the most stable anomaly detector. After threshold optimisation on TR and direct application on TS, the detector based on $\hat{y}_i^{\mathrm{b,OLS}}$ reaches sensitivity 0.730, specificity 0.722, accuracy 0.7248, and testing AUC 0.750. This result is more informative than the prediction RMSE alone, because it shows that the corrected simulation-supported trajectory also separates normal and anomalous samples better on the plant grid. In contrast, HF-only models can fit the few laboratory points well but remain sensitive to overfitting, while the simulation surrogate contains useful physical information but still carries the simulator bias.

The comparison also shows that additional model flexibility is not automatically helpful. The NN simulation surrogate does not improve the final result, even though it can represent nonlinear relations. In this case, the nonlinear model can learn patterns specific to the simplified steady-state simulation data, which do not transfer reliably to the industrial plant. The OLS surrogate gives a smoother and more transferable baseline, so the following GP correction has a simpler residual to learn. The fixed-threshold evaluation is important for interpreting the detector results. Some models can reach favourable ROC regions for certain TS thresholds, but this does not mean that they would work reliably in deployment. The threshold δ^* must be selected on TR and then applied unchanged to TS. Under this condition, $\hat{y}_i^{\mathrm{b,OLS}}$ gives the best trade-off between missed anomalies and false alarms. The main limitation is that the anomaly labels y_i^{pGT} are pseudo labels derived from the proposed MF reference trajectory, not manually confirmed plant fault records. This is a practical solution for an industrial dataset without complete fault annotation, but the labels should be interpreted as a reproducible evaluation reference rather than absolute GT. A second limitation is the simplified steady-state simulator. It captures dominant mass-balance relations, but it cannot describe transient effects, catalyst changes, or all operating-mode transitions. Even with these limitations, the results show that a simplified simulation model can support anomaly detection when its systematic mismatch to the plant is corrected using sparse laboratory measurements.

Chapter 8

Conclusions and Future Research

This thesis developed a multi-fidelity framework for industrial process monitoring. The proposed framework combines these information sources through residual correction. Dense LF plant data $\mathbf{X}^{\mathrm{LF}}$ and $\mathbf{y}^{\mathrm{LF}}$ describe the process on the full plant time grid. Sparse HF laboratory values $\mathbf{y}^{\mathrm{HF}}$ define the trusted correction target. When available, simulation data $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$ provide an additional LF source based on simplified physical structure. The main idea is to first construct a LF prediction that captures the dominant process trend, and then learn the remaining difference to the HF laboratory reference using a GP residual model.

The early contributions focused on industrial data preprocessing, classification of online process data, and anomaly detection in petrochemical applications [46, 47, 48, 49]. The later work moved toward regression-based soft sensing, refinery monitoring, dynamic MF modelling, and soft-sensor-enhanced monitoring of the alkylation unit [50, 51, 52, 53]. These studies provided the basis for the final workflow used in this thesis, especially DK-guided feature selection, LF baseline modelling, laboratory-based GP correction, and validation on industrial data. The data-based soft-sensing part follows from the journal article on online sensor correction [31], while the simulation-supported soft-sensing and anomaly-detection part builds on the work on simplified simulation models and physics-guided monitoring of the alkylation unit [54, 55, 56].

The first implementation part applied the framework to data-based MF soft sensing for online sensor correction. The main correction structure was predictor correction, where the LF prediction $\hat{y}_i^{\mathrm{LF}}$ was used as an additional input in the MF regressor. This structure was more reliable than direct correction of the online analyzer signal because the LF predictor first captured the dense process trend and reduced the influence of local analyzer fluctuations. The GP correction then learned only the remaining residual to the laboratory reference. In the Tennessee Eastman Process benchmark, the predictor-correction model reduced the testing RMSE from 0.9770 for the LF analyzer to 0.4827, which corresponds to a 50.60 % improvement. This result was

obtained in a controlled setting where the pseudo-HF reference was constructed by smoothing and downsampling the LF signal. In the industrial alkylation unit, the same structure reduced the testing RMSE from 0.7945 to 0.6071, corresponding to a 23.59 % improvement. This result is more relevant for industrial use.

The second implementation part extended the framework to simulation-supported soft sensing and anomaly detection. A simplified steady-state gPROMS model generated $\mathbf{X}^{\mathrm{LF_s}}$ and $\mathbf{y}^{\mathrm{LF_s}}$ in the same six-variable DK feature space as the plant data. The simulator was not used as an exact plant model. Its role was to provide a physically consistent LF trend based on the main material-flow structure of the alkylation unit. After GP residual correction with sparse HF laboratory samples, the corrected OLS simulation surrogate $\hat{y}_i^{\mathrm{b,OLS}}$ reduced the RMSE at HF timestamps from 0.5165 for the online analyzer to 0.2048. The corrected simulation-supported predictor also gave the best anomaly-detection result. The pseudo ground-truth labels y_i^{pGT} were constructed from the MF reference trajectory and a tolerance based on the typical LF–HF disagreement. Each candidate predictor was converted into a detector by comparing its deviation from the online analyzer with a threshold optimised only on the TR subset. The detector based on $\hat{y}_i^{\mathrm{b,OLS}}$ achieved the most balanced TS performance, with sensitivity 0.730, specificity 0.722, accuracy 0.7248, and testing AUC 0.750.

