KNOWLEDGE EXTRACTION FOR FISCHER-TROPSCH SYNTHESIS USING LOCAL LEARNING APPROACHES

by

Ayça Soylu
B.S., Chemical Engineering, Boğaziçi University, 2015

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Chemical Engineering
Boğaziçi University
2018

to my family

and their dedicated partnership for success in my life

ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor Prof. Ramazan Yıldırım. The door to Prof. Yıldırım office was always open whenever I ran into a trouble spot or had a question about my paper. He consistently allowed this paper to be my own work, but steered me in the right the direction whenever he thought I needed it.

I wish to express my sincere appreciations for the members of thesis committee, Assoc. Prof. Burak Alakent and Assist. Prof. M.Erdem Günay, for accepting to be a member of the thesis committee, devoting their valuable time to read and comment on my thesis.

I also thank my friends Sercan Altundemir, Ayşe Eren, Ahmet Hallaçeli, Ece Çiğdem Mutlu, Gözde Öztürk, Cihat Öztepe for their friendships through my M.Sc. years.

A special thank to Berkay Bulut. Words cannot describe how lucky I am to have him in my life. He has selflessly given more to me than I ever could have asked for.

Finally, I wish to thank my family; my father, my mother, my brother, my two sisters Manolya and Petek and my uncles Aleaddin and Kemal. They inspired me to come this far.

ABSTRACT

KNOWLEDGE EXTRACTION FOR FISCHER-TROPSCH SYNTHESIS USING LOCAL LEARNING APPROACHES

In this study, local empirical models for Fischer-Tropsch Synthesis (FTS) were constructed by using machine learning algorithms on the experimental data published in the literature. CO conversion in FTS was modeled as a function of catalyst design variables, physical properties of the catalyst, and operating conditions by using gradient boosting method (GBM), artificial neural network (ANN) and random forest in R 3.2.2 environment. The FTS database was taken from a previous study. Before modeling FTS database, local and global learning techniques were compared using three different simpler and known databases involving selective CO oxidation reaction, water gas shift reaction and steam reforming of methane reaction. These databases were modeled by using GBM, ANN and random forest with both local and global techniques to see the difference in a known model. Local learning showed better performance in each database on the prediction of unseen data. Then the Fischer-Tropsch synthesis database was analyzed by using local learning with a different number of query data. Optimal parameters were determined for each method by training the model with each combination of parameters the standard deviation and mean were calculated and compared to determine the optimal parameters. The model with the best results was chosen as the final model. The R^2 values ranged between 0.86 and 0.87 with GBM, 0.68 and 0.75 with ANN and 0.82 and 0.86 with random forest algorithms. RMSE values ranged between 8.60 and 9.16 with GBM, 12.24 and 13.98 with ANN and 9.23 and 10.32 with random forest algorithms. R^2 and RMSE values were better than global learning results. GBM algorithm performed better on FTS database both in terms of R^2 and RMSE.

ÖZET

FISCHER-TROPSCH SENTEZİ İLE İLGİLİ YEREL ÖĞRENME KULLANILARAK BİLGİ ÇIKARIMI

Bu çalışmada, literatürde yayımlanan deneysel veriler üzerine özdevimli öğrenme algoritmaları kullanılarak Fischer-Tropshc Sentezi (FTS) üzerine yerel ampirik modeller oluşturulmuştur. FTS'deki CO dönüşümü R 3.2.2 ortamında gradyan güçlendirme, yapay sinir ağı (YSA) ve rastgele orman algoritmaları kullanılarak katalizör tasarımı değişkenleri, katalizörün fiziksel özellikleri ve operasyon koşullarının bir fonksiyonu olarak modellenmiştir. Çalışmada, önceki çalışmalarda kullanılan FTS veritabanı kullanılmıştır. FTS veritabanını modellenmeden önce yerel ve global öğrenme teknikleri seçici CO oksidasyonu reaksiyonu, su-gazı geçiş reaksiyonu ve metan buhar reformu reaksiyonlarına ait olmak üç farklı ve önceden bilinen veritabanı kullanılarak karşılaştırılmıştır. Bu veri tabanları, hem yerel hem global teknikler ile gradyan güçlendirme, YSA ve rastgele orman kullanılarak modellenmiştir. Yerel öğrenme her bir veri tabanında görülmemiş veriyi tahmin etmede daha iyi performans göstermiştir. Bu yüzden FTS veritabanı yerel öğrenme kullanılarak farklı sorgu verileri ile analiz edilmiştir. En iyi parametreler çeştili paremetre kümeleri denenerek standart sapma ve ortalamalar kıyaslanarak belirlenmiştir. En iyi sonucu veren parametrelerle oluşturulan model nihai model olarak seçilmiştir. R^2 değerleri gradyan güçlendirme için 0.86 ve 0.87 aralığında, YSA için 0.68 ve 0.75 aralığında ve rastgele orman için 0.82 ve 0.86 aralığında, kök ortalama kare hatası (KOKH) ise gradyan güçlendirme için 8.60 ve 9.16 aralığında, YSA için 12.24 ve 13.98 aralığında ve rastgele orman için 9.23 ve 10.32 aralığında bulunmuştur. R^2 ve KOKH değerleri global öğrenme sonuçlarına göre daha iyi çıkmış, gradyan güçlendirme algoritması FTS veritabanında hem R^2 hem KOKH bakımında daha iyi performans göstermiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

