

https://dspace.org/

Development of a Soft Sensor to Estimate Fermentation Rate Using Temperature Measurements During Ethanol Fermentation

Naidoo, Adam Kyle

2025
University of the Witwatersrand, Johannesburg
https://hdl.handle.net/10539/50264
Naidoo, Adam Kyle. (2025). Development of a Soft Sensor to Estimate Fermentation Rate Using Temperature Measurements During Ethanol Fermentation [Master's dissertation, University of the Witwatersrand, Johannesburg]. WIReDSpace. https://hdl.handle.net/10539/50264

Downloaded from a DSpace repository. https://dspace.org/



UNIVERSITY OF THE
WITWATERSRAND,
JOHANNESBURG

Development of a Soft Sensor to Estimate Fermentation Rate Using Temperature Measurements During Ethanol Fermentation

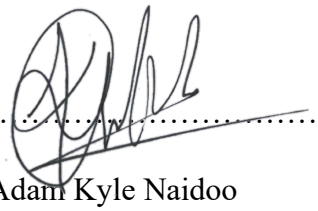
Adam Kyle Naidoo

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering

2025

DECLARATION

I declare that this dissertation is my own unaided work. It is being submitted to the Degree of Master of Science to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other university.



……

Adam Kyle Naidoo

…03…… day of ……November……, ……2025……

ABSTRACT

This dissertation presents the development of a soft sensor designed to estimate ethanol fermentation rates using only temperature-derived measurements. The sensor was constructed through a delta-based machine learning approach, tracking the rate of change rather than absolute values, with thermodynamic reasoning embedded into the feature engineering process. Ten fed-batch fermentation experiments were conducted using *Saccharomyces cerevisiae* in a 100 L stainless-steel bioreactor, with data collected on fermenter, jacket, and ambient temperatures, alongside specific gravity. Ethanol concentrations were estimated via a validated model based on the work of Cutaia et al. (2009) and cross-verified against HPLC readings, achieving a combined R^2 of 0.75 and NRMSE of 0.17 across nine experiments. A Gaussian Process Regression (GPR) model, trained on four delta-based input features and smoothed via a 40-point moving average, emerged as the top performer. The final model reduced RMSE by 79.18%, from 5.62 g/L (ABS) to 1.17 g/L (DELTA) and achieved a maximum R^2 of 0.81 during validation. Testing on three unseen experiments, including disturbed and feedback-controlled scenarios, demonstrated robust trend tracking, with the best case yielding an RMSE of 0.34 g/L and R^2 of 0.50. This research confirms the viability of using thermal data to infer yeast activity and ethanol production in real-time, offering a low-cost, non-invasive alternative to traditional analytical methods for fermentation monitoring in industrial environments.

To Anthea

To Jet & Nimbus

ACKNOWLEDGEMENTS

To my supervisor, Dr. Antony Higginson, for his invaluable guidance, coaching, and inspiration throughout this journey. His unwavering support and mentorship mean more to me than he may ever know.

To the Data Analytics, Numerics, and Control Engineering (DANCE) Unit, led by Prof. Kevin Brooks and Dr. Antony Higginson at the University of the Witwatersrand's School of Chemical and Metallurgical Engineering. I am deeply grateful for the camaraderie, the stimulating discussions, and the many probing questions that not only strengthened the quality of this research but also enriched the quality of my thinking.

To the Faculty of Engineering and the Built Environment at the University of the Witwatersrand, whose financial support through the Postgraduate Master's Merit Award (PMA) made this work possible. Without this assistance, the successful completion of this research would not have been attainable.

CONTENTS

DECLARATION ii

ABSTRACT iii

DEDICATION iv

ACKNOWLEDGEMENTS v

CONTENTS vi

LIST OF FIGURES ix

LIST OF TABLES x

NOMENCLATURE xi

1. INTRODUCTION 1

1.1 Overview 1

1.2 Motivation 2

1.3 Aims 3

2. LITERATURE REVIEW 5

2.1 Mechanistic & Metabolic Basis of Ethanol Fermentation 5

2.2 Instrumentation for Fermentation Monitoring & Control 8

2.3 Soft Sensors & Predictive Modelling 11

2.4 Performance Evaluation Metrics for Soft Sensors 16

2.5 Challenges & Limitations in Soft Sensor Deployment 20

2.6 MLOps & Real Time Deployment 24

2.7 Overview of Regression Models Applied in This Study 26

3. EXPERIMENTAL SETUP & METHODOLOGY 34

3.1 Experimental Setup 34
3.1.1 Fermentation System 34
3.1.2 Instrumentation & Data Collection 37

3.2 Soft Sensor Development 39
3.2.1 Data Preparation & Modelling 39
3.2.2 Validation with HPLC 41

3.3 Soft Sensor Methodical Approach 43
3.3.1 Rationale for Variable Selection 44
3.3.2 Data Preprocessing 45
3.3.3 Feature Engineering & Thermodynamic Reasoning 45
3.3.4 Model Training & Testing 49

4. RESULTS & DISCUSSION 51

4.1 Fermentation Dynamics Across Varying Conditions 51

4.2 Verification of Ethanol Targets for Soft Sensor Training 53

4.3 Comparison of Delta and Absolute Value Approaches 58

4.4 Ethanol Soft Sensor Validation 61
4.4.1 Realistic Training & Generalizable Testing 61
4.4.2 Input Configuration & Smoothing Effects on Model Accuracy 63

4.5 Soft Sensor Testing 65

4.6 Output Smoothing 68

4.7 Developed Soft Sensor 70

5. CONCLUSION & RECOMMENDATION 77

5.1 Conclusion 77

5.2 Recommendation 78

5.3 Future Work 78

REFERENCES 81

APPENDICES 89
A1 – Temperature & Specific Gravity Profile for Each Experiment 89
A2 – Modelled Ethanol Concentration from SG Measurements Validation 94
A3 – Soft Sensor Data Preprocessing Code 95

A4 – Raw Training RMSE Data for Delta and Absolute Scenarios 97
A5 – Soft Sensor Raw Output Code 99
A6 – Top Performing Model by Family Predictive vs Actuals Validation 101
A7 – Variable Smoothing Ethanol Prediction Code 105

LIST OF FIGURES

Figure 2—1: Metabolic pathways in K. lactis and S. cerevisiae 7
Figure 2—2 Fermentation sensor-feedback via PID and SCADA 9
Figure 2—3: Fermenter headplate with inline pH, DO, and temperature sensors 10
Figure 2—4: GA-MK-SVM soft sensor architecture 12
Figure 2—5: DBN and DNN architecture 13
Figure 2—6: Adaptive FNN + SVR architecture 14
Figure.2—7: AE + SVR + GA architecture for rotor deformation 15
Figure 2—8: Fuzzy Clustered RVM with Hybrid Kernel architecture 16
Figure 2—9: Visual representation of generation of synthetic data 18
Figure 2—10: Sensor fault types and detection strategies 21
Figure 2—11: Hybrid control realities in industrial fermentation 22
Figure 2—12: MLOps deployment pipeline for fermentation soft sensors 25
Figure 3—1: Ethanol fermentation setup at the Wits Microbrewery 35
Figure 3—2: iSpindle™ electronic hydrometer 38
Figure 4—1: Experiment 1 time profile of temperature and specific gravity during fermentation 52
Figure 4—2: HPLC and predicted ethanol measurement parity plot for experiments 1-10 55
Figure 4—3: Overall parity plot for HPLC vs predicted ethanol 57
Figure 4—4: Absolute and Delta prediction error and accuracy analysis 60
Figure 4—5: RMSE comparison between EO and NEO splits on validation and test data sets 62
Figure 4—6: GPR actual (blue) vs prediction (mustard) validation of top performing model 65
Figure 4—7: GPR test response 67
Figure 4—8: Smoothing response of Gaussian Process Regression output 70
Figure 4—9: Soft sensor prediction vs actual for experiment 5 72
Figure 4—10: Soft sensor prediction vs actual for experiment 8 73
Figure 4—11: Soft sensor prediction vs actual for experiment 10 75
Figure 4—12: Parity plot for each of the test experiments after smoothing 76

LIST OF TABLES

Table 3—1: Initial experiment conditions 36
Table 3—2: Experiment feed rate conditions 36
Table 3—3: Variable update frequency during live fermentations 41
Table 3—4: Train and test data split 49
Table 4—1: RMSE validation results for ABS and DELTA soft sensor approaches 58
Table 4—2: R^2 validation results across input configurations and smoothing 63
Table 4—3: Summary of soft sensor test result outputs 66
Table 4—4: RMSE & R^2 results after smoothing raw soft sensor outputs 69

NOMENCLATURE

Symbol / Abbreviation	Explanation / Units
A	Heat transfer surface area (m^2)
AE	Apparent Extract °P (Plato)
ATP	Adenosine Triphosphate
$A_{w/w}$	Alcohol by weight (mass fraction) % (w/w)
CO_2	Carbon Dioxide
DBN	Deep Belief Network
DNN	Deep Neural Network
DO	Dissolved Oxygen
DS	Differential Smoothing
EO	Experiment Optimization
FNN	Feedforward Neural Network
GA	Genetic Algorithm
GAMP	Good Automated Manufacturing Practice (GAMP 5)
GMP	Good Manufacturing Practice
GPR	Gaussian Process Regression
HPLC	High-Performance Liquid Chromatography
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MK	Multi-Kernel
ML	Machine Learning
MLOps	Machine Learning Operations
NDS	No Differential Smoothing

NEO	No Experiment Optimization
NRMSE	Normalised Root Mean Square Error
OE	Original Extract (initial sugar concentration) °P (Plato)
PAT	Process Analytical Technology
PCA	Principal Component Analysis
PID	Proportional–Integral–Derivative (controller)
PLC	Programmable Logic Controller
PLS	Partial Least Squares
PSI	Population Stability Index
$\mathbf{Q_{acc}}$	Accumulated heat in the system (kJ)
$\mathbf{Q_{gen}}$	Heat generated by yeast metabolism (kJ)
$\mathbf{Q_{in}}$	Heat added to the system (kJ)
$\mathbf{Q_{loss}}$	Heat lost to the environment (kJ)
$\mathbf{Q_{out}}$	Heat removed from the system (kJ)
$\mathbf{R^2}$	Coefficient of Determination
RBF	Radial Basis Function
RE	Real Extract °P (Plato)
RMSE	Root Mean Square Error
RVM	Relevance Vector Machine
SAE	Stacked Autoencoder
SCADA	Supervisory Control and Data Acquisition
SG	Specific Gravity (dimensionless)
SVM	Support Vector Machine
SVR	Support Vector Regression

T	Temperature (°C)
UKF	Unscented Kalman Filter
V	Volume (L)
VAE	Variational Autoencoder
VFD	Variable Frequency Drive
pH	Measure of acidity/alkalinity (dimensionless)
ΔT	Change in temperature (°C)

1. INTRODUCTION

1.1 Overview

Modern process industries depend on accurate, real-time information to make decisions that optimize process performance and ensure product quality. In biochemical processes such as ethanol fermentation, decisions regarding process control are heavily reliant on available process information (López Pérez et al., 2013). However, direct, real-time measurement of ethanol concentration remains a significant challenge. Analytical techniques like high-performance liquid chromatography (HPLC), though precise, are expensive, labour-intensive, and unsuitable for continuous, real-time monitoring in industrial environments (Badino et al., 2020). A critical barrier to the broader automation of bioprocesses such as fermentation is the lack of reliable, fast, and affordable sensors for key variables like ethanol concentration (Saucedo et al., 1995). While several easily measurable variables including temperature, pH, agitation speed, dissolved oxygen, and off-gas composition are readily available and inexpensive to monitor, they provide indirect information about fermentation progress and do not explicitly reflect ethanol concentration (Doan et al., 2007). Soft sensors, also known as inferential sensors, present a viable solution to this challenge by providing real-time estimates of difficult-to-measure variables through the use of process data and mathematical models (Kadlec et al., 2009). Unlike hardware sensors, soft sensors employ computational models to infer information from existing measurements, requiring no additional hardware investment. The use of soft sensors has become prevalent in industries such as chemicals, power generation, and petrochemicals due to their capability to extract hidden process information, enhance operational efficiency, detect faults, and identify optimal operating conditions (Khatibisepehr et al., 2013; Y. Liu & Xie, 2020). The term “soft” refers to their basis in software, contrasting with

hardware-based physical sensors (Kadlec et al., 2009). Despite their growing application in other industries, the development of soft sensors specifically for ethanol estimation in fermentation processes, using variables like temperature, remains underexplored. Temperature offers a promising avenue for soft sensing due to its routine measurement in fermentation systems and its correlation with the exothermic nature of yeast metabolism. However, the complex, nonlinear dynamics of fermentation and the indirect relationship between temperature and ethanol production present significant modelling challenges. This research addresses this gap by developing a machine learning-based soft sensor for the estimation of ethanol concentration during fermentation, using temperature as the primary input. A key focus of this study is on modelling changes in ethanol concentration (delta ethanol), which serves as a better indicator of yeast metabolic activity and fermentation dynamics than absolute concentration alone. The soft sensor is developed using various machine learning model families and evaluated under different smoothing strategies to improve predictive stability and accuracy. The adoption of soft sensors aligns with the broader trends of the Fourth Industrial Revolution, where artificial intelligence and data-driven modelling are increasingly employed to improve industrial processes, enhance environmental performance, and maintain competitiveness (Maschler et al., 2021; Matheri et al., 2021).

1.2 Motivation

Although there is potential for a variety of returns from the research conducted and presented in this dissertation, the main motivations are environmentally and cost driven. Fossil fuels are scarce and come with an array of environmental risks with the most threatening being climate change. As a result, the world has shifted its attention to alternative more environmentally friendly energy sources. These sources are known as biofuels and are available in various forms such as bio-alcohol, biodiesel, biocrude, biochar, biogas and biohydrogen (Liew et al., 2014). Of all these products processes, the world's largest bioprocess is the production of bioethanol via fermentation of

biomass containing starch and sugar, with the sale of more than 85 million tons of ethanol worldwide in 2018 (Naveed et al., 2024). There is an incentive to look at ethanol production, biochemical processes, and biofuels from an optimization approach based on the environmental and market factors. Reaction rates of fermentation can be estimated by performing laboratory analysis of the solution or using mathematical equations based on different process variables. The former estimate method is normally expensive and is time inefficient whilst the latter provides estimates that are accurate only under specific conditions, which is not a popular solution for large scale manufacturing companies where product specifications are likely to change often. Live reaction rates during fermentation allow for early detection by providing real time product measurement which can correct anomalous batches, allow effective scheduling of production equipment by predicting fermentation endpoint, and reduce the burden of manual sampling by operators (Bowler et al., 2021).

1.3 Aims

Industrial sensors are the backbone of operating an industrial process (Brooks & Bauer, 2018). This research aims to optimize how industrial sensors are applied. The research objectives are mentioned below in detail:

- To acquire fermentation data from multiple fermentations and determine fermentation rates.
- To evaluate the relationship between fermentation temperature and ethanol production.
- To develop a soft sensor based on heat transfer and heat produced in the reactor.
- To compare estimated fermentation rate with experimental data.
- To validate the soft sensor based on additional experimental data.

Temperature data such as ambient, bioreactor/fermenter, and jacket or cooling data will be used as the primary data to achieve the objectives listed above to determine how

accurate reaction rates (ethanol production and sugar consumption) can be predicted using machine learning.

2. LITERATURE REVIEW

The design and implementation of a soft sensor to estimate ethanol fermentation rate from temperature data requires an interdisciplinary understanding of biochemical processes, industrial instrumentation, and data-driven modelling techniques. This literature review synthesizes foundational and contemporary research across these domains to establish a theoretical and methodological basis for the study. The review is structured to first present the biochemical mechanisms and process parameters governing ethanol fermentation. This is followed by an overview of common instrumentation used in fermentation systems, including their limitations in industrial applications. Subsequently, the focus shifts to soft sensing methodologies, highlighting their increasing relevance in process industries. Special emphasis is placed on regression-based machine learning techniques, which form the computational backbone of the soft sensor developed in this research. Finally, the review explores data preprocessing strategies and validation techniques essential for building robust, real-time predictive models. Where relevant, current gaps in literature are identified to motivate the specific approach adopted in this study.

2.1 Mechanistic & Metabolic Basis of Ethanol Fermentation

Ethanol fermentation in *Saccharomyces cerevisiae* is a tightly regulated metabolic process in which glucose is converted to ethanol and carbon dioxide, even in the presence of oxygen, a phenomenon known as the Crabtree effect (De Deken, 1966). This metabolic strategy is triggered by high external glucose concentrations, which promote a shift toward fermentative metabolism rather than respiration, regardless of oxygen availability (Duncan et al., 2024). This evolutionary adaptation offers *S. cerevisiae* a competitive advantage by allowing rapid energy production and ethanol

accumulation, which suppresses the growth of competing microorganisms in mixed cultures (Duncan et al., 2024). The biochemical sequence begins with glycolysis, in which glucose is enzymatically broken down into two molecules of pyruvate. This process yields a net gain of two ATP molecules and two NADH per glucose, providing energy and redox balance to the cell (van Aalst et al., 2022). Under anaerobic or oxygen-limited conditions, *Saccharomyces cerevisiae* regenerates nicotinamide adenine dinucleotide (NAD^+) to sustain glycolytic activity by reducing acetaldehyde to ethanol through the action of alcohol dehydrogenase. This follows the decarboxylation of pyruvate into acetaldehyde, catalysed by pyruvate decarboxylase, forming a core part of the organism's native ethanol production pathway (van Rossum et al., 2016).

The stoichiometric equation representing this pathway is (Brown et al., 2015):

$$C_6H_{12}O_6(s) \xrightarrow{yeast} 2C_2H_5OH(l) + 2CO_2(g) \quad (2\text{-}1)$$

This equation outlines the transformation of glucose into ethanol and carbon dioxide, along with energy generation in the form of adenosine triphosphate (ATP) (van Rossum et al., 2016). It provides the basis for industrial ethanol fermentation and is widely accepted across biotechnological applications. Figure 2—1 showcases the different pathways involved in glucose and ethanol assimilation under aerobic conditions, underlining the differences between a Crabtree-negative yeast *Kluyveromyces lactis* (thin arrows), which degrades hexoses directly to CO2, and a Crabtree-positive yeast, *Saccharomyces cerevisiae* (thin and thick arrows), which can accumulate ethanol. P: phosphate; TCA cycle: tri-carboxylic acid cycle (Hellborg & Piškur, 2009).

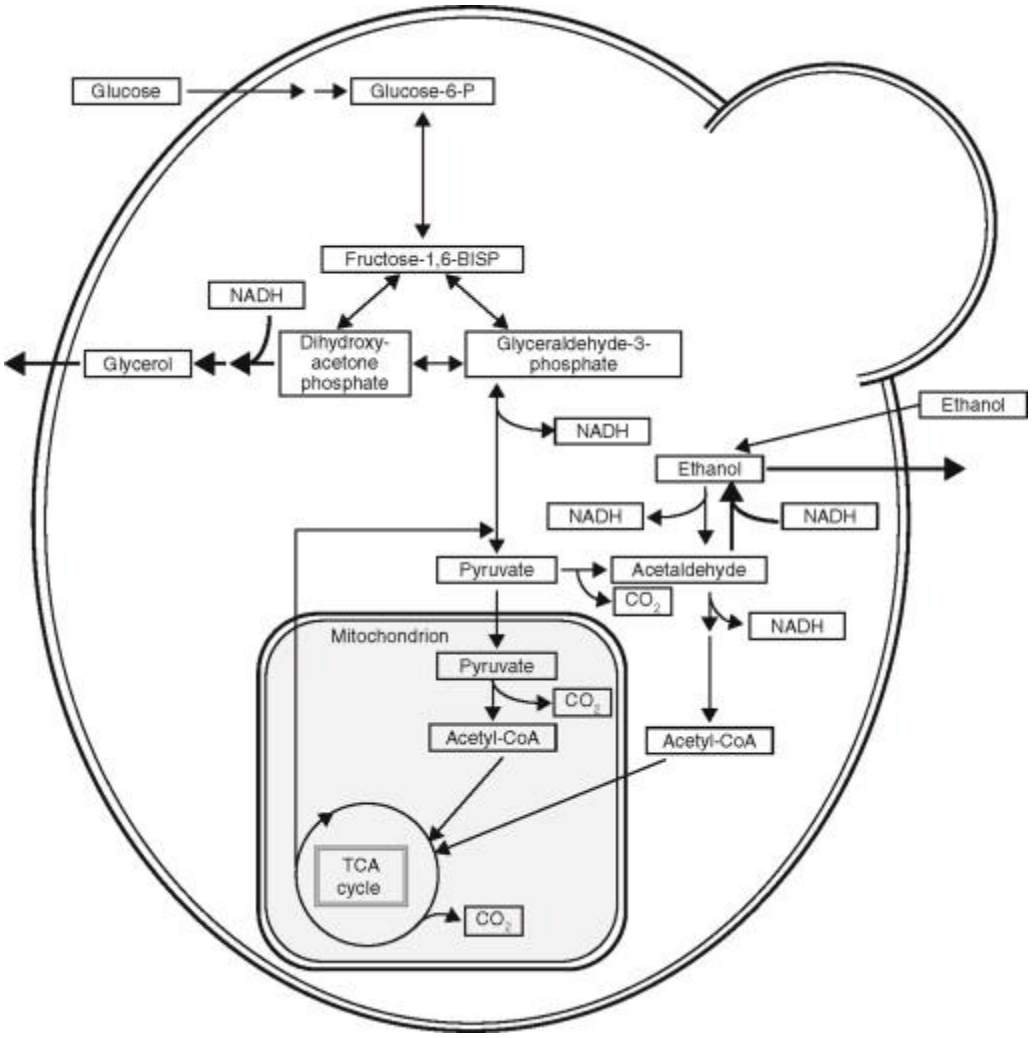


Figure 2—1: Metabolic pathways in K. lactis and S. cerevisiae

adapted from *(Hellborg & Piškur, 2009)*

In comparison, Crabtree-negative yeasts such as *Kluyveromyces lactis* maintain respiratory metabolism in the presence of oxygen, leading to biomass accumulation over ethanol production. The metabolic divergence is further detailed in (Y. Zhang et al., 2022), who present a rewired *S. cerevisiae* strain with inhibited hexose transporters and downregulated ethanol-producing enzymes, restoring respiratory dominance even under glucose-rich conditions (Y. Zhang et al., 2022). Ethanol yield in *Saccharomyces cerevisiae* is significantly influenced by environmental conditions such as temperature,

ethanol concentration, and pH. Optimal fermentative performance occurs at temperatures between 28 to 33 °C, with temperatures above this range risking enzyme denaturation and thermal stress on yeast cells (Lip et al., 2020). This range ensures the effective activity of alcohol dehydrogenase and other key glycolytic enzymes. Elevated ethanol concentrations can also suppress yeast viability and performance. Concentrations exceeding 12 to 15 percent by volume lead to disruption of membrane integrity and decreased population growth, which directly impacts the final yield (Douradinho et al., 2023). Furthermore, osmotic stress caused by high ethanol or sugar concentrations triggers the production of glycerol, which diverts metabolic flux away from ethanol generation (Veloso et al., 2019). Yeast performance is also closely tied to pH. *S. cerevisiae* is considered acidophilic and thrives best in environments with a pH range of 4.0 to 6.0. Outside this range, the proton gradient is disrupted, leading to impaired nutrient uptake and enzyme activity (Narendranath & Power, 2005).

2.2 Instrumentation for Fermentation Monitoring & Control

In industrial fermentation systems, especially those used for ethanol production, real-time instrumentation control plays a critical role in maintaining optimal microbial activity and maximizing process yield. Process control relies heavily on accurate sensor readings and automated regulation via control architectures like PLCs and SCADA systems. Literature has consistently emphasized the importance of primary variables such as temperature, pH, and dissolved oxygen (DO), while also introducing the potential of soft sensing to reduce dependency on extensive physical instrumentation. Temperature, pH, and dissolved oxygen (DO) remain the most consistently monitored variables in fermentation bioreactors, as shown in Figure 2—2, due to their established role in bioprocess control and instrumentation (Bellon-Maurel et al., 2003). These parameters are known to directly impact microbial viability, enzymatic activity, and metabolite production during fermentation processes (Lip et al., 2020). Temperature sensors, particularly resistance temperature detectors (RTDs), are commonly used in industrial bioreactors and are based on the principle that platinum wire resistance varies

with temperature, enabling reliable real-time thermal monitoring (Vojinović et al., 2006). Sensors for pH and DO, although standard in most fermenters, are known to suffer from limitations in robustness and sensitivity under industrial conditions, which has historically slowed the adoption of more advanced monitoring platforms (Reardon, 2021). Despite the maturity of physical sensor platforms, no published studies have investigated temperature as the sole variable for inferring ethanol production in *S. cerevisiae* fermentations. Most fermentation operations integrate their physical sensors with programmable logic controllers (PLCs) and supervisory control and data acquisition (SCADA) platforms to enable dynamic process control. These systems rely on proportional–integral–derivative (PID) controllers to regulate flow rates, temperature, and nutrient addition (Hocalar et al., 2011). However, PID algorithms assume linear dynamics and short time delays, which are rarely present in microbial systems. Many fermentation processes are inherently non-linear and involve significant time lags, which limits the effectiveness of PID control in maintaining tight regulation (O' Connor et al., 2002).

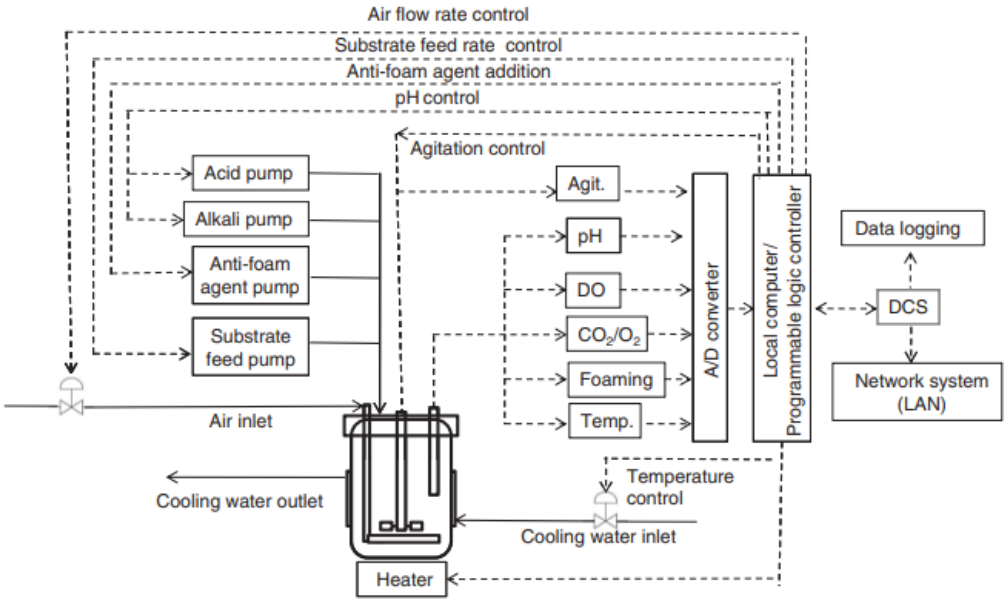


Figure 2—2 Fermentation sensor-feedback via PID and SCADA

adapted from **(Katoh et al., 2015)**

To address limitations in physical sensors, soft sensors have emerged to estimate difficult-to-measure fermentation parameters like ethanol or biomass concentration. These models typically leverage multivariate input data and are developed using machine learning algorithms such as regression models, support vector machines (SVMs), or neural networks (Goldrick et al., 2019). However, most reported soft sensors rely on complex input arrays, often including DO, pH, off-gas CO_2, and agitation rate, all of which remain vulnerable to sensor drift, cost, or maintenance overhead (Vojinović et al., 2006). Figure 2—3 shows a schematic representation of a laboratory-scale fermenter headplate equipped with inline sensors for pH, DO, and temperature monitoring. Such configurations are typical in stirred-tank bioreactors used in microbial fermentations, where in-situ control of environmental conditions is vital for maintaining optimal growth and metabolite production. The arrangement supports automated feedback control and is representative of standard industry instrumentation for bench-scale process development (Gill et al., 2008).

