Machine Learning Modeling of a Direct-Injected Dual-Fuel Engine Based on Low Density Experimental Data

by

Michael Karpinski-Leydier

B.A.Sc., The University of Waterloo, 2010

M.A.Sc, The University of Waterloo 2012

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

August 2021

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the dissertation entitled:

Machine Learning Modeling of a Direct-Injected Dual-Fuel Engine Based on Low Density Experimental Data

submitted by Michael Karpinski-Leydier in partial fulfillment of the requirements for

the degree of Doctor of Philosophy

in Mechanical Engineering

Examining Committee:

Dr. Patrick Kirchen, Mechanical Engineering, UBC
Co-supervisor

Dr. Ryozo Nagamune, Mechanical Engineering, UBC
Co-supervisor

Dr. Steven Rogak, Mechanical Engineering, UBC
Supervisory Committee Member

Dr. Rajeev Jaiman, Mechanical Engineering, UBC
University Examiner

Dr. Anthony Lau, Chemical and Biological Engineering, UBC
University Examiner

Additional Supervisory Committee Members:

Dr. Bhushan Gopaluni, Chemical and Biological Engineering, UBC
Supervisory Committee Member

Dr. Rudolf Seethaler, Mechanical Engineering, UBC
Supervisory Committee Member

Abstract

Automotive systems are constantly increasing in complexity, requiring advanced modeling methods with large data sets to analyze these systems. This work proposes a machine learning approach to rapidly developing, steady state, control oriented, engine models that use optimization methods and engineering knowledge to reduce the burden of data collection and improve model performance and reliability. Data is collected from a pilot ignited direct injection natural gas engine using a full factorial approach for a high density data set and a design of experiments approach for a low density training data set with randomized validation data. An optimization approach for selecting hyperparameters for neural network and Gaussian process regression models is proposed. Models for emissions and performance metrics are created and compared to response surface models. The hyperparameter optimized models show an improvement in robustness and model performance, reducing the normalized root mean square error by 26% compared to other hyperparameter configurations. Gaussian process regression hyperparameter optimization shows the lowest error, 46% lower than response surface models. The Gaussian process regression hyperparameter optimized models are further improved using multi-region modeling, sensitivity analysis based input reduction, layered modeling, and hybrid layered modeling. The sensitivity based input reduction reduces the normalized root mean square error for all models by an average of 8% and up to 19%. The layered models reduce the normalized root mean square error for the CO by 52%, NO_X by 30%, and particulate matter by 33%. The multi-region models reduce the normalized root mean square error for the O_2 by 40% and thermal efficiency by 16%. Using the best techniques for each output, the error is reduced by 19%, compared to hyperparameter optimization alone and 45% compared to typical Gaussian process regression models. These results show that hyperparameter optimization combined with the other techniques presented here

significantly reduce model error. Using these techniques, it is possible to reduce the reliance on data for engine modeling. Future research in energy conversion technologies can use these techniques to rapidly develop new technologies without the cost in time and funding typically reserved for extensive data collection.

Lay Summary

Modern automotive research has resulted in the development of complex systems. Significant data collection is needed to learn how to make predictions on engine performance. This dissertation proposes a machine learning approach to modeling that uses optimization, data sciences, and engineering intuition to predict engine performance with less than a month of data. The Gaussian process regression machine learning technique with optimization to configure model settings significantly improves the reliability and performance compared to other methods. The performance can further be improved by using data science methods to remove unneeded data and engineering knowledge to either separate data into different groups or organize models into layers based on other important parameters than can be predicted and used as inputs. Using these techniques, the performance of the emissions and outputs, such as power and efficiency, can be predicted with more accuracy and less data than was previously needed.

Preface

This dissertation is prepared by the author as part of the graduation requirements for the PhD Graduate Studies program in the Mechanical Engineering Department at The University of British Columbia Vancouver Campus. This dissertation contributes to the academic world in the form of developing a machine learning, reduced data set, modeling approach for engine research utilizing Gaussian process regression, hyperparameter optimization, sensitivity based input reduction, multi-region modeling, layered modeling, and hybrid layered modeling.

In December 2015, the test bench that was to be used in this research was rebuilt due to a hardware failure. Most of the mechanical rebuild was undertaken by Robert Parry, Aditya Singh, Rene Zepeda and Pooyan Khierkhah with the assistance of the author. During this time, this author was also responsible for the development of a new control and data acquisition system to operate the test bench and manage both data collection and data post processing. This author has since been responsible for the upkeep and maintenance of the test bench throughout this work as well as any data collection taken as part of this research. The only exception to this is the data used in the full factorial set data discussed in Section 3.2.1. This data set was collected by Aditya Singh as part of his research and repurposed for this work. All analysis and further usage of this data beyond its original collection is the work of the author of this dissertation. All model development, assessment and analysis is the work of the author of this dissertation.

Some of the work presented here has been previously published. Chapter 5 contains research that has previously been published by this author in conjunction with Dr. Patrick Kirchen and Dr. Ryozo Nagamune in an SAE Technical Paper with the title: “A Machine Learning Modeling

Approach for High Pressure Direct Injection Dual Fuel Compressed Natural Gas Engines." This paper was presented at the 2020 SAE Powertrains, Fuels & Lubricants Meeting. This author was responsible for the data collection and analysis of the work presented in this section. Dr. Kirchen and Dr. Nagamune both supervised the work and provided valuable guidance for the research. Chapter 6 represents work that is currently under review for publication as an SAE Technical Paper. As with the other work, this author was responsible for the data collection, modeling, and analysis while Dr. Kirchen and Dr. Nagamune provided supervision and guidance for the work.

Table of Contents

List of Tables

List of Figures

List of Symbols

cov	Covariance
dp/dCA	Cylinder pressure gradient (bar/°CA)
f^*	Test output for Gaussian process regression model
$H_2O_{\%}$	Percentage of oxygen trapped in water (%)
K	Gaussian process regression covariance kernel
m_{air}	Air mass (kg)
m_{diesel}	Diesel fuel mass (kg)
$\dot{m}_{dil}$	Dilution air mass flow rate (kg/hr)
m_f	Total equivalent fuel mass (kg)
m_{ng}	Natural gas fuel mass (kg)
m_{O_2}	Oxygen mass (kg)
N	Artificial neural network node
n_{data}	Number of measurements
n_{nodes}	Number of nodes in the previous layer
$n_{perturb}$	Number of perturbations for one input
O_{frac}	Mass fraction of oxygen in air
p_{air}	Intake air pressure (kPa)
p_{cyl}	Peak cylinder pressure (bar)
p_{diesel}	Diesel injection pressure (MPa)
p_{exh}	Exhaust back pressure (kPa)
P_{gross}	Gross indicated power (kW)

p_{ng}	Natural gas injection pressure (MPa)
Pt	Input sensitivity perturbation (%)
PW_{diesel}	Diesel injection pulse width (ms)
PW_{ng}	Natural gas injection pulse width (ms)
s	Standard deviation
S_j	Sensitivity of the jth input for the given output
s_{meas}	Standard deviation of experimentally measured values
s_{norm}	Normalized standard deviation
SOI_{diesel}	Diesel start of injection (°CA)
SOI_{ng}	Natural gas start of injection (°CA)
s_{pred}	Standard deviation of model predicted values
st	Stoichiometric ratio
T_{air}	Air temperature (°C)
T_{cool}	Coolant temperature (°C)
u_{egr}	Exhaust gas recirculation valve position (%)
w	Weighted artificial neural network node to node connection
X	Model training data
X_{new}	Sensitivity modified input
X^*	Test point data
Y_{meas}	Experimentally measured output
Y_{new}	Sensitivity modified predicted output
Y_{pred}	Model predicted output

η	Thermal efficiency (%)
θ_{10}	Angle of 10% heat release rate (°CA)
θ_{50}	Angle of 50% heat release rate (°CA)
$\mu(x)$	Gaussian process regression mean function
σ^n	Gaussian process regression gaussian signal noise
ω_{eng}	Engine speed (rpm)

List of Abbreviations

ANN	Artificial neural network
CH_4	Methane
CO	Carbon monoxide
CO_2	Carbon dioxide
COD	Coefficient of determination
EGR	Exhaust gas recirculation
EQR	Equivalence ratio based on air
EQRO	Equivalence ratio based on oxygen
ESD	Emergency shut down
GPR	Gaussian process regression
HPDI	High pressure direct injection
LHV	Lower heating value
NO_X	Nitrogen oxide and nitrogen dioxide
nRMSE	Normalized root mean square error
PIDING	Pilot ignited direct injection natural gas
PM	Particulate matter
PSEP	Pulse separation time between fuel injections
RIT	Relative injection timing
RMSE	Root mean square error
RS	Response surface
SCRE	Single cylinder reciprocating engine

Acknowledgements

I would like to thank the faculty, staff, and my fellow students at UBC who have helped and guided me through my time as a PhD student. Completing a PhD is a difficult task that cannot be accomplished alone. I have been fortunate to have the support of many people throughout the years.

I owe particular thanks to my supervisor, Dr. Patrick Kirchen, whose exceptional support and guidance has continually helped progress this work towards its completion. Without Dr. Kirchen's mentorship, expertise, and helping hand, including working on the test bench, none of this work would have been possible.

I would also like to thank my co-supervisor, Dr. Ryozo Nagamune, who helped guide me throughout this research and my time at UBC, providing valuable advice and experience. I also want to thank my committee members; Dr. Steven Rogak, Dr. Bhushan Gopaluni, and Dr. Rudolf Seethaler, who all helped direct my research over the years.

I would also like to directly thank my colleagues; Rene Zepeda Gutierrez, Aditya Prakash Singh, Jeff Son, Jeff Meiklejohn, Troy Hurren, Mahdiar Khosravi, Jeremy Rochussen, Pooyan Kheirkhah, Jeff Yeo, Isaac Becker, and Matthew Knight. Each of these people has played an important part in supporting my research and helping me see it through to the end. Additionally, I would like to thank the university support staff including Bob Perry, Glenn Jolly and Sean Buxton whose extensive knowledge and expertise directly contributed to ensuring a safe and working test bench.

I would also like to thank the various funding agencies that supported my work over the years including NSERC CGS-D, Discovery Grant, CREATE Clean Combustion Engines, Westport Fuel Systems, and the British Columbia Graduate Scholarship.

Finally, I would like to thank my parents, who have supported me throughout the many years of my education in more ways than I can ever express in words.

Thank you to everyone who made this work possible.

Dedication

I dedicate this work to my family. It was through their guidance and continual support that I have accomplished so much in my life. This dissertation and the degree that comes with it are just one part of what I have been able to do with their incredible support.

Chapter 1: Introduction

The automotive industry is one of the larger industries in the world [1]. With few exceptions, people's daily lives are influenced by the automotive sector. From streetlights and crosswalks, to traffic laws and the importance of having a driver's license, the automobile permeates many aspects of life. Therefore, it no surprise that the automotive industry is one of the faster moving, research driven industries [1], [2].

Not every aspect of the automotive industry has improved people's lives. While the automotive industry has contributed to the creation of an interconnected world, it has also contributed to climate change [2]. Science has shown that the mass burning of fossil fuels releases CO_2 and CH_4 which are the leading contributors to climate change [1]. While the amount of CH_4 being emitted is much lower than CO_2 on average, CH_4 has a global warming potential that is up to 36 times that of CO_2 [3]. This indicates that both these emissions are very dangerous for the environment and need to be reduced. New technologies are constantly being developed to limit the automotive industry's impact on the environment. This is the focus of the majority of automotive research [2]. Many different technologies have been proposed to address the climate change problem. Some alternative fuel technologies have examined combusting natural gas in either spark or compression ignition systems and have shown to significantly reduce emissions of harmful greenhouse gasses [4]–[7]. Technologies such as hydrogen fuel cells provide a potential zero emission energy conversion technology for the automotive industry [8], [9]. Each of these technologies requires an initial investment of time and funding to facilitate their development and implementation. This is a limiting factor to the mass adoption for many of these technologies [10]. The initial investment allows for the initial research phase, the early design prototyping and testing phases leading all the

way to the final modeling and calibration phase. This final phase is important as any new technology needs to be utilized properly to gain the desired benefits for which it was developed. This necessitates modeling the new technology to enable the understanding of how the new system integrates with the existing usage cases for the vehicle. Once these models are understood, an engineer can develop system calibrations to facilitate the actual use of the system in such a way so that its benefits can be realized. The key element in this process is the model.

1.1 Limitations of Conventional Engine Modeling

Model development can be a costly and time consuming process. The number of control inputs, such as variable injection timings, pressure regulation on the exhaust and intake air, and exhaust gas recirculation, each contribute to modeling complexity [11]. The implementation of newer technologies, such as cylinder disconnect and either alternative or multiple fuels, further complicates the problem [12]–[15]. Many new technologies create controllable inputs that add additional degrees of freedom to the system control problem. This expands the size of the operating space [15]. To fully map out a large operating space using a brute force method requires a large amount of data which can be difficult to collect for a calibration engineer. One modeling technique used in automotive research programs to combat this issue is to divide the complete operating space into many smaller operating spaces and apply a response surface model to each region independently [14], [16], [17]. Many of these regions are defined to limit the number of inputs to reduce the size of the data set that is needed. This results in each small region having its own data set [16]. Even with this technique, this is still a daunting task requiring months or years of testing to complete before a system is properly modeled and calibrated. This lengthy process can be costly. This can also extend the timeline to developing the new automotive technologies needed to meet

emissions requirements. It is for this reason that different approaches to automotive system modeling are being studied.

1.2 Machine Learning for Engine Modeling

One method of model development currently being studied is machine learning. Many conventional modeling methods have their basis in fundamental mathematics and physics [11]. These methods use a human understanding of the scientific principles that define the systems being modeled to create complex algorithms using techniques such as computational fluid dynamics and physics based equations to solve the modeling problem [12], [18]–[20]. Machine learning takes a very different approach. At its core, machine learning is about programmatically identifying patterns in data sets [21]. A properly configured machine learning algorithm can be given a large number of high quality data points and can use them to learn to identify the trends in the data. A user could then ask the algorithm to predict the output for a new set of inputs and the algorithm would to be able to accomplish this task by having been exposed to similar data points [22]. Noisy data or data sets that lack the quantity or variety of data required for the algorithm to capture a compete picture of the desired patterns can limit the capabilities of machine learning [22]. This is one of the limitations of machine learning algorithms [23]. Despite this drawback, machine learning has found its way into a number of varying fields including battery cell health, acoustic modeling, rainfall analysis, sediment load studies, and stock market index prediction, as well as engine research [24]–[29]. Many of these fields can rely of having an abundance of data. This is one of the limitations of machine learning in the automotive field. While researchers have applied machine learning techniques to automotive engine research, much of this work has been focused on limited operating spaces where the size of the data set can be kept to a minimum [30]–[32]. As

each engine system is unique and many of the technologies being developed change the inner workings of the engine system, old data from past engine technologies is often no longer relevant [7]. New data is therefore often needed for each new engine being modeled.

There are many different machine learning based models that have been developed over the years and applied to engine research. Artificial neural networks have traditionally been the better known and more commonly used machine learning method [33]. However, other methods such as Gaussian process regression have recently been shown to be useful for engine modeling [24], [34], [35].

1.2.1 Artificial Neural Networks

The artificial neural network is one of the better known machine learning modeling methods. Originally developed as a model for the interconnection between neurons in the brain, this method for modeling biological systems has evolved into an algorithm for creating data driven models of many different systems [36]. The artificial neural network is envisioned as a series of interconnected nodes organized into distinct layers [22]. The first layer of the network represents the inputs of a system while the final layer represents the outputs being modeled. The layers situated between the inputs and outputs are referred to as hidden layers. Each node in a layer is directly connected to each of the nodes in the previous and next layers through a weighted connection. Figure 1.1 shows an example of how a neural network can be visualized.

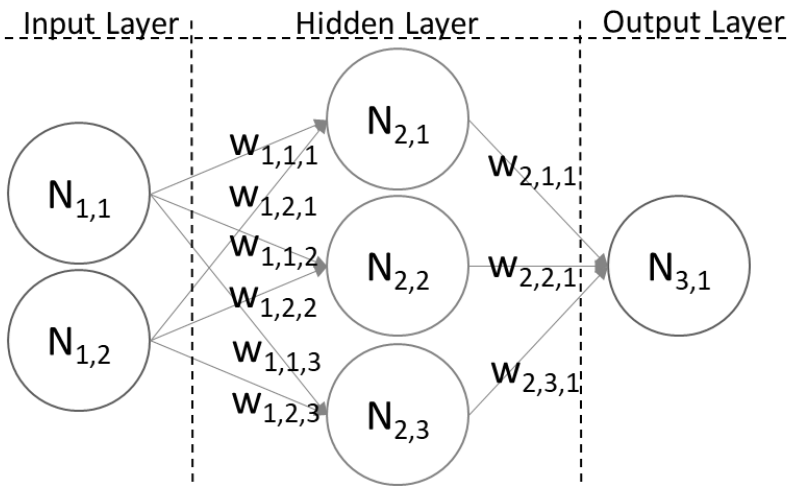


Figure 1.1: Neural Network Structure Example

As the nodes, N, in the input are assigned values, these values propagate through the weighted connections, w, to the subsequent layers as a weighted summation according to Equation 1.1 and the node's activation function. n_{nodes} is the number of nodes in the i-1 layer and i is the index of the current layer. j is the index of the current node being evaluated while k iterates through the nodes in the i-1 layer [37].

$$N_{i,j} = \sum_{k=1}^{n_{nodes}} \left[N_{i-1,k} \times w_{i-1,k,j} \right] \tag{1.1}$$

This determines the value of a given node. As the inputs propagate through the nodes, they eventually reach the output layer. The value of the nodes at the output layer is taken as the prediction of the output for that given input set. These weighted summations that occur throughout the layers approximate the complex interactions between the inputs [22].

Setting these weightings properly for each connection is an important part of developing a good neural network model. This is accomplished through a technique called training. Training a neural network is accomplished using vast quantities of data where the inputs and expected outputs to the model are already known. For each data point, the neural network makes a prediction and then compares the prediction to the expected output. This creates an error term that is then sent backwards through the neural network and is used to adjust the initially randomized weights so that they better fit the data point [22]. This is done for each training data point, creating minor adjustments each time. As this process is repeated for several iterations, the error terms begin to decrease as the weightings work to reflect the system being modeled more closely. Once a desired error threshold is met, the model is said to be trained [22]. Here is where the neural network is shown to be heavily data dependent. In order for this model to be accurate at predicting new data points, the training data points need to provide sufficient information with a low enough overall level of noise and erroneous data to enable the neural network to capture the trends in the data to a degree that reflects the actual system's operation.

Many different variations of neural network algorithms exist that have been used for engine research. Some of these techniques assign specific functions to specific layers such as in the case of object identification or classification which might use multiple layers to determine categories and subcategories for a given structure. Shahid et al, for example, used a neural network for classification of varying engine loads as part of a real-time control system to recalibrate an engine control scheme for different operating loads [38].

Neural networks are often used for predicting emissions measurements from various different types of engines [39], [40]. They are also commonly used to evaluate new alternative fuels. Jaliliantabar et al, used neural networks combined with a sensitivity analysis to predict engine parameters for a set of key operating conditions using varying compositions of biodiesel [41]. Similarly, Mao et al, used four different types of artificial neural networks to model brake specific fuel consumption, thermal efficiency, exhaust temperature, CO, hydrocarbons, NO_X, and particulate matter emissions from an ethanol-diesel engine [42]. In the same way, Cho et al, used neural networks to model the effects of hydrogen addition to a turbo-charged direct injection gasoline engine based on data initially collected by a GT power model and later verified with experimental results [43]. Each of these research projects demonstrates the capability of neural networks for modeling the outputs of alternative fuels in modern engines.

1.2.2 Gaussian Process Regression

Gaussian process regression is a relatively new technique in the engine modeling space compared to artificial neural networks [44]. Gaussian process regression has been applied in a variety of fields such as battery capacity, aircraft engine fuel flow, and recently, automotive engine modeling [29], [45], [46].

At its core, Gaussian process regression is a probabilistic modeling method. In the simplest terms, a Gaussian process regression model finds the most probable function that fits a given training data set and uses this function to determine the most likely output for a given input vector. This function weaves through known data points in a continuous fashion, which is why Gaussian

process regression modeling is often well suited to continuous processes as opposed to discrete functions [44].

The Gaussian process regression assumes that the parameters that define the function which represents the system being modeled can be represented as random variables and can themselves be modeled as a Gaussian distribution. This Gaussian distribution is defined by a covariance matrix and a mean function which can be subdivided into components based on the known training data and the test data point that is being evaluated. The key derivations of the model are shown in Equations 1.2 and 1.3 which are taken from the work in Chapter 2 of Rasmussen [44].

$$\mathrm{cov}(f^*) = K(X^*, X^*) - K(X^*, X)[K(X, X) + (\sigma^n)^2]^{-1}K(X, X^*) \tag{1.2}$$

$$\mu(f^*) = K(X^*, X)[K(X, X) + (\sigma^n)^2 I]^{-1}y \tag{1.3}$$

In Equations 1.2 and 1.3, μ is the mean and cov represents the covariance, corresponding to the test output, f^*. K is an element of the covariance matrix corresponding to X, the training data, and X^*, the test point being evaluated. σ^n is the Gaussian noise variance that is assumed to be a part of the training data and y represents the training data outputs. To complete the model, aside from populating it with training data, a value for σ^n must be selected and a function for K needs to be determined. This function, K, is called the kernel or covariance function. Many kernels have been proposed in the literature but are otherwise selected depending on the modeling problem [29], [35], [44]–[49].

Gaussian process regression has only just recently started to be used for engine control based modeling research. Aran et al, used a Gaussian process regression model to model the diesel engine air path as part of a calibratable controller architecture [50]. Similarly, Bergmann et al used Gaussian process regression to model the fuel pathway on a heavy duty diesel engine [49]. Berger et al, took the Gaussian process regression model and developed a robust engine model for diesel engine emissions, using the ability of the Gaussian process regression to reduce the influence of outliers to improve model performance [45].

Comparisons have been done that show that Gaussian process regression outperforms traditional polynomial engine modeling techniques and performs similarly or better than other machine learning techniques [51]. For this reason, Gaussian process regression is an important machine learning method to consider when developing engine models for control and calibration.

1.2.3 Comparing Machine Learning Modeling Methods

The artificial neural network and Gaussian process regression modeling methods have both been used in different types of research and both methods are capable of producing machine learning models for various different purposes. Kamath et al. compared both these methods in a study on modeling potential energy surfaces and found that Gaussian process models produced lower error models with less than one third of the root mean square error of the neural network models [52]. Similarly, Taki et al. compared the performance of both neural network and Gaussian process regression models for modeling the energy output of wheat production and showed that the neural network models produced overall lower root mean square error in the range of 0.05-0.14% compared to 10.85-16.94% for Gaussian process regression models [53]. These two papers

demonstrate that the performance of a particular modeling method is dependent on the application and that both methods should be considered in the early stages of an engine based research project.

1.2.4 The Data Problem

Machine learning can be a useful tool for engine modeling, with one of the larger limitations being its dependency on data. Vabalas et al studied the correlation between data size and machine learning modeling performance [54]. This work demonstrated that insufficiently sized data sets lead to issues such as overfitting. These models accurately reflect the provided data set but are unable to characterize operating points outside of these specific conditions as insufficient information has been provided for the model to correctly determine the characteristics of the system [54]. While many systems being modeled by machine learning processes have the benefit of having large stores of available historical data, engine research often requires a new data set for each new technology being studied. Techniques such as transfer learning have only recently been studied to overcome this issue and are only applicable to processes that are sufficiently similar so that the model trends would be compatible with the new system [55]. The limitation of data availability is one of the bigger barriers to developing engine models using machine learning techniques. With a means to overcome this issue, rapid and accurate modeling of new advanced engine technologies becomes easier to achieve.

1.3 Research Objective

The objective of this work is to develop a method of creating accurate models for engine control and calibration without the need for extensive data collection. The modeling techniques developed here are intended to be practically usable for engine research and for the analysis of new engine

technologies and should only require as much data as can reasonably collected within a few weeks to a month by a single researcher. These techniques will need to be modified to adapt their structure to the new engine architecture, however, the methodology should be transferable. The goal of this work is to make engine modeling more accessible and more practical for the development of new technologies with techniques that can be adapted to different engine architectures. Adapting these methods would require developing new models and model structures with new data and a potentially different set of inputs. The underlying methodology should still remain consistent. With this method, the benefits of new technologies and alternative fueling strategies should be easier to evaluate, allowing for new engine research to be more practically achievable without the burden of extensive data collection.

1.4 Project Overview

The objective of this work is to develop and present a method of steady state engine modeling that reduces the burden of data collection for new engine technologies. To accomplish this task, this dissertation uses machine learning models combined with intelligent data collection methods and combines them with optimization techniques and various methods of enhancing the machine learning model structure. All machine learning models used in this work are developed in MATLAB using the MATLAB Statistics and Machine Learning Toolbox. These methods are evaluated on a pilot ignited direct injection dual fuel natural gas compression ignition research engine located at the University of British Columbia Vancouver Campus. These methods are tested and evaluated against each other as well as conventional machine learning and statistical modeling methods to demonstrate their performance and the capabilities of the different techniques.

Chapter 2 begins with a layout of the test bench and its new data acquisition and control system. This section describes the detailed system and conditions under which the research took place. Chapter 3 provides a description of the data that is collected from the engine and the different strategies used for selecting operating points. The model performance metrics and system inputs and outputs are defined here. In this section the system repeatability is also assessed. This explores the limitations on what can be expected for the performance of the different metrics.

The different data sets are used in Chapter 4 to perform a survey of machine learning models with different hyperparameters for artificial neural networks and Gaussian process regression. These models are compared to a response surface modeling approach to demonstrate the capabilities of machine learning model. All three algorithms are evaluated for a full factorial research data set and a more sparsely populated data set for a much larger operating space.

In Chapter 5, a technique referred to as hyperparameter optimization is presented and applied to both the artificial neural network and Gaussian process regression modeling methods. Models are created for the full factorial, high density data set and the sparsely populated data set. The machine learning modeling survey results are compared against these new models. From this point forward, Gaussian process regression with the hyperparameter optimization is selected as the basis for the remaining techniques.

Chapter 6 implements four methods of enhancing the hyperparameter optimized models. Each of these techniques is evaluated against the hyperparameter optimized model from Chapter 5 to demonstrate the potential of each method.

The first technique, described in Section 6.1, is the multi-region method. The multi-region approach evaluates a means of dividing the modeling problem into different operating regions and modeling these regions independently. This work is evaluated using the two sparsely populated data sets presented in Chapter 3 and is compared against a combined hyperparameter optimized model.

Section 6.2 presents a method that uses a sensitivity analysis of the hyperparameter optimized models to determine which inputs have the least impact on the model outputs. This technique then uses this knowledge to intelligently remove inputs that contribute more noise than valuable data to the model.

The third method, presented in Section 6.3, is referred to as the layered model. This method separates the modeling problem into several networked submodels based on engineering knowledge and an understanding of the processes that impact the parameters being studied. This method introduces knowledge of the process into the overall model in by modeling intermediary parameters that can provide more relevant inputs to the subsequent models.

Section 6.4 presents an extension of the layered model. This method explores different approaches to introducing process equations and definitions into the layered modeling method. By replacing submodels with known, calculatable parameters, this method removes some of the modeling burden from the overall model structure.

These four methods are compared against each other and the hyperparameter optimization method to determine how each method impacted the model performance. An assessment of the benefits of each technique is then made based on the overall results.

The overall work is summarized in Chapter 7, and the final conclusions are drawn regarding which methods could improve the model performance in which ways. This section concludes with an analysis of the overall benefits of the work and the future implications of what has been accomplished.

Chapter 2: Dual Fuel Experimental Engine Facility

The objective of this project is to produce a modeling method for small data sets that will work with almost any energy conversion technology, whether that is a conventional gasoline engine or a more radical hydrogen fuel cell. Such a technique will provide a means of rapidly modeling, analyzing and then generating operating rules for such these energy conversion technologies. This early prototype focuses on an implementation for one such technology. That technology is the pilot ignited direct injection natural gas (PIDING) engine which serves as a test bench for this work.

2.1 Facility Overview

To facilitate any research with such a scope, a physical test bench is required to test and evaluate the techniques being presented. In this case, this test bench is a research engine referred to as the SCRE (single cylinder reciprocating engine). The SCRE is a research engine located in the Clean Energy Research Center at the University of British Columbia Vancouver Campus [56]. A picture of the main components of this engine is shown in Figure 2.1.

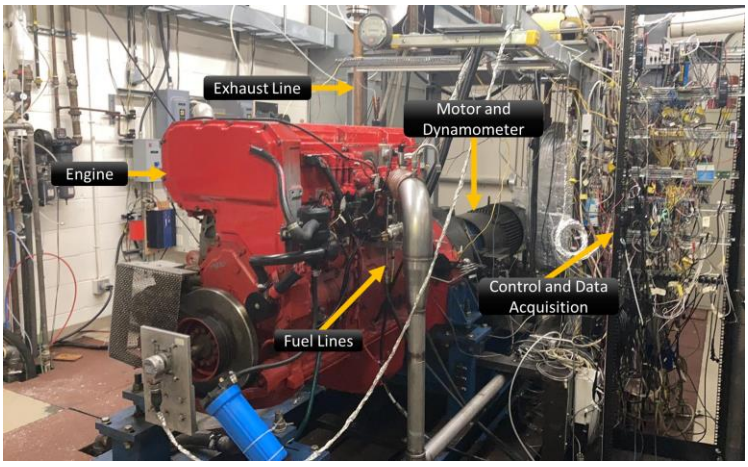


Figure 2.1: Single Cylinder Reciprocating Engine Test Bench

The SCRE test bench is comprised of several major systems spanning multiple rooms. An overview of these connected systems is shown in a diagram in Figure 2.2.

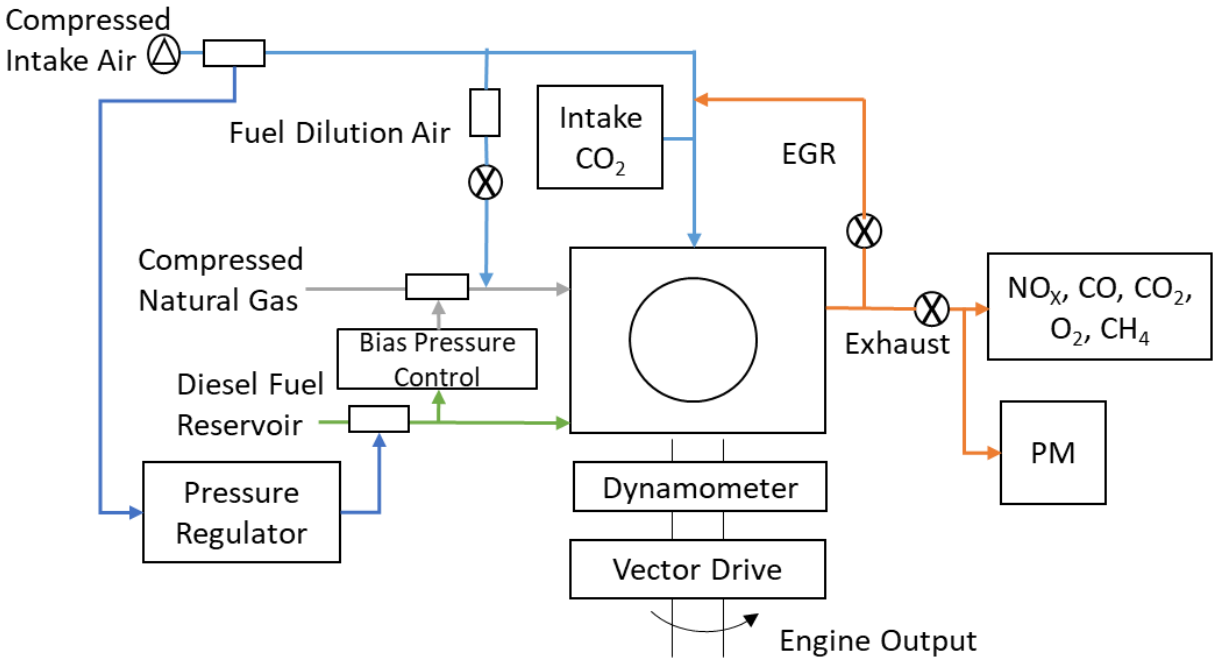


Figure 2.2: Engine Hardware System Diagram

These systems can be categorized into three major groups. These are, the SCRE and connected engine systems, the emissions measurement systems, and the control and data acquisition systems. These groups are examined in more detail in the following sections.

2.2 The SCRE and Connected Engine Systems

The SCRE is based on a Cummins ISX 400 Engine that has been heavily modified [56]. The original ISX 400 engine is a heavy duty diesel engine typically used in large trucks. The

modifications include a conversion to single cylinder operation with a dual fuel direct injection system using both diesel and natural gas. An external exhaust gas recirculation (EGR) system has been added and instead of a turbo charger the engine has a direct connection to a compressed air intake line. The control system has also been completely replaced with a new custom system that allows for fine tuned control of the engine and all connected systems. These systems are pictured in Figure 2.2 above.

The test bench's primary means of reducing emissions, compared to conventional engines, is through the combustion of directly injected natural gas with a diesel pilot fuel combined with controlling the fuel ratios and injection timings [30], [56]. The addition of a second fuel adds three new degrees of freedom. These are the new injection timing, duration, and pressure for the natural gas. More degrees of freedom increase the dimensions of the operating space. To account for the additional degrees of freedom, more data is often required to model the system. This demonstrates that this test bench is well suited to evaluating reduced data modeling techniques as determining ways to overcome this complexity can make modeling such systems easier. The overall goal of this work is to develop a means of reducing the experimental effort required to characterize the operation of a complex alternative energy conversion system, such as the one presented in the test bench. If this can be accomplished, then new alternative fuel and energy technologies could become easier and less costly to develop. Utilizing this test bench allows for a direct field test of the ideas and methodologies presented in this work.

While these methods are intended to be universally applicable to any alternative energy conversion technology, the experiments and results are directly applied to this test system and its unique

configuration. To properly understand the implementation and results, a discussion of the major components of the test bench is provided in the following sections with additional components discussed in Appendix A.

2.2.1 Fuel Systems

The SCRE is fueled with directly injected natural gas that is ignited with a diesel pilot fuel injection. 7 of the 8 cylinders on the engine have been disabled with the injectors being removed and replaced with dummy plugs [56]. The injector for the remaining active cylinder has been replaced with an HPDI (High Pressure Direct Injection) dual fuel injector provided by Westport [57]. This injector allows for the simultaneous concentric direct injection of the two different fuels, natural gas, and diesel. Both fuels can be injected at different timings which allows for significant variation in the in-cylinder fuel mixing. Under this configuration, the engine uses natural gas as the primary fuel and diesel as a pilot fuel in compression ignition. The diesel fuel's primary purpose is to ignite the natural gas. To facilitate this operation, a fuel regulation system has been developed that pressurizes both fuels with a near constant bias pressure of 1 MPa between the two fuels with the diesel being at the higher of the two pressures [30]. The diesel fuel pressure is controlled using a dome loaded pressure regulator. Additional valves have been included to regulate the fuel pressure fluctuations, shutoff the fuel systems, and depressurize the natural gas line. The natural gas is initially pressurized by a natural gas compressor system which maintains the initial pressure of the natural gas between 3200 and 3900 psi before it is regulated down to the runtime pressure set by the operator [30].

The fuel dilution system was originally developed for another research project and was designed to allow for the controlled premixing of air and natural gas prior to the natural gas injection [30]. An air compressor compresses ambient air and injects it directly into the fuel line in a safe and controlled manner. The injected air mixes with the natural gas and dilutes it prior to the injection. This changes the mixing of oxygen with the fuel and results in a reduction in particulate matter formation as well as both hydrocarbon and CO emissions. A more detailed discussion of this work can be found in the work of Singh et al. [30]. For most of the work presented here, the fuel dilution system was not used and was instead fully disassembled and disconnected from the engine. However, a study was needed of the differences between high density data, representing the more traditional data sets, and low density data. A full factorial data set was repurposed from the fuel dilution work to fill the role of the high density data set.

2.2.2 Air Pathway

Typically for an engine of this type, a turbocharger would be used to pressurize the air before it is entered into the cylinder. This is to increase the power output of the engine. While the SCRE does have such a system, it is currently disabled in favour of using a more easily controllable intake air pressurization system using a pressure regulator and an air compressor.

The exhaust system has a valve to control the back pressure as well as a pathway that is made available for exhaust gas recirculation (EGR). As the name states, this allows for the recirculation of exhaust gas into the cylinder to limit in cylinder temperature and the resulting pollutant formation. The EGR system is controlled by three valves. The first valve is the back pressure valve that forces air down the EGR pathway due to the increased pressure, when the second valve, the

EGR valve is open. The degree to which this EGR valve is open can be used to regulate the pressure on the EGR pathway. A third valve referred to as the waste gate valve is controlled electronically and allows for some of the back pressure to be relieved before it reaches the main back pressure valve. This allows for additional regulation of the EGR and back pressure. A diagram of the air pathway is shown in Figure 2.3.

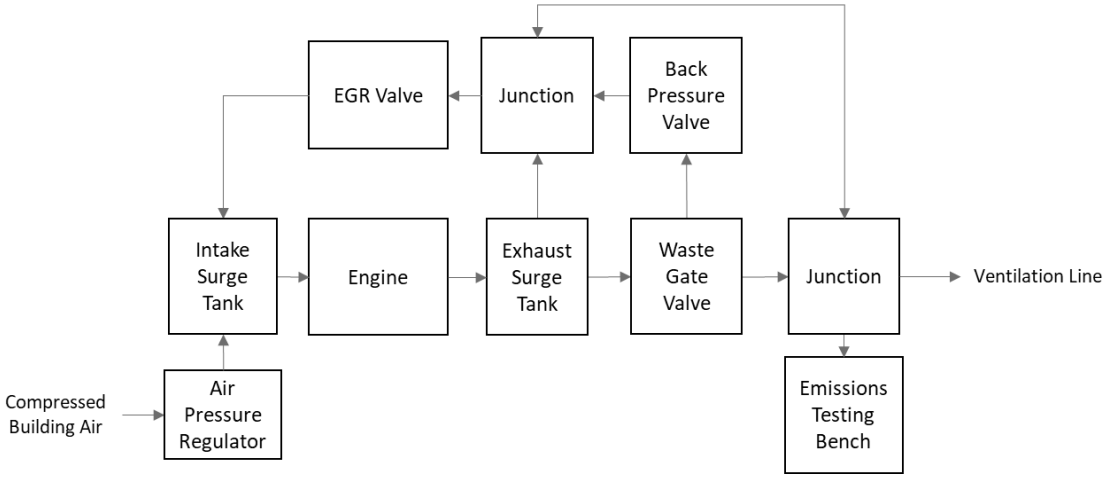


Figure 2.3: Air Flow Pathway Diagram

While the main EGR and back pressure valves are manually controlled using dials. These controls are very coarse. The main fine tuning control comes from the waste gate valve. This valve is controlled using a dual layered proportional–integral–derivative controller that controls the valve actuator to adjust the valve position based on a back pressure setpoint. For the work presented here, the EGR valve is set to a constant setting of approximately 20% open. EGR is then primarily regulated using the exhaust back pressure which is manually controlled using the back pressure valve in conjunction with the waste gate valve.

2.2.3 Vector Drive and Dynamometer

In the single cylinder configuration, the engine requires additional power to overcome friction. This is facilitated by the Vector Drive, which is an induction motor, attached to the end of the drive shaft on the engine. Once started, the vector drive provides continuous power in lieu of the deactivated cylinders and keeps the engine operating at a consistent speed. Once powered, the vector drive uses a voltage regulation dial to determine the amount of torque to apply to the engine. Typically, this value is held constant and is kept as low as reasonably possible during operation as the Vector Drive is prone to overheating. For this work, this value is set to 3800 mV which corresponds to 123.43 Nm of torque. This system is cooled independently of the engine using its own built in air cooler.

Working in conjunction with the Vector Drive is an engine dynamometer. The dynamometer is used to apply a regulated braking torque to the engine to maintain it at a constant speed and simulate a road load. This is managed by a specialized control panel which monitors the engine speed and applies a braking torque to maintain the speed setpoint. The setpoint is manually controlled using a dial. This is the primary means by which the engine speed is set for testing.

2.3 The Emissions Measurement Systems

The emissions measurement systems are comprised of two major components. These components are the primary emissions bench and the particulate matter measurement bench. After combustion and prior to reaching the emissions measurement systems, the exhaust is vented through a surge tank that smooths out the pressure pulses and vents the exhaust to an external fan. A T junction is located shortly before the vent fan which pulls a sampling of the exhaust to the emissions

measurement systems. The two systems are described in the following sections with a diagram of shown in Figure 2.4.

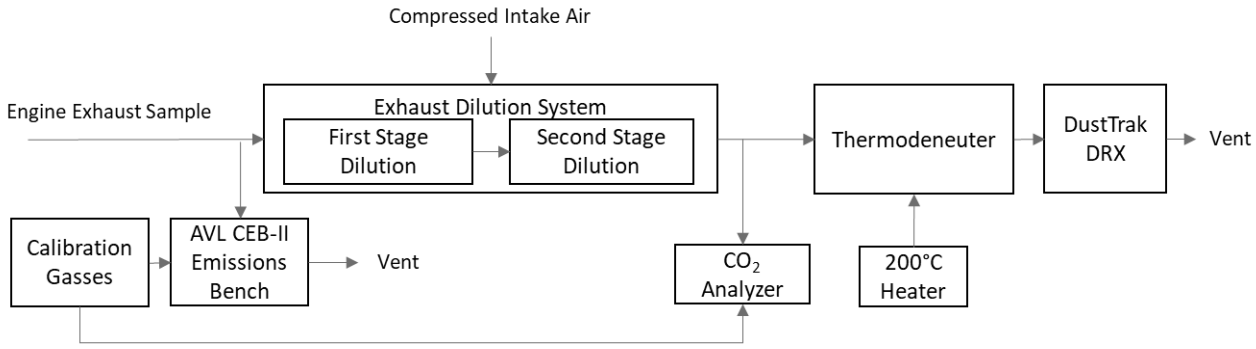


Figure 2.4: Emissions Measurement System

2.3.1 The Primary Emissions Measurement Bench

The primary emissions bench sits outside the engine test cell and is shared between the two engine test benches in the Clean Energy Research Center. The emissions bench is an AVL CEB-II emissions measurement system which samples the emissions from the engine in real time and reports the composition of the emissions at a frequency of 1 Hz. The emissions bench is calibrated on a daily basis, whenever it is used. Calibration uses a collection of calibration gas cylinders for each of the measured gasses. Calibration consists of measuring high, low and, zero (nitrogen) calibration gasses and running a self-calibration routine by comparing the results against the known values from the cylinders. The AVL CEB-II measures CO, CO_2, NO_X, O_2, and CH_4, which are all used as key parameters in this project. In addition, the emissions measurement bench also measures the intake CO_2 which is used to evaluate the EGR.

2.3.2 The Particulate Matter Measurement Bench

The particulate matter (PM) measurement bench runs separately from the primary emissions bench and is comprised of four major components. These components are the exhaust dilution system, the CO_2 analyzer, the thermodenuder, and the DustTrak DRX. The PM bench works by extracting a sample of the exhaust gas from the exhaust line and diluting it using the same compressed air used at the engine intake. The dilution ratio is controlled in two stages using two rotameters which allow for both coarse and fine adjustment. The CO_2 analyzer measures the concentration of the CO_2 in the diluted sample. This value is compared in post processing with the CO_2 measurement from the primary emissions bench to determine the dilution ratio. The dilution ratio is used to scale particulate matter measurements to their correct value. The diluted exhaust is sent through the thermodenuder which heats the sample to 200ºC to remove the volatile compounds. The now dry and treated exhaust gas is passed through the DustTrak where it is sampled and a measurement of the mass flow rate of the particulate matter is recorded.

2.4 The Control and Data Acquisition System

The combined control and data acquisition system is the single largest aspect of the engine that has received significant updating as part of this work. At its core, the control and data acquisition system has three main tasks. These tasks are, controlling the test bench, providing real-time feedback to the operator, and recording data.

The first task is to control the operation of the engine. Commands are sent from the control system to the various actuators to manage the operation of the engine according to both the user input and the custom built safety control system. The safety controller operates by constantly monitoring the

safety critical systems and sensors, as well as responding to the user's requests to adjust the operating state of the engine in a safe and reliable manner. In the event of a system approaching a critical threshold, it notifies the user prior to the safety threshold being reached and reacts to the threshold by either preventing the threshold from being reached or by safely and swiftly shutting down the entire test bench. This includes depressurizing the fuel and air lines, and deenergizing all mechanical and electrical systems.

The second main task is to present an intuitive user interface that provides sufficient real-time information to the user to allow the user to monitor the engine's operation. The main part of this user interface is shown in Appendix B.1. Using this information, the operator can evaluate the current operating point and adjust the engine operating state for whatever research is being done on the engine.

The final task is to facilitate the simultaneous recording of all data for the engine from all sensors. This needs to be done at a sufficiently high frequency so that the data can be easily post processed and used by researchers in whatever way is needed. Furthermore, the control and data acquisition system is designed to be expandable so that researchers can easily implement customizable sensors and monitor said sensors on the engine in real time.

Prior to this work, the engine control and data acquisition systems were comprised of a unified National Instruments system that operated using highly specialized hardware integrated into a Pentium 4 computer system [56]. This system included an FPGA controller and several interconnected sensor systems. Unfortunately, this system had many limitations based on

processing power and aging hardware that prevented the system from being able to provide real-time display and simultaneous data collection of all critical sensors. Additional reliability issues relating to injection timings had resulted in a need to replace this system with something more modern to facilitate this work and future work on the test bench. Developing this new system was a very extensive undertaking.

The new control system for the engine is based on a joint National Instruments CompactDAQ and CompactRIO system [58]. The CompactDAQ is used primarily for data collection and real-time display of various parameters that the operator needs to assess the current operating state. The CompactRIO is used for control and management of the main safety system. A diagram of how these systems integrate with the safety circuit is shown in Appendix B.2. Both these systems are managed by two separate computers. During operation, continuous communication takes place between both the CompactRIO and CompactDAQ, as well as the control and data acquisition computers. A high level diagram of this configuration is shown in Figure 2.5. A description of the connected sensors and actuators controlled by this system is summarized in Figure 2.6.