f	Activation function
R^2	Coefficient of determination
x_j	j^{th} input variable of an instance
y	Output of an instance
ΔH	Enthalpy of reaction
ω	Weight of the connection
θ	Bias

LIST OF ACRONYMS/ABBREVIATIONS

ANN	Artificial neural network
FTS	Fischer-Tropsch Synthesis
GA	Genetic algorithm
GBM	Gradient boosting model
HTFT	High-temperature Fischer-Tropsch Synthesis
HTS	High-temperature stage
LOOCV	Leave-one-out cross-validation
LTFT	Low-temperature Fischer-Tropsch Synthesis
LTS	Low-temperature stage
RF	Random forest
RMSE	Root mean squared error
SMR	Steam methane reforming reaction
SVM	Support vector machines
WGS	Water gas shift reaction

1. INTRODUCTION

The rapidly increasing human population causes an increase in energy consumption in the world. (Li, 2005); The global energy consumption continuously increases as a result of population growth and rapid industrial development. (Tanksale *et al*., 2010). As the result, CO_2 emissions are increased due to the extensive fossil fuels use (Song, 2002). In addition, the petroleum sources are limited and they are estimated to be depleted in the near future. Hence, the search of alternative technologies and renewable energy sources has significantly increased in recent years.

Fischer-Tropsch Synthesis (FTS) is an alternative way to produce liquid fuels from coal, natural gas, shale gas, coal-bed gas and biogas feedstocks. FTS gains popularity because the direct transformation of energy sources to liquid fuels is expensive and impractical. Long hydrocarbon chains, precursors of fine chemicals, organic oxygenates are also the products of FTS. FTS consists of two main steps: gasification or reforming of the raw material, and hydrogenation of the produced synthesis gas (Röper, 1983). There are numerous studies focus on FTS. The main purpose of these studies is mostly designing and developing novel catalyst with high activity and preferred selectivity (Zhai *et al*., 2013). Co, Fe, Ru, and Ni are the most commonly used active metals in FTS catalysts. Co-based catalysts are more efficient in the production of long hydrocarbon chains while Fe-based catalysts are mostly used in coal-to-liquid and biomass-to-liquid processes. Ru based catalysts are the most efficient ones, but its high price limits its usage in the industry. Studies focus on improving the performance of Co and Fe based catalysts (Zhai *et al.*, 2010).

In the literature, a huge amount of experimental studies about Fischer-Tropsch synthesis have been reported. Because of the size and the complexity of the related literature, it is very difficult to obtain valuable knowledge from this literature with naked eyes. Data mining tools can be used in order to analyze these studies. Data mining is the process of analyzing data and getting useful information from it (Rokach and Maimon, 2008). The applications of the data mining methods have become quite widespread due to the great development in computer hardware and software in the last two decades. Some of the common data mining tools are clustering, classification, multiple linear regression, decision trees, k-nearest neighbor algorithm, and artificial neural networks (Romero and Ventura, 2007).

The problem of extracting knowledge from observed data have been the object of several disciplines. Studies about this issue proposed two main methods: local memory-based versus global methods (Bontempi *et al.*, 1999).

A single model is built on the dataset in the global method. Traditionally, neural network and other types of non-linear regression had been used for this purpose. These learning models compute a function from the given dataset and predict the unknown data queries based on the computed function whereby not utilizing the readily learned dataset in its future predictions unlike the local memory-based algorithms.

Prediction models such as the classical nearest neighbor method are local memory-based algorithms. Unlike decision trees or regression models where there is a rule or a function is calculated and predefined, local memory based models process and compute the entire dataset for a specifically prediction query. The labeled data used to build the memory is held in the memory, and the prediction is derived from the neighbors of the prediction query point. This process is iterated for each of the query points to predict.

One of the most popular machine learning techniques is the Artificial Neural Network (ANN). The artificial neural networks give the output as a function of inputs by creating the very large number of correlations within the network (Khajeh-Hosseini-Dalasm *et al.*, 2011). Random Forest is another method of machine learning that fits many of decision trees on the dataset. Each tree is trained by randomly and independently selected attributes, instead of all the attributes in the dataset (Tufféry, 2011).