Across both implementation chapters, the same conclusion appears. More flexible models did not automatically perform better. Nonlinear single-fidelity models did not consistently improve validation error, direct MF regression showed signs of overfitting, and the NN simulation surrogate did not transfer reliably from simplified simulation data to the real plant. The most useful models were simpler and more constrained. DK feature selection kept the input space physically meaningful, LF predictors supplied dense trends, and GP residual models corrected these trends toward sparse HF references. The main contribution of the thesis is therefore a deployment-oriented MF monitoring workflow for industrial datasets with dense LF measurements and sparse HF references. The workflow improves online sensor correction, supports anomaly detection, and remains interpretable enough for industrial use. It does not replace laboratory analysis, process expertise, or simulation models. Instead, it gives each source a clear role: LF plant data describe the process timeline, HF laboratory data define the trusted correction target, and simulation data add physical structure when available.

Future research should first focus on online updating. In the current implementation, the GP residual model is trained offline. In plant operation, new laboratory measurements become available over time, so the correction model should be updated when new y_j^{HF} values arrive. Future work should decide whether a new laboratory sample should

update only the residual model, or whether it should also trigger feature re-selection, LF model re-identification, or GP kernel retuning. A second direction is uncertainty-aware monitoring. GPR provides predictive uncertainty, but this uncertainty was not fully used in the final detector. Future work could use it to construct adaptive tolerance bands around $\hat{y}_i^{\mathrm{MF}}$ or $\hat{y}_i^{\mathrm{b,OLS}}$. This would allow the detector to be more conservative in poorly supported operating regions and more sensitive where the model is well supported by historical data. A third direction is independent validation of the pseudo ground-truth labels. In this thesis, y_i^{pGT} was derived from the MF reference trajectory because manually confirmed fault labels were not available. This is practical, but the labels should be compared with operator logs, maintenance records, analyzer calibration reports, and manually reviewed abnormal intervals. A fourth direction is the use of dynamic simulation-supported predictors. The simulator used here was steady-state and therefore could not describe transients, residence-time effects, catalyst changes, or operating-mode transitions. A dynamic simulator could provide a better LF trend in processes where delays and recycle dynamics strongly affect the monitored output, but its benefit should be tested against the simpler steady-state approach.

Finally, the framework should be tested on additional units and extended toward multiple monitored outputs. Many industrial plants contain several sparse laboratory variables and several unreliable analyzers. A multi-output MF framework could use shared DK feature spaces, coupled residual models, or shared simulation structures to monitor several related quality variables at once. The corrected MF soft sensor could also be evaluated as a signal for advanced process control and real-time optimisation, where latency, robustness, and closed-loop behaviour become important. In conclusion, the thesis shows that MF residual correction can connect plant data, laboratory measurements, and simplified simulation models in a practical monitoring framework. The approach improves soft sensing, supports anomaly detection, and keeps the modelling structure understandable. The next step is to move from offline validation toward adaptive and operationally validated monitoring systems that can support real-time industrial decision making.

Appendix A

Curriculum Vitae

Ing. Rastislav Fáber
Date of Birth: 25/02/1997
Citizenship: Slovakia
E-mail: rastislav.faber97@gmail.com
LinkedIn: linkedin.com/in/rastislav-fáber-9a0a52221

Education

- **Doctoral Studies** September 2022 – Present
  - Slovak University of Technology in Bratislava
  - Faculty of Chemical and Food Technology
  - Institute of Information Engineering, Automation, and Mathematics
  - Programme: Process Control
  - Research focus: multi-fidelity modelling, process monitoring, soft sensing, and anomaly detection in industrial processes
- **Master's Degree** completed May 2022
  - Slovak University of Technology in Bratislava
  - Faculty of Chemical and Food Technology
  - Institute of Information Engineering, Automation, and Mathematics
  - Programme: Automation and Control Engineering in Chemistry and Food Industry
  - Graduated with red diploma
- **Bachelor's Degree** completed July 2020
  - Slovak University of Technology in Bratislava
  - Faculty of Chemical and Food Technology
  - Programme: Biotechnology
- **Secondary Education** completed June 2016
  - Gymnázium Ľudovíta Štúra, Zvolen
  - General education

Foreign Research Stays

- **Research Stay at Università di Pisa, Department of Civil and Industrial Engineering, Italy** [February-May/2024]
  - Duration: 4 months
  - Research focus: multi-fidelity modeling for industrial soft sensing and process monitoring
  - Main activities: preparation of multi-fidelity residual-correction models, discussion of modeling results, and preparation of joint scientific outputs

- **Research Stay at Università di Pisa, Department of Civil and Industrial Engineering, Italy** [February–March/2026]
  - Duration: 2 months
  - Research focus: development and validation of simplified process models for industrial multi-fidelity modeling and process optimization
  - Main activities: work on simulation-based modeling of an industrial alkylation unit, generation of physically consistent low-fidelity simulation data, and integration of simulation data with industrial plant and laboratory data

Research Fields

- Multi-fidelity modeling
- Soft sensing and online sensor correction
- Process monitoring and anomaly detection
- Machine learning for process systems engineering
- Industrial data analysis
- Hybrid data-driven and simulation-based modeling
- Process optimization
- Chemical and petrochemical process applications