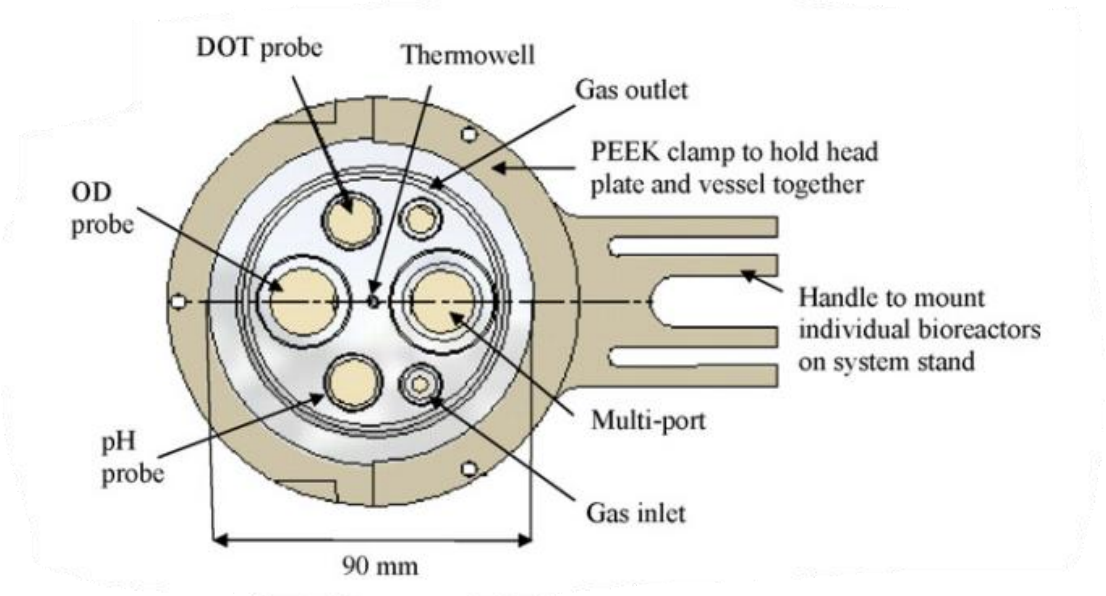


Figure 2—3: Fermenter headplate with inline pH, DO, and temperature sensors adapted from (Gill et al., 2008).

2.3 Soft Sensors & Predictive Modelling

Soft sensors, or inferential models, are designed to estimate variables that are difficult to measure in real time by leveraging easily accessible process data. These variables often require costly laboratory analysis or are subject to significant measurement delays (Souza et al., 2016). With the proliferation of machine learning in industrial systems, soft sensors have evolved to incorporate data-driven methods like support vector machines (SVM), neural networks, and kernel-based regression frameworks, enabling robust prediction in highly nonlinear and noisy environments. The development of soft sensors has transitioned from traditional linear methods to more advanced machine learning-based systems. H. Liu et al. (2020) explain that early soft sensors often relied on linear variable models such as principal component analysis (PCA) and partial least squares (PLS), but these approaches assume linear correlation and perform poorly when applied to complex, nonlinear industrial data. To overcome these limitations, modern soft sensors incorporate techniques like support vector machines (SVM), deep neural networks (DNN), and hybrid learning structures. These tools offer superior generalization and noise resilience, especially when dealing with high-dimensional or nonlinear data. The earliest generation of soft sensors were built using linear statistical methods. These included multiple linear regression and partial least squares (PLS), valued for their simplicity and interpretability. However, as H. Liu et al. (2020) observe, linear models often fail under nonlinear, multivariate process dynamics. While purely linear methods are being replaced, many hybrid soft sensors still begin with traditional steps like variable selection, regression initialization, and system identification. These preprocessing layers retain structural logic from classical regression pipelines, forming the foundation for more adaptive and data-driven approaches. SVMs provide a strong alternative to linear models by allowing for nonlinear mappings of data into higher-dimensional feature spaces. H. Liu et al. (2020) applied an SVM-based soft sensor in a fermentation process using genetic simulated annealing to optimize kernel parameters, achieving high prediction accuracy in biomass concentration estimation. In a separate case, Zhai et al. (2023) developed a

hybrid model for coal temperature prediction using genetic algorithm (GA) optimized multi-kernel (MK) SVM. The model combined polynomial and Radial Basis Function (RBF) kernels with adaptive weighting.

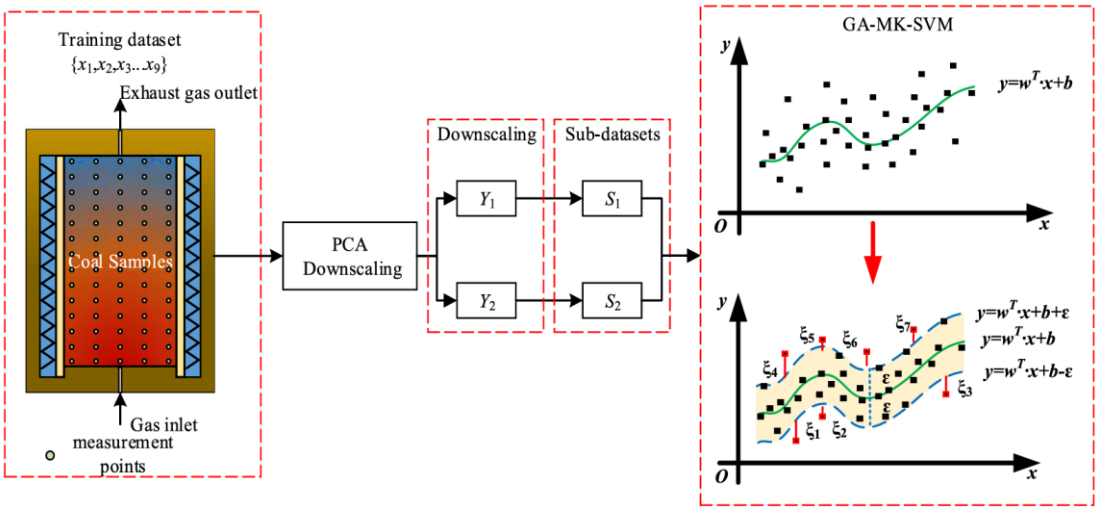


Figure 2—4: GA-MK-SVM soft sensor architecture

adapted from (Zhai et al., 2023)

This structure integrates PCA-based input compression, sub-dataset formation, and GA-optimized kernel modelling. Although applied to coal combustion, this example demonstrates how easily measurable variables can be used to estimate complex process parameters, an approach directly relevant to fermentation systems. Neural networks, particularly deep architectures, have emerged as a powerful modelling tool for soft sensors. Shang et al. (2014) describe how deep belief networks (DBNs) and deep neural networks (DNNs) allow for hierarchical feature abstraction, improved representation of nonlinear process data, and strong generalization even with limited labelled data.

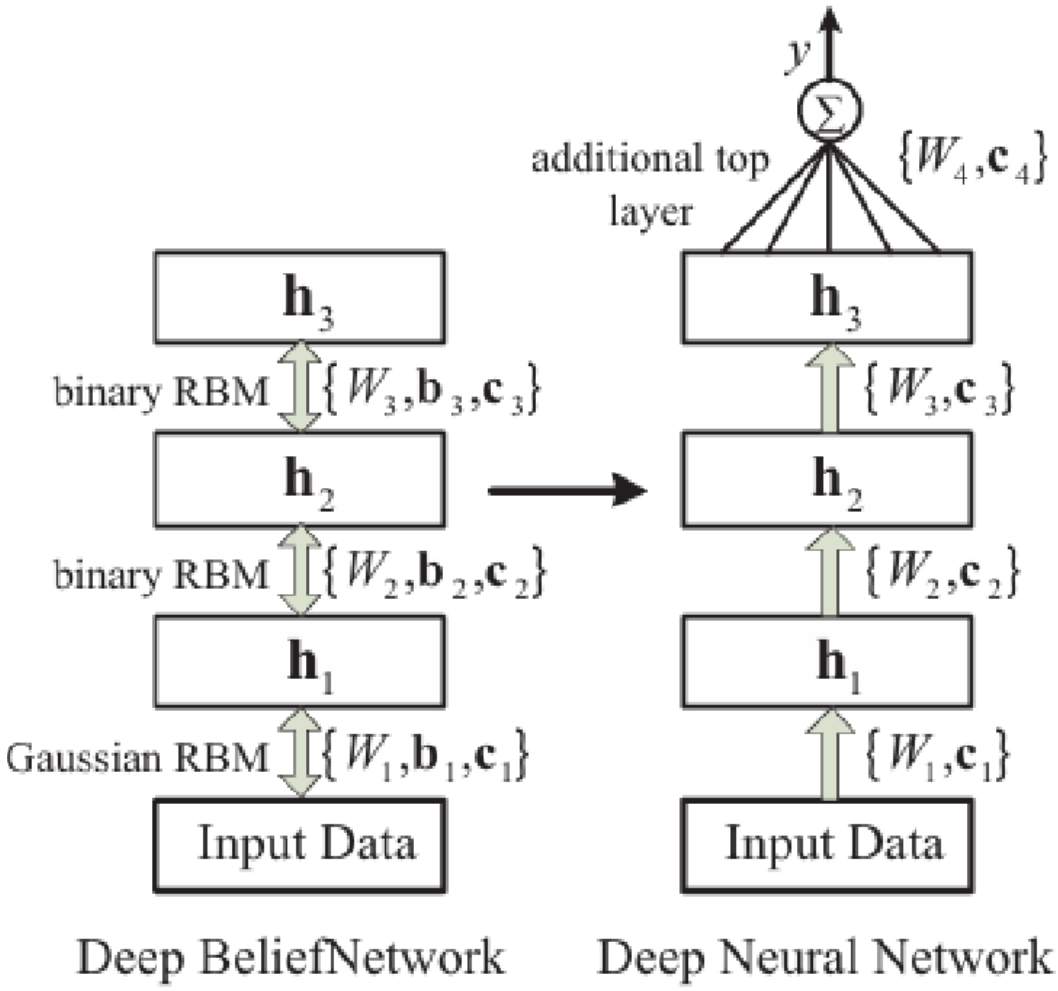


Figure 2—5: DBN and DNN architecture

adapted from (Shang et al., 2014)

This structure illustrates unsupervised pretraining using Restricted Boltzmann Machines (RBMs), followed by supervised fine-tuning in a DNN layer, offering stable performance in sparse data conditions. As industrial systems become more nonlinear and high-dimensional, single-method soft sensor models often fall short. M. Zhang and Liu (2013) emphasizes that hybrid architectures, such as those combining fuzzy logic with SVR, can better generalize in dynamic environments. Wang and Liu (2018)

similarly employed a stacked autoencoder (SAE) with SVR and genetic algorithm optimization for deformation prediction in thermal plants.

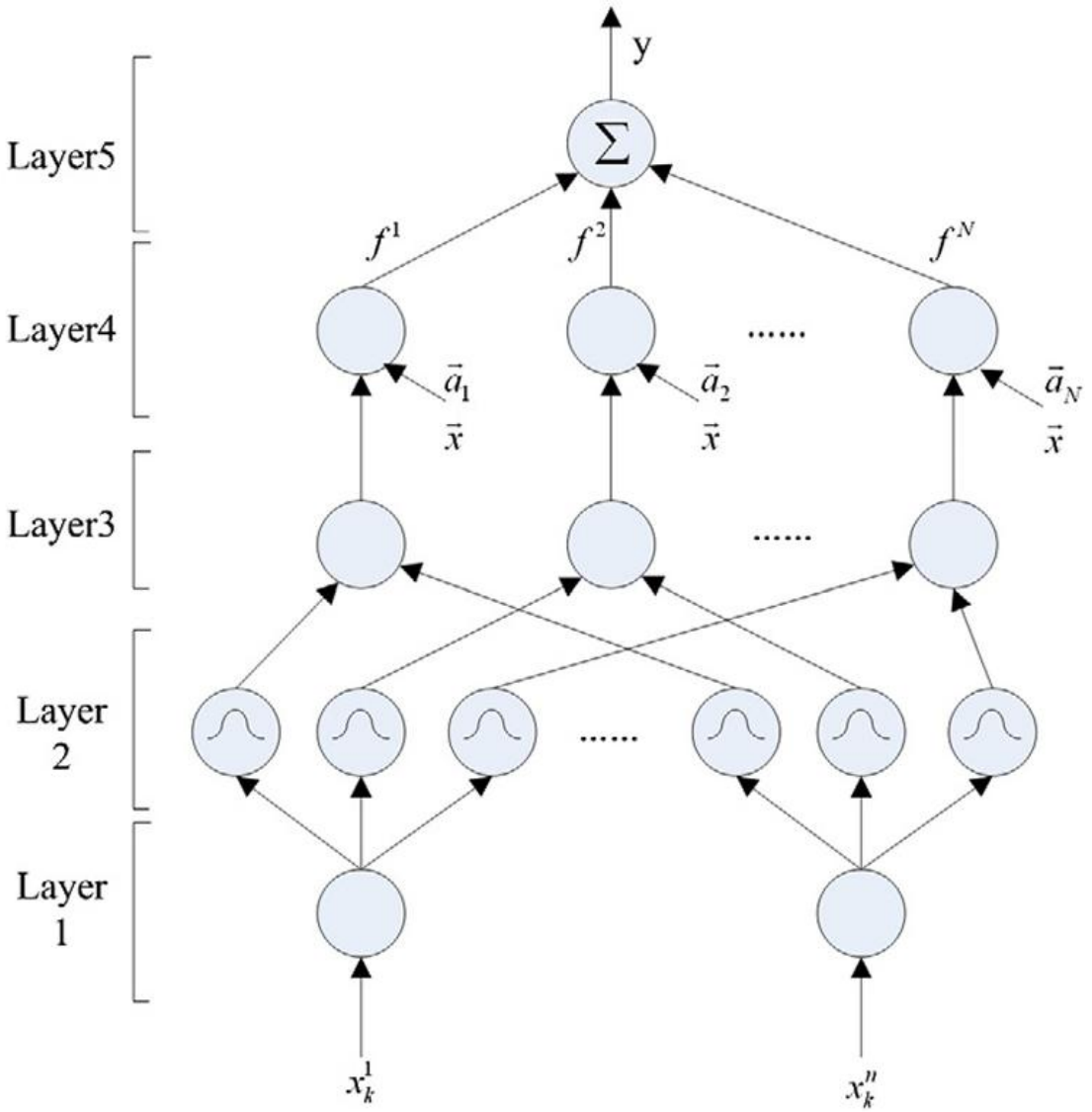


Figure 2—6: Adaptive FNN + SVR architecture

adapted from (M. Zhang & Liu, 2013)

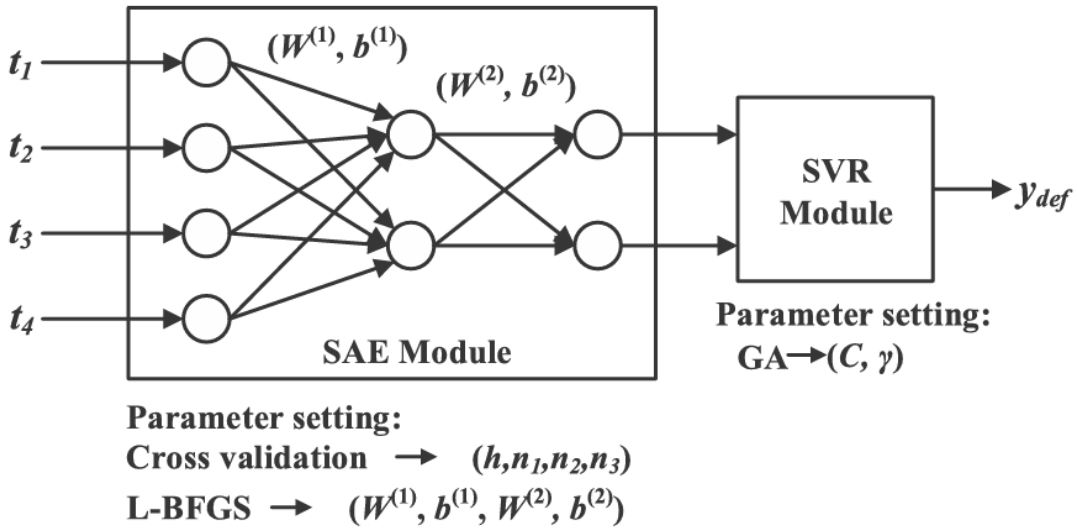


Figure.2—7: AE + SVR + GA architecture for rotor deformation

adapted from (Wang & Liu, 2018)

These non-bioprocess examples highlight the versatility of machine-learning soft sensors in estimating unmeasured variables from accessible data, a principle equally applicable to ethanol fermentation. In another example, Hu et al. (2024) proposed a relevance vector machine (RVM) model that uses fuzzy clustering to handle incomplete outputs in sintering.

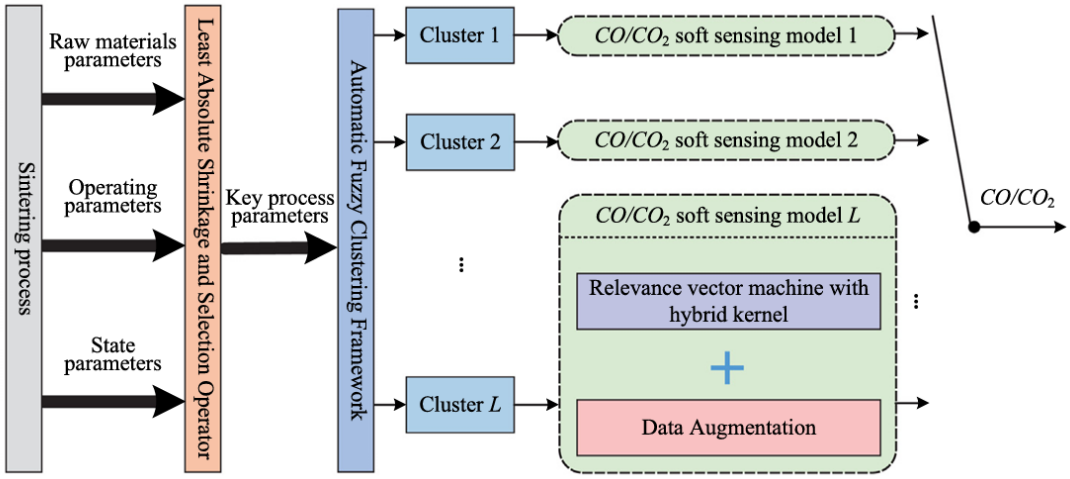


Figure 2—8: Fuzzy Clustered RVM with Hybrid Kernel architecture

adapted from (Hu et al., 2024)

Morales et al. (2023) used an Unscented Kalman Filter (UKF) to estimate supersaturation in sugar crystallization. This model-based soft sensor achieved <1.5% MAE under varying process conditions. Zhai et al. (2023) applied GA-MK-SVM to underground coal combustion, achieving 1.616% MAPE. Hu et al. (2024) modelled CO/CO_2 ratios across fuzzy-clustered modes using RVMs. Each demonstrates how soft sensors leverage easily measurable signals to estimate otherwise unobservable process variables, a concept directly transferable to fermentation monitoring.

2.4 Performance Evaluation Metrics for Soft Sensors

Accuracy metrics are central to evaluating the predictive performance of soft sensors in fermentation modelling. Among the most consistently used are the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and the Coefficient of Determination (R^2). These metrics are typically used in combination to provide a multidimensional view of model performance. Kwon et al. (2024) provide formal expressions for RMSE, MAE, and R^2, describing RMSE as an

aggregate error magnitude, MAE as the mean difference between predicted and observed values, and R^2 as the proportion of variance explained by the model. Similarly, Fisher et al. (2022) emphasizes that the most statistically reliable soft sensor is one that "has an R^2 closest to 1 while minimizing RMSE, MAE and MAPE," reflecting standard practice in evaluating fermentation model outputs. Zhu et al. (2020) apply these metrics in a case study on penicillin fermentation, reporting RMSE and MAE values of 0.1028 and 0.067, respectively, for cell concentration prediction using a Particle Swarm Optimization Least Square Support Vector Machine (PSO-LS-SVM) soft sensor model. These examples underscore the practical role of these metrics not only in model selection but also in benchmarking across architectures and experimental conditions. Robustness and generalization are critical performance dimensions in fermentation soft sensor modelling. Zhu et al. (2020) note that "the values of the kernel parameters and penalty parameters of LS-SVM play a significant role in the generalization ability and accuracy of the model," with poor parameter tuning resulting in "over-fitting and poor generalization ability". To address this, they used a Particle Swarm Optimization (PSO) strategy to tune hyperparameters of LS-SVM for a penicillin fermentation task, which enhanced generalization on validation data after k-fold cross-validation. Kwon et al. (2024) further demonstrated the importance of generalization using Leave-One-Out Cross-Validation (LOOCV). In their work on synthetic time-series for ethanol fermentation, the authors observed that models trained on raw experimental data exhibited significant fluctuation in performance, attributed to overfitting specific batches. By introducing synthetic augmentation through Variational Autoencoders (VAE), they generated datasets that "facilitated better generalization of the model, thereby reducing the risk of overfitting".

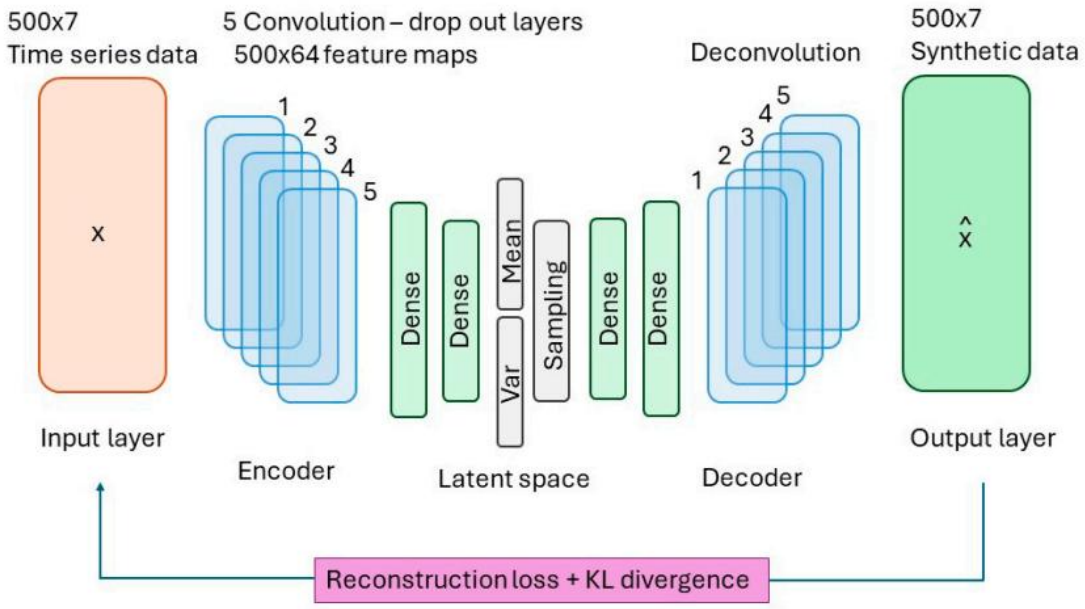


Figure 2—9: Visual representation of generation of synthetic data

adapted from (Kwon et al., 2024)

The effectiveness of these strategies was evidenced by consistent improvements in test performance, RMSE in testing dropped from 0.3326 to 0.02948 when moving from the original dataset to the Augmented100 variant, while R^2 increased from -2.37 to 0.96. Lastly, Zhu et al. (2020) emphasizes that robust evaluation frameworks, such as separating training, validation, and testing datasets, are essential to objectively estimate a soft sensor's true predictive capability, especially in multi-batch fermentation data settings where time-series dependencies may bias results if not handled carefully. Murugan (2021) addresses the challenge noise and error sensitivity in the context of cellulase production, where real-time estimation of biomass is confounded by both process and measurement noise. The author incorporated a Nonlinear Autoregressive model with exogenous input (NARX) structure, noting that this approach was “less sensitive to noise and hence can be applied to process control systems. To formalize the model’s noise resilience, Murugan (2021) specified the process noise covariance matrix and the measurement error covariance, reflecting the non-uniform error

sensitivity across state variables. The model's robustness was further reflected in its Mean Square Error (MSE) values under varying load conditions, indicating that the NARX structure could stabilize output predictions despite sharp fluctuations in substrate concentrations. Murugan (2021) notes that conventional feedforward neural networks showed higher error variance, making them less suited for online prediction under variable feed and agitation profiles. Kwon et al. (2024) addressed noise indirectly through synthetic data augmentation, which smoothed performance across batches by populating the training set with variability-informed samples. While this improved test RMSE significantly, it did not explicitly model error covariance, a distinction from Murugan's (2021) embedded stochastic model formulation. These findings reinforce the view that noise-handling can be approached through two complementary paths. Zhu et al. (2020) explicitly acknowledge that performance evaluation of soft sensors remains an "open question" due to a lack of consensus on which metrics or procedures are most appropriate for different types of models. The authors emphasize that numerical indicators such as RMSE or MSE are widely used, but they often fail to account for operational variability or noise effects in complex biochemical processes. Zhu et al. (2020) also note that while a five-stage development pipeline is often proposed (data collection, variable selection, model training, validation, and maintenance), in practice, model maintenance and real-world update cycles are rarely addressed in published research. Kwon et al. (2024) add that the interpretability of data-driven models remains a bottleneck. While models trained with synthetic augmentation perform better on test sets, they often offer limited insights into mechanistic behaviour or root causes of prediction errors, particularly when dealing with unexpected microbial behaviour or batch drift. Zhu et al. (2020) critique the trend that the field lacks benchmark datasets and consistent baseline models against which new soft sensors can be compared noting that "performance evaluation is highly related to the selection of the soft sensor type" and the specific conditions under which models are trained and tested. In their review, soft sensors are broadly classified into three categories: mechanism-based, data-driven, and hybrid models.

2.5 Challenges & Limitations in Soft Sensor Deployment

Brunner et al. (2021) defines sensor faults as “deviations of the observed sensor reading from the true value,” categorizing them as abrupt (stepwise) or incipient (driftwise), with manifestations including bias, precision degradation, and complete failure. The authors emphasize that if faults are present during soft sensor training, “these deviations may be reflected in the model coefficient and the prediction” thereby distorting all future inferences. This issue becomes more severe during real-time prediction. Faults in input variables directly corrupt output estimates, potentially leading to poor decisions in downstream control (Brunner et al., 2021). Brunner et al. (2021) go further to describe four strategies for sensor fault detection; Hardware Redundancy (using duplicate sensors to cross-check values), Symptom Signal Methods (indirect markers like off-gas or pH to validate core readings), Variable Contribution in Multivariate Statistical Process Control (MSPC), and Pattern Recognition (machine learning classifiers trained to detect faults).

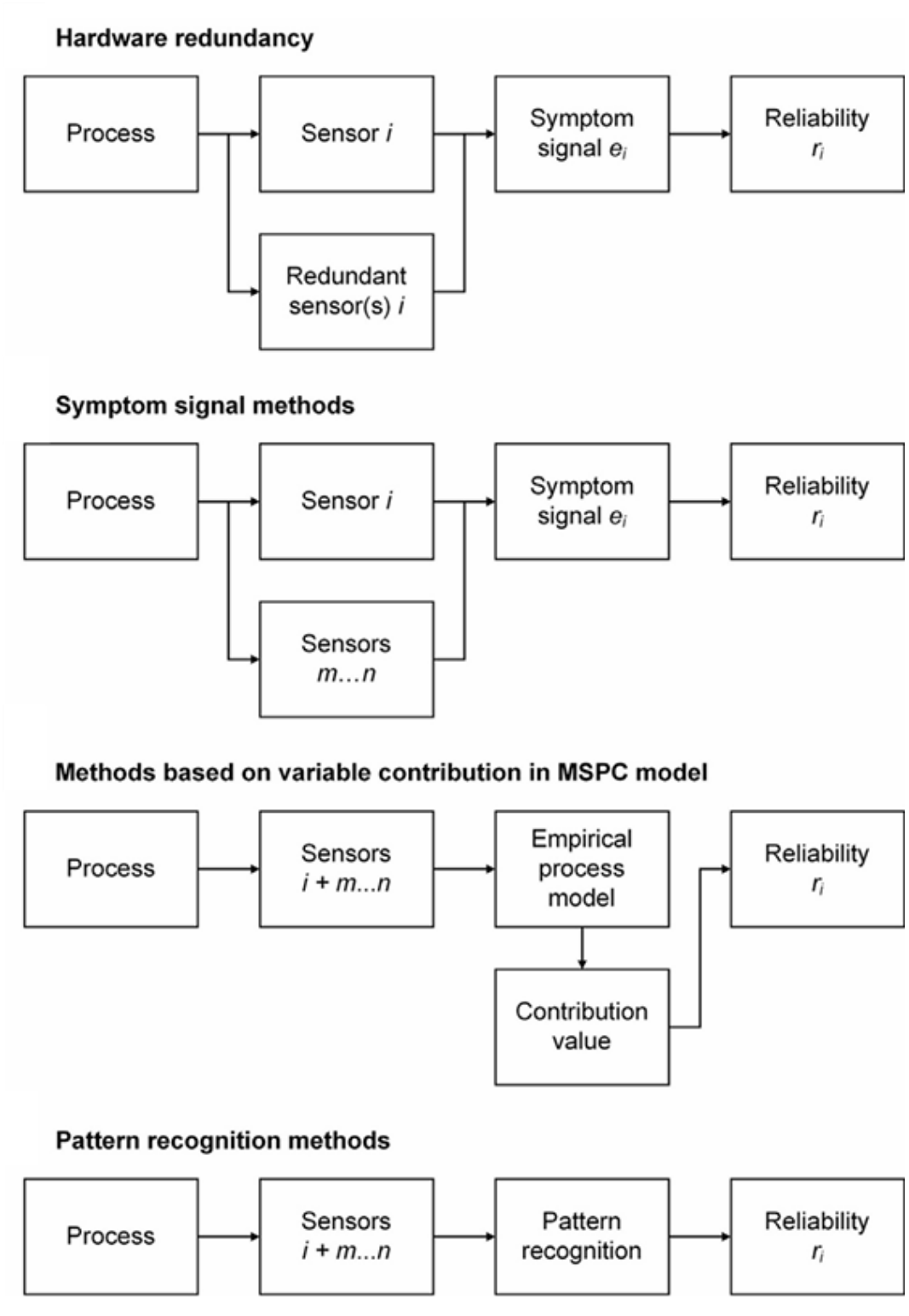


Figure 2—10: Sensor fault types and detection strategies

adapted from **(Brunner et al., 2021)**

Metcalfe et al. (2025) demonstrate that even after training on 100 batches of historical fermentation data, their Long Short-Term Memory (LSTM) based soft sensor exhibited significant performance degradation on future batches, prompting automated retraining within their proposed MLOps framework. The system applied Population Stability Index (PSI) thresholds to detect batch-level drift and trigger updates to restore predictive accuracy in real-time. Goldrick et al. (2019) working with the Industrial Penicillin Fermentation Simulation (IndPenSim) industrial simulator, describe that many control variables in large-scale bioreactors still rely on manual intervention or operator overrides, particularly for substrate feed and antifoam. This hybrid control mode can result in temporal misalignment between soft sensor outputs and decision-making processes.