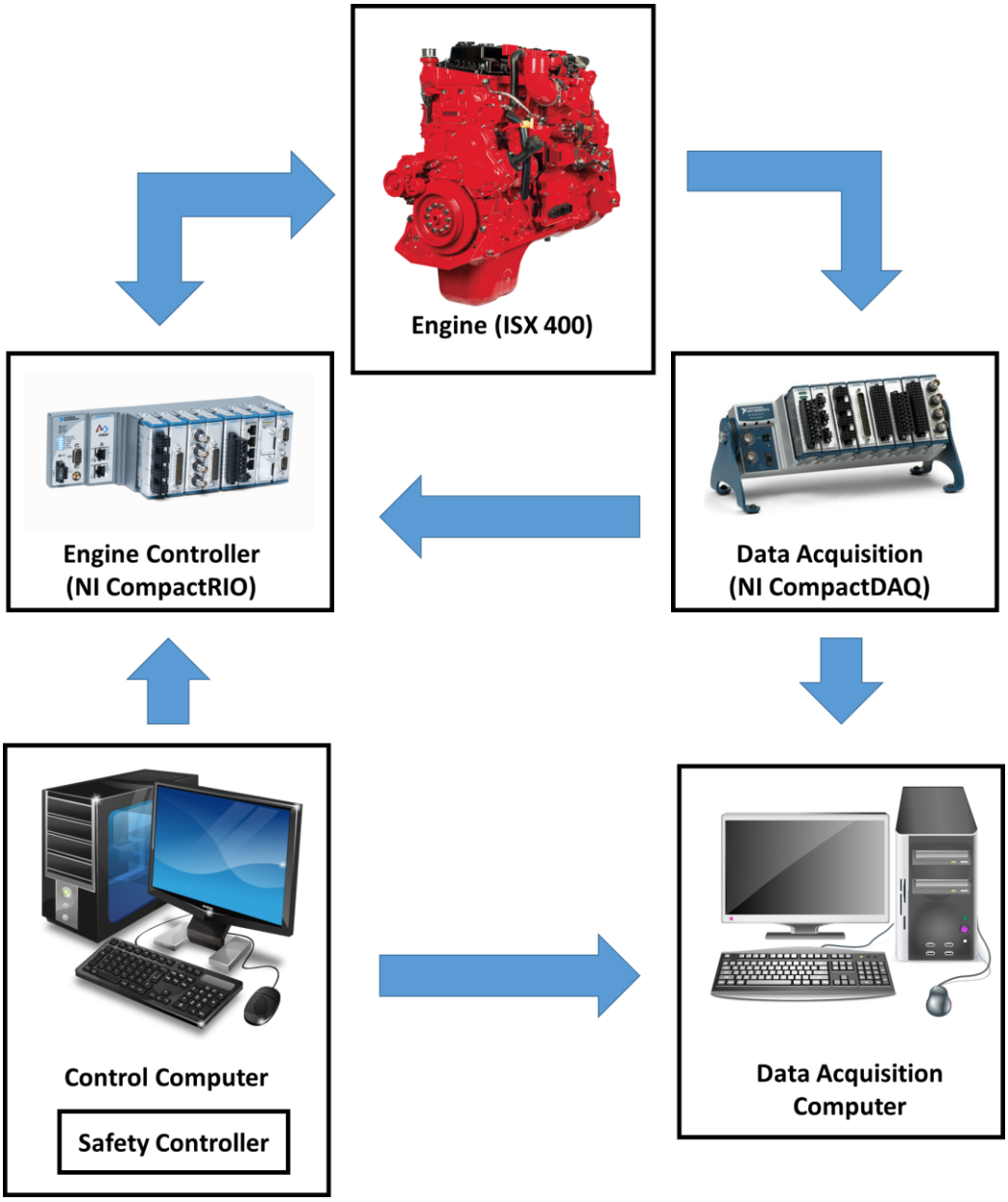
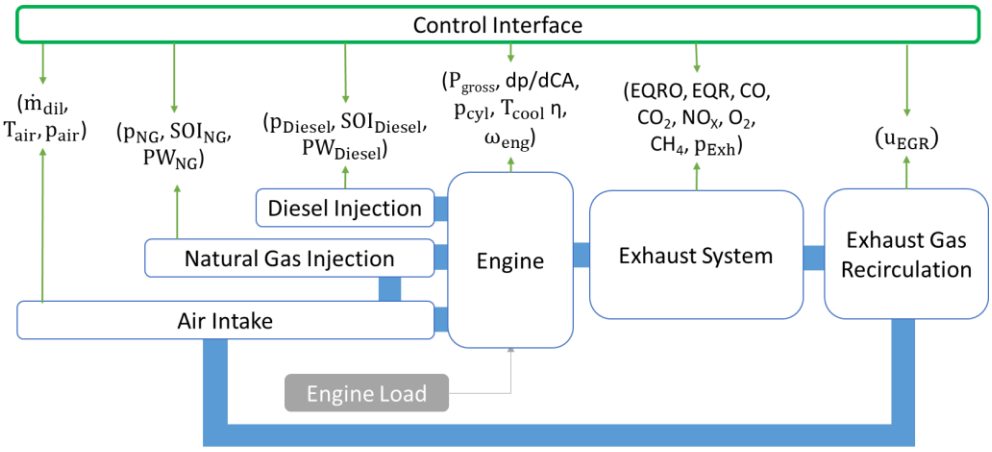


Figure 2.5: Engine Control and Data Acquisition System Overview



$\dot{m}_{dil}$	fuel dilution air mass flow rate	p_{diesel}	diesel injection pressure	T_{cool}	coolant temperature	CO_2	carbon dioxide mass flow rate
T_{air}	air temperature	SOI_{diesel}	diesel start of injection	η	thermal efficiency	NO_x	nitrogen oxides mass flow rate
p_{air}	intake air pressure	PW_{diesel}	diesel injection pulse width	ω_{eng}	engine speed	O_2	oxygen mass flow rate
p_{ng}	natural gas injection pressure	P_{gross}	gross indicated power output	EQRO	oxygen based equivalence ratio	CH_4	methane mass flow rate
SOI_{ng}	natural gas start of injection	dp/dCA	max cylinder pressure gradient	EQR	equivalence ratio	p_{exh}	exhaust back pressure
PW_{ng}	natural gas injection pulse width	p_{cyl}	peak cylinder pressure	CO	carbon monoxide mass flow rate	u_{egr}	exhaust gas recirculation valve position

Figure 2.6: System Control/Sensor Diagram

2.4.1 The CompactDAQ

The CompactDAQ (cDAQ-9188) is National Instruments' modular data acquisition solution [58]. The CompactDAQ is a standalone network controlled device that can facilitate up to 8 plug and play modular sensor IO packages capable of reading a variety of external sensors depending on the module configuration. Because of this, the CompactDAQ allows for a customizable setup

capable of being extended for future research. For this project, the CompactDAQ is configured in its default configuration which includes the following hardware modules as wired according to the wiring diagram in Appendix B.3:

- 1 X NI 9401: 8 Channel Digital I/O Module
- 1 X NI 9205: 32 Channel Analog Input Module
- 1 X NI 9213: 16 Channel Temperature Input Module
- 2 X NI 9415: 4 Channel Fast Analog Input Module

The chassis has two additional inputs used for signal timing and measuring the pulses coming from an encoder attached to the flywheel. This encoder is used to detect the start of each combustion cycle and to synchronize the timing of the high speed sensor signal recordings. These sensors are recorded on a NI 9415 module. The NI 9205 module and the remaining NI 9415 module are used for analog input sensors which are recorded at a slower interval of 10 Hz based on a hardware timer. The NI 9213 module is used for all temperature sensors and is recorded at the same interval on the same internal hardware timer. The final NI 9401 module is used for safety communication between the CompactDAQ and the CompactRIO. A single 9V line is driven high by the CompactDAQ when it is running to indicate to the CompactRIO that it is active, and no errors or safety issues have been detected. This line is otherwise driven low which triggers an emergency shut down through the CompactRIO.

2.4.1.1 Data Acquisition Software

The CompactDAQ is programmed in LabView with the previously mentioned hardware timers to manage data collection and sensor operation. The hardware can operate independently from a host computer system, however, in this configuration, the CompactDAQ is intended to always run with

a computer connected. This allows for the continuous real-time monitoring of the engine's operation using the user interface shown in Appendix B.1.

The data acquisition system is the more complex of the two software systems used to operate the test bench. The data acquisition system must simultaneously read data from four different sources, calculate a variety of operation critical parameters (such as EGR Percentage) in real-time, continuously display this data and the post processed real-time parameters in a user readable format, monitor safety critical systems that are not part of the main control system, and record all the data from all connected systems to a single data file for future use when requested. These tasks are all completed in the LabView front end code. An overview of the software logic is shown in Figure 2.7.

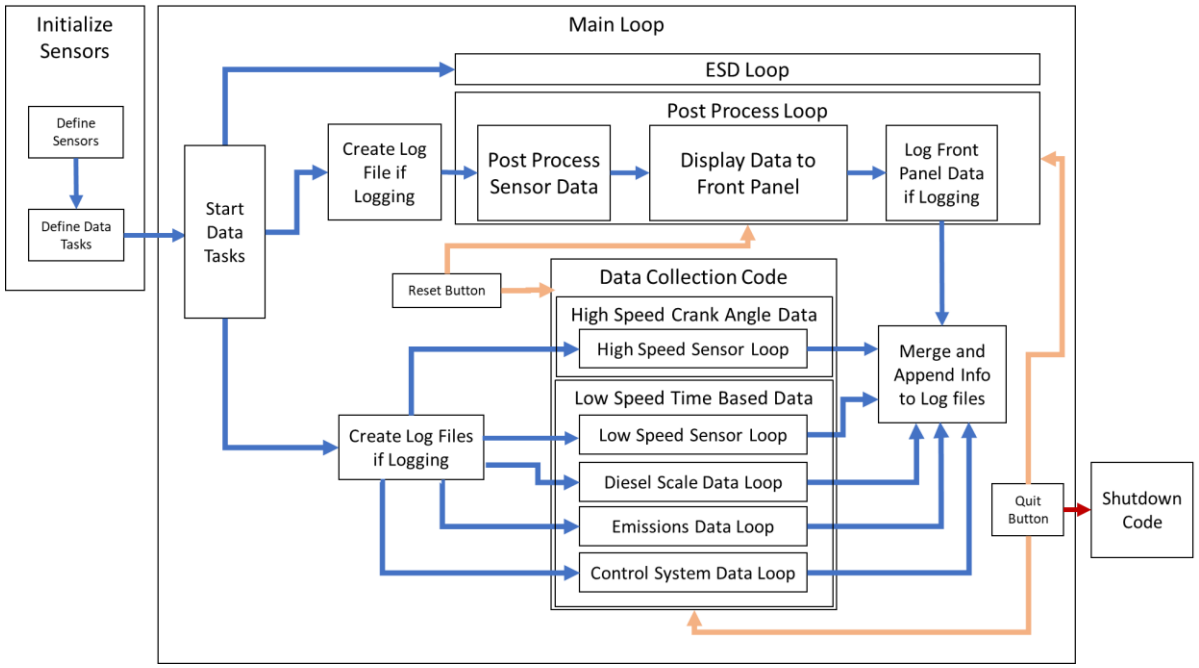


Figure 2.7: Data Acquisition Logic Diagram

Data collection tasks are broken up into two main groups. These groups are the high speed crank angle based data and the low speed time based data. The high speed data is collected at a variable speed as the engine rotates with one data point being measured every 0.5 degrees of crank angle rotation. The first recorded measurement is triggered when the engine passes the 0 degree point on the engine encoder. The system measures and records one full combustion cycle at a time. This data primarily consists of 2 sensor measurements intended to determine the cycle to cycle properties of the combustion within the cylinder. These sensors are the instantaneous cylinder pressure and the intake manifold pressure. While this data is recorded in its raw format, this data is also post processed in real time to determine parameters such as the heat release rate, which is presented on a graph in the user interface, and the 50% integrated heat release rate crank angle. Both these parameters are critical to monitoring the engine operating state and ensuring its safe operation.

The low speed time based data is recorded at an interval of 1 Hz and is comprised of four key components:

- Low Speed Sensor Data
- Diesel Mass Scale Data
- Primary Emissions Bench Data
- Control System Data

The low speed sensor data is comprised of the remaining temperature, pressure, and flow rate sensors on the engine. This data is measured at an interval of 10 Hz and averaged to 1 Hz. The diesel fuel flow rate is measured through an RS232 connection to a mass scale that reports the mass of the diesel fuel reservoir at an interval of 10 Hz. This value is then processed to produce

an average flow rate. As with the low speed data, this is averaged to 1 Hz when recorded. The primary emissions bench and control systems communicate with the data acquisition system at an interval of 1 Hz over a network connection. This data is recorded at the 1 Hz rate. The emissions bench reports the raw flow rates of the emissions in the exhaust gas. These values are post processed to determine the input/output element balance ratios. By measuring the carbon input/output ratio as well as the ratios for hydrogen, nitrogen and oxygen, the reliability of the data and system calibration can be validated in real-time while measurements are being taken. In general, these values should not vary by more than 10% from a value of 1. Some variation is expected as there is a variable time delay and mixing of gasses in the exhaust surge tank that cannot be accounted for in the calculations even though the engine should be running at a steady state. The control data consists of the entire control setpoint as displayed on the user interface of the control system. This is directly sent from the control computer to the data acquisition computer when data logging is enabled and is included in the log file. This log file is later post processed using custom made MATLAB scripts to enable it to be used for this research.

2.4.2 The CompactRIO

The CompactRIO (cRIO NI-9144) is a standalone controller with a built in FPGA and Real-Time Operating System [58]. The CompactRIO is controlled by a single computer and manages many of the actuators on the engine including the injector driver and the various valves for air and cooling flow. This system is also responsible for monitoring the safe operation of the engine and responding to safety issues by mitigating them according to the following strategy:

- Erroneous/Excessive User Input
    - Ignore new input and proceed with previous safe operation

    - Activate a warning light
- Approaching Safe Operating Limits:
    - Continue operation as requested but notify the operator of the approaching limit
    - Activate a warning light for the appropriate issue
- Exceeding Safe Operating Limits:
    - Stop injection and trigger emergency shut shown.
        - Purge/depressurize fuel lines
        - Depressurize compressed air line
        - Deenergize engine and vector drive

As with the CompactDAQ, the CompactRIO also supports 8 modular I/O packages. For this research, the CompactRIO is configured using the default configuration for the engine and utilizes the following modules with a detailed wiring diagram provided in Appendix B.3:

- NI 9411: 6 Channel Digital I/O Module
- NI 9205: 32 Channel Analog Input Module
- NI 9264: 16 Channel Analog Output Module
- NI 9375: 32 Channel Digital I/O Module

The NI 9411 module is the primary timing module. This module is directly connected to an encoder on the flywheel which provides high speed timing information. This encoder reports back the position of the crank shaft in increments of 0.5 degrees. In addition, the encoder reports it's zero position which is used to detect the start of a new injection cycle. This module also receives the ESD (emergency shut down) information from the CompactDAQ in the form of an "always on" signal that is pulled low by the CompactDAQ when an error is detected.

The NI 9375 module is wired to the injector driver and controls both the timing and duration of the injections for the diesel and natural gas as a square wave digital signal. The injector driver processes this signal into an appropriate injection signal for the solenoids that operate the injector. The 9375 module also controls the ESD line and compressed air enable switch. These signals are gatewayed through a MOSFET package that is used to isolate the circuits and scale the voltages between the different systems.

The NI 9205 module is used to measure other critical sensors including system temperatures and pressures. The temperatures are reported as a voltage by an external thermocouple signal processor while the pressures and waste gate valve position are directly measured.

The remaining actuators are controlled using a NI 9264 module which sends analog voltages to control the waste gate valve, engine cooling valve, EGR cooling valve and intake air pressure.

2.4.2.1 Control Software

The CompactRIO can run independently of a computer, however, in this configuration, the CompactRIO is intended to be used continuously with a control computer that manages each of the setpoints that operate the engine through a user interface. The control software is both the simplest and most complex of the two systems. From a user perspective the user interface is a simple series of switches and displays indicating the various actuators and safety critical systems. A diagram of the high level control system logic is shown in Figure 2.8.

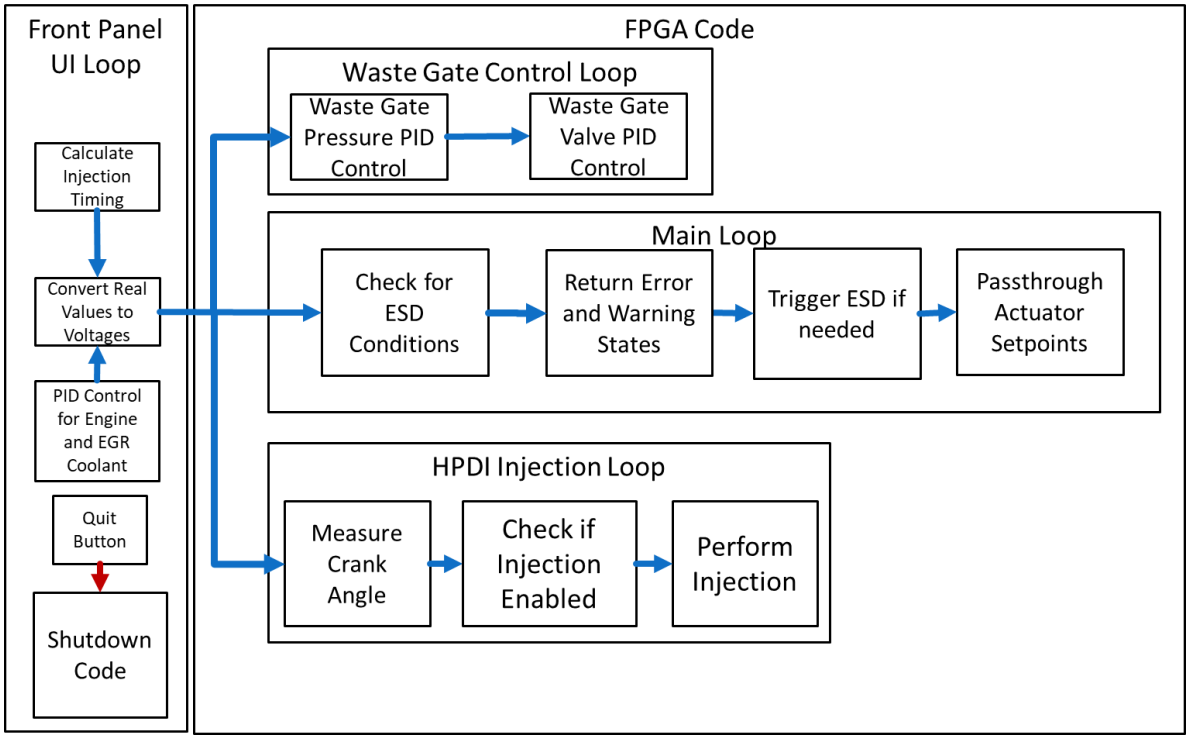


Figure 2.8: Control Logic Diagram

The actuator of the engine cooling valve is controlled using a direct passthrough switch that opens or closes the valve based on the engine temperature setpoint and the current measured temperature. Similarly, the intake air line is switched between the compressed air line or ambient air based on a direct passthrough switch that controls the appropriate valve.

The waste gate valve is controlled using a back pressure setpoint and two nested proportional–integral–derivative controllers that regulate the back pressure by regulating the waste gate valve position. This waste gate valve position is the actual output that is controlled using a variable voltage from 0 to 5.

The intake air pressure is controlled using a setpoint that directly converts to a voltage. Depending on the back pressure, EGR, and waste gate valve position, this setpoint can vary significantly from the actual pressure reported by the sensors.

An ESD (emergency shut down) system monitors various safety critical sensors such as the oil pressure, cylinder pressure, engine temperature, engine speed, and data acquisition system status. Exceeding a set warning value lights up an indicator that informs the user that a threshold is approaching. Once a pre-set threshold is reached, the ESD system is driven low, opening the ESD relay, triggering a safe emergency shutdown which purges the fuel lines and deenergizes the entire test bench. A diagram of this circuit is given in Appendix B.2.

The most complex aspect of the control code is the injection system. The injection system has two sets of injection parameters to allow for two completely independent sets of fuel injections per cycle. Each injection command has the following parameters:

- Injection Enable Switch
- Pilot Injection Angle
- Pilot Injection Duration (ms)
- Natural Gas Injection Angle
- Natural Gas Injection Duration (ms)
- Constant Pulse Separation Enable Switch
- Pulse Separation Time (ms)

The injection enable switch enables or disables the injection command. The injection angles and duration indicate the exact timing of the fuel injection for the diesel and the natural gas. These are

sent to the injection driver as a square wave digital signal for each fuel. When fuel injection is enabled, and the engine is running at or above its minimum start-up speed of 333 RPM, the engine tracks the current angle of the encoder using a counter in a high speed loop on the built-in FPGA controller. At the appropriate angle in each cycle, the signals are enabled and remain enabled for the indicated durations measured using the internal clock of the FPGA. In this way, the injection timing is precisely controlled to within 0.5 degrees of the crank angle position, which is the highest resolution possible on the engine encoder. The pulse separation enable switch allows for the overriding of the natural gas injection angle based on a desired time based separation between the diesel pilot and natural gas injections. When this is enabled, the user can specify a pulse separation timing in milliseconds that automatically calculates the gas injection angle based on the engine's current speed and the desired setpoint. This calculation is based on the relative injection timing (RIT). RIT is calculated according to Equation 2.1 where SOI is the start of injection and ng stands for natural gas. ω_{eng} is the engine speed in rpm and is multiplied by 0.006 to convert it to degrees per millisecond.

$$\mathrm{RIT} = \frac{\left(\mathrm{SOI_{ng}} - \mathrm{SOI_{diesel}}\right)}{\omega_{\mathrm{eng}} \times 0.006} \tag{2.1}$$

RIT is defined as the time between the start of the initial pilot injection and the start of the primary natural gas fuel injection [7]. The fuel pulse separation (PSEP) is used to control the crossover between the two fuel injections and is calculated as the RIT with the diesel pulse width subtracted. A diagram of this is shown in Figure 2.9.

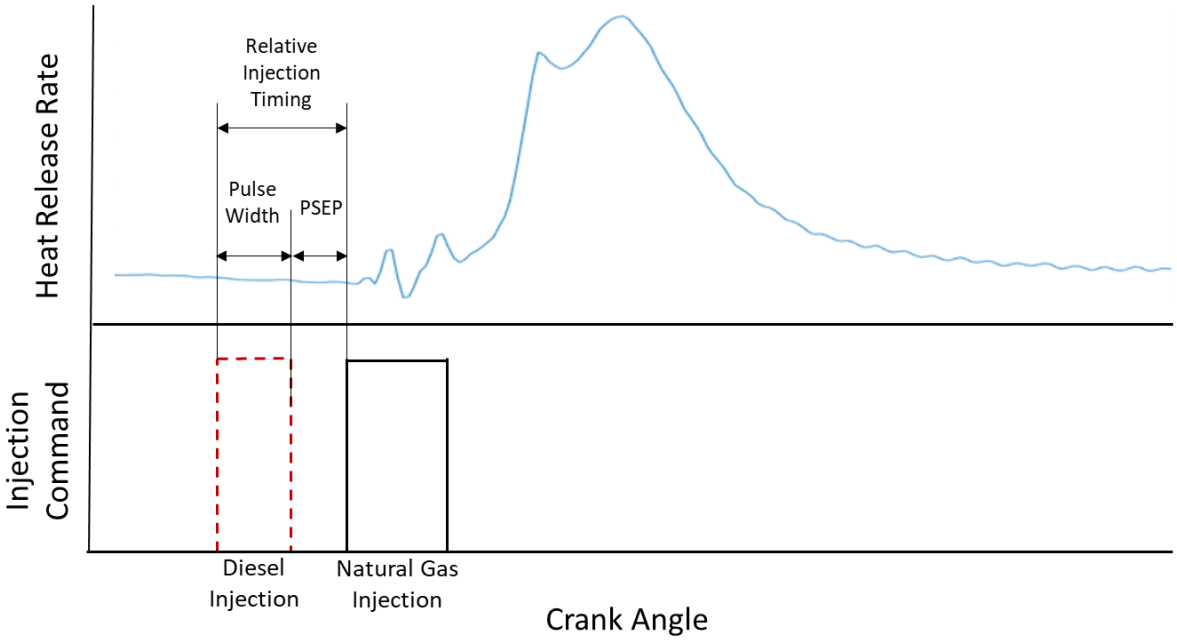


Figure 2.9: Diagram of Fuel Injection Timings Relative to Heat Release Rate

A slew rate safety control algorithm enforces a gradual change in the setpoints for the injection timing and duration. As these values can be controlled by entering numbers directly into the control panel, the slew rate control is needed to prevent a user accidentally setting an unreasonable value or accidentally shifting the engine to a drastically different operating point. The slew rate control detects rapid changes in the setpoint and holds the engine at its last known setpoint, informing the user of their error. Once the user restores the setpoint to its previous state, the error automatically clears, and the engine can continue to be adjusted as normal.

2.5 Facility Summary

The single cylinder research engine test bench provides an ideal test bench for engine modeling research. Through the pilot ignited direct injection natural gas fueling and combustion strategy, this test bench demonstrates a non-conventional, complex engine technology with several more

degrees of freedom than conventional diesel or gasoline engines. This provides an ideal modeling challenge for reduced data machine learning modeling techniques. The system is fully instrumented and is fully controllable through the new control and data acquisition system that was developed for this work. This system integrates with advanced emissions analysis tools that allow for the full assessment of the engine's operating state. Models can be generated from this information to predict various important engine performance and emissions metrics. The biggest limitation is the need for large amounts of data due to the system complexity. It is this limitation that this research seeks to overcome.

Chapter 3: High and Low Density Data Sets for Engine Model Development

The core of this research is based on developing machine learning engine models. This requires the collection of data from the research engine test bench and the selection of a set of model parameters to be studied. To facilitate this work, several different data sets are needed. These data sets span different operating spaces and utilize different strategies for selecting operating points. The following sections describe how the model parameters are selected, how they are assessed, and the different data sets that are evaluated as part of this research.

3.1 Modeling Inputs/Outputs

The model inputs and outputs are selected based on Figure 2.2 and Figure 2.6 which provide an overview of the test bench. Table 3.1 presents the inputs and outputs used throughout this work. The listed parameters represent typical engine control and performance metrics as they relate to the unique engine being studied in this research [7], [11], [30], [59]. All the work presented here uses this input and output set as the baseline for all data collection and model development. It is assumed that each of these inputs represents independent and identically distributed random variables. It is known that some of these parameters interact with each other, but these inputs are selected as they represent the inputs that are typically readily available for an engine control system. While some of the data points are not as densely selected as is typically done for machine learning, the methods used to select the model inputs are designed to ensure that they are distributed across the operating space. Only validation data, which is not used for model creation, does not follow this identical distribution assumption.

Table 3.1: Engine model inputs and outputs

Inputs		Outputs	
Name	Symbol	Name	Symbol
Intake Air Pressure (kPa)	p_{air}	Thermal Efficiency (%)	η
Diesel Gas Pressure (MPa)	p_{diesel}	Peak Cylinder Pressure (bar)	p_{cyl}
Natural Gas Pressure (MPa)	p_{ng}	Cylinder Pressure Gradient (bar/ºCA)	dp/dCA
Diesel Start of Injection (ºCA)	SOI_{diesel}	Particulate Matter (g/hr)	PM
Diesel Pulse Width (ms)	PW_{diesel}	Oxygen Based Equivalence Ratio	EQRO
Gas Start of Injection (ºCA)	SOI_{ng}	CO (g/hr)	CO
Gas Pulse Width (ms)	PW_{ng}	CO_2 (kg/hr)	CO_2
Engine Speed (rpm)	ω_{eng}	NO_X (g/hr)	NO_X
Exhaust Back Pressure (kPa)	p_{exh}	O_2 (kg/hr)	O_2
Intake Air Temperature (ºC)	T_{air}	CH_4 (g/hr)	CH_4
Coolant Temperature (ºC)	T_{cool}	Gross Indicated Power (kW)	P_{gross}
Fuel Dilution flow rate (kg/hr)	$\dot{m}_{dil}$		

3.2 Measurement Campaigns

Several measurement campaigns were completed. These measurement campaigns use different operating regions as the basis for the data collection. The different operating regions are listed in Table 3.2.

Table 3.2: Measurement campaign operating regions

	Fuel Dilution Space	Box Behnken Space	Negative PSEP Space
Exhaust Gas Recirculation (%)	0 – 26	0 – 30	0 – 30
CNG Fuel Pressure (MPa)	20 – 22	18 – 23	20
Fuel Pulse Separation (ms)	1.6 – 7.2	0 – 1.5	-1 – 0
Engine Speed (rpm)	1200 – 1350	1200 – 1500	1200 – 1500
Air Intake Pressure (kPa)	60 – 220	100 – 240	100 – 240
Diesel Start of Injection (ºCA aTDC)	-33.5 – -21	-27 – -10	-27 – -10
Diesel Pulse Width (ms)	0.55 – 0.73	0.5 – 1	0.6
CNG Pulse Width (ms)	1.19 – 4.71	0.6 – 2.5	0.6 – 2.5
Fuel Dilution Flow Rate (kg/hr)	0 – 7.64	0	0

Each measurement campaign was undertaken with a different goal/objective in mind and characterizes a different style of data collection. The different measurement campaigns that were undertaken during this research are discussed in the following sections.

3.2.1 Full Factorial Data Set

The full factorial data set represents a high density data set that was originally collected as part of an extensive research project into the benefits of air dilution of the natural gas stream. This data set is used here to demonstrate the difference between high density and low density data sets as well as to illustrate the limitations of different modeling techniques. The test bench was heavily

modified to facilitate this work and an extensive, meticulous measurement campaign was completed. This work is detailed in Singh et al [30].

The full factorial method is the most rigorous method of data collection used in this work [30]. In this method, a step size is selected for each input based on the resolution of data desired. Each input is then systematically varied by its corresponding step size until every permutation of the given inputs within the operating space have been measured. Assuming a sufficiently high resolution is selected, this method provides the most detailed data set possible [30]. The largest limitation of this method is that if it is applied to anything more than a small, specific operating region, then it produces a large quantity of data points that may not be feasible to measure due to experimental limitations. In this research, for example, examining a small set of 5 steps for each parameter listed in Table 3.2 would yield a data set of 1,953,125 data points. Even a smaller set of 3 steps would yield 19,683 data points. At a rate of 1 data point per half hour and not including the start-up and calibration times, collecting this data set would not fall within the realistic time constraints of this research. This brute force method, however, represents one of the most fundamental and detailed measurement methods available.

To ensure that this level of fidelity would remain manageable, only a small operating region was used and the size and quantity of step sizes were varied to reduce the data set to something more manageable while still focusing on the key parameters being studied [30]. Using this approach only 131 data points were needed. Of this 131 data points, 121 are used for model training and 10 are randomly selected to assess model performance The operating region for this data set is given in Table 3.2 as the “Fuel Dilution Space”.

3.2.2 Box Behnken Design Data Set

The Box Behnken design data set is based on the Box Behnken Design of Experiments method. The design of experiments methods are designed to select operating points that strategically perturb a system in such a way so that the influence of each individual input can be independently characterized [60]. This also ensures that the data set follows the independent and identically distributed input assumption. This is a common method used to select operating point for real-world engine calibration tests [50], [61], [62]. The Box-Behnken Design of Experiments defines an multi dimensional space based on the number of control parameters and attempts to characterize the space by selecting experimental points that provide the most information on the impact of each parameter on the output of the space.

This method works by finding a center point of the space and then determining the midpoint of each edge of the intersecting subspaces that defines the boundaries of the space being studied [63]. The number of data points that need to be measured grows accordingly with the number of dimensions or parameters being considered. The final data set is typically designed to be used with a response surface modeling method but can be used for other modeling approaches [60]. Despite the small data size, this data set is expected to characterize the relationships between each of the input parameters for each measured output.

The Box Behnken data set is the primary data set used in this research and represents a data set that is intended to characterize a complete operating space. The operating region characterized is given in Table 3.2 as the “Box Behnken Data Space”. This data set was developed using the Box-

Behnken method as implemented by the MATLAB Statistics and Machine Learning Toolbox. Using 8 degrees of freedom, this method contains a total of 120 data points.

3.2.3 Negative PSEP Data Set

The negative PSEP data set is similar to the Box-Behnken data set in Section 3.2.2, except that it measures data on the other side of the PSEP boundary where an overlap of the two fuel injections takes place. This data is used when data from distinctly different operating regions is needed. One of the methods, discussed later, uses this data to simplify the complexity of individual models by dividing the modeling problem into different regions. This data characterizes the operating space in the negative PSEP region which is excluded from the main Box Behnken data set. This operating space is shown in Table 3.2 and is listed as "Negative PSEP Space". This data was collected using the Box-Behnken design of experiments method as implemented by the MATLAB Statistics and Machine Learning Toolbox with 6 degrees of freedom and contains 54 data points. Note that the data size is kept smaller as it was known that varying the gas rail pressure and diesel pulse width did not have a significant impact on the combustion in this region.

3.2.4 Randomized Data Set

One randomized data set is collected for each of the Box Behnken and negative PSEP regions according to Table 3.2. This method generates randomized data points within an operating space. This method of selecting operating points is not meant to give a complete representation of the operating space and is instead designed to represent real world operating conditions. The selected points are rounded to their nearest controllable value and duplicate points within a distance of 2

control steps in any direction are automatically rejected. The Box Behnken randomized data set consists of 39 data points and the negative PSEP randomized data set consists of 9 data points.

This data is not used for model creation and is instead referred to as “validation data”. This is the main data set used for model performance measurements. As this data is not used in training, this data represents a “post-model measurement”. A model prediction for these data sets mimics a real-world comparison between a running engine and an already existing model [64]. This represents the same performance that would be seen under real use where a given operating condition may not have been considered during model development or controller calibration [65]. As such, this data is exclusively used for model performance validation.

3.3 Experimental Repeatability

Prior to evaluating the model performance there is a need to assess the confidence in the measurements, and the degree of accuracy to which the test bench can be controlled [66]–[69]. This is done through a repeatability study. Evaluating the repeatability of the experimental process provides an understanding of the margin of error in the measurements based on the measurement variability [66]. The experimental repeatability is evaluated based on a single repeat point, established in Singh et al [30]. This repeat point is based on the midpoint of the full factorial data without the use of air injection. This data point is measured repeatedly throughout the data collection process. Each repeat point measurement is completed over a period of 30 minutes to 1 hour. This is done three times throughout the day for each day measurements were performed:

1. At the start of measurements after the completed 1.5 hour warmup and calibration

2. At the midpoint of the day, after the system has been restarted if a break was taken and a faster 30 minute warm up has been completed or otherwise after several minutes of idling at a medium load point
3. At the end of the day, prior to shut down and the completion of the day's testing

It is important that each repeat point is set very precisely as any variation in the inputs or outputs should be the result of system variation and not user introduced error. It is, however, expected that small variations in the user set control parameters will be part of this assessment. The repeat point used for these measurements is detailed in Table 3.3.

Table 3.3: Repeat point

Inputs	Value
Exhaust Gas Recirculation (%)	12.5
CNG Fuel Pressure (MPa)	20.5
Fuel Pulse Separation (ms)	0.75
Engine Speed (rpm)	1350
Air Intake Pressure (kPa)	170
Diesel Start of Injection (ºCA aTDC)	-18.5
Diesel Pulse Width (ms)	0.75
CNG Pulse Width (ms)	1.55
Fuel Dilution Flow Rate (kg/hr)	0
Exhaust Back Pressure (kPa)	180

This repeat point was selected as it has a history of prior use throughout multiple measurements and has been shown to be successful in characterizing the reliability of the engine [30]. The repeatability is assessed based on the standard deviation normalized by the mean for all repetitions of the operations condition. Equation 3.1 shows the calculation of the normalized standard deviation.

$$s_{norm} = \frac{s}{\sum[Y_i]/N_{data}} \tag{3.1}$$

Here, s represents the standard deviation, Y is the value being assessed, s_{norm} is the standard deviation after being normalized and N_{data} is the number of total measurements. Figure 3.1 shows the normalized standard deviation for the control inputs and Figure 3.2 shows the normalized standard deviation for the outputs.

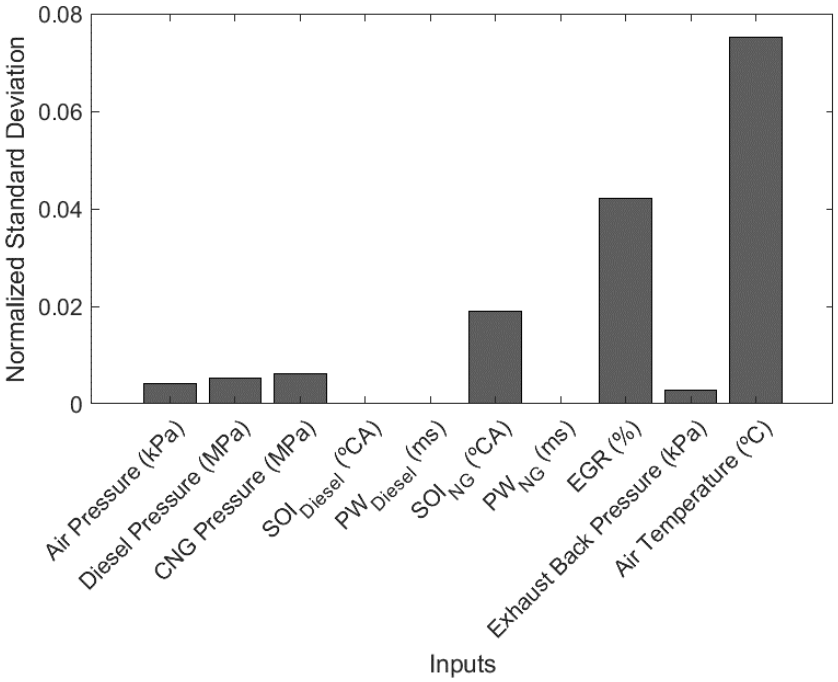


Figure 3.1: Repeat Point Input Normalized Standard Deviation

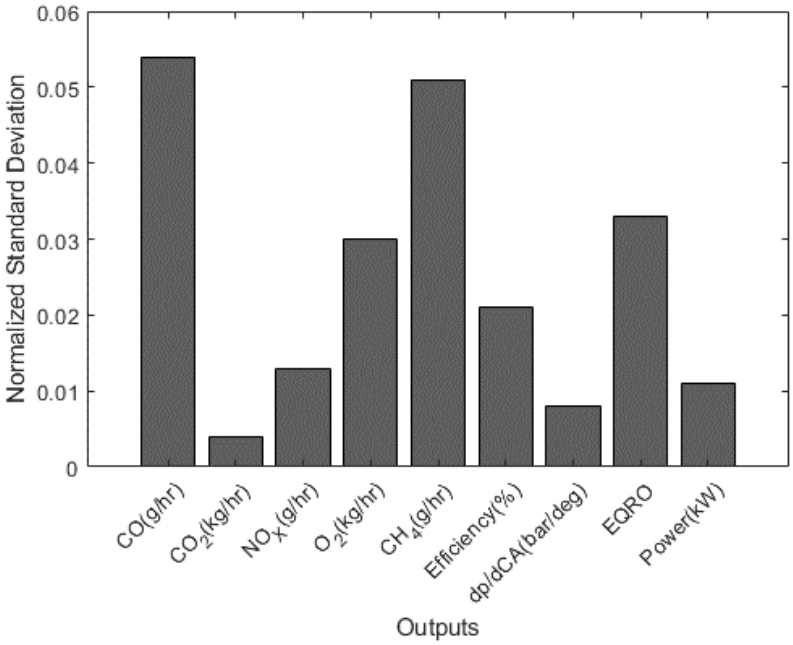


Figure 3.2: Repeat Point Output Normalized Standard Deviation

The range of the measured values for the repeat points is shown in Table 3.4. Both the input range and the output range is presented.

Table 3.4: Repeat point measurement range

Inputs		Outputs	
Name	Measured Range	Name	Measured Range
Air Pressure (kPa)	187 – 187	CO (g/hr)	5.31 – 7.02
Diesel Pressure (MPa)	23.1 – 23.5	CO_2 (kg/hr)	0.45 – 0.45
CNG Pressure (MPa)	21.8 – 22.2	NO_X (g/hr)	2.17 – 2.37
Diesel Start of Injection (ºCA aTDC)	-25 – -25	O_2 (kg/hr)	0.37 – 0.40
Diesel Pulse Width (ms)	0.45 – 0.45	CH_4 (g/hr)	0.40 – 0.45
CNG Start of Injection (ºCA aTDC)	-18.8 – -18.7	Thermal Efficiency (%)	0.43 – 0.46
CNG Pulse Width (ms)	2.18 – 2.18	Cylinder Pressure Gradient (bar/ºCA)	6.52 – 7.25
Exhaust Gas Recirculation (%)	18.1 – 18.7	Oxygen Based Equivalence Ratio	0.57 – 0.62
Exhaust Back Pressure (kPa)	182.7 – 182.9	Gross Indicated Power (kW)	44.8 – 46.0
Air Temperature (ºC)	25.9 – 26.4		

The normalized standard deviation of the inputs is consistently below 0.08. The highest standard deviation comes from the intake air temperature and is approximately 0.075. This is expected as the air temperature is not controlled in the current facility. The other parameters that vary are controlled using feedforward control systems with varying tolerances on the output. Figure 3.2 shows that these variations do not have a significant impact on the outputs as the normalized standard deviation is similarly low, with the highest value being 0.054.

The particulate matter measurement is not shown in Figure 3.2 as the variability changes substantially with the engine operating point. This makes it difficult to characterize the performance without performing a full sweep of the operating space to assess its variability. This work has previously been done, showing a 14% variability for the particulate matter measurements [30]. This assessment was performed around the same time that the bulk of the data for this research was collected. As there has not been much variation in the repeat point measurements, this assessment of the reliability is maintained to be true. A 13.51% variability is higher than those presented in Figure 3.2. This is still considered reasonable as particulate matter measurements are known to be difficult to measure and analyze [33], [70], [71].

3.4 Measurement Protocol

Each data set is measured according to protocols that were developed to ensure the consistency and validity of the data. The protocol for collecting the fuel dilution data set is laid out in the work by Singh et al [30]. Collecting the data from the remaining data sets is completed over a period of several months starting with the Box Behnken positive PSEP data set, followed by the negative PSEP data set and then the randomized validation data sets.

Calibration and system warm up takes approximately 1.5 hours. This begins with calibrating the air intake venturi zero offset, followed by the start up of the engine. The engine is then set to a 1200 RPM idling mode where it runs until the engine oil temperature reaches a nominal 80°C. The operating mode is adjusted as the engine runs to speed up the warmup process once the other system calibrations are completed. Once the engine is running, the primary emissions measurement bench is started and all the calibration gas bottles are opened. The bench is then set to run an automated calibration process. This is repeated a minimum of 4 times, to ensure that the calibration gases are correct and the connecting lines between the calibration gas and the measurement system are properly primed. A minimum of 2 successful consecutive calibrations is required to ensure that the system is properly calibrated. Once this is completed, the particulate matter measurement bench is calibrated. This primarily involves calibrating the CO_2 analyzer which is done using a zero and span gas. These two gases are sampled and the zero and span values are adjusted. The adjustments are completed when the measurements agree with the sample bottles and do not drift for a minimum of 30 minutes. Once this is done, the sample line is opened. The heating element for the sample line is then turned on and allowed to reach the 200°C setpoint.

Once calibration is complete and the engine has finished warming up, the first repeat point measurement takes place. The engine setpoint is gradually adjusted to the repeat point and allowed to idle at that point for a minimum of 15 to 30 minutes while the EGR value settles. During this time, the DustTrak is turned on and the analog output is adjusted to measure in the range of the current PM output. This is necessary as the accuracy of the DustTrak analog output is limited at the lower voltage range due to signal noise. To address this, the range of measurements is adjusted

for every sample point. Similarly, the dilution ratio is also adjusted to keep the measurements within the valid range of the DustTrak. After the repeat point has settled, a 2 minute recording of all the sensor data takes place. This is compared to past measurements to ensure the engine is operating as expected.

The main measurements then begin by gradually adjusting the engine settings from one operating point to the next. The DustTrak settings and dilution ratio are adjusted and the EGR is allowed to settle at each data point before measurements are taken. These are primarily related to the intake and back pressure as these fluctuate as the engine settles. Once the system is stable and the operating point has been reached, a 2 minute measurement is taken. This is then repeated for each operating point. Operating points continue to be collected for 7 to 10 hours. During this time, additional repeat points are collected at the midpoint and end of the day. The engine is then shut down and the process is repeated for several days until the measurement campaign is completed.

3.5 Model Performance Metrics

There are several ways to assess the performance of a machine learning model. In this work, the goal is to primarily focus on validation data results as these results represent the true performance of a model in a real-world operational setting. Additional results are presented for the training data, however, as the models in this work are produced using the training data, these results present a false impression on the overall model performance. It is easy for a model to predict something it has been trained with. The true test of a model is in predicting something not used during training. This is where the validation data results come in.

Model performance itself is measured using the normalized root mean square error, nRMSE, which is calculated using Equation 3.2 and the coefficient of determination, COD, which is calculated using Equation 3.3. N_{data} is the number of data points, Y represents the outputs for the measured and model predicted values. cov is the covariance of the measured and predicted outputs and s represents the corresponding standard deviations.

$$\mathrm{nRMSE} = \frac{\sqrt{\frac{\sum\left[\left(Y_{i,meas} - Y_{i,pred}\right)^2\right]}{N_{data}}}}{\sum\left[Y_{i,meas}\right]/N_{data}} \tag{3.2}$$

$$\mathrm{COD} = \left(\frac{\mathrm{cov}\left(Y_{meas}, Y_{pred}\right)}{s_{meas} \times s_{pred}}\right)^2 \tag{3.3}$$

The coefficient of determination provides a measurement of how well the model can predict the trends in the data. A value near 1 represents a model that accurately depicts the way in which the results vary from data point to data point. Error may still exist, but given the data set measured, the model is able to accurately replicate some representation of the system. While this is a useful metric, the main value being studied is the nRMSE. nRMSE is a representation of the true error between the expected value and the predicted value for a given set of data points. This data is normalized against the mean value of the training data for representation purposes but otherwise indicates how much the results deviate and how accurate the models are to the measured data. While both metrics are valuable, the COD is only used to indicate if a true model is created. The nRMSE is used to measure how well the model works. Model nRMSE is considered low if the validation data results are below 0.1 as this would result in values that approach the repeatability

of the system as shown in section 3.3. Some researchers have developed models that show nRMSE values below this value. Lotfan et al. demonstrated a CO neural network model that showed an equivalent nRMSE of 0.03 while Kokkulunk et al. presented a CO_2 neural network model with an nRMSE of 0.065 [39], [40]. These works used more data over a smaller domain than what is considered in this work. Vabalas et al. demonstrated that machine learning model accuracy is expected to decrease dramatically as the ratio of features, or degrees of freedom, to samples increases indicating that the nRMSE is expected to be higher for this work due to the larger ratio of the operating space to the size of the data sets [54]. A value of 0.1 approaches the typical range of nRMSE seen in these other works. This value also approaches a point where system repeatability can become a significant component of the error as it can account for up to 6% of the error based on Figure 3.2. As such, 0.1 is taken as the primary benchmark for determining whether or not a model is successful with the understanding that better results may still be possible.

Chapter 4: Survey of Machine Learning Model Performance for Varying Density Data Sets

This chapter presents a collection of machine learning models that represent the range of the performance expected for modeling the data sets presented in the previous section. Both neural network and Gaussian process regression models are created to model each of the outputs in Table 3.1. Each of the models discussed in the following sections is developed using the MATLAB Statistics and Machine Learning Toolbox. The models are developed and trained using randomly selected hyperparameters. These models are presented to demonstrate the current limitations of the machine learning techniques, the importance of identifying proper hyperparameters, and to provide a basis from which these models can be improved.

Both the artificial neural network (ANN) and Gaussian process regression (GPR) techniques are utilized in this section to generate a set of 500 models for each model output. Each of the models is produced with a different set of randomly selected hyperparameters. The hyperparameters are settings which define how the models are configured. The randomly selected hyperparameters include the neural network node count, layer count, and training algorithm as well as the Gaussian process regression kernel function, and noise standard deviation [44], [72]. These hyperparameters are discussed in further detail in appendix C.1 and C.2. Each output in Table 3.1 is modeled separately from the other outputs as a multiple input, single output model using all the inputs listed in the table. These models are then also compared to a typical response surface (RS) model based on linear regression.

The three modeling methods are evaluated on the full factorial data set from Section 3.2.1 and the Box Behnken data set from Section 3.2.2. These two data sets make it possible to evaluate the two different cases where data is very tightly packed, as in the full factorial data, and when data is only sparsely available, as in the Box Behnken data set. It is the latter case that this overall research project focuses on, as this is the case where it becomes necessary to use the more advanced techniques presented later in this work [34].

4.1 Survey of Hyperparameter Model Performance Results

500 models are generated for each output using both Gaussian process regression and artificial neural networks with each model having a different hyperparameter configuration. In typical machine learning modeling, no prior knowledge is available for selecting these hyperparameters and as such, either a rule of thumb or trial and error approach is often used [41], [73]. In this work, hyperparameters are selected randomly, and the aggregated results of all 500 models are presented. The primary metrics for performance are the coefficient of determination, normalized root mean square error, and finally, the robustness, which can be assessed based on the size of the range of the other two metrics across multiple model iterations.

4.1.1 Full Factorial Data Training Results

Evaluating the performance of the machine learning models begins with the idealized full factorial data. The following figures present the results for the training data.

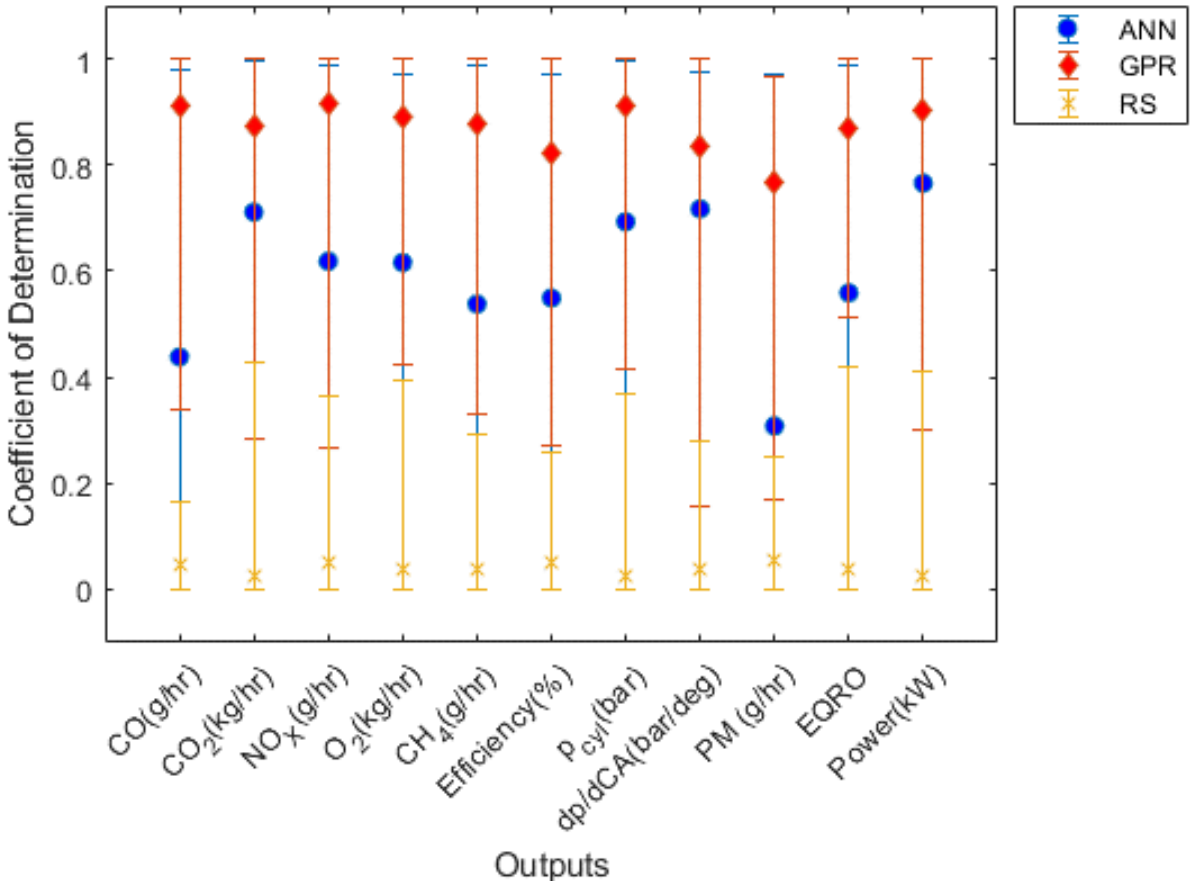


Figure 4.1: Coefficient of Determination for the Modeling Survey with Full Factorial Data (Training Results)

Figure 4.1 shows the coefficient of determination results for the full factorial training data. It is immediately apparent that there is a wide range of performance across all modeling methods. For most models, this range spans approximately half of the possible range of values from 0 to 1. This indicates the importance of properly selecting hyperparameters as the different hyperparameters can lead to vastly different modeling results when it comes to the coefficient of determination.