Digital Skills

- Python (advanced)
- Matlab and Simulink (advanced)
- AVEVA Process Simulation (beginner)
- Honeywell UniSim Design (beginner)
- gPROMS ProcessBuilder (beginner)
- MySQL (intermediate)
- PHP (advanced)
- JavaScript, TypeScript, Vue.js, Nuxt.js (advanced)
- HTML, CSS, Git (advanced)
- React Native (beginner)
- LaTeX (advanced)
- MS Office (intermediate)

Language Skills

- Slovak: mother tongue
- English: upper-intermediate level

Awards

- **Diploma from the Dean of the Faculty** 2021
  - Excellent fulfilment of study obligations
- **Commendation from the Dean of the Faculty** 2022

  - Excellent fulfilment of study obligations during the entire study programme

- **Commendation from the Rector** 2022
  - Excellent fulfilment of study obligations during the entire study programme

Conference Presentations and Research Dissemination

- **International venues:**
  - Sheffield, United Kingdom
  - Busan, Republic of Korea
  - Visegrád, Hungary
  - Florence, Italy
  - Ghent, Belgium
  - Toronto, Canada
- **National venues:**
  - Bratislava, Slovakia
  - Štrbské Pleso, High Tatras, Slovakia
  - Tatranské Matliare, High Tatras, Slovakia
  - Jasná, Low Tatras, Slovakia
- **Conference series:**
  - ESCAPE – European Symposium on Computer Aided Process Engineering
  - EUCA – European Control Conference (ECC)
  - IFAC DYCOPS – Symposium on Dynamics and Control of Process Systems
  - IFAC ADCHEM – Symposium on Advanced Control of Chemical Processes
  - IFAC World Congress
  - IEEE Cybernetics & Informatics (K&I)
  - IEEE International Conference on Process Control (PC)
  - SSCHE – International Conference of the Slovak Society of Chemical Engineering

Appendix B

Author Publications

Journal articles

1. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Data-based multi-fidelity modeling for online sensors correction*. Computers & Chemical Engineering, Vol. 211, Article 109665, 2026.

2. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Physics-Guided Soft Sensing for Anomaly Detection: Simulation Surrogates in Industrial Processes*. Chemical Engineering Research and Design (ChERD), (submitted).

3. **M. Wadinger, R. Fáber, E. Pavlovičová, and R. Paulen**: *Carbon neutral greenhouse: Economic model predictive control framework for education*. European Journal of Control, Vol. 83, Article 101297, 2025.

Conference contributions

1. **R. Fáber, R. Valo, M. Roman, and R. Paulen**: *Towards Temperature Monitoring in Long-Term Grain Storage*. In 2022 Cybernetics & Informatics (K&I), pp. 1–6, 2022.

2. **R. Fáber, K. Ľubušký, M. Mojto, and R. Paulen**: *Enhancing Industrial Data Analysis through Machine Learning-based Classification of Petrochemical Datasets*. In 49th International Conference of the Slovak Society of Chemical Engineering SSCHE 2023, Slovak Society of Chemical Engineering, Bratislava, Slovakia, p. 160, 2023.

3. **R. Fáber, K. Ľubušký, and R. Paulen**: *Machine Learning-based Classification of Online Industrial Datasets*. In R. Paulen and M. Fikar (Eds.), Proceedings of the 2023 24th International Conference on Process Control, IEEE, Slovak University of Technology in Bratislava, pp. 132–137, 2023.

4. **R. Fáber, M. Mojto, K. Ľubušký, and R. Paulen**: *From Data to Alarms: Data-driven Anomaly Detection Techniques in Industrial Settings*. In Flavio Manenti and G. V. Rex Reklaitis (Eds.), 34th European Symposium on Computer Aided Process Engineering / 15th International Symposium on Process Systems Engineering, Vol. 53, pp. 2857–2862, 2024.

5. **R. Fáber, M. Mojto, K. Ľubušký, and R. Paulen**: *Integrated Data Analytics and Regression Techniques for Real-time Anomaly Detection in Industrial Processes*. In 12th IFAC Symposium on Advanced Control of Chemical Processes, pp. 320–325, 2024.

6. **R. Fáber, M. Klaučo, K. Ľubušký, and R. Paulen**: *Integrating Linear and Nonlinear Models for Enhanced Process Monitoring*. In R. Paulen, M. Fikar, and J. Oravec (Eds.), Proceedings of the 2025 25th International Conference on Process Control, IEEE, pp. 1–6, 2025.

7. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Multi-fidelity Modeling for Process Optimization in Refinery Operations*. In 51st International Conference of the Slovak Society of Chemical Engineering SSCHE 2025, Slovak Society of Chemical Engineering, Bratislava, Slovakia, p. 140, 2025.

8. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Improving Process Monitoring via Dynamic Multi-Fidelity Modeling*. In 14th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems, Elsevier, pp. 199–204, 2025.

9. **R. Fáber, M. Vaccari, R. Bacci di Capaci, K. Ľubušký, G. Pannocchia, and R. Paulen**: *Soft-Sensor-Enhanced Monitoring of an Alkylation Unit via Multi-Fidelity Model Correction*. Systems and Control Transactions, Vol. 4, pp. 1389–1395, 2025.