Gradient Boosting Model is a relatively new method of machine learning. Gradient Boosting is based on shallow trees as small as decision stumps, and it reduces the error by reducing the bias of the whole model by adding a new learner to improve the already trained learner by trying to find the optimal linear combination of tress.

In this work, a database for CO conversion in FTS was taken from a previous study (Burnak, 2015) and modeled by using local learning algorithms. Artificial Neural Network, Random Forest, and Gradient Boosting methods were tested and compared. Before using local learning algorithms on FTS database, the applicability of local learning techniques for this type of database was tested using the databases of selective oxidation of CO by Cu catalyst (Günay and Yıldırım, Neural network analysis of selective CO oxidation over copper-based catalysts for knowledge extraction from published data in the literature, 2011), Water Gas Shift Reaction (Odabasi *et al.*, 2014) and Steam Reforming (Baysal *et al*., 2017).

This study comprises of five Chapters. In chapter 2, selective CO Oxidation, water gas shift reactions, steam reforming, Fischer-tropsch synthesis and local learning algorithms were briefly reviewed. The details of the database and the computational details are given in Chapter 3. In Chapter 4, the obtained result and discussion were presented. Finally, the conclusion of this study and recommendations of future studies were done in Chapter 5.

2. LITERATURE SURVEY

2.1. Selective CO Oxidation

In order to convert chemical energy into electrical energy, fuel cells are used. Hydrogen is the energy source of a fuel cell (Choudhary and Goodman, 2002). The hydrogen source is the main problem of using fuel cells. Partial oxidation of hydrocarbons can be used for producing H_2. However, partial oxidation of hydrocarbons generates CO as a side product (Son and Lane, 2001). That means H_2 should be purified totally because CO poisons the noble metal anode of the fuel cell cells and it reduces the power output of the cell (Manasilp and Gulari, 2002).

There are many methods that use to remove CO but the most simple and cheap method is selective catalytic CO oxidation (Avgouropoulos *et al.*, 2002). In order to reduce CO level, the catalysts with high oxidation rate of CO and high selectivity with respect to H_2 are needed (Manasilp and Gulari, 2002). Selective CO oxidation reactions can be seen in equation 2.1 and 2.2 as desired and undesired reaction.

$$Desired\ Reaction:\ CO + \frac{1}{2}O_2 \rightarrow CO_2 + 67.6\frac{kcal}{mol} \tag{2.1}$$

$$Undesired\ Reaction: H_2 + \frac{1}{2}O_2 \rightarrow H_2O + 58.6\frac{kcal}{mol} \tag{2.2}$$

CO oxidation over Cu-based catalyst was studied by Günay and Yıldırım, and 1337 data points from 20 publications are modeled by using artificial neural network algorithms. CO conversion was successfully predicted as the function of operating conditions and catalyst preparation methods (Günay and Yıldırım, 2011).

Another study of Günay and Yıldırım is focused on to predict the CO conversion for the selective CO oxidation over promoted Pt/Al_2O_3 catalysts by using modular neural network modeling. 520 data point was successfully modeled (Günay and Yıldırım, 2010).

2.2. Water Gas Shift Reaction

The water gas shift reaction is commonly used in the industry. It is generally applied to hydrogen production processes from hydrocarbons and ammonium synthesis. The main aim of the WGS reaction is to remove CO from the off-gas and increase the H_2/CO molar ratio. The water gas shift reaction is partially exothermic and controlled by the thermodynamic equilibrium. The equilibrium conversion is said to be independent of pressure and favored under low temperatures because of its exothermic characteristic. Hence, the WGS reactor is a large and heavy process component, and the WGS reactor volume is determined by the equilibrium limitations (Francesconi *et al.*, 2007).

The WGS reaction can be catalyzed by several materials; there are two stages that use two different catalysts: high-temperature stage (HTS) and low-temperature stage (LTS). Iron-based catalysts are used in HTS and operation temperature of HTS is range between 300 ℃ and 450 ℃. HTS converts the largest amount of CO. Typically, in the outlet stream, CO levels are between 2% and 4%. Low-temperature stage operates at between 180 ℃ and 230 ℃, and copper-zinc catalysts are used in LTS (Francesconi *et al.*, 2007).

In spite of the common use of two-stage WGS processes, it is technically complex and inappropriate for mobile applications (Bartolomew and Farrauto, 2006) hence, a single-stage WGS process becomes more desirable. Supported noble metal (Au, Rh, Pt, and Pd) based catalysts seem to be promising single stage WGS catalysts because of their robustness, suitability to operate at higher temperatures, being less sensitive to poisons and more active than Fe/Cr oxide-based HTS catalysts (Azzam *et al.*, 2007).