On average, the coefficient of determination for Gaussian process regression is in the higher range above 0.8 which indicates that most of the randomized hyperparameters produce models that

accurately fit the trends of the data set. The neural network coefficients of determination are lower, around 0.6 on average, and the response surface coefficients of determination are the lowest with most models producing results close to 0.

The response surface performance is largely a result of the data complexity. In this case, the response surface model is poorly suited to modeling this data set, where the machine learning models do not have this problem. This underscores the major limitation of response surface modeling. This limitation is the dependence on a data set that has been carefully curated for the response surface.

The full factorial data collection technique is very powerful for capturing the characteristics of complex systems and for providing large amounts of data. This is very beneficial for machine learning. Response surface modeling, however, is also dependent on the spacing between the data points and the complexity of the system being modeled. In this case it is unable to utilize the full factorial data.

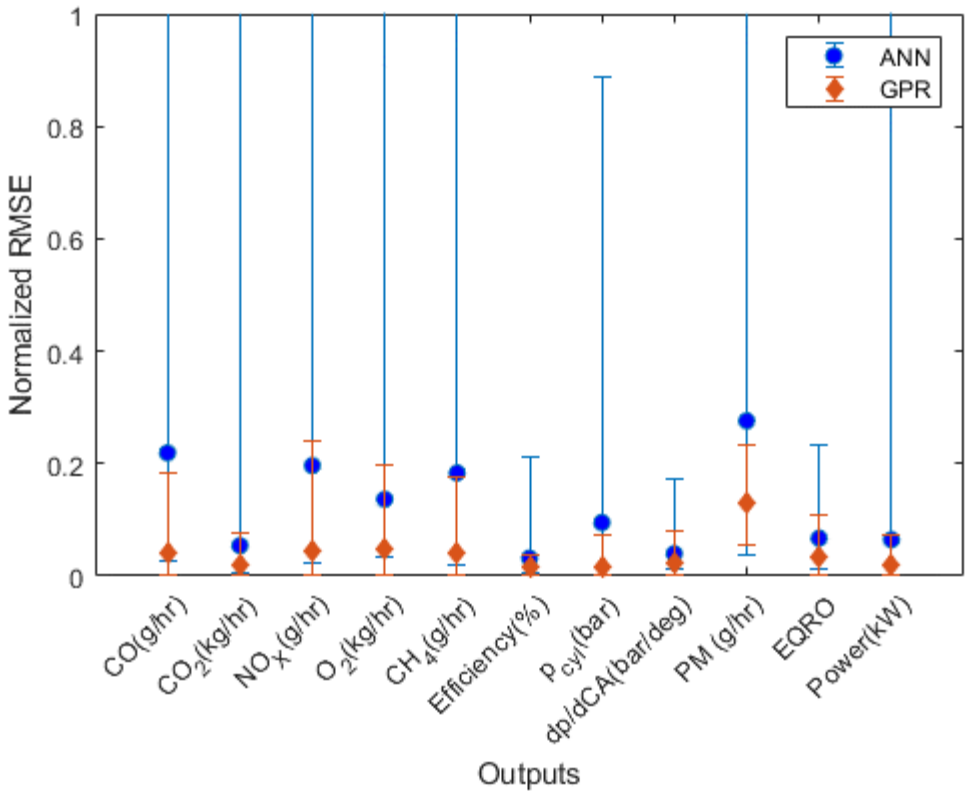


Figure 4.2: Normalized RMSE for the ANN and GPR Modeling Survey with Full Factorial Data (Training Results)

Figure 4.2 shows the normalized root mean square error for the full factorial training data set. The response surface results have been omitted from the figure as the models proved to be unusable, with nRMSE results exceeding 10^{14}. A summary of the nRMSE results for all three sets of models is shown in Table 4.1.

Table 4.1: Normalized RMSE for the modeling survey using the full factorial data (training results)

Output	Mean Normalized Root Mean Square Error *		
	Response Surface	Artificial Neural Network	Gaussian Process Regression
CO	1.37E+14	0.219	**0.041**
CO_2	1.31E+13	0.054	**0.018**
NO_X	1.15E+14	0.196	**0.045**
O_2	1.06E+14	0.136	**0.049**
CH_4	1.33E+14	0.183	**0.042**
Efficiency	1.22E+13	0.032	**0.014**
p_{cyl}	7.71E+12	0.095	**0.016**
dp/dCA	2.02E+13	0.040	**0.023**
PM	1.40E+14	0.276	**0.130**
EQRO	4.12E+13	0.067	**0.034**
Power	5.48E+12	0.065	**0.019**
Average	6.64E+13	0.124	**0.039**

* Lowest nRMSE in bold.

The machine learning methods all perform much better than the response surface with significantly lower normalized root mean square error values consistently averaging under 0.3. Based on the system repeatability study, ideal values would be below 0.1. For training data this value should be much lower and closer to 0.01. This indicates that there is still room for improvement. Excluding the particulate matter results, the Gaussian process regression models come close to this goal with

an average nRMSE of 0.039. With a few exceptions, the neural network models consistently fail to meet this metric with an average nRMSE of 0.124.

The ANN range for the nRMSE, as indicated by the error bars, shows a significant performance variation that exceeds 1 for several outputs. Even the value of 1 would not be considered robust as the error on the prediction for a given data point spans the range of the experimentally measured values. This further illustrates the importance of selecting proper hyperparameters. The overall range of possible values makes it impossible to rely on this model performance when the hyperparameters are selected arbitrarily. GPR shows a greater degree of robustness with all the models having a nRMSE variation of approximately 0.2. While not as severe as the neural network results, this is still well above what would be expected for training data performance.

4.1.2 Full Factorial Data Validation Results

For each model, 10 data points were selected to be removed and reserved to be used as validation data. The validation data is used to assess the performance of the models with data that was not used for the model training. These results provide for a more realistic view of the overall model performance compared to the results from the training data set. The following figures show the performance for the validation data.

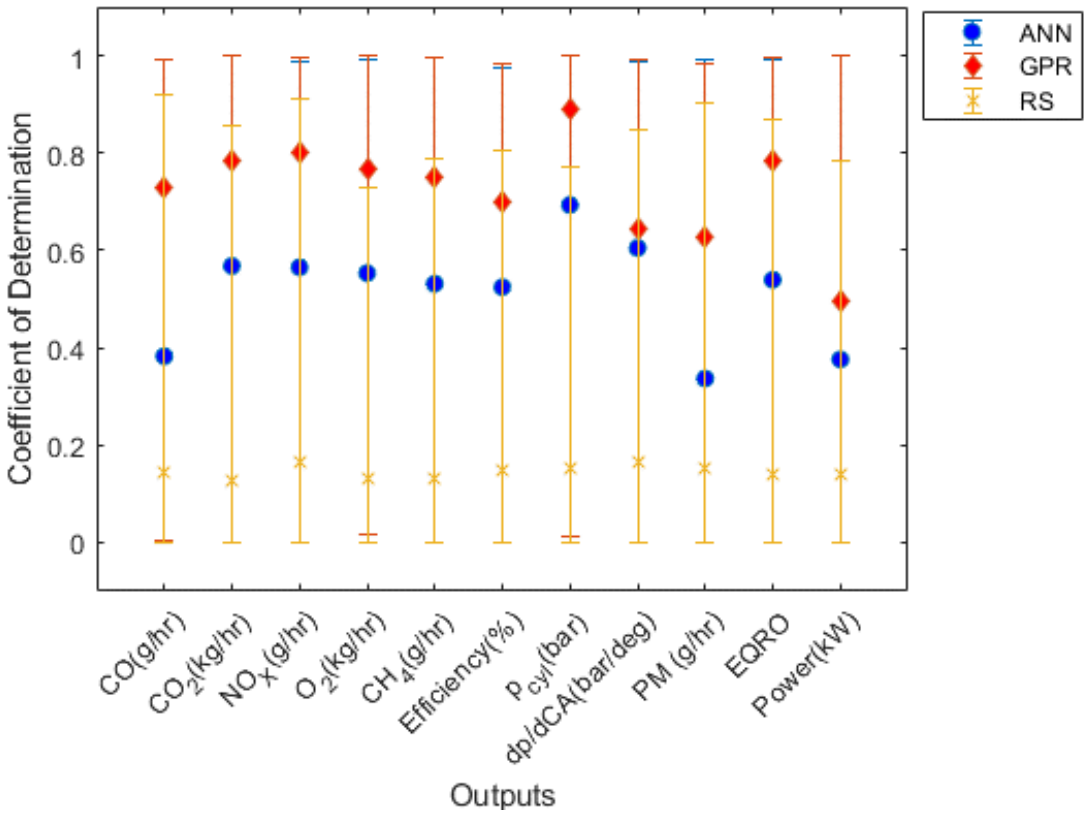


Figure 4.3: Coefficient of Determination for the Modeling Survey with Full Factorial Data (Validation Results)

Figure 4.3 shows a comparison of the validation data results for the coefficient of determination of the three modeling techniques using the full factorial data set. The average coefficient of determination for the GPR models is approximately 0.7, which is lower than the ideal 0.8 or 0.9 target values. The ANN coefficient of determination results are consistently lower with most models showing an average just above 0.5 across the 500 models generated for each output. The range of coefficient of determination results across the 500 models for each output, as indicated by the error bars, spans the entire 0 to 1 range of possible values. This further illustrates that without a means of determining the best model hyperparameters to use, model robustness cannot

be assured, and many models will be unable to properly model the trends in the data. As with the training data, the response surface underperforms compared to the machine learning models with the average coefficient of determination consistently under 0.2.

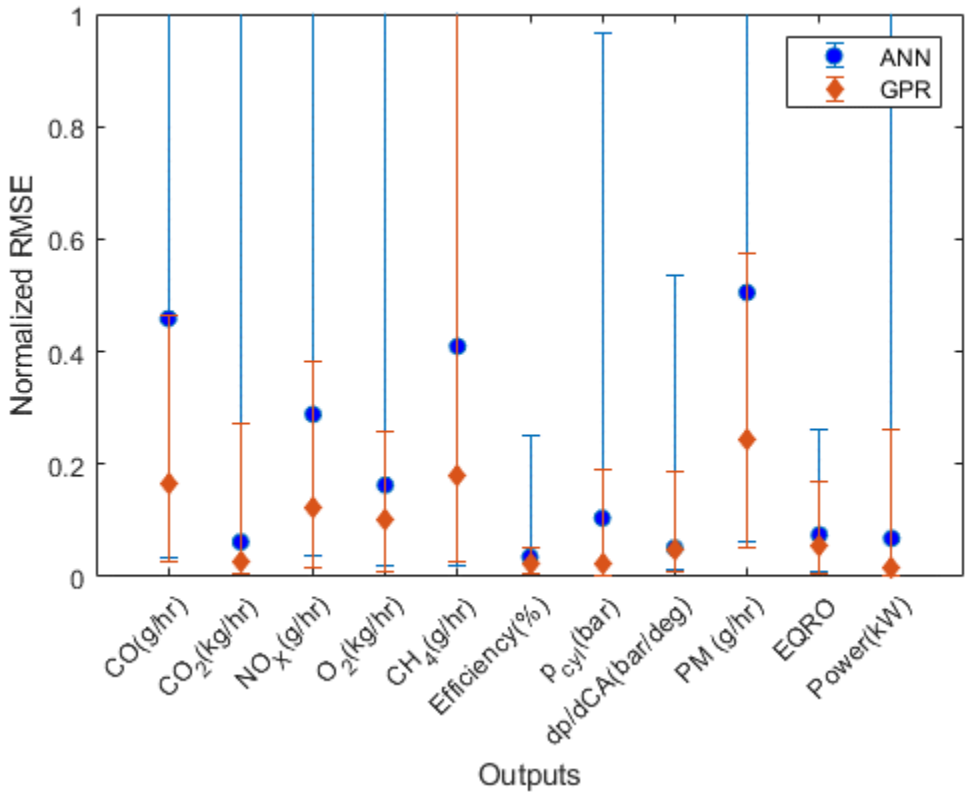


Figure 4.4: Normalized RMSE for the ANN and GPR Modeling Survey with Full Factorial Data (Validation Results)

Figure 4.4 shows the normalized RMSE results for the machine learning models alone. The response surface results have been omitted as the errors again approach values in the range of 10^{14}. The range of the normalized root mean square error varies significantly as it did with the training data. In general, the ANN models have more variation than the GPR models with nRMSE values

that reach beyond 1 for 7 of the 11 modeled outputs. For GPR, only the CH_4 exceeds a normalized root mean square error of 1. These results are summarized in Table 4.2.

Table 4.2: Normalized RMSE results for the modeling survey using the full factorial data (validation results)

Outputs	Mean Normalized Root Mean Square Error *		
	Response Surface	Artificial Neural Network	Gaussian Process Regression
CO	3.68E+14	0.459	**0.165**
CO_2	1.44E+13	0.062	**0.026**
NO_X	2.08E+14	0.289	**0.124**
O_2	1.22E+14	0.163	**0.100**
CH_4	4.39E+14	0.409	**0.180**
Efficiency	9.52E+12	0.035	**0.022**
p_{cyl}	7.32E+12	0.104	**0.022**
dp/dCA	1.35E+13	0.051	**0.049**
PM	2.93E+14	0.505	**0.245**
EQRO	4.59E+13	0.075	**0.054**
Power	5.75E+12	0.068	**0.017**
Average	1.39E+14	0.202	**0.091**

* Lowest nRMSE in bold.

The normalized root mean square error results demonstrate that the response surface is unable to model this data. The response surface nRMSE is consistently high, exceeding 10^{14} on average. The

neural network nRMSE is much lower at 0.202 on average. The Gaussian process regression nRMSE is lower still, at 0.091 on average. 8 of the 11 GPR models show a normalized RMSE at or below 0.1. This indicates that this data set properly characterizes this operating space and that it contains sufficient information for machine learning models to be able to properly model it.

The variation in the nRMSE results, combined with the coefficient of determination results indicate that the modeling method is not robust. This shows that properly selected hyperparameters are needed to narrow the range of the normalized root mean square error and coefficient of determination to guarantee a particular model performance. Chapter 5 discusses a method for selecting hyperparameters using optimization techniques. Prior to evaluating this method, an understanding of the effects of data set size is still needed.

4.1.3 Box Behnken Data Training Results

The Box Behnken data presents an opportunity to see how the machine learning modeling methods perform with a sparse data set that reduces the experimental burden. While the Box Behnken data contains a similar number of data points to the full factorial data, the Box Behnken data is spread out over a much larger operating space. As with the full factorial data set, 500 models were created for each output being modeled. Each model uses a randomized model hyperparameter configuration that varies the layer count, node count and training method for neural network models and the kernel function and noise variance for Gaussian process regression models. The first set of results that are examined are the training data results. These results represent the best possible performance for the models. The coefficient of determination for the Box Behnken training data is presented in Figure 4.5.

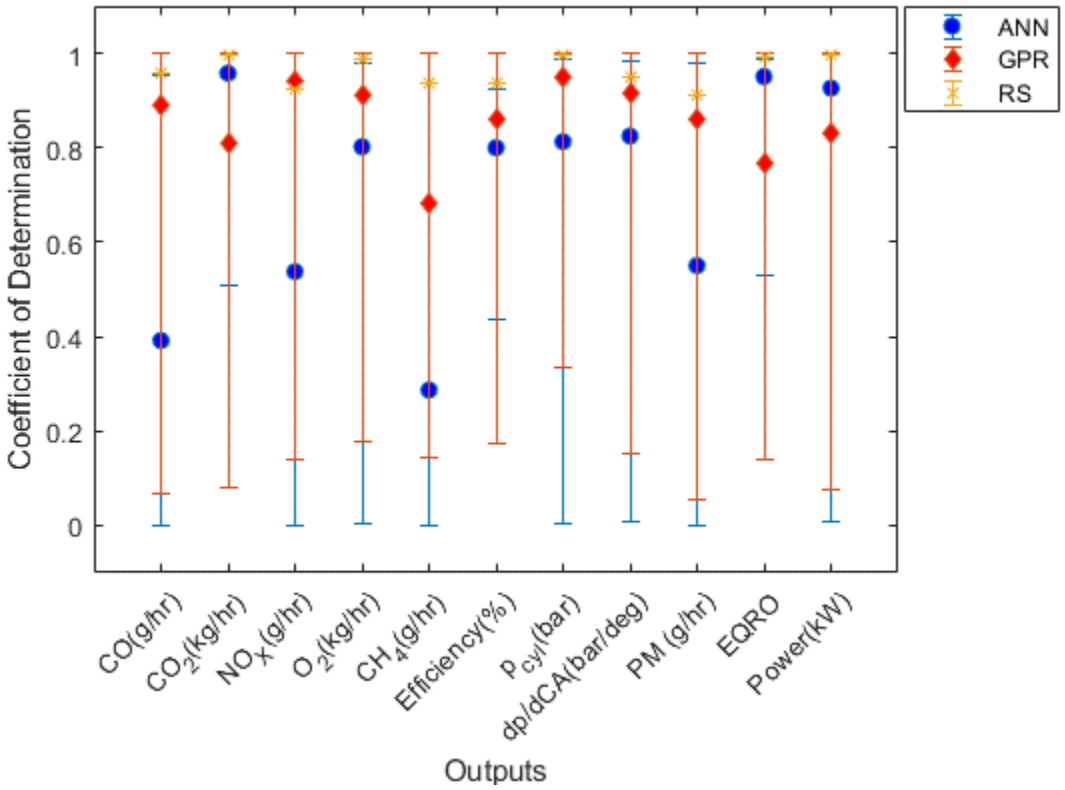


Figure 4.5: Coefficient of Determination for the Modeling Survey with Box Behnken Data (Training Results)

From Figure 4.5, the response surface results show a high correlation with the coefficient of determination being consistently above 0.9. This is in sharp contrast to the coefficient of determination results in Sections 4.1.1 and 4.1.2. The Box-Behnken DOE method was originally designed for response surface modeling [60]. This data set is therefore properly structured to support the response surface modeling technique. Except for the CO_2, EQR and gross indicated power, the neural network models are noticeably lower at approximately 0.8 with the CO, NO_X, CH_4, and particulate matter below 0.6 on average. The neural network results have a large span across all the possible values for the coefficient of determination with the shortest range being

those of the thermal efficiency, EQRO and CO_2, all of which have a range of about 0.5. The range of the Gaussian process regression coefficient of determination results spans the entire space from 0 to 1. Despite this, almost all of the averaged coefficient of determination results are above 0.8. The Gaussian process regression still shows lower average coefficients of determination values than the response surface. Figure 4.6 shows the normalized RMSE results.

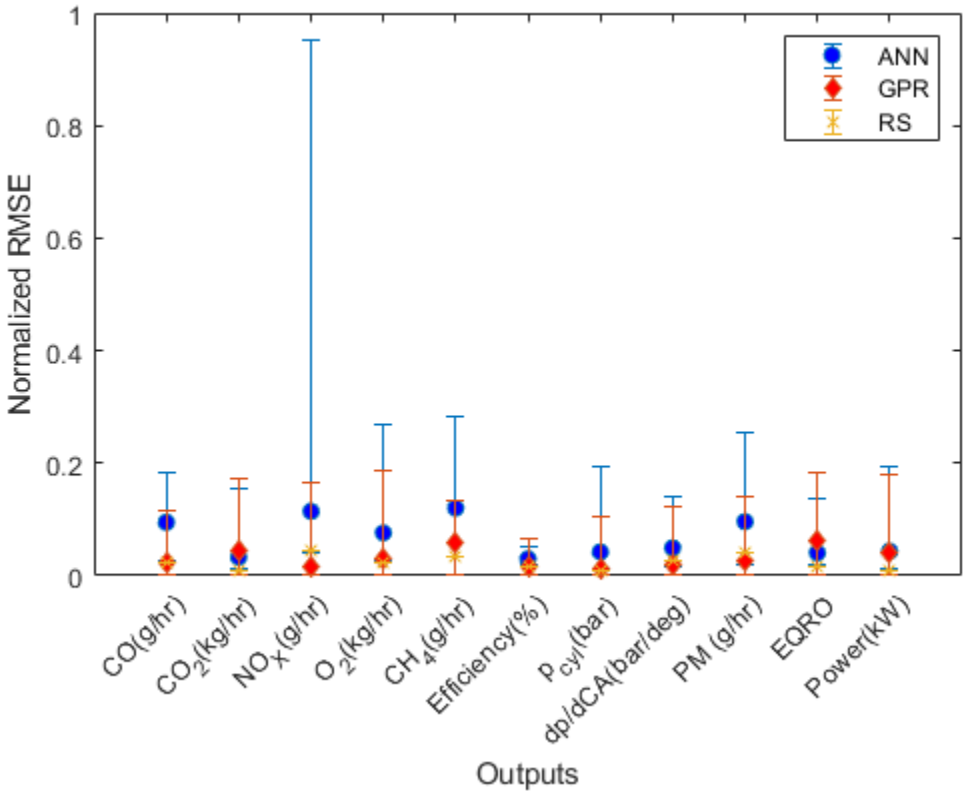


Figure 4.6: Normalized RMSE for the ANN and GPR Modeling Survey with Box Behnken Data (Training Results)

Figure 4.6 shows normalized root mean square error results. The summarized mean training data nRMSE results are shown in Table 4.3.

Table 4.3: Normalized RMSE results for the modeling survey using the Box Behnken data (training results)

Outputs	Mean Normalized Root Mean Square Error *		
	Response Surface	Artificial Neural Network	Gaussian Process Regression
CO	0.025	0.095	**0.023**
CO_2	**0.008**	0.033	0.045
NO_X	0.045	0.114	**0.017**
O_2	**0.022**	0.076	0.030
CH_4	**0.034**	0.120	0.058
Efficiency	0.017	0.030	**0.016**
p_{cyl}	**0.008**	0.042	0.011
dp/dCA	0.027	0.049	**0.019**
PM	0.042	0.096	**0.028**
EQRO	**0.015**	0.041	0.061
Power	**0.009**	0.043	0.040
Average	**0.023**	0.067	0.032

* Lowest nRMSE in bold.

The response surface model shows the overall lowest normalized root mean square error with the Gaussian process regression showing lower results for the CO, NO_X, thermal efficiency, dp/dCA, and particulate matter. The neural network normalized root mean square error results show the highest average error and the widest range of performance for each of the modeled outputs with

the error varying by about 0.2 on average and up to 0.9 in the case of NO_X. From these initial results, the response surface shows the overall best performance for the training data.

4.1.4 Box Behnken Data Validation Results

In addition to the Box Behnken training data, a randomized validation data set has been collected. This randomly selected data set includes operating points that do not exist in the training data set and provide a more accurate representation of the model performance [64]. Figure 4.7 shows a comparison of the results for the coefficient of determination for the three modeling techniques using the Box Behnken validation data set.

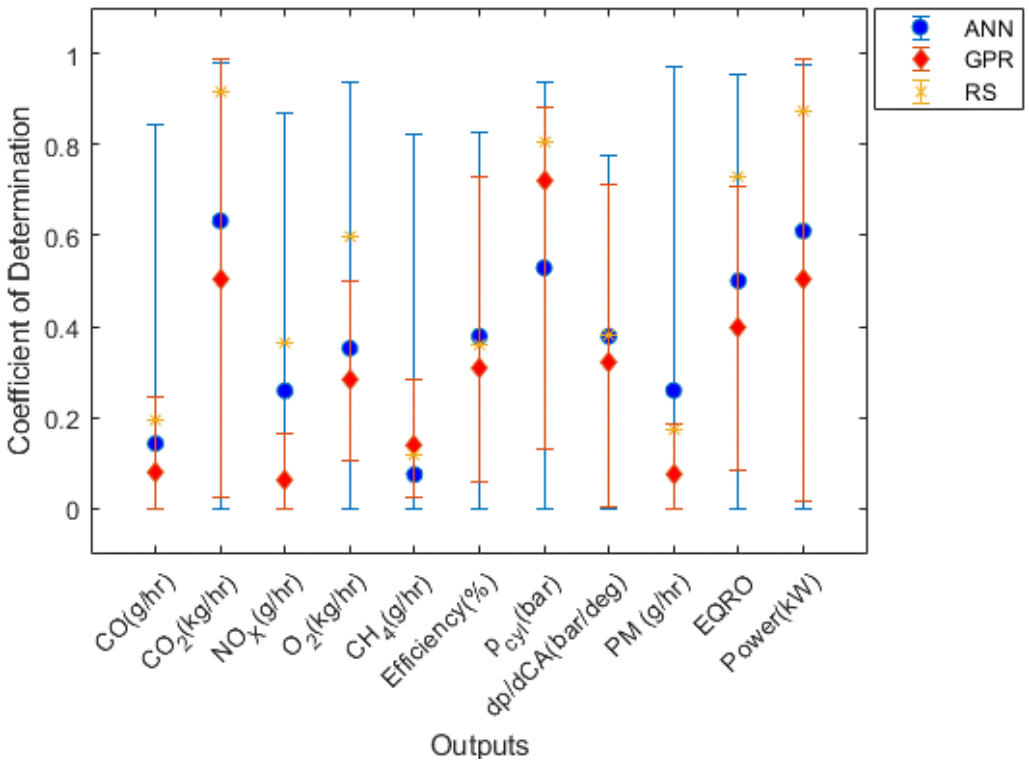


Figure 4.7: Coefficient of Determination for the Modeling Survey with Box Behnken Data (Validation Results)

Figure 4.7 shows that the 500 model averaged coefficient of determination across all outputs has decreased dramatically with many values below 0.5. As expected, the validation results show lower performance than the training data results. With a coefficient of determination around 0.3 on average, the Gaussian process regression models underperform compared to the response surface models. Only the CH_4 coefficient of determination has an average value for the response surface that is lower than the Gaussian process regression value. The neural network models show better coefficient of determination results than the Gaussian process regression models with an average of approximately 0.4 overall. This is still lower than the response surface results.

For both machine learning modeling methods, the wide range of performance indicates a lack of reliability in the models. In some cases, the Gaussian process regression is more constrained with the CO, NO_X, CH_4 and particulate matter all showing coefficients of determination that have a range of 0.2 to 0.3. These same models also show some of the lowest average coefficient of determination results across all methods. Figure 4.8 shows a comparison of the three modeling techniques using the normalized RMSE.

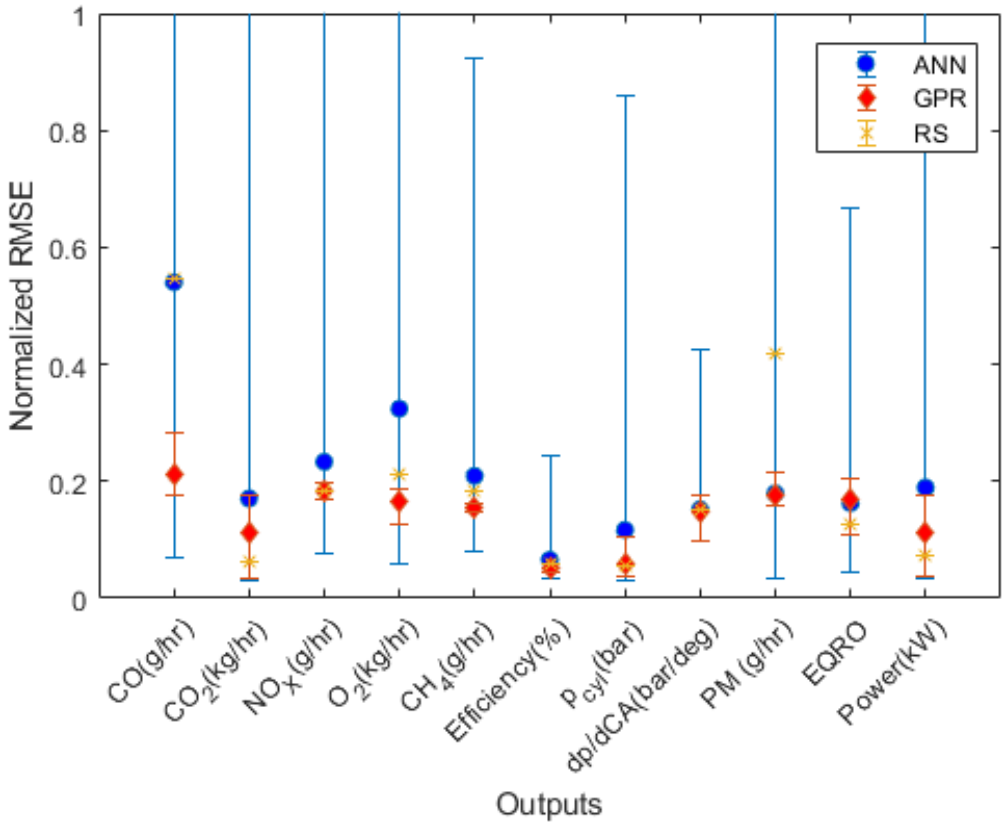


Figure 4.8: Normalized RMSE for the ANN and GPR Modeling Survey with Box Behnken Data (Validation Results)

The range of the nRMSE for the neural network across each of the outputs is very large, exceeding 1 in 6 of the 11 models. This indicates that these models are unreliable. The Gaussian process regression mean nRMSE results are more constrained compared to the neural network values. The average variation in the Gaussian process regression nRMSE is only 0.07. The highest range of the nRMSE approaches 0.14 which is still significant but much less than the neural network results. This indicates that the Gaussian process regression generally has a more consistent nRMSE even though the coefficient of determination results were worse. Table 4.4 shows a summary of the mean values.

Table 4.4: Normalized RMSE results for the modeling survey using the Box Behnken data (validation results)

Outputs	Mean Normalized Root Mean Square Error *		
	Response Surface	Artificial Neural Network	Gaussian Process Regression
CO	0.545	0.541	**0.212**
CO_2	**0.064**	0.171	0.111
NO_X	0.185	0.233	**0.183**
O_2	0.212	0.324	**0.165**
CH_4	0.185	0.209	**0.156**
Efficiency	0.058	0.066	**0.053**
p_{cyl}	**0.055**	0.117	0.060
dp/dCA	0.152	0.152	**0.147**
PM	0.417	0.179	**0.177**
EQRO	**0.128**	0.163	0.168
Power	**0.073**	0.189	0.111
Average	0.189	0.213	**0.140**

* Lowest nRMSE in bold.

In the training data, the response surface showed the best results. This is no longer the case when the non-training data is considered. Some cases still exist where the response surface shows lower nRMSE results than the neural network and Gaussian process regression models. This includes the CO_2, peak cylinder pressure, EQRO, and gross indicated power. The overall average nRMSE

results now indicate that Gaussian process regression has the lowest overall nRMSE of 0.14. The neural network models on average show the highest nRMSE of 0.213. The lower normalized RMSE combined with the significantly lower variations in both the normalized error and coefficient of determination indicates that the Gaussian process regression models are more robust and have the best performance. Despite this, there is still a wide range of variations in the results. Many of the models could also be improved to reduce their normalized root mean square errors, bringing them below 0.1. Techniques for improving the robustness and performance of these models are presented in the following chapters.

Chapter 5: Hyperparameter Optimization for Improved Machine Learning Performance and Robustness

This chapter proposes a method for improving machine learning model robustness and reducing modeling error by automatically selecting model hyperparameters using Bayesian optimization. Chapter 4 shows that despite being able to model more complex data sets, the machine learning model performance decreases as the density of data reduces to the point where response surface models outperform the machine learning models. The results also show a large variation in the coefficient of determination and normalized root mean square error due to the variation in the hyperparameters. This demonstrates that hyperparameter selection can have a significant impact on model performance and that a means of selecting appropriate hyperparameters is needed. This section addresses this by proposing an automated means of hyperparameter selection for the Gaussian process regression and artificial neural network models.

5.1 Bayesian Optimization

Chapter 4 uses a randomized method of selecting model hyperparameters. An optimization based approach for identification of viable and high performing hyperparameter configurations is presented here that uses Bayesian optimization to rapidly iterate through different hyperparameters and select the best possible configuration. For the neural network this means selecting a training function, the number of hidden layers and the number of nodes in each of the hidden layers. For the Gaussian process regression, the hyperparameters being selected are the kernel function and the initial input noise variance. Appendix C.1 and C.2 show the ranges of these different hyperparameters.

The Bayesian optimization utilizes the properties of the gaussian distribution and gaussian processes to search out and evaluate different hyperparameters as it attempts to find the lowest RMSE producing hyperparameter configuration for the training data [44]. The Bayesian optimization creates a prior based on the gaussian distribution and uses the probability distribution to identify the next sample point for the optimization method. The function that selects the next point is called the acquisition function. In this work, the "expected improvement" acquisition function is used to find the best hyperparameter configuration for the training data. This acquisition function seeks out and evaluates the expected location of the lowest model RMSE, according to the gaussian process that is defined by the already evaluated optimization points. The optimization iterates on this process, updating the gaussian process optimization model, as it re-evaluates the location of the lowest RMSE point to try to reduce the error [74]. The MATLAB Statistics and Machine Learning Toolbox is used to implement the Bayesian optimization for hyperparameter selection.

5.2 Hyperparameter Optimization Methodology

Generating machine learning models with hyperparameter optimization adds an extra layer of complexity to the modeling process. The methodology for developing the hyperparameter optimized models is described in Figure 5.1.

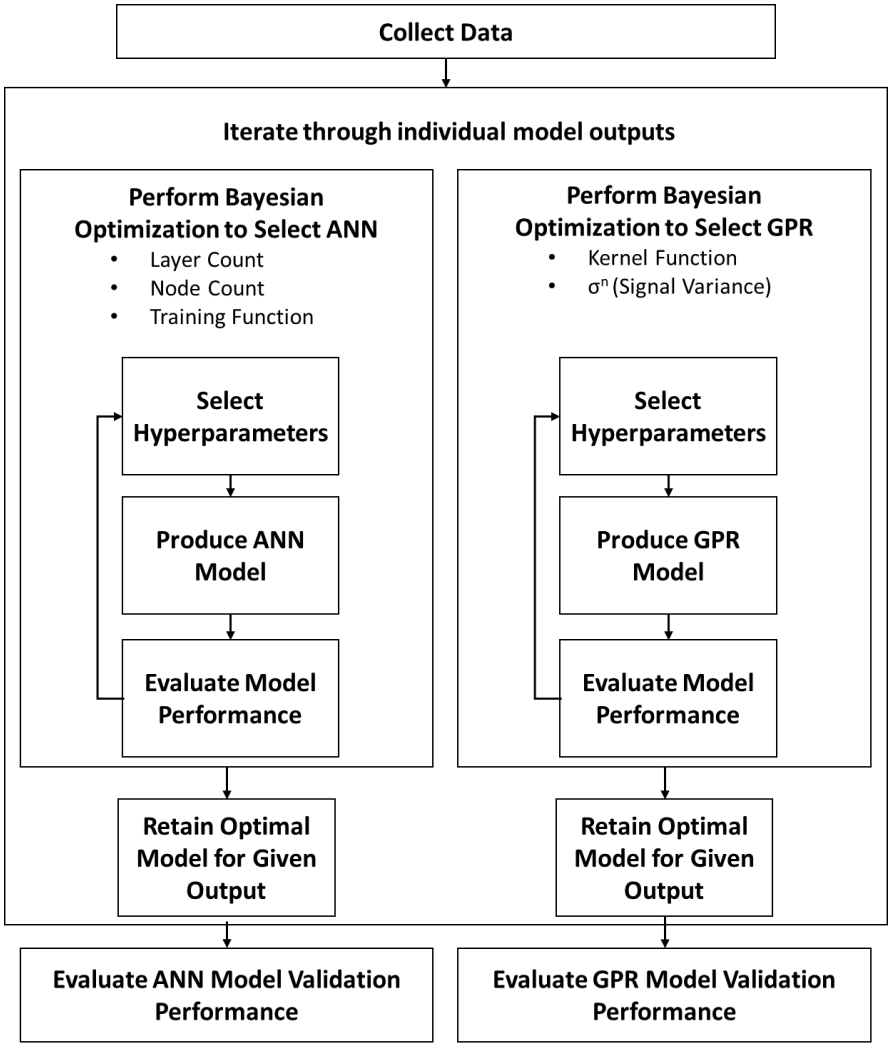


Figure 5.1: Hyperparameter Optimization Methodology

The model creation process in Figure 5.1 begins with collecting the data set, continues with the Bayesian optimization for each model, and ends with measuring the validation performance of the complete set of models for both ANN and GPR. In this process, each output is modeled independently based on the complete input vector from Table 3.1. Each output is modeled using both ANN and GPR. As each model is being created, a Bayesian optimization is used to select the hyperparameters for the model. Up to 1000 different sets of hyperparameters are evaluated for

each output for both ANN and GPR. The optimization creates a model for each evaluated set of hyperparameters. The hyperparameter optimization then calculates the training data based RMSE and then iterates to find the hyperparameter configuration that will yield the lowest RMSE. Once complete the model that produces the lowest RMSE is selected and retained [34]. This process then repeats for the other outputs.

To assess the modeling method robustness, this entire process is repeated ten times for output, generating a different set of final hyperparameters for each optimization instance. These variations are expected as the hyperparameters do not provide for a completely continuous optimization problem and local minimums within the optimization process are expected to exist. Studying repeated iterations provides an understanding for how much these values can vary even within the better performing models [34].

5.3 Hyperparameter Optimization Model Results

The hyperparameter optimization approach is applied to both the neural network and Gaussian process regression modeling methods and compared to response surface models. Both the full factorial data set and Box Behnken data set are modeled using the hyperparameter optimization techniques. For the ANN, the Box Behnken data, Bayesian Regularization was most commonly determined to be the best training method. The full factorial data set used Bayesian Regularization, Levenberg-Marquardt and BFGS quasi-Newton approximately equally. Of these three, only BFGS quasi-Newton is unexpected as the other two are commonly used in engine modeling [41]. For Gaussian process regression, the different permutations of the Matérn based kernels were the most selected. These kernels use spatial statistics to calculate the distance between measured operating

points to determine the covariance which appeared to be the best approach to modeling engine systems [44].

Most of the final optimized neural network models tended towards 3 or 4 layers. Higher layer counts tend to produce higher performing results for the training data as they can represent more complex processes. The following sections show the model performance results for the two different operating spaces.

5.3.1 Full Factorial Data Hyperparameter Optimization Validation Results

This section presents the hyperparameter optimized model results for the full factorial data set. Only the validation data results are presented. The training data results for the full factorial hyperparameter optimized models are shown in Appendix E.1. They are included for completeness sake as Chapter 4 already demonstrated the there is a noticeable reduction in the model performance when transitioning from training to validation data results. The validation results are used as they show a more typical model performance measurement corresponding to data sets that were not made available during training. Figure 5.2 shows the coefficient of determination results for the validation data.

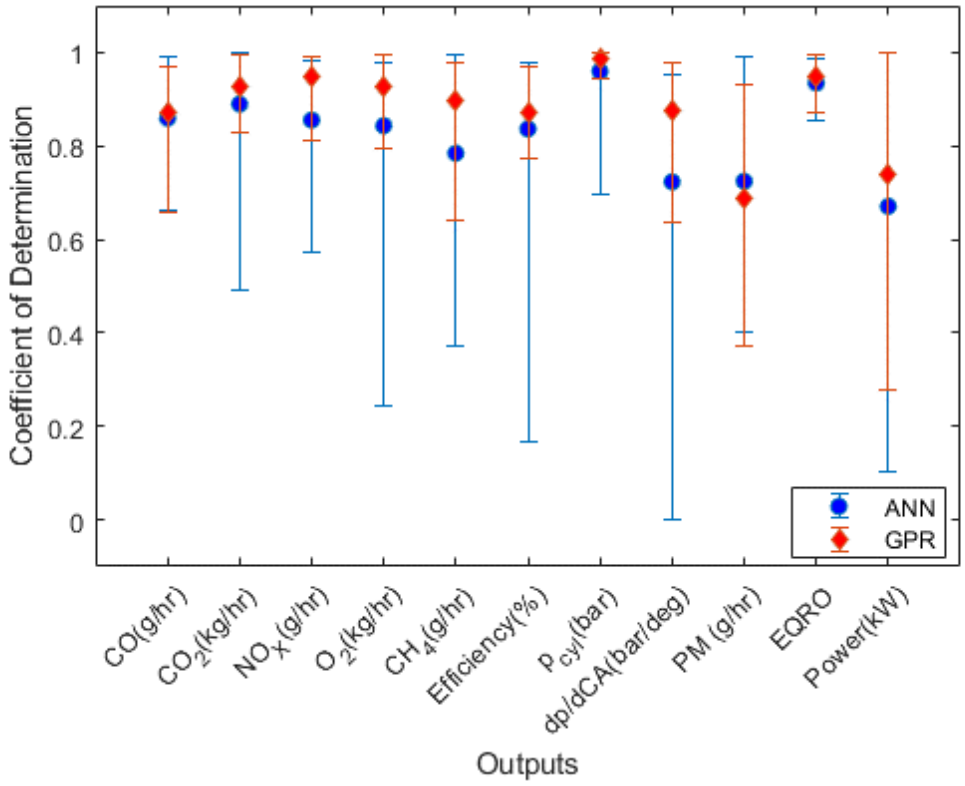


Figure 5.2: Coefficient of Determination for Hyperparameter Optimized Modeling with Full Factorial Data (Validation Results)

From Figure 5.2 it is apparent that the coefficient of determination varies by a wide range, particularly for neural network models. For neural networks, this variation averages about 0.5. For the Gaussian process regression, the range tends to be much more constrained at around 0.3. The only exceptions to this are the gross indicated power and the particulate matter. On average the mean coefficient of determination values are above 0.7 with the Gaussian process regression generally being above 0.8. This is a large improvement over the results presented in section 4.1.2. Figure 5.3 shows the normalized RMSE results.

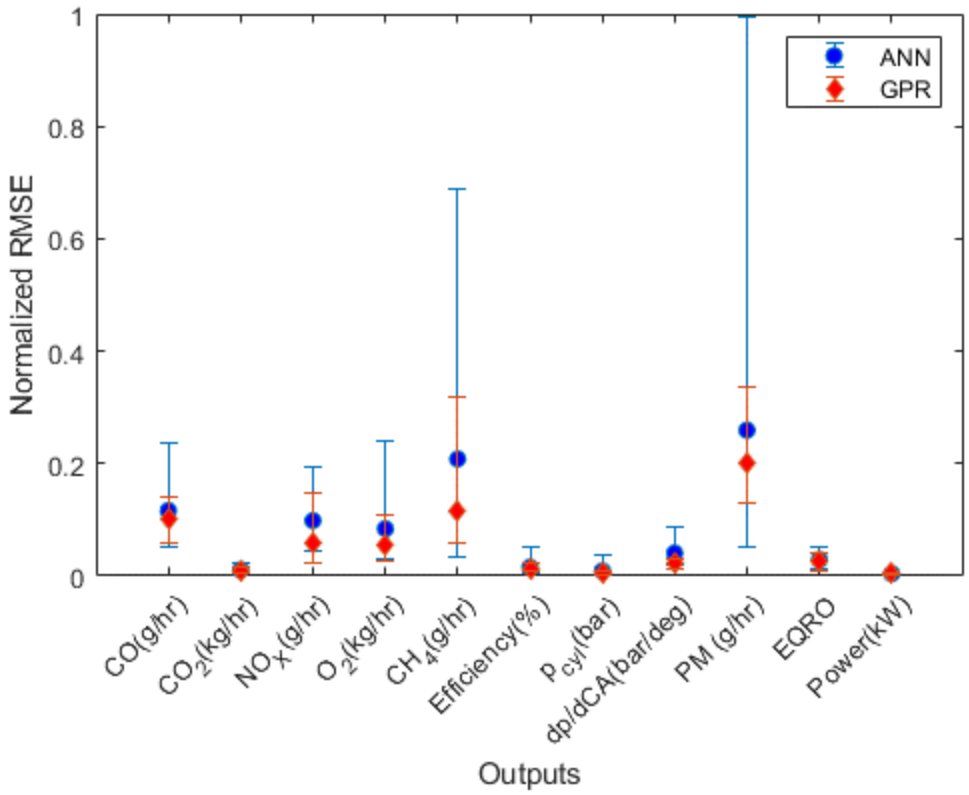


Figure 5.3: Normalized RMSE for Hyperparameter Optimized Modeling with Full Factorial Data (Validation Results)

Figure 5.3 shows that the nRMSE results are consistently below 1. This is a significant improvement over the nRMSE results from Section 4.1.2 which had ranges that exceeded 1 for most outputs, particularly for neural networks. In Figure 5.3, the highest nRMSE measured is for the particulate matter using neural networks and just approaches 1. The average nRMSE for all models is below 0.3, indicating an improvement in model performance, particularly for neural networks. The Gaussian process regression results show much less variation than the neural network results as well as an overall lower error. The average nRMSE compared to the survey nRMSE in Chapter 4 can be seen in Table 5.1.

Table 5.1: Comparison of nRMSE for hyperparameter optimized modeling and the modeling survey using the full factorial data (validation results)

Outputs	Machine Learning Survey		Hyperparameter Optimization	
	Artificial Neural Network	Gaussian Process Regression	Artificial Neural Network	Gaussian Process Regression
CO	0.459	0.165	0.116	**0.100**
CO_2	0.062	0.026	0.011	**0.010**
NO_X	0.289	0.124	0.098	**0.059**
O_2	0.163	0.100	0.084	**0.056**
CH_4	0.409	0.180	0.208	**0.114**
Efficiency	0.035	0.022	0.015	**0.013**
p_{cyl}	0.104	0.022	0.009	**0.006**
dp/dCA	0.051	0.049	0.041	**0.023**
PM	0.505	0.245	0.259	**0.200**
EQRO	0.075	0.054	0.029	**0.025**
Power	0.068	0.017	**0.004**	**0.004**
Average	0.202	0.091	0.079	**0.055**

* Lowest nRMSE in bold.

The results in Table 5.1 show that the Gaussian process regression models performs extremely well with an overall average nRMSE of only 0.06. This is well within the margin of error for the

measurements from the repeatability study. The artificial neural network results show a higher average nRMSE of 0.08. With very few exceptions, the hyperparameter optimized models all have nRMSE results below 0.1. Only the CO, CH_4 and particulate matter nRMSE for both models are at or above 0.1 on average.

Comparing these results to the average nRMSE from the models from Section 4.1.2 shows that the hyperparameter optimization has significantly decreased the nRMSE for every model. This shows that for a high density data set, hyperparameter optimization improves the model performance to the point where the models can properly characterize the operating space. This shows that sufficient data is available to support the modeling when the right hyperparameters are selected and that hyperparameter selection is crucial to obtaining sufficient model performance.

5.3.2 Box Behnken Data Hyperparameter Optimization Validation Results

The Box Behnken data set is the main data set used throughout this research. It represents a large operating space that has been sampled using the Box Behnken design of experiments method to measure as little data as possible while still capturing the core relationships between the inputs and outputs [75]. This section presents the hyperparameter optimized model validation data performance results for the Box Behnken based models. The training data results are provided in Appendix E.2. Figure 5.4 shows the Box Behnken coefficient of determination results for the validation data.