10. **R. Fáber, K. Ľubušký, and R. Paulen**: *A Simplified Framework for Process Model Development in Industrial Simulation Environments*. In 52nd International Conference of the Slovak Society of Chemical Engineering SSCHE 2026, Slovak Society of Chemical Engineering. Bratislava, Slovakia, 2026. (accepted)

11. **R. Fáber, K. Ľubušký, and R. Paulen**: *A Hybrid Data-Driven Approach for the Optimization of an Industrial Alkylation Unit*. In *ESCAPE 36 – European Symposium on Computer Aided Process Engineering*, Sheffield, UK. June 21–24, 2026, *Syst. Control Trans.*. (accepted)

12. **R. Fáber, M. Vaccari, R. Bacci di Capaci, G. Pannocchia, and R. Paulen**: *A Multi-Fidelity Residual-Correction Framework for Industrial Soft Sensing*. In 24th European Control Conference (ECC) in Reykjavík, Iceland. July 07–10, 2026. (accepted)

13. **P. Arbetová, R. Fáber, K. Ľubušký, and R. Paulen**: *A Practical Framework for Process Anomaly Detection Analysis in Multivariate Time Series*. In 23rd IFAC World Congress. August 23–28, 2026. Busan, Republic of Korea. (accepted)

Appendix C

Resumé

Predložená dizertačná práca sa venuje vývoju a overeniu tzv. multi-fidelity modelov pre pokročilé monitorovanie priemyselných procesov. Práca vychádza z praktického problému, ktorý sa často objavuje v chemickom a petrochemickom priemysle. Moderné prevádzky zbierajú veľké množstvo online dát, no najdôležitejšie kvalitatívne veličiny nie sú vždy merané dostatočne presne, spoľahlivo alebo nepretržite. Online analyzátory poskytujú rýchlu informáciu o aktuálnom stave procesu, ale ich merania môžu byť ovplyvnené šumom, posunom, systematickou odchýlkou, kalibračnými chybami alebo krátkodobými poruchami merania. Laboratórne analýzy sú naopak presnejšie a z pohľadu prevádzky dôveryhodnejšie, ale sú dostupné iba zriedkavo, s oneskorením a za vyššiu cenu. Vzniká tak rozpor medzi tým, čo je dostupné často a v reálnom čase, a tým, čo je presné, ale dostupné len občas. Tento rozpor výrazne obmedzuje použitie klasických soft-senzorov. Model trénovaný iba na online dátach môže zachytiť časový priebeh procesu, ale zároveň môže prevziať aj chyby samotného merania. Ak je online analyzátor zle nakalibrovaný alebo krátkodobo poskytuje chybné hodnoty, model sa môže naučiť práve túto chybnú trajektóriu. Model trénovaný iba na laboratórnych dátach je síce založený na spoľahlivejšej referencii, ale zvyčajne má k dispozícii príliš málo vzoriek na to, aby vedel opísať celý vývoj procesu v čase. Základná myšlienka práce preto spočíva v tom, že tieto dátové zdroje netreba posudzovať oddelene. Každý z nich nesie inú časť informácie o tom istom procese a ich vhodné spojenie môže poskytnúť lepší odhad sledovanej veličiny než ktorýkoľvek zdroj samostatne.

V práci sa táto situácia opisuje ako multi-fidelity problém, teda problém, v ktorom sú dostupné dáta s rôznou frekvenciou, spoľahlivosťou a fyzikálnym významom. Prvým zdrojom sú LF dáta, teda online procesné merania dostupné na hustej časovej mriežke. Ich hlavnou výhodou je vysoká časová frekvencia a priame napojenie na prevádzku, no ich nevýhodou je nižšia spoľahlivosť. Druhým zdrojom sú HF dáta, teda laboratórne merania, ktoré predstavujú dôveryhodnú referenciu pre kvalitatívne veličiny, ale sú riedke a oneskorené. Tretím zdrojom, ak je dostupný, sú simulačné dáta zo zjednodušeného fyzikálneho modelu. Takýto model nemusí presne opisovať celú priemy-

selnú jednotku, ale môže zachytiť základný fyzikálny trend vyplývajúci z materiálových vzťahov a štruktúry procesu. Navrhnuté riešenie je založené na reziduálnej korekcii. Najskôr sa vytvorí LF prediktor, ktorý zachytí hlavný trend procesu na hustej časovej mriežke. Potom sa v bodoch, kde sú dostupné laboratórne merania, vypočíta rozdiel medzi touto predikciou a laboratórnou hodnotou. Tento rozdiel sa modeluje pomocou separátneho predikčného modelu a následne sa používa ako korekcia LF predikcie. Výsledkom je odhad, ktorý si zachováva frekvenciu online meraní, ale zároveň sa približuje k laboratórnej referencii. Cieľom práce nie je nahradiť laboratórne merania ani procesnú expertízu. Cieľom je vytvoriť praktický spôsob, ako tieto zdroje spojiť tak, aby bol výsledný model použiteľný aj v priemyselnej prevádzke. Preto sa práca nezameriava iba na zníženie predikčnej chyby, ale aj na zrozumiteľnosť modelu, dostupnosť použitých meraní, fyzikálny význam vstupných premenných a možnosť dlhodobej údržby. Z tohto dôvodu sa v práci spája štatistický výber premenných s doménovou znalosťou procesu. Relevantné premenné sa nevyberajú iba podľa toho, či znižujú validačnú chybu, ale aj podľa toho, či dávajú procesne zmysel, či sú merania spoľahlivé a či sú vhodné pre dlhodobé použitie v prevádzke. V práci sa porovnávajú viaceré metódy výberu premenných, napríklad PCA, PLS, LASSO a stepwise regresia. Ich výsledky sa následne posudzujú z pohľadu fyzikálneho významu a prevádzkovej použiteľnosti. Takto sa vytvorí menšia množina vstupov, ktorá je jednoduchšia na interpretáciu a zároveň znižuje riziko pretrénovania. Tento krok je dôležitý najmä pri priemyselných dátach, kde veľký počet meraní nemusí automaticky znamenať lepší model. Niektoré signály môžu byť síce štatisticky zaujímavé, ale v praxi môžu byť nestabilné, zle dostupné, závislé od konkrétneho nastavenia regulácie alebo nevhodné pre dlhodobé nasadenie. Preto má doménová znalosť v práci rovnakú dôležitosť ako samotné štatistické hodnotenie.