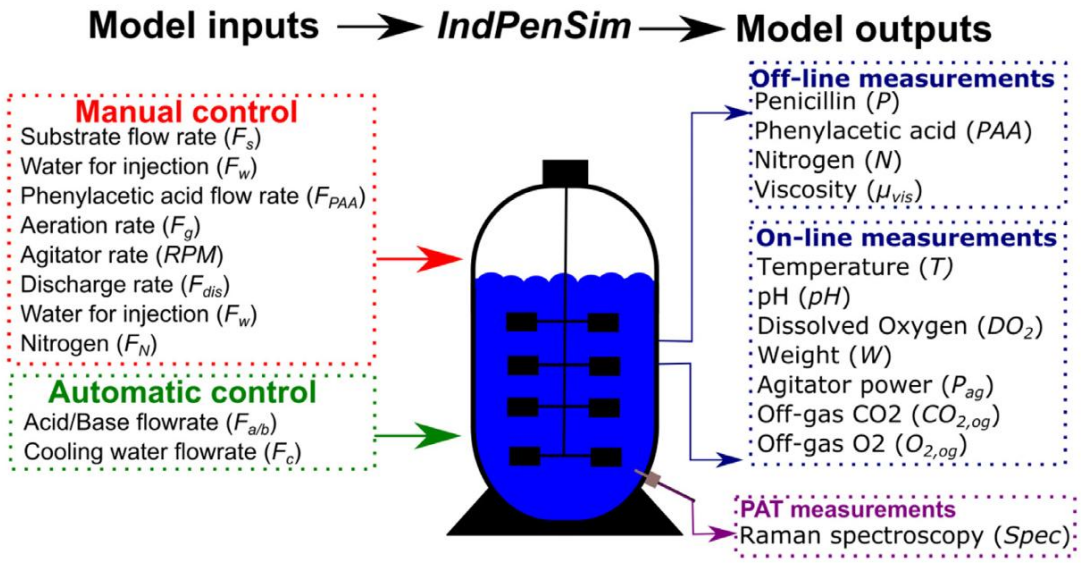


Figure 2—11: Hybrid control realities in industrial fermentation

adapted from (Goldrick et al., 2019)

Figure 2—11 summarizes the interplay of automated (PID-controlled) and manual (operator-dependent) control strategies in IndPenSim, capturing the fragmented landscape of industrial soft sensor integration.

Clearly stated by Metcalfe et al. (2025), “Some of these variables can be measured continuously on frequent time intervals (real-time/online) by fermentation equipment sensors and actuators, while others can only be measured offline due to technological limitations,” and this disjunction poses significant constraints on building effective predictive models). Even when inline and at-line measurement strategies are employed, the authors note that the lack of devices for specific molecules of interest continues to hinder the broader application of Process Analytical Technology (PAT) frameworks. The challenge is further exacerbated by variable omission during model development, either due to unavailable sensors or a lack of standardized protocols for selecting relevant inputs. Zhu et al. (2020) emphasizes that many soft sensor applications rely on expert judgement to select variables, and that improper or excessive input inclusion can increase overfitting risk while complicating real-time implementation. Validation of soft sensors in fermentation requires more than predictive accuracy, it must account for sensor performance, measurement frequency, and computational feasibility. Brunner et al. (2021) note that low analyte concentrations and measurement drift can limit model reliability, especially in mammalian systems. They also explain that faster microbial processes demand higher measurement frequencies per the Nyquist–Shannon theorem, which, combined with intensive preprocessing, may exceed real-time computational limits. Luttmann et al. (2012) stress that soft sensor operability depends heavily on proper handling of drift, outliers, and missing values, which are frequently encountered in real-world bioprocesses. They suggest that only when these issues are adequately addressed can soft sensors deliver consistent performance in production environments, particularly in areas like raw material assessment and seed culture quality, both of which have significant impact on batch variability and downstream yield. Brunner et al. (2021) further reinforce the gap, noting that regulatory frameworks such as GAMP-5 and EU GMP Annex 11 impose rigorous expectations on any model used in automated control, yet few soft sensor deployments meet these standards. Goldrick et al. (2019) point out that retraining infrastructure, version control, and lifecycle monitoring are rarely integrated into soft sensor architectures. Their IndPenSim platform illustrates how prediction accuracy degrades under realistic

variability, emphasizing the need for ongoing validation pipelines and adaptive retraining triggers.

2.6 MLOps & Real Time Deployment

Metcalfe et al. (2025) emphasizes the necessity for end-to-end automation to achieve practical, real-time deployment. They propose that deploying a soft sensor in industrial biotechnology (IB) production "requires more than just a classical ML model with an unseen test set to evaluate the quality prediction of the model; it demands a comprehensive architecture that is deployable within an IB setting." This architecture is achieved through Machine Learning Operations (MLOps**)**, a set of tools and methodologies designed to handle the full model lifecycle, from development and validation to retraining and redeployment. The pipeline emphasizes data ingestion, drift detection, and model monitoring, allowing bioprocess soft sensors to operate under conditions of variability in substrates, process disturbances, and production drift. While traditional soft sensor development ends at validation with historical data, MLOps introduces a continuous lifecycle pipeline. As described by Metcalfe et al. (2025) their architecture consists of both a virtual environment (used for development, model tuning, and simulation) and a real environment (fed by live process data from the industrial plant). The trained model is first tested offline in the virtual environment using historical data, then promoted to the real-time layer upon satisfying performance benchmarks. Once deployed, the model output is compared with new incoming data, enabling concept drift detection via statistical indicators such as the Population Stability Index (PSI). When PSI thresholds are breached, signalling a meaningful change in the data distribution, the pipeline triggers automated retraining procedures. This ensures the soft sensor remains accurate even as process conditions, media lots, or microbial behaviour shift across production campaigns.

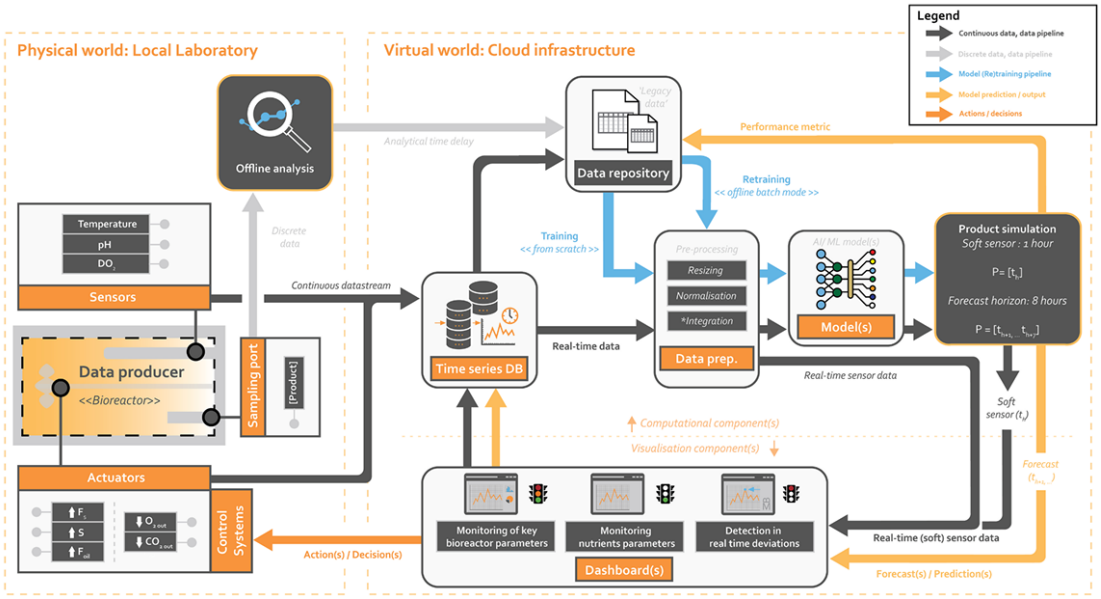


Figure 2—12: MLOps deployment pipeline for fermentation soft sensors adapted from (Metcalfe et al., 2025)

In addition to model retraining, Metcalfe et al. (2025) emphasizes the importance of model version control and traceability as central to production-scale machine learning in industrial biotechnology. Their pipeline logs model metadata, hyperparameters, drift status, and retraining triggers to ensure that each version of the soft sensor is both reproducible and auditable. To move from experimental modelling to industrial deployment, soft sensors must be embedded within existing process control infrastructure. Goldrick et al. (2019) illustrate this integration using their IndPenSim simulation of a penicillin bioreactor. In their framework, soft sensor outputs feed into higher-level supervisory systems. Metcalfe et al. (2025) extend this view by discussing the need for dedicated model-serving layers, isolated from critical control systems. Their architecture employs containerized deployment of LSTM soft sensors, served through Application Programming Interfaces (APIs) that pull pre-processed features from a historian and return predictions with sub-second latency. Crucially, this structure ensures that model failures do not interfere with plant safety, and versioned outputs can be traced back during audits. While the soft sensor developed in this

research was trained and validated using historical, time-resolved data, its structure and performance analysis were guided by the operational realities of real-time deployment. The model was designed to be modular, with a fixed input-output interface and pre-processed features that reflect real-world sampling rates and sensor noise profiles, making it compatible with modern model-serving infrastructure.

2.7 Overview of Regression Models Applied in This Study

In this study, several regression-based machine learning models were employed to develop the soft sensor, all of which are commonly available within MATLAB's Regression Learner App. While the following regression algorithms are outlined based on general formulations by Rao et al. (2023), their application in soft sensor modelling for fermentation and bioprocess monitoring has been extensively reported (Kadlec et al., 2009; López Pérez et al., 2013; Luttmann et al., 2012; Zhu et al., 2020). Although the models were applied through MATLAB's graphical interface, their underlying mathematical foundations are summarized here to provide context for their selection and relevance to soft sensor development.

Linear Regression

Linear regression is interpreted here as a method that models the relationship between a dependent variable and one or more independent variables using a linear combination of inputs. Rao et al. (2023) provide the general form:

$$y = b_0 + b_1 x_1 + \cdots + b_n x_n \tag{2-2}$$

In this equation, y is the dependent output variable, x_i represents input feature, b_0 is the intercept, and b_i denotes the coefficient associated with each predictor. The coefficients are estimated using the closed-form matrix solution:

$$y = \beta^* = H^{-1}T \tag{2-3}$$

This expression is referred to in the text as providing "the smallest norm of the mean squared error," which indicates the model is fitted by minimizing prediction errors in the least squares (Rao et al., 2023).

Linear Interactions Regression

Linear interaction models are interpreted as an extension of simple linear regression that includes terms to capture how two variables may interact multiplicatively in their effect on the response. Rao et al. (2023) illustrate such a model using:

$$y = \beta_0 + \beta_1 X_1 + \beta_3 X_1 X_2 + \epsilon \tag{2-4}$$

Here, y is the predicted output, X_1 and X_2 are input variables, β_0 is the intercept, β_1 and β_3 are regression coefficients, and ε is the error term. The inclusion of the product term X_1X_2 allows the model to capture combined effects of features.

Robust Linear Regression

Robust regression is interpreted as a method for fitting linear models that are less sensitive to outliers in the data. This interpretation is based on Rao et al.'s (2023) expression of the residual error formulation:

$$y_i = x_i\beta + \varepsilon_i \tag{2-5}$$

$$\varepsilon_i = y_i - x_i\beta_i \tag{2-6}$$

In equation (2-5) and equation (2-6), y_i represents the observed output for the i^{th} sample, x_i is the vector of input features for that sample, β is the coefficient vector, and ε_i is the residual error associated with that observation.

Stepwise Linear Regression

Stepwise regression is interpreted here as a variable selection method that sequentially adds or removes predictors based on statistical significance or contribution to model fit. Rao et al. (2023) present a standardized coefficient calculation used in this process:

$$b_{j,std} = b_j\left(\frac{S_{xj}}{S_y}\right) \tag{2-7}$$

In this expression, b_j is the raw coefficient of predictor x_j, S_{xj} is the standard deviation of predictor xj, and S_y is the standard deviation of the response variable. Standardized coefficients are used to compare the relative influence of each feature.

Regression Trees

Regression trees are interpreted as models that recursively partition the feature space and assign a constant prediction to each region. This is supported by Rao et al. (2023), who defines the output of a terminal node as:

$$y_t = \frac{1}{n}\sum Y_i \tag{2-8}$$

where y_t is the predicted output for a terminal node containing n samples, and Y_i are the actual observed outputs for those samples. The tree splits are determined by minimizing an impurity function:

$$I(Q,\gamma) = \frac{n_l}{n}S\big(Q_l(\gamma)\big) + \frac{n_r}{n}S(Q_r(\gamma)) \tag{2-9}$$

Here, Q is the parent node, $Q_l(\gamma)$ and $Q_r(\gamma)$ are the resulting left and right child nodes after a split on parameter γ, $S(\cdot)$ is a measure of impurity, and n_l and n_r are the number of samples in the left and right nodes, respectively.

Support Vector Machine (SVM)

Support Vector Regression (SVR) is interpreted as a method that fits a linear or nonlinear function within a margin of tolerance ϵ, while minimizing model complexity (Zhu et al., 2020). Rao et al. (2023) defines the linear SVR model as:

$$f(x) = x^T\beta + b \tag{2-10}$$

The epsilon-insensitive loss function used is:

$$L_\epsilon = \begin{cases} 0, & |y - f(x)| \le \epsilon \\ |y - f(x)| - \epsilon, & otherwise \end{cases} \tag{2-11}$$

In equation (2-10) and equation (2-11), x is the input vector, β is the coefficient vector, b is the bias term, and L_ϵ quantifies the penalty when predictions fall outside the epsilon margin. When using kernel functions for nonlinear regression, the prediction is given by:

$$f(x) = \sum (a_n - a_n^*) G(x_n, x) \tag{2-12}$$

where a_n and a_n^* are Lagrange multipliers, $G(x_n, x)$ is the kernel function, and x_n are training data points.

Ensemble Methods

Ensemble models are interpreted as aggregations of multiple weak learners, usually trees, combined to improve generalization.

(a) Bagged Trees

The ensemble prediction using bagging is defined as:

$$K(x) = \frac{1}{K}\sum_{t=1}^{K} K_k(x) \tag{2-13}$$

In this expression $K_k(x)$ is the prediction of the k^{th} tree, and K is the total number of trees in the ensemble.

(b) Boosted Trees

Boosted ensembles are defined by:

$$Y(x) = sign\sum_{t=1}^{T} \alpha_t h_t(x) \tag{2-14}$$

Here, $h_t(x)$ is the t^{th} weak learner, α_t is its associated weight, and T is the total number of learners.

Neural Network

The Extreme Learning Machines (ELMs) are interpreted as single-hidden-layer neural networks with randomly assigned hidden weights and biases, where only output weights are learned. Rao et al. (2023) defines the model as

$$f_1(x) = \sum_{i=1}^{l} \beta_i G(a_i, b_i, x) \tag{2-15}$$

The activation function used is:

$$G(a, b, x) = \frac{1}{1 + \exp(-a \cdot x + b)} \quad (2\text{-}16)$$

In these equations, β_i is the output weight of the i^{th} hidden node, $G(\cdot)$ is the sigmoid activation function, and a_i and b_i are input weights and biases. In matrix form, the network is trained using:

$$H\beta = T \rightarrow \beta^* = H^{-1}T \quad (2\text{-}17)$$

Where H is the hidden layer output matrix, β is the output weight vector, and T is the target output vector.

Gaussian Process Regression

Gaussian Process Regression is interpreted as a non-parametric method that models outputs using probability distributions defined by kernel functions. Rao et al. (2023) provide several kernel formulations, including:

(a) Squared Exponential

$$k(x_i, x_j) = \sigma_f^2 \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma_l^2}\right) \quad (2\text{-}18)$$

(b) Matern 5/2

$$k(x_i, x_j) = \sigma_f^2(1 + \sqrt{3}r)\exp(-\sqrt{3}r \quad (2\text{-}19)$$

(c) Exponential

$$k(x_i, x_j) = \sigma_f^2 \exp\left(-\frac{r}{\sigma_l}\right) \quad (2\text{-}20)$$

In these equations, x_j are input vectors, r is their Euclidean distance, σ_f^2 is the signal variance, and σ_l is the kernel length scale. These kernel functions define the smoothness and flexibility of the GPR model (Kadlec et al., 2009; Rao et al., 2023; Zhu et al., 2020). Collectively, these regression models form the methodological basis for data-driven soft sensors in fermentation, where similar frameworks have been applied to estimate key variables such as biomass, substrate and ethanol concentrations from measurable parameters like temperature and dissolved oxygen (López Pérez et al., 2013; Luttmann et al., 2012; Zhu et al., 2020).

3. EXPERIMENTAL SETUP & METHODOLOGY

Developing a soft sensor necessitates a well-designed system capable of generating reliable data, which can then be processed and analysed using machine learning techniques. In this research, the setup involves configuring a large-scale bioreactor system at the Wits Microbrewery, equipped with a combination of cost-effective instruments and robust sensors. This system will provide the necessary data to train and evaluate the soft sensor, enabling the assessment of its feasibility and accuracy.

3.1 Experimental Setup

The fermentation experiments were conducted using a 100 L fed-batch bioreactor system developed for ethanol production from sugar feedstock by *Saccharomyces cerevisiae*. The system was designed to control key process parameters, including temperature and substrate feeding rate, while continuously monitoring density and other critical variables to facilitate real-time data collection. 10 experiments were conducted.

3.1.1 Fermentation System

The bioreactor vessel was a 100 L jacketed cylindroconical fermenter, fabricated from stainless steel by Falcon Engineering. The fermenter was covered with a stainless-steel plate to prevent contamination while allowing gas escape. The fermentation medium consisted of sucrose, yeast extract, and water, with varying concentrations of yeast. The initial fermentation conditions were controlled by adjusting substrate feeding rates across different experimental runs.

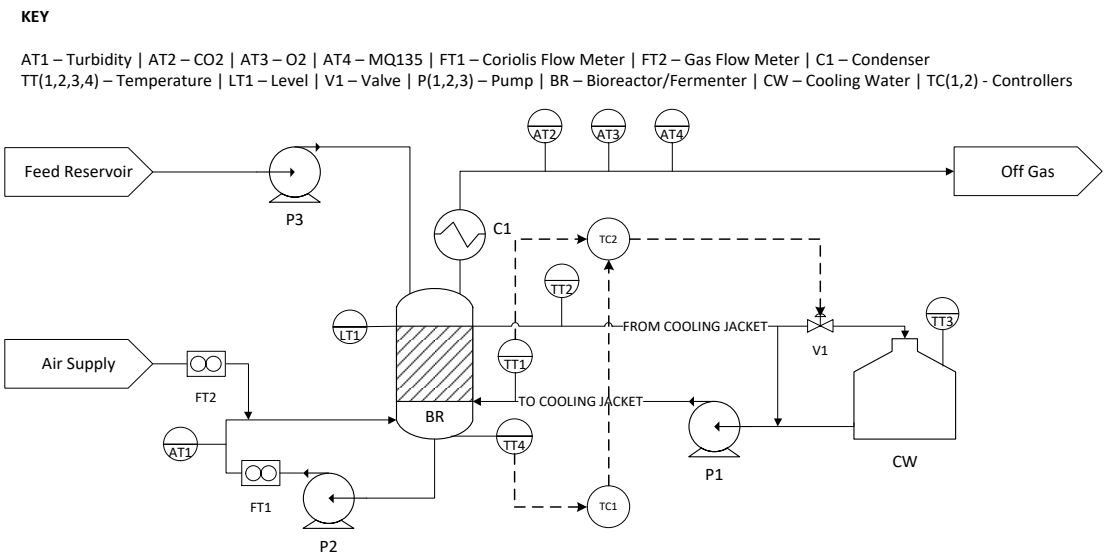


Figure 3—1: Ethanol fermentation setup at the Wits Microbrewery

Temperature regulation is critical in ethanol fermentation, influencing yeast metabolism and ethanol yield. The bioreactor temperature was controlled using a cascade control system implemented on a PLC. The primary controller (TC1) functioned as an integral controller, ensuring steady-state temperature control. The secondary controller (TC2) operated as a proportional-only controller, providing rapid response to changes. Temperature was monitored by two probes: TT4 (PT100 probe) for precise fermenter liquid temperature monitoring and TT1 (DS18B20 probe) for measuring cooling jacket temperature. A Lowara CEA 70/3 centrifugal pump circulated water through the cooling jacket, with flow regulated by a Siemens Micromaster 440 variable frequency drive (VFD). The bioreactor employed a fed-batch feeding strategy. The Grothen G328 peristaltic pump (P3/4) was used for precise feed control. The pump was regulated by an Arduino Uno via an L298N H-Bridge driver. Different feed rate profiles were implemented across experiments such as constant flow, adaptive feeding (feedback-controlled based on specific gravity setpoints), and

manual interventions in cases of pump failures or experimental variations. The initial conditions and feed rates for each of the 10 experiments are shown below.

Table 3—1: Initial experiment conditions

Experiment	Sugar (kg)	Yeast extract (g)	Water (kg)	Yeast (g)
1	5	100	40	100
2	5	100	40	100
3	5	100	40	100
4	5	100	40	100
5	5	100	40	200
6	5	200	40	150
7	5	300	40	300
8	5	400	40	200
9	5	200	40	200
10	15	100	50	100

Table 3—2: Experiment feed rate conditions

Experiment	Reaction type
1	Constant feed at 0.985 kg/h
2	Constant feed at 0.713 kg/h
3	Constant feed at 0.283 kg/h
4	Constant feed at 0.385 kg/h
5	Varied feed
6	Varied feed
7	Constant feed at 3.333 kg/h
8	Feed added to maintain SG at setpoint
9	Feed added to maintain sugar concentration at setpoint
10	Batch reaction with water addition

3.1.2 Instrumentation & Data Collection

Accurate real-time data collection is essential for monitoring the fermentation process and developing a soft sensor for ethanol concentration estimation. The instrumentation setup for this study integrates temperature sensors, density measurement devices, and automated data acquisition systems, ensuring reliable monitoring and control of key fermentation variables. Although there were many instruments and information that could be drawn from the experimental setup, only the following measurements were extracted and used during each of the 10 experiments for this research:

- Fermenter Jacket Temperature
- Fermenter Temperature
- Ambient Temperature
- Specific Gravity of the Fermenter Contents

The temperature measurements were obtained via temperature sensors throughout fermentation. The specific gravity was measured using an *iSpindle*™ electronic hydrometer, which is a type of tilt hydrometer which works on the principle of buoyancy (iSpindle, 2025).



Figure 3—2: iSpindle™ electronic hydrometer

The measured tilt creates an angle between the instrument and the fermenting liquid over time. The tilt is then used to infer gravity based on the following calibration which can be found on the calibration manual of the product:

$$SG = a_1(tilt)^3 + a_2(tilt)^2 + a_3(tilt) + C \quad (3\text{-}1)$$

Where a_1, a_2, a_3, and C, are constants dependent on the calibration. The control of the bioreactor/fermenter was performed by PLC and then collected and stored remotely using the ThingSpeak service. Data was then imported by excel for use in this research.

3.2 Soft Sensor Development

The soft sensor developed in this study required reliable reference values for ethanol concentration across time to serve as targets during model training. Since direct ethanol concentration measurements from HPLC are time-consuming and expensive, a validated empirical approach was chosen instead. For each of the 10 fermentation experiments, three temperature variables, jacket, fermenter, and ambient were recorded along with specific gravity (SG) measurements using the *iSpindle*™ digital hydrometer. All data were captured with time recorded. To estimate ethanol concentration from SG readings, the method described by (Cutaia et al., 2009) was adopted. Their work demonstrated the use of polynomial regression equations derived from a large set of pilot and commercial brewery fermentations, offering a high level of confidence in predictive performance. This was the premise used to model ethanol concentration which was validated against HPLC measurements and ultimately used to train the soft sensor.

3.2.1 Data Preparation & Modelling

SG alone does not give direct ethanol content, a multi-step modelling procedure was adopted using established brewing equations, as discussed below. SG was first converted into Apparent Extract (AE) in degrees Plato (°P) using the 8th-degree polynomial regression derived by Cutaia et al. (2009). This model offers very high accuracy (R^2 of 0.99) across brewing-relevant ranges.

$$\begin{aligned} AE = {} & 2.569 \times 10^2 * (SG-1) + 6.7126 \times 10^0 * (SG-1)^2 \\ & - 1.4482 \times 10^4 * (SG-1)^3 + 5.1758 \times 10^5 * (SG-1)^4 \\ & - 1.0746 \times 10^7 * (SG-1)^5 + 1.3011 \times 10^8 * (SG-1)^6 \\ & - 8.5079 \times 10^8 * (SG-1)^7 + 2.3231 \times 10^9 * (SG-1)^8 \end{aligned} \quad (3\text{-}2)$$

Using the AE prediction above, the alcohol weight by weight (mass fraction) $A_{w/w}$ was obtained using the empirical quadratic model from Cutaia et al. (2009) was used shown below.

$$A_{W/W} = 0.38726 * (OE - AE) + 0.00307 * (OE - AE)^2 \quad (3\text{-}3)$$

The Original Extract (OE) that was used in equation (3-3) was obtained by knowing the number of solids present in the solution before the alcoholic fermentation commences. Since the amount of sugar added per fermentation at the beginning and during fermentation is known, the following calculation was used to calculate OE for use in equation (3-3).

$$OE = \frac{Mass\ Sugar\ in\ Vessel + Mass\ Sugar\ Added\ to\ Vessel}{Total\ Mass\ of\ Solution\ in\ Vessel} \times 100 \quad (3\text{-}4)$$

Equation (3-4) provided the OE units of degrees Plato. There were other linear equations used to predict the alcohol mass fraction but the reason equation (3-3) was used was because the other equations came from literature over 100 years old with no real clear evidence of how the relationship was formed. In addition, the other equations rely on many variables that many companies, organizations, brewers and engineers may not have access to like Real Extract (RE). The last step in utilizing the model now that the alcohol by weight is known was to determine the ethanol concentration in g/L. Using the SG and alcohol mass fraction the formula below was derived and used.

$$Ethanol\ \left(\frac{g}{L}\right) = \frac{A_{(w/w)}}{100} * SG * \rho_{water} = A_{(W/W)} * SG * 10 \quad (3\text{-}5)$$

This equation assumed the density of water is 1000 g/L.