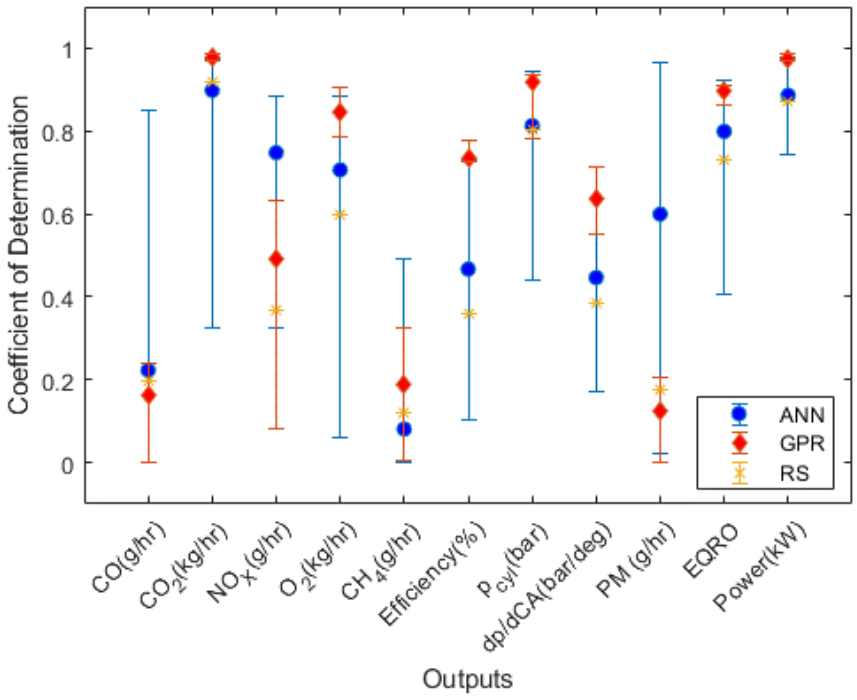


Figure 5.4: Coefficient of Determination for Hyperparameter Optimized Modeling with Box Behnken Data (Validation Results)

In Figure 5.4, the ANN results show the most variation with most values showing a range of 0.5 to 0.7 in the coefficient of determination. The particulate matter shows the largest variation of 0.9, while the gross indicated power shows the smallest variation of 0.3. The GPR models are more constrained with variations close to 0 for the gross indicated power and CO_2, ranging from 0.1 to 0.3 for most other models. NO_X shows the largest range of 0.5.

In most cases the GPR models show the highest average coefficient of determination. The only exceptions are the CO, NO_X, and particulate matter. Comparing these results to the training data in Appendix E.2 shows that the Gaussian process regression models were able to maintain a greater percentage of their overall performance compared to the other two modeling techniques. Despite

this, the CO, CH_4, and particulate matter models show unusually low average coefficient of determination results, indicating that the GPR models were unable to fully characterize these data sets. Figure 5.5 shows the nRMSE results for the Box Behnken validation data.

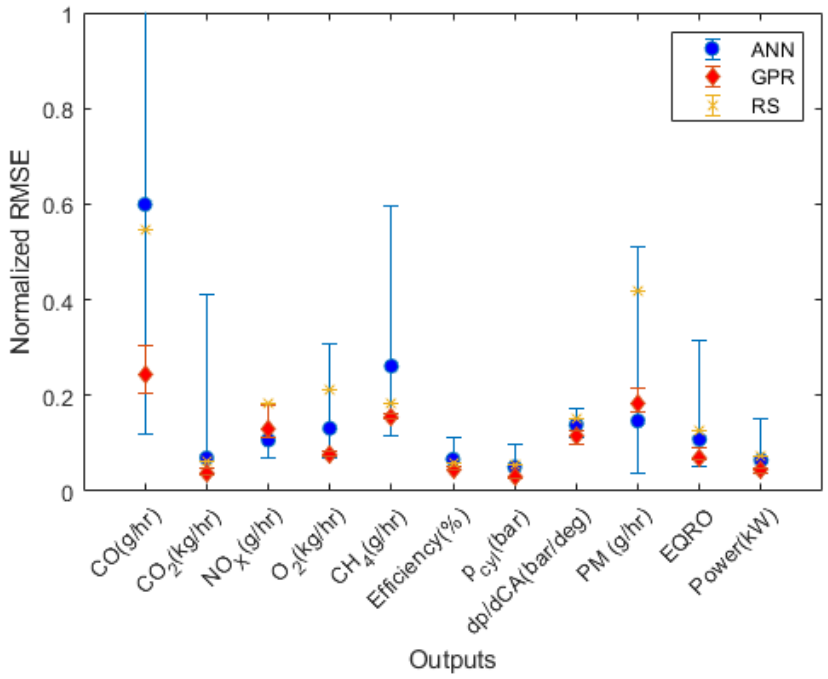


Figure 5.5: Normalized RMSE for Hyperparameter Optimized Modeling with Box Behnken Data (Validation Results)

In Figure 5.5 it is apparent that the ANN results, besides having a higher variation in COD, also show a much higher range for the normalized root mean square error. The GPR nRMSE values have a lower variation and consistently lower average value. This is true even for the CO which shows a lower coefficient of determination for GPR. This is a case where the measurement of correlation does not reflect the magnitude of the error in the results. This is the main reason that the focus of the model performance relies primarily on nRMSE as a measurement of the prediction

error itself. While the coefficient of determination is valuable in demonstrating the model's ability to reflect the original system across the operating space, a high coefficient of determination alone does not guarantee accurate predictions. Table 5.2 shows the summarized average nRMSE results.

Table 5.2: Comparison of nRMSE results for hyperparameter optimized modeling and the modeling survey using the Box Behnken data (validation results)

Outputs	Modeling Survey			Hyperparameter Optimization	
	Response Surface	Artificial Neural Network	Gaussian Process Regression	Artificial Neural Network	Gaussian Process Regression
CO	0.545	0.541	**0.212**	0.599	0.243
CO_2	0.064	0.171	0.111	0.069	**0.038**
NO_X	0.185	0.233	0.183	**0.107**	0.129
O_2	0.212	0.324	0.165	0.131	**0.077**
CH_4	0.185	0.209	0.156	0.261	**0.155**
Efficiency	0.058	0.066	0.053	0.066	**0.046**
p_{cyl}	0.055	0.117	0.060	0.050	**0.032**
dp/dCA	0.152	0.152	0.147	0.139	**0.116**
PM	0.417	0.179	0.177	**0.147**	0.184
EQRO	0.128	0.163	0.168	0.108	**0.070**
Power	0.073	0.189	0.111	0.065	**0.043**
Average	0.189	0.213	0.140	0.158	**0.103**

* Lowest nRMSE in bold.

Table 5.2 shows that overall, the hyperparameter optimized Gaussian process regression models have the lowest nRMSE at 0.103 on average. Besides being more robust, the average normalized RMSE is consistently lower for everything except the particulate matter and the NO_X results, where the neural network model performs better on average. The average survey GPR model showed a slightly lower nRMSE for CO. This is the only case where the survey average outperforms the hyperparameter optimized models.

Comparing these nRMSE results to the full factorial data from Table 5.1 shows that the reduced data resolution of the Box Behnken data set increased the nRMSE values for all the machine learning modeling methods. Despite this, the hyperparameter optimized GPR model performed well for most models with nRMSE values below 0.1 for the CO_2, O_2, thermal efficiency, peak cylinder pressure, EQRO, and gross indicated power. Only the CO shows an nRMSE above 0.2 and the average nRMSE across all GPR models is just above 0.1 at 0.103. Compared to the survey GPR models, the average reduction of 0.04 in the nRMSE caused by the hyperparameter optimization has allowed the GPR model to nearly meet the 0.1 overall threshold set for model nRMSE performance.

The objective of this section was to analyze the performance of a machine learning engine modeling method using Bayesian optimization to enhance the selection of the model hyperparameters. Both the full factorial data set and the Box Behnken data set were evaluated. The results confirmed that a smaller operating space with more densely packed data provides better performance for the machine learning models. As the models were tested on the lower density data

set, it was demonstrated that although there was a model performance drop, the hyperparameter optimization improved the overall model performance. The continued use of hyperparameter optimization can provide significant benefits for future research. The remaining modeling work in this thesis builds upon this technique and continues to use it for all machine learning models.

Focusing on the hyperparameter optimization results alone, GPR models yielded overall higher performing and more robust models for both data sets. The robustness of Gaussian process regression was demonstrated by the range of the coefficient of determination and normalized root mean square error results, which were much more constrained, showing less than 10% of the performance variation that neural network models showed. With the exception of NO_X, and particulate matter, GPR consistently performed better compared to the other methods. These results demonstrate the capabilities of the Gaussian process regression model for engine modeling and research. Future engine modeling research can benefit from using this technique, particularly with the hyperparameter optimization. It is for these reasons that the hyperparameter optimized GPR model is the focus of the model enhancement work discussed in the remaining chapters.

Chapter 6: Techniques for Improving Hyperparameter Optimization Model Performance

Chapter 5 demonstrates that hyperparameter optimization provides an improvement to model performance, especially in model robustness. It also demonstrates that Gaussian process regression outperforms both neural networks and the response surface modeling method. Despite this, there is still room for improvement to reduce modeling error for smaller data sets.

The following sections present four techniques have been developed using custom made MATLAB scripts to further reduce modeling error using data sciences and engineering knowledge to alter the structure of the models. These methods are multi-region modeling, sensitivity based input reduction, layered modeling, and hybrid modeling. These methods are applied to the Gaussian process regression based hyperparameter optimized modeling approach using the low density Box Behnken data set with 10 sets of models being generated for each output to assess the modeling performance and robustness. Each technique is presented separately, with each section focusing on a single technique and a separate section presented at the end of this chapter which compares the results.

6.1 Multi-Region Modeling

The goal of multi-region modeling is to reduce the modeling error by dividing the operating region into different operation modes and simplifying the modeling problem. This approach can be applied to any identifiable distinct set of regions within an operating space. The example used in this work is based on the pulse separation, PSEP. This work uses the main Box Behnken data

operating space discussed in Section 3.2.2. This space covers the positive PSEP operating space of the engine which represents a non-premixed combustion mode where the injections of the natural gas and the diesel pilot fuel do not overlap. A separate mode, represented by the negative PSEP data set in Section 3.2.3, exists based on the opposite partially premixed case where the natural gas injection partially overlaps with the diesel injection. This change in injection timing creates a drastically different fuel mixing within the cylinder which leads to very different performance and emissions [7], [56]. The multi-region model uses engineering knowledge of these different operating modes to create independent models of each mode and switch accordingly between them based on the input vector. The detailed methodology of this technique is outlined in the following section.

6.1.1 Methodology

Taking inspiration from engine calibration techniques, the operating space is divided into smaller regions of similar data points that are expected to have similar combustion and emissions behaviours [11], [76]. These regions are then modelled independently. This approach identifies aspects of the engine modeling that contribute to the model complexity and separates the model based on those aspects. In this case, the attribute that was considered is the relative injection timing (RIT) which represents the timing between the natural gas and diesel pilot fuel injections. This is distilled into the pulse separation (PSEP), which is the calculated time in milliseconds between the end of the diesel injection and the start of the natural gas injection. When the injections overlap or the natural gas injection occurs before the diesel injection, this creates a different combustion mode compared to when the injections are spaced apart with the diesel injection taking place first. The

difference in combustion timings, mixing and overall heat release rate makes for a markedly different form of combustion [56].

The engine models presented in the multi-region method are developed using both the Box Behnken data set and the Negative PSEP data set. First, a combined model that uses both training data sets is created to act as a baseline. This model is developed using the hyperparameter optimized Gaussian process regression modeling technique presented in Chapter 5. The multi-region model is developed using the same two data sets, with each data set being used to create a separate set of models for that data set's respective region. One set of models is created for the positive PSEP region represented by the Box Behnken data set and another set of models is created for the Negative PSEP data region. For each region, each of the outputs is modeled independently using the hyperparameter optimized Gaussian process regression modeling method. Figure 6.1 shows the structure of the multi-region model.

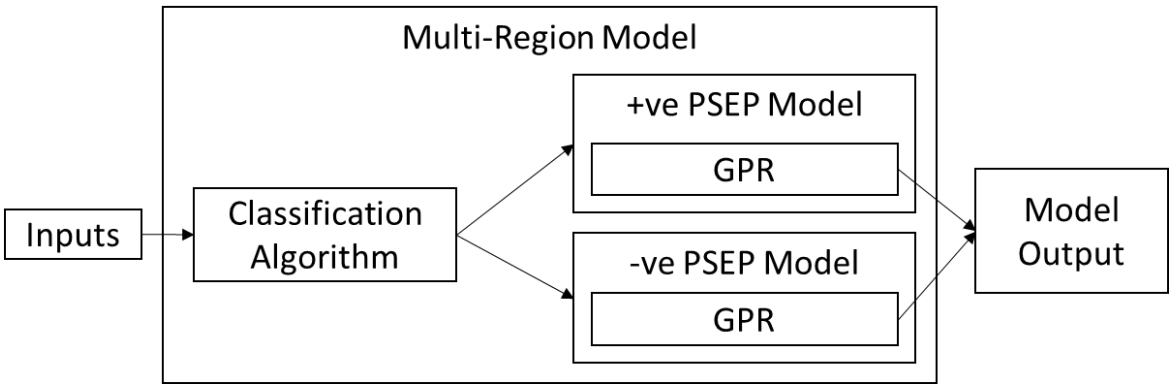


Figure 6.1: Multi-Region Model Structure

The classification algorithm is used to categorize the input data so that it is sent to the correct modeled region when a prediction is being made. For this example, where PSEP, based on RIT, is being used, the inputs themselves can be directly used to mathematically determine which region to use according to Equation 2.1 and Figure 2.9.

6.1.2 Multi-Region Validation Data Results

As with the other machine learning model improvement techniques evaluated in this work, the multi-region technique uses the Gaussian process regression method with hyperparameter optimization as the basis for the individual models. The multi-region model results are presented as both the individual region "+ve PSEP" and "-ve PSEP" results as well as the averaged results of both regions. These two sets of metrics are compared against a combined model that was trained with the data from both regions. The training data performance is presented in Appendix E.3. The results presented here use the randomized validation data from the Box Behnken operating space and the Negative PSEP operating space. 10 sets of models are created for each model type and the aggregated results are presented. Figure 6.2 shows the coefficient of determination results for the validation data.

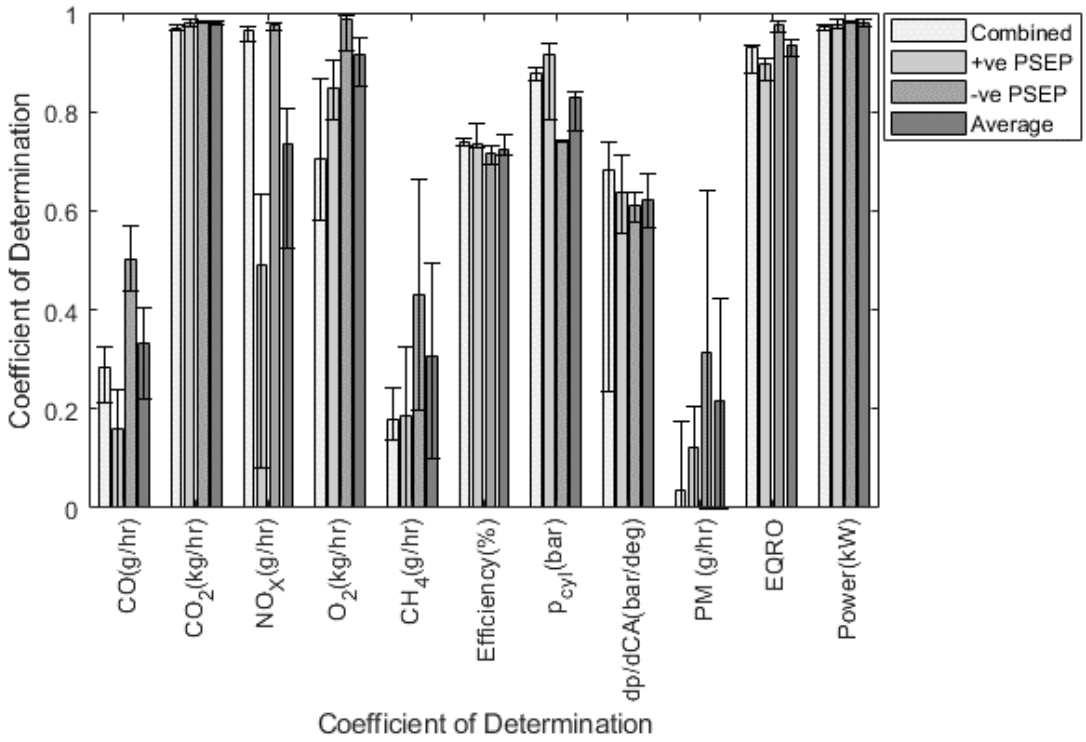


Figure 6.2: Coefficient of Determination for Multi-Region Models Compared with a Combined Single Region Model (Validation Results)

Figure 6.2 demonstrate mixed results for the multi-region model. The +ve PSEP data set overall performs poorly for a number of models, specifically the CO and NO_X where both the -ve PSEP region and combined models perform much better. The particulate matter results are especially interesting as the they indicate that the models perform extremely poorly for this data set. This should be reflected in the normalized RMSE which is shown in Figure 6.3.

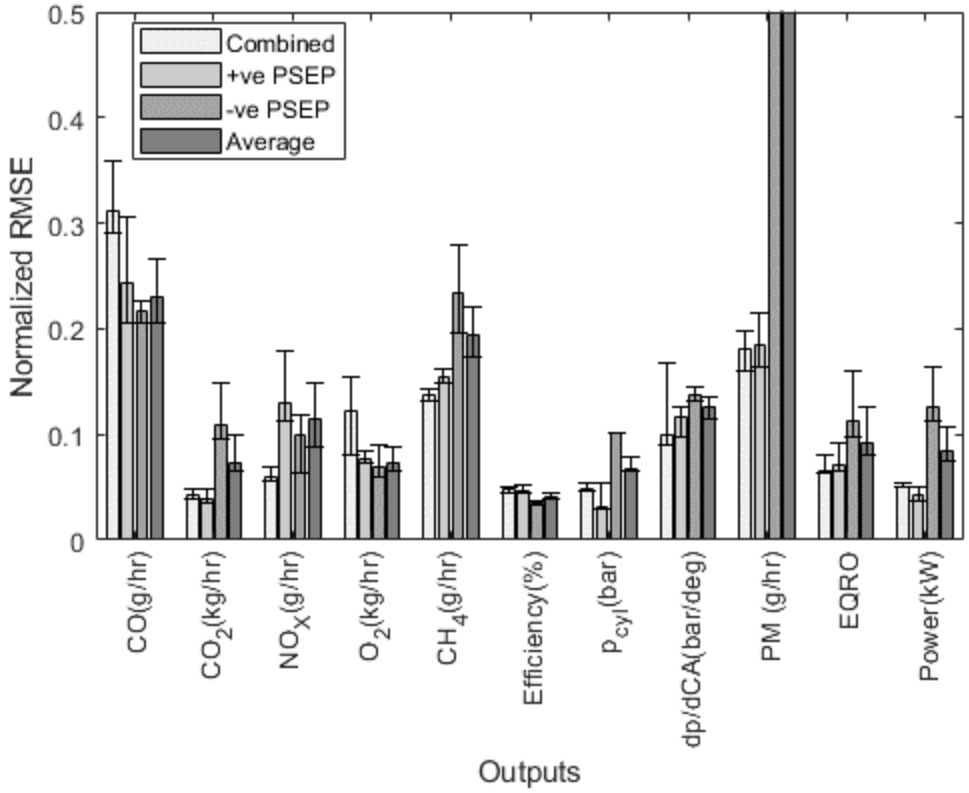


Figure 6.3: Normalized RMSE for Multi-Region Models Compared with a Combined Single Region Model (Validation Results)

As expected, Figure 6.3 demonstrates that the -ve PSEP model for particulate matter is unable to predict the outputs for the validation data points. This is partially a result of the particulate matter emissions being low in this region and not having sufficient variability in the data to overcome the effects of signal noise from the particulate matter measurement bench [30]. This directly impacts the averaged nRMSE for the multi-region model. Despite the low coefficient of determination, the nRMSE remains low, at approximately 0.2 for the combined model and +ve PSEP regions. Examining the range of the nRMSE shows that the overall models are robust with most models varying by less that 0.05. While many of the ranges between the combined model and the multi-

region models do cross over, the shift in the average and the error bars indicates that the performance differences should be consistent across most models. Table 6.1 summarizes the nRMSE results.

Table 6.1: Normalized RMSE for the multi-region model compared to a combined single region model (validation data)

Outputs	Multi-Region nRMSE *			Combined Data nRMSE	% Reduction in nRMSE *		
	+ve PSEP	-ve PSEP	Multi-Region Average		+ve PSEP	-ve PSEP	Multi-Region Average
CO	**0.243**	**0.216**	**0.230**	0.311	**21.86**	**30.41**	**26.21**
CO_2	**0.038**	0.109	0.074	0.043	**9.827**	-155.8	-70.93
NO_X	0.129	0.099	0.114	0.060	-117.1	-66.89	-90.00
O_2	**0.077**	**0.07**	**0.074**	0.123	**37.15**	**43.43**	**40.24**
CH_4	0.155	0.235	0.195	0.138	-12.61	-70.25	-41.30
Efficiency	**0.046**	**0.035**	**0.041**	0.048	**2.995**	**26.69**	**15.63**
p_{cyl}	**0.032**	0.102	0.067	0.048	**33.03**	-113.6	-39.58
dp/dCA	0.116	0.138	0.127	0.100	-15.46	-37.59	-27.00
PM	0.184	2.060	1.122	0.180	-2.118	-1044	-523.3
EQRO	0.070	0.113	0.092	0.065	-7.638	-73.65	-40.77
Power	**0.043**	0.127	0.085	0.051	**15.63**	-148.3	-66.67
Average	0.109	0.318	0.202	0.106	-3.129	-146.3	-74.32

* Lowest nRMSE in bold.

As Table 6.1 shows, the multi-region model produces better results for 3 out of 11 outputs. For CO, the multi-region models reduces the nRMSE compared to the combined model by more than the margin of error presented by the error bars. The same is true of the O_2 model and to a lesser extent, thermal efficiency. This result is notable as it demonstrates that in some cases there is a benefit to splitting a data set between two different regions and modeling the regions separately. These models showed a nRMSE decrease ranging from 16% to 40%.

The overall best performing models are the combined data models. The average increase in nRMSE using the multi-region method, applied to all models, is 74%. While the injection crossover creates a fundamentally different form of combustion, this does not have a large enough impact on the physics such that the complexity cannot be overcome by machine learning. In most cases, the benefit of combining the data outweighs the benefit of removing the complexity. The combined model performs much better for cases where one region has an especially difficult time modeling a certain parameter. This is shown for the particulate matter and CH_4 nRMSE results which are improved by combining both data sets. This shows that the machine learning modeling methods are capable of correcting for erroneous or incomplete data if enough useful data is made available to the modeling method.

While response surface modeling, as it is used in industry, often uses multiple regions to simplify the modeling method, these preliminary results indicate that machine learning methods do not benefit from this approach in most cases [11]. These results show that quantity of data is very valuable for machine learning and any attempt to improve machine learning model performance

should not reduce the size of the data set used to train the models. This observation motivates the other methods demonstrated in this work. While these results do not indicate that the multi-region modeling method should always replace the normal machine learning, it does present an alternative modeling approach that could be used when even hyperparameter optimized modeling is unable to produce usable models.

6.2 Sensitivity Analysis Based Input Reduction

The sensitivity analysis based input reduction method analyzes the inputs used for each of the output models and programmatically determines what inputs to use for each model. The hyperparameter optimized models from Chapter 5 use all of the inputs listed in Table 3.1 for each output. This is done to ensure that each output model has access to the same data set and that all the useful control and system information that defines the operating point is available for the machine learning models. While this input set can be considered to be complete, not every input is useful for every output model [11]. It is expected that the machine learning models would intuitively determine which inputs are important and ignore the irrelevant inputs by reducing their weighting in the model. Previous research on a biodiesel fuel engine and a common-rail diesel engine shows that machine learning models only need access to the most important inputs when producing models, as some inputs have little to no benefit [41], [77]. Additional research outside the engine space that focused on water resource sediment load analysis demonstrated that removing inputs that have a low impact on the system performance from a sensitivity standpoint reduces signal noise and improves the overall modeling performance [26]. By using every available input, it is possible that some of the inputs have a negative impact on model performance for certain outputs.

To address this, a sensitivity analysis is used to identify the most relevant inputs for each desired output. This sensitivity analysis starts by using the models produced in Chapter 5 for the Box Behnken data set with hyperparameter optimization and systematically varying the inputs from the training data set. For each data point, each input is systematically varied by +/-50 %, +/-25 % and +/-10 % of the range of values for that input. The resulting 362,797,056 new sets of input for each original data point are evaluated for each output. This is done for each of the 10 sets of models created in Chapter 5. The sensitivity of an output for a particular input is calculated as the change in the output relative to the percentage that the input has been varied as shown in Equation 6.1. The sensitivity is averaged across all data points and permutations of the input for each of those data points.

$$S_k = \frac{\sum_{i=1}^{n_{data}} \left\{ \sum_{j=1}^{n_{perturb}} \left| \frac{Y_{new,i,j} - Y_{pred,i}}{Pt_j} \right| \right\}}{n_{data} \times n_{perturb}} \tag{6.1}$$

S_k is the sensitivity for the kth input and i represents the index of the original data point. Pt_j is the percentage variation of the input, and j represents the index of the different perturbation percentages being evaluated. n_{data} is the number of data points in the data set, $n_{perturb}$ is the number of perturbations for a given input, Y_{new} is the newly calculated model output for the jth perturbation of the ith data point, and Y_{pred} is the ith data point model predicted value without any perturbations. For each output, the inputs are ranked from highest to lowest sensitivity, as shown in Table 6.2 using the Gaussian process regression models as a baseline.

Table 6.2: Input sensitivity for each output based on averaged per percentage variation of the inputs ranked from highest to lowest

Output	Sensitivity Ranking for Each Input										
	1	2	3	4	5	6	7	8	9	10	11
CO	p_{diesel}	PW_{ng}	T_{air}	p_{exh}	ω_{eng}	SOI_{diesel}	p_{air}	SOI_{ng}	p_{ng}	T_{cool}	PW_{diesel}
CO_2	PW_{ng}	p_{diesel}	SOI_{diesel}	T_{air}	SOI_{ng}	PW_{diesel}	p_{air}	p_{ng}	ω_{eng}	p_{exh}	T_{cool}
NO_X	p_{diesel}	PW_{diesel}	PW_{ng}	T_{cool}	T_{air}	p_{ng}	SOI_{ng}	SOI_{diesel}	p_{air}	ω_{eng}	p_{exh}
O_2	p_{air}	SOI_{ng}	SOI_{diesel}	ω_{eng}	T_{cool}	T_{air}	p_{ng}	p_{exh}	p_{diesel}	PW_{ng}	PW_{diesel}
CH_4	p_{diesel}	p_{exh}	T_{air}	ω_{eng}	PW_{diesel}	p_{ng}	SOI_{ng}	PW_{ng}	p_{air}	T_{cool}	SOI_{diesel}
η	p_{ng}	ω_{eng}	PW_{diesel}	T_{air}	T_{cool}	p_{air}	p_{diesel}	SOI_{diesel}	PW_{ng}	p_{exh}	SOI_{ng}
p_{cyl}	SOI_{diesel}	p_{air}	p_{diesel}	ω_{eng}	PW_{ng}	p_{ng}	T_{cool}	PW_{diesel}	p_{exh}	SOI_{ng}	T_{air}
dp/dCA	p_{diesel}	ω_{eng}	PW_{diesel}	SOI_{diesel}	T_{air}	p_{air}	p_{exh}	p_{ng}	SOI_{ng}	PW_{ng}	T_{cool}
PM	p_{diesel}	SOI_{ng}	T_{air}	p_{Exh}	PW_{ng}	SOI_{diesel}	p_{air}	ω_{eng}	p_{ng}	T_{cool}	PW_{diesel}
EQRO	p_{air}	PW_{diesel}	SOI_{diesel}	T_{cool}	PW_{ng}	p_{exh}	p_{ng}	SOI_{ng}	p_{diesel}	ω_{eng}	T_{air}
P_{gross}	SOI_{diesel}	p_{diesel}	T_{cool}	p_{exh}	PW_{ng}	p_{ng}	p_{air}	PW_{diesel}	SOI_{ng}	ω_{eng}	T_{air}

Table 6.2 shows the averaged per percentage variation input sensitivity ranking for each of the outputs. Based on this table, the fuel and air pressures consistently have some of the highest sensitivities for the models. This reflects the engineering intuition and the literature as the pressures impact the quantity and mixing of fuel and air injected, which impacts most of the outputs. [69]. This is often closely followed by an injection duration or starting angle. According to Table 6.2, the temperatures, engine speed, and some of the diesel injection parameters have the lowest effect on the outputs. As the temperatures do not vary substantially during the data collection it is not

expected to have a significant impact on the models. The engine speed is also less likely to directly influence the emissions when the injection durations are time based and not calculated based on crank angle. This means that the engine speed is not a significant factor for the quantity of fuel injected. In the positive PSEP region, the diesel injection parameters have less of an impact on the model compared to the natural gas since it contributes only a small amount to the fuel input from an energy perspective.

6.2.1 Sensitivity Analysis Based Input Reduction Validation Data Results

Based on the results from Table 6.2, new models are generated by removing the least sensitive inputs. The first models start with 11 inputs. The next models have 10, 9, 8 and continuing until only the highest sensitivity input remains. Each of these new models is produced 10 times to allow for an assessment of the robustness and variability in the results. As Gaussian process regression is shown to be the best modeling method, all the results presented in this section apply the input reduction exclusively to Gaussian process regression with hyperparameter optimization technique. Appendix E.4 shows the training data results. Figure 6.4 shows the coefficient of determination results for the validation data.

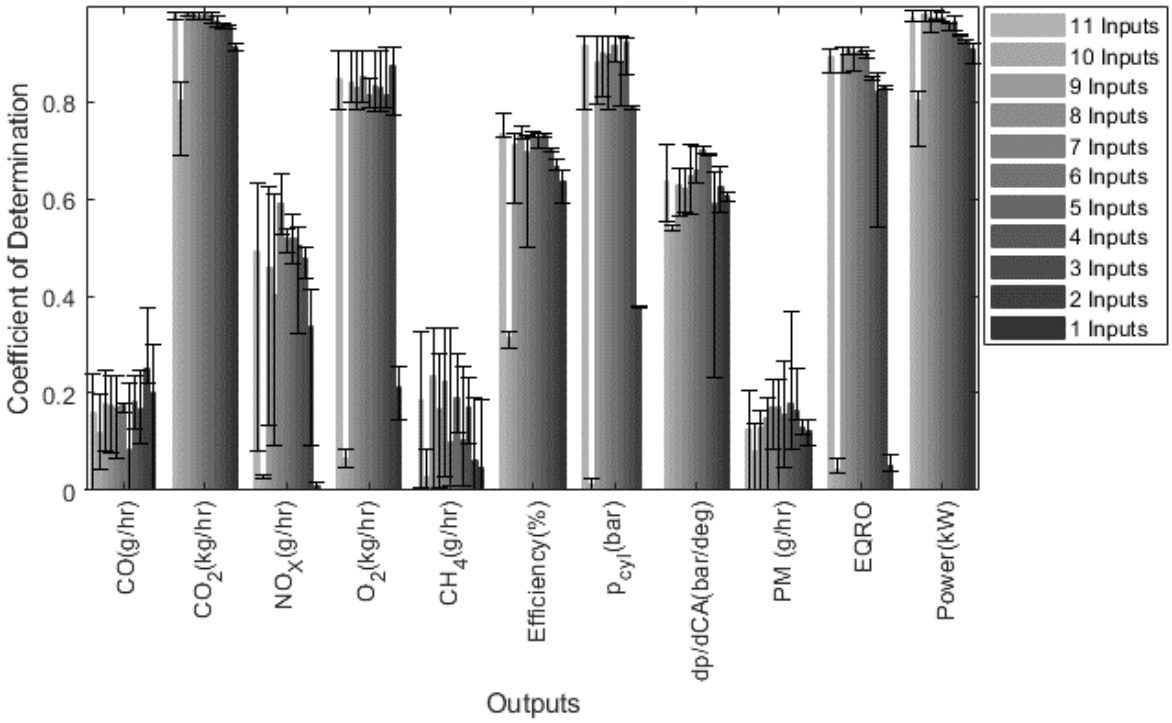


Figure 6.4: Coefficient of Determination for Sensitivity Analysis Based Input Reduction (Validation Data)

Figure 6.4 shows that overall, as the inputs are removed, the ability of the model to follow the trends of the data decreases. This is consistent for all outputs except for CO and O_2. Despite this, many models show an initial increase in the coefficient of determination. This is especially visible for the particulate matter and dp/dCA. What these results indicate is that some of the lower sensitivity inputs are adding additional noise to the modeling process and removing them is improving the models. However, there is some minimum set of inputs that each model requires to properly characterize the data set.

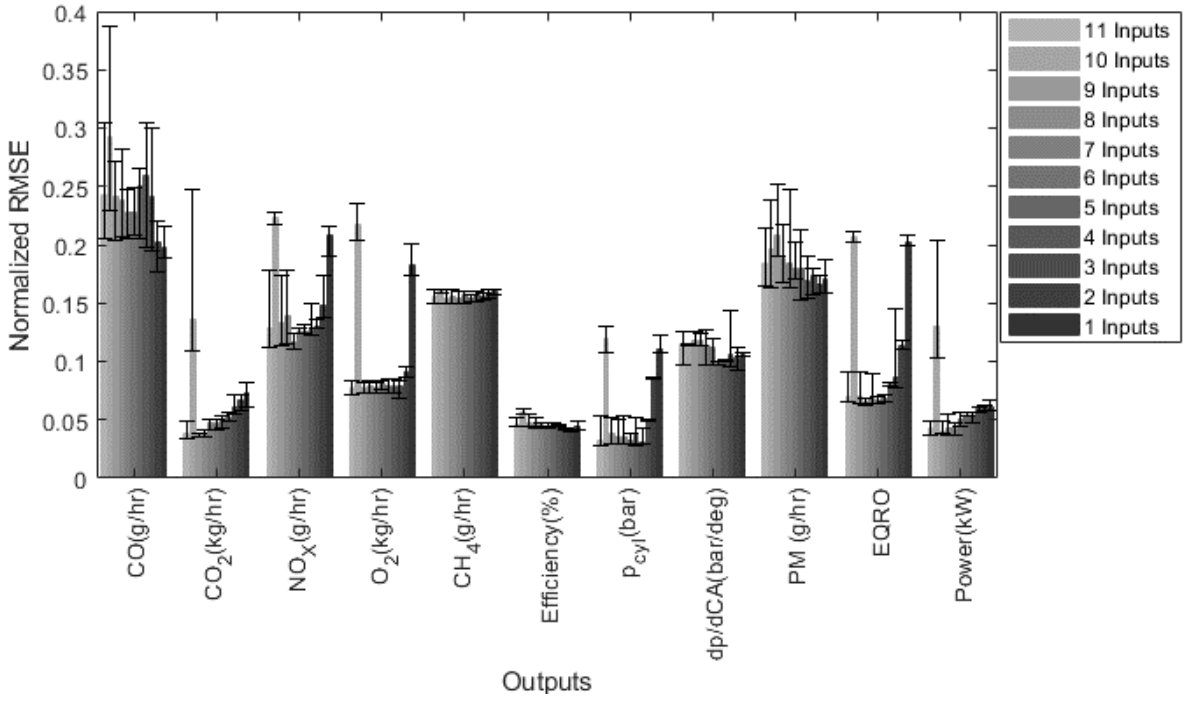


Figure 6.5: Normalized RMSE for Sensitivity Analysis Based Input Reduction (Validation Data)

Based on the results from Figure 6.5, the relative change in nRMSE as inputs are reduced is small. This indicates that much of the performance of the engine is describable by 1 or 2 parameters. There are several cases (CO, NO_X, O_2, CH_4, thermal efficiency, peak cylinder pressure, dp/dCA, and PM) where model performance improves with the decreasing size of the input vector. In some cases, the normalized RMSE results have large variances. For example, the nRMSE for particulate matter and CO vary by up to 0.1 for almost all input iterations. This is significant enough to make it difficult to determine the impact of the input selection. Table 6.3 summarizes these results with respect to the hyperparameter optimization models.

Table 6.3: Normalized RMSE results for sensitivity analysis based input selection (validation results)

Outputs	Sensitivity Based Results	Hyperparameter Based Results	% Reduction in nRMSE	Inputs Used
CO	**0.197**	0.243	**18.85**	1
CO_2	**0.037**	0.038	**4.444**	9
NO_X	**0.117**	0.129	**9.461**	7
O_2	**0.077**	0.077	**0.645**	7
CH_4	**0.154**	0.155	**1.071**	9
Efficiency	**0.041**	0.046	**10.78**	2
p_{cyl}	**0.031**	0.032	**3.731**	4
dp/dCA	**0.098**	0.116	**15.01**	5
PM	**0.166**	0.184	**9.729**	2
EQRO	**0.065**	0.070	**7.416**	8
Power	**0.040**	0.043	**6.908**	9
Average	**0.093**	0.103	**8.004**	

* Lowest nRMSE in bold.

As can be seen in Table 6.3, removing inputs reduced the average model error across all outputs by 8%. CO improved by 19% while dp/dCA improved by 15%. Thermal efficiency, particulate matter, and NO_X showed improvements of 10%. On average each output used only 6 inputs.

6.2.2 Sensitivity Analysis Conclusions

The sensitivity analysis based input reduction method presented a system agnostic approach to identifying the core inputs required for a model and removing inputs that are contributing more system noise than beneficial data. The importance of properly selecting inputs should not be ignored as some inputs have a negative impact on the model which could potentially be made worse by the limited data size. While other approaches to improving model performance without requiring additional data are discussed in the following sections, this method presents a simple, low complexity approach to improving Gaussian process regression models.

6.3 Layered Model

In the layered model, knowledge of the mechanical, thermodynamic, and chemical processes that connect the model inputs to the outputs is used to separate the overall machine learning modeling problem into several smaller interconnected modeling problems. When collecting data from a fully instrumented research engine, additional sensor data is often collected. Many of these measured values represent intermediate steps in the various processes that produce the outputs being modeled. These intermediate parameters can be used to replace some of the inputs from Table 3.1 and can themselves be modeled as hyperparameter optimized Gaussian process regression models. These newly modeled intermediate parameters are then connected together as a network of submodels connecting the original inputs to the final output from Table 3.1.

The objective of this approach is to shift the complexity of the processes being modeled out of the machine learning model and into the model structure. This way, each submodel focuses on a simpler pattern with fewer inputs and less interactions. This has the potential to result in

relationships that are easier to identify with machine learning techniques which reduces the overall error. By extracting the process knowledge from the problem and injecting it into the model structure, this creates a grey box modeling problem which allows an engineer to introduce this knowledge to the model structure as another type of data, in addition to typical numerical sensor data. This also provides the engineer with a better understanding of the inner workings of the engine model and more control over the modeling process, addressing one of the criticisms of machine learning modeling. A sample of the difference between a normal model structure and a layered model is shown in Figure 6.6.

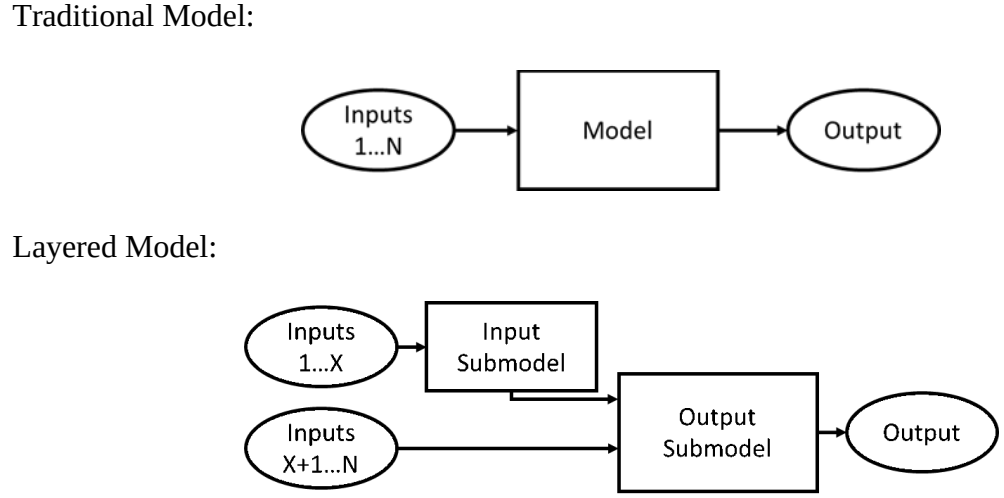


Figure 6.6: Sample Layered Model Comparison

The majority of the submodels represent parameters that are not directly measurable on a production engine. However, a fully instrumented research engine, would typically have access to substantially more sensor data which includes these additional parameters. As such, these parameters can be modeled with information gathered from a research engine and used to generate the layered model that is ultimately implemented on a production engine.

The layered model structure is laid out in Figure 6.7 and is comprised of a total of 6 submodel layers and one input layer. The input layer contains the same inputs from Table 3.1. In this structure, each submodel is dependent on at least one output from the previous layer. For example, the combustion phasing which represented by the parameters, θ_{50} and θ_{10}, are dependent on the value of EQR predicted in layer 2. Similarly, the gross indicated power in layer 4 is dependent on θ_{50}. The detailed interconnections between the submodels are shown in Appendix D.1.

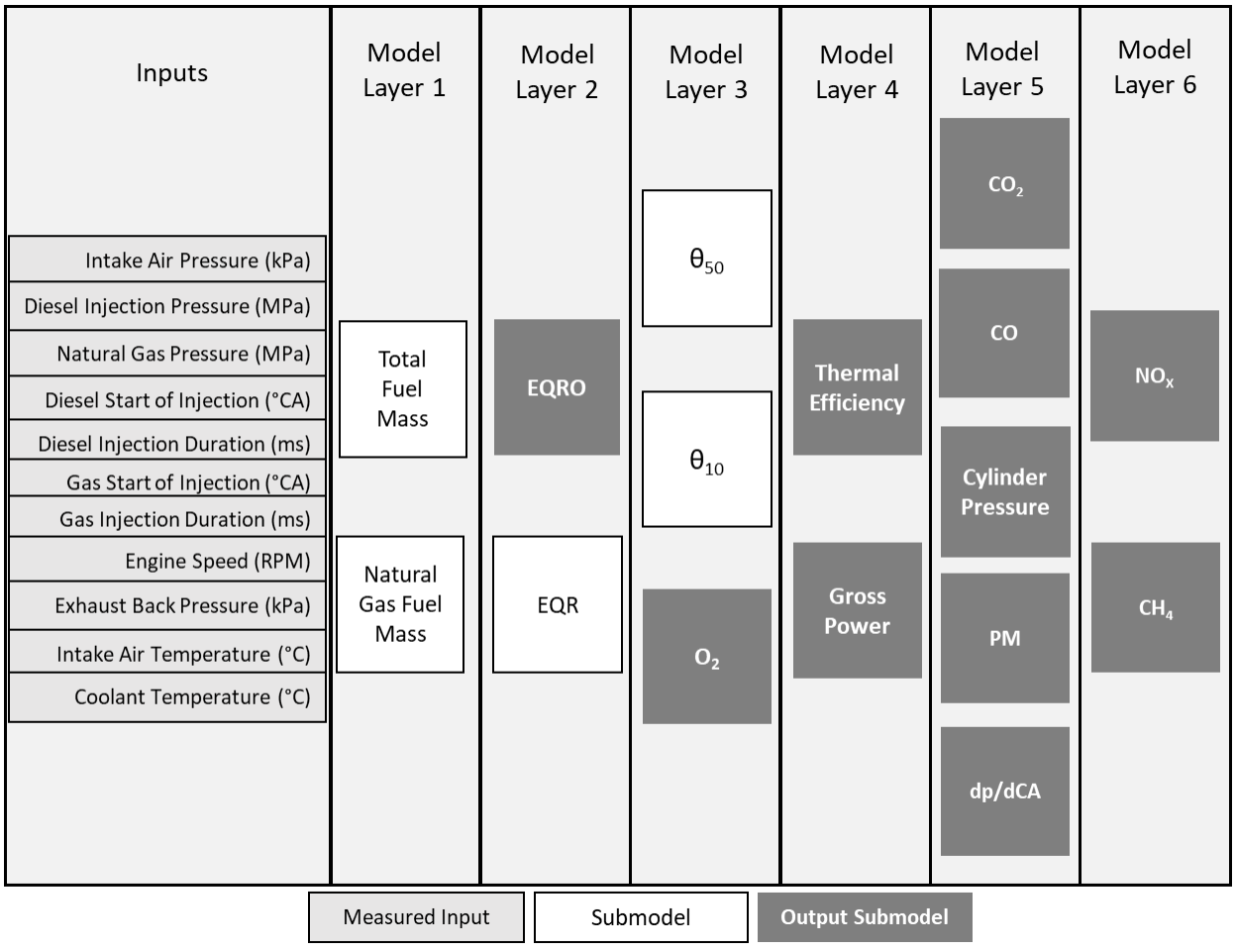


Figure 6.7: Layered Model Structure

The layered model structure is determined based on a understanding of the processes taking place throughout the engine combined with engineering intuition coming from experience working with this system [11]. The equivalence ratio is determined based on the fuel and air mass. The injection parameters for the natural gas and diesel fuels are converted into an overall fuel mass term to predict EQR. The oxygen model is then dependent on the equivalence ratio and the fuel mass. The thermal efficiency and power output are based on EQR and the combustion phasing which are dependent on the injection timings and natural gas fuel mass. The power is also dependent on the fuel mass. The emissions are heavily related to the heat release rate curve. The combustion phasing parameters represent the heat release rate curve and are used as inputs for the models of the emissions along with the thermal efficiency and gross indicated power. Finally, the NO_X and CH_4 are also dependent on the peak cylinder pressure which is determined based on the combustion phasing, fuel mass, equivalence ratio and power. Through this interconnected network of intermediate parameters, each model has its own dependencies based on the previous models in the network.

6.3.1 Training the Layered Model

As the layered model has several interconnected machine learning models, training the layered model has an added element of complexity over the hyperparameter optimization model presented in Chapter 5. Each intermediate parameter is measured and then modeled using a machine learning model. Two different approaches were developed for training each of the submodels used to create the overall layered model. In the first approach, each intermediate submodel is trained using only the measured experimental sensor values from the engine. This is referred to as “Measured” training as it only uses measured data. The second training approach uses predicted inputs to train

the model based on the outputs of the submodels in the previous layers of the layered model. This is referred to as “Predicted” training. Measured training results in individual submodels that most accurately represent the measured experimental data. With this approach, all submodels can be trained simultaneously. Predicted training submodels must be trained one layer at a time as each layer requires training inputs that are generated from the submodels in the previous layers. Predicted training introduces modeling error in the intermediate inputs used for training. This more accurately represents the input data made available to the submodels when the model performance is assessed.

The following sections present the results of the layered modeling approach to improving model performance. Both the measured and predicted training methods are presented. As in the previous sections, each model is trained using the Gaussian process regression method with hyperparameter optimization. Each model is trained a total of 10 times with the mean results and the range shown in the figures. These models are compared to the hyperparameter optimization method first demonstrated in Chapter 5.

6.3.2 Submodel Performance Results

Before examining the layered model results, it is important to look at the submodel performance. Using the measured approach, each submodel is generated and the performance of the submodels is assessed.

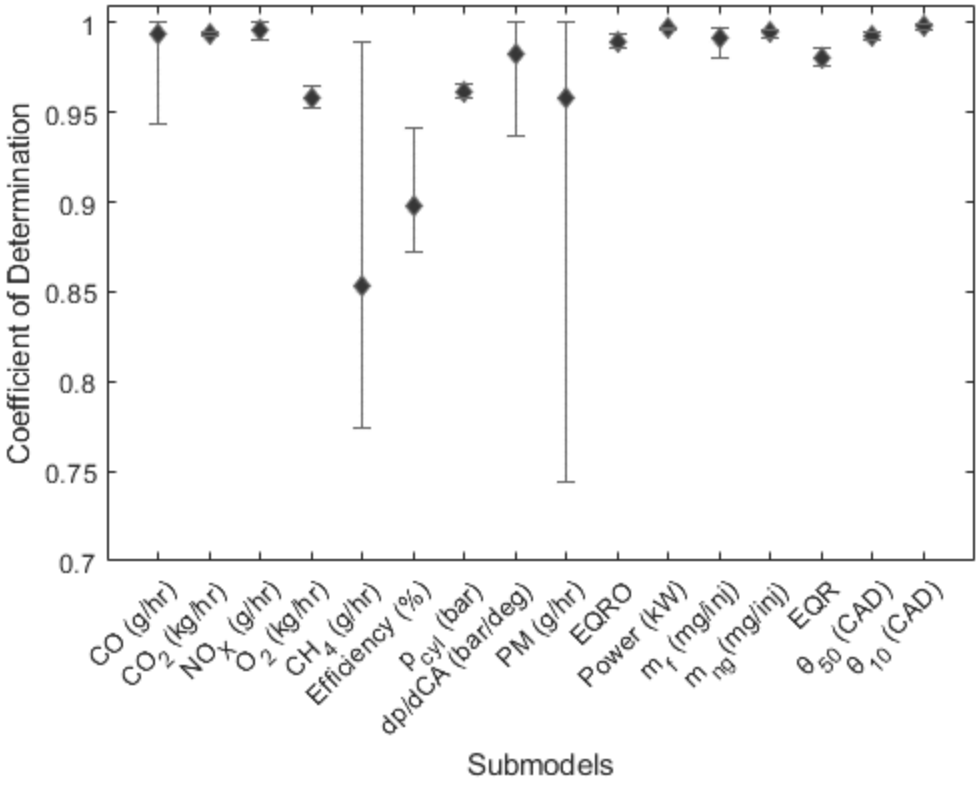


Figure 6.8: Training Data Coefficient of Determination for the Layered Model Submodels

Figure 6.8 shows that the coefficient of determination has some variation but is consistently above 0.7. The largest variation is in the particulate matter and the CH_4. Each of these is an output model in the final layers of Figure 6.7. As such, these lower coefficients of determination would not affect any the other submodels. The CH_4 shows the lowest coefficient of determination of 0.85. As these results are based on the training data, a value of 0.85 can be considered to be low. Generally, these values would be expected to be above or approaching 0.9. The remaining submodels are all at or above this value.