Metodická časť práce postupne opisuje štruktúru priemyselných dát, časový nesúlad medzi online a laboratórnymi meraniami, formuláciu predikčného problému a rozšírenie smerom k detekcii anomálií. Osobitná pozornosť sa venuje párovaniu laboratórnych vzoriek s online meraniami. Laboratórne hodnoty totiž nie sú dostupné na tej istej časovej mriežke ako online dáta, a preto treba každému laboratórnemu bodu priradiť zodpovedajúci čas v prevádzkových dátach alebo krátke časové okno. Tento krok má priamy vplyv na tréning korekčného modelu, finálne hodnotenie aj interpretáciu výsledkov. Ak je časové priradenie zvolené nevhodne, model sa môže učiť z nesprávne spárovaných hodnôt a výsledná korekcia nemusí odrážať skutočný vzťah medzi online analyzátorom a laboratórnou referenciou.

Navrhnutý postup sa overuje na dvoch procesných systémoch. Prvým je benchmark Tennessee Eastman Process, ktorý slúži ako kontrolovaný prípad na overenie metodiky. Keďže tento dataset neobsahuje skutočné laboratórne merania, pseudo-HF referencia

sa vytvorí zo signálu online analyzátora pomocou vyhladenia a riedkeho vzorkovania. Tento prípad umožňuje overiť multi-fidelity korekciu v prostredí so známou štruktúrou a dostupným úplným časovým priebehom. Druhým prípadom je reálna priemyselná alkylačná jednotka v rafinérii Slovnaft, a.s. Tento prípad je prakticky náročnejší, pretože obsahuje skutočné prevádzkové dáta, šum, zmeny režimu, malý počet laboratórnych vzoriek a online analyzátor, ktorý sa od laboratórnych hodnôt miestami výrazne odchyľuje. Prvá implementačná časť práce sa venuje dátovo založenému multi-fidelity soft-senzoru pre korekciu online senzora. V tejto časti sa porovnávajú modely založené iba na LF dátach, modely založené iba na HF dátach a modely, ktoré spájajú oba zdroje informácií. Najspoľahlivejšia bola štruktúra označená ako predictor correction. V nej sa najskôr vytvorí dynamický LF prediktor, ktorý zachytí hlavný trend procesu, a rozdiel medzi touto predikciou a laboratórnou hodnotou sa modeluje Gaussovou procesnou regresiou. Tento postup bol stabilnejší než priama korekcia online analyzátora, pretože LF prediktor čiastočne filtruje krátkodobé výkyvy a vytvára hladší základ pre reziduálnu korekciu. V prípade benchmarku Tennessee Eastman Process tento model znížil testovacie RMSE z hodnoty 0.9770 pre LF analyzátor na 0.4827, čo predstavuje zlepšenie o 50.60 %. V prípade priemyselnej alkylačnej jednotky sa testovacie RMSE znížilo z hodnoty 0.7945 na 0.6071, teda o 23.59 %. Tento výsledok je dôležitý najmä preto, že bol dosiahnutý na reálnych prevádzkových dátach s obmedzeným počtom laboratórnych meraní. Druhá implementačná časť rozširuje tento postup o využitie simulačných dát a o detekciu anomálií. Pre priemyselnú alkylačnú jednotku bol vytvorený zjednodušený stacionárny model v prostredí gPROMS. Tento model nebol použitý ako presný opis celej prevádzky. Jeho úlohou bolo generovať fyzikálne konzistentné simulačné dáta s rovnakými vstupnými veličinami ako v prvej časti. Simulačný model tak poskytuje ďalší LF zdroj, ktorý dopĺňa reálne merania o trend vychádzajúci z materiálovej štruktúry procesu. Na simulačných dátach boli trénované predikčné modely, ktoré sa potom vyhodnotili na reálnych priemyselných vstupoch. Ukázalo sa, že jednoduchší OLS model poskytoval trend, ktorý sa dal účinne korigovať laboratórnymi dátami. Po Gaussovskej reziduálnej korekcii sa RMSE na HF časových bodoch znížilo z hodnoty 0.5165 pre online analyzátor na 0.2048 pre korigovaný OLS model založený na simulácii. Tento výsledok ukazuje, že aj zjednodušený fyzikálny model môže byť užitočný, ak sa nepoužíva ako presný obraz reality, ale ako zdroj dodatočnej informácie, ktorá sa následne upraví voči laboratórnym hodnotám.