In studies of Odabaşı, Günay and Yıldırım, decision trees, support vector machines (SVM) and artificial neural networks (ANN) techniques were performed for a database containing the past published works from 2002 to 2012 for water gas shift reaction over noble metal catalysts (Pt and Au). Decision trees were used to determine the empirical rules and conditions that lead to high catalytic performance (high CO conversion); artificial neural networks (ANNs) was carried out to determine the relative importance of various catalyst preparation and operational variables and their effects on CO conversion (Odabasi *et al.*, 2014).

2.3. Steam Reforming

Steam reforming is the most vital way to convert CH_4 and H_2O into H_2 as a green energy carrier and CO as a by-product. There are two steps in steam reforming of methane. Steam methane reforming reaction (SMR) and water gas shift reaction (WGS) are carried out respectively. The chemical reactions involved in steam reforming of methane are indicated as follows:

Steam methane reforming (SMR):

$$CH_4 + H_2O \leftrightarrow CO + 3H_2 \quad \Delta H_{298} = 206 \text{ kJ/mol} \tag{2.3}$$

Water gas shift reaction (WGS):

$$CO + H_2O \leftrightarrow CO_2 + H_2 \quad \Delta H_{298} = -41 \text{ kJ/mol} \tag{2.4}$$

Overall Steam Reforming Reaction:

$$CH_4 + 2H_2O \leftrightarrow CO_2 + 4H_2 \quad \Delta H_{298} = 165 \text{ kJ/mol} \tag{2.5}$$

The steam reforming step is an endothermic process. Thus, the process usually operates at nearly 850°C to obtain desirable conversion (Gallucci *et al.*, 2006). In order to produce more hydrogen, more steam is added, and the water gas shift reaction is carried out as the second step.

Traditionally, the process is performed in multi-tubular fixed-bed reactors in the presence of a metal catalyst. Since reforming reactions are highly endothermic, steam reforming reactions together is carried out at high temperature (700-900 °C) to achieve high conversions (Gallucci *et al.*, 2004).

Usually, nickel-based catalysts are used for methane steam reforming (Bengaard *et al.*, 2002). According to previous studies, Ru can be considered as one of the most active metals while catalyzing the methane steam reforming reaction (Rostrup-Nielsen, 1993). Other noble metals such as Rh, Pt, Pd, and Ir also have important effects on methane steam reforming activity (Jones, et al., 2008).

In the study of Baysal in 2015, decision trees and artificial neural networks (ANN) techniques were performed for a database containing the past published works for steam reforming reaction over Ni, Ru, and Rh based catalysts. Analysis of total data with decision tree modeling resulted in 20.83% training error and 22.91% testing error. With ANN algorithm R^2 and RMSE values of training were found to be 0.97 and 6.03, whereas they were 0.93 and 8.78 for testing. (Baysal *et al.*, 2017).

2.4. Fischer-Tropsch Synthesis

One of the clean and most efficient methods to produce sulfur and nitrogen-free fuels is Fischer-Tropsch Synthesis (FTS) method producing the elements from syngas which is derived from natural gas, coal, or biomass (Shimura *et al.*,2014). Carbon monoxide is hydrogenated by the reactions written in the equations 2.6, 2.7 and 2.8 (Trépanier *et al.*, 2010).

$$2n\ H_2 + nCO \rightarrow C_nH_{2n} + nH_2O \tag{2.6}$$

$$(2n+1)\ H_2 + n\ CO \rightarrow C_nH_{2n+2} + n\ H_2O \tag{2.7}$$

$$2n\ H_2 + n\ CO \rightarrow C_nH_{2n+1}OH + (n\text{-}1)\ H_2O \tag{2.8}$$

The main purpose of FTS reaction is maximizing the selectivity of higher hydrocarbon chains and minimizing the methane conversion. One of the main challenges of FTS is that equation 2.7 can yield methane formation reaction. Water gas shift reactions, the formation of bulk carbides and carbonaceous reactions and formation of carbon monoxide which is produced by the reaction of carbon and carbon dioxide are other undesired reactions (Dalai and Davis, 2008), (Lögdberg *et al.*, 2009).

Metal oxide supported Cobalt and Ruthenium catalysts and unsupported Iron-based catalysts are used for FTS (Zhang *et al.*, 2013.). On-going research both focus on supported and unsupported Fe-based catalysts. Galvis et al. compared performances of unsupported base iron with α-Al2O3 supported iron catalyst. They reported that supported iron yields better stability along with higher CO conversion and C5+ selectivity (Galvis *et al.*, 2013).