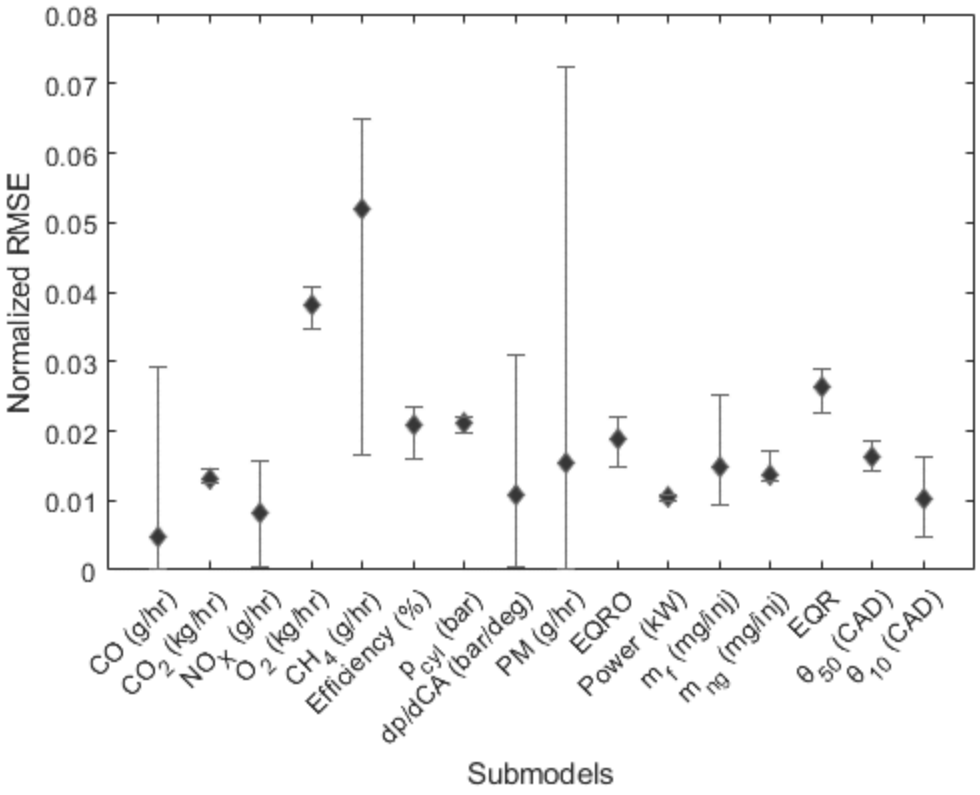


Figure 6.9: Training Data Normalized RMSE for the Layered Model Submodels

In Figure 6.9 the nRMSE is consistently below 0.08 with the average value being below 0.06. This indicates that the submodels are likely to perform well in the layered model as the model error is low. The CH_4, and particulate matter measurements both present a larger range of performance spanning a range of 0.05 to 0.07. This is consistent with the coefficient of determination. This is a wide range of performance relative to the mean value and indicates that these models have more variability which may affect the layered model nRMSE range.

6.3.3 Layered Model Validation Data Results

This section presents the layered model validation data results. The training data performance is discussed in Appendix E.5. Results are presented for the full layered model as laid out in Figure 6.7, developed using both the measured and predicted training methods. These results are compared against the hyperparameter optimization results in Chapter 5.

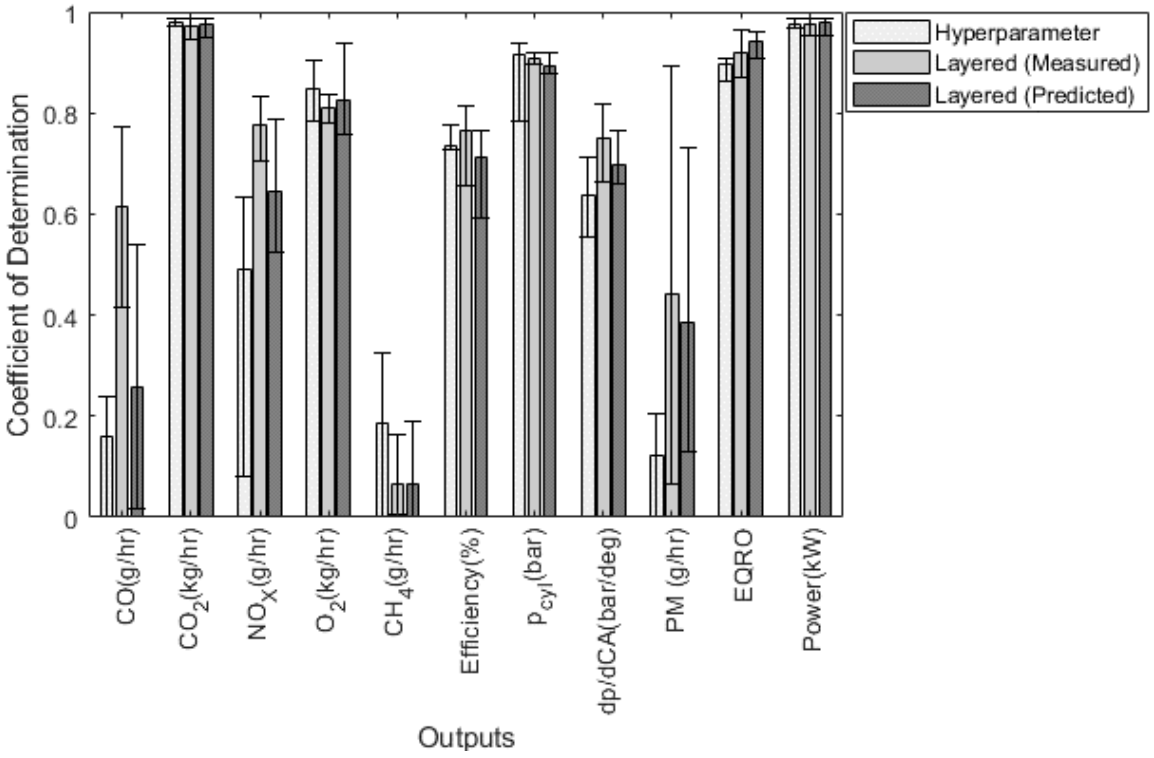


Figure 6.10: Coefficient of Determination for Layered Modeling Method (Validation Data)

Figure 6.10 shows that the CH_4 has a very low coefficient of determination, under 0.2, which is made worse in the layered model structure. While many of the results are very similar, there are three outputs where the layered model shows significant improvement. These are the particulate matter, CO, and NO_X results. In each of these cases the measured approach shows significant

improvement of approximately 0.3 while the predicted method still shows improvement but to a lesser degree.

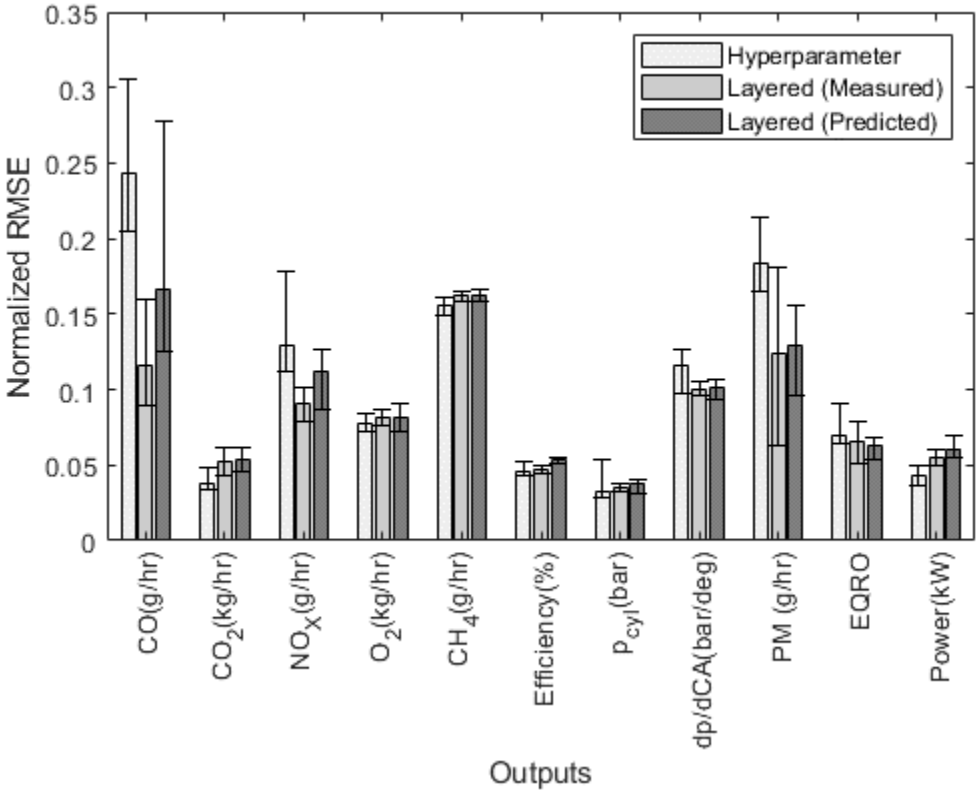


Figure 6.11: Normalized RMSE for Layered Modeling Method (Validation Data)

The results from Figure 6.11 show a similar results to the coefficient of determination. The CO nRMSE has been reduced as has the nRMSE for the particulate matter and NO_X. The model variability indicated by Figure 6.11 is lower compared to the hyperparameter optimized models for many of the outputs with error bars indicating a range of less than 0.03 in most cases. This indicates that these models are more robust and reliable in addition to having a lower nRMSE for

5 of the 11 models. Only the CO and particulate matter show higher variability which is consistent with the coefficient of determination results.

Table 6.4:Normalized RMSE results for the layered model (validation results)

Outputs	Layered Model nRMSE *		Hyperparameter	% Reduction in nRMSE *	
	Measured	Predicted	Optimized nRMSE	Measured	Predicted
CO	**0.117**	**0.166**	0.243	**52.00**	**31.46**
CO_2	0.053	0.053	0.038	-37.60	-38.76
NO_X	**0.091**	**0.112**	0.129	**29.77**	**13.25**
O_2	0.082	0.082	0.077	-5.935	-5.782
CH_4	0.162	0.163	0.155	-4.218	-4.905
Efficiency	0.047	0.053	0.046	-1.041	-15.76
p_{cyl}	0.035	0.038	0.032	-10.03	-20.41
dp/dCA	**0.100**	**0.101**	0.116	**13.81**	**12.78**
PM	**0.124**	**0.129**	0.184	**32.53**	**29.71**
EQRO	**0.065**	**0.062**	0.070	**7.298**	**11.16**
Power	0.055	0.061	0.043	-26.78	-41.41
Average	**0.085**	0.093	0.103	**4.527**	-2.606

* Lowest nRMSE in bold.

Table 6.4 shows the numerical average values from Figure 6.11. The layered models for CO, NO_X, dp/dCA, particulate matter and EQRO outperform the hyperparameter optimized models and show a lower nRMSE. As both the particulate matter and CO have previously shown some of the highest

nRMSE results, the improvement of 33% to 52% with the measured training approach is substantial and makes a strong case for using the layered modeling method. The remaining models showed an increase in nRMSE. Of those models, only the CO_2 and power show a larger increase in nRMSE of about of approximately 0.015. The measured approach showed the greatest overall average nRMSE improvement of about 4.5 % on average. This is consistent with the coefficient of determination results which showed that the measured approach often yielded the best model performance on average.

6.3.4 Layered Model Conclusions

The layered model proved beneficial for several outputs even if many of the models has a slight increase in RMSE. Particulate matter is a difficult metric to model. This is demonstrated many times in literature through the work being done to better understand soot formation and formulate both empirical and machine learning particulate matter models [70], [71], [78]. The layered model decreased the particulate matter nRMSE by approximately 32% using the measured layered model approach without requiring the addition of new data points. The hyperparameter optimized CO models also had a larger nRMSE of 0.24. The measured layered model decreased this by 52%. These two models were the biggest remaining problematic models from the original hyperparameter optimized modeling results. The model improvements provided by the layered modeling approach shows that layered modeling can be strategically implemented to improve model performance for problematic outputs if there is sufficient knowledge of the process available to identify the intermediate steps and model them accordingly.

6.4 Hybrid Model

The hybrid model is an extension of the layered model. Where the layered model approximates the interconnected processes of the engine by creating a network of machine learning models, the hybrid model takes this one step further by injecting equation based models into the layered model structure to approximate some of the intermediate parameters. For this implementation, the following parameters are replaced with mathematics based models:

1. EQR
2. EQRO
3. Thermal efficiency
4. O_2

The mathematics based models each follow relatively simple equations shown in Equations 6.2 through 6.5 [11].

$$\mathrm{EQR} = \frac{\left(\frac{m_{air}}{m_f}\right)_{st}}{\left(\frac{m_{air}}{m_f}\right)} \tag{6.2}$$

$$\mathrm{EQRO} = \frac{\left(\frac{m_{O_2}}{m_f}\right)_{st}}{O_{frac}\left(\frac{(1+\mathrm{EGR})m_{air}}{m_f}\right)} \tag{6.3}$$

$$\eta = \frac{P_{gross} \times 3600}{m_{ng} \times LHV_{ng} + m_{diesel} \times LHV_{diesel}} \tag{6.4}$$

$$O_2 = m_f\left(\frac{m_{O_2}}{m_f}\right)_{st}\left[\frac{1}{\mathrm{EQR}} - 1\right][1 - H_2O_{\%}] \tag{6.5}$$

The oxygen mass fraction (O_{frac}), lower heating values (LHV), percentage of water in the exhaust ($H_2O_{\%}$), and the stoichiometric ratios (st) are all constants. The oxygen mass (m_{O_2}) is calculated based on the percentage of oxygen in the compressed intake air, which is also constant. The exhaust gas recirculation percentage (EGR), diesel fuel mass (m_{diesel}), and intake air mass (m_{air}) are taken from new hyperparameter optimized Gaussian process regression submodels according to the new hybrid layered model structure in Figure 6.12.

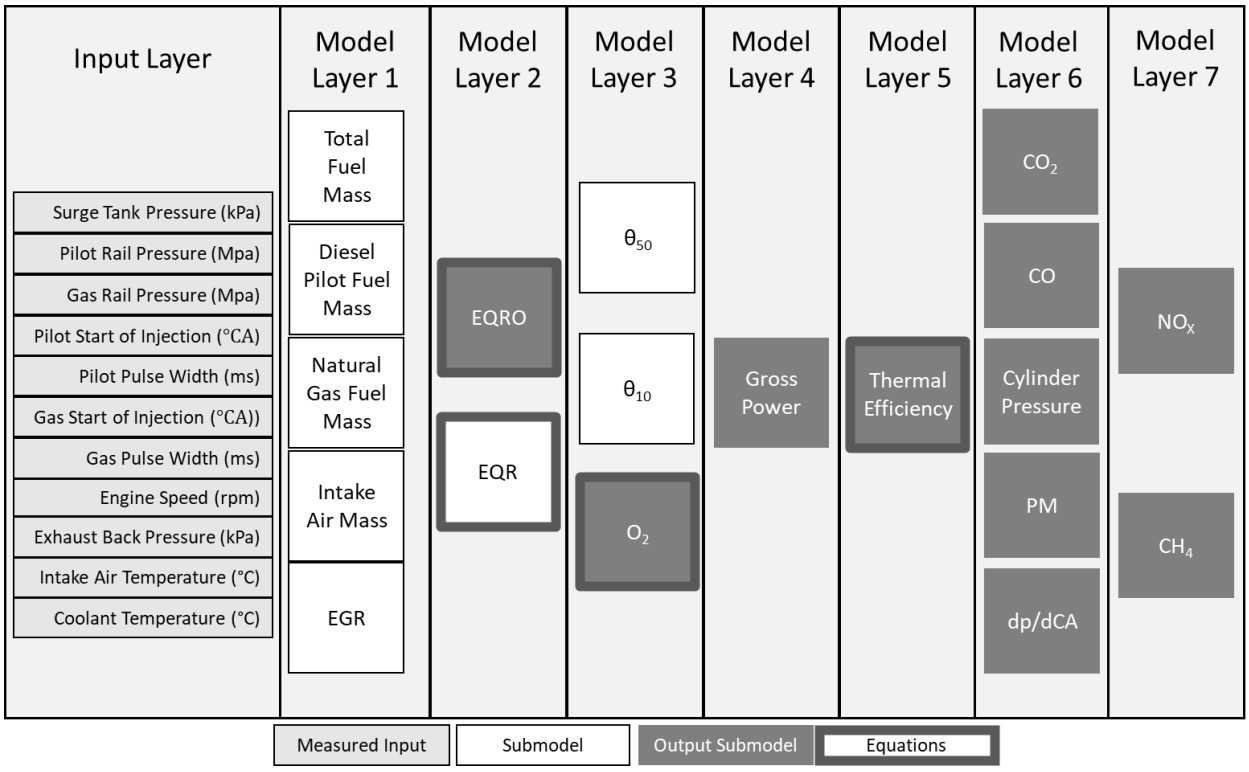


Figure 6.12: Hybrid Model Structure

The model structure has changed in Figure 6.12 compared to the layered model. In addition to replacing some models with equations, new submodels have been added to facilitate the new inputs required for the EGR, diesel fuel mass, and intake air mass. The thermal efficiency has also been

moved to its own layer as it is now dependent on the gross indicated power. The new input matrix is given in Appendix D.2.

6.4.1 Error Correcting Hybrid Model

While Equations 6.2 through 6.5 are well known in literature and are based on the system diagram in Figure 2.2, they are also based on idealized system processes [11]. These equations do directly account for measurement error and system nonlinearities. For this reason, an error correcting method is developed that takes the output of each equation and applies a correcting GPR model to account for system variations. Figure 6.13 shows how this error correcting model is integrated.

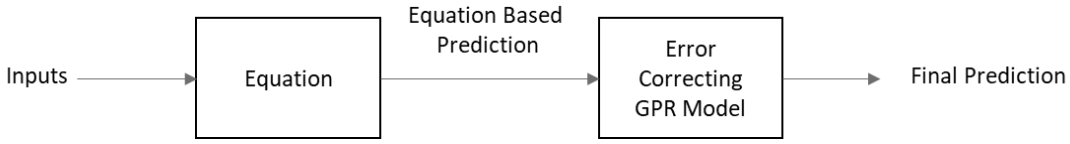


Figure 6.13: Error Correcting Hybrid Model Example

The following sections detail the results of the hybrid modelling approach to reducing model error for the Box Behnken data set. Performance metrics are first shown for the new Gaussian process regression based submodels, followed by the full hybrid models. Results for both the hybrid and error correcting hybrid models are presented. These are compared to the measured training approach from the layered models in Section 6.3 and the Gaussian process regression models with hyperparameter optimization from Chapter 5. All models are generated 10 times with the aggregated results presented, showing the mean and range of the results.

6.4.2 New Hybrid Model Submodel Results

Figure 6.14 shows both the coefficient of determination and normalized root mean square error for the new submodels introduced in the hybrid modeling method. These submodels are used in conjunction with the submodels from the layered model structure in Section 6.3.2.

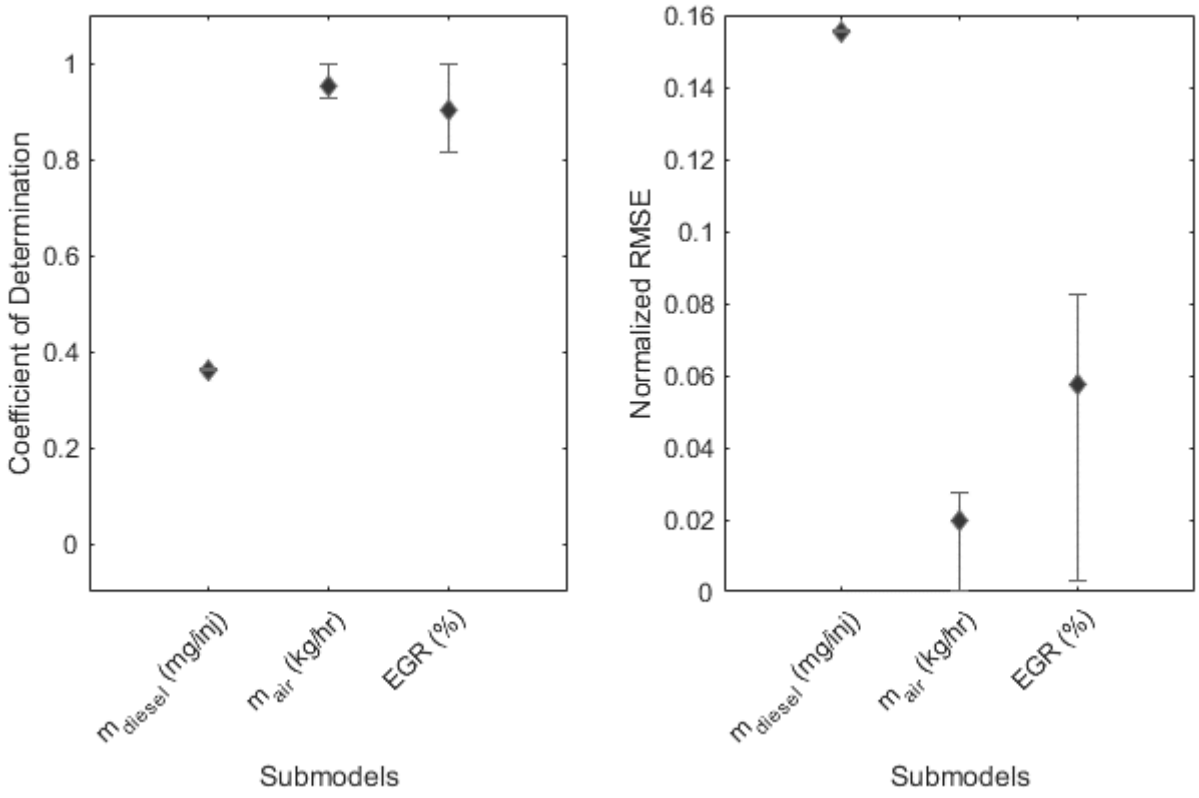


Figure 6.14: New Hybrid Submodel Coefficient of Determination and Normalized RMSE (Training Data)

Figure 6.14 shows that the diesel fuel mass results have a low coefficient of determination, below 0.4, and a high normalized RMSE of 0.16. This is expected as the diesel fuel flow rate is difficult to measure accurately. The diesel flow rate is measured by monitoring a mass scale that weighs the diesel fuel container. The per injection fuel mass is calculated by measuring the change in the

mass over the 2 minute duration of sampling a given operating point, and then dividing it by the number of combustion cycles. This is problematic as the mass scale is highly inaccurate due to pressure and flow rate fluctuations. The mass of diesel used in each injection is significantly smaller than the natural gas, and the noise created by these minor fluctuations make up a large portion of the signal. This is the main reason that this input is not used in the layered model structure from Figure 6.7. Fortunately, the volume of diesel fuel used is low enough that noise in these measurements should not impact the calculations used in the hybrid model. In every case where this value is used, the natural gas measurement is the dominant term. The diesel fuel still needs to be considered but the errors introduced by the diesel fuel mass should not heavily impact any of the equation based models.

The intake air mass performs well with a coefficient of determination close to 1 and a normalized RMSE that is consistently less than 0.03. The EGR measures the percentage of the intake air that is made up of recirculated exhaust gas. EGR is known to be difficult to measure directly and is highly dependent on the complex interactions of the intake air pressure, the exhaust back pressure and the engine speed [30], [34], [56]. It is for this reason that this submodel is not used in the layered model in Figure 6.7. For the hybrid model, this parameter is required for some of the equation based models. While the coefficient of determination shows results above 0.8, this is lower than expected for the training data, which would ideally be consistently above 0.9. The range of the nRMSE for the training data is also larger than desired, spanning a value of 0.08. While neither of these results are especially poor, they do introduce some concern and it is likely that this model may introduce some error that could negatively impact the final output models.

6.4.3 Hybrid Model Validation Data Results

This section presents the validation data results for the hybrid models. The model performance presented here is based on the Box Behnken randomized validation data set and is compared to the hyperparameter optimized results from Chapter 5 as well as the layered model trained using the measured approach from Section 6.3.

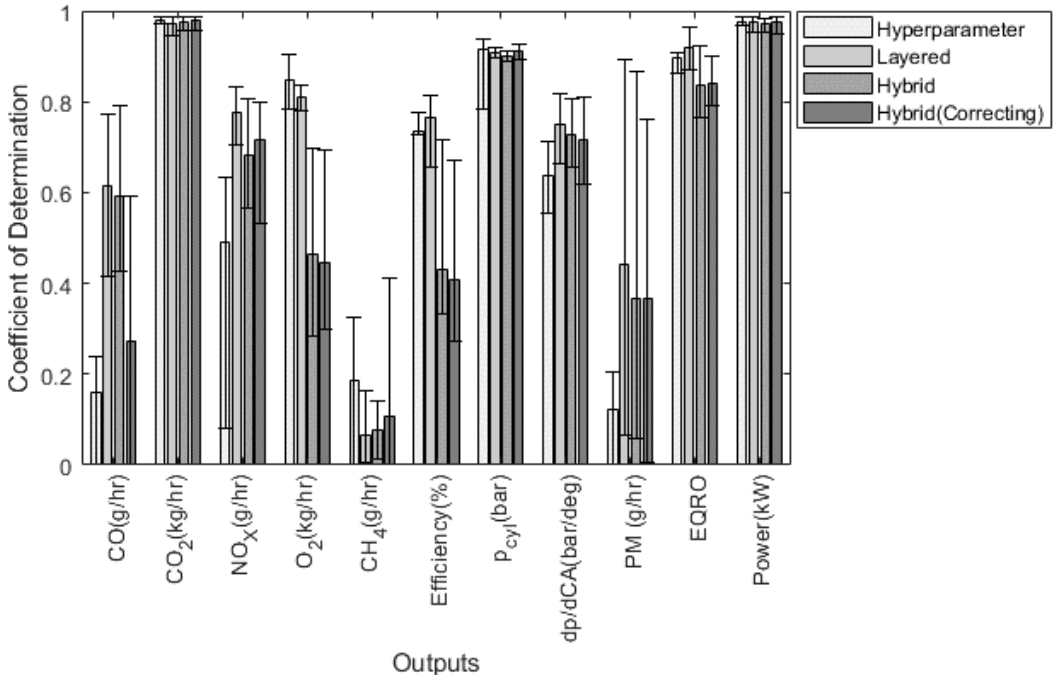


Figure 6.15: Coefficient of Determination for Hybrid Modeling Method (Validation Data)

Figure 6.15 shows that the coefficient of determination results for the hybrid models are lower on average compared to the other models. While the coefficients of determination for the hybrid models is higher than the hyperparameter optimized models for CO, NO_X, the cylinder pressure gradient, and particulate matter, they are still consistently lower than the layered model.

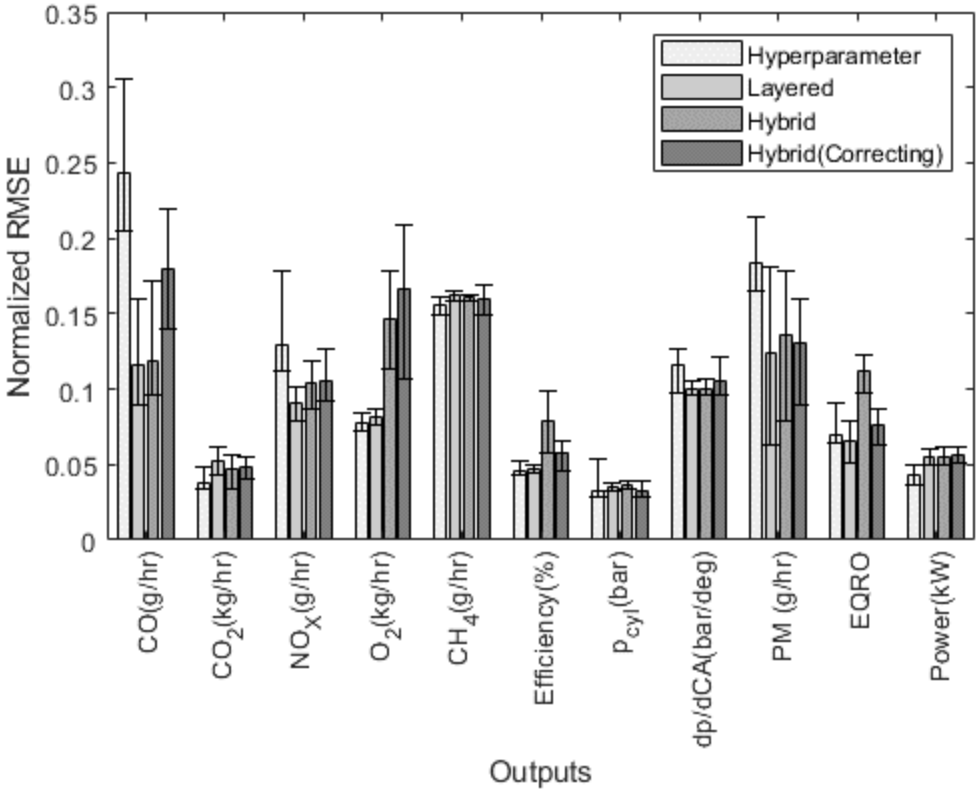


Figure 6.16: Normalized RMSE for Hybrid Modeling Method (Validation Data)

Figure 6.16 shows the mean normalized root mean square error. The hyperparameter optimized models show the lowest nRMSE on average, although, the hybrid and layered models show lower nRMSE results for the CO, cylinder pressure gradient, particulate matter, and NO_X. The range of the nRMSE values also indicates that these differences should be consistent across most versions of the models as there is very little crossover in the error bars between the hyperparameter optimized models and the other methods.

Table 6.5: Normalized RMSE results for the hybrid model (validation data)

Outputs	Hybrid Model	Comparison		% Reduction in nRMSE *	
		Hyperparameter	Layered	Hyperparameter	Layered
CO	0.118	0.243	0.117	**51.32**	-1.435
CO_2	0.046	0.038	0.053	-20.85	**12.17**
NO_X	0.104	0.129	0.091	**19.85**	-14.12
O_2	0.146	0.077	0.082	-89.59	-78.97
CH_4	0.161	0.155	0.162	-3.898	**0.307**
Efficiency	0.079	0.046	0.047	-70.47	-68.72
p_{cyl}	0.037	0.032	0.035	-14.89	-4.415
dp/dCA	0.100	0.116	0.100	**13.52**	-0.327
PM	0.137	0.184	0.124	**25.72**	-10.10
EQRO	0.112	0.070	0.065	-59.88	-72.47
Power	0.056	0.043	0.055	-28.77	-1.569
Average	0.100	0.103	0.085	-16.18	-21.79

* Lowest nRMSE in bold.

Table 6.5 shows the non-error correcting hybrid model nRMSE compared to the layered models and the hyperparameter optimized models. Table 6.5 demonstrates that on average, the non-error correcting hybrid model introduces more error into the modeling method with an average increase of 16% compared to the original hyperparameter models and 22% compared to the layered models. None of the models show consistent improvement across both the layered and hyperparameter optimized models. In every case where the hybrid model performs better than the hyperparameter

optimized model, the layered model outperforms the hybrid model. The performance differences between the hybrid and layered models for the CO and dp/dCA, are within 1.5%, however, this is not enough to justify using one model over the other.

Table 6.6: Normalized RMSE results for the error correcting hybrid model (validation data)

Outputs	Error Corrected Hybrid Model	nRMSE			% Reduction in nRMSE *		
		Hyper-parameter	Layered	Hybrid	Hyper-parameter	Layered	Hybrid
CO	0.179	0.243	0.117	0.118	**26.080**	-54.011	-51.832
CO_2	0.048	0.038	0.053	0.046	-24.748	**9.341**	-3.222
NO_X	0.106	0.129	0.091	0.104	**18.011**	-16.739	-2.297
O_2	0.167	0.077	0.082	0.146	-116.166	-104.056	-14.016
CH_4	0.160	0.155	0.162	0.161	-3.257	**0.922**	**0.617**
Efficiency	0.058	0.046	0.047	0.079	-26.039	-24.740	**26.065**
p_{cyl}	0.033	0.032	0.035	0.037	-3.177	**6.230**	**10.195**
dp/dCA	0.105	0.116	0.100	0.100	**8.916**	-5.672	-5.328
PM	0.131	0.184	0.124	0.137	**28.767**	-5.584	**4.106**
EQRO	0.076	0.070	0.065	0.112	-8.804	-17.370	**31.947**
Power	0.057	0.043	0.055	0.056	-31.956	-4.084	-2.475
Average	0.102	0.103	0.085	0.100	-12.03	-19.62	-0.567

* Lowest nRMSE in bold.

Table 6.6 shows the nRMSE of the error correcting hybrid modeling method compared to the layered modeling method, hyperparameter optimized modeling method and the non-error correcting hybrid modeling method. The error correcting hybrid models have lower nRMSE than the hybrid models for CH_4, peak cylinder pressure, thermal efficiency, and EQRO. However, the error correcting hybrid model is still outperformed overall by most other models. Compared to the hyperparameter optimized models alone, the error correcting hybrid models show a decrease in the nRMSE of 26% for CO, 18% for NO_X, 9% for the cylinder pressure gradient and 29% for the particulate matter. These are all notable improvements over the original models. The layered models, however, consistently show lower nRMSE values for each of these models. For these outputs, the error correcting hybrid model shows between 5.6% to 54% higher nRMSE compared to the layered models. This indicates that the layered modeling method is the better method for these outputs.

6.4.4 Hybrid Model Conclusions

Both the hybrid and error correcting hybrid models have been compared to the layered models and the original hyperparameter optimization based models. Based on the results, the hybrid models overall underperform compared to both the layered models and the hyperparameter optimization models. There are several cases where the nRMSE is close to the layered models and lower than the hyperparameter optimization based models. This is mostly the case for the non-error correcting hybrid model. In each of these cases the layered model shows better performance with an overall lower complexity. This slightly higher error most likely comes as a result of the introduction of the new submodels combined with the inability of the equations to correct for variations in the engine. The error correcting hybrid model attempts to address this issue, however, this approach

still needs additional refinement. It is possible that a different implementation of the error correcting hybrid model would be able to address this issue. While the hybrid method does not outperform the other methods shown in this work, a different implementation, using different submodels, better sensors or a different approach to error correction could allow this method to further reduce the modeling error.

6.5 Summary of Model Improvement Techniques

The work shown in Sections 6.1 through 6.4 demonstrates several methods for improving model performance for hyperparameter optimized machine learning models with small data sets. The foundation of this work is the hyperparameter optimization technique presented in Chapter 5 which also demonstrates that Gaussian process regression presents the best modeling method for modeling complex engine systems with reduced data sets [1]. These new techniques are therefore built on top of the Gaussian process regression modeling method with hyperparameter optimization. The techniques that were evaluated are, multi-region modeling, a sensitivity analysis input reduction and selection method, a layered modeling method, a hybrid mathematics and machine learning method, and an error correcting version of the hybrid method. Table 6.7 summarizes the overall results of all the modeling methods with the multi-region modeling results summarized in Table 6.8.

Table 6.7: Model improvement technique summary (validation data)

Outputs	Modeling Method nRMSE *					Best	% nRMSE decrease	Best Method
	Hyper-parameter	Sensitivity	Layered	Hybrid	Hybrid EC			
CO	0.243	**0.197**	**0.117**	**0.118**	**0.179**	**0.117**	**52.00**	Layered
CO_2	0.038	**0.037**	0.053	0.046	0.048	**0.037**	**4.444**	Sensitivity
NO_X	0.129	**0.117**	**0.091**	**0.104**	**0.106**	**0.091**	**29.77**	Layered
O_2	0.077	**0.077**	0.082	0.146	0.167	**N/A**	**40.24**	Multi-Region
CH_4	0.155	**0.154**	0.162	0.161	0.160	**0.154**	**1.071**	Sensitivity
Efficiency	0.046	**0.041**	0.047	0.079	0.058	**N/A**	**15.63**	Multi-Region
p_{cyl}	0.032	**0.031**	0.035	0.037	0.033	**0.031**	**3.731**	Sensitivity
dp/dCA	0.116	**0.098**	**0.100**	**0.100**	**0.105**	**0.098**	**15.01**	Sensitivity
PM	0.184	**0.166**	**0.124**	**0.137**	**0.131**	**0.124**	**32.53**	Layered
EQRO	0.070	**0.065**	**0.065**	0.112	0.076	**0.065**	**7.416**	Sensitivity
Power	0.043	**0.040**	0.055	0.056	0.057	**0.040**	**6.908**	Sensitivity
Average	0.103	**0.093**	**0.085**	**0.100**	**0.102**		**18.98**	Sensitivity

* Lowest nRMSE in bold.

Table 6.8: Model improvements multi-region summary (validation data)

Outputs	Multi-Region nRMSE *			Combined nRMSE	% Reduction in nRMSE *		
	+ve PSEP	-ve PSEP	Average		+ve PSEP	-ve PSEP	Average
CO	**0.243**	**0.216**	**0.230**	0.311	**21.86**	**30.41**	**26.21**
O_2	**0.077**	**0.07**	**0.074**	0.123	**37.15**	**43.43**	**40.24**
Efficiency	**0.046**	**0.035**	**0.041**	0.048	**2.995**	**26.69**	**15.63**

* Lowest nRMSE in bold.

From Table 6.7 and Table 6.8, the nRMSE for every model output is reduced using some combination of these techniques. This demonstrates that using engineering intuition and advanced modeling techniques, machine learning modeling methods can be improved upon. In particular, the hyperparameter optimization method has already demonstrated an overall model improvement. Adding these new techniques has further improved the model performance beyond what was possible using only hyperparameter optimization. The sensitivity analysis results show that reducing the input vector size and removing the less relevant inputs improves the model performance by 8% on average. This is the overall most consistently successful method. By removing inputs that are not being used, the model input noise is reduced during the training process which in turn improves the model performance.

Multi-region modeling improves the oxygen and thermal efficiency models by 15% and 40% respectively. While the input sensitivity improves these models, the multi-region approach shows the highest decrease in nRMSE. Unfortunately, this method is highly dependent on the regions that the operating space can be divided into. While this is a limitation, anytime a large operating space

is being studied, appropriate boundaries should be possible to identify in order to characterize distinctly different operating regions.

The layered model showed the second lowest nRMSE after the sensitivity based input reduction. Here, the particulate matter nRMSE improved by 33% and the CO improved 52%. As particulate matter is one of the most difficult to measure engine emissions, this is a very significant improvement [70], [71], [78]. A 33% decrease in model error demonstrates that this technique can overcome both the limitations of a small data set and the complexity of particulate matter modeling.

The hybrid model improved the nRMSE for the CO by 51% and the particulate matter by 26%. These are also significant improvements but are still outperformed by the layered model. The introduction of additional, difficult to model submodels that were required to facilitate the hybrid model structure likely introduced too much error for the benefits of the hybrid model to improve on the layered technique. Additional refinements to this method or a slightly different implementation may improve these results.

The summarized nRMSE results in Table 6.7 and Table 6.8 show the benefit of introducing engineering knowledge and process related information into machine learning models. This additional information acts as new data that can overcome the limitations of small or otherwise insufficient data sets and reduce the difficulty of the modeling process. Reducing the input vector size provided the most consistent benefit while using the layered models improved the performance in those cases where the relationships between the inputs and outputs were difficult

for the machine learning model to determine. The multi-region method also provided a noticeable benefit for some outputs but can only be used where an identifiable boundary between two different operating spaces can be defined.

Overall, these methods provided a significant benefit over the hyperparameter optimization method which already presented a significant benefit over typical modeling methods. With these techniques, future work can expand upon these methods and utilize them for complex calibration work such as engine model transfer learning and optimal operation point identification.

Chapter 7: Conclusion

The objective of this work was to develop a steady state engine modeling method that reduces the burden of data collection required for control and calibration of new engine technologies. A heavy duty pilot ignited direct injection natural gas compression ignition engine was rebuilt and retrofitted with a new control and data acquisition system to act as a case study and test bench for this work. Typical performance and emissions metrics were selected as model outputs. The main control parameters of the engine were selected as model inputs. These inputs represent the typical control parameters used by a production engine controller. A full factorial data set was selected to represent typical engine modeling data. Two low density data sets were collected based on the Box-Behnken design of experiments method with two validation data sets also being collected in the same operating space.

Modeling hyperparameters were identified for the neural network and Gaussian process regression modeling methods and a survey of 500 different hyperparameter configurations was performed for the full factorial and Box Behnken data sets using both methods. This was compared to 10 sets of similar models with hyperparameters that were selected using Bayesian optimization. Multiple iterations of the hyperparameter optimization models were used to ensure the robustness of the method. Hyperparameter optimization proved to be effective in guaranteeing model robustness and reducing the normalized root mean square error. Between the two machine learning methods, Gaussian process regression showed the lowest error and variability.

Hyperparameter optimized Gaussian process regression models were used as the basis for four different proposed model improvement methodologies. The methodologies included multi-region

modeling, sensitivity based input reduction, layered modeling, and hybrid modeling. Each method reduced the model error beyond what was achievable with hyperparameter optimization alone, further improving the modeling results beyond the original hyperparameter modeling survey from Chapter 4. These results are summarized in Table 7.1.

Table 7.1: Summary of model improvement compared to hyperparameter optimization from Chapter 5 and the hyperparameter modeling survey from Chapter 4

Outputs	Best Model Improvement Method	% Reduction in nRMSE for the Best Methods	
		Hyperparameter Optimization	Hyperparameter Survey
CO	Layered	**52.0**	**45.0**
CO_2	Sensitivity	**4.44**	**67.3**
NO_X	Layered	**29.8**	**50.5**
O_2	Multi-Region	**40.2**	**72.1**
CH_4	Sensitivity	**1.07**	**1.71**
Efficiency	Multi-Region	**15.6**	**26.1**
p_{cyl}	Sensitivity	**3.73**	**48.7**
dp/dCA	Sensitivity	**15.0**	**32.9**
PM	Layered	**32.5**	**29.9**
EQR	Sensitivity	**7.42**	**61.4**
Power	Sensitivity	**6.91**	**63.9**
Average		**19.0**	**45.4**

Sensitivity based input reduction showed improvements in the normalized root mean square error across all models, with an average decrease in error of 8%. Layered modeling improved the CO by 52%, NO_X by 30% and particulate matter by 33%, which surpassed the sensitivity based method for these outputs. Multi-region modeling improved the O_2 by 40% and thermal efficiency by 16%, which also surpassed the sensitivity based approach. Hybrid models also showed improvements but were surpassed by the layered models. By strategically using the different techniques, an overall reduction in error of 19% was achieved over the hyperparameter optimized models with a 45% error reduction over the initial model survey.

Testing of each component was done during the set up of the test bench and a repeatability study was done to assess the variability of the engine. While training data performance is presented in the appendix, the results focused on validation data. This data represents a more realistic evaluation of the models as this data showed the expected performance for any arbitrary data point that has not had the benefit of being part of the training data set. All data was collected in the same time frame to reduce the effects of system variation. This was verified with the repeatability study. The model results demonstrated an error that approached the system variability, of approximately 0.1, showing that the models are as accurate as can be expected. This work uses a full factorial data set which is similar in structure to the data sets used in other publications [51], [61]. This full-factorial data demonstrated a decrease in the normalized RMSE with hyperparameter optimization which carried over to the Box Behnken data set. This shows that the performance gained through the techniques presented here were consistent and not dependent on the data set. Each result presented in this research is calculated multiple times to ensure robustness. The results demonstrated that the

models are robust, with little variation in most cases and that the normalized root mean square error reduces overall as the techniques are applied.

The various methods presented in this work were successful in reducing the modeling error for the small data set. While larger data sets are still beneficial, the sensitivity based input reduction, layered modeling, multi-region modeling, and potentially hybrid modeling methods demonstrate a potential approach to overcoming the data burden limitation and rapidly producing engine models with smaller data sets. In either case, hyperparameter optimization consistently reduced the modeling error regardless of the size of the data set.

7.1 Main Contributions

Prior to the initiation of this work, Gaussian process regression only had preliminary implementation for engine related studies [45], [49], [50]. Even now, few papers exist, relative to other more common techniques such as neural networks. The results shown in Chapter 4 demonstrated that Gaussian process regression is well suited for engine research as it has produced engine models showing 45% to 66% of the error shown by the artificial neural network models.

A method for selecting ideal hyperparameters for complex systems through hyperparameter optimization was presented and it was further demonstrated that this approach produced a low variability, low error model performance. Most engine modeling work using neural networks or Gaussian process regression, selects model hyperparameters according the most commonly used methods [41], [50]. While these hyperparameters create usable models, the hyperparameter optimization demonstrated that other better hyperparameter configurations may be found through

optimization. Techniques such as the Matérn based kernels were shown to be better than the more commonly used squared exponential and BFGS Quasi-Newton was shown to produce effective neural network engine models despite not being commonly used in engine research. Similarly, neural network node counts are often selected based on the number of inputs and outputs. Through hyperparameter optimization, it found that identifying optimal node configurations may present lower error results.

Engine research is often done on conventional gasoline or diesel engines, focusing on production systems or a modifications to the fuel [38], [40]–[43]. The methods presented here demonstrated a means of modeling not just a complex and unusual system, but one with many degrees of freedom. This is something that the production systems do not typically have, as they are heavily constrained by an existing engine control unit and the built in calibration for the engine. Using the unusual PIDING engine proved that models can be made for complex engine systems with smaller data sets by using these techniques.

Many engine modeling papers discuss the performance of models for engines using conventional machine learning methods [42], [43]. This work expanded upon existing machine learning methods by factoring engineering knowledge into the model development. One criticism of machine learning is that the inner workings of the model are hidden behind the algorithm, obscuring it from the researcher's view. The layered and hybrid modeling methods changed this by introducing the physical process into the model structure, creating a machine learning model that contains engineering knowledge for the system in addition to the purely data driven machine learning elements. The major limitation of this approach is that an understanding of the physics

and system properties is required to create these types of models. While this adds an additional knowledge burden to the model developer, it provides a means of introducing additional information in the model without collecting additional experimental data points.

This work presents a means of modeling new engine technologies without the normal cost of extensive data collection. While new data is still required for each new engine technology, these methods can be used in either industry or academia to determine an initial performance expectation for a given engine technology and then provide a starting point for calibration that takes a fraction of the time that typical methods would require. This may reduce the cost of new engine technology development and potentially allows future researchers to explore new engine technologies in a more efficient manner.

7.2 Future Work

The different model improvement techniques can be applied in numerous ways to benefit engine research. Each new engine technology would require a new data set and may require a new model structure with new inputs and outputs that reflect the new system. Despite this, the core methodology presented in this work may still be applied to new technologies once the model development procedure is properly adapted. This can be done using engineering intuition and the sensitivity analysis method discussed in section 6.2. In those cases where data size is a concern, these methods may provide a means of reducing the model error, which can benefit various avenues of research.

Using these models, future implementations of this work may be able to rapidly generate calibrations for new engine technologies. This could be done using an optimization algorithm, such as particle swarm optimization, to rapidly select operating points. These rapidly developing calibrations can be modified in such a way that could allow for self improving real-time modeling and engine calibration.

Other implementations of the model improvement techniques such as the layered model could focus on integrating transfer learning to utilize even smaller data sets from varying sources. Transfer learning enables the use of model data from one research engine on a different engine [55]. Gaussian process regression has only recently been adapted for transfer learning with several implementations having been developed [29], [79]–[81]. The implementation of transfer learning in addition to the above techniques could allow for rapid development of new engine technologies with vastly smaller data sets than was previously required.

The techniques presented here could also be further developed to include a more advanced integration of engineering knowledge in the model structure. More complex equations and process models could be used to replace submodels in the hybrid model structure and different methods of error correction could be applied to the hybrid model to enhance the hybrid model performance. More advanced implementations of the layered model structure could also be developed with more complex inputs being predicted. Exhaust gas recirculation, for example, has been experimentally observed to have a close relation to emissions formation for this type of engine [30], [82]. Creating more complex models that factor in these parameters combined with different techniques such as a feedback approach to iterative model prediction could enhance the layered model performance.