3.2.2 Validation with HPLC

To assess the accuracy of the soft sensor ethanol model, predicted ethanol concentrations were compared to experimental values obtained via high-performance liquid chromatography (HPLC). Validation was performed across nine of the ten experiments (excluding EXP 4 due to missing HPLC data). Predicted ethanol values were available as time-series data with values available approximately every minute. In contrast, HPLC measurements were taken at irregular time points. The irregular timing of HPLC samples resulted primarily from practical and cost considerations, as each analysis required manual sampling and laboratory processing. Sampling was therefore conducted opportunistically at key fermentation stages, such as mid-exponential and late phases to validate model predictions across representative process conditions.

Table 3—3: Variable update frequency during live fermentations

Variable	**Update Frequency**
Temperature	10 seconds
Specific Gravity	10 minutes
HPLC Samples	Irregular

To enable direct comparison, predicted ethanol values were linearly interpolated at the HPLC time points using the MATLAB function interp1. This method estimates values between two known points using the linear interpolation formula:

$$y = y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) \tag{3-6}$$

The formula works by knowing two points, (x_1,y_1) and (x_2,y_2), and estimates y at some value x where x is between x_1 and x_2. In practical applications, it is common for time-series data to be recorded at irregular intervals or contain missing values due to various limitations during data collection. This poses challenges when consistent, time-aligned data is required for tasks such as model calibration, comparison, or prediction. As a result, interpolation techniques are frequently used to fill these gaps or align datasets with differing sampling rates, an essential step in data preprocessing for non-uniform or incomplete time-series (Lepot et al., 2017). This was an obvious choice to correlate the data and because of this several other comparison statistics could be produced on the modelled ethanol concentration from SG due to the same time stamps as measured by HPLC. Three performance metrics were calculated. The first being the Coefficient of Determination (R^2): Measures how well predicted values match observed data. The MATLAB function *corrcoef* was used to calculate the R^2 value.

$$R^2 = \left(\frac{cov(y_{true}, y_{pred})}{\sigma_{true}\sigma_{pred}}\right)^2 \tag{3-7}$$

Where Cov(y_{true}, y_{pred}) is the covariance between true and predicted value, σ_{true} is the standard deviation of true values, and σ_{pred} is the standard deviation of predicted values.

The second parameter used to showcase performance is the Root Mean Square Error (RMSE). Whilst the RMSE communicates the quality of the data only relative to other model performances it will be useful later on. It will still be beneficial in comparisons across data sets for this section of the dissertation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_{pred,i} - y_{true,i}\right)^2} \quad (3\text{-}8)$$

This measures the average magnitude of the error between predicted and observed values. It's in the same unit as the variable (g/L for ethanol concentration). To enable scale-independent comparison, the normalized RMSE (NRMSE) was also computed by dividing RMSE by the mean of the measured values:

$$NRMSE = \frac{RMSE}{\bar{y}_{true}} \quad (3\text{-}9)$$

These metrics were implemented using MATLAB, following standard practices for model performance evaluation in regression and process modelling.

3.3 Soft Sensor Methodical Approach

The development of the soft sensor involved several key steps: data preprocessing, feature engineering, and model training and testing. Temperature measurements from three sources (fermenter, jacket, and ambient), along with specific gravity readings and corresponding calculated ethanol concentrations, were used to create a dataset suitable for training a machine learning model to estimate fermentation performance. The modelling process was designed to reflect real-world industrial applications, with emphasis on generalizability and robustness. The approach was designed to leverage low-cost, non-invasive sensor data to estimate fermentation rate dynamics, making it potentially suitable for industrial applications where real-time ethanol quantification is desirable.

3.3.1 Rationale for Variable Selection

Temperature was selected as the primary process variable for soft sensor development because it provides a robust, non-invasive, and continuously available signal that reflects the metabolic heat released during ethanol fermentation. Although temperature has not been widely used in the literature as a direct estimator of ethanol concentration, it offers several distinctive advantages that align with the objectives of this research. First, temperature measurement is inherently low-cost and readily available in almost every fermentation system. Unlike other process variables, temperature can be monitored without direct contact with the liquid phase, minimizing contamination risk and making it less invasive than probes such as DO electrodes. This characteristic enhances sterility assurance and simplifies long-term operation. In addition, modern bioreactors typically include multiple temperature sensors, for the fermenter, jacket, and ambient environment, enabling multi-point thermal profiling that captures heat transfer dynamics. In contrast, DO, pH, and CO_2 measurements usually provide only a single reading per vessel, limiting spatial or operational resolution. Second, temperature sensing requires minimal calibration and maintenance compared with DO or pH probes, which are prone to fouling, drift, and sterilisation damage. Off-gas CO_2 analysis can deliver useful information on metabolic activity, but it requires additional gas-handling hardware and incurs significantly higher cost and maintenance demands. By contrast, temperature sensors are inexpensive, durable, and have a long record of industrial reliability, making them ideal for scalable, low-cost deployment. While pH is universally recognised as a key fermentation variable, it was not included as an input in this study because no suitable non-invasive, low-cost pH measurement solution was available. It is acknowledged that variables such as pH, DO, and CO_2 evolution can complement temperature-based soft sensing and may further enhance predictive accuracy in future studies. However, the intent of this research was to evaluate the feasibility of using a single, widely available variable that possesses a strong physical and mathematical relationship to ethanol formation through heat-transfer principles.

Temperature uniquely satisfies this requirement, as its link to metabolic heat generation allows direct thermodynamic modelling of ethanol production rate and concentration.

3.3.2 Data Preprocessing

Raw data from each of the ten fermentation experiments was imported and standardized in MATLAB. Temperature readings (fermenter, jacket, ambient) were originally recorded every 10 seconds, while specific gravity was logged every 10 minutes. To create a consistent, clean dataset, all variables were interpolated to hourly intervals using MATLAB's interp1 function with linear interpolation. This allowed alignment of all time series and ensured comparability across experiments. Smoothed versions of the temperature and ethanol time series were created using a moving average filter to reduce high-frequency noise while preserving dynamic trends. Differential (delta) values were computed using the first-order difference between each time step to highlight the rate of change of the process variables. This was applied to both temperature and ethanol values.

$$\Delta x_i = x_i - x_{i-1} \tag{3-10}$$

Equation (3-10) was applied to both the input variables (temperature) and output variable (ethanol), resulting in a dynamic representation of the fermentation process.

3.3.3 Feature Engineering & Thermodynamic Reasoning

The dynamics of heat transfer in bioreactors are critical to maintaining optimal conditions for fermentation, particularly for ethanol production. Heat balance equations are fundamental for understanding and managing the energy flows within the bioreactor system. The general heat balance equation is expressed as follows:

$$Q_{in} - Q_{out} + Q_{gen} - Q_{loss} = \Delta Q_{acc} \quad (3\text{-}11)$$

Where Q_{in} is the heat supplied to the system (e.g., via a heating jacket or ambient conditions), Q_{out} is the heat removed (e.g., through cooling), Q_{gen} is the heat generated by metabolic or yeast activity, Q_{loss} is the heat lost to the environment, and ΔQ_{acc} is the change in heat stored in the system. This equation has been widely used for process monitoring and optimization in technical-scale fermentations as shown in (Türker, 2004). Metabolic heat generation during ethanol fermentation is proportional to the rate of ethanol production. This relationship can be derived from the stoichiometric and thermodynamic principles of glucose metabolism. The overall reaction produces ethanol and releases heat, with a reaction enthalpy ($\Delta H_{reaction}$). During ethanol fermentation, metabolic activity generates heat that correlates with the rate of ethanol production. This relationship can be modelled as:

$$Q_{gen} = k * R_{ethanol} \quad (3\text{-}12)$$

$$k = -\frac{\Delta H_{reaction}}{2} \quad (3\text{-}13)$$

The division by two in equation (3-13) reflects that one mole of glucose produces two moles of ethanol, so the reaction enthalpy is expressed per mole of ethanol formed. Where $R_{ethanol}$ is the rate of ethanol production, and k is a proportionality constant depending on the thermodynamic efficiency of the system. (Türker, 2004) confirmed this relation experimentally by calculating heat production rates in a 100 m^3 industrial fermenter based on the enthalpies of glucose, biomass, and ethanol. Experimental results validated that ethanol production could be reliably estimated from heat

generation rates. To apply this relationship practically, the use of temperature differences in the bioreactor offers a non-invasive method for real-time estimation. The following expression relates metabolic heat to measurable temperature gradients:

$$Q_{gen} = UA\Delta T \tag{3-14}$$

Where U is the overall heat transfer coefficient, A is the heat transfer area, and ΔT is the temperature difference between the reactor contents and the cooling medium. Substituting Equation (3-12) into the thermodynamic model provides:

$$R_{ethanol} = \frac{UA\Delta T}{\Delta H_{reaction}} \tag{3-15}$$

This approach eliminates the need for costly offline checks, enabling real-time process monitoring based solely on heat transfer and thermodynamic principles. On-line calorimetry has been shown to accurately correlate heat release with microbial activity, validating this approach for bioprocess control (Von Stockar et al., 1997). A more comprehensive model incorporates additional thermal contributions, including heat lost to the environment:

$$Q_{gen} = UA\left(T_{fermenter} - T_{cooling}\right) + U_{loss}A_{loss}(T_{fermenter} - T_{ambient}) \tag{3-16}$$

In this model, the first term represents the heat exchanged with the cooling system, and the second term accounts for environmental losses. Using this full energy balance, ethanol production rate and cumulative concentration can be estimated via:

$$R_{ethanol} = \frac{Q_{gen}}{\Delta H_{reaction}} \tag{3-17}$$

$$\Delta C_{ethanol} = \int_{t_o}^{t} R_{ethanol} dt \tag{3-18}$$

Therefore, the soft sensor was trained using four engineered input features:

- ΔT_{Jac}: Change in jacket temperature
- ΔT_{Fer}: Change in fermenter temperature
- ΔT_{Amb}: Change in ambient temperature
- T_{Load}: Estimated heat load

The estimated heat load was calculated as:

$$T_{Load,i} = \sum_{j=1}^{i} T_{Jac,j} \tag{3-19}$$

This cumulative value was introduced to approximate the total thermal energy delivered to the system via the jacket. Although actual heat flow data was unavailable, this surrogate feature was used as a proxy for metabolic heat, based on the thermodynamic principle that microbial activity correlates with heat release (Von Stockar et al., 1997). In industrial bioreactors, this would correspond to a more complete energy balance:

$$Q_{load} = \sum_{i=1}^{n} mc_p\left(T_{in,\iota} - \dot{T}_{out,\iota}\right) * \Delta t \tag{3-20}$$

3.3.4 Model Training & Testing

The soft sensor was trained using a supervised regression learning approach. The data split between training and testing data was as follows:

Table 3—4: Train and test data split

Train	**Test**
Exp 1 (constant feed)	Exp 5 (pump failure + large yeast addition)
Exp 2 (lower feed rate)	Exp 8 (feedback control - SG setpoint)
Exp 3 (even lower feed rate)	Exp 10 (batch + dilution)
Exp 4 (moderate feed rate)	
Exp 6 (increased yeast & extract)	
Exp 7 (constant feed, high yeast extract)	
Exp 9 (feedback control - sugar concentration)	

The division of experiments into training and testing sets was done strategically to ensure both diversity and realism in model evaluation. The training set (Experiments 1, 2, 3, 4, 6, 7, and 9) includes a broad range of fermentation conditions, such as varying feed rates, yeast concentrations, and yeast extract levels, providing the model with exposure to different process dynamics and metabolic behaviours. The testing set (Experiments 5, 8, and 10) was selected to simulate real-world application challenges.

These experiments include process disturbances (e.g., pump failures), feedback-controlled feeding strategies, and dilution phases, scenarios that are likely to occur in industrial settings. Testing the model against these conditions allows for a realistic assessment of its robustness and generalizability. This split also reduces the risk of overfitting. By excluding irregular and feedback-driven experiments from the training set, the model is prevented from simply memorizing patterns and is instead encouraged to learn generalizable relationships. This ensures it performs well not only on controlled laboratory conditions but also under variable or unexpected process dynamics. The structure for training and testing data was as follows:

$$Input\ Vector\ x_i = \begin{bmatrix} \Delta T_{Jac,i} \\ \Delta T_{Fer,i} \\ \Delta T_{Amb,i} \\ T_{Load,i} \end{bmatrix} \tag{3-21}$$

$$Output\ Vector\ y_i = [\Delta Eth_i] \tag{3-22}$$

Model training and evaluation were conducted using MATLAB's Regression Learner App. Multiple algorithms were explored, and the final model was selected based on performance on the test set. The model families explored in MATLAB were Linear Regression, Stepwise Linear Regression, Trees, SVM, Ensemble, GPR, NN, & Kernel. The mathematical formulations of all regression models used, as directly implemented in MATLAB's Regression Learner App can be viewed in the literature review.

4. RESULTS & DISCUSSION

This section presents the results obtained from the fermentation experiments, the development and evaluation of the ethanol soft sensor, and a critical analysis of the data. The discussion integrates experimental findings with model predictions, addressing key challenges in soft sensor development and the rationale behind model choices. The structure of this section follows a logical flow, beginning with the presentation of raw experimental data, progressing through model estimation and validation, and concluding with a comprehensive evaluation of the soft sensor's performance and subsequent optimization.

4.1 Fermentation Dynamics Across Varying Conditions

The dataset generated from the ten fermentation experiments provided a robust and diverse foundation for soft sensor development. Each fermentation trial involved the monitoring of jacket temperature (TJac), fermenter temperature (TFer), ambient temperature (TAmb), and specific gravity (SG) as shown in Figure 4—1. The variation in these profiles reflects the distinct operational strategies implemented across experiments, including differences in feed rate, yeast concentration, and deliberate process disturbances such as equipment malfunctions and adaptive control strategies. Other experiment profiles can be viewed in the appendix.

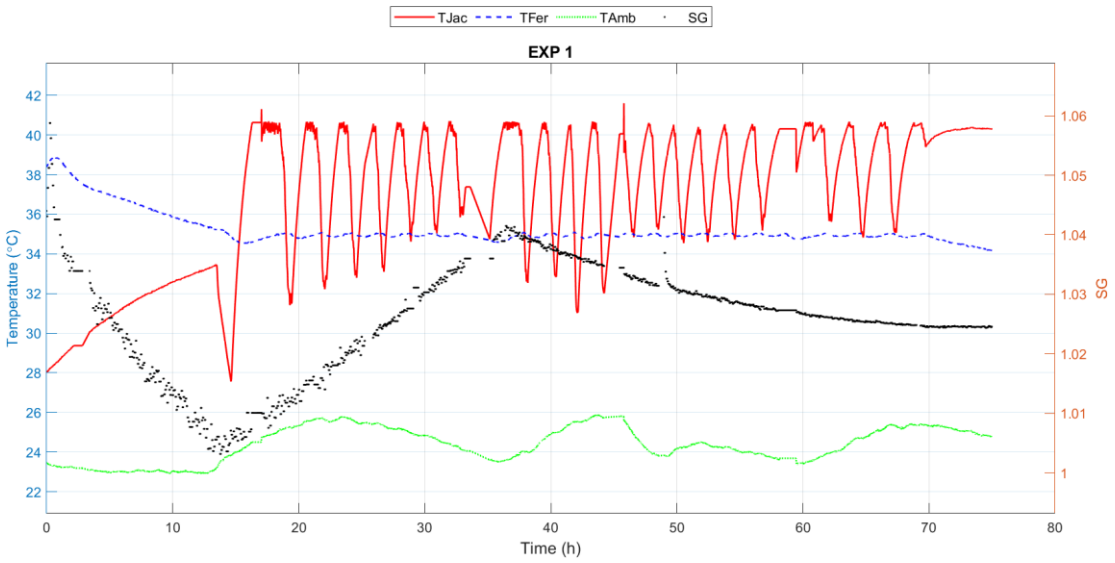


Figure 4—1: Experiment 1 time profile of temperature and specific gravity during fermentation

Experiments 1 to 4 were characterized by constant feed rates, allowing for direct comparison of fermentation kinetics under relatively stable conditions. Experiment 1, with the highest feed rate (0.985 kg/h), exhibited the most rapid decline in SG, indicating accelerated substrate conversion and ethanol formation. By contrast, experiments 2 and 3, which employed lower feed rates (0.713 and 0.283 kg/h, respectively), demonstrated slower fermentation kinetics, likely due to substrate limitation. Experiment 4, which had a moderate feed rate (0.385 kg/h), showed a fermentation profile closely resembling that of Experiment 1, albeit with marginally reduced efficiency. Experiment 5 introduced a significant deviation from expected behaviour. A malfunction in the peristaltic pump at approximately 27 hours resulted in the unregulated addition of the remaining feed volume. This sudden influx of sugar-rich solution caused a spike in SG, highlighting the sensitivity of the fermentation process to uncontrolled feed disturbances. In Experiments 5 to 7, the impact of yeast concentration was explored. Increased inoculation rates led to a noticeably faster initial SG decline, confirming that higher yeast concentrations enhance early fermentation kinetics. Nevertheless, the final SG values across these experiments converged to similar endpoints, suggesting that while initial activity is enhanced, the extent of

fermentation is still governed by substrate availability. Experiments 8 to 10 incorporated feedback-based adaptive control strategies. In Experiment 8, SG was maintained at a setpoint of 1.010 by toggling the feed pump on and off in response to measured SG values. The result was a more stable and predictable fermentation trajectory. In Experiment 9, sugar concentration was held at 50 g/L for the first phase, followed by the addition of both feed and approximately 20 kg of water. The water addition initially increased the SG due to the concentrated feed solution, followed by dilution effects. In Experiment 10, a batch-style fermentation was initially performed, with a subsequent water addition at around 38 hours. This phase-shift strategy was intended to maintain ethanol concentrations within the yeast's tolerance limits, thereby sustaining metabolic activity. Overall, the experimental data set captured a wide range of operational scenarios, from ideal batch reactions to dynamic, real-world process variability. The inclusion of both controlled and disturbed fermentations provides a rigorous foundation for soft sensor development, ensuring the trained models are both robust and generalizable across industrial conditions.

4.2 Verification of Ethanol Targets for Soft Sensor Training

To develop a soft sensor capable of estimating ethanol production using only temperature data, it was essential to establish a reliable target variable against which model performance could be evaluated. In this research, ethanol concentrations were estimated using a validated specific gravity (SG)-based approach proposed by Cutaia et al. (2009), as described in Section 3. This method uses SG readings to calculate apparent extract (AE), which is then combined with original extract (OE) to determine alcohol concentration. Prior to using these values as training targets for the soft sensor, it was necessary to verify their predictive accuracy against experimentally measured ethanol concentrations obtained via high-performance liquid chromatography (HPLC). Predicted ethanol concentrations from the SG model were compared with HPLC readings across nine of the ten fermentation experiments (Experiment 4 was excluded due to incomplete HPLC data). These comparisons were visualized using parity plots

for each experiment, illustrating the relationship between model-predicted ethanol concentrations and experimentally measured values. Each parity plot includes a 1:1 reference line to indicate perfect agreement, thereby highlighting the accuracy and deviations of the model across different fermentation conditions. As shown in Figure 4—2, the SG-based estimation model demonstrated strong agreement with HPLC measurements under a variety of fermentation conditions. Particularly high R^2 values were observed in Experiments 1, 3, 6, and 7, indicating a strong linear relationship between predicted and measured ethanol concentrations. While NRMSE values varied slightly across these experiments, they remained within acceptable bounds, confirming that the model effectively captured fermentation trends even under moderate thermal disturbances. Experiments 6 and 7 are especially noteworthy because they included unplanned temperature disturbances due to power failures, yet the model maintained high predictive accuracy. This suggests that the SG-ethanol relationship remained robust despite temporary deviations in process conditions.

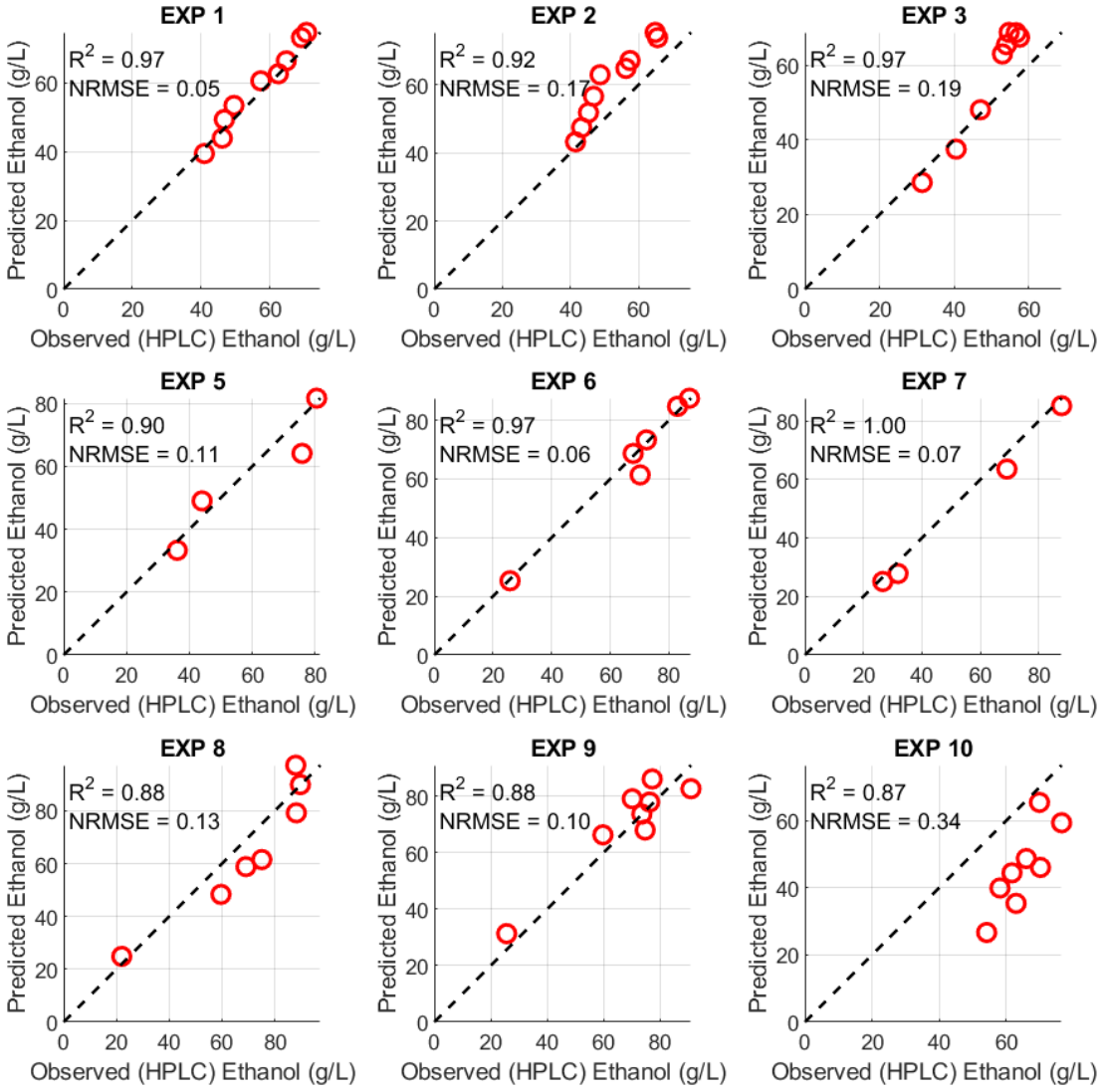


Figure 4—2: HPLC and predicted ethanol measurement parity plot for experiments 1-10

Experiments 2 and 5 yielded slightly lower performance, with R^2 values of 0.92 and 0.90 and NRMSE values of 0.17 and 0.11, respectively. In Experiment 2, a power disruption at approximately 24 hours may have altered yeast activity and fermentation dynamics, while in Experiment 5, a peristaltic pump failure caused an abrupt addition of concentrated feed solution, leading to a non-linear spike in SG. These operational anomalies contributed to reduced model performance, but the prediction error remained within acceptable limits. Experiments 8, 9, and 10, which involved adaptive control

strategies and batch-fed transitions, exhibited the largest discrepancies between predicted and measured ethanol concentrations. R^2 values for these experiments ranged from 0.87 to 0.88, and NRMSE values were higher, peaking at 0.34 in experiment 10. These elevated errors are attributable to the dynamic nature of feed and dilution strategies. In Experiment 9, for example, the concurrent addition of feed and water altered SG non-linearly. In Experiment 10, a significant water addition mid-process introduced a dilution effect that temporarily decoupled the SG signal from actual ethanol concentration, leading to prediction deviations. To assess the model's overall accuracy across all fermentation scenarios, a combined parity plot was generated by aggregating the data points from all experiments where both SG and HPLC data were available. This cumulative view enables evaluation of the global performance of the ethanol estimation model under both stable and disturbed conditions. Figure 4—3 presents the combined parity plot for all applicable experiments, yielding an overall coefficient of determination R^2 of 0.75 and an NRMSE of 0.17. While performance declined modestly under complex operational conditions, the model consistently captured the underlying ethanol production trends. The robustness of the model across varying feed strategies, yeast concentrations, and thermal fluctuations validates its use as a training target in the development of a soft sensor. In summary, the SG-based ethanol estimation model demonstrated sufficient predictive fidelity across a diverse range of fermentation profiles. Even under scenarios involving significant disturbances, the model provided a reliable approximation of ethanol concentration. This verification step confirms that the model can serve as an effective surrogate for direct ethanol measurement, enabling the training of data-driven soft sensors in scenarios where continuous analytical instrumentation is either unavailable or impractical.

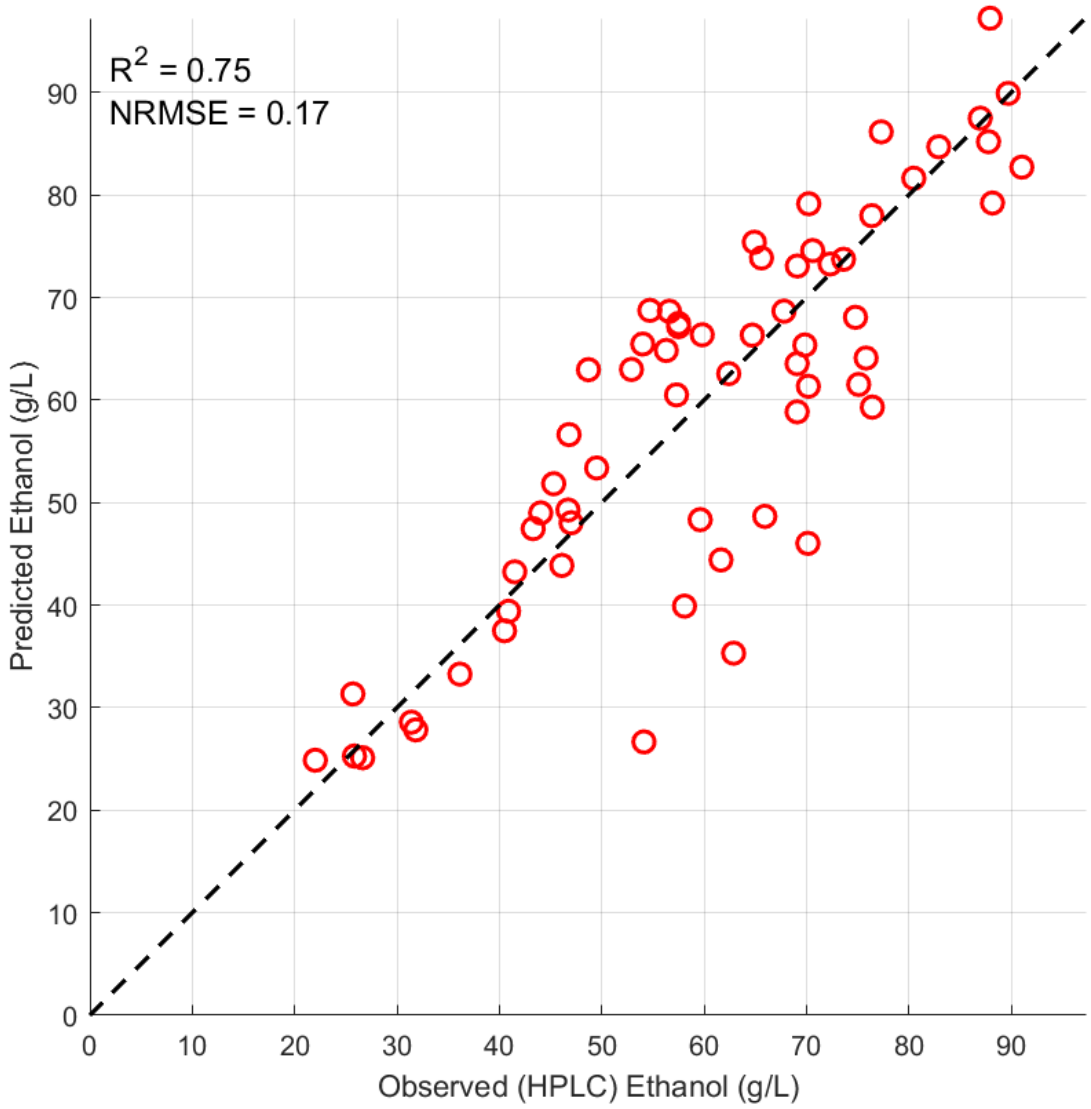


Figure 4—3: Overall parity plot for HPLC vs predicted ethanol

A critical limitation of this approach is that, while the model predicts ethanol levels accurately, it does not directly reflect yeast metabolic activity, which is a more fundamental indicator of fermentation progress and health. This is particularly important in real-world applications, where factors such as vitality, stress, and nutrient limitations may affect fermentation outcomes independently of ethanol concentration. Nonetheless, the results presented in this section confirm that the SG-derived ethanol

values are sufficiently accurate and consistent to be used as a training target in the development of a soft sensor. Establishing this accuracy is essential, as it forms the basis for the supervised learning process presented in subsequent sections, where temperature measurements will be used to estimate ethanol production in real time.