Physical properties of catalyst support such as surface area, pore volume, and pore diameter have an important effect on FTS activity; the interaction between the support and active side is also important. If the interaction is too strong, the extent of reduction of the active side will be decreased, if the interaction is too weak the active phase cannot disperse properly on the surface (Zhang *et al.*, 2010). Nurunnabi et al. conducted a study with Ru as the active metal, and compared FTS activity on Al_2O_3 and SiO_2. They revealed that large particle size and high reducibility of Ru on Al_2O_3 increases CO conversion and C21+ selectivity, compared to SiO_2 support (Nurunnabi *et al.*, 2008), (Mirzaei *et al*., 2013).

Another key factor that affects the FTS activity is promoters. While Cobalt and Iron-based catalyst use alkali metals ions, noble metals and transition metals, Ruthenium-based catalysts don't need any promoters. Most of the Fe-based catalysts in the literature are promoted with alkali metals, especially with Li, Na, and K. (Zhang *et al.*, 2010).

FTS is divided into two groups according to its operating conditions. These are Low-Temperature Fisher Tropsch (LTFT) and High-Temperature Fisher Tropsch (HTFT). LTFT operates between 220℃ - 250℃ and HTFT operate between 330℃-350℃ (Zhang *et al.*, Development of novel catalysts for Fischer–Tropsch synthesis: tuning the product selectivity, 2010). While LTFT can carry out on ruthenium, iron, and cobalt based catalysts (Lu and Lee, 2007), HTFT can usually operate with iron-based catalysts (Kwack *et al.*, 2011). Reactor types also differ from each other. Multi-tubular fixed bed and slurry bed reactors are used in LTFT whereas in HTFT multi-tubular fixed bed, fixed the fluidized bed and circulating fluidized bed reactors are used.

The first study on FTS used the data mining algorithms was published by Sharma et al. in 1998. It was reported that ANN could be used to optimize hydrocarbon selectivity by predicting $CO+H_2$ conversion and steady state product concentrations under different reactor operating conditions (Sharma *et al.*, 1998). Rahimpour et al. developed the accuracy of ANN models on FTS performance by combining with genetic algorithm (GA) (Adib *et al.*, 2013).

2.5. Data Mining for Knowledge Extraction

Data mining can be considered as the combination of selection, exploration, and modeling of large databases. Unknown correlations, patterns, and trends can be found by using data mining tools. In other words, data mining can be used to extract knowledge from any data (Giudici and Castelo, 2003).

Data mining algorithms are commonly used in several fields. In the banking sector, the data mining applications are widely used for credit applications, fraud detection, and stock markets. Data mining algorithms are also used in other fields such as medicine, telecommunications, physics and biology (Alpaydin, 2010). On heterogeneous catalytic reaction fields, artificial neural network algorithms are used by various investigators since the work of Hattori et al in 1995 (Hattori and Kito, 1995).

2.5.1. Gradient Boosting

Gradient Boosting builds and additive model using a forward fashion. At each iteration for subsequent trees, decision trees or decision stumps which are called weak learners are built and added to the existing weak learners to minimize the loss of the model (Friedman, 2002).

A gradient boosting has three main elements as a loss function, weak learners and an additive model. A loss function is for optimization, weak learners is for making predictions and an additive model is for adding weak learners to minimize the loss function.

Gradient boosting model is a greedy algorithm of the ensemble models and can over fit the data very easily. Therefore, there are several hyper parameters (number of trees, random sampling, penalized learning) to tune which can decrease the variance of the model created. A number of trees is an essential parameter to tune to increase accuracy and decrease variance. A random sampling of the data is a big insight into ensemble models. Another approach to decreasing variance is stochastic gradient boosting (random sampling) (Friedman, 2002). At each iteration, a subsample data is drawn randomly and without replacement from the training dataset. Instead of the full sample, the randomly selected subsample data is used to fit the base learner. (Elith *et al.*, 2008)

2.5.2 Neural Networks

Artificial neural networks (ANN) is a powerful learning algorithm inspired by human neural processing. It is a simplified version of the human brain; converting coming inputs into outputs (Rud, 2001).

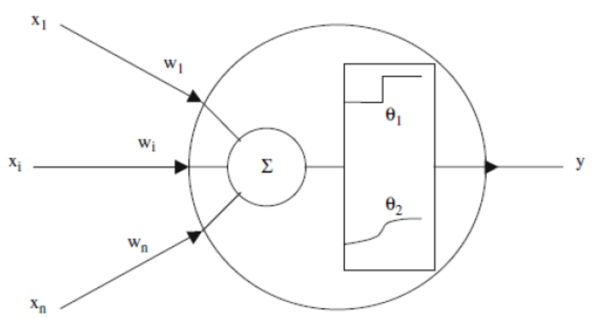


Figure 2.1. Artificial Neural Network Model (Cios *et al.*, 2007)