Data is an important element of machine learning engine research. With the complexity of engines increasing, researchers are increasing their reliance on machine learning modeling methods, always aware of the data collection burden that using such methods entails. When developing new engine technologies, it is recommended that researchers both in academia and industry consider employing the hyperparameter optimized Gaussian process regression modeling method with multi-region modeling, layered modeling, and sensitivity based input reduction to potentially expediate the modeling process and possibly reduce the time and cost of development for these new technologies. With the techniques presented here, based on hyperparameter optimization, and extended with data sciences and engineering process knowledge, researchers may be able to shift their focus away from data collection and more towards the research itself.

References

[1] A. G. Mamalis, K. N. Spentzas, and A. A. Mamali, “The impact of automotive industry and its supply chain to climate change: Somme techno-economic aspects,” *Eur. Transp. Res. Rev.*, vol. 5, no. 1, pp. 1–10, Mar. 2013, doi: 10.1007/s12544-013-0089-x.

[2] G. Walther, J. Wansart, K. Kieckhäfer, E. Schnieder, and T. S. Spengler, “Impact assessment in the automotive industry: mandatory market introduction of alternative powertrain technologies,” *Syst. Dyn. Rev.*, vol. 26, no. 3, pp. 239–261, Jul. 2010, doi: 10.1002/sdr.453.

[3] “Understanding Global Warming Potentials,” *United States Environmental Protection Agency*. https://www.epa.gov/ghgemissions/understanding-global-warming-potentials (accessed Aug. 06, 2021).

[4] A. Supee, M. S. Shafeez, R. Mohsin, and Z. A. Majid, “Performance of Diesel-Compressed Natural Gas (CNG) Dual Fuel (DDF) Engine via CNG-Air Venturi Mixjector Application,” *Arab. J. Sci. Eng.*, vol. 39, no. 10, pp. 7335–7344, 2014, doi: 10.1007/s13369-014-1313-2.

[5] K. Bhandari, A. Bansal, A. Shukla, and M. Khare, “Performance and emissions of natural gas fueled internal combustion engine: A review,” *J. Sci. Ind. Res. (India).*, vol. 64, no. 5, pp. 333–338, 2005.

[6] Semin and R. A. Bakar, “A Technical Review of Compressed Natural Gas as an Alternative Fuel for Internal Combustion Engines,” *Am. J. Eng. Appl. Sci.*, vol. 1, no. 4, pp. 302–311, Apr. 2008, doi: 10.3844/ajeassp.2008.302.311.

[7] J. Rochussen, G. McTaggart-Cowan, and P. Kirchen, “Parametric study of pilot-ignited direct-injection natural gas combustion in an optically accessible heavy-duty engine,” *Int.*

J. Engine Res., vol. 21, no. 3, pp. 497–513, Mar. 2020, doi: 10.1177/1468087419836877.

[8] M. Karpinski-Leydier, *Optimal Control of Li-Ion Hydrogen Fuel Cell Hybrid Vehicles*. Master Thesis, Waterloo: University of Waterloo, 2012.

[9] P. Corbo, F. Migliardini, and O. Veneri, *Hydrogen Fuel Cells for Road Vehicles*, vol. 11. London: Springer London, 2011.

[10] P. Kranabitl, M. Bajzek, M. Atzwanger, D. Schenk, and H. Hick, "Automotive Powertrain Development Process," in *Systems Engineering for Automotive Powertrain Development*, Springer, Cham, 2020.

[11] L. Guzzella and C. H. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[12] M. Nazemi and M. Shahbakhti, "Modeling and analysis of fuel injection parameters for combustion and performance of an RCCI engine," *Appl. Energy*, vol. 165, pp. 135–150, Mar. 2016, doi: 10.1016/j.apenergy.2015.11.093.

[13] Firmansyah, A. R. A. Aziz, and M. R. Heikal, "Double Stage Wiebe: An Approach to Single Zone Modeling of Dual Fuel HCCI Combustion," *Asian J. Sci. Res.*, vol. 6, no. 2, pp. 388–394, Mar. 2013, doi: 10.3923/ajsr.2013.388.394.

[14] G. H. Abd Alla, H. A. Soliman, O. A. Badr, and M. F. Abd Rabbo, "Combustion quasi-two zone predictive model for dual fuel engines," *Energy Convers. Manag.*, vol. 42, no. 12, pp. 1477–1498, Aug. 2001, doi: 10.1016/S0196-8904(00)00143-6.

[15] F. Zurbriggen, R. Hutter, and C. Onder, "Diesel-Minimal Combustion Control of a Natural Gas-Diesel Engine," *Energies*, vol. 9, no. 1, p. 58, Jan. 2016, doi: 10.3390/en9010058.

[16] E. Song, X. Shi, C. Yao, and Y. Li, "Research on real-time simulation modelling of a

diesel engine based on fuel inter-zone transfer and an array calculation method," *Energy Convers. Manag.*, vol. 178, pp. 1–12, Dec. 2018, doi: 10.1016/j.enconman.2018.10.014.

[17] T. M. Dvorak and R. L. Hoekstra, "Optimizing Internal Combustion Engine Performance Through Response Surface Methodology," *SAE Tech. Pap.*, Dec. 1996, doi: 10.4271/962525.

[18] N. N. Li, S. H. Li, and X. X. Li, "Simulation Research of Natural Gas Injection Timing on Combustion Process in a Diesel Engine Fueled with Pilot-Ignited Directly-Injected Natural Gas," *Appl. Mech. Mater.*, vol. 192, pp. 132–138, Jul. 2012, doi: 10.4028/www.scientific.net/AMM.192.132.

[19] C.-W. Lee, H.-W. Ge, R. D. Reitz, E. Kurtz, and W. Willems, "Computational Optimization of a Down-Scaled Diesel Engine Operating in the Conventional Diffusion Combustion Regime Using a Multi-Objective Genetic Algorithm," *Combust. Sci. Technol.*, vol. 184, no. 1, pp. 78–96, Jan. 2012, doi: 10.1080/00102202.2011.620051.

[20] M. Ezzeddinne, E. Castro, and R. Lengellé, "Dynamic Design of Experiments for Engine Pollutants Emissions Modeling and Optimization," *SAE Tech. Pap.*, no. 724, Oct. 2008, doi: 10.4271/2008-01-2454.

[21] P. P. Balestrassi, E. Popova, A. P. Paiva, and J. W. Marangon Lima, "Design of experiments on neural network's training for nonlinear time series forecasting," *Neurocomputing*, vol. 72, no. 4–6, pp. 1160–1178, Jan. 2009, doi: 10.1016/j.neucom.2008.02.002.

[22] W. S. Sarle, "Neural Networks and Statistical Models," in *Proceedings of the Nineteenth Annual SAS Users Group International Conference*, 1994, pp. 1538--1550.

[23] D. Ellis and N. Morgan, "Size matters: an empirical study of neural network training for

large vocabulary continuous speech recognition," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings.*, 1999, vol. 2, pp. 1013–1016, doi: 10.1109/ICASSP.1999.759875.

[24] C. Kavuri and S. L. Kokjohn, "Exploring the potential of machine learning in reducing the computational time/expense and improving the reliability of engine optimization studies," *Int. J. Engine Res.*, vol. 21, no. 7, pp. 1251–1270, Sep. 2020, doi: 10.1177/1468087418808949.

[25] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic Modeling Using Deep Belief Networks," *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 20, no. 1, pp. 14–22, Jan. 2012, doi: 10.1109/TASL.2011.2109382.

[26] R. Asheghi, S. A. Hosseini, M. Saneie, and A. A. Shahri, "Updating the neural network sediment load models using different sensitivity analysis methods: a regional application," *J. Hydroinformatics*, vol. 22, no. 3, pp. 562–577, May 2020, doi: 10.2166/hydro.2020.098.

[27] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Syst. Appl.*, vol. 38, no. 8, pp. 10389–10397, Aug. 2011, doi: 10.1016/j.eswa.2011.02.068.

[28] D. P. Solomatine and K. N. Dulal, "Model trees as an alternative to neural networks in rainfall—runoff modelling," *Hydrol. Sci. J.*, vol. 48, no. 3, pp. 399–411, Jun. 2003, doi: 10.1623/hysj.48.3.399.45291.

[29] A. A. Chehade and A. A. Hussein, "A Collaborative Gaussian Process Regression Model for Transfer Learning of Capacity Trends Between Li-Ion Battery Cells," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 9542–9552, Sep. 2020, doi: 10.1109/TVT.2020.3000970.

[30] A. P. Singh, "Characterization and System Level Study of Air Addition in a Pilot Ignited

Direct Injection Natural Gas Engine," University of British Columbia, Vancouver, 2019.

[31] Y. Karagöz, T. Sandalcı, U. O. Koylu, A. S. Dalkılıç, and S. Wongwises, "Effect of the use of natural gas–diesel fuel mixture on performance, emissions, and combustion characteristics of a compression ignition engine," *Adv. Mech. Eng.*, vol. 8, no. 4, p. 168781401664322, Apr. 2016, doi: 10.1177/1687814016643228.

[32] Y. Zhang and C. Ling, "A strategy to apply machine learning to small datasets in materials science," *npj Comput. Mater.*, vol. 4, no. 1, p. 25, Dec. 2018, doi: 10.1038/s41524-018-0081-z.

[33] Hu *et al.*, "Comparison of Physics-Based, Semi-empirical and Neural Network-based Models for Model-based Combustion Control in a 3.0 L Diesel Engine," *Energies*, vol. 12, no. 18, p. 3423, Sep. 2019, doi: 10.3390/en12183423.

[34] M. Karpinski-Leydier, R. Nagamune, and P. Kirchen, "A Machine Learning Modeling Approach for High Pressure Direct Injection Dual Fuel Compressed Natural Gas Engines," *SAE Tech. Pap.*, Sep. 2020, doi: 10.4271/2020-01-2017.

[35] V. Aran and M. Unel, "Feedforward mapping for engine control," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, Oct. 2016, pp. 154–159, doi: 10.1109/IECON.2016.7793288.

[36] H. Wang, B. Raj, and E. P. Xing, "On the Origin of Deep Learning," *CoRR*, vol. abs/1702.0, Feb. 2017, Accessed: May 05, 2021. [Online]. Available: http://arxiv.org/abs/1702.07800.

[37] N. Yadav, A. Yadav, and M. Kumar, *An Introduction to Neural Network Methods for Differential Equations*. Dordrecht: Springer Netherlands, 2015.

[38] S. M. Shahid, S. Ko, and S. Kwon, "Real-time classification of diesel marine engine loads

using machine learning," *Sensors*, vol. 19, no. 14, p. 3172, Jul. 2019, doi: 10.3390/s19143172.

[39] S. Lotfan, R. A. Ghiasi, M. Fallah, and M. H. Sadeghi, "ANN-based modeling and reducing dual-fuel engine's challenging emissions by multi-objective evolutionary algorithm NSGA-II," *Appl. Energy*, vol. 175, pp. 91–99, Aug. 2016, doi: 10.1016/j.apenergy.2016.04.099.

[40] G. Kokkulunk, E. Akdogan, and V. Ayhan, "Prediction of emissions and exhaust temperature for direct injection diesel engine with emulsified fuel using ANN," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 21, no. SUPPL. 2, pp. 2141–2152, 2013, doi: 10.3906/elk-1202-24.

[41] F. Jaliliantabar, B. Ghobadian, G. Najafi, and T. Yusaf, "Artificial neural network modeling and sensitivity analysis of performance and emissions in a compression ignition engine using biodiesel fuel," *Energies*, vol. 11, no. 9, p. 2410, Sep. 2018, doi: 10.3390/en11092410.

[42] G. Mao, C. Zhang, K. Shi, and P. Wang, "Prediction of the performance and exhaust emissions of ethanol-diesel engine using different neural network," *Energy Sources, Part A Recover. Util. Environ. Eff.*, vol. 0, no. 0, pp. 1–15, Aug. 2019, doi: 10.1080/15567036.2019.1656307.

[43] J. Cho and S. Song, "Prediction of hydrogen-added combustion process in T-GDI engine using artificial neural network," *Appl. Therm. Eng.*, vol. 181, no. May, p. 115974, Nov. 2020, doi: 10.1016/j.applthermaleng.2020.115974.

[44] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge: The MIT Press, 2006.

[45] B. Berger and F. Rauscher, “Robust Gaussian Process Modelling For Engine Calibration,” *IFAC Proc. Vol.*, vol. 45, no. 2, pp. 159–164, 2012, doi: 10.3182/20120215-3-AT-3016.00028.

[46] Y. S. Chati and H. Balakrishnan, “A Gaussian process regression approach to model aircraft engine fuel flow rate,” in *Proceedings of the 8th International Conference on Cyber-Physical Systems*, Apr. 2017, pp. 131–140, doi: 10.1145/3055004.3055025.

[47] X. Xu and Y. Zhao, “Prediction of Fuel Consumption per 100km for Automobile Engine Based on Gaussian Processes Machine Learning,” *Appl. Mech. Mater.*, vol. 34–35, pp. 1951–1955, Oct. 2010, doi: 10.4028/www.scientific.net/AMM.34-35.1951.

[48] S. Castric, L. Denis-Vidal, Z. Cherfi, G. J. Blanchard, and N. Boudaoud, “Modeling Pollutant Emissions of Diesel Engine based on Kriging Models: a Comparison between Geostatistic and Gaussian Process Approach,” *IFAC Proc. Vol.*, vol. 45, no. 6, pp. 1708–1715, May 2012, doi: 10.3182/20120523-3-RO-2023.00038.

[49] D. Bergmann, R. Geiselhart, and K. Graichen, “Modelling and Control of a Heavy-Duty Diesel Engine Gas Path with Gaussian Process Regression,” in *2019 18th European Control Conference (ECC)*, Jun. 2019, pp. 1207–1213, doi: 10.23919/ECC.2019.8796016.

[50] V. Aran and M. Unel, “Gaussian Process Regression Feedforward Controller for Diesel Engine Airpath,” *Int. J. Automot. Technol.*, vol. 19, no. 4, pp. 635–642, Aug. 2018, doi: 10.1007/s12239-018-0060-x.

[51] C. Friedrich, M. Auer, and G. Stiesch, “Model Based Calibration Techniques for Medium Speed Engine Optimization: Investigations on Common Modeling Approaches for Modeling of Selected Steady State Engine Outputs,” *SAE Int. J. Engines*, vol. 9, no. 4, pp. 2016-01–2156, Oct. 2016, doi: 10.4271/2016-01-2156.

[52] A. Kamath, R. A. Vargas-Hernández, R. V. Krems, T. Carrington, and S. Manzhos, "Neural networks vs Gaussian process regression for representing potential energy surfaces: A comparative study of fit quality and vibrational spectrum accuracy," *J. Chem. Phys.*, vol. 148, no. 24, pp. 1–7, 2018, doi: 10.1063/1.5003074.

[53] M. Taki, A. Rohani, F. Soheili-Fard, and A. Abdeshahi, "Assessment of energy consumption and modeling of output energy for wheat production by neural network (MLP and RBF) and Gaussian process regression (GPR) models," *J. Clean. Prod.*, vol. 172, pp. 3028–3041, 2016, doi: 10.1016/j.jclepro.2017.11.107.

[54] A. Vabalas, E. Gowen, E. Poliakoff, and A. J. Casson, "Machine learning algorithm validation with a limited sample size," *PLoS One*, vol. 14, no. 11, p. e0224365, Nov. 2019, doi: 10.1371/journal.pone.0224365.

[55] G. Xiong, W. Ma, N. Zhao, J. Zhang, Z. Jiang, and Z. Mao, "Multi-Type Diesel Engines Operating Condition Recognition Method Based on Stacked Auto-Encoder and Feature Transfer Learning," *IEEE Access*, vol. 9, pp. 31043–31052, 2021, doi: 10.1109/ACCESS.2021.3057399.

[56] E. Faghani, "Effect of Injection Strategies on Particulate Matter Emissions From HPDI Natural-Gas Engines," University of British Columbia, 2015.

[57] P. Ouelette, D. Goudie, and G. McTaggart-Cowan, "Progress in the development of natural gas high pressure direct injection for Euro VI heavy-duty trucks," in *Internationaler Motorenkongress 2016. Proceedings.*, Springer Vieweg, Wiesbaden, 2016, pp. 591–607.

[58] "Specifications Explained: CompactDAQ and CompactRIO Chassis and Controllers - NI." https://www.ni.com/sv-se/support/documentation/supplemental/17/specifications-

explained--cdaq-and-crio-chassis-and-controllers.html (accessed Mar. 30, 2021).

[59] P. Kirchen, “Steady-State and Transient Diesel Soot Emissions: Development of a Mean Value Soot Model and Exhaust-Stream and In-Cylinder Measurements,” ETH Zürich, 2008.

[60] T. J. Robinson, “Box-Behnken Designs,” in *Wiley StatsRef: Statistics Reference Online*, Chichester, UK: John Wiley & Sons, Ltd, 2014.

[61] S. d’Ambrosio and A. Ferrari, “Potential of double pilot injection strategies optimized with the design of experiments procedure to improve diesel engine emissions and performance,” *Appl. Energy*, vol. 155, pp. 918–932, Oct. 2015, doi: 10.1016/j.apenergy.2015.06.050.

[62] D. M. Probst *et al.*, “Evaluating Optimization Strategies for Engine Simulations Using Machine Learning Emulators,” *J. Eng. Gas Turbines Power*, vol. 141, no. 9, Sep. 2019, doi: 10.1115/1.4043964.

[63] A. K. Das and S. Dewanjee, “Optimization of Extraction Using Mathematical Models and Computation,” in *Computational Phytochemistry*, Elsevier, 2018, pp. 75–106.

[64] T. F. Yusaf, D. R. Buttsworth, K. H. Saleh, and B. F. Yousif, “CNG-diesel engine performance and exhaust emission analysis with the aid of artificial neural network,” *Appl. Energy*, vol. 87, no. 5, May 2010, doi: 10.1016/j.apenergy.2009.10.009.

[65] W. H. Kurniawan, S. Abdullah, Z. M. Nopiah, and K. Sopian, “The Development of Artificial Neural Network for Prediction of Performance and Emissions in a Compressed Natural Gas Engine with Direct Injection System,” in *SAE Technical Paper Series*, Oct. 2007, no. 724, pp. 776–790, doi: 10.4271/2007-01-4101.

[66] C. W. J. Mabson, “Emissions characterization of paired gaseous jets in a pilot-ignited

natural-gas compression-ignition engine," University of British Columbia, 2015.

[67] S. Roy, A. K. Das, V. S. Bhadouria, S. R. Mallik, R. Banerjee, and P. K. Bose, "Adaptive-neuro fuzzy inference system (ANFIS) based prediction of performance and emission parameters of a CRDI assisted diesel engine under CNG dual-fuel operation," *J. Nat. Gas Sci. Eng.*, vol. 27, pp. 274–283, Nov. 2015, doi: 10.1016/j.jngse.2015.08.065.

[68] A. M. Bertram, Q. Zhang, and S.-C. Kong, "A novel particle swarm and genetic algorithm hybrid method for diesel engine performance optimization," *Int. J. Engine Res.*, vol. 17, no. 7, pp. 732–747, Sep. 2016, doi: 10.1177/1468087415611031.

[69] G. McTaggart-Cowan, J. Huang, and S. Munshi, "Impacts and Mitigation of Varying Fuel Composition in a Natural Gas Heavy-Duty Engine," *SAE Int. J. Engines*, vol. 10, no. 4, pp. 2017-01–0777, Mar. 2017, doi: 10.4271/2017-01-0777.

[70] D. Sciortino, F. Bonatesta, E. Hopkins, C. Yang, and D. Morrey, "A Combined Experimental and Computational Fluid Dynamics Investigation of Particulate Matter Emissions from a Wall-Guided Gasoline Direct Injection Engine," *Energies*, vol. 10, no. 9, p. 1408, Sep. 2017, doi: 10.3390/en10091408.

[71] Y.-H. Pu, J. Keshava Reddy, and S. Samuel, "Machine learning for nano-scale particulate matter distribution from gasoline direct injection engine," *Appl. Therm. Eng.*, vol. 125, pp. 336–345, Oct. 2017, doi: 10.1016/j.applthermaleng.2017.07.021.

[72] V. Cocco Mariani, S. Hennings Och, L. dos Santos Coelho, and E. Domingues, "Pressure prediction of a spark ignition single cylinder engine using optimized extreme learning machine models," *Appl. Energy*, vol. 249, no. November 2018, pp. 204–221, Sep. 2019, doi: 10.1016/j.apenergy.2019.04.126.

[73] F. Yang, H. Cho, H. Zhang, J. Zhang, and Y. Wu, "Artificial neural network (ANN) based

prediction and optimization of an organic Rankine cycle (ORC) for diesel engine waste heat recovery," *Energy Convers. Manag.*, vol. 164, no. February, pp. 15–26, May 2018, doi: 10.1016/j.enconman.2018.02.062.

[74] P. I. Frazier, *A Tutorial on Bayesian Optimization*. 2018.

[75] "Box-Behnken Design - an overview | ScienceDirect Topics." https://www.sciencedirect.com/topics/engineering/box-behnken-design (accessed Mar. 31, 2021).

[76] C. Guardiola, B. Pla, D. Blanco-Rodriguez, and P. Cabrera, "A learning algorithm concept for updating look-up tables for automotive applications," *Math. Comput. Model.*, vol. 57, no. 7–8, pp. 1979–1989, Apr. 2013, doi: 10.1016/j.mcm.2012.02.001.

[77] K. Nikzadfar and A. H. Shamekhi, "Investigating the relative contribution of operational parameters on performance and emissions of a common-rail diesel engine using neural network," *Fuel*, vol. 125, pp. 116–128, Jun. 2014, doi: 10.1016/j.fuel.2014.02.021.

[78] R. Finesso, D. Misul, and E. Spessa, "Development and validation of a semi-empirical model for the estimation of particulate matter in diesel engines," *Energy Convers. Manag.*, vol. 84, pp. 374–389, Aug. 2014, doi: 10.1016/j.enconman.2014.04.053.

[79] M. Papež and A. Quinn, "Transferring model structure in Bayesian transfer learning for Gaussian process regression," pp. 1–13, Jan. 2021, [Online]. Available: http://arxiv.org/abs/2101.06884.

[80] K. Wang, J. Chen, L. Xie, and H. Su, "Transfer learning based on incorporating source knowledge using Gaussian process models for quick modeling of dynamic target processes," *Chemom. Intell. Lab. Syst.*, vol. 198, no. December 2019, p. 103911, Mar. 2020, doi: 10.1016/j.chemolab.2019.103911.

[81] N. Wagle and E. W. Frew, “Forward Adaptive Transfer of Gaussian Process Regression,” *J. Aerosp. Inf. Syst.*, vol. 14, no. 4, pp. 214–231, Apr. 2017, doi: 10.2514/1.I010437.

[82] J. Rochussen, J. Yeo, M. Khosravi, and P. Kirchen, “Investigation of Pilot-Ignited Dual-Fuel Natural Gas Combustion in an Optically Accessible Engine,” pp. 18–23, 2015.

Appendices

Appendix A Additional Engine Components

This section contains information on additional components that are part of the test bench and were used as part of the research conducted in this work but otherwise are not critical to the understanding of how data was collected. These components are critical to the operation of the test bench and knowledge of their operation is important for the replication of data collection should someone want to reproduce this work. For this reason, they are included here.

A.1 Starter Motor

In addition to the Vector Drive and Engine Dynamometer, the engine system has a connected starter motor. The starter motor is powered by a completely independent deep cycle battery system that is charged using an inverter. This is necessary as the instantaneous power draw at start up can approach 600 amps from the 48V circuit. This is a significant amount of power but is absolutely necessary to overcome the static friction of the engine and get the engine up to its start-up speed of 333 RPM. Depending on the starting temperature of the engine, this power draw can be very significant and may need to be continued for some time before the engine can reach that speed. A diagram of the start motor circuit is shown in Figure A.1.

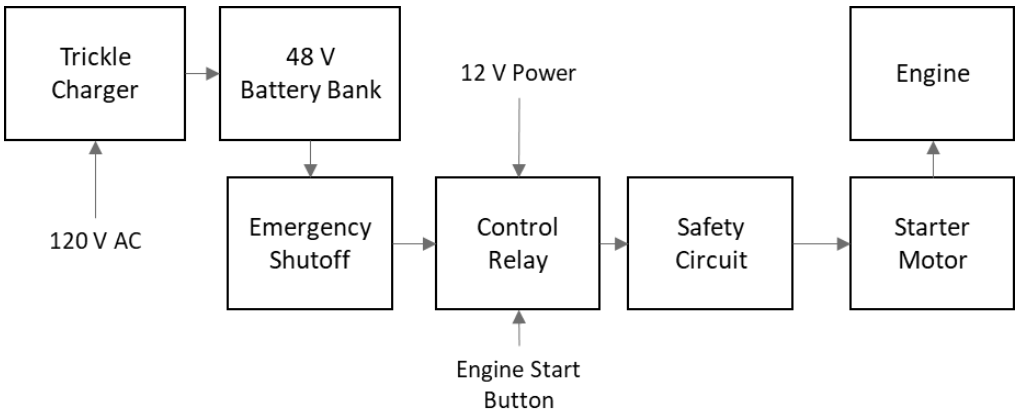


Figure A.1: Starter Motor Diagram

A.2 Cooling System

The cooling system serves two purposes. The first is to keep the temperature of the engine and dynamometer within their safe operating conditions. The second purpose is to heat the engine when it is not in use. This is required as the engine has very little power available at start up and a completely cold start is taxing on the engine system, especially in terms of wear and tear. At a completely cold start, which can approach 0 degrees depending on the outside air temperature, the SCRE is barely able to turn over and is unable to reach the minimum start-up speed. This is even with the full power of the Vector Drive and the starter motor. To facilitate an easy start up, the engine is kept at a constant 32 degrees Celsius or higher prior to starting. The cooling system is one of the many aspects of the engine that have received a significant amount of attention over the past several years as the system runs through all the major mechanical components of the engine and maintains them at a constant temperature. When the engine is not running, the coolant is cycled through the engine using an external pump. The coolant runs through a heat exchanger that runs to a separate cold water line. In order for cooling to take place, the cold water line valve must be opened to allow for the flow of cold non-potable water to run through the heat exchanger and cool

the coolant. Otherwise, the coolant is heated using an external heater. There is an additional valve on the cold water line which regulates the percentage of new cold water is brought into the heat exchanger. This is controlled electronically. A diagram of the system is given in Figure A.2.

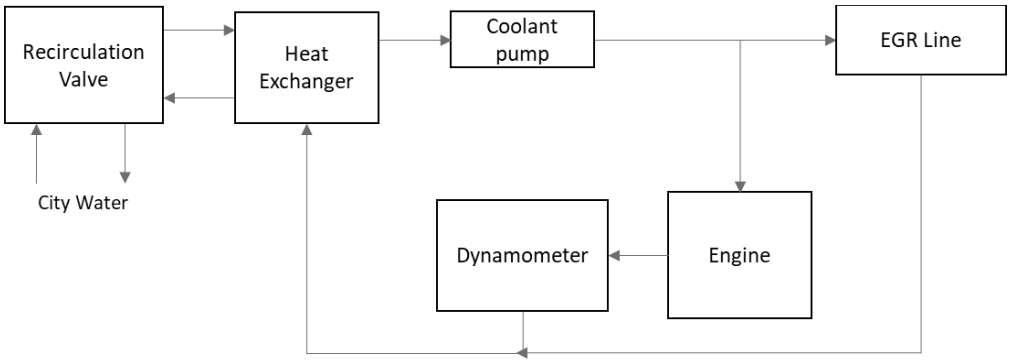


Figure A.2: System Cooling Loop

A.3 Control Panel

The final aspect of the engine control and data acquisition system is the control panel. The control panel is unique from the rest of the system as it is not directly interfaced with a computer. Instead, the user controls a variety of knobs, dials and switches that adjust various key parameters that maintain the engine in a stable operating state. This also means that some of these setpoints are not directly recorded by the data acquisition system. The main aspects of the engine that are controlled though these means are the fuel pressures, the engine dynamometer setpoint, the vector drive setpoint, and both the EGR and back pressure valve positions. A picture of the control panel is shown in Figure A.3.

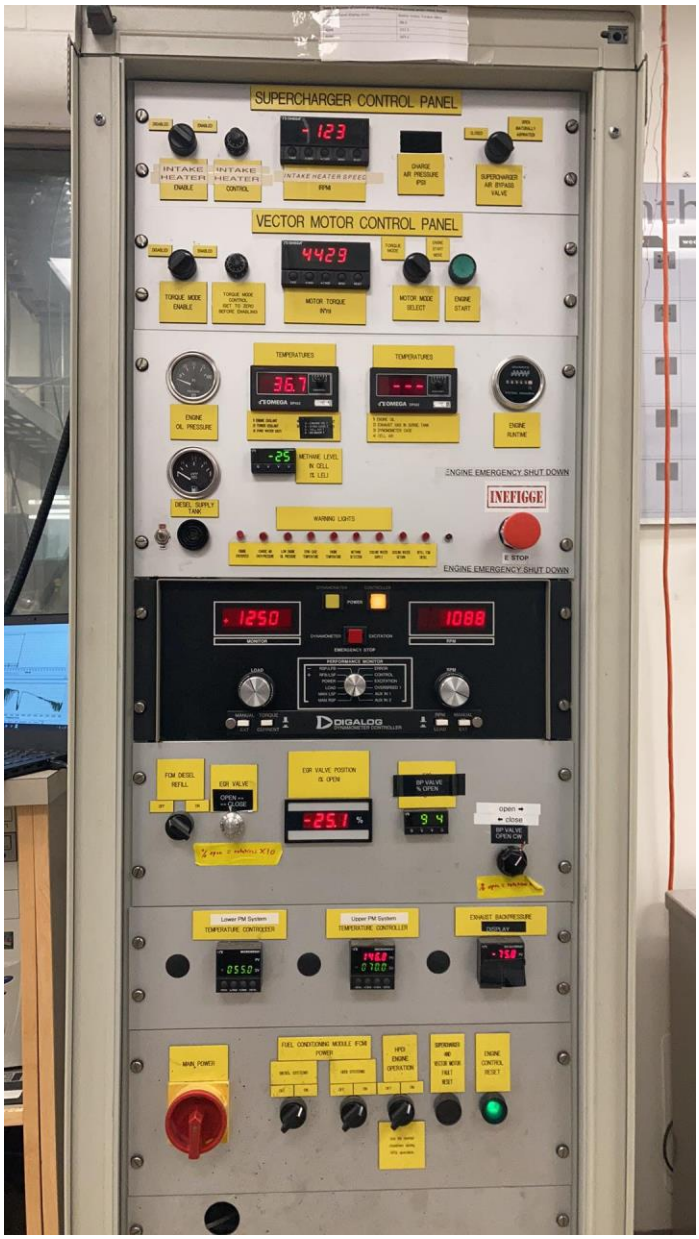


Figure A.3: Control Panel

The fuel pressures are controlled using a single pressure regulator which adjusts the diesel pressure and the corresponding natural gas pressure which is kept at a constant bias pressure of approximately 1 MPa. The engine dynamometer is controlled using a proprietary controller that monitor's the engine speed and torque. A speed setpoint is set and the dynamometer applies a

braking torque to maintain the speed. The back pressure and EGR valves are controlled using potentiometers that set the valve position. A coarse feedback is returned to a display on the panel indicating the valve position for the back pressure while the EGR does not have a valve setpoint display. Of the above parameters only the valve positions are not recorded in the data acquisition system. Future work is expected to improve on this limitation. The control panel also contains a manual ESD button, a system wide ESD reset, a start button to engage the starter motor, and a variety of switches that are used during start up to enable and power the various engine subsystems.

Appendix B Additional Control and Data Acquisition System Diagrams

This section contains additional diagrams detailing the control and data acquisition system. These diagrams are not necessary for understanding how these systems operate at a higher level but instead provide information on the detailed systems and how they are configured.

B.1 CompactDAQ UI Overview

This section provides a snapshot of the high level user interface for the data acquisition system which is used for the real-time monitoring of the engine test bench as well as for data collection. The diagram is given in Figure B.1.

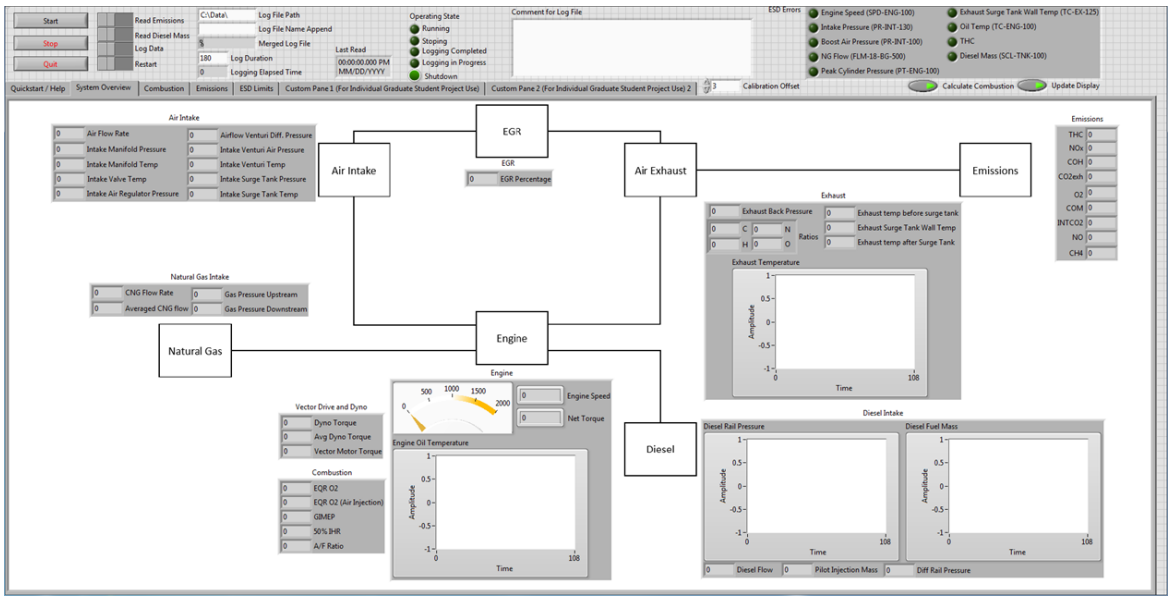


Figure B.1: Data Acquisition User Interface

B.2 ESD Circuit Wiring

This section provides a diagram of the emergency shutdown circuit and all the relays that are connected to it. The diagram is given in Figure B.2.

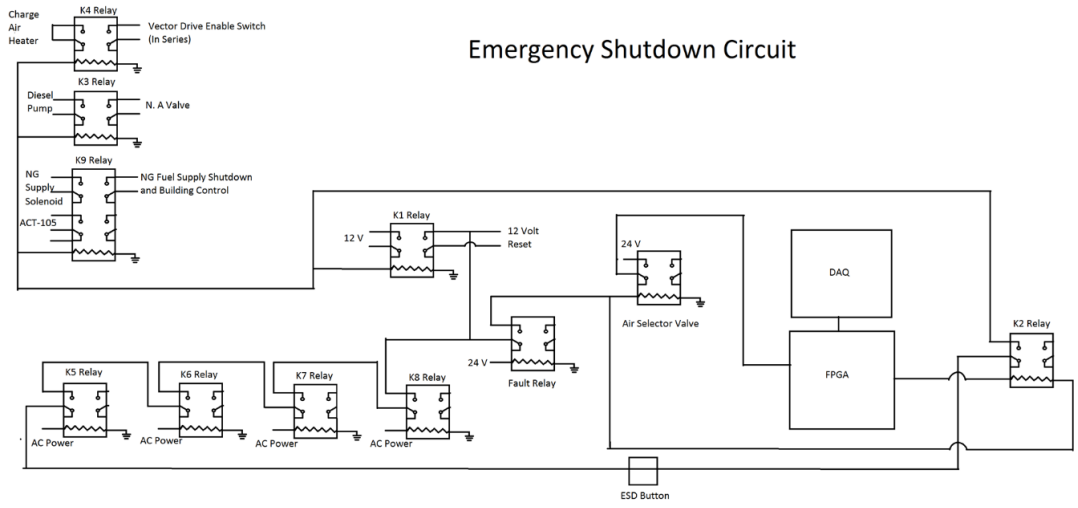


Figure B.2: Emergency Shutdown Circuit

B.3 CompactDAQ and CompactRIO Signal Wiring

This section gives the wiring diagrams for the control and data acquisition system which details all the connected components and the detailed hardware used in the CompactDAQ and CompactRIO systems. Figure B.3 shows the data acquisition system wiring and Figure B.4 shows the control system wiring.

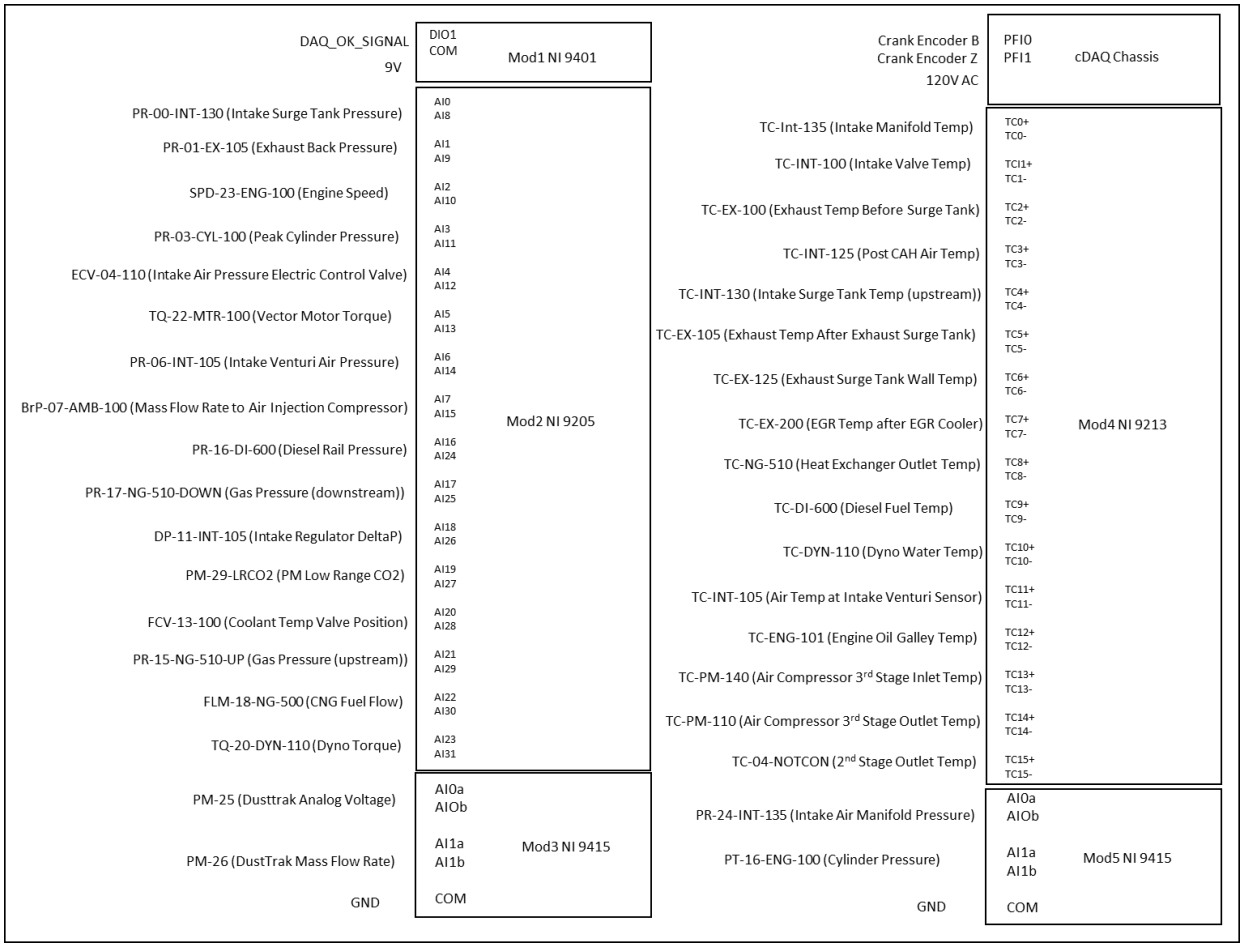


Figure B.3: CompactDAQ Signal Wiring

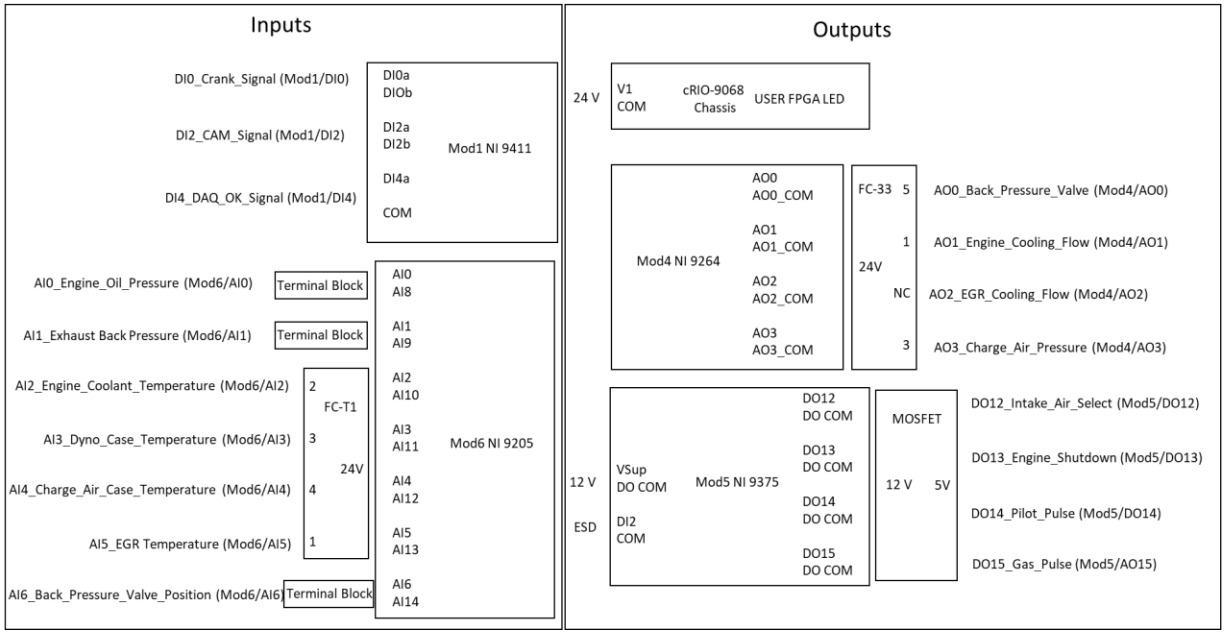


Figure B.4: CompactRIO Control and Sensor Wiring

Appendix C Machine Learning Hyperparameters

Machine learning models have a variety of hyperparameters. The artificial neural network and Gaussian process regression model hyperparameters that were considered in this work are discussed in the following sections.

C.1 Artificial Neural Network Hyperparameters

The artificial neural network has several different hyperparameters. This includes the training function, the number of hidden layers, and the number of nodes in each of the hidden layers. Variations in each of these parameters can have a drastic impact on the model performance [72]. The most common training functions used in neural network engine research are Levenberg-Marquart and Bayesian Regularization with one or two hidden layers and the number of nodes

being dependent on the number of inputs and outputs in the model [41]. Table 7.1 shows the hyperparameter options for the neural network models.

Table 7.1: Neural network hyperparameters

Hyperparameters	Training Function	Number of Layers	Number of Nodes (Unique for each layer)
Range	• Levenberg-Marquardt • Bayesian Regularization • BFGS quasi-Newton • Resilient • Scaled conjugate gradient • Conjugate gradient with Powell-Beale restarts • One-step secant • Gradient descent with momentum and adaptive learning rate • Gradient descent with momentum • Gradient descent	1 to 6	2 to 12

C.2 Gaussian Process Regression Hyperparameters

The Gaussian process regression has two main hyperparameters. The first hyperparameter is the kernel function which defines the shape of the covariance between training data points and also defines the final curvature of the mean function [44]. The squared exponential is the most commonly used kernel function [50]. The second hyperparameter is σ^n or the initialized value for

the signal variance. This impacts how closely the kernel function tries to fit the training data set [44]. The range of the hyperparameters is shown in Table 7.2.

Table 7.2: Gaussian process regression hyperparameters

Hyperparameter	Kernel Function	Initial Value for Noise Variance (σ^n)
Range	• Exponential kernel • Squared exponential kernel • Matérn kernel with covariance parameter 3/2 • Matérn kernel with covariance parameter 5/2 • Rational quadratic kernel • Exponential kernel with a separate length scale per predictor • Squared exponential kernel with a separate length scale per predictor • Matérn kernel with covariance parameter 3/2 and a separate length scale per predictor • Matérn kernel with covariance parameter 5/2 and a separate length scale per predictor • Rational quadratic kernel with a separate length scale per predictor	0.0001 to 30

Appendix D Layered Model Structure Matrices

The following sections present the layered model input/output matrices which show the detailed interconnection between the various submodels in the model structure.

D.1 Layered Model Input/Output Matrix

Table 7.3 shows a description of the inputs and outputs for each submodel in the layered model.

Table 7.3: Input layout for each layered model submodel

Model Layer	Output	Model Inputs																		
		Measured Inputs											Calculated Inputs Model Layer							
													1		2	3		4		5
		p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}	m_f	m_{ng}	EQR	θ_{50}	θ_{10}	**η**	$\mathbf{P_{gross}}$	$\mathbf{p_{cyl}}$
1	m_f	X	X	X	X	X	X	X												
	m_{ng}	X		X			X	X												
2	**EQRO**	X							X	X	X		X							
	EQR	X							X	X	X		X							
3	θ_{50}	X			X		X	X	X	X				X	X					
	θ_{10}	X	X		X	X	X		X	X				X	X					
	$\mathbf{O_2}$	X							X	X	X		X		X					
4	**η**	X							X	X					X	X				
	$\mathbf{P_{gross}}$	X		X	X		X	X	X	X			X		X	X				
5	$\mathbf{CO_2}$								X				X		X			X		
	CO	X	X		X	X			X	X	X	X			X	X			X	
	$\mathbf{p_{cyl}}$	X								X	X	X	X		X	X	X		X	
	PM	X		X	X		X		X	X	X	X			X				X	
	dp/dCA	X			X		X		X	X	X	X	X		X	X			X	
6	$\mathbf{NO_X}$	X							X	X	X	X	X		X	X			X	X
	$\mathbf{CH_4}$	X							X	X	X	X			X	X			X	X

* Bolded values are outputs.

D.2 Hybrid Model Input/Output Matrix

Table 7.4 shows the input and output matrix for the submodels in the hybrid modeling method. The bolded values represent outputs and underlined values represent equations.

Table 7.4: Input loyout for each hybrid model submodel

Model Layer	Output	Model Inputs																					
		Measured Inputs											Calculated Inputs Model Layer										
													1					2	3		4	5	6
		p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}	m_f	m_{diesel}	m_{ng}	m_{air}	EGR	EQR	θ_{50}	θ_{10}	$\mathbf{P_{gross}}$	$\boldsymbol{\eta}$	$\mathbf{p_{cyl}}$
1	m_f	X	X	X	X	X	X	X															
	m_{diesel}	X	X		X	X																	
	m_{ng}	X		X			X	X															
	m_{air}	X							X	X	X												
	EGR	X							X	X													
2	**EQRO**												X			X							
	EQR												X			X	X						
3	θ_{50}	X			X		X	X	X	X					X			X					
	θ_{10}	X	X		X	X	X		X	X					X			X					
	O_2												X			X		X					
4	$\mathbf{P_{gross}}$	X		X	X		X	X	X	X			X					X	X				
5	$\boldsymbol{\eta}$													X	X						X		
6	**CO_2**								X				X					X				X	
	CO	X	X		X	X			X	X	X	X						X	X		X		
	$\mathbf{p_{cyl}}$	X								X	X	X	X					X	X	X	X		
	PM	X		X	X		X		X	X	X	X						X			X		
	dp/dCA	X			X		X		X	X	X	X	X					X	X		X		
7	$\mathbf{NO_X}$	X							X	X	X	X	X					X	X		X		X
	$\mathbf{CH_4}$	X							X	X	X	X						X	X		X		X

* Underlined values represent equations. Bolded values are outputs.

Appendix E Training Data Results

The following sections contain the training data results for the different modeling methods presented in this work. These results are included for completeness and to visualize the variation in nRMSE and COD when transitioning from the idealized training data to the final validation data.

E.1 Full Factorial Data Hyperparameter Optimization Training Results

The full factorial training data represents a data set for a small operating space that has been measured at a high resolution. Using the hyperparameter optimization method, improvements in the machine learning models is expected. As the response surface models presented very poor results in section 4.1.1 these results are not repeated here. Instead, only the machine learning results are presented. Figure E.1 shows the coefficient of determination for the full factorial hyperparameter optimized models using the training data.