4.3 Comparison of Delta and Absolute Value Approaches

A critical decision in the development of the ethanol soft sensor was whether to use absolute values or delta (change) values for model training and prediction. Absolute values refer to direct measurements obtained during fermentation, such as ethanol concentration or temperature at each time point. In contrast, delta values represent the change between consecutive time steps, capturing the dynamic behaviour of the process over time. The rationale for investigating delta values lies in the nature of fermentation itself. Industrial fermentation processes frequently exhibit non-stationary behaviour due to fluctuations in feed rates, temperature regulation, and yeast activity. Delta values may better capture such dynamic transitions, potentially improving the model's ability to generalise under variable operating conditions. To assess this, soft sensor models were developed using both absolute and delta representations. Table 4—1 presents the validation root mean square error (RMSE) values for each model family trained on both types of data.

Table 4—1: RMSE validation results for ABS and DELTA soft sensor approaches

Model Family	**Model Type**	**ABS Validation RMSE**	**DELTA Validation RMSE**
Gaussian Process Regression	Rational Quadratic GPR	9.5526	0.69692
Ensemble	Bagged Trees	10.905	0.69797
SVM	Fine Gaussian SVM	11.143	0.72232
Kernel	Least Squares Regression Kernel	11.641	0.72020

Linear Regression	Linear	13.274	0.82029
Neural Network	Medium Neural Network	11.405	0.87684
Stepwise Linear Regression	Stepwise Linear	13.055	0.82944
Tree	Medium Tree	11.618	0.75868

The results indicate that, across all model families, training on delta values produced significantly lower RMSE values compared to training on absolute values. This improvement, however, should be interpreted with care. Since absolute values are generally of larger magnitude than deltas, their associated errors also tend to be higher. As such, comparing RMSE directly between these two data types can be misleading due to scale differences. To enable a fair comparison, all models were evaluated using a test dataset that was normalised and expressed entirely in delta form. This ensured that models trained on either representation were tested under identical conditions, focusing on their ability to predict change rather than static values. Figure 4—4 presents parity plots comparing predicted versus actual ΔEthanol (g/L) for each regression model, with separate predictions shown for models trained on absolute values (blue points) and delta values (red points). The black reference line indicates the ideal prediction (1:1 line). Across all models, delta-trained predictions are more tightly clustered around the ideal line, with substantially lower RMSE values than their absolute-trained counterparts. This pattern is most pronounced in models such as Bagged Trees, Rational Quadratic GPR, and Least Squares Kernel. The bottom right subplot aggregates all predictions across models, reaffirming the consistent superiority of delta-based training for capturing ethanol dynamics.

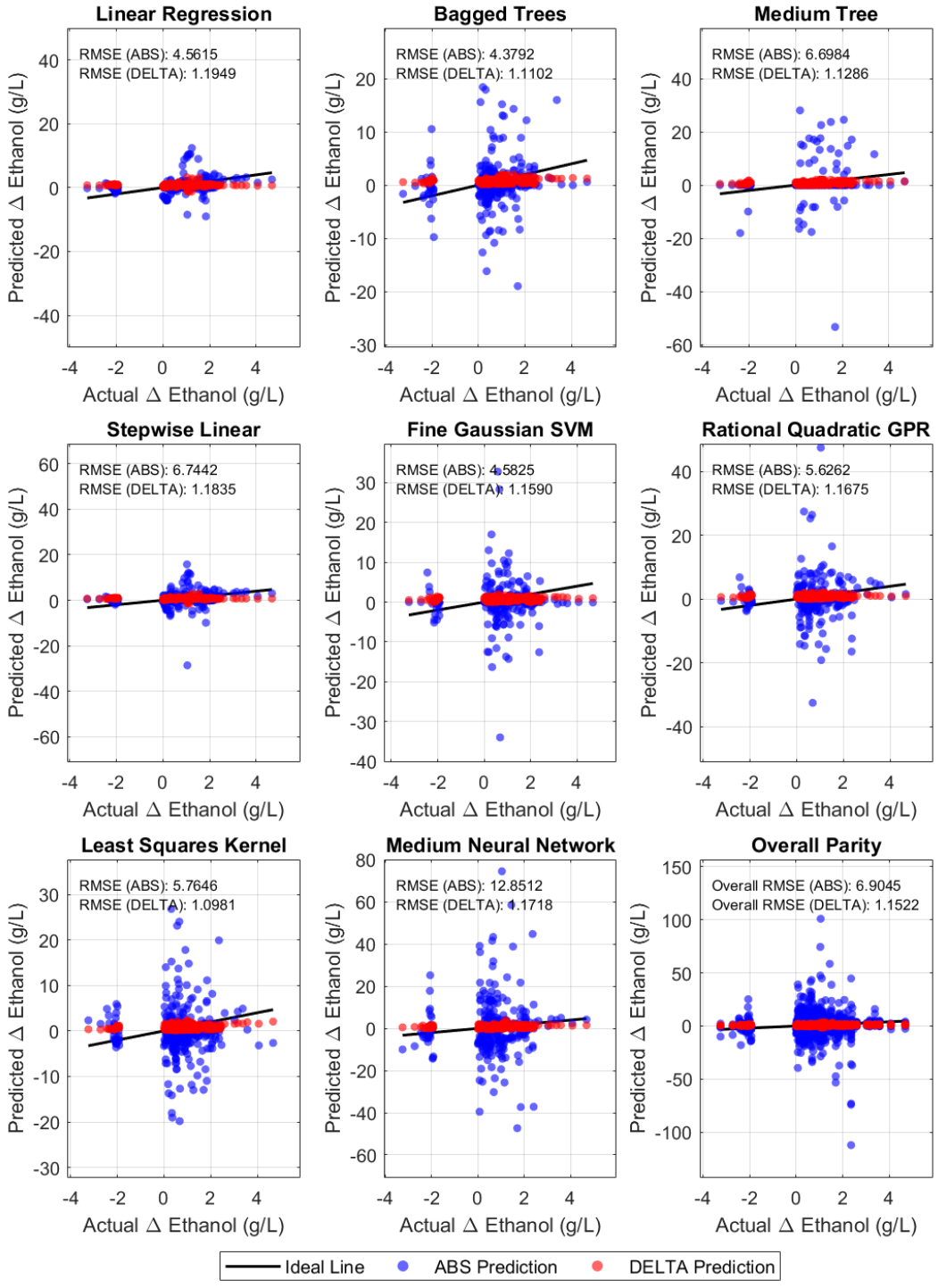


Figure 4—4: Absolute and Delta prediction error and accuracy analysis

The analysis confirms that delta values offer a superior input structure for soft sensor development in this context. The scale used in Figure 4—4 is fair as the data for the absolute set was converted to delta values before RMSE was calculated. This insight guided the subsequent design and optimization of the soft sensor, where delta representations were adopted for all future model configurations.

4.4 Ethanol Soft Sensor Validation

4.4.1 Realistic Training & Generalizable Testing

An essential step in the development of a robust ethanol soft sensor was to design a validation approach that balanced training diversity with realistic test scenarios. The traditional strategy of splitting data randomly across experiments may lead to overfitting or inflated performance metrics that do not reflect real-world implementation. Instead, an experiment optimization (EO) strategy was employed to carefully select which fermentation experiments would be used for model training and which would be reserved exclusively for testing. Experiments 1, 2, 3, 4, 6, 7, and 9 were selected for training because they represented a broad range of operational conditions. These included variations in feed rates (Exp 2 and 3), changes in yeast and yeast extract concentrations (Exp 6 and 7), and basic feedback control mechanisms (Exp 9). This design ensured that the models were exposed to diverse process dynamics and could learn generalizable trends in ethanol production behaviour. In contrast, experiments 5, 8, and 10 were used exclusively for testing. These experiments featured realistic process disturbances: Experiment 5 experienced a peristaltic pump failure, Experiment 8 employed a feedback control loop on specific gravity, and Experiment 10 included a batch-fed transition and mid-process dilution event. By using these as unseen test cases, the soft sensor models were evaluated under conditions that better represent the complexities of industrial deployment. This EO strategy improves upon a naive random-split approach by ensuring that the models are not inadvertently tuned to noise or specific patterns from test cases. It also provides a more rigorous

performance evaluation by challenging the models to predict ethanol dynamics during unsteady-state, feedback-driven, and perturbed process scenarios.

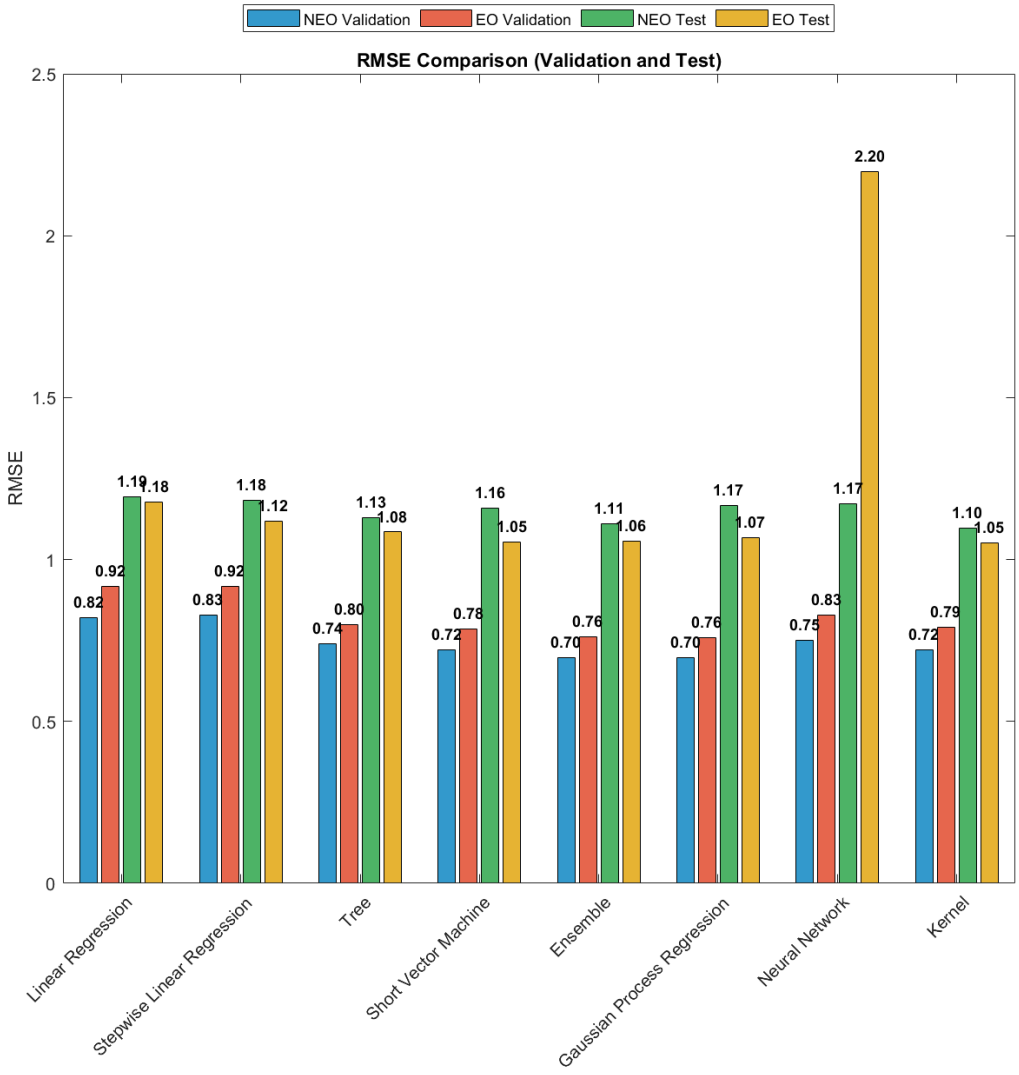


Figure 4—5: RMSE comparison between EO and NEO splits on validation and test data sets

The results of this training/testing scheme are illustrated in Figure 4—5, which compares the RMSE for both validation and test sets across all model families. Performance using EO is compared with a no optimization (NEO) split, where the first

7 experiments were used for training and the last 3 for testing without consideration of process dynamics. As seen in the figure, EO generally improves testing RMSE across most model families, indicating improved generalization.

4.4.2 Input Configuration & Smoothing Effects on Model Accuracy

To enhance the predictive power and generalizability of the soft sensor, additional preprocessing strategies were evaluated. This included the inclusion of a fourth input variable, cumulative heat load (tload), and the application of differential smoothing (DS) on the input signals. The base case used three inputs (tamb, tferm, and tjack), while the extended configuration introduced a fourth input (tload) to account for accumulated thermal energy exposure, which is expected to influence fermentation kinetics. Differential smoothing was implemented by applying a moving average filter to the delta-transformed features prior to model training. The goal was to suppress measurement noise while preserving underlying dynamic trends. Model validation performance was evaluated using R^2 (coefficient of determination) across all configurations and for each model. The full results are presented in Table 4—2, which compares model performance across 26 model types, spanning multiple model families.

Table 4—2: R^2 validation results across input configurations and smoothing

Model Family	**Model Type**	**3-Input NDS**	**3-Input DS**	**4-Input NDS**	**4-Input DS**
Linear Regression	Linear	0.03	0.12	0.15	0.25
Linear Regression	Interactions Linear	0.04	0.12	0.15	0.31
Linear Regression	Robust Linear	0.05	0.09	0.15	0.24
Stepwise Linear Regression	Stepwise Linear	0.03	0.11	0.14	0.29
Tree	Fine Tree	0.17	0.47	0.39	0.62
Tree	Medium Tree	0.27	0.48	0.34	0.6

Tree	Coarse Tree	0.22	0.45	0.37	0.46
SVM	Linear SVM	0.01	0.06	0.14	0.22
SVM	Quadratic SVM	0.04	0.08	0.17	0.27
SVM	Cubic SVM	-0.3	0.08	-0.02	-0.51
SVM	Fine Gaussian SVM	0.27	0.56	0.42	0.71
SVM	Medium Gaussian SVM	0.23	0.42	0.32	0.58
SVM	Coarse Gaussian SVM	0.09	0.2	0.18	0.31
Ensemble	Boosted Trees	0.34	0.58	0.52	0.69
Ensemble	Bagged Trees	0.32	0.52	0.52	0.71
GPR	Squared Exponential GPR	0.3	0.6	0.48	0.78
GPR	Matern 5/2 GPR	0.31	0.61	0.5	0.8
GPR	Exponential GPR	0.34	0.62	0.54	0.8
GPR	Rational Quadratic GPR	0.33	0.61	0.53	0.81
Neural Network	Narrow Neural Network	0.22	0.4	0.32	0.5
Neural Network	Medium Neural Network	0.08	0.4	0.21	0.52
Neural Network	Wide Neural Network	-1.29	0.26	-0.29	0.54
Neural Network	Bilayered Neural Network	-0.4	0.3	0.23	0.51
Neural Network	Trilayered Neural Network	-0.2	0.45	0.16	0.47
Kernel	SVM Kernel	0.25	0.47	0.26	0.47
Kernel	Least Squares Regression Kernel	0.23	0.48	0.3	0.47

Across all model families, both the addition of the fourth input (tload) and the application of differential smoothing resulted in consistent improvements in validation R^2. This suggests that smoothing helped reduce noise, and the tload variable provided complementary information capturing the long-term heating effect on fermentation. Among all configurations, the 4-input DS combination emerged as the most robust, delivering the highest R^2 values in nearly every model family. The top-performing

model under this configuration was the Rational Quadratic GPR, achieving an R^2 of 0.81, followed closely by other GPR and ensemble variants.

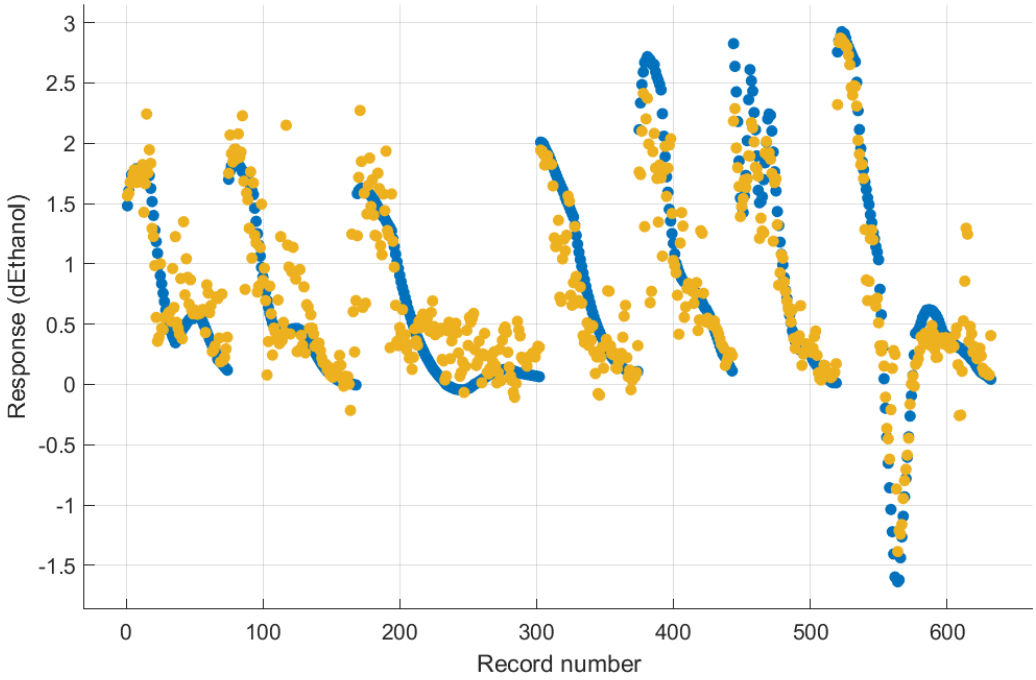


Figure 4—6: GPR actual (blue) vs prediction (mustard) validation of top performing model

Conversely, models trained without smoothing or with only three inputs performed considerably worse, with many linear and SVM variants falling below $R^2 = 0.2$ in their base configurations. These findings reinforce the importance of advanced feature engineering and preprocessing in soft sensor development. Other model validation graphs can be found in the appendix.

4.5 Soft Sensor Testing

Following the training and validation phases of soft sensor development, a final testing phase was conducted to evaluate the generalization capacity of the top-performing models from each machine learning family. The selected models were trained using the

optimal configuration identified during validation, a 4-input delta model with differential smoothing, and were tested on experiments 5, 8, and 10. These experiments were deliberately excluded from training to serve as challenge cases, representing real-world variability, including process disturbances (peristaltic pump failure in experiment 5), adaptive control (SG-based feedback in experiment 8), and abrupt dilution (large water addition in experiment 10). Each model was evaluated using two plots: a time-series prediction plot comparing actual and predicted Δ ethanol values, and a parity plot showing predicted versus actual values with corresponding RMSE and R^2 statistics. The results are presented in Table 4—3 and a visualization of the GPR model is shown in Figure 4—7.

Table 4—3: Summary of soft sensor test result outputs

Model	**RMSE**	**R^2**
GPR	0.98042	-0.22102
Ensemble	1.0147	-0.30784
Tree	1.1114	-0.56896
Kernel	1.0881	-0.50383
SVM	0.97671	-0.21179
Linear Regression	1.0476	-0.39397
Stepwise Linear Regression	1.0258	-0.3367
NN	4.3146	-22.6469

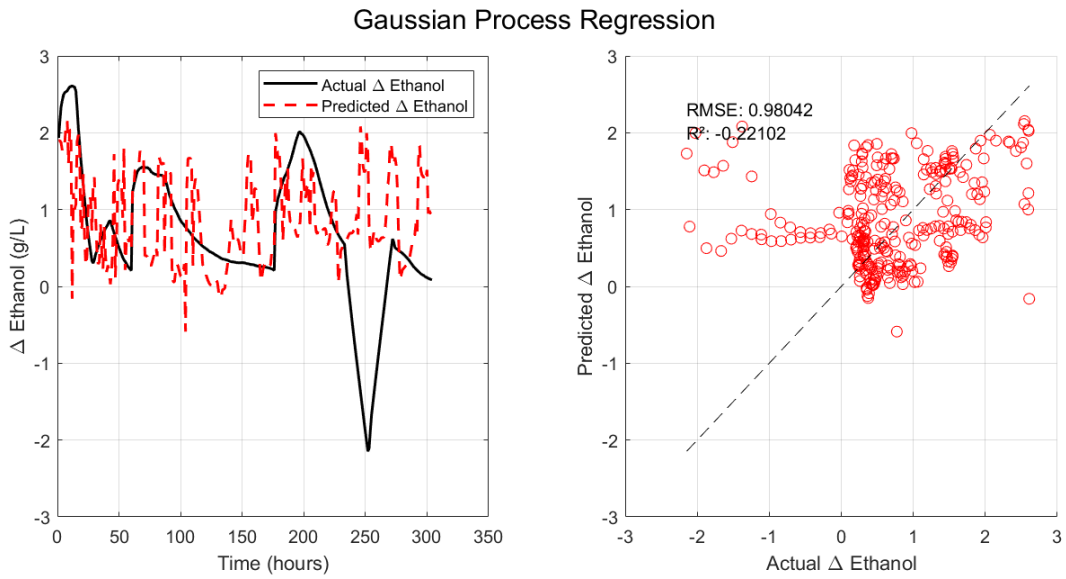


Figure 4—7: GPR test response

GPR performed moderately well, though it failed to capture the full range of ethanol dynamics. Predictions were consistently flattened relative to the true signal, with an RMSE of 0.98 and a negative R^2 (-0.22), indicating poor correlation in test conditions. Ensemble displayed similar underperformance with an R^2 of -0.31. The model exhibited a smoothing bias and failed to respond to local spikes in ethanol production, especially during non-linear transitions in Experiment 10. Decision Tree overfit the training data and produced discrete output steps, resulting in the poorest test set R^2 (-0.57). This lack of smoothness and adaptability to process noise rendered the model ineffective for real-time predictions. Kernel Regression and SVM provided somewhat better alignment in trajectory but still underperformed, showing prediction clustering around mean values and reduced responsiveness to sharp process changes. Linear and Stepwise Linear Regression models exhibited consistent underfitting, characterized by narrow dynamic range and weak correlation structures. Neural Network, despite high flexibility, was highly unstable under test conditions, producing erratic predictions and an extremely negative R^2 (-22.65), revealing an inability to generalize under unfamiliar conditions. Collectively, these results highlight a key limitation in current model

performance: even the most accurate models during validation showed significant deterioration when exposed to real-world disturbances in the test set. While RMSE values remained modest in some cases, the consistently negative R^2 values across families underscore a lack of correlation between actual and predicted ethanol rates. This gap between predicted and actual responses, especially visible in sharp transients and noisy phases, suggests that the model outputs are vulnerable to fluctuations in process signals. In practice, such unfiltered predictions would be impractical for use in real-time control or decision-making contexts, where consistency and trend-following are paramount. To address this limitation, the next section explores the application of smoothing techniques to the raw soft sensor output. These techniques aim to reduce noise and improve trend alignment, enhancing the practical utility of the soft sensor in operational environments where stability and interpretability are critical. The code for the raw test response outputs are available in the appendix.

4.6 Output Smoothing

In industrial applications, the utility of a soft sensor must extend beyond accurate predictions it must also offer stable and interpretable outputs suitable for real-time monitoring and decision-making. Raw soft sensor predictions, particularly when modelled on dynamic delta values, often display high-frequency noise due to model uncertainty and process variability as depicted in the previous section. To address this, a post-processing smoothing technique was evaluated. This section applies moving average filters of varying window sizes (10, 20, 30, 40 samples) to the outputs of the best-performing model from each machine learning family, as identified in Section 4.4. The goal was to assess the extent to which smoothing improves output clarity without significantly distorting the ethanol production dynamics. The effect of smoothing was assessed both visually and quantitatively, using RMSE and R^2 as evaluation metrics.

Table 4—4: RMSE & R^2 results after smoothing raw soft sensor outputs

Model Family	Window 10 RMSE	Window 10 R^2	Window 20 RMSE	Window 20 R^2	Window 30 RMSE	Window 30 R^2	Window 40 RMSE	Window 40 R^2
GPR	0.8943	-0.0160	0.8602	0.0602	0.8525	0.0786	0.8510	0.0801
Ensemble	0.9583	-0.1557	0.9403	-0.1232	0.9393	-0.1208	0.9413	-0.1255
Tree	1.0031	-0.2781	0.9845	-0.2312	0.9837	-0.2293	0.9916	-0.2490
Kernel	1.0510	-0.4032	1.0413	-0.3773	1.0442	-0.3850	1.0540	-0.4112
SVM	0.9436	-0.1311	0.9286	-0.0954	0.9227	-0.0814	0.9193	-0.0736
Stepwise Linear Regression	0.8986	-0.0257	0.8810	0.0140	0.8791	0.0183	0.8821	0.0116
Linear Regression	0.9053	-0.0410	0.8836	0.0083	0.8806	0.0149	0.8824	0.0110
NN	3.2446	-12.3729	2.7507	-2.6118	2.5738	-7.4147	2.4602	-6.6882

Across families, smoothing with a window size of 30 consistently yielded the best trade-off between noise reduction and signal preservation. This window was selected for standardization in the final sensor implementation. Neural Networks performed poorly, with noisy outputs persisting even after aggressive smoothing, indicating overfitting or inadequate training on the delta-based feature space. Gaussian Process Regression (GPR) models demonstrated the strongest post-smoothing performance, with the RMSE decreasing and R^2 improving consistently across increasing window sizes. At a smoothing window of 40, GPR reached an RMSE of 0.8510 and R^2 of 0.0801 which is the best combined metric performance across all families.

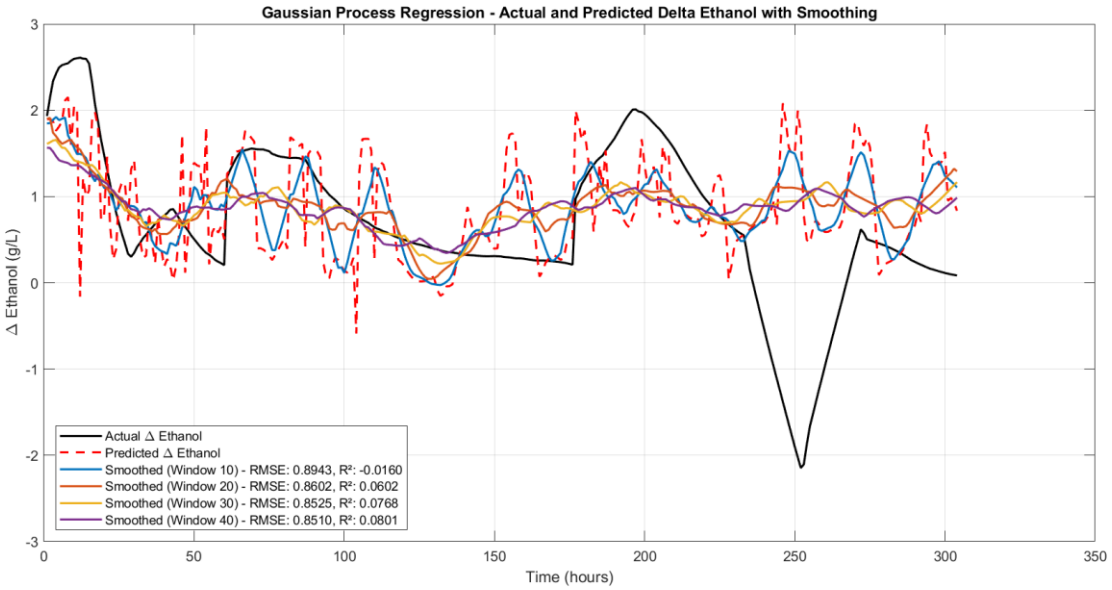


Figure 4—8: Smoothing response of Gaussian Process Regression output

Based on these findings, GPR was selected as the final soft sensor model for individual experiment-level performance visualization and discussion in the next section. Its combination of low error, consistent output shape, and robustness under smoothing filters makes it the most viable candidate for real-world deployment. The code which generated the smoothing response graphs for all top performing experiments is available in the appendix.