Výsledky zároveň ukázali, že zložitejší model nemusí automaticky priniesť lepšie riešenie. Neurónová sieť síce veľmi presne aproximovala simulačné dáta, ale jej prenos na reálne prevádzkové dáta bol slabší. Pravdepodobným dôvodom je rozdiel medzi simulátorom a skutočnou prevádzkou. Neurónová sieť sa mohla naučiť nelinearity typické pre simulátor, ktoré sa v reálnych dátach neprejavili rovnakým spôsobom. Naopak, jednoduchší lineárny model predikoval hladší procesný trend, ktorý sa dal lepšie korigovať pomocou

malého počtu HF vzoriek. Tento výsledok podporuje hlavnú myšlienku práce: v priemyselnom monitorovaní nie je vždy najvhodnejší najkomplexnejší model, ale model, ktorý je stabilný, zrozumiteľný a použiteľný v prevádzke. Rovnaký princíp sa následne použil aj na detekciu anomálií. Keďže v priemyselnom datasete neboli dostupné manuálne potvrdené anomálne intervaly, práca používa náhradné referenčné značky odvodené z MF predikcie. Táto predikcia predstavuje odhad laboratórne-kvalitného výstupu na celej časovej osi prevádzkových dát. Ak sa online analyzátor od tejto referencie odchýli viac, než povoľuje zvolený rozsah tolerancie, príslušná vzorka sa označí ako anomálna. Tento postup nenahrádza skutočné prevádzkové záznamy o poruchách, ale umožňuje systematicky porovnať rôzne detekčné metódy v situácii, keď manuálne anotácie nie sú dostupné. Detektory boli vyhodnocované pomocou citlivosti, špecificity, presnosti a ROC analýzy. Detekčný rozsah bol optimalizovaný iba na tréningovej množine a následne bez ďalších úprav prenesený na testovaciu množinu. Najvyváženejší výsledok dosiahol detektor založený na korigovanom OLS modeli zo simulačných dát. Na testovacej množine dosiahol citlivosť 0.730, špecificitu 0.722, presnosť 0.7248 a AUC 0.750. Z celkového pohľadu práca prináša niekoľko hlavných príspevkov. Práca ukazuje, ako možno v jednom postupe využiť husté online dáta, riedke laboratórne merania a simulačné dáta tak, aby mal každý zdroj jasnú úlohu. Online dáta poskytujú časový priebeh procesu, laboratórne merania určujú dôveryhodnú referenciu a simulačný model dodáva fyzikálne konzistentný trend. Práca taktiež ukazuje význam doménovo riadeného výberu premenných a jednoduchších modelových štruktúr pre priemyselné použitie, a overuje navrhnutý postup na benchmarku aj na reálnej alkylačnej jednotke. Výsledný MF model následne slúži v detekcii anomálií ako referenčná trajektória, ktorá aproximuje laboratórnu kvalitu výstupu na celej časovej osi.

Budúci výskum by sa mohol zamerať na nezávislé overenie anomálií. V tejto práci boli referenčné označenia anomálií odvodené z MF predikcie, pretože nebol dostupný kompletný súbor potvrdených porúch. Tento postup je praktický, ale nemožno ho považovať za úplnú náhradu skutočných prevádzkových záznamov. V práci bol použitý zjednodušený stacionárny simulátor, ktorý nezachytáva prechodové deje, časové oneskorenia, zmeny katalyzátora ani dynamiku recyklov. Napriek tomu poskytol užitočný trend pre multi-fidelity korekciu. Budúci výskum by mal preto overiť, či dynamický simulátor prinesie výrazné zlepšenie oproti jednoduchšiemu stacionárnemu modelu, alebo či by vyššia zložitosť znamenala najmä vyššie náklady na vývoj, kalibráciu a údržbu. Na záver možno konštatovať, že dizertačná práca ukazuje praktickú hodnotu multi-fidelity reziduálnej korekcie pre priemyselné monitorovanie. Navrhnutý postup umožňuje spojiť husté prevádzkové dáta, nepravidelné laboratórne merania a zjednodušené simulačné modely do jednej zrozumiteľnej modelovej štruktúry. Výsledky potvrdzujú, že takýto prístup dokáže zlepšiť online soft sensing, podporiť detekciu anomálií a zároveň zostať dostatočne jednoduchý pre priemyselné použitie.

Práca neprezentuje univerzálny black-box model pre všetky procesy, ale praktický a prenositeľný postup pre situácie, v ktorých tú istú kvalitatívnu veličinu opisujú viaceré dátové zdroje s rozdielnou frekvenciou, spoľahlivosťou a fyzikálnym významom.

Bibliography

[1] Shen Yin, Xiao Li, Huijun Gao, and Okyay Kaynak. Data-based techniques focused on modern industry: An overview. *IEEE Transactions on Industrial Electronics*, 62(1):657–667, 2015.

[2] Giovanni Volpe and Jan Wehr. Effective drifts in dynamical systems with multiplicative noise: a review of recent progress. *Reports on Progress in Physics*, 79(5):053901, 2016.

[3] Zhiqiang Ge, Zhihuan Song, and Furong Gao. Review of recent research on data-based process monitoring. *Industrial & Engineering Chemistry Research*, 52(10):3543–3562, mar 2013.

[4] Markus Hammer, Ken Somers, Hugo Karre, and Christian Ramsauer. Profit per hour as a target process control parameter for manufacturing systems enabled by big data analytics and industry 4.0 infrastructure. *Procedia CIRP*, 63:715–720, 2017. Manufacturing Systems 4.0 - Proceedings of the 50th CIRP Conference on Manufacturing Systems.

[5] R.K. Pearson. Exploring process data. *Journal of Process Control*, 11(2):179–194, 2001.