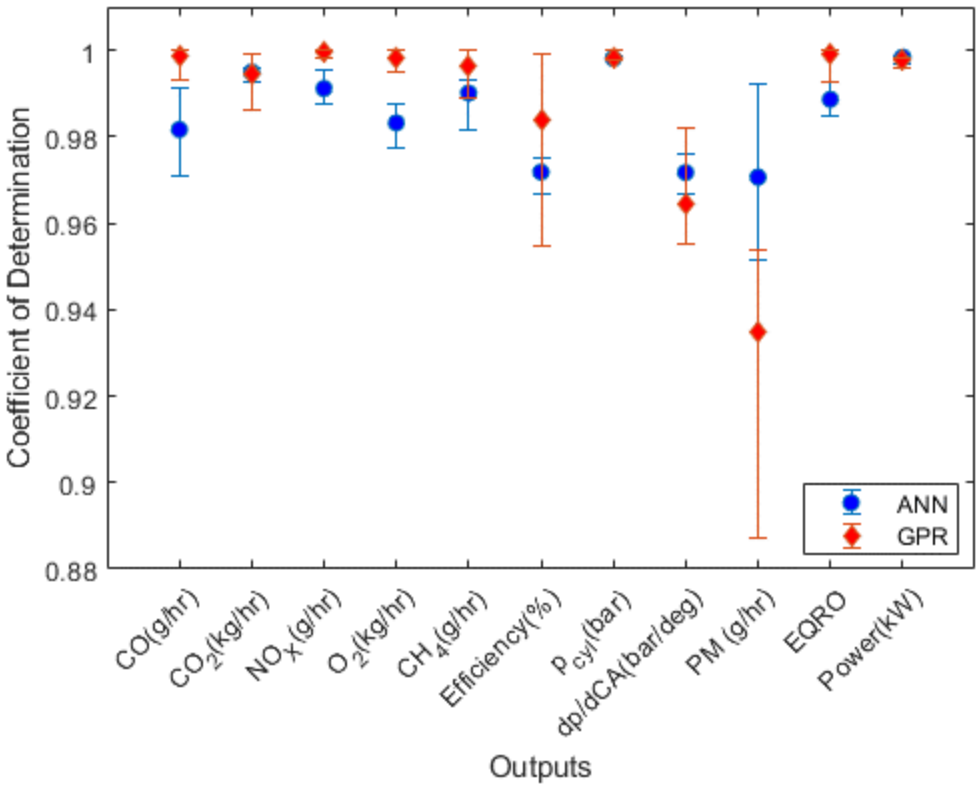


Figure E.1: Coefficient of Determination for Hyperparameter Optimized Modeling with Full Factorial Data (Training Results)

Figure E.1 shows that the training results with hyperparameter optimization are immediately much better than the results presented in 4.1.1. Even the lowest measured coefficient of determination is above 0.88 with average values consistently about 0.93. This shows that the hyperparameter optimization allowed the models to find a configuration that properly characterized the training data set for both neural networks and Gaussian process regression. Figure E.2 shows the normalized root mean square error for these models.

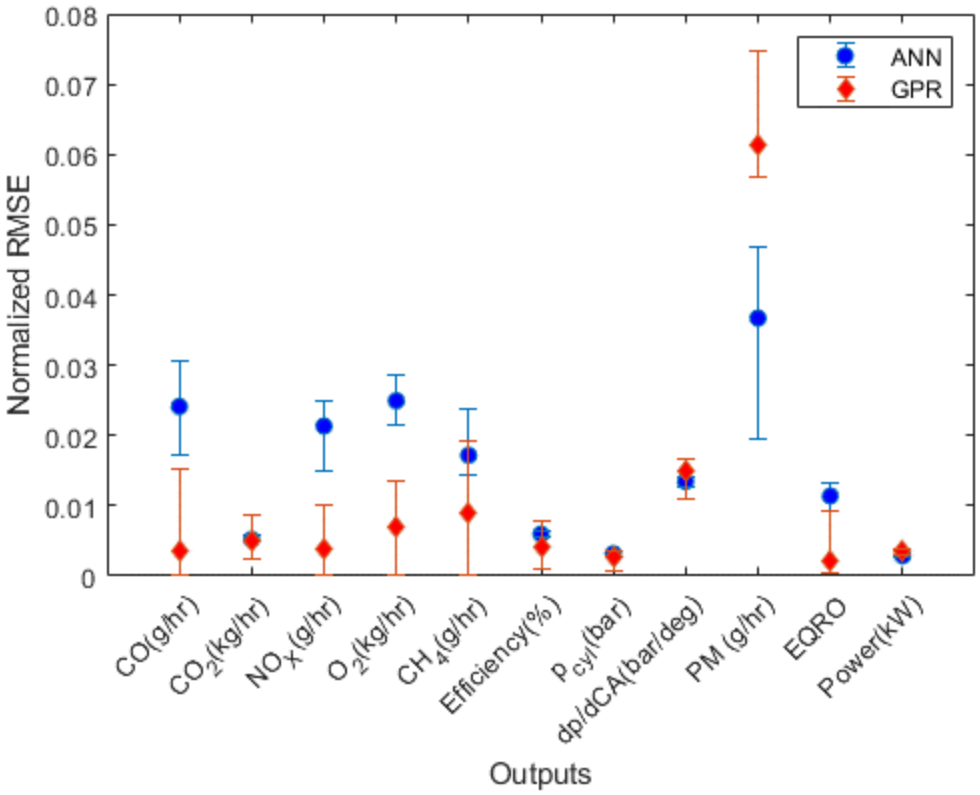


Figure E.2: Normalized RMSE for Hyperparameter Optimized Modeling with Full Factorial Data (Training Results)

Figure E.2 confirms the previous results. Except for the particulate matter, the normalized root mean square error for all models is consistently below 0.04. The particulate matter shows nRMSE values that approach 0.08 for the Gaussian process regression and even this value, which is twice as high as the other results, is still considered reasonable. Table 5.1 shows a summary of these results.

Table 7.5: Comparison of nRMSE results for hyperparameter optimized modeling with the modeling survey using the full factorial data (training results)

Outputs	Machine Learning Survey		Hyperparameter Optimization *	
	Artificial Neural Network	Gaussian Process Regression	Artificial Neural Network	Gaussian Process Regression
CO	0.219	0.041	0.024	**0.004**
CO_2	0.054	0.018	**0.005**	**0.005**
NO_X	0.196	0.045	0.021	**0.004**
O_2	0.136	0.049	0.025	**0.007**
CH_4	0.183	0.042	0.017	**0.009**
Efficiency	0.032	0.014	0.006	**0.004**
p_{cyl}	0.095	0.016	**0.003**	**0.003**
dp/dCA	0.040	0.023	**0.013**	0.015
PM	0.276	0.130	**0.037**	0.061
EQRO	0.067	0.034	0.011	**0.002**
Power	0.065	0.019	**0.003**	**0.003**
Average	0.124	0.039	0.017	**0.011**

* Lowest nRMSE in bold.

Overall, the Gaussian process regression models showed the lower error when compared to the neural network models. While these results only represent the training data it is apparent that

proper selection of hyperparameters has had a positive effect on the results when compared to the modeling survey results.

E.2 Box Behnken Data Hyperparameter Optimization Training Results

This section presents the training data model performance for the hyperparameter optimized Box Behnken modeling method. Figure E.3 shows the coefficient of determination results for the Box Behnken training data set neural network, Gaussian process regression, and Response Surface models.

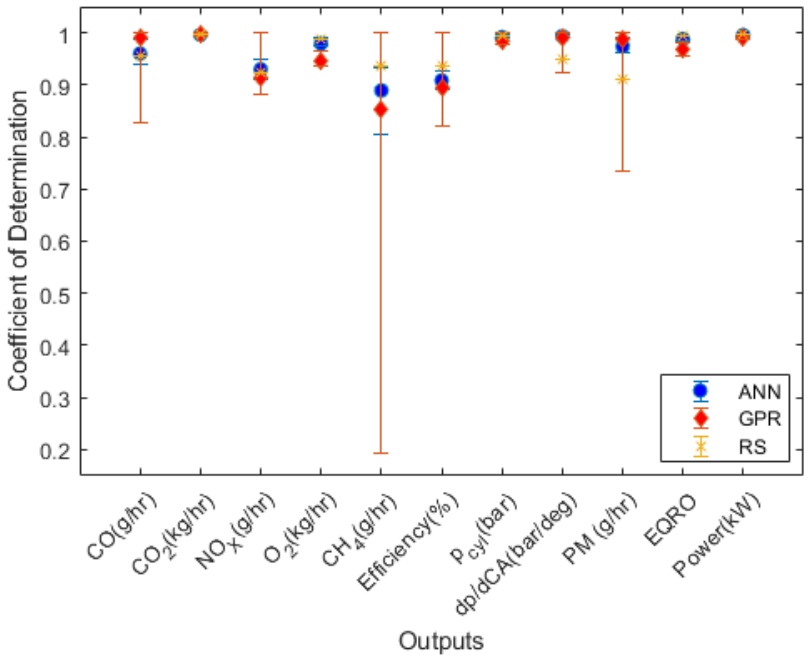


Figure E.3: Coefficient of Determination for Hyperparameter Optimized Modeling with Box Behnken Data (Training Results)

As with the full factorial data set, the Box Behnken data set shows high coefficient of determination results for the training data. The average values for all the models in Figure E.3 are above 0.8 indicating that the models were able to accurately follow the patterns in the Box Behnken data set. The only points of concern are the Gaussian process regression model results for the CH_4 which show a range that extends down to 0.2. This indicates that some optimizations were unable to find capable models that tracked the trends for this data set. Figure E.4 shows the normalized RMSE results for the training data.

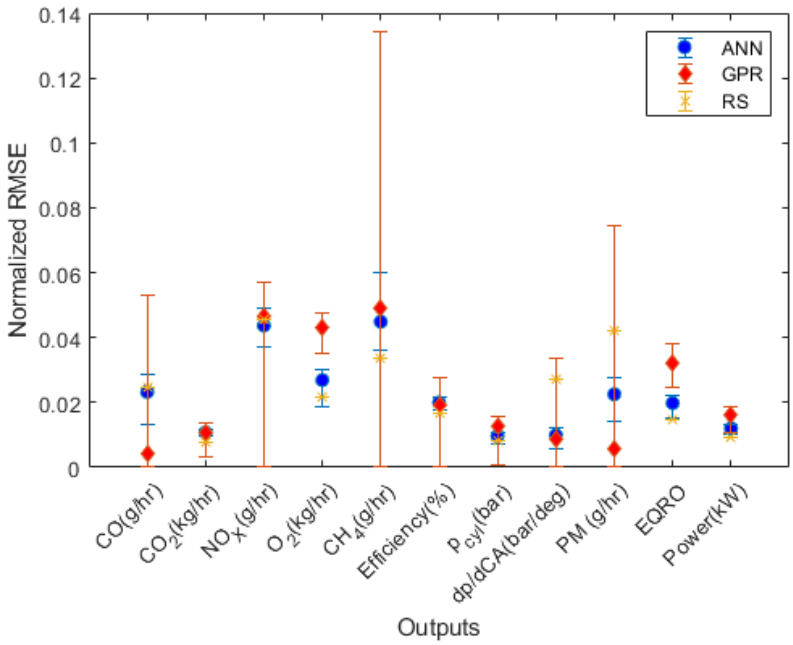


Figure E.4: Normalized RMSE for Hyperparameter Optimized Modeling with Box Behnken Data (Training Results)

In Figure E.4, the results show that Gaussian process regression has the highest variation of all the data sets. This is especially true for the hydrocarbons. Otherwise, the average nRMSE is roughly

on par with the neural network results and the response surface results. Table 7.6 shows a summary of these results.

Table 7.6: Comparison of nRMSE results for hyperparameter optimized modeling and the modeling survey using the Box Behnken data (training results)

Outputs	Modeling Survey *			Hyperparameter Optimization *	
	Response Surface	Artificial Neural Network	Gaussian Process Regression	Artificial Neural Network	Gaussian Process Regression
CO	0.025	0.095	0.023	0.023	**0.004**
CO_2	**0.008**	0.033	0.045	0.011	0.011
NO_X	0.045	0.114	**0.017**	0.044	0.047
O_2	**0.022**	0.076	0.030	0.027	0.043
CH_4	**0.034**	0.120	0.058	0.045	0.049
Efficiency	**0.017**	0.030	0.016	0.020	0.019
p_{cyl}	**0.008**	0.042	0.011	0.010	0.013
dp/dCA	0.027	0.049	0.019	0.010	**0.009**
PM	0.042	0.096	0.028	0.023	**0.006**
EQRO	**0.015**	0.041	0.061	0.020	0.032
Power	**0.009**	0.043	0.040	0.012	0.016
Average	**0.024**	0.067	0.032	**0.024**	**0.024**

* Lowest nRMSE in bold.

As expected from Figure E.4, Table 7.6 shows that the overall results for the training data are essentially equivalent with average nRMSE values of 0.024 for both hyperparameter optimized methods and the response surface. In most cases, the hyperparameter optimized models show lower average nRMSE compared to the modeling survey results. The response surface still shows the lowest nRMSE compared to the other modeling methods for 7 of the outputs. This indicates that the response surface shows the best training data performance overall.

E.3 Multi-Region Training Results

This section presents the multi-region model training data results. These results present the idealized, peak model performance and show how well the models were able to determine the patterns and trends in the original training data. Figure E.5 shows the training data coefficient of determination for the multi-region model for the two Box Behnken regions compared to a single combined data set model for both regions. Both the individual component results and the averaged results for the multi-region model are presented.

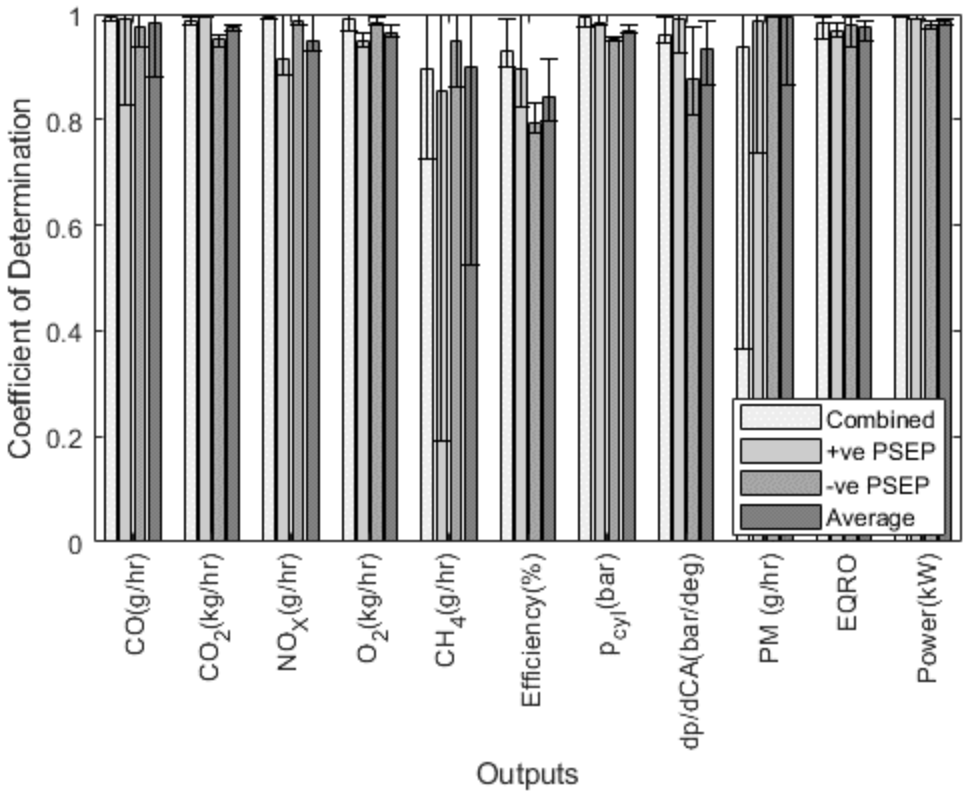


Figure E.5: Coefficient of Determination for Multi-Region Models Compared with a Combined Single Region Model (Training Results)

The results from Figure E.5 show that most models have coefficients of determination above 0.8. CH_4 and particulate matter show variations in the coefficient of determination ranging as much as 0.8 for the positive PSEP Box Behnken data set CH_4 models and 0.6 for the combined data set particulate matter models. The combined model typically presents the highest average coefficient of determination by a margin of less than 0.05 for most models. Figure E.6 presents the normalized RMSE results for these models using the training data.

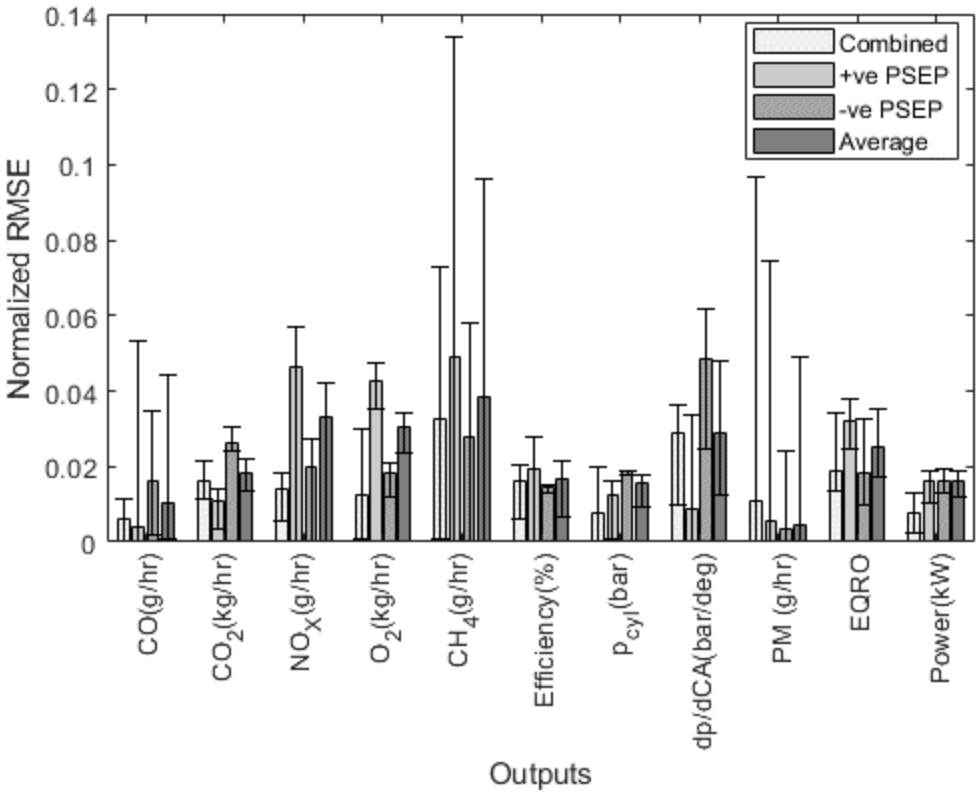


Figure E.6: Normalized RMSE for Multi-Region Models Compared with a Combined Single Region Model (Training Results)

Figure E.6 shows that based solely only the training data, the combined model outperforms the multi-region model with the lowest nRMSE on average and the least variation as compared to the average model results. From these results, the extra data provided to the Gaussian process regression model by combing both data sets proves to be more beneficial than simplifying the model by separating the data sets into different combustion regions. These results are reinforced by Table 7.7 where the results are summarized and the percentage reduction in nRMSE is shown.

Table 7.7: Normalized RMSE for the multi-region model compared to a combined single region model (training data)

Outputs	Multi-Region nRMSE *			Combined nRMSE	% Reduction in nRMSE *		
	+ve PSEP	-ve PSEP	Average		+ve PSEP	-ve PSEP	Average
CO	**0.004**	0.016	0.01	0.006	**31.96**	-174.54	-66.67
CO_2	**0.011**	0.026	0.019	0.016	**33.40**	-63.07	-15.63
NO_X	0.047	0.020	0.034	0.014	-232.4	-43.04	-139.3
O_2	0.043	0.018	0.031	0.012	-249.7	-50.56	-154.2
CH_4	0.049	**0.028**	0.039	0.032	-51.08	**14.05**	-20.31
Efficiency	0.019	**0.015**	0.017	0.016	-18.52	**10.28**	-6.250
p_{cyl}	0.013	0.019	0.016	0.008	-61.49	-137.5	-99.99
dp/dCA	**0.009**	0.049	**0.029**	0.029	**69.95**	-69.04	**0.001**
PM	**0.006**	**0.003**	**0.005**	0.011	**48.37**	**70.85**	**59.09**
EQRO	0.032	**0.018**	0.025	0.019	-69.56	**3.584**	-31.58
Power	0.016	0.016	0.016	0.008	-104.7	-103.7	-100.1
Average	0.023	0.021	0.022	0.016	-54.89	-49.33	-52.25

From the nRMSE results in Table 7.7 there is a clear decrease in performance for the training data of approximately 52% on average when splitting the model into two regions. This reinforces the general statement that machine learning models benefit from having as much data available as possible [43]. These results, however, only show the training performance which is not indicative of how these models perform for data that they were not trained with. It is those results that

represent the real-world usage of the model and as such, are the main results that determine the method's validity.

E.4 Sensitivity Analysis Based Input Reduction Training Data Results

Figure E.7 shows the coefficient of determination results for each of the different input sets using the training data with the sensitivity analysis based input reduction method.

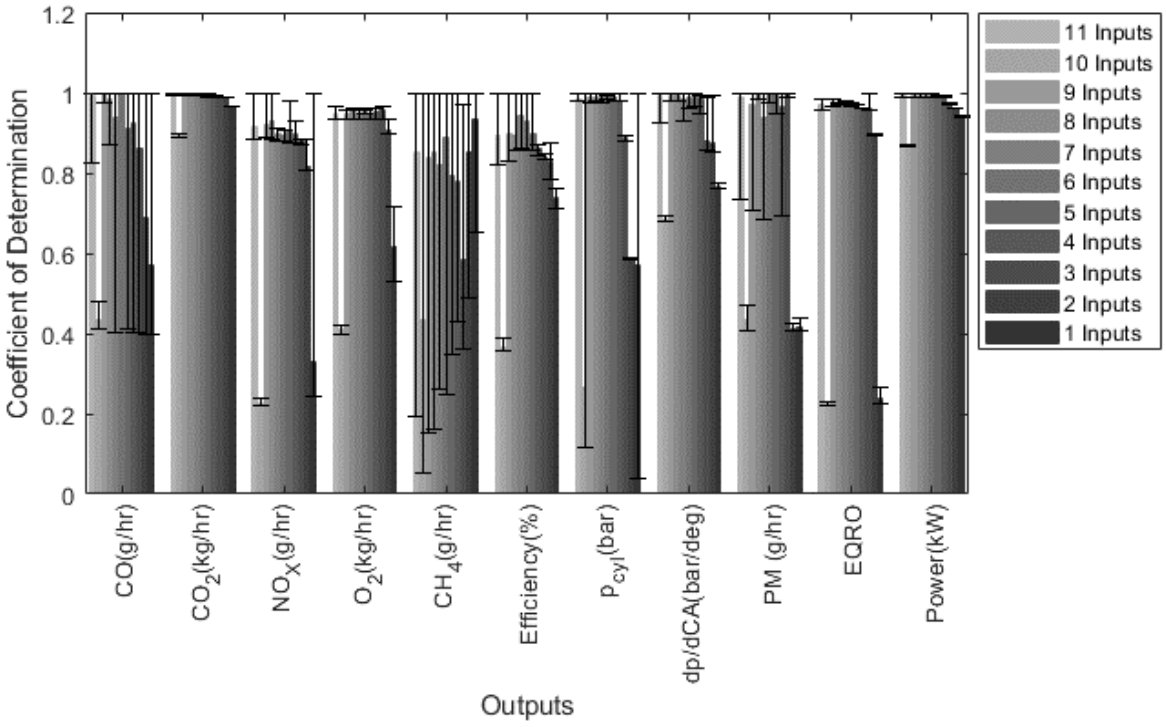


Figure E.7: Coefficient of Determination for Sensitivity Analysis Based Input Reduction (Training Data)

The results from Figure E.7 demonstrate that in most cases, the number of inputs do not impact the ability of the Gaussian process regression models to determine a model that reflects the trends in the training data. In many cases, the coefficient of determination drops as the input vector

reduces to one or two parameters. This is especially visible in the EQRO, particulate matter and NO_X. The EQRO is directly related to the quantity of air and fuel in the cylinder. While the quantity of air is captured in the most sensitive input, the diesel and natural gas quantity is difficult to determine after the fuel pulse widths are removed. Similarly, the NO_X is heavily dependent on the natural gas and diesel quantity, which are characterized by the top three most sensitive inputs. The loss of one of these inputs, such as the natural gas pulse width, removes information that is needed to create the NO_X model. The particulate matter is dependent on the localized distribution of heat, air, and fuel in the cylinder, which is partially determined by the natural gas start of injection and the diesel pressure. As the exhaust back pressure, which is closely related to the intake pressure, and the air temperature are removed, the coefficient of determination starts to decrease. Reducing the input vector down to one parameter eliminates important information that is needed to distinguish the training data points from one another.

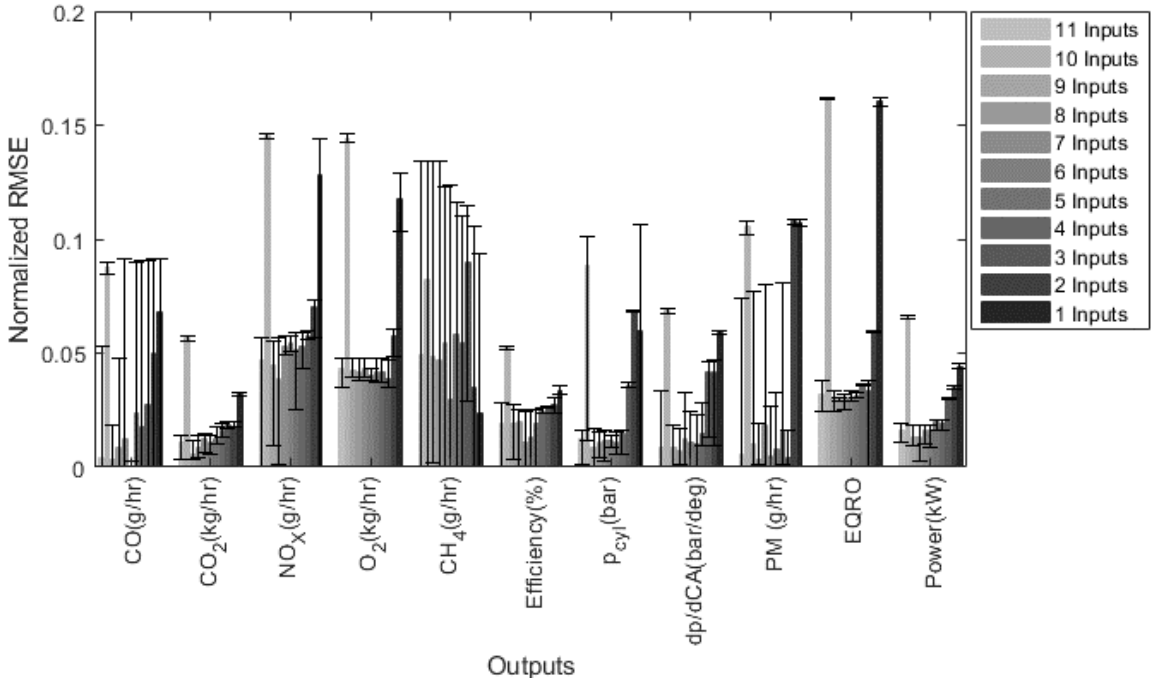


Figure E.8: Normalized RMSE for Sensitivity Analysis Based Input Reduction (Training Data)

Most of the nRMSE results shown in Figure E.8 are well below 0.15 indicating that the models are able to predict the training data output accurately on average despite the reduction of inputs. As inputs are removed, the nRMSE increases, with the removal of the last few inputs having the greatest impact on the error. This is similar to the coefficient of determination results. Some models have unusually high error at certain input levels. NO_X, Gross Indicated Power and EQRO all show examples of this. This is likely an indication that there is some sensitivity to the noise that specific inputs introduce as they provide less reliable or unnecessary information to the model. This noise might be harder to reject as the input vector shrinks in size and the unnecessary or noisy parameters take up a larger percentage of the available inputs. The summarized normalized RMSE results for the sensitivity based analysis technique as compared to the hyperparameter based GPR results are show in Table 7.8.

Table 7.8: Normalized RMSE results for sensitivity analysis based input selection (training results)

Outputs	Sensitivity Based Results	Hyperparameter Based Results	% Reduction in nRMSE	Inputs Used
CO	**0.001**	0.004	**84.65**	6
CO_2	**0.006**	0.011	**43.89**	9
NO_X	**0.039**	0.047	**17.08**	8
O_2	**0.039**	0.043	**8.951**	3
CH_4	**0.024**	0.049	**51.50**	1
Efficiency	**0.011**	0.019	**44.45**	7
p_{cyl}	**0.009**	0.013	**31.12**	9
dp/dCA	**0.007**	0.009	**17.49**	8
PM	**0.003**	0.006	**42.45**	8
EQRO	**0.030**	0.032	**7.593**	7
Power	**0.013**	0.016	**17.85**	9
Average	**0.017**	0.023	**33.37**	

* Lowest nRMSE in bold.

Table 7.8 shows that compared to the original hyperparameter based models, reducing the input vector results in a consistent decrease in model error. For CO, the model error decreases by 85% by using only the 6 most relevant inputs. Similarly, the particulate matter improves by 42% by using only 8 inputs. Overall, there is an average performance gain of 33% by removing unneeded inputs.

E.5 Layered Model Training Data Results

In this section, the submodels from 6.3.2 are arranged into the layered structure from Figure 6.7 and evaluated using the training data from the Box Behnken data set in section 3.2.2. Figure E.9 shows the coefficient of determination results for the layered modeling method using the training data.

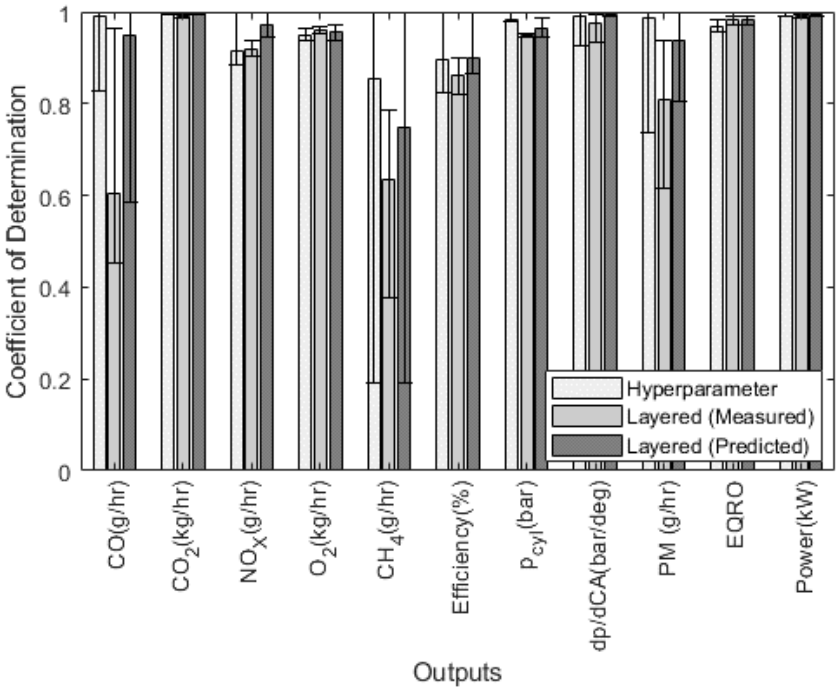


Figure E.9: Coefficient of Determination for Layered Modeling Method (Training Data)

The predicted approach, where the models are trained using the predicted inputs from the layered submodels, performs much better than the measured approach, where the models are trained using the experimental data. This is a result of the fact that the layered structure is used to originally produce the submodels in the predicted approach. As such, the "predicted" submodels are trained

to correct for the predicted submodel error in the training data. This is not the case with the measured approach. In most cases, the hyperparameter optimized model shows the highest average coefficient of determination. This is expected as the training data presents the ideal case for the Gaussian process regression models. The submodels are also trained using this data but have an additional level of complexity added through the layered model structure that could result in more error for the training data results.

It is important to note that certain models, such as the CO, CH_4, and the particulate matter, show substantially lower coefficients of determination for the measured approach, where the coefficient of determination is reduced to as little as 0.5. This is important as these outputs have consistently been difficult to model and may present problematic cases in the validation results. The measured approach is more sensitive to these issues for the training data as the measured submodels are not trained using the predicted values from layered model structure.

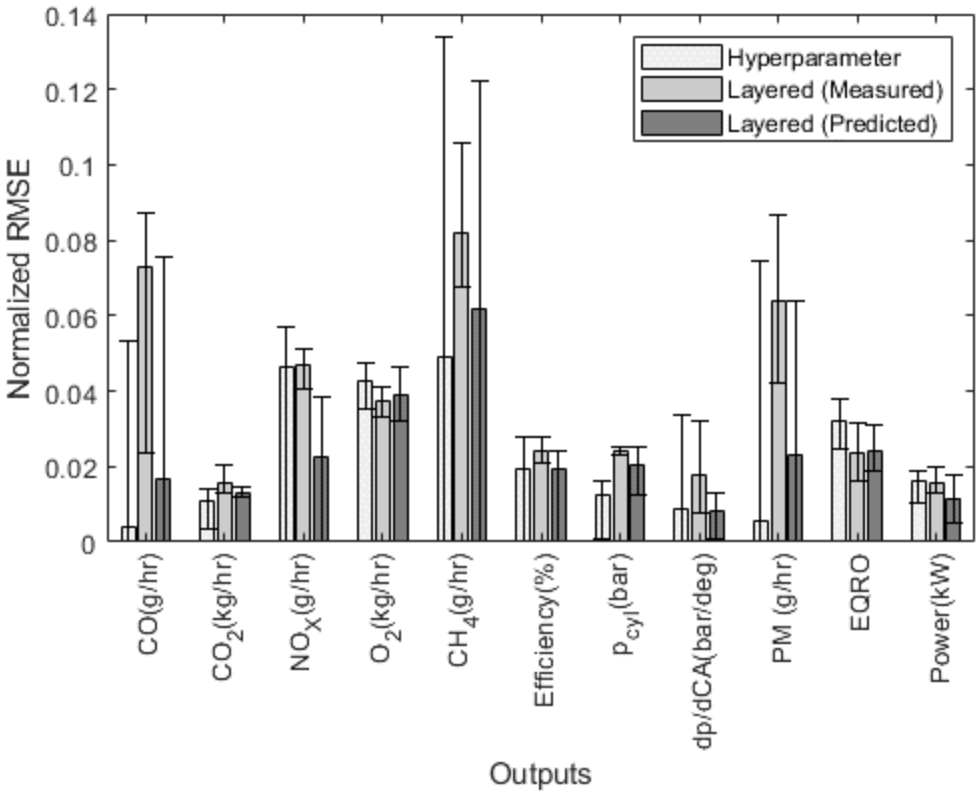


Figure E.10: Normalized RMSE for Layered Modeling Method (Training Data)

Figure E.10 shows the normalized RMSE results. These results present similar behaviour to the coefficient of determination. Again, the measured approach shows the worst training results with the highest nRMSE overall. This is expected as the training data results use the layered model structure and the "measured" submodels are not trained in this structure. As with the coefficient of determination results, the CO, CH_4, and particulate matter show the lowest performing results with the highest nRMSE. Although the training results show a worse overall average performance for the measured approach, the range of nRMSE is also smaller than either of the other two methods.

Table 7.9: Normalized RMSE results for the layered model (training results)

Outputs	Layered Model nRMSE *		Hyperparameter	% Reduction in nRMSE *	
	Measured	Predicted	Optimized nRMSE	Measured	Predicted
CO	0.073	0.017	0.004	-1721	-311.5
CO_2	0.016	0.013	0.011	-47.92	-20.79
NO_X	0.047	**0.022**	0.047	-0.894	**51.96**
O_2	**0.037**	**0.039**	0.043	**13.02**	**8.927**
CH_4	0.082	0.062	0.049	-66.77	-26.20
Efficiency	0.024	0.020	0.019	-26.69	-1.850
p_{cyl}	0.024	0.020	0.013	-91.41	-60.79
dp/dCA	0.018	0.008	0.009	-106.4	3.069
PM	0.064	0.023	0.006	-1014	-306.6
EQRO	**0.023**	**0.024**	0.032	**27.32**	**25.46**
Power	**0.016**	**0.012**	0.016	**2.343**	**27.54**
Average	0.039	0.024	0.022	-275.7	-55.53

* Lowest nRMSE in bold.

Table 7.9 shows the magnitude of the change in performance. Both methods show significantly worse results with some errors exceeding a 10 fold increase in nRMSE over the original hyperparameter optimization method. Some parameters such as the NO_X, O_2, EQRO, and power output did improve. This is a promising result as, again, the hyperparameter optimization method is favoured with this data set.

E.6 Hybrid Model Training Data Results

This section presents the hybrid model training data modeling performance results. These results demonstrate how well the model can capture the relationships between the inputs and outputs from the training data and create output models that reflects those relationships. The hybrid model is trained using the Box Behnken data set and is compared against the measured version of the layered model from Section 6.3 and the hyperparameter optimized model from Chapter 5. Each model is generated 10 times and the average result as well as the range of results is presented.

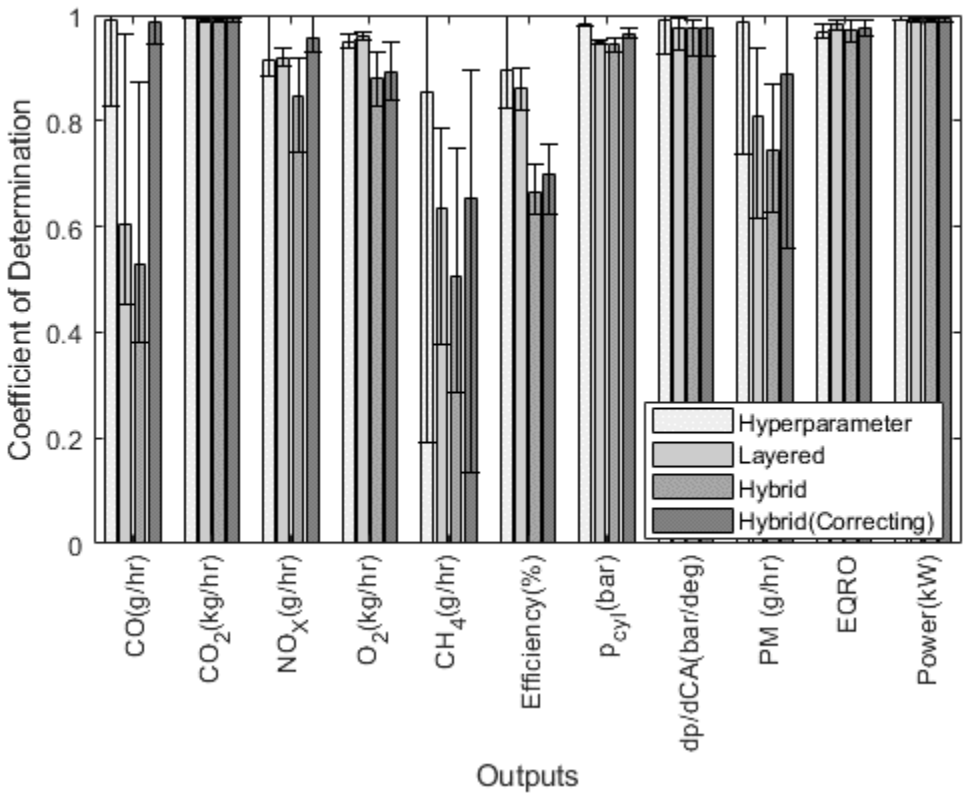


Figure E.11: Coefficient of Determination for Hybrid Modeling Method (Training Data)

Figure E.11 shows the coefficient of determination training data results for the hybrid model. The most obvious initial result is that the coefficient of determination for the hybrid model is the lowest for the training data across all the models. The hybrid model with error correction, however, shows a significant improvement over the basic hybrid model and the layered model. For CO, the coefficient of determination is close to 1 which puts it in the same performance bracket as the hyperparameter optimized model and a full 0.5 above the basic hybrid model. This trend remains consistent across all the other models with the error correcting hybrid model showing the highest coefficient of determination of all the layered model types. In most cases the hyperparameter optimized model shows the highest coefficient of determination.

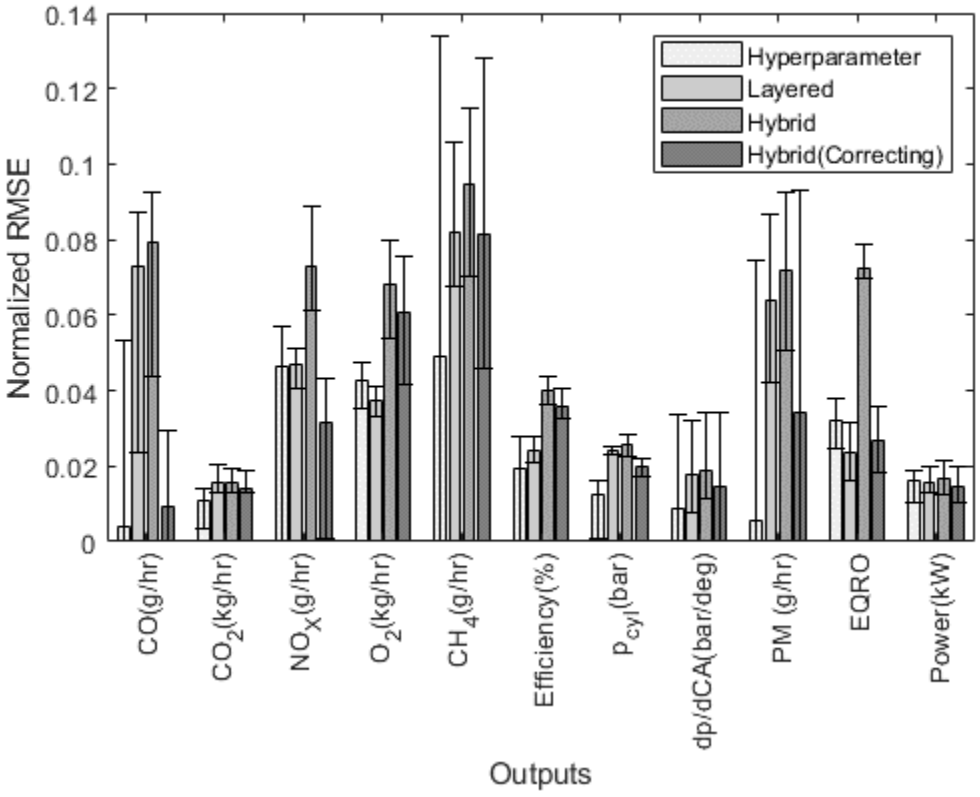


Figure E.12: Normalized RMSE for Hybrid Modeling Method (Training Data)

The results in Figure E.12 show that the hyperparameter optimized model has the overall lowest nRMSE for the training data. These results are similar to the coefficient of determination results. For the hybrid and layered models, the error correcting hybrid model shows the lowest error on average. In the case of thermal efficiency, O_2 and EQRO, the layered model performs better.

Table 7.10: Normalized RMSE results for the hybrid model (training data)

Model	Hybrid Model	Comparison		% Reduction in nRMSE *	
		Hyperparameter	Layered	Hyperparameter	Layered
CO	0.079	0.004	0.073	-1872	-8.285
CO_2	0.015	0.011	0.016	-45.31	**1.765**
NO_X	0.073	0.047	0.047	-56.21	-54.83
O_2	0.068	0.043	0.037	-58.79	-82.55
CH_4	0.095	0.049	0.082	-93.14	-15.81
Efficiency	0.040	0.019	0.024	-107.9	-64.11
p_{cyl}	0.026	0.013	0.024	-101.2	-5.136
dp/dCA	0.019	0.009	0.018	-118.0	-5.597
PM	0.072	0.006	0.064	-1157	-12.84
EQRO	0.073	0.032	0.023	-125.1	-209.7
Power	0.017	0.016	0.016	-3.281	-5.759
Average	0.052	0.023	0.039	-339.8	-42.08

* Lowest nRMSE in bold.

Table 7.10 shows a summary of the hybrid model results compared to the hyperparameter and layered model results. The hybrid model without error correction shows consistently higher nRMSE than the layered model and the hyperparameter optimized model. Only the CO_2 result shows a slight improvement of 1.8% over the layered model.

Table 7.11: Normalized RMSE results for the error correcting hybrid model (training data)

Outputs	Error Corrected Hybrid Model *	Normalized RMSE			% Reduction in nRMSE *		
		Hyper-parameter	Layered	Hybrid	Hyper-parameter	Layered	Hybrid
CO	0.009	0.004	0.073	0.079	-136.1	**87.03**	**88.03**
CO_2	0.014	0.011	0.016	0.015	-32.32	**10.55**	**8.940**
NO_X	**0.031**	0.047	0.047	0.073	**32.74**	**33.34**	**56.95**
O_2	0.061	0.043	0.037	0.068	-41.51	-62.69	**10.88**
CH_4	0.081	0.049	0.082	0.095	-65.87	**0.540**	**14.12**
Efficiency	0.036	0.019	0.024	0.040	-87.31	-47.85	**9.913**
p_{cyl}	0.020	0.013	0.024	0.026	-56.89	**18.03**	**22.04**
dp/dCA	0.014	0.009	0.018	0.019	-66.62	**19.28**	**23.56**
PM	0.034	0.006	0.064	0.072	-500.1	**46.14**	**52.27**
EQRO	0.027	0.032	0.023	0.073	**16.38**	-15.04	**62.85**
Power	**0.015**	0.016	0.016	0.017	**7.769**	**5.556**	**10.70**
Average	0.031	0.023	0.039	0.052	-84.53	**8.627**	**32.75**

* Lowest nRMSE in bold.

The results presented in Table 7.11 show that the error correcting hybrid model has a lower nRMSE than the layered and basic hybrid models. With an overall nRMSE decrease of 33% over the hybrid model and 8.7% over the layered model, the hybrid model shows better results for everything except the O_2, thermal efficiency, and EQRO. The error correcting hybrid model is even able to outperform the hyperparameter optimized model for the EQRO, gross indicated power and NO_X. The hyperparameter optimized model still shows the lowest nRMSE overall.

The hyperparameter optimized model is trained using this training data and the results presented are a direct reflection of the training process. The layered and hybrid model submodels are also trained with this data, however, the final models include many predicted inputs that come as a result of the more complex model structure. These inputs introduce error into the model that is not present if the submodels are being tested independently using the actual submodel training data inputs. Similarly, the equations introduced in the hybrid models do not perfectly represent the system. While they may help with predictions for data that wasn't used in training, the training data represents a special case where the equations introduce more error than a machine learning model with perfect knowledge of the data point. The error correcting hybrid model corrects for this by adjusting the equation output. This is the reason the error correcting hybrid model performs so well. What remains to be seen is if this carries over to the validation data results.

Appendix F Data Sets

This section lists the input and output data for the data sets collected during this research.

F.1 Box Behnken Training Data

This section lists the input and output training data for the main positive PSEP data set used in this work. Table F.1 shows the training input data and Table F.2 shows the training output data.