4.7 Developed Soft Sensor

This section presents the final performance assessment of the developed soft sensor using the best-performing model configuration: Gaussian Process Regression (GPR) trained on delta values with 4 inputs and differential smoothing (DS), and post-processed using a moving average smoothing filter with a window size of 40. The evaluation focuses on three independent test experiments, experiment 5, experiment 8, and experiment 10, which were excluded from the training phase to provide realistic industrial validation scenarios. These experiments were intentionally selected to

include complex fermentation dynamics, such as pump failures, adaptive feeding strategies, and batch-to-feed transitions. The purpose of this section is to illustrate the model's strengths and shortcomings in reproducing ethanol production behaviour under such real-world variations. The core objective of the soft sensor is to estimate delta ethanol, which serves as a proxy for yeast metabolic activity. As such, the primary focus of this evaluation is on the model's ability to predict changes in ethanol concentration over time, rather than cumulative production. While cumulative ethanol profiles are included for reference, the emphasis remains on accurate dynamic tracking of delta ethanol.

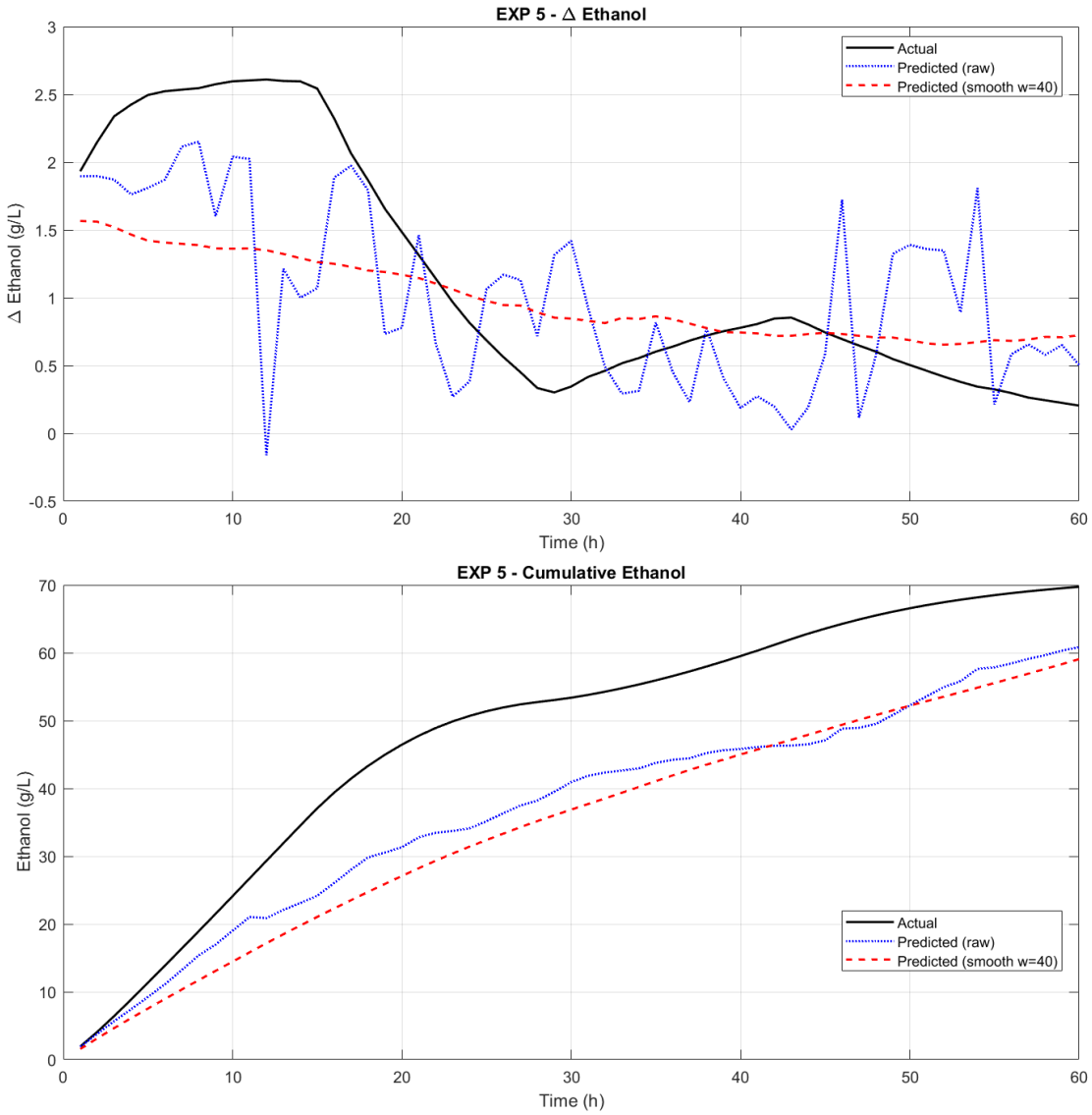


Figure 4—9: Soft sensor prediction vs actual for experiment 5

Experiment 5 featured an operational disruption caused by a pump failure that resulted in a non-linear spike in SG and subsequent dilution. As shown in Figure 4—9, the raw predicted Δethanol (blue) exhibited significant noise. However, the smoothed prediction (red) better followed the general trend of the actual Δethanol signal (black). Notably, the model was able to capture the broad phases of ethanol change despite the abrupt disturbance. The corresponding cumulative ethanol profile Figure 4—9

illustrates how early underestimation leads to progressive divergence from the actual profile. The parity plot Figure 4—12 shows an RMSE of 0.6377 and an R^2 of 0.4600, indicating moderate alignment and partial retention of yeast activity trends even under process disturbances.

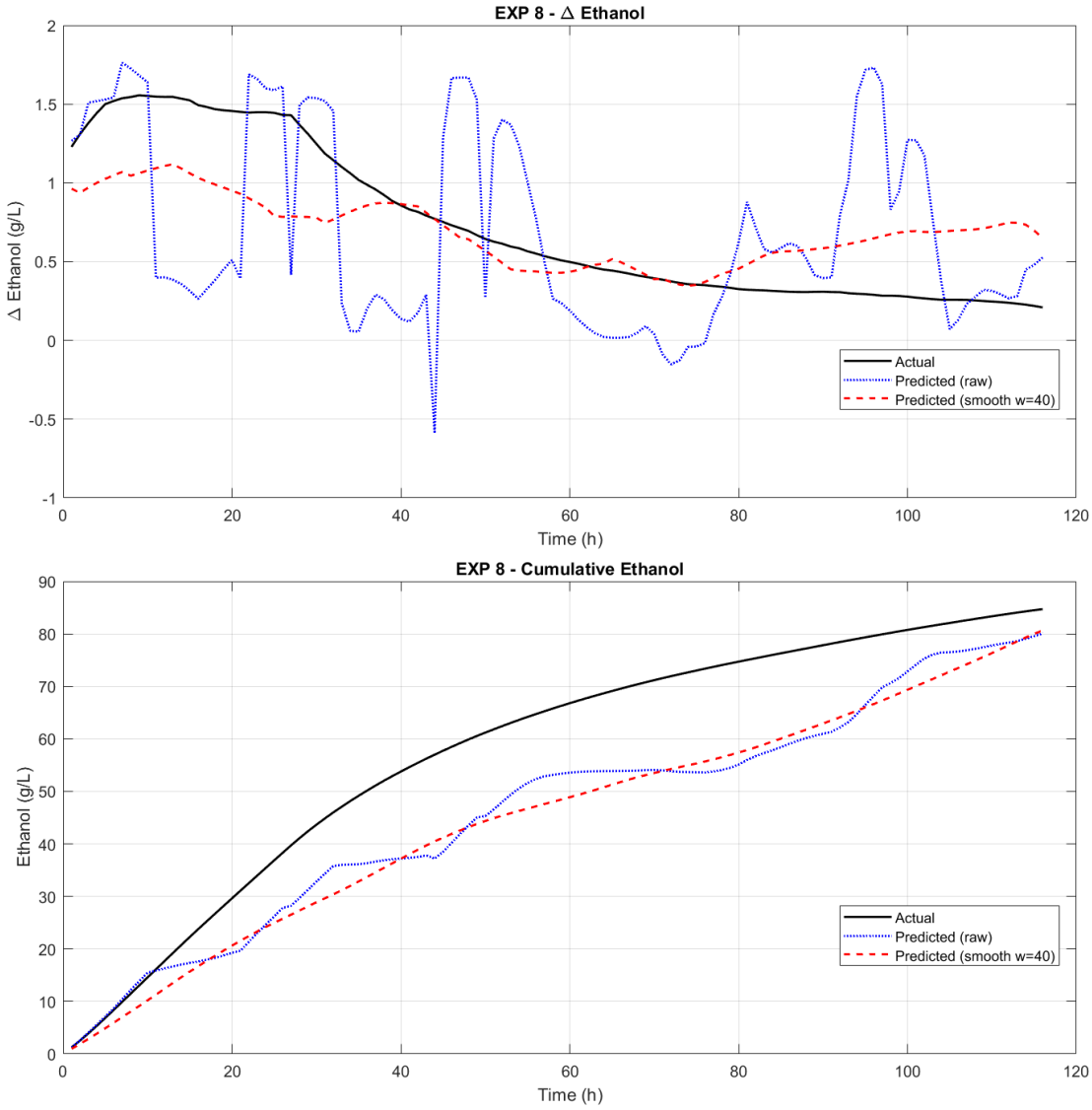


Figure 4—10: Soft sensor prediction vs actual for experiment 8

In this experiment, a feedback control strategy was employed to regulate sugar addition via SG control, leading to more gradual fermentation kinetics. The smoothed prediction curve Figure 4—10 shows strong agreement with the actual Δethanol profile, capturing both peak and decay regions. The cumulative profile remains closely aligned throughout, confirming consistent tracking. The parity plot yields the best accuracy among all test cases (RMSE: 0.3404, R^2: 0.5008), underscoring the model's robustness in well-regulated environments. Importantly, the ability to reproduce Δethanol dynamics under control feedback validates the model's suitability for predictive soft sensing of yeast behaviour in semi-automated fermentation systems. Experiment 10 was designed to challenge the model with abrupt process transitions. A late-stage switch from batch to fed-batch feeding, coupled with a major water addition event, introduced substantial non-linearity. The smoothed Δethanol curve Figure 4—11 failed to capture the sharp negative dip caused by dilution, and lagged behind the recovery phase. The cumulative ethanol prediction diverged significantly in the mid-to-late stages. The parity plot Figure 4—12 highlights the model's poor performance here, with an RMSE of 1.1641 and an R^2 of -0.1953, reflecting a breakdown in predictive reliability under uncontrolled and irregular disturbances. This experiment reinforces the model's limitations when subjected to conditions far outside the training domain.

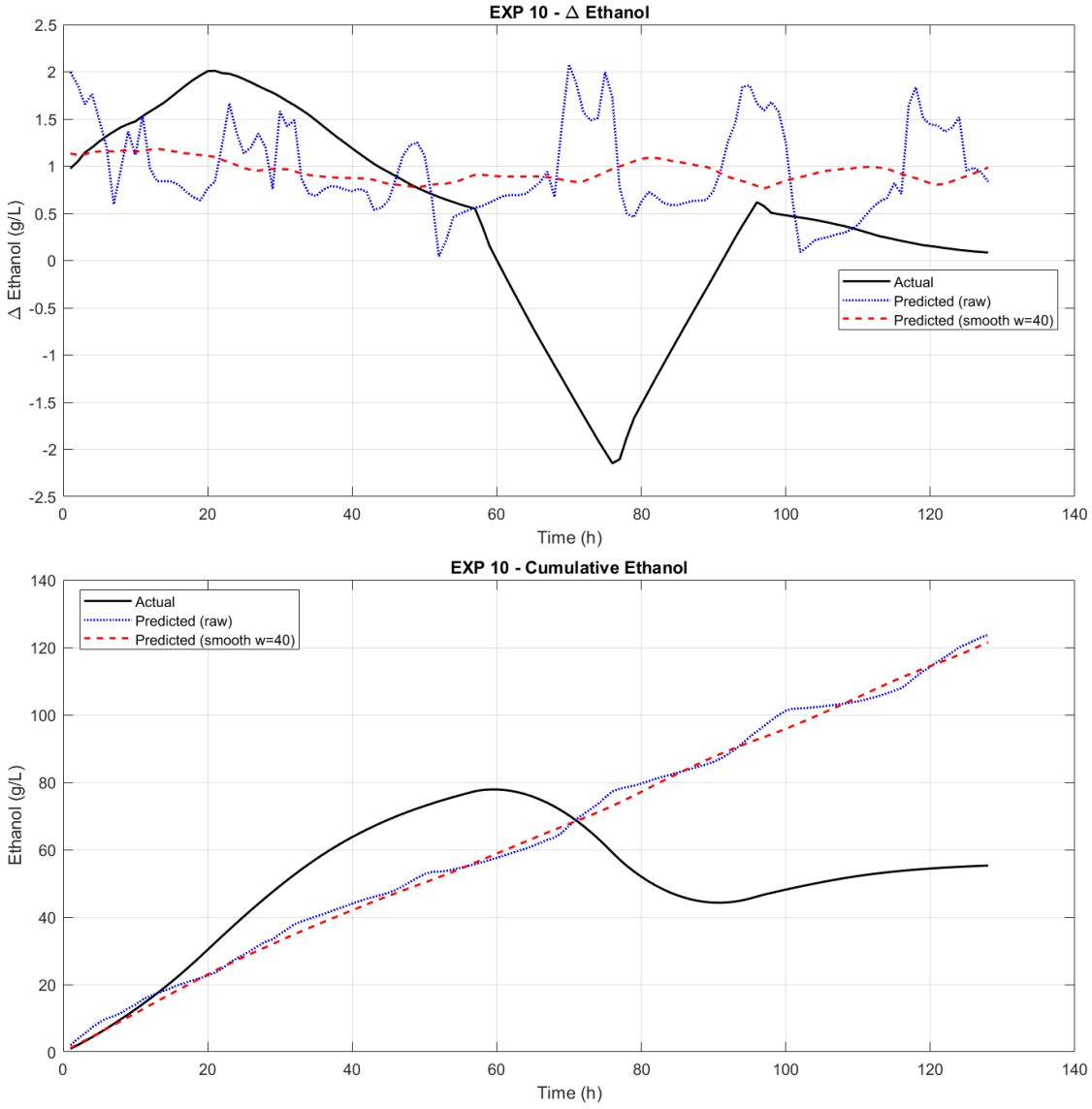


Figure 4—11: Soft sensor prediction vs actual for experiment 10

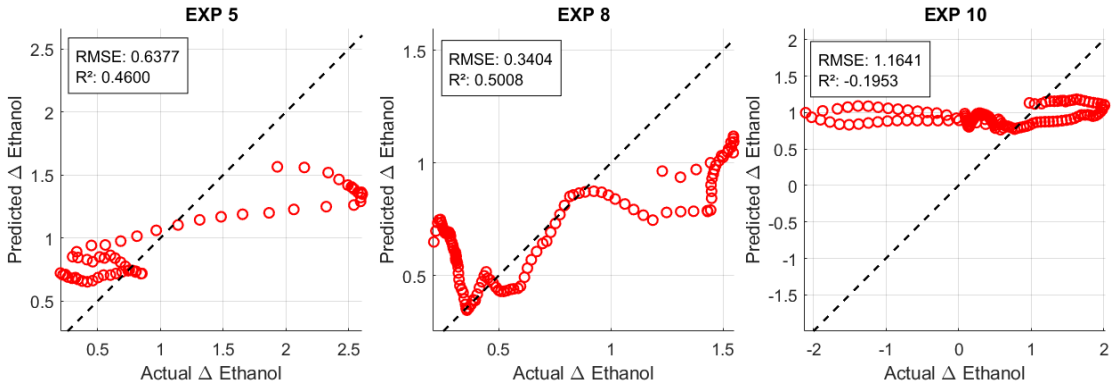


Figure 4—12: Parity plot for each of the test experiments after smoothing

Across all test experiments, the smoothed GPR model successfully tracked Δethanol behaviour under moderate and controlled disturbances, particularly in Experiment 8. While performance in Experiment 10 revealed limitations under severe deviation, the ability to follow Δethanol trajectories under varying dynamics confirms the model's core functionality, providing a proxy for yeast activity using only temperature-derived data. These results underscore the practical utility of the soft sensor for process monitoring and early deviation detection in real-world applications. They also establish a foundation for the next phase of development, improving prediction quality through temporal smoothing and adaptive model tuning.

5. CONCLUSION & RECOMMENDATION

5.1 Conclusion

This research successfully developed a soft sensor to estimate ethanol fermentation rates using only temperature-derived measurements. By implementing a delta-based machine learning approach trained on thermodynamically informed features, the model achieved high accuracy in capturing yeast metabolic activity trends across diverse fermentation profiles. The ethanol estimation model based on specific gravity (SG) achieved a strong overall correlation with HPLC measurements, with a combined R^2 of 0.75 and a normalized RMSE (NRMSE) of 0.17 across nine fermentation runs. These values validated the SG-derived ethanol estimates as a suitable training target. A critical shift in methodology, using delta values rather than absolute ones, led to substantial performance improvements. Across all regression model families tested, delta-trained models consistently outperformed their absolute-trained counterparts. For example, the Rational Quadratic Gaussian Process Regression (GPR) model reduced its test RMSE from 5.62 g/L (ABS) to 1.17 g/L (DELTA). This represented a 79.18% improvement in error magnitude, underscoring the advantage of tracking rate-based dynamics in soft sensing. The final soft sensor architecture used a 4-input delta model with differential smoothing and GPR as the chosen algorithm. After post-processing with a 40-point moving average filter, the model achieved its best performance in Experiment 8 with an RMSE of 0.34 g/L and R^2 of 0.50. Even under disturbed conditions (pump failure in Experiment 5), the model still captured trend directionality, returning an R^2 of 0.46 and RMSE of 0.64 g/L. While performance degraded under complex and irregular conditions such as dilution spikes, Experiment 10, $R^2 = –0.20$, RMSE = 1.16 g/L, these results demonstrate important aspects of the soft sensors robustness under severely disturbed conditions. In summary, this work demonstrates

that accurate real-time estimation of fermentation rate is possible using only thermal data and statistical learning, providing an accessible and low-cost solution to one of the most persistent measurement challenges in fermentation engineering.

5.2 Recommendation

Based on these findings, various recommendations are made. Implement the soft sensor in parallel with existing HPLC or sampling systems, particularly in pilot or production-scale fermentations. Under controlled or semi-disturbed conditions, the model can offer immediate insights into fermentation dynamics without sampling delays. Use delta-based modelling and smoothing as standard preprocessing. Across all model families, this consistently improved performance, with reduction in RMSE compared to absolute value models. Adopt GPR as the default regression approach for ethanol estimation tasks of this nature, particularly when working with limited or noisy datasets. The Rational Quadratic GPR outperformed all others in both validation and smoothed output metrics, with a top R^2 of 0.81 during cross-validation. Validate the sensor across higher volumes and alternative substrates, especially since this work was conducted using a 100 L stainless-steel fermenter with sugar feedstock. Broadening the test domain will verify generalizability. It is further recommended that the smoothing framework demonstrated in this dissertation be refined, as optimization of this approach holds potential to significantly improve model output.

5.3 Future Work

This project opens several future research directions. Incorporating additional sensor inputs such as CO_2 off-gas rates, pH, or DO would likely enhance model robustness. While the 4-input model performed well with R^2 of 0.81 with smoothing, many models underperformed in complex test scenarios. Real-time deployment via MLOps infrastructure should be explored to move from offline prediction to industrial control. Adaptive retraining mechanisms could improve long-term accuracy. For instance,

integrating feedback from daily lab samples or cumulative fermentation outputs may help recalibrate the model in response to seasonal or microbial shifts. Physiological modelling of yeast vitality and stress, building on Δethanol as a proxy, could provide even deeper insight into fermentation health. This could support early warnings in case of stalled fermentations or nutrient limitations. Cost-benefit and economic analysis should be conducted to quantify the savings from reduced HPLC sampling, earlier fault detection, and more efficient process scheduling. Estimating the ROI of soft sensor deployment could further justify industrial implementation.

"Part of the journey is the end"

— Tony Stark

REFERENCES

Badino, A. C., Veloso, I. I. K., Rodrigues, K. C. S., Ribeiro, M. P. A. & Cruz, A. J. G. (2020). Temperature influence in real-time monitoring of fed-batch ethanol fermentation by mid-infrared spectroscopy. *Industrial and Engineering Chemistry Research*, *59*(41), 18425–18433. https://doi.org/10.1021/acs.iecr.0c03717

Bellon-Maurel, V., Orliac, O. & Christen, P. (2003). Sensors and measurements in solid state fermentation: A review. *Process Biochemistry*, *38*(8), 881–896. https://doi.org/doi.org/10.1016/S0032-9592(02)00093-6

Bowler, A., Escrig, J., Pound, M. & Watson, N. (2021). Predicting alcohol concentration during beer fermentation using ultrasonic measurements and machine learning. *Fermentation*, *7*(1). https://doi.org/10.3390/fermentation7010034

Brooks, K. & Margret, B. (2018). Sensor validation and reconstruction: Experiences with commercial technology. *Control Engineering Practice*, *77*, 28–40. https://doi.org/10.1016/j.conengprac.2018.04.003

Brown, T. L., LeMay, H. Eugene., Bursten, B. Edward., Murphy, C. J. ., Woodward, P. M. . & Stoltzfus, Matthew. (2015). *Chemistry: The central science* (14th ed.). Pearson.

Brunner, V., Siegl, M., Geier, D. & Becker, T. (2021). Challenges in the development of soft sensors for bioprocesses: a critical review. *Frontiers in Bioengineering and Biotechnology*, *9*. https://doi.org/10.3389/fbioe.2021.722202

Cutaia, A. J., Reid, A. J. & Alex Speers, R. (2009). Examination of the relationships between original, real and apparent extracts, and alcohol in pilot plant and commercially produced beers. *Journal of the Institute of Brewing*, *115*(4), 318–327. https://doi.org/10.1002/j.2050-0416.2009.tb00387.x

De Deken, R. H. (1966). The Crabtree effect: A regulatory system in yeast. *Journal of General Microbiology*, *44*, 149–156.

Doan, X. T., Srinivasan, R., Bapat, P. M. & Wangikar, P. P. (2007). Detection of phase shifts in batch fermentation via statistical analysis of the online measurements: A case study with rifamycin B fermentation. *Journal of Biotechnology*, *132*(2), 156–166. https://doi.org/10.1016/j.jbiotec.2007.06.013

Douradinho, R., Sica, P., Tonoli, F., Mattos, E., Oliveira, M., Pinto, A., Mota, L., Faria, T., Costa, V. F., Leite, G., Arthur, V., Coelho, S. & Baptista, A. (2023). Osmotic stress alleviation in *Saccharomyces cerevisiae* for high ethanol fermentations with different wort substrates. *Stresses*, *3*(4), 813–826. https://doi.org/10.3390/stresses3040055

Duncan, J. D., Devillers, H., Camarasa, C., Setati, M. E. & Divol, B. (2024). Oxygen alters redox cofactor dynamics and induces metabolic shifts in Saccharomyces cerevisiae during alcoholic fermentation. *Food Microbiology*, *124*. https://doi.org/10.1016/j.fm.2024.104624

Fisher, O. J., Watson, N. J., Porcu, L., Bacon, D., Rigley, M. & Gomes, R. L. (2022). Data-driven modelling for resource recovery: Data volume, variability, and visualisation for an industrial bioprocess. *Biochemical Engineering Journal*, *185*. https://doi.org/10.1016/j.bej.2022.108499

Gill, N. K., Appleton, M., Baganz, F. & Lye, G. J. (2008). Design and characterisation of a miniature stirred bioreactor system for parallel microbial fermentations. *Biochemical Engineering Journal*, *39*(1), 164–176. https://doi.org/10.1016/j.bej.2007.09.001

Goldrick, S., Duran-Villalobos, C. A., Jankauskas, K., Lovett, D., Farid, S. S. & Lennox, B. (2019). Modern day monitoring and control challenges outlined on an industrial-scale benchmark fermentation process. *Computers and Chemical Engineering*, *130*. https://doi.org/10.1016/j.compchemeng.2019.05.037

Hellborg, L. & Piškur, J. (2009). Yeast Diversity in the Brewing Industry. In V. R. Preedy (Ed.), *Beer in Health and Disease Prevention*, (pp. 77–88). Academic Press. https://doi.org/10.1016/B978-0-12-373891-2.00007-9

Hocalar, A., Türker, M., Karakuzu, C. & Yüzgeç, U. (2011). Comparison of different estimation techniques for biomass concentration in large scale yeast fermentation. *ISA Transactions*, *50*(2), 303–314. https://doi.org/10.1016/j.isatra.2010.12.003

Hu, J., Li, H., Li, H., Wu, M., Cao, W. & Pedrycz, W. (2024). Relevance vector machine with hybrid kernel-based soft sensor via data augmentation for incomplete output data in sintering process. *Control Engineering Practice*, *145*. https://doi.org/10.1016/j.conengprac.2024.105850

iSpindle. (2025). *iSpindle Electronic Hydrometer*. https://www.ispindel.de/docs/README_en.html

Kadlec, P., Gabrys, B. & Strandt, S. (2009). Data-driven Soft Sensors in the process industry. *Computers and Chemical Engineering*, *33*(4), 795–814. https://doi.org/10.1016/j.compchemeng.2008.12.012

Katoh, S., Horiuchi, J. & Yoshida, F. (2015). *Biochemical engineering*. Wiley.