[6] John F. MacGregor and Ali Cinar. Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods. *Computers & Chemical Engineering*, 47:111–120, 2012.

[7] Shen Yin, Steven X. Ding, Xiaochen Xie, and Hao Luo. A review on basic data-driven approaches for industrial process monitoring. *IEEE Transactions on Industrial Electronics*, 61(11):6418–6428, 2014.

[8] Mohammed Alauddin, Faisal Khan, Syed Imtiaz, Salim Ahmed, and Paul Amyotte. Integrating process dynamics in data-driven models of chemical processing systems. *Process Safety and Environmental Protection*, 174:158–168, 2023.

[9] Dan Shen, Erik Blasch, Peter Zulch, Marcello Distasio, Ruixin Niu, Jingyang Lu, Zhonghai Wang, and Genshe Chen. A joint manifold leaning-based framework for heterogeneous upstream data fusion. *Journal of Algorithms & Computational Technology*, 12(4):311–332, 2018.

[10] Petr Kadlec, Bogdan Gabrys, and Sibylle Strandt. Data-driven soft sensors in the process industry. *Computers & Chemical Engineering*, 33(4):795–814, 2009.

[11] Luigi Fortuna, Salvatore Graziani, Alessandro Rizzo, and Maria Gabriella Xibilia. *Soft Sensors for Monitoring and Control of Industrial Processes*. Springer, London, 2007.

[12] Svante Wold, Arnold Ruhe, Herman Wold, and W. J. Dunn. The collinearity problem in linear regression. the partial least squares approach to generalized inverses. *SIAM Journal on Scientific and Statistical Computing*, 5(3):735–743, 1984.

[13] Paul Geladi and Bruce R. Kowalski. Partial least-squares regression: A tutorial. *Analytica Chimica Acta*, 185:1–17, 1986.

[14] Francesco Curreri, Luca Patanè, and Maria Gabriella Xibilia. RNN- and LSTM-based soft sensors transferability for an industrial process. *Sensors*, 21(3):823, 2021.

[15] Xiaofeng Yuan, Lin Li, Yuri A. W. Shardt, Yalin Wang, and Chunhua Yang. Deep learning with spatiotemporal attention-based LSTM for industrial soft sensor model development. *IEEE Transactions on Industrial Electronics*, 68(5):4404–4414, 2021.

[16] Kai Wang, Chang Shang, Liwei Liu, Yaochu Jiang, Deqing Huang, and Fuwen Yang. Dynamic soft sensor development based on convolutional neural networks. *Industrial & Engineering Chemistry Research*, 58(26):11521–11531, 2019.

[17] Liang Cao. *Interpretable and stable soft sensor modeling for industrial applications*. PhD thesis, University of British Columbia, 2024.

[18] Eugeniu Strelet, Ivan Castillo, You Peng, and Marco S. Reis. Data fusion: Integrating heterogeneous information sources in the chemical processing industry. *Journal of Chemometrics*, 39(11):e70075, 2025. e70075 CEM-24-0355.

[19] Y. Jiang, S. Yin, J. Dong, and O. Kaynak. A review on soft sensors for monitoring, control, and optimization of industrial processes. *IEEE Sensors Journal*, 21(11):12868–12881, 2021.

[20] P. H. Rangegowda, J. Valluru, S. C. Patwardhan, and S. Mukhopadhyay. Simultaneous and sequential state and parameter estimation using receding-horizon nonlinear kalman filter. *Journal of Process Control*, 109:13–31, 2022.

[21] C. C. Pantelides and J. G. Renfro. The online use of first-principles models in process operations: Review, current status and future needs. *Computers & Chemical Engineering*, 51:136–148, 2013.

[22] Johannes Lips, Hendrik Lens, and Paolo Conti. Soft sensor design using hierarchical multi-fidelity modeling with bayesian optimization for input variable selection. *IFAC-PapersOnLine*, 59(6):139–144, 2025. 14th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2025.

[23] Qi Zhou, Yuda Wu, Zhendong Guo, Jiexiang Hu, and Peng Jin. A generalized hierarchical co-Kriging model for multi-fidelity data fusion. *Structural and Multidisciplinary Optimization*, 62:1885–1904, 2020.

[24] Alexander I. J. Forrester, András Sóbester, and Andy J. Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3251–3269, 2007.

[25] Maziar Raissi and George Em Karniadakis. Deep multi-fidelity gaussian processes. *arXiv preprint arXiv:1604.07484*, 2016.

[26] Benjamin Peherstorfer, Karen Willcox, and Max Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Review*, 60(3):550–591, 2018.

[27] M. Giselle Fernández-Godino. Review of multi-fidelity models. *Advances in Computational Science and Engineering*, 1(4):351–400, 2023.

[28] S. Joe Qin. Statistical process monitoring: Basics and beyond. *Journal of Chemometrics*, 17(8–9):480–502, 2003.

[29] Peter J. Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[30] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A. Lozano. A review on outlier/anomaly detection in time series data. *ACM Computing Surveys*, 54(3):1–33, 2021.

[31] Rastislav Fáber, Marco Vaccari, Riccardo Bacci di Capaci, Karol Ľubušký, Gabriele Pannocchia, and Radoslav Paulen. Data-based multi-fidelity modeling for online sensors correction. *Computers & Chemical Engineering*, 211:109665, 2026.

[32] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[33] John A. Nelder and Roger Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.