Table F.1: Box Behnken training input data

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
1	105.3	20.26	-18.50	0.75	-12.55	1.55	16.72	1358	102.9	24.04	80.40
2	105.4	20.60	-18.50	1.00	-4.90	1.55	16.28	1349	102.9	23.04	56.13
3	108.0	20.66	-18.50	0.75	-8.21	1.55	21.14	1198	105.8	22.26	46.76
4	103.8	18.02	-18.50	0.75	-6.90	1.55	13.59	1350	100.9	21.74	87.52
5	180.3	20.85	-27.00	0.75	-15.39	0.70	11.41	1352	176.8	22.18	71.53
6	181.3	20.68	-27.00	0.75	-15.40	2.50	11.30	1352	178.2	23.71	35.73
7	185.0	20.66	-10.00	0.75	1.62	2.50	16.79	1358	182.9	24.64	31.89
8	219.1	20.12	-18.50	0.75	-6.60	2.50	17.89	1334	216.3	25.55	90.10
9	218.9	20.39	-18.50	0.75	-6.57	0.60	14.64	1337	215.0	25.89	50.30
10	232.9	20.10	-18.50	0.75	-6.61	1.55	26.31	1333	233.0	26.32	85.20
11	208.4	20.12	-18.50	0.75	-6.58	1.55	7.49	1336	203.8	26.55	75.20
12	100.2	20.06	-18.50	0.75	-6.45	1.55	25.78	1351	99.7	25.22	80.12
13	172.9	20.41	-18.50	0.75	-7.06	0.60	1.20	1350	163.9	26.07	15.32
14	190.5	20.34	-18.50	0.75	-7.10	0.60	28.44	1345	189.6	24.82	10.16
15	195.6	20.23	-18.50	0.75	-7.11	2.50	35.93	1354	193.2	24.32	21.35
16	177.4	20.21	-18.50	0.75	-7.08	2.50	2.27	1352	170.1	25.40	24.87
17	177.2	20.25	-18.50	0.75	-7.08	1.55	16.84	1350	173.1	25.79	19.55
18	177.1	20.17	-18.50	0.75	-7.10	1.55	16.99	1346	173.1	25.86	19.08

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
19	177.3	20.24	-18.50	0.75	-7.09	1.55	16.90	1345	172.9	25.91	21.84
20	177.1	20.25	-18.50	0.75	-7.09	1.55	16.95	1346	172.9	25.90	22.11
21	177.0	20.25	-18.50	0.75	-7.09	1.55	16.90	1346	172.9	25.96	19.83
22	176.9	20.18	-18.50	0.75	-7.10	1.55	17.00	1345	173.0	25.94	22.00
23	177.2	20.17	-18.50	0.75	-7.10	1.55	16.80	1345	173.0	25.93	26.61
24	177.1	20.23	-18.50	0.75	-7.10	1.55	16.78	1344	173.0	25.87	25.27
25	177.1	20.23	-18.50	0.75	-7.10	1.55	16.66	1345	173.0	25.72	24.28
26	112.6	19.71	-18.50	0.75	-5.73	2.50	16.88	1385	108.9	24.80	66.61
27	113.2	19.73	-18.50	0.75	-4.12	2.50	18.17	1365	108.7	24.76	68.77
28	181.3	20.66	-18.50	0.75	-6.63	2.30	25.50	1363	176.0	24.68	67.52
29	176.2	20.58	-10.00	0.75	1.94	0.75	15.57	1359	172.0	24.98	73.25
30	182.5	20.37	-27.00	0.75	-15.13	1.55	29.75	1350	181.9	24.41	50.32
31	175.8	20.41	-18.50	0.75	-12.32	1.05	15.39	1350	171.8	25.42	74.92
32	175.9	20.48	-18.50	0.75	-0.06	0.80	15.89	1358	171.9	25.69	84.69
33	176.0	20.40	-18.50	0.65	-0.98	1.05	16.07	1350	172.0	25.40	90.15
34	173.0	20.31	-18.50	0.75	-0.15	1.55	0.39	1352	161.1	25.26	81.62
35	181.4	20.31	-18.50	0.75	-12.31	1.55	25.70	1350	179.7	24.77	85.23
36	225.8	20.32	-18.50	0.75	-12.33	1.55	23.03	1345	223.2	25.35	55.89
37	176.5	20.29	-27.00	0.75	-8.65	1.55	15.30	1352	171.9	25.50	78.55
38	176.2	20.26	-10.00	0.75	5.92	1.55	15.90	1351	172.0	25.29	84.60
39	177.1	20.26	-10.00	0.75	-3.81	1.55	17.71	1351	172.9	25.77	75.98
40	176.9	20.29	-10.00	1.00	3.89	1.55	17.49	1351	172.8	26.12	87.36
41	177.1	20.24	-27.00	1.00	-13.10	1.55	17.39	1353	172.9	26.47	78.69
42	181.2	20.23	-18.50	0.75	-5.31	1.55	30.00	1503	181.0	25.48	49.36
43	169.9	20.26	-18.50	0.75	-5.29	1.55	0.42	1504	159.4	24.85	66.11
44	187.7	17.99	-18.50	0.75	-6.63	1.55	0.42	1351	179.8	24.97	87.39
45	182.1	17.97	-18.50	0.75	-7.32	1.55	17.40	1204	177.9	27.36	27.24
46	175.8	17.87	-18.50	0.75	-5.64	1.55	17.12	1479	172.0	27.50	27.85

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
47	175.0	23.09	-18.50	0.75	-5.61	1.55	16.85	1482	171.9	27.62	32.99
48	175.5	23.15	-18.50	0.75	-8.08	1.55	16.22	1194	168.9	27.30	42.84
49	174.7	23.08	-18.50	0.50	-8.72	1.55	12.53	1360	169.9	27.19	29.21
50	174.7	23.02	-18.50	1.00	-4.63	1.55	12.48	1360	170.0	27.61	30.25
51	174.7	23.11	-18.50	0.75	-12.38	1.55	12.74	1360	170.1	28.35	31.46
52	174.8	23.12	-18.50	0.75	-0.15	1.55	12.85	1361	169.9	28.58	33.74
53	174.8	23.16	-27.00	0.75	-15.15	1.55	12.78	1361	169.9	28.63	34.38
54	174.6	23.14	-10.00	0.75	1.85	1.55	12.98	1361	170.0	28.74	44.93
55	107.5	23.11	-18.50	0.75	-6.59	1.55	16.86	1370	103.9	28.57	46.68
56	224.9	23.15	-18.50	0.75	-6.69	1.55	20.78	1358	222.2	28.54	34.45
57	173.5	23.02	-18.50	0.75	-6.70	2.20	20.30	1357	170.8	27.85	68.03
58	169.8	23.06	-18.50	0.75	-6.70	1.55	0.34	1357	159.9	28.39	36.10
59	180.6	23.05	-18.50	0.75	-6.69	1.55	30.31	1358	180.3	27.98	40.81
60	175.1	17.79	-18.50	0.50	-9.03	1.55	16.73	1350	171.8	27.87	78.05
61	175.0	17.81	-18.50	1.00	-4.96	1.55	16.88	1352	171.9	28.02	53.76
62	175.0	17.82	-18.50	0.75	-12.67	1.55	16.54	1352	171.9	27.78	45.06
63	175.0	17.73	-18.50	0.75	-0.50	1.55	16.79	1353	171.9	27.44	59.90
64	175.0	17.74	-27.00	0.75	-15.50	1.55	16.19	1352	171.6	27.42	52.88
65	175.5	17.76	-10.00	0.75	1.54	1.55	17.24	1355	172.4	27.35	65.55
66	105.4	17.80	-18.50	0.75	-6.89	1.55	12.11	1364	102.8	26.88	71.73
67	218.2	17.75	-18.50	0.75	-7.15	1.55	14.21	1334	214.5	27.75	45.65
68	177.5	17.91	-18.50	0.75	-7.23	0.60	14.65	1324	173.9	28.18	31.84
69	178.9	17.66	-18.50	0.75	-6.91	2.50	10.62	1362	175.0	28.15	25.26
70	172.6	17.79	-18.50	0.75	-6.97	1.55	30.48	1355	173.9	27.07	34.66
71	174.2	20.51	-18.50	0.51	-9.66	1.55	15.42	1204	170.8	27.10	55.50
72	176.8	20.43	-18.50	1.00	-6.09	1.55	15.11	1207	174.6	26.56	99.85
73	175.5	20.95	-18.50	0.75	-13.24	1.55	13.37	1207	171.8	26.64	85.69
74	175.5	20.94	-18.50	0.75	-2.39	1.55	13.23	1206	171.9	26.36	55.44

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
75	174.3	20.87	-18.50	0.75	1.52	1.55	19.44	1502	171.9	26.55	23.35
76	174.3	20.84	-18.50	0.75	-11.98	1.55	19.38	1496	172.0	26.80	30.29
77	172.4	20.91	-18.50	0.50	-7.93	1.55	16.72	1501	169.2	26.99	46.02
78	172.2	20.86	-18.50	1.00	-3.44	1.55	17.20	1504	169.6	26.92	45.51
79	171.9	20.81	-27.00	0.75	-14.19	1.55	17.71	1496	169.1	26.87	43.93
80	174.1	20.99	-27.00	0.75	-16.71	1.55	12.81	1200	173.7	26.96	49.44
81	172.9	20.92	-10.00	0.75	0.24	1.55	13.40	1199	171.4	27.05	37.99
82	171.5	20.81	-10.00	0.75	2.83	1.55	19.10	1499	169.4	26.98	41.94
83	106.3	20.71	-18.50	0.75	-8.10	1.55	21.54	1207	103.0	23.66	23.93
84	104.5	20.66	-18.50	0.75	-5.54	1.55	25.74	1497	101.2	22.76	19.42
85	226.1	20.60	-18.50	0.75	-5.69	1.55	26.42	1485	225.2	22.57	21.36
86	175.1	20.79	-18.50	0.75	-8.03	0.60	11.79	1213	169.6	24.11	19.71
87	174.0	20.77	-18.50	0.75	-5.48	0.60	15.80	1508	169.6	24.47	9.92
88	176.2	20.52	-18.50	0.75	-5.45	2.50	19.23	1512	174.3	24.69	25.42
89	175.1	20.67	-18.50	0.75	-8.18	2.50	10.88	1188	169.2	25.20	26.73
90	175.9	20.32	-18.50	0.50	-14.21	1.55	14.54	1359	170.6	25.66	32.49
91	176.2	20.32	-18.50	0.50	-1.99	1.55	14.83	1358	171.2	25.84	26.48
92	177.1	20.22	-18.50	1.00	2.09	1.55	15.50	1359	172.4	25.90	38.54
93	176.2	20.25	-18.50	1.00	-10.13	1.55	15.22	1359	171.3	25.96	23.74
94	174.6	20.44	-27.00	0.50	-17.08	1.55	13.23	1348	169.3	25.39	38.51
95	176.2	20.44	-10.00	0.50	-0.08	1.55	14.77	1348	171.1	25.90	85.96
96	106.9	20.55	-18.50	0.50	-8.45	1.55	16.18	1360	102.9	26.19	64.74
97	107.0	20.42	-18.50	1.00	-4.35	1.55	16.25	1362	102.9	25.68	92.63
98	222.1	20.42	-18.50	1.00	-4.51	1.55	18.36	1347	217.7	24.92	91.89
99	219.6	20.47	-18.50	0.50	-8.55	1.55	15.99	1347	214.7	25.36	84.56
100	177.0	20.55	-18.50	0.50	-8.49	0.60	17.47	1354	172.9	26.14	29.07
101	176.9	20.57	-18.50	1.00	-4.41	0.60	17.08	1356	172.8	26.25	28.81
102	180.2	20.22	-18.50	1.00	-4.50	2.49	21.81	1348	177.6	26.66	33.47

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
103	177.9	20.34	-18.50	0.50	-8.53	2.49	14.13	1350	172.9	25.70	45.13
104	170.6	20.40	-18.50	1.00	-4.49	1.55	0.34	1350	159.0	25.83	76.54
105	179.2	20.38	-18.50	1.00	-4.49	1.55	29.28	1349	178.0	25.03	85.78
106	179.0	20.32	-18.50	0.50	-8.54	1.55	29.24	1348	178.0	24.85	76.28
107	177.2	20.31	-27.00	0.75	-20.66	1.55	13.49	1351	171.9	24.44	75.46
108	177.0	20.13	-18.50	0.75	0.02	2.50	11.48	1353	172.0	26.54	24.06
109	178.2	20.22	-18.50	0.75	-12.18	2.50	13.22	1347	173.7	26.77	34.65
110	107.3	20.25	-18.50	0.75	-12.20	1.55	16.87	1364	102.9	25.62	64.11
111	107.5	20.32	-18.50	0.75	0.08	1.55	16.87	1365	102.9	25.52	86.26
112	225.6	20.24	-18.50	0.75	-0.14	1.55	21.46	1349	221.9	25.81	88.11
113	147.5	20.27	-18.50	0.75	-12.22	1.55	0.49	1361	124.4	25.83	83.76
114	176.6	20.21	-18.50	0.75	-0.03	1.55	26.78	1357	174.8	25.14	75.90
115	106.6	20.76	-27.00	0.75	-15.05	1.55	14.70	1351	102.2	26.37	77.25
116	106.1	20.75	-10.00	0.75	1.97	1.55	14.17	1353	101.6	25.26	96.20
117	221.2	20.76	-10.00	0.75	1.92	1.55	15.33	1347	216.2	24.49	92.52
118	221.0	20.71	-27.00	0.75	-15.08	1.55	14.74	1346	216.0	25.23	87.26
119	176.5	20.69	-10.00	0.75	1.98	1.55	26.78	1353	174.5	25.85	90.54
120	181.2	20.71	-10.00	0.75	1.98	1.55	0.38	1353	172.0	25.80	78.52

Table F.2: Box Behnken training output data

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
1	106.9	14.6	51.9	14.0	11.2	0.4	96.4	4.1	0.8	0.5	31.1
2	23.3	15.0	35.5	13.8	8.6	0.4	73.2	2.1	0.2	0.5	30.4
3	41.3	13.1	36.7	13.6	9.6	0.4	95.2	2.7	0.3	0.5	28.0
4	18.7	12.4	44.1	18.4	8.3	0.4	72.8	2.1	0.1	0.5	26.2
5	10.4	5.1	89.1	43.0	6.1	0.5	115.8	4.3	0.1	0.1	11.4
6	403.0	21.8	95.9	18.9	13.2	0.5	123.2	4.3	6.1	0.6	45.5
7	64.0	23.6	28.0	14.4	4.9	0.4	95.8	3.0	0.5	0.6	40.3
8	76.5	22.7	33.5	20.4	9.8	0.4	113.1	3.4	0.9	0.6	44.2
9	11.6	4.3	34.4	48.8	8.1	0.5	112.0	3.4	0.1	0.1	8.9
10	30.3	14.9	29.7	28.8	17.4	0.5	117.4	3.6	0.3	0.3	32.0
11	14.3	14.9	78.7	36.9	8.5	0.5	109.2	3.3	0.1	0.3	32.4
12	67.0	14.7	15.6	10.1	14.9	0.4	70.8	2.1	0.3	0.6	28.8
13	12.4	4.8	44.9	54.0	9.6	0.5	94.8	2.8	0.1	0.1	10.2
14	14.0	4.2	27.2	36.8	11.0	0.4	100.8	3.0	0.1	0.1	8.6
15	919.2	21.5	3.6	8.0	108.6	0.4	77.2	3.1	4.6	0.7	39.8
16	68.2	23.2	99.1	22.0	5.4	0.4	80.3	2.9	0.4	0.6	43.9
17	28.4	14.9	45.9	26.1	12.9	0.5	97.3	2.9	0.3	0.4	31.8
18	26.4	14.6	46.4	26.5	12.9	0.5	97.4	2.9	0.3	0.4	31.5
19	26.3	14.7	46.3	26.3	13.0	0.5	97.4	2.9	0.3	0.4	31.5
20	26.5	14.8	46.7	26.4	13.0	0.5	97.4	2.9	0.3	0.4	31.5
21	27.2	14.7	46.4	26.3	12.8	0.5	97.3	2.9	0.3	0.4	31.6
22	26.6	14.6	46.4	26.4	12.8	0.5	97.3	2.9	0.3	0.4	31.5
23	26.0	14.6	46.9	26.7	12.8	0.5	97.3	2.9	0.3	0.4	31.1
24	25.6	14.6	47.2	26.6	12.8	0.5	97.4	2.9	0.2	0.4	31.3
25	26.5	14.6	47.5	26.6	12.6	0.5	97.4	2.9	0.3	0.4	31.3
26	260.0	21.8	21.1	5.3	3.1	0.4	69.8	2.2	2.2	0.8	38.3
27	275.9	21.6	17.9	4.8	1.8	0.4	72.2	2.2	0.8	0.8	37.2

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
28	244.4	21.5	14.7	11.9	20.9	0.4	73.9	2.9	1.9	0.6	40.9
29	31.6	4.7	18.3	41.1	30.6	0.4	92.0	2.9	0.2	0.1	9.4
30	68.8	14.4	40.8	20.6	19.7	0.5	126.1	4.7	0.8	0.4	32.5
31	22.0	10.9	81.7	32.2	12.6	0.5	114.9	3.5	0.1	0.3	24.8
32	28.5	5.7	27.4	39.5	24.2	0.4	95.5	2.9	0.0	0.1	12.5
33	38.2	9.6	35.5	33.5	25.1	0.4	95.9	2.9	0.0	0.3	21.5
34	17.0	14.6	85.4	49.2	8.9	0.4	95.8	2.9	0.1	0.3	29.6
35	56.9	14.9	41.4	22.1	20.5	0.5	115.3	2.9	0.9	0.4	32.5
36	29.5	14.5	55.1	30.8	17.0	0.5	128.6	3.9	0.5	0.3	32.8
37	21.7	14.7	58.8	26.9	10.6	0.5	102.6	4.1	0.1	0.4	31.9
38	94.1	13.4	26.9	28.4	45.8	0.4	92.8	2.9	0.2	0.4	25.5
39	58.8	14.3	39.0	26.8	28.2	0.4	93.3	2.9	0.2	0.4	30.1
40	65.3	13.9	27.7	27.2	29.5	0.4	93.2	2.9	0.4	0.4	27.0
41	28.0	13.9	72.7	24.7	11.4	0.5	119.0	4.0	0.4	0.4	32.6
42	75.9	15.3	13.7	20.9	42.8	0.4	92.6	2.8	0.5	0.4	31.7
43	14.5	15.3	82.9	44.1	8.8	0.4	90.0	2.7	0.1	0.3	33.1
44	12.6	12.6	81.0	43.8	7.9	0.5	99.9	3.0	0.0	0.3	26.8
45	15.8	11.7	50.3	29.5	10.3	0.5	106.5	3.2	0.1	0.3	25.5
46	32.0	13.7	32.8	29.1	17.5	0.4	90.9	2.7	0.1	0.4	29.3
47	28.4	18.8	41.7	22.2	15.0	0.4	92.3	2.7	0.1	0.5	38.4
48	19.6	15.6	75.6	23.8	11.1	0.5	123.1	4.7	0.1	0.4	33.4
49	22.7	16.7	77.0	24.8	11.8	0.5	108.0	3.1	0.0	0.4	36.1
50	15.3	18.0	64.5	23.7	9.9	0.4	96.7	2.9	0.1	0.5	36.6
51	51.0	17.8	102.5	23.6	12.6	0.4	125.5	5.2	0.6	0.5	38.5
52	25.9	17.3	52.2	24.3	11.1	0.4	96.4	2.8	0.0	0.5	34.2
53	55.1	17.6	138.3	23.7	12.6	0.5	137.0	6.3	0.3	0.5	38.3
54	40.4	17.2	48.2	24.4	13.6	0.4	92.4	2.8	0.1	0.5	32.9
55	69.9	17.4	33.9	10.9	8.8	0.4	81.6	2.1	0.5	0.6	33.8

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
56	30.6	17.7	44.3	27.6	15.9	0.5	114.5	3.4	0.4	0.4	37.9
57	442.9	24.0	22.0	9.3	12.1	0.4	97.0	2.9	4.1	0.8	44.6
58	14.1	18.4	141.4	38.1	7.1	0.4	102.4	2.9	0.1	0.4	38.0
59	122.4	17.6	14.8	15.1	26.3	0.5	98.2	2.9	1.2	0.5	36.3
60	43.5	11.8	44.3	29.8	23.1	0.5	94.4	2.9	0.0	0.3	26.4
61	20.6	12.9	36.6	28.4	11.2	0.5	95.6	2.9	0.1	0.3	27.4
62	25.5	12.5	62.4	29.0	12.0	0.5	109.2	2.9	0.2	0.3	28.2
63	44.2	12.4	30.8	29.3	22.8	0.4	95.2	2.9	0.0	0.3	25.3
64	15.1	12.1	85.6	29.4	10.1	0.5	118.7	4.1	0.1	0.3	28.0
65	68.8	12.1	26.7	29.7	37.0	0.4	92.4	2.9	0.2	0.3	24.2
66	19.4	12.7	43.3	18.9	8.2	0.4	72.2	2.1	0.1	0.4	26.5
67	18.4	12.2	45.2	38.0	10.5	0.5	111.7	3.5	0.1	0.3	26.6
68	10.1	3.0	24.7	43.0	7.7	0.5	97.4	2.9	0.1	0.1	5.9
69	33.7	19.9	50.0	22.6	8.0	0.4	97.4	2.9	0.3	0.5	39.7
70	45.8	12.4	16.4	21.5	23.6	0.5	94.3	2.8	0.3	0.4	26.7
71	23.3	13.2	75.5	26.7	10.5	0.5	119.2	3.7	0.1	0.3	29.1
72	16.9	14.0	61.9	25.9	9.2	0.5	106.7	3.4	0.2	0.4	29.9
73	28.5	14.0	127.3	27.1	9.8	0.5	137.9	6.1	0.2	0.4	31.6
74	13.9	14.1	57.7	27.0	8.6	0.4	104.1	3.1	0.0	0.4	29.9
75	108.7	16.1	23.8	25.2	45.6	0.4	91.0	2.8	0.1	0.4	31.0
76	72.2	17.2	46.4	23.5	18.9	0.4	103.5	2.8	1.1	0.5	37.2
77	50.9	15.9	41.2	25.8	22.4	0.4	89.9	2.7	0.1	0.4	34.4
78	31.4	17.2	33.9	24.5	14.2	0.4	90.5	2.7	0.3	0.4	34.9
79	67.9	16.9	63.5	24.1	14.4	0.5	111.9	4.0	0.8	0.4	37.0
80	16.3	13.7	199.9	26.4	9.3	0.5	152.1	7.7	0.1	0.4	31.3
81	23.3	13.8	51.8	26.7	10.8	0.4	99.3	3.1	0.1	0.4	28.5
82	159.8	15.6	20.7	25.2	64.8	0.4	87.4	2.7	0.5	0.4	29.1
83	42.9	13.4	34.3	12.9	9.8	0.4	94.3	2.8	0.3	0.5	28.4

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
84	78.9	15.9	12.3	11.1	21.3	0.4	68.3	2.0	0.2	0.6	31.0
85	41.9	16.6	23.1	27.9	22.4	0.4	108.6	3.3	0.3	0.4	35.3
86	10.0	4.0	45.8	41.8	6.7	0.5	106.7	3.1	0.0	0.1	8.4
87	16.1	4.4	26.9	42.0	10.5	0.4	91.1	2.7	0.2	0.1	9.1
88	145.0	26.0	19.6	11.5	10.2	0.4	92.5	2.8	0.7	0.8	46.2
89	138.6	20.4	67.8	18.2	8.0	0.4	115.8	3.1	1.0	0.6	40.6
90	36.1	14.4	93.8	27.1	13.9	0.5	121.2	4.2	0.1	0.4	32.3
91	98.1	13.8	41.0	28.2	66.4	0.4	94.7	2.9	0.0	0.4	28.1
92	44.0	15.1	36.2	26.5	18.0	0.4	96.6	2.9	0.0	0.4	29.4
93	37.9	15.6	63.5	25.6	11.5	0.5	107.3	3.0	0.8	0.4	33.5
94	25.0	14.4	150.3	27.5	11.3	0.5	134.9	5.9	0.1	0.4	32.8
95	77.4	14.2	40.3	27.3	35.2	0.4	93.3	2.9	0.0	0.4	28.6
96	41.5	13.8	42.2	15.5	13.4	0.4	80.5	2.9	0.0	0.5	29.3
97	21.9	15.1	33.3	14.3	8.9	0.4	72.6	2.1	0.1	0.6	30.1
98	23.0	15.6	41.4	31.5	12.5	0.4	113.2	3.5	0.2	0.4	32.5
99	29.4	14.6	56.2	33.6	14.6	0.5	112.2	3.4	0.0	0.3	32.0
100	22.1	4.0	31.9	40.2	16.6	0.5	95.5	2.9	0.0	0.1	9.1
101	12.9	4.4	30.4	40.0	7.9	0.5	96.4	2.8	0.0	0.1	9.0
102	206.1	23.2	17.5	11.2	11.2	0.4	98.2	2.9	1.5	0.7	43.2
103	179.1	22.0	39.7	16.4	8.7	0.4	99.6	2.9	0.7	0.6	43.1
104	12.1	15.7	105.8	44.2	6.5	0.5	95.8	2.9	0.1	0.3	32.1
105	43.6	15.0	18.4	19.8	20.4	0.4	97.6	2.9	0.3	0.4	31.2
106	75.6	13.9	21.2	20.8	32.0	0.4	96.6	2.9	0.0	0.4	30.5
107	18.5	14.7	224.4	27.6	11.6	0.5	149.8	7.1	0.1	0.4	33.7
108	34.2	23.3	43.8	16.8	5.3	0.4	97.9	2.9	0.1	0.6	42.4
109	364.5	22.3	51.6	16.7	11.8	0.4	116.9	3.0	3.1	0.6	45.3
110	72.5	14.9	49.0	14.0	11.7	0.4	95.0	3.6	0.4	0.6	31.5
111	43.3	14.3	28.3	14.9	12.1	0.4	72.3	2.1	0.0	0.5	27.9

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
112	47.9	15.0	29.2	30.9	24.7	0.4	113.6	3.5	0.1	0.4	30.5
113	24.0	15.5	161.4	54.3	7.4	0.5	109.3	4.6	0.3	0.3	33.4
114	80.3	14.6	17.6	21.0	36.4	0.4	96.0	2.9	0.0	0.4	29.1
115	83.9	15.3	77.8	14.0	10.2	0.4	107.0	4.6	0.3	0.6	32.2
116	49.0	14.8	34.7	15.2	10.8	0.4	69.1	2.1	0.1	0.6	27.3
117	58.8	15.0	37.4	34.2	27.8	0.4	109.1	3.4	0.1	0.3	30.2
118	15.4	15.0	118.2	34.2	11.3	0.5	141.6	5.6	0.1	0.3	34.5
119	158.6	14.4	13.7	20.7	75.7	0.4	93.0	2.9	0.4	0.4	26.9
120	27.2	15.3	80.5	44.0	10.4	0.4	95.5	3.0	0.0	0.3	30.0

F.2 Box Behnken Validation Data

This section shows the validation data set used to validate the positive PSEP Box Behnken based models. Table F.3 shows the validation input data and Table F.4 shows the validation output data.

Table F.3: Box Behnken validation input data

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
1	191.7	17.98	-26	0.5	-14.17	1.25	14.13	1300	186.2	21.24	34.84
2	151.3	17.85	-10.5	0.8	-2.97	1.2	18.30	1366	147.1	21.86	34.44
3	190.6	17.86	-17.5	0.5	-3.98	1.45	19.20	1392	186.6	21.31	37.16
4	114.0	17.85	-18.5	0.6	-10.41	2	12.81	1470	109.5	21.52	49.84
5	181.9	18.11	-26.5	0.6	-9.69	2	0.34	1459	171.0	20.90	49.20
6	178.1	17.82	-22	0.6	-12.47	1.4	4.03	1200	170.8	21.49	34.68
7	108.2	17.67	-23	0.7	-6.46	2.35	18.44	1300	104.8	22.02	46.68
8	112.1	17.62	-26	0.8	-11.01	2.2	17.03	1300	107.6	22.30	45.46
9	128.7	17.65	-18	0.5	-6.09	1.3	0.42	1397	121.4	20.61	41.43

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
10	169.5	17.95	-24.5	0.7	-18.52	1.05	10.24	1446	163.6	21.17	36.32
11	169.3	18.60	-11.5	0.6	0.59	1.2	1.78	1193	162.3	21.14	37.95
12	134.2	18.37	-17	0.9	-4.43	1.9	21.48	1406	131.1	21.37	49.78
13	166.9	18.43	-27	0.8	-15.90	1.8	19.16	1436	163.0	20.65	49.46
14	166.0	18.37	-20	0.9	-4.46	1.7	19.00	1447	162.3	20.64	45.57
15	108.2	18.51	-25	0.5	-20.35	1.3	14.08	1318	101.3	20.96	37.42
16	171.5	18.36	-11.5	0.6	1.72	1.9	0.46	1307	156.1	21.15	43.44
17	109.5	18.55	-23	0.7	-14.40	1.25	24.89	1315	107.5	21.37	36.03
18	215.8	18.83	-16	0.5	-11.90	2.1	14.28	1395	211.3	20.12	57.57
19	158.1	19.70	-14	0.6	-9.54	1.75	0.33	1262	151.0	21.19	45.32
20	157.0	19.78	-18	0.9	-8.29	1.25	11.98	1255	152.0	21.80	37.68
21	185.6	19.63	-16	0.6	-7.56	2.2	0.32	1291	175.1	21.21	51.95
22	185.1	19.91	-23.5	0.9	-8.44	1.6	18.10	1297	181.7	21.30	42.60
23	219.6	19.88	-20	0.8	-3.67	1.4	19.19	1335	216.1	21.56	40.72
24	224.1	19.91	-23.5	0.6	-16.56	1.6	20.06	1429	221.1	22.15	48.04
25	207.3	19.90	-19.5	0.7	-9.98	1.2	7.18	1433	201.5	21.86	47.55
26	230.2	19.85	-21	0.8	-4.63	1.6	28.96	1236	231.0	21.33	39.72
27	152.5	19.84	-16	0.8	-9.14	1.85	25.27	1417	152.3	22.33	58.94
28	209.5	20.46	-27	0.6	-19.49	2.1	8.25	1246	204.1	21.40	47.80
29	121.6	20.56	-16.5	0.6	-9.39	0.75	16.86	1304	117.8	22.30	11.94
30	212.7	20.33	-21	0.5	-11.71	1.7	12.26	1397	208.2	21.81	64.01
31	209.8	20.28	-14	0.9	-0.95	1.7	12.23	1441	205.3	22.58	43.76
32	210.7	20.34	-14	0.9	-0.96	1.7	12.70	1440	206.5	22.65	51.12
33	209.7	21.03	-19.5	0.7	-7.99	2.05	6.08	1193	201.5	26.60	44.82
34	134.3	21.25	-12.5	0.8	-3.36	0.6	21.99	1262	131.8	26.88	9.50
35	120.2	21.03	-21	0.9	-13.40	1.25	2.93	1259	114.5	26.58	32.17
36	205.1	21.02	-24	0.6	-9.35	1	15.32	1431	201.4	27.08	50.06
37	161.6	21.51	-14	0.8	-5.39	1.05	11.77	1303	157.1	25.07	36.47

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
38	191.1	21.41	-23	0.8	-9.72	1.4	13.22	1301	186.6	25.52	40.14
39	168.3	21.41	-24	0.9	-10.37	2.45	17.61	1418	165.4	26.10	78.75

Table F.4: Box Behnken validation output data

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
1	57.77	9.20	74.81	38.24	79.46	0.45	116.94	3.85	0.03	0.24	21.19
2	48.17	10.20	28.60	29.93	46.86	0.42	82.20	2.58	0.17	0.29	20.89
3	95.55	10.46	26.15	35.62	193.46	0.42	96.15	3.00	0.02	0.26	22.25
4	55.84	15.91	44.74	18.33	18.44	0.41	76.11	2.52	0.18	0.54	32.61
5	20.41	16.75	95.69	44.94	17.81	0.43	95.81	3.92	0.25	0.34	34.93
6	10.55	9.32	114.16	40.96	13.58	0.49	121.51	4.46	0.03	0.23	21.70
7	105.00	17.27	25.11	11.39	16.41	0.39	75.31	2.91	0.08	0.63	31.84
8	209.37	15.88	38.50	13.96	17.62	0.40	89.06	3.68	0.93	0.59	31.74
9	43.85	9.81	62.65	35.35	50.02	0.45	76.49	2.35	0.02	0.28	21.15
10	15.76	8.67	117.44	39.91	15.64	0.49	116.40	4.59	0.05	0.21	19.98
11	23.58	8.64	51.08	40.58	24.99	0.42	98.09	3.46	0.03	0.24	18.31
12	32.65	15.46	23.17	18.17	21.27	0.38	78.62	2.38	0.29	0.52	30.43
13	43.31	14.67	64.21	25.80	19.04	0.43	94.51	5.19	0.89	0.42	32.47
14	25.35	14.46	29.28	26.61	20.29	0.41	86.85	2.69	0.17	0.40	29.47
15	31.09	9.37	164.76	23.52	17.27	0.49	101.69	6.16	0.03	0.31	21.24
16	30.47	13.95	72.22	52.32	24.19	0.41	94.08	2.95	0.10	0.27	27.24
17	23.38	10.19	48.93	19.30	17.44	0.44	97.45	3.37	0.03	0.37	21.93
18	38.55	16.37	58.89	33.85	23.17	0.45	112.91	3.66	0.26	0.37	35.55
19	18.72	13.90	134.77	33.55	11.63	0.45	107.52	5.44	0.11	0.37	30.16
20	13.54	11.22	71.17	30.61	12.10	0.45	101.68	2.96	0.14	0.31	24.46
21	17.19	17.88	118.09	42.43	11.21	0.44	103.60	3.77	0.12	0.36	36.43

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
22	19.77	13.87	54.68	29.15	14.39	0.44	107.32	3.77	0.25	0.36	29.87
23	22.79	12.68	35.84	36.13	20.62	0.42	112.42	3.54	0.12	0.30	26.83
24	21.28	13.69	81.31	35.58	20.58	0.48	133.87	7.68	0.21	0.31	31.94
25	15.47	11.89	73.06	44.18	14.88	0.46	107.38	3.70	0.15	0.27	26.70
26	24.45	12.95	25.63	30.04	22.46	0.42	122.75	3.94	0.19	0.32	27.62
27	95.92	16.40	23.00	17.60	27.88	0.41	85.53	2.65	0.95	0.51	33.43
28	18.10	15.56	287.80	35.44	13.58	0.47	167.15	8.81	0.18	0.37	35.61
29	17.61	4.24	39.24	33.35	15.46	0.46	84.06	2.87	0.01	0.12	9.05
30	22.07	14.96	76.81	36.50	15.88	0.46	116.54	5.09	0.14	0.34	33.41
31	28.61	15.93	41.96	36.07	19.80	0.41	102.37	3.50	0.23	0.36	32.03
32	29.00	16.01	41.21	35.60	19.93	0.41	102.86	3.38	0.24	0.36	32.13
33	18.29	16.33	101.64	34.42	11.31	0.42	122.37	4.62	0.17	0.41	34.88
34	17.27	4.00	21.36	33.03	18.22	0.46	82.51	2.65	0.05	0.10	7.88
35	11.89	12.04	175.44	26.59	9.36	0.45	113.39	4.46	0.04	0.38	26.16
36	21.98	10.46	51.16	40.32	20.52	0.44	105.35	3.35	0.02	0.25	23.56
37	18.14	11.24	57.34	31.94	13.41	0.45	91.78	3.30	0.07	0.30	24.22
38	14.39	13.64	81.62	32.83	11.56	0.44	116.24	6.08	0.10	0.35	29.91
39	402.55	23.28	34.58	13.72	13.10	0.37	101.80	3.81	5.85	0.71	44.28

F.3 Negative PSEP Training Data

This section presents the negative PSEP training data for the multi-region model. Table F.5 shows the input data and Table F.6 shows the output data.

Table F.5: Negative PSEP training input data

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
1	153.6	20.47	-16.50	0.50	-20.38	1.89	11.09	1293	165.9	25.39	30.80
2	210.3	20.44	-16.50	0.50	-20.38	1.89	19.34	1290	204.6	25.62	58.20
3	194.5	20.42	-18.50	0.50	-19.28	1.89	6.23	1291	187.9	24.99	44.60
4	141.1	20.45	-18.50	0.50	-19.27	1.89	14.57	1295	135.7	24.77	84.52
5	158.3	20.09	-18.50	0.50	-18.50	1.21	20.94	1302	161.8	24.98	74.56
6	203.7	20.10	-18.50	0.50	-18.50	1.21	11.72	1297	201.4	24.55	86.24
7	154.2	20.06	-16.50	0.64	-19.30	1.21	12.59	1303	152.2	25.16	89.12
8	208.4	20.04	-16.50	0.64	-19.30	1.21	17.99	1296	211.3	25.61	90.15
9	152.3	20.14	-21.50	0.50	-25.71	1.21	12.35	1403	171.3	24.84	12.46
10	213.1	20.09	-19.50	0.50	-21.18	1.21	20.30	1396	221.2	25.96	15.31
11	201.4	20.12	-16.50	0.50	-19.01	1.21	11.38	1396	215.5	25.42	16.47
12	157.7	20.56	-19.50	0.50	-21.83	1.21	10.69	1296	171.9	27.16	26.87
13	204.4	20.83	-18.50	0.50	-20.06	1.21	14.55	1292	240.9	27.69	41.57
14	153.0	20.65	-19.50	0.64	-22.30	1.89	10.16	1297	166.8	26.84	48.27
15	158.6	20.61	-16.50	0.64	-16.97	1.80	16.77	1296	195.6	26.79	79.26
16	204.5	20.65	-16.50	0.64	-16.97	1.80	8.25	1293	211.2	26.53	75.45
17	218.9	20.95	-17.00	0.60	-14.57	1.55	12.60	1333	240.9	23.81	15.71
18	182.6	21.11	-17.00	0.60	-14.57	0.60	15.86	1349	211.4	24.92	15.82
19	183.9	20.88	-17.00	0.60	-14.57	2.50	16.90	1350	226.9	25.41	25.31
20	164.2	21.10	-17.00	0.60	-14.57	1.55	0.35	1352	122.6	25.63	16.81
21	181.1	20.93	-17.00	0.60	-14.57	1.55	21.49	1344	243.9	24.99	15.56
22	176.2	21.01	-17.00	0.60	-14.57	1.55	16.50	1345	213.3	25.01	21.03
23	180.5	20.88	-17.00	0.60	-14.57	1.55	13.94	1345	207.4	24.99	28.44
24	180.8	21.21	-18.00	0.60	-15.57	1.55	14.98	1345	212.2	24.97	16.39
25	180.7	21.16	-17.00	0.60	-13.76	1.55	15.35	1345	212.4	24.93	25.62
26	181.3	21.05	-17.00	0.60	-16.19	1.55	14.85	1345	212.3	24.67	17.52
27	182.6	20.80	-17.00	0.60	-14.57	1.55	13.06	1351	204.4	22.68	16.80

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
28	182.5	20.81	-17.00	0.60	-14.57	1.55	13.05	1351	204.4	22.77	16.01
29	182.4	20.82	-17.00	0.60	-14.57	1.55	13.06	1351	204.5	22.98	15.26
30	182.9	20.80	-17.00	0.60	-14.57	1.55	15.22	1350	214.3	23.29	17.05
31	183.7	20.74	-17.00	0.60	-14.57	1.55	14.91	1350	214.2	23.46	17.86
32	181.4	20.88	-17.00	0.60	-14.57	1.55	11.67	1352	197.8	24.95	32.97
33	183.7	20.90	-17.00	0.60	-14.57	1.55	14.59	1351	210.4	25.13	33.01
34	183.3	20.85	-17.00	0.60	-14.57	1.55	14.58	1351	210.0	25.20	27.07
35	182.0	20.82	-17.00	0.60	-14.57	1.55	14.24	1351	208.9	25.22	37.23
36	184.5	20.84	-17.00	0.60	-14.57	1.55	14.05	1352	211.0	25.24	31.12
37	160.6	21.18	-21.00	0.50	-21.00	1.89	19.29	1404	226.1	24.94	22.56
38	206.5	21.11	-21.00	0.50	-21.00	1.89	10.22	1400	226.3	24.79	18.00
39	151.2	21.12	-16.50	0.50	-16.50	1.89	10.11	1408	166.9	25.37	20.28
40	188.2	21.06	-16.50	0.50	-16.50	1.89	16.50	1402	243.9	26.36	21.81
41	155.2	21.09	-20.50	0.64	-18.48	1.21	16.68	1405	210.1	26.20	17.56
42	195.2	21.13	-20.50	0.64	-18.48	1.21	9.55	1400	211.4	26.52	15.43
43	152.2	21.12	-16.50	0.64	-14.48	1.21	10.96	1404	171.8	26.52	21.23
44	188.2	21.08	-16.50	0.64	-14.48	1.21	15.23	1398	236.0	26.61	16.10
45	183.0	19.89	-19.00	0.60	-16.58	1.55	12.79	1348	213.3	23.89	41.75
46	182.7	21.93	-19.00	0.60	-16.58	1.55	12.25	1348	213.4	24.21	47.61
47	184.8	20.92	-19.00	0.60	-16.84	1.55	12.33	1198	213.3	25.22	70.90
48	178.0	20.79	-19.00	0.60	-16.29	1.55	12.13	1498	213.1	25.90	31.18
49	182.7	20.77	-19.00	0.80	-14.94	1.55	11.98	1353	213.4	26.02	20.52
50	183.2	20.84	-18.00	0.60	-23.68	1.55	11.85	1351	213.3	25.57	21.74
51	181.7	20.77	-19.00	0.60	-6.02	1.55	12.02	1353	213.3	25.21	26.59
52	182.9	20.75	-23.00	0.60	-20.57	1.55	11.83	1353	213.3	25.42	31.17
53	184.1	20.60	-10.00	0.60	-7.57	1.55	16.60	1342	214.2	22.10	18.37
54	106.7	20.62	-19.00	0.60	-16.57	1.55	15.60	1350	125.5	22.82	12.64

Table F.6: Negative PSEP training output data

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
1	49.23	16.56	273.6	21.00	22.79	0.44	151.3	11.38	9.02	0.52	36.99
2	92.83	16.83	180.7	26.06	40.09	0.48	162.6	8.44	10.01	0.43	38.91
3	40.33	16.57	319.3	31.41	21.39	0.48	164.7	9.69	0.35	0.42	38.94
4	61.54	16.70	229.8	17.22	23.12	0.44	140.6	8.28	0.27	0.55	36.39
5	112.14	10.43	119.9	26.08	74.23	0.48	124.0	5.13	0.03	0.32	25.31
6	71.84	10.87	198.6	37.77	32.17	0.51	150.0	7.73	0.07	0.26	26.98
7	95.76	10.85	196.3	28.82	47.71	0.48	128.5	6.32	0.05	0.33	26.32
8	125.68	10.90	153.2	34.66	69.68	0.48	143.7	5.39	0.08	0.29	27.09
9	190.76	10.43	138.7	30.06	133.40	0.49	119.2	4.34	0.06	0.32	26.39
10	131.86	11.44	140.6	34.52	81.33	0.49	142.8	6.22	0.15	0.29	29.00
11	144.58	11.36	161.1	37.89	97.80	0.47	129.8	4.48	0.15	0.29	28.14
12	103.48	10.84	234.5	30.25	42.87	0.50	137.4	7.06	0.08	0.31	26.46
13	81.73	11.45	218.6	34.84	31.64	0.49	157.3	8.43	0.09	0.30	28.41
14	34.54	17.11	332.7	20.38	21.01	0.45	164.8	13.82	0.36	0.53	38.50
15	86.85	16.88	127.5	18.54	25.77	0.46	139.4	9.06	0.49	0.53	38.18
16	41.24	15.96	228.8	32.91	20.84	0.47	155.9	8.34	0.37	0.40	38.02
17	41.73	14.51	131.6	35.95	27.39	0.50	142.4	5.39	0.03	0.34	35.88
18	64.84	4.40	60.6	41.19	51.35	0.48	111.4	3.44	0.02	0.12	10.71
19	355.33	20.89	61.1	16.99	26.37	0.41	128.5	5.87	2.76	0.63	45.14
20	29.33	15.12	222.8	39.91	14.42	0.47	128.5	6.61	0.14	0.36	35.67
21	61.68	14.49	73.2	23.47	31.16	0.47	127.9	5.19	0.04	0.43	34.69
22	48.47	14.69	99.4	25.12	24.67	0.48	128.6	5.71	0.03	0.42	35.06
23	42.85	14.60	116.5	27.47	22.73	0.48	130.1	5.67	0.03	0.40	34.82
24	42.56	14.91	125.0	26.46	23.19	0.47	135.6	6.38	0.03	0.42	35.45
25	43.47	14.81	100.2	26.46	22.76	0.46	127.7	4.85	0.04	0.43	34.92
26	54.29	14.58	138.5	27.00	26.95	0.48	136.5	6.85	0.03	0.41	34.83
27	43.65	14.59	130.1	28.64	23.05	0.47	132.1	5.89	0.03	0.40	35.01

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
28	44.39	14.61	130.3	28.68	23.31	0.48	132.2	5.82	0.03	0.40	35.28
29	44.51	14.66	129.9	28.63	23.86	0.48	132.2	5.79	0.03	0.40	35.41
30	47.78	14.52	114.4	27.49	26.16	0.49	131.4	5.61	0.03	0.40	35.11
31	47.09	14.36	115.3	27.75	26.14	0.48	131.4	5.56	0.03	0.40	35.05
32	40.35	14.88	131.4	28.83	22.70	0.47	130.0	5.05	0.03	0.40	34.73
33	43.05	14.48	111.7	28.02	25.00	0.47	130.1	5.31	0.04	0.41	34.69
34	42.68	14.42	111.2	28.02	24.23	0.47	129.8	5.24	0.03	0.40	34.42
35	45.10	14.79	112.9	27.34	25.52	0.47	129.1	5.15	0.03	0.41	34.54
36	43.25	14.84	115.2	27.74	24.88	0.46	129.9	5.11	0.03	0.41	34.52
37	97.38	17.49	132.4	18.22	32.27	0.45	143.4	8.99	0.09	0.54	39.82
38	39.08	16.87	256.6	31.77	23.60	0.49	160.1	9.50	0.21	0.41	41.07
39	95.35	17.47	141.7	21.00	22.45	0.46	122.8	7.11	0.11	0.53	39.82
40	86.50	17.04	103.5	24.34	34.61	0.46	130.6	5.86	0.12	0.47	40.07
41	40.03	12.45	138.5	24.82	25.67	0.48	128.3	7.82	0.02	0.37	28.84
42	32.02	12.48	210.6	36.41	20.85	0.49	143.0	8.10	0.03	0.31	30.37
43	44.65	12.15	124.8	28.38	24.72	0.48	113.8	5.67	0.02	0.36	28.61
44	54.49	12.25	105.5	31.83	32.92	0.47	123.1	5.17	0.04	0.33	29.29
45	32.24	12.94	157.0	30.86	20.56	0.48	134.5	6.38	0.06	0.36	32.24
46	32.62	14.91	173.4	28.38	20.34	0.47	141.7	7.19	0.05	0.41	35.28
47	18.64	12.36	216.0	30.60	16.23	0.48	156.8	8.68	0.04	0.35	29.95
48	43.86	15.57	120.7	29.30	24.77	0.48	121.2	4.53	0.09	0.40	36.19
49	29.67	14.59	135.5	29.39	18.15	0.47	131.0	5.83	0.28	0.39	34.05
50	54.80	14.15	303.2	29.75	24.36	0.47	156.9	9.55	0.03	0.38	33.08
51	21.04	14.21	62.6	29.59	17.97	0.44	98.3	2.97	0.02	0.39	31.62
52	21.97	14.23	229.4	29.52	20.60	0.48	150.0	8.18	0.04	0.38	33.76
53	73.99	13.54	58.0	28.61	49.83	0.45	96.6	2.99	0.14	0.37	30.96
54	51.15	13.94	101.6	15.40	20.07	0.44	111.3	6.90	0.07	0.53	30.23

F.4 Negative PSEP Validation Data

This section presents the validation data used for the negative PSEP multi-region and combined region models. Table F.7 shows the input data and Table F.8 shows the output data.

Table F.7: Negative PSEP validation input data

Index	p_{air}	p_{diesel}	p_{ng}	SOI_{diesel}	PW_{diesel}	SOI_{ng}	PW_{ng}	ω_{eng}	p_{exh}	T_{air}	T_{cool}
1	121.6	19.36	-12.50	0.60	-13.93	1.10	20.19	1196	119.8	23.38	41.43
2	162.4	19.56	-16.50	0.60	-14.33	0.75	13.10	1200	158.9	23.47	36.32
3	211.5	19.38	-22.50	0.60	-19.65	1.70	4.61	1185	206.1	25.23	37.95
4	209.4	19.42	-13.50	0.60	-15.73	2.05	0.38	1252	202.8	26.06	49.78
5	211.4	19.62	-14.00	0.60	-13.99	1.45	7.22	1249	206.4	25.88	49.46
6	197.5	19.58	-16.00	0.60	-15.24	2.05	23.57	1248	196.6	26.45	45.57
7	201.8	19.63	-11.00	0.60	-10.24	2.20	9.13	1250	196.7	26.93	37.42
8	123.4	19.57	-14.50	0.60	-12.98	2.20	26.44	1256	122.6	25.86	43.44
9	190.4	19.59	-15.50	0.60	-14.52	1.42	14.63	1232	187.3	25.27	36.03

Table F.8: Negative PSEP validation output data

Index	CO	CO_2	NO_X	O_2	CH_4	Efficiency	p_{cyl}	dp/dCA	PM	EQRO	Power
1	113.81	8.45	79.39	24.56	72.04	0.44	98.2	3.33	0.01	0.29	18.39
2	47.49	3.98	62.47	40.65	33.20	0.46	112.4	4.36	0.02	0.10	8.41
3	11.84	11.47	313.33	42.66	7.08	0.48	168.2	11.66	0.07	0.27	26.97
4	49.99	15.11	230.60	40.82	13.16	0.45	110.6	6.22	0.31	0.36	34.98
5	48.63	11.40	137.62	42.47	14.62	0.46	137.1	6.89	0.05	0.27	26.95
6	69.05	15.13	69.07	25.08	28.73	0.47	135.5	6.75	0.32	0.39	33.91
7	50.97	16.63	89.02	31.82	21.68	0.44	117.6	7.72	0.23	0.40	35.36
8	439.70	16.41	25.33	10.91	32.73	0.41	96.8	3.78	0.32	0.61	32.69
9	55.06	10.17	89.72	36.07	25.52	0.46	128.34	6.00	0.13	0.26	23.09