A neural network model consists of an input layer, hidden layer(s) and an output layer as shown in Figure 2.1. Each layer has elements called neuron; a neuron takes its input from the previous neuron and calculates the output using the weights from a transfer function. The result of the function is then transferred to the next neuron and is further propagated to calculating an output prediction value (Callan, 1999). The simplest ANN model is called a perceptron where there is only a single layer of neurons calculating an output. This model is not any different from a linear regression model or a logistic regression model. If a linear activation function, as described in Equation 2.9, is used in a perceptron, the model results in a linear model with an only linear understanding of the data, falling behind the potential of an ANN which is a non-linear representation of the data (Alpaydin, 2010).

$$f(x) = a + bx \tag{2.9}$$

Similar case holds true when a sigmoid function, as described in Equation 2.10, is used for an activation function; the model is similar to logistic regression.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.10}$$

Other widely used transfer functions are tangent sigmoid function and step function which is described in Equation 2.11 and 2.12 respectively (Alpaydin, 2010) (Tufféry, 2011).

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.11}$$

$$f(x) = \begin{cases} a, & x > c \\ b, & otherwise \end{cases} \tag{2.12}$$

In Equation 2.12, a, b and c are constants. A and c usually equal to 1, and b equals to 0, yielding the sign-activation function.

The full potential of an ANN is utilized with the functions used in the model and the complexity of the model in which increasing number of layers. Each neuron is a representation of an xn with a weight ωn with an activation function, f. The functions are propagated through each layer of the model with a resulting output. Each function's weights are calculated by the output of the previous function. Incoming signals and weights are summed by dot product and compared with the bias, θ, to produce an output, y. The output of a single neuron is scaled by an output activation function, f. Accordingly, the output of a neural network can be described as in Equation 2.10 (Cios *et al.*, 2007).

$$\hat{y} = f(\textstyle\sum_{i=1}^{n} w_i x_i - \theta) \tag{2.12}$$

The complex structure of functions allows a non-linear interpretation of the train data whereby allowing a higher accuracy but more susceptible to variance – overfitting.

The weights of the connections are determined by implementing a backpropagation algorithm. The back-propagation algorithm is applied in the prediction error for a certain data set. This algorithm has the rules shown in Equation 2.13 and Equation 2.14:

$$w_{i\,j,\ new} = w_{i\,j,\ current} + \Delta w_{i\,j} \tag{2.13}$$

$$\Delta w_{i\,j} = \eta\ \delta_j x_{i\,j} \tag{2.14}$$

where η is the learning rate (0 < η < 1), δj is the responsibility for an error belonging to node j, and xi j is the ith input to node j.

2.5.3. Random Forest

Random forest is a model derived from the ensemble model – bagging trees. In bagging, random decision trees are constructed independently of each other where in the end a simple majority vote is taken for predicting the output (James *et al.*, 2013). In 2001, Breiman proposed random forests which add a layer of randomness to bagging, whereby adding a layer of randomness by constructing the trees based on the best subset of predictors randomly chosen at splitting decision tree nodes (Breiman, 2001). One of the main advantage of random forest is the low number of hyper parameters to search for which are the number of variables in the random subset and the number of trees in the random forest model.

The random forest algorithm model first draws random bootstrap samples from the train data. For each bootstrap samples, the model grows an unpruned classification or regression tree by randomly sampling the best variables, choosing the best split among the chosen variables. The result of each decision tree is aggregated, and the prediction is made by taken a majority vote (a majority vote is 0.5 cut-off value for classification problems). In unbalanced data cases, the cut-off value can also be tuned for higher precision; however, caution must be taken for the variance caused by overfitting. (James *et al.*, 2013).

2.5.4. Local Learning Algorithms

The learning process is described through three components (Guyon I. *et al.*, 1992); First random vectors x are drawn independently from a fixed but unknown distribution P(x). Secondly, a supervisor returns an output vector y to input vector x according to a conditional distribution function P(y|x), also fixed but unknown.

$$(x_1, y_1) , \ldots , (x_t, y_t)$$

Lastly, a learning machine and algorithm capable of implementing a set of functions called activation functions.

The learning problem has been formulated as of selecting the best class of functions providing the best approximation to the response supervisor however the set function might not contain a good predictor for the full input space but contain good approximators for local regions; local learning.

A simple local learning algorithm's method starts by selecting a few training examples located near the test examples, followed by training a machine learning algorithm with the selected training examples in the previous step and lastly by applying the trained machine learning algorithm to the test data (Bottou and Vapnik, 1992).

In the present literature, (Vapnik, 1992), devised a theoretical analysis for local learning, thereby introducing a new component named locality for the trade-off between the size of the train data and the capacity of the learning algorithm. The generalization performance is affected by the trade-off between the number of training examples and the capacity of the learning system where various parameters monotonically control the capacity of the learning system (Guyon I. *et al.*, 1992). Bottou and Vapnik's research attempts and demonstrates local learning algorithms might be very efficient for certain tasks. They proposed to retrain a simple classifier every time a new test pattern is presented. The linear classifier is only trained on the k patterns in the training set that is closest to the current test pattern. Their demonstrations show that a careful control of the locality parameter provides a performance breakthrough for an optical character recognition problem, improving both the error rate and the rejection performance.

In other domains where distributions vary in the input space or the distributions change in time, research shows that lazy learning algorithms are among the most effective models (Widmer and Kubat, 1996). On the other hand, before considering any new unseen knowledge, eager learning systems determine their generalization mechanism by building a model based on the training dataset.

3. COMPUTATIONAL DETAILS

3.1. Preparation of Data

In this work, FTS database taken from a previous study (Burnak, 2015) was modeled by using lazy learning algorithms. In FTS dataset 249 research articles were reviewed. These 249 articles yield 4755 data points and 140 variables. The variables in the database are presented in Table 3.1 with their attributes and identities.

Table 3.1. Input attributes and their identities.

Attribute	Identity
Base metal	Co,Fe,Ru
Promoter metal	Ag, Au, Ca, Cu, K, La, Mg, Mn, Mo, Na, Ni, P, Pt, Re, S, V, Zr, Ba, Li, Cs, Ce,Cr, Nb, Ta, Ti, W, Rb, Rh, Zn,Sr, Pd
Support type	CeO2, CNF, H-ZSM-5, MWCNT, OMC, SBA- 15, SiC, SiO2, SWCNT, TiO2, ZnO, ZrO2, ZSM-5, α- Al2O3, γ-Al2O3,Na-Bentonite, SHS (mesoporous silica hollow sphere), MgO, MnO2, La2O3, beta-Mo2C, carbon sphere, Activated carbon, mesoporous carbon, carbon black, graphite, pure carbon, oxidized diamond, Silicate-1, H-ZSM-5, SBA-15 (ordered mesoporous carbon template), CMK-3 (ordered mesoporous carbon, SBA-15 used as template), HMS (ordered mesoporous carbon), MCM-22, MCM-41, MCM-48, TiO2-anatase, TiO2-rutile
Catalyst Preparation Method	Co-precipitation, precipitation, Impregnation, Co-impregnation, aqueous incipient wetness impregnation, slurry impregnation, spray-dried, Pore volume impregnation, incipient wetness impregnation, homogeneous precipitation, incipient wetness coimpregnation, evaporation to dryness, wetness impregnation, deposition precipitation, homogeneous deposition precipitation, pH precipitation, water-in-oil microemulsion
Calcination conditions	Calcination temperature, calcination time
Reduction conditions	Reduction temperature, reduction time, and reduction medium
Promoter loading method and order	Sequel impregnation, coimpregnation, reverse sequel impregnation, coprecipitation
Catalyst characterization	Surface area, pore volume, pore diameter, metal particle size
Operating conditions	Time on stream, temperature, pressure, GHSV, feed composition
Reactor type	Fixed bed, CSTR, Slurry, Monolith, Microchannel,

In catalyst characterization part, missing data points were considerable high, and the effects of catalyst characterization on conversion are comparatively low. Consequently, catalyst characterization part was removed from the database.

Some of the variables values are fixed and don't deviate among observations. This means that these variables do not affect the model. Therefore, constant variables were removed from the database.

After these removals, 2432 data points and 100 variables remained. Because these 100 variables were not in the same unit normalization was needed. All variables were mapped between 0 and one by normalization, described by Equation 3.1.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

Where x' is the transformed value, x is the values that the instance takes at a specific attribute, x_{min}, and x_{max} are the minimum and maximum values of the attribute, respectively.

3.2. Computational Tools

Before using local learning algorithms on FTS database, the applicability of local learning techniques for this type of database was tested using the databases of selective oxidation of CO by Cu catalyst, Water Gas Shift Reaction, and Steam Reforming.

Leave-one-out cross-validation (LOOCV) was used for evaluating in training phase for the best model. LOOCV technique splits the data into two parts. A single observation (x_1, y_1) is used for the test data, and the remaining observations $\{(x_2, y_2), \ldots, (x_n, y_n)\}$ are used as a training data. The statistical learning method is applied to $n - 1$ training observations, and a prediction $\hat{y}_1$ is made for the excluded observation, using x_1. The procedure is repeated by electing (x_2, y_2) for the test data, training the statistical learning procedure on the $n - 1$ data points.

While fitting a model on the train data, parameters were tuned. "caret" package in R environment was used for the tuning process; "train" function was used to tune parameters. A grid of parameters was created for a specific model by using "expand.grid" function and the model was trained for each combination of parameters. For each combination, the standard deviation and mean was calculated. As the final model, the parameters with the best results were chosen.