[34] Katalin M. Hangos and Ian T. Cameron. *Process Modelling and Model Analysis. Method and Manner*. Academic Press, London, 2001.

[35] Katalin M. Hangos and Ian T. Cameron. *Process Modelling and Model Analysis. Matter*. Academic Press, London, 2001.

[36] Dale E. Seborg, Thomas F. Edgar, Duncan A. Mellichamp, and Francis J. Doyle. *Process Dynamics and Control. Theoretical Models of Chemical Processes*. John Wiley & Sons, 3 edition, 2010.

[37] Siemens. gPROMS – digital process twin technology. `https://www.siemens.com/global/en/products/automation/industrysoftware/gproms-digital-process-design-and-operations.html`. Accessed: 2026-05-17.

[38] AVEVA. AVEVA Process Simulation. `https://www.aveva.com/en/products/process-simulation/`, 2024. Accessed: 2026-05-17.

[39] Giuseppe Armenise, Marco Vaccari, Riccardo Bacci di Capaci, and Gabriele Pannocchia. An open-source system identification package for multivariable processes. In *2018 UKACC 12th International Conference on Control (CONTROL)*, pages 152–157. IEEE, 2018.

[40] Y. Liu, C. Yang, Z. Gao, and Y. Yao. Rebooting data-driven soft-sensors in process industries: A review of kernel methods. *Journal of Process Control*, 89:58–73, 2020.

[41] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[42] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021.

[43] Iftikhar Ahmad, Ahsan Ayub, Manabu Kano, and Iqbal I. Cheema. Gray-box soft sensors in process industry: Current practice, and future prospects in era of big data. *Processes*, 8(2):243, 2020.

[44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[45] Francois Chollet et al. Keras, 2015.

[46] Rastislav Fáber, Karol Ľubušký, Martin Mojto, and Radoslav Paulen. Enhancing industrial data analysis through machine learning-based classification of petrochemical datasets. In *49th International Conference of the Slovak Society of Chemical Engineering SSCHE 2023*, page 160, Bratislava, Slovakia, 2023. Slovak Society of Chemical Engineering.

[47] Rastislav Fáber, Karol Ľubušký, and Radoslav Paulen. Machine learning-based classification of online industrial datasets. In Radoslav Paulen and Miroslav Fikar, editors, *Proceedings of the 2023 24th International Conference on Process Control*, pages 132–137, Slovak University of Technology in Bratislava, 2023. IEEE.

[48] Rastislav Fáber, Martin Mojto, Karol Ľubušký, and Radoslav Paulen. From data to alarms: Data-driven anomaly detection techniques in industrial settings. In Flavio Manenti and G. V. Rex Reklaitis, editors, *34th European Symposium on Computer Aided Process Engineering / 15th International Symposium on Process Systems Engineering*, volume 53, pages 2857–2862, 2024.

[49] Rastislav Fáber, Martin Mojto, Karol Ľubušký, and Radoslav Paulen. Integrated data analytics and regression techniques for real-time anomaly detection in industrial processes. In *12th IFAC Symposium on Advanced Control of Chemical Processes*, pages 320–325, 2024.

[50] Rastislav Fáber, Martin Klaučo, Karol Ľubušký, and Radoslav Paulen. Integrating linear and nonlinear models for enhanced process monitoring. In Radoslav Paulen, Miroslav Fikar, and Ján Oravec, editors, *Proceedings of the 2025 25th International Conference on Process Control*, pages 1–6. IEEE, 2025.

[51] Rastislav Fáber, Marco Vaccari, Riccardo Bacci di Capaci, Karol Ľubušký, Gabriele Pannocchia, and Radoslav Paulen. Multi-fidelity modeling for process optimization in refinery operations. In *51st International Conference of the Slovak Society of Chemical Engineering SSCHE 2025*, page 140, Bratislava, Slovakia, 2025. Slovak Society of Chemical Engineering.

[52] Rastislav Fáber, Marco Vaccari, Riccardo Bacci di Capaci, Karol Ľubušký, Gabriele Pannocchia, and Radoslav Paulen. Improving process monitoring via dynamic multi-fidelity modeling. In *14th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems*, pages 199–204. Elsevier, 2025.

[53] Rastislav Fáber, Marco Vaccari, Riccardo Bacci di Capaci, Karol Ľubušký, Gabriele Pannocchia, and Radoslav Paulen. Soft-sensor-enhanced monitoring of an alkylation unit via multi-fidelity model correction. *Systems and Control Transactions*, 4:1389–1395, 2025.

[54] Rastislav Fáber, Karol Ľubušký, and Radoslav Paulen. A hybrid data-driven approach for the optimization of an industrial alkylation unit. In *ESCAPE 36 – European Symposium on Computer Aided Process Engineering*, Systems and Control Transactions, Sheffield, United Kingdom, 2026. Accepted.

[55] Rastislav Fáber, Karol Ľubušký, and Radoslav Paulen. A simplified framework for process model development in industrial simulation environments. In *52nd International Conference of the Slovak Society of Chemical Engineering SSCHE 2026*, Bratislava, Slovakia, 2026. Accepted.

[56] Rastislav Fáber, Marco Vaccari, Riccardo Bacci di Capaci, Karol Ľubušký, Gabriele Pannocchia, and Radoslav Paulen. Physics-guided soft sensing for anomaly detection: Simulation surrogates in industrial processes. *Chemical Engineering Research and Design*, 2026. Submitted.