Khatibisepehr, S., Huang, B. & Khare, S. (2013). Design of inferential sensors in the process industry: A review of Bayesian methods. *Journal of Process Control*, *23*(10), 1575–1596. https://doi.org/10.1016/j.jprocont.2013.05.007

Kwon, H. J., Shiu, J. H., Yamakawa, C. K. & Rivera, E. C. (2024). Enhancing fermentation process monitoring through data-driven modeling and synthetic time series generation. *Bioengineering*, *11*(8). https://doi.org/10.3390/bioengineering11080803

Lepot, M., Aubin, J. B. & Clemens, F. H. L. R. (2017). Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. *Water*, *9*(10). https://doi.org/10.3390/w9100796

Liew, W. H., Hassim, M. H. & Ng, D. K. S. (2014). Review of evolution, technology and sustainability assessments of biofuel production. *Journal of Cleaner Production*, *71*, 11–29. https://doi.org/10.1016/j.jclepro.2014.01.006

Lip, K. Y. F., García-Ríos, E., Costa, C. E., Guillamón, J. M., Domingues, L., Teixeira, J. & van Gulik, W. M. (2020). Selection and subsequent physiological characterization of industrial *Saccharomyces cerevisiae* strains during continuous growth at sub- and- supra optimal temperatures. *Biotechnology Reports*, *26*. https://doi.org/10.1016/j.btre.2020.e00462

Liu, H., Yang, C., Huang, M. & Yoo, C. K. (2020). Soft sensor modeling of industrial process data using kernel latent variables-based relevance vector machine. *Applied Soft Computing Journal*, *90*. https://doi.org/10.1016/j.asoc.2020.106149

Liu, Y. & Xie, M. (2020). Rebooting data-driven soft-sensors in process industries: A review of kernel methods. *Journal of Process Control*, *89*, 58–73. https://doi.org/10.1016/j.jprocont.2020.03.012

López Pérez, P. A., Yescas, R. M., Gomez Acata, R. V., Caballero, V. P. & López, R. A. (2013). Software sensors design for the simultaneous saccharification and fermentation of starch to ethanol. *Fuel*, *110*, 219–226. https://doi.org/10.1016/j.fuel.2012.11.020

Luttmann, R., Bracewell, D. G., Cornelissen, G., Gernaey, K. V., Glassey, J., Hass, V. C., Kaiser, C., Preusse, C., Striedner, G. & Mandenius, C. F. (2012). Soft sensors in bioprocessing: A status report and recommendations. *Biotechnology Journal*, *7*(8), 1040–1048. https://doi.org/10.1002/biot.201100506

Maschler, B., Ganssloser, S., Hablizel, A. & Weyrich, M. (2021). Deep learning based soft sensors for industrial machinery. *Procedia CIRP*, *99*, 662–667. https://doi.org/10.1016/j.procir.2021.03.115

Matheri, A. N., Ntuli, F., Ngila, J. C., Seodigeng, T. & Zvinowanda, C. (2021). Performance prediction of trace metals and cod in wastewater treatment using

artificial neural network. *Computers and Chemical Engineering*, *149*. https://doi.org/10.1016/j.compchemeng.2021.107308

Metcalfe, B., Acosta-Pavas, J. C., Robles-Rodriguez, C. E., Georgakilas, G. K., Dalamagas, T., Aceves-Lara, C. A., Daboussi, F., Koehorst, J. J. & Corrales, D. C. (2025). Towards a machine learning operations (MLOps) soft sensor for real-time predictions in industrial-scale fed-batch fermentation. *Computers and Chemical Engineering*, *194*. https://doi.org/10.1016/j.compchemeng.2024.108991

Morales, H., Sciascio, F. di, Aguirre-Zapata, E. & Amicarelli, A. N. (2023). A model-based supersaturation estimator (inferential or soft-sensor) for industrial sugar crystallization process. *Journal of Process Control*, *129*. https://doi.org/10.1016/j.jprocont.2023.103065

Murugan, C. (2021). Soft sensors for biomass monitoring during low cost cellulase production. In *Biotechnological Applications of Biomass*. https://doi.org/10.5772/intechopen.96027

Narendranath, N. V. & Power, R. (2005). Relationship between pH and medium dissolved solids in terms of growth and metabolism of *Lactobacilli* and *Saccharomyces cerevisiae* during ethanol production. *Applied and Environmental Microbiology*, *71(5)*, 2239–2243. https://doi.org/10.1128/AEM.71.5.2239-2243.2005

Naveed, M. H., Khan, M. N. A., Mukarram, M., Naqvi, S. R., Abdullah, A., Haq, Z. U., Ullah, H. & Mohamadi, H. Al. (2024). Cellulosic biomass fermentation for biofuel production: Review of artificial intelligence approaches. *Renewable and Sustainable Energy Reviews*, *189*. https://doi.org/10.1016/j.rser.2023.113906

O' Connor, B., Riverol, C., Kelleher, P., Plant, N., Bevan, R., Hinchy, E. & D'arcy, J. (2002). Integration of fuzzy logic based control procedures in brewing. *Food Control*, *13*, 23–31. www.elsevier.com/locate/foodcont

Rao, A., Chen, T., Liu, Y. & Ma, F. (2023). Computational analysis of performances for a hydrogen enriched compressed natural gas engine' by advanced machine learning algorithms. *Fuel*, *347*. https://doi.org/10.1016/j.fuel.2023.128244

Reardon, K. F. (2021). Practical monitoring technologies for cells and substrates in biomanufacturing. *Current Opinion in Biotechnology*, *71*, 225–230. https://doi.org/10.1016/j.copbio.2021.08.006

Saucedo, V. M., Valentinotti, S. C., Karim, M. N. & Collins, H. W. (1995). Feedback control of a recombinant fed-batch fermentation using online HPLC measurements. *Journal of Biotechnology, 39(1),* 24–28. https://doi.org/doi.org/10.1016/S1474-6670(17)45596-0

Shang, C., Yang, F., Huang, D. & Lyu, W. (2014). Data-driven soft sensor development based on deep learning technique. *Journal of Process Control*, *24*(3), 223–233. https://doi.org/10.1016/j.jprocont.2014.01.012

Souza, F. A. A., Araújo, R. & Mendes, J. (2016). Review of soft sensor methods for regression applications. In *Chemometrics and Intelligent Laboratory Systems*, *152*, 69–79. https://doi.org/10.1016/j.chemolab.2015.12.011

Türker, M. (2004). Development of biocalorimetry as a technique for process monitoring and control in technical scale fermentations. *Thermochimica Acta*, *419*(1–2), 73–81. https://doi.org/10.1016/j.tca.2004.01.036

van Aalst, A. C. A., de Valk, S. C., van Gulik, W. M., Jansen, M. L. A., Pronk, J. T. & Mans, R. (2022). Pathway engineering strategies for improved product yield in yeast-based industrial ethanol production. *Synthetic and Systems Biotechnology*, *7*(1), 554–566. https://doi.org/10.1016/j.synbio.2021.12.010

van Rossum, H. M., Kozak, B. U., Pronk, J. T. & van Maris, A. J. A. (2016). Engineering cytosolic acetyl-coenzyme A supply in *Saccharomyces cerevisiae*: Pathway stoichiometry, free-energy conservation and redox-cofactor balancing.

Metabolic Engineering, *36*, 99–115. https://doi.org/10.1016/j.ymben.2016.03.006

Veloso, I. I. K., Rodrigues, K. C. S., Sonego, J. L. S., Cruz, A. J. G. & Badino, A. C. (2019). Fed-batch ethanol fermentation at low temperature as a way to obtain highly concentrated alcoholic wines: Modeling and optimization. *Biochemical Engineering Journal*, *141*, 60–70. https://doi.org/10.1016/j.bej.2018.10.005

Vojinović, V., Cabral, J. M. S. & Fonseca, L. P. (2006). Real-time bioprocess monitoring: Part I: In situ sensors. *Sensors and Actuators, B: Chemical*, *114*(2), 1083–1091. https://doi.org/10.1016/j.snb.2005.07.059

Von Stockar, U., Duboc, P., Menoud, L. & Marison, I. W. (1997). On-line calorimetry as a technique for process monitoring and control in biotechnology. *Thermochimica Acta*, *300*, 225–236.

Wang, X. & Liu, H. (2018). Soft sensor based on stacked auto-encoder deep neural network for air preheater rotor deformation prediction. *Advanced Engineering Informatics*, *36*, 112–119. https://doi.org/10.1016/j.aei.2018.03.003

Zhai, X., Hao, L., Ma, T., Song, B., Wang, K. & Luo, J. (2023). Non-linear soft sensing method for temperature of coal spontaneous combustion. *Process Safety and Environmental Protection*, *170*, 1023–1031. https://doi.org/10.1016/j.psep.2022.12.043

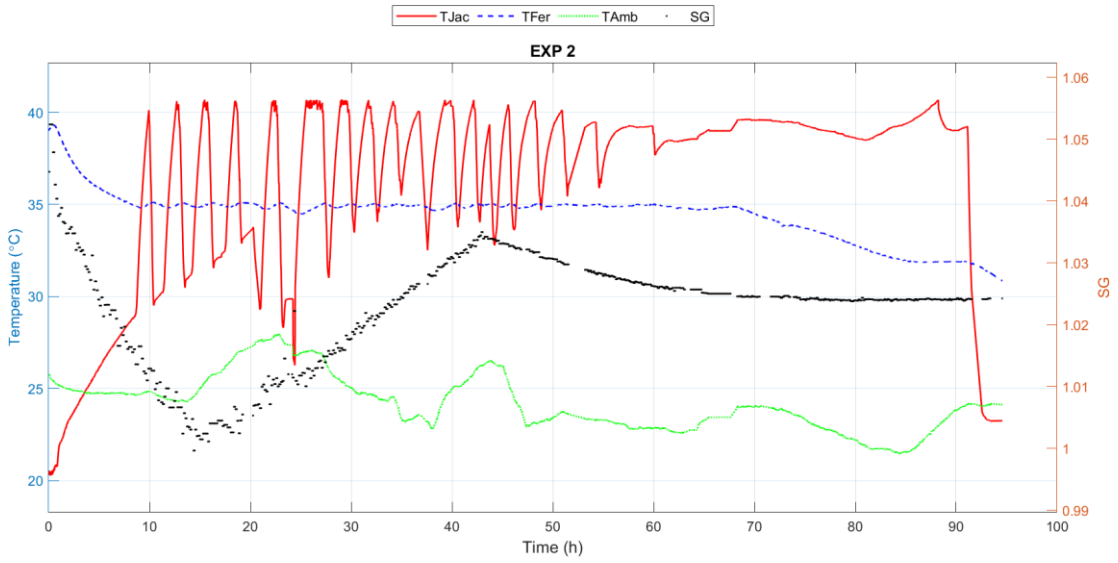
Zhang, M. & Liu, X. (2013). A soft sensor based on adaptive fuzzy neural network and support vector regression for industrial melt index prediction. *Chemometrics and Intelligent Laboratory Systems*, *126*, 83–90. https://doi.org/10.1016/j.chemolab.2013.04.018

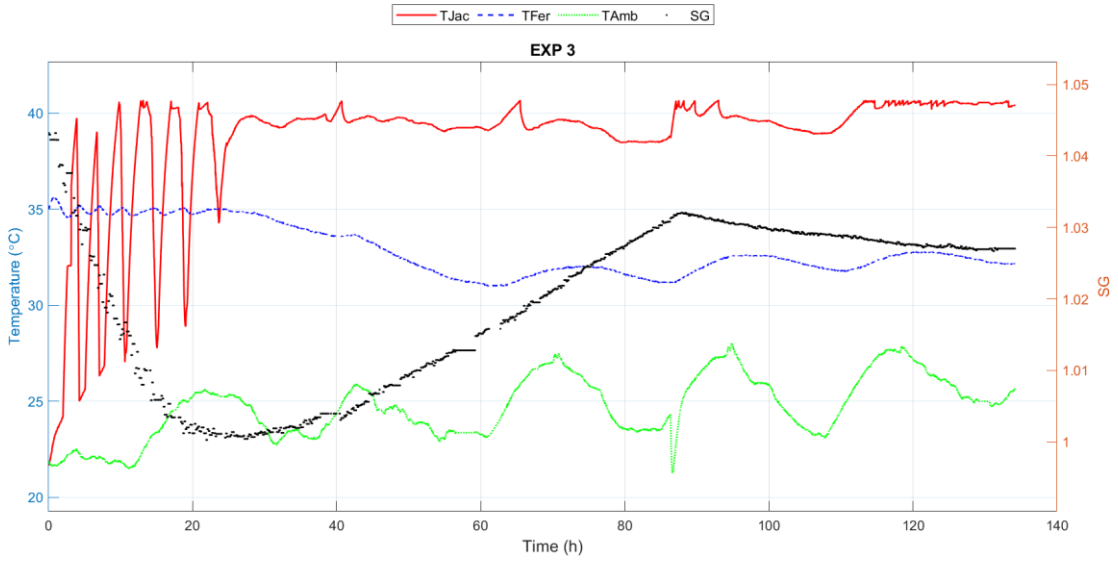
Zhang, Y., Su, M., Wang, Z., Nielsen, J. & Liu, Z. (2022). Rewiring regulation on respiro-fermentative metabolism relieved Crabtree effects in *Saccharomyces cerevisiae*. *Synthetic and Systems Biotechnology*, *7*(4), 1034–1043. https://doi.org/10.1016/j.synbio.2022.06.004

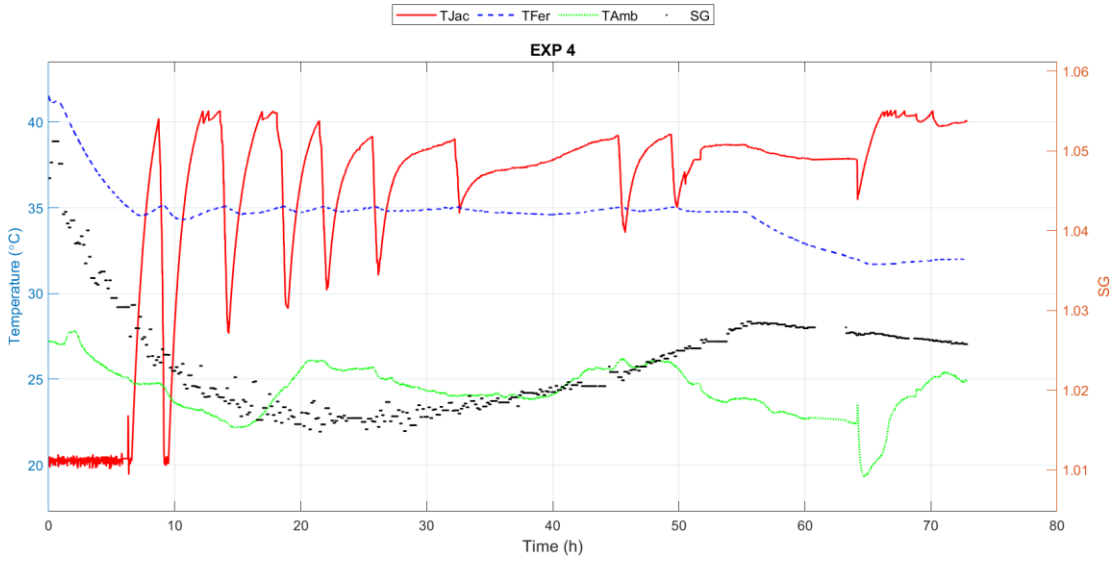
Zhu, X., Rehman, K. U., Wang, B. & Shahzad, M. (2020). Modern soft-sensing modeling methods for fermentation processes. *Sensors*, *20*(6). https://doi.org/10.3390/s20061771

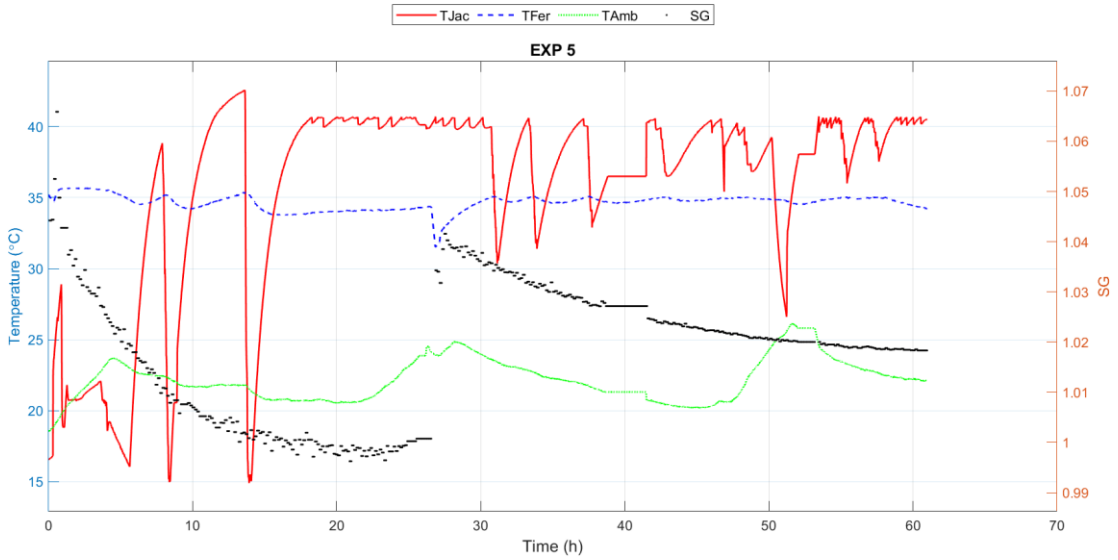
APPENDICES

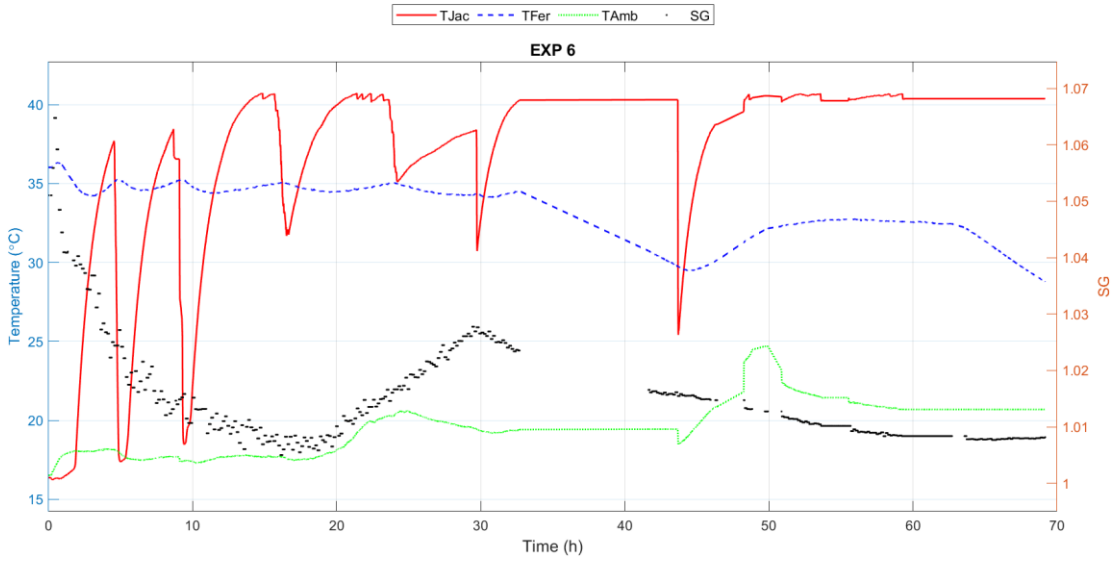
A1 – Temperature & Specific Gravity Profile for Each Experiment

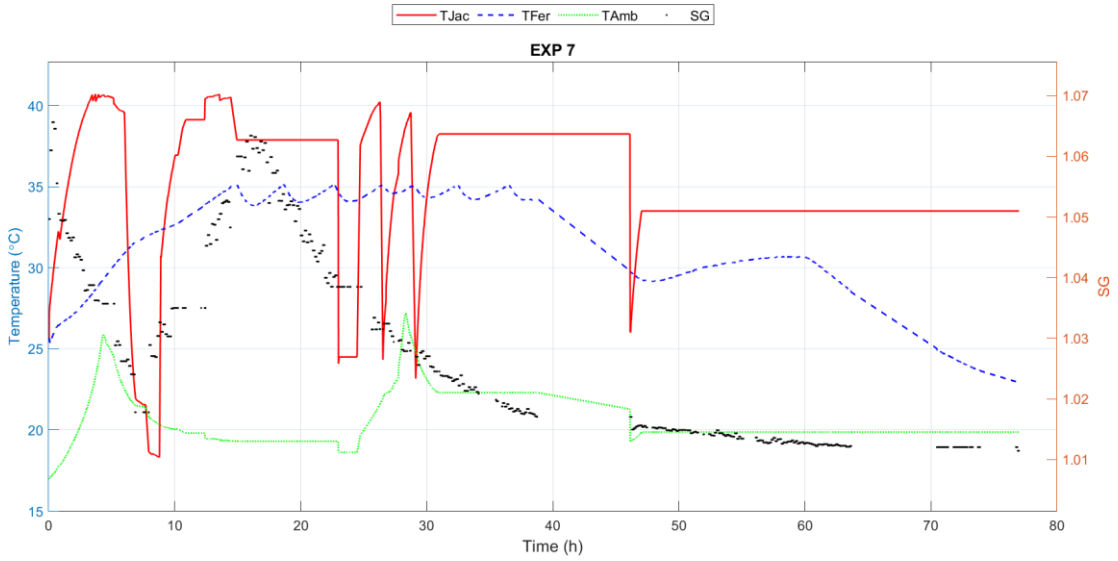


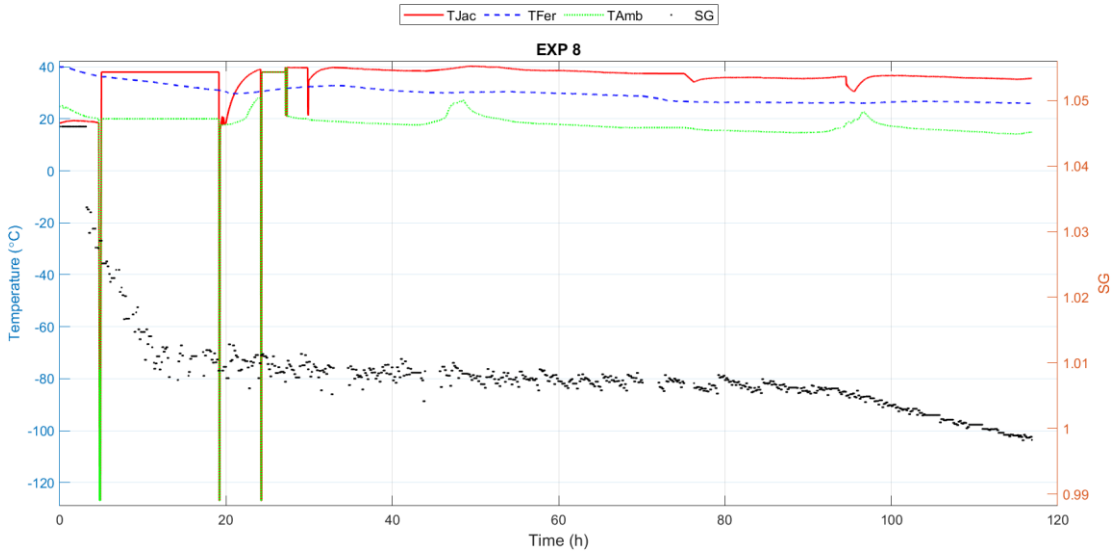


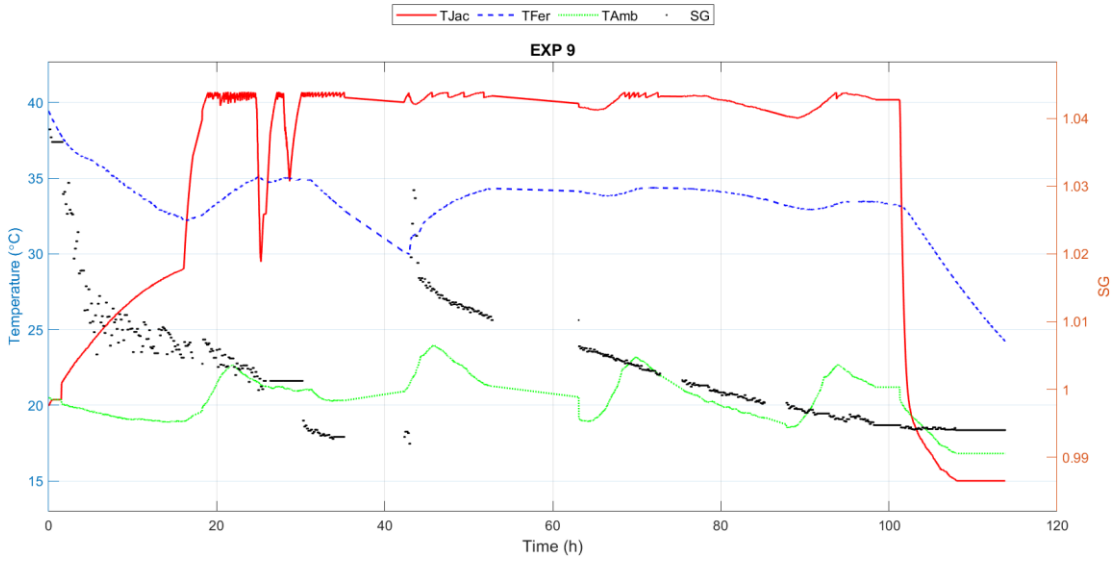

TJac
TFer
TAmb
SG
EXP 4
Temperature (°C)
SG
Time (h)
0
10
20
30
40
50
60
70
80
20
25
30
35
40
1.01
1.02
1.03
1.04
1.05
1.06

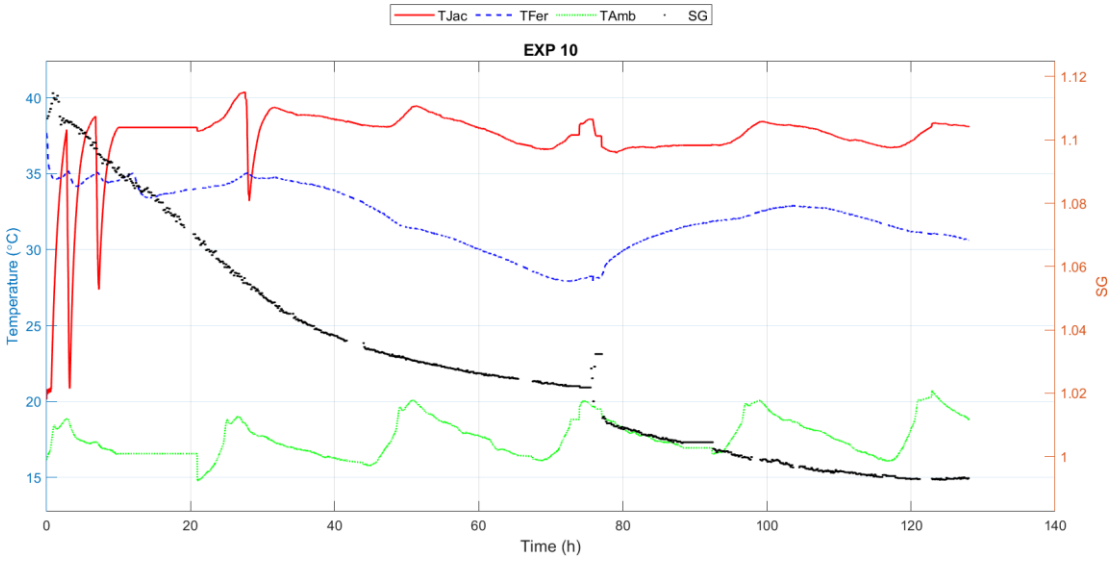

TJac
TFer
TAmb
SG
EXP 5
Temperature (°C)
SG
Time (h)
0
10
20
30
40
50
60
70
15
20
25
30
35
40
0.99
1
1.01
1.02
1.03
1.04
1.05
1.06
1.07


TJac
TFer
TAmb
SG
EXP 6
Temperature (°C)
SG
Time (h)
15
20
25
30
35
40
1
1.01
1.02
1.03
1.04
1.05
1.06
1.07
0
10
20
30
40
50
60
70

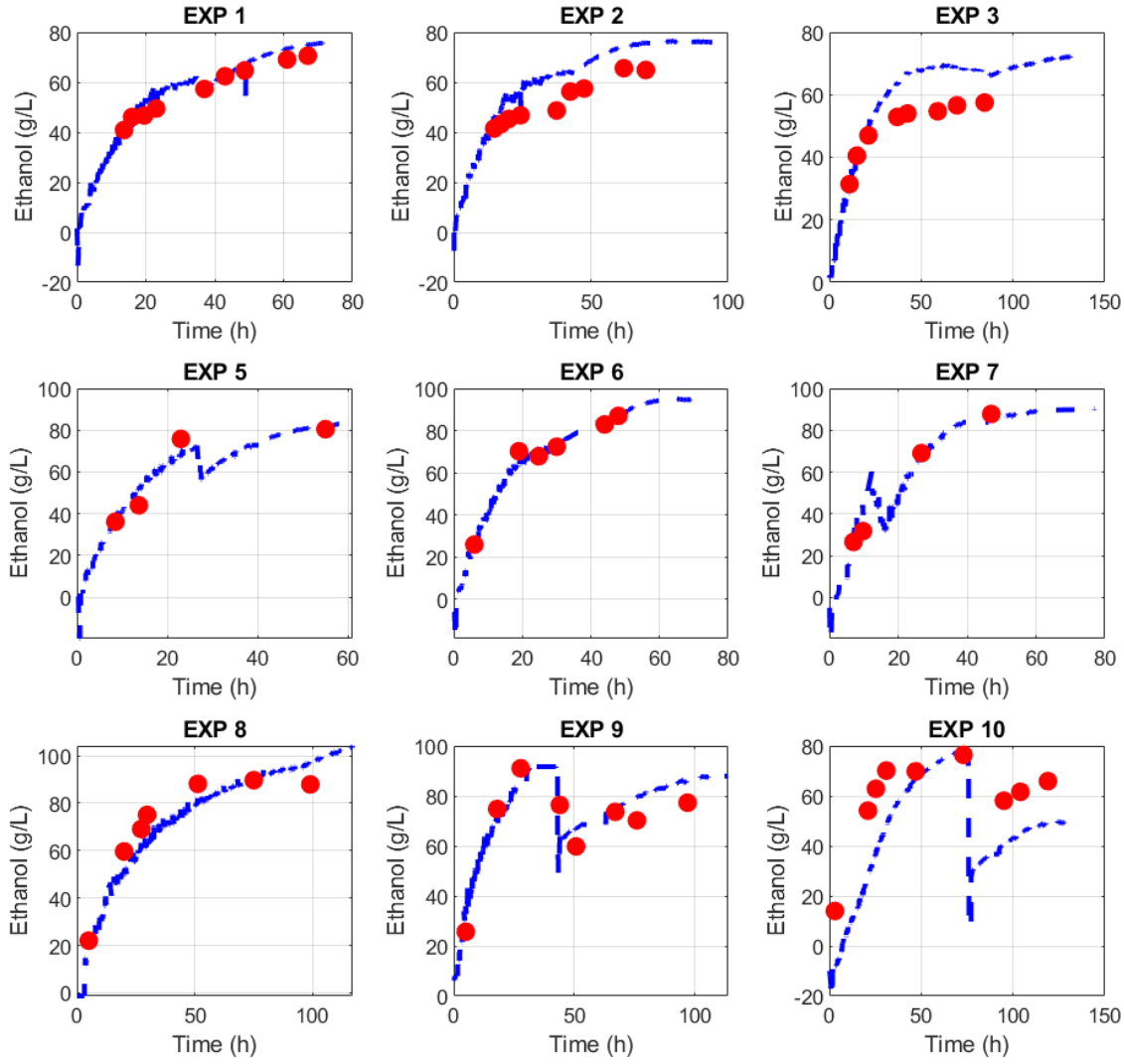

TJac
TFer
TAmb
SG
EXP 7
Temperature (°C)
SG
Time (h)
15
20
25
30
35
40
1.01
1.02
1.03
1.04
1.05
1.06
1.07
0
10
20
30
40
50
60
70
80