Three different machine learning algorithms were used as random forest, artificial neural network (ANN) and gradient boosting (GBM). Different parameters were tuned for each algorithm. For Random Forest, ‘mtry’, ‘ntree’ and ‘nodesize’ were tuned for each model. ‘mtry’ is the number of variables used to include in a tree building, ‘ntree’ is the number of trees to grow and ‘nodesize’ is the minimum size of terminal nodes. For ANN, size and decay parameters were specified. Size is the number of units in hidden layer and decay is the regularization parameter to avoid over-fitting. For GBM, interaction depth, shrinkage, number of minimum observation in a node (n.minobsinnode) and number of trees (n.trees) were tuned. Interaction depth is the number of splits that are performed on a tree; shrinkage defines the steps taken in the gradient boosting, tree building process would stop when the number of observations is equal to n.minobsinnode, and n.trees parameter defines the number of trees that constructed.

The success of the model is determined by its accuracy to predict the new data. Root-mean-square error (RMSE) was calculated to observe the accuracy of the model as described in Equation 3.2. Smaller RMSE means better fit of the model.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - t_i)^2} \tag{3.2}$$

Where, n is the number of instances in the testing set, p_i and t_i are the predicted and observed values of the i^{th} instance, respectively.

The coefficient of determination (R^2) was also used as the additional indicator for the accuracy of the models. R^2 measures how successful the fit is in explaining the variation of the data. The coefficient of determination was computed by Equation 3.3. R^2 can take on any value between 0 and 1, with a value closer to 1 indicating that a greater proportion of variance is accounted for by the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(t_i - p_i)^2}{\sum_{i=1}^{n}(t_i - \bar{t})^2} \tag{3.3}$$

3.2.1. Gradient Boosting

For each database, gradient boosting algorithm was applied with two separate way in R environment. First, gradient boosting model was used for local learning. Then, gradient boosting model was applied to global learning techniques.

For local learning applications, the first distance matrix was calculated by “dist” function. “dist” function computes and returns the distance matrix by computing the distance between the rows of a database. In “dist” function method was specified as Euclidian distance.

After calculating the distances, for each data point, the closest data points were chosen, and a new dataset was created with these closest data points (local data). The local datasets were subsamples of the global dataset, partitions of the global dataset, where each local dataset was created by selecting the shortest k number of observations to each unique observation in the global dataset.

In the train and test data separation, k closest number of observations were set as the train data, and the one unique observation was set as the test data to be predicted from the learned train data (local training).

In the next step, gradient boosted model was fitted to each newly created train data. To fit gradient boosting model “caret::train” function in R 3.3.2 was used. Before setting a model, parameters were chosen and tuned. The range of the parameters was set by "expand.grid" function in "caret" package.

The parameters were n.trees, shrinkage, interaction depth and n.minobsinnode for GBM; N.trees values were set between 100 and 500. Shrinkage values were adjusted between 0.001 and 0.1. Interaction.depth took values between 2 and 7. Finally, n.minobsinnode were adjusted as 2 and 10.

In “caret::train” function, for a specific model, the model was trained on a different dataset for each combination of tuning parameters – model tunning. Across each data set, the performance of hold-out samples was calculated, and the mean and standard deviation were summarized for each of the combination mixes. The combination mix with the optimal resampling statistic was selected as the final model, and the entire training set was used to train to fit a final model for final prediction.

In “caret::train” function, control method and modeling method were specified. For control method, 10-fold cross validation was chosen. That means that the training set was divided into 10 folds concerning experimental runs, for a 10-fold cross-validation. Cross-validation was used as a validation method to evaluate the complete training set with the same parameters. As modeling method, “gbm” was used. “gbm” fitted generalized boosted regression model on each train data. Gradient boosting is the process of iteratively adding basis functions greedily so that each additional basis function further reduces the selected loss function.

The model was built on the training data after “caret::train” function. Then test data was predicted by using the obtained model. These procedures were applied to each data point, and for each data point, one prediction result was obtained. A table of prediction results and parameters were created. After calculating every prediction results for every data points, RMSE and R^2 values were calculated. The process for constructing a gradient boosting model with local learning techniques was summarized and visualized in Figure 3.1.

For global learning, first parameters were defined from the table created with local learning. In this table, for every data point, there was one best-fit parameter. To apply global learning, these parameters were averaged, and the averaged parameters were fitted into the model. Leave-one-out cross-validation (LOOCV) was used for evaluating in training phase for the best model. “gbm” function was used for modeling as in the local learning. Parameters in “gbm” function were set as n.trees=500, shrinkage=0.1, interaction.depth=7, n.minobsinnode = 