TJac
TFer
TAmb
SG
EXP 8
Temperature (°C)
SG
Time (h)


TJac
TFer
TAmb
SG
EXP 9
Temperature (°C)
SG
Time (h)


TJac
TFer
TAmb
SG
EXP 10
Temperature (°C)
SG
Time (h)
40
35
30
25
20
15
1.12
1.1
1.08
1.06
1.04
1.02
1
0
20
40
60
80
100
120
140

A2 – Modelled Ethanol Concentration from SG Measurements Validation



A3 – Soft Sensor Data Preprocessing Code

```
expNumbersSS = 1:10;
allDataSS = table();

for expNumSS = expNumbersSS
    % Read data
    TSS = readtable("TEMPETH.xlsx", "Sheet", "EXP " + expNumSS);

    % Extract columns
    timeSS = table2array(TSS(:,1));
    tjackSS = table2array(TSS(:,2));
    tfermSS = table2array(TSS(:,3));
    tambSS = table2array(TSS(:,4));
    ethSS = table2array(TSS(:,6));

    % Interpolate to hourly points
    timehSS = (0:max(timeSS))';
    tjackhSS = interp1(timeSS, tjackSS, timehSS);
    tfermhSS = interp1(timeSS, tfermSS, timehSS);
    tambhSS = interp1(timeSS, tambSS, timehSS);
    ethhSS = interp1(timeSS, ethSS, timehSS);

    % Smooth data before differentiation
    tjacksSS = smoothdata(tjackhSS);
    tfermsSS = smoothdata(tfermhSS);
    tambsSS = smoothdata(tambhSS);
    ethsSS = smoothdata(ethhSS);

    % Calculate Deltas (differentiation)
    dtjackSS = diff(tjacksSS);
    dtfermSS = diff(tfermsSS);
    dtambSS = diff(tambsSS);
    dethSS = diff(ethsSS);

    % Apply stronger smoothing to the delta values
    %smoothWindowSS = 10;
    dtjackSS = smoothdata(dtjackSS);
    dtfermSS = smoothdata(dtfermSS);
    dtambSS = smoothdata(dtambSS);
    dethSS = smoothdata(dethSS);

    % NEW FEATURE: Cumulative Heat Load
    cumulativeHeatSS = cumsum(dtjackSS);

    % Trim time vector to match diff size
    timehSS = timehSS(2:end);

    % Build table for this experiment
    expTableSS = table(repmat(expNumSS, size(timehSS)), timehSS, ...
```

```
                    dtjackSS, dtfermSS, dtambSS, cumulativeHeatSS, dethSS,
...
                    'VariableNames', {'Experiment', 'Time', ...
                                      'dT_Jacket', 'dT_Fermenter',
'dT_Ambient', 'Cumulative_Heat', 'dEthanol'});

    % Append to full dataset
    allDataSS = [allDataSS; expTableSS];
end

% Define training and testing experiments
trainExpSS = [1, 2, 3, 4, 6, 7, 9];
testExpSS = [5, 8, 10];

% Include Cumulative Heat Load in Training
TRAIN_IN_SS = allDataSS(ismember(allDataSS.Experiment, trainExpSS),
{'dT_Jacket', 'dT_Fermenter', 'dT_Ambient', 'Cumulative_Heat'});
TRAIN_OUT_SS = allDataSS(ismember(allDataSS.Experiment, trainExpSS),
'dEthanol');
TRAIN_SS = [TRAIN_IN_SS TRAIN_OUT_SS];

TEST_IN_SS = allDataSS(ismember(allDataSS.Experiment, testExpSS),
{'dT_Jacket', 'dT_Fermenter', 'dT_Ambient', 'Cumulative_Heat'});
TEST_OUT_SS = allDataSS(ismember(allDataSS.Experiment, testExpSS),
'dEthanol');
TEST_SS = [TEST_IN_SS TEST_OUT_SS];

% Save combined data for reference
writetable(allDataSS, "ProcessedData_CumulativeHeat_SS.xlsx")

% Save TRAIN table for export_metrics.m
save('ProcessedData_Combined_CumulativeHeat_SS.mat', 'TRAIN_SS');
```

A4 – Raw Training RMSE Data for Delta and Absolute Scenarios

Model Family	Model Type	ABS Validation RMSE	ABS Test RMSE	DELTA Validation RMSE	DELTA Test RMSE
Gaussian Process Regression	Rational Quadratic GPR	9.5526	28.202	0.69692	1.1761
Gaussian Process Regression	Matern 5/2 GPR	9.5606	28.836	0.70727	1.1709
Gaussian Process Regression	Squared Exponential GPR	9.5887	27.483	0.71742	1.1667
Gaussian Process Regression	Exponential GPR	9.6593	27.184	0.69797	1.181
Ensemble	Bagged Trees	10.905	23.387	0.69797	1.1147
SVM	Fine Gaussian SVM	11.143	24.812	0.72232	1.1661
Ensemble	Boosted Trees	11.217	25.636	0.71014	1.1234
Neural Network	Wide Neural Network	11.817	137.13	1.2609	2.1079
Neural Network	Trilayered Neural Network	11.342	38.236	1.1504	2.206
Tree	Medium Tree	11.618	23.501	0.75868	1.1344
Kernel	Least Squares Regression Kernel	11.641	32.397	0.7202	1.1032
Neural Network	Medium Neural Network	11.405	57.765	0.87684	1.1618

SVM	Medium Gaussian SVM	11.928	23.932	0.74233	1.1439
Tree	Coarse Tree	12.049	26.418	0.73931	1.0981
Neural Network	Bilayered Neural Network	12.119	121.35	0.84922	1.2889
Tree	Fine Tree	12.13	31.135	0.78447	1.2104
Neural Network	Narrow Neural Network	12.454	20.389	0.75005	1.4512
SVM	Cubic SVM	12.68	64.548	1.0533	10.736
SVM	Coarse Gaussian SVM	12.92	22.508	0.81669	1.1356
Stepwise Linear Regression	Stepwise Linear	13.055	23.611	0.82944	1.1834
SVM	Quadratic SVM	13.076	33.781	0.84638	1.4673
Linear Regression	Interactions Linear	13.175	23.059	0.8286	1.1833
Linear Regression	Linear	13.274	22.426	0.82029	1.1948
Linear Regression	Robust Linear	13.455	24.101	0.83659	1.2081
SVM	Linear SVM	13.628	25.389	0.86833	1.1724
Kernel	SVM Kernel	16.47	25.83	0.74465	1.148

A5 – Soft Sensor Raw Output Code

```
%% MATLAB Script to Evaluate Model Performance on Test Data
% Load test data (assumes the test data is in table format)
% Using test data directly from workspace
TEST_IN_SS = TEST_SS(:, 1:end-1);
TEST_OUT_SS = TEST_SS.dEthanol;

% Define model names and full titles
model_names = {'GPR', 'EN', 'SVM', 'TREE', 'NN', 'KER', 'LR', 'SLR'};
model_full_titles = {'Gaussian Process Regression', 'Ensemble', 'Support
Vector Machine', 'Decision Tree', 'Neural Network', 'Kernel', 'Linear
Regression', 'Stepwise Linear Regression'};

% Initialize result containers
results = struct();

% Loop through each model to predict and calculate RMSE and R²
for i = 1:length(model_names)
    model = eval(model_names{i});
    y_pred = model.predictFcn(TEST_IN_SS);
    y_actual = TEST_OUT_SS;

    % Calculate delta ethanol and RMSE
    delta_ethanol = y_actual - y_pred;
    RMSE = sqrt(mean(delta_ethanol.^2));
    R2 = 1 - sum((y_actual - y_pred).^2) / sum((y_actual -
mean(y_actual)).^2);

    % Store results
    results.(model_names{i}).y_pred = y_pred;
    results.(model_names{i}).RMSE = RMSE;
    results.(model_names{i}).R2 = R2;

    % Plotting Actual vs Predicted (Delta Ethanol)
    figure;
    subplot(1, 2, 1);
    time_vector = (1:length(y_actual))';
    plot(time_vector, y_actual, 'k', 'LineWidth', 1.5); hold on;
    plot(time_vector, y_pred, 'r--', 'LineWidth', 1.5);
    xlabel('Time (hours)'); ylabel('\Delta Ethanol (g/L)');
    % Unified title for both plots
% Centered title between both graphs
sgtitle(model_full_titles{i});
    legend('Actual \Delta Ethanol', 'Predicted \Delta Ethanol');
    grid on;

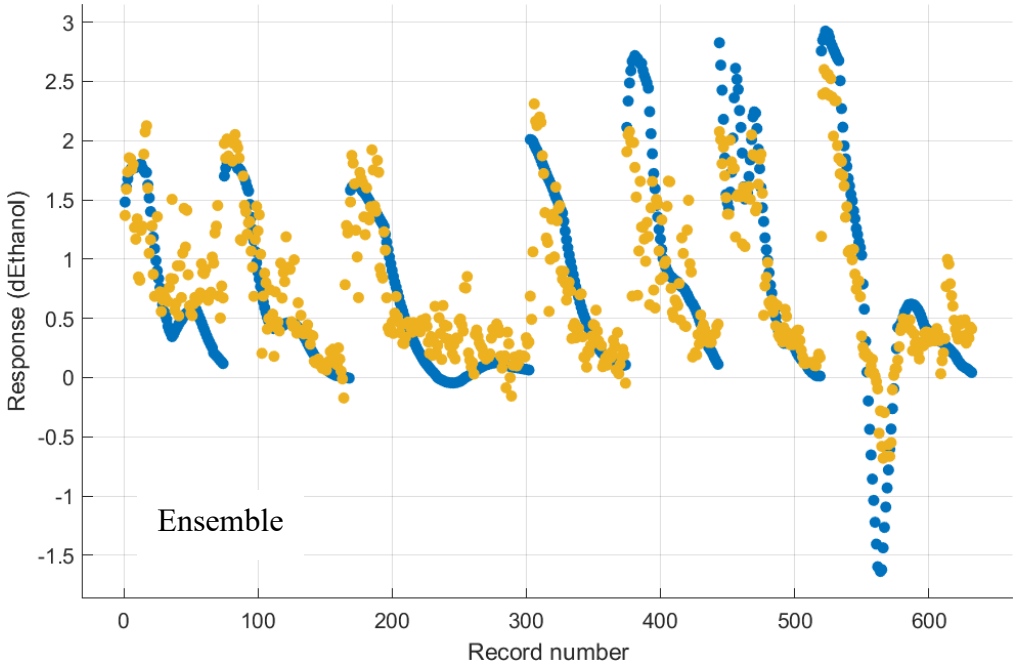
    % Parity Plot
    subplot(1, 2, 2);
    scatter(y_actual, y_pred, 36, 'r', 'o');  % Red hollow circles
```

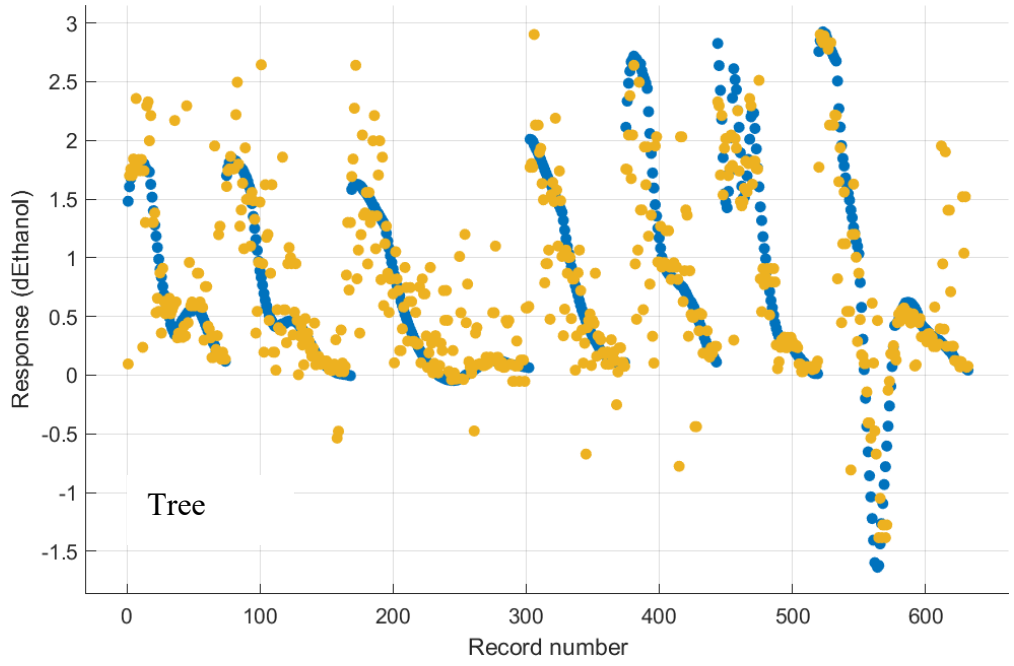
```
    hold on; plot([min(y_actual), max(y_actual)], [min(y_actual),
max(y_actual)], 'k--');
    xlabel('Actual \Delta Ethanol'); ylabel('Predicted \Delta Ethanol');
    % Remove individual subplot titles to keep one unified title

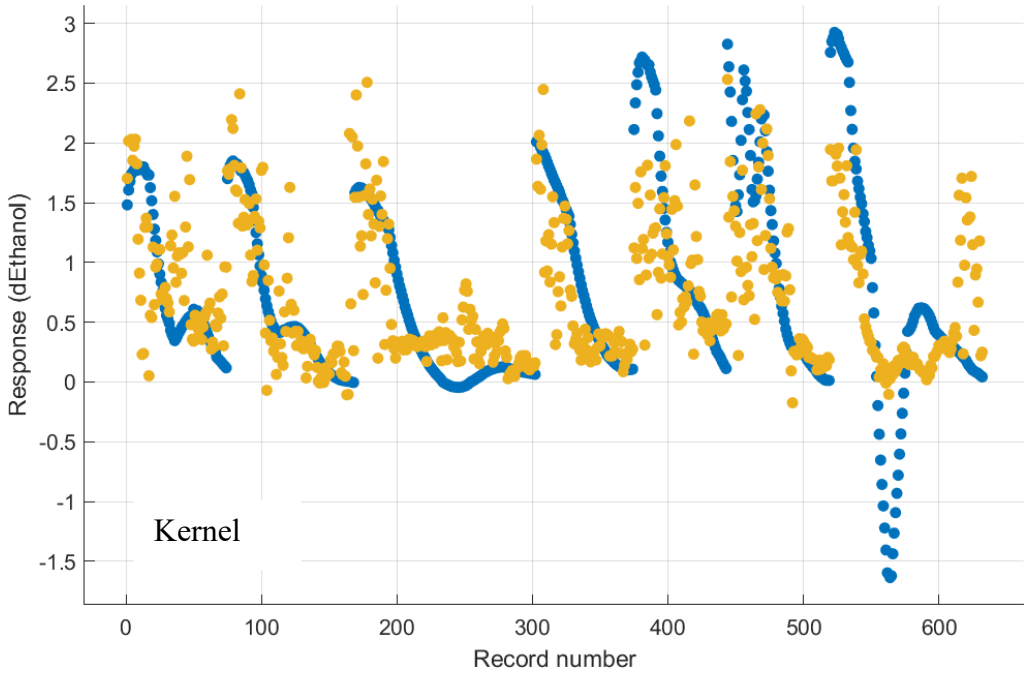
    text(min(y_actual), max(y_pred), {['RMSE: ', num2str(RMSE)], ['R²: ',
num2str(R2)]}, 'FontSize', 10, 'BackgroundColor', 'none');
    grid on;
end

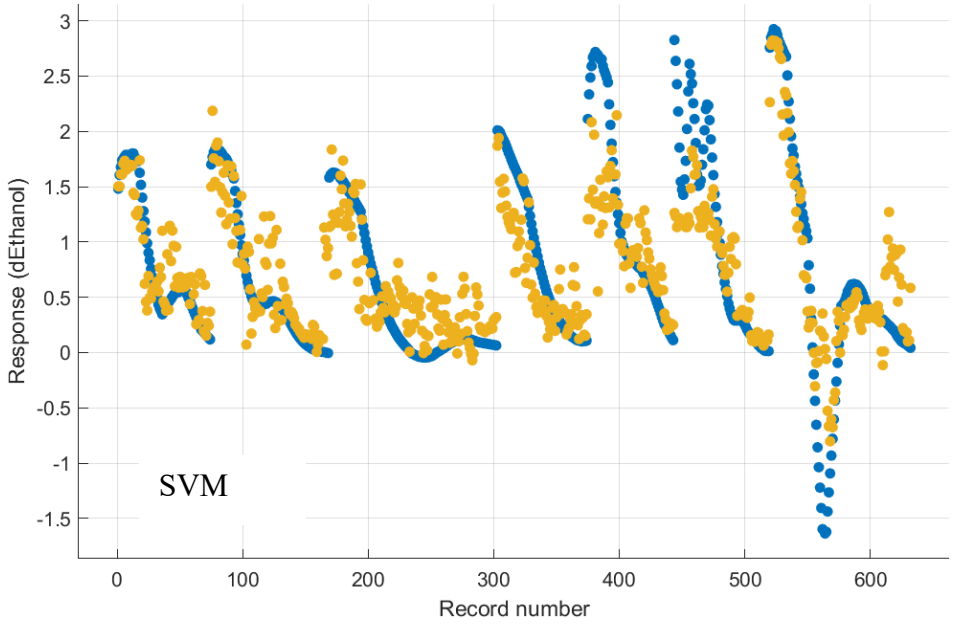
disp('Model performance analysis completed.');
```

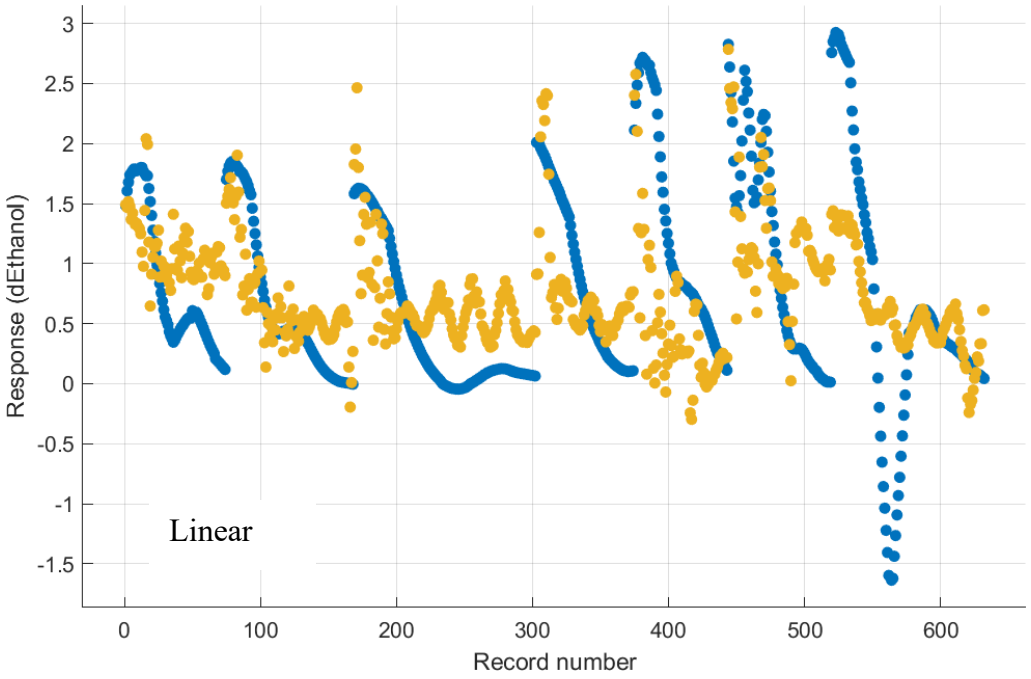
A6 – Top Performing Model by Family Predictive vs Actuals Validation

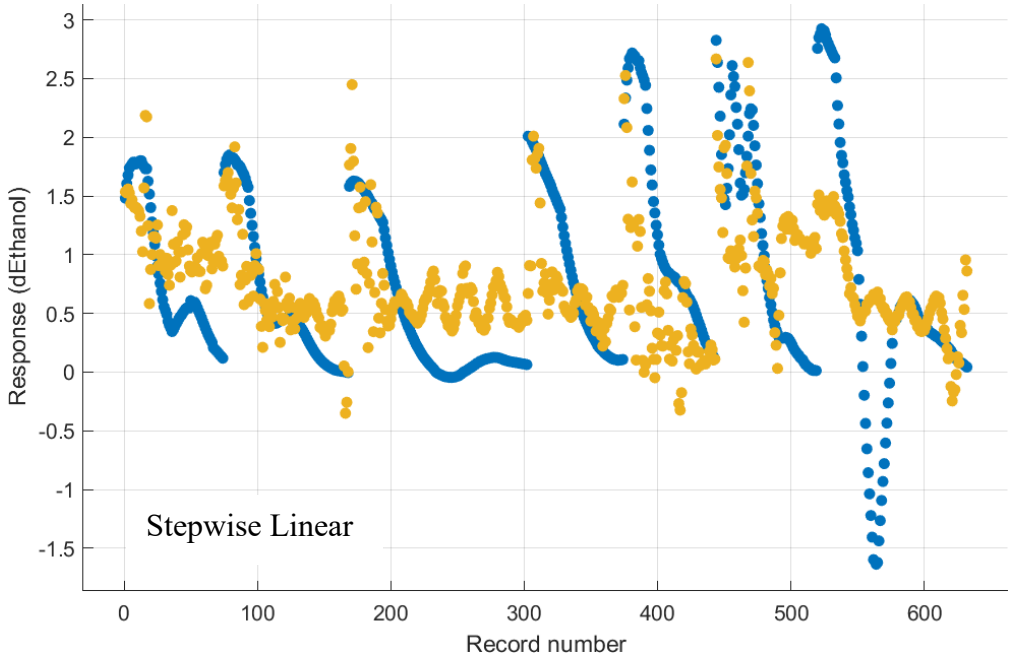


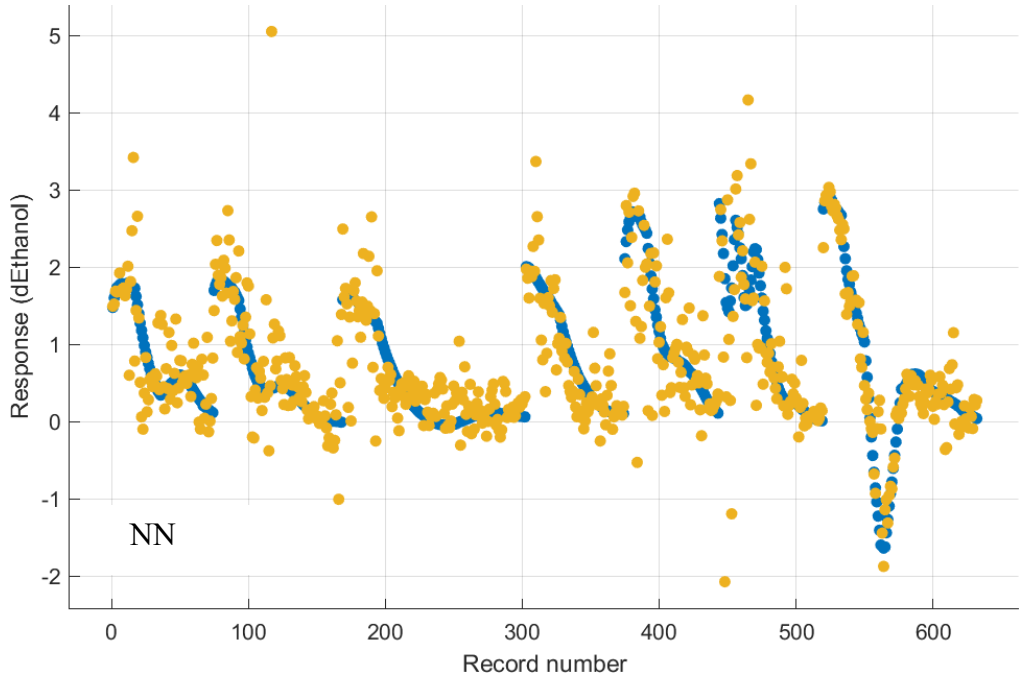



Response (dEthanol)
Record number
Kernel
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
100
200
300
400
500
600


Response (dEthanol)
Record number
SVM
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
100
200
300
400
500
600


Response (dEthanol)
Record number
Linear
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
100
200
300
400
500
600


Response (dEthanol)
Record number
Stepwise Linear
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
100
200
300
400
500
600


Response (dEthanol)
Record number
NN
5
4
3
2
1
0
-1
-2
0
100
200
300
400
500
600

A7 – Variable Smoothing Ethanol Prediction Code

```
%% MATLAB Script to Evaluate Smoothed Model Performance on Test Data
% Uses smoothed predicted delta ethanol with different smoothing windows

% Using test data directly from workspace
TEST_IN_SS = TEST_SS(:, 1:end-1);
TEST_OUT_SS = TEST_SS.dEthanol;

% Define model names
model_names = {'GPR', 'EN', 'SVM', 'TREE', 'NN', 'KER', 'LR', 'SLR'};

% Smoothing windows
smooth_windows = [10, 20, 30, 40];

% Initialize result containers
results = struct();

% Loop through each model to predict and calculate RMSE and R²
for i = 1:length(model_names)
    model = eval(model_names{i});
    y_pred = model.predictFcn(TEST_IN_SS);
    y_actual = TEST_OUT_SS;

    % Plotting Actual vs Predicted (Delta Ethanol)
    figure;
    time_vector = (1:length(y_actual))';
    plot(time_vector, y_actual, 'k', 'LineWidth', 1.5); hold on;
    plot(time_vector, y_pred, 'r--', 'LineWidth', 1.5);
    xlabel('Time (hours)'); ylabel('\Delta Ethanol (g/L)');
    legend_entries = {'Actual \Delta Ethanol', 'Predicted \Delta Ethanol'};

    % Loop through each smoothing window and plot the smoothed predictions
    for j = 1:length(smooth_windows)
        y_pred_smooth = smoothdata(y_pred, 'movmean', smooth_windows(j));
        RMSE = sqrt(mean((y_actual - y_pred_smooth).^2));
        R2 = 1 - sum((y_actual - y_pred_smooth).^2) / sum((y_actual -
mean(y_actual)).^2);

        % Plot the smoothed prediction
        colorOrder = get(gca, 'ColorOrder');
        colorIndex = mod(j-1, size(colorOrder, 1)) + 1;
        plot(time_vector, y_pred_smooth, 'LineWidth', 1.5, 'Color',
colorOrder(colorIndex, :));

        % Add RMSE and R² directly in the legend entry
        legend_entries{end+1} = sprintf('Smoothed (Window %d) - RMSE: %.4f,
R²: %.4f', ...
                                        smooth_windows(j), RMSE, R2);
    end
```

```
    % Finalize plot with combined legend
    legend(legend_entries, 'Location', 'southwest');

    % Model full name mapping
    full_model_name = model_names{i};
    switch full_model_name
        case 'GPR', full_model_name = 'Gaussian Process Regression';
        case 'EN', full_model_name = 'Ensemble';
        case 'SVM', full_model_name = 'Support Vector Machine';
        case 'TREE', full_model_name = 'Decision Tree';
        case 'NN', full_model_name = 'Neural Network';
        case 'KER', full_model_name = 'Kernel';
        case 'LR', full_model_name = 'Linear Regression';
        case 'SLR', full_model_name = 'Stepwise Linear Regression';
    end

    % Title with model name
    title([full_model_name, ' - Actual and Predicted Delta Ethanol with
Smoothing']);
    grid on;
end

disp('Smoothed model comparison analysis completed.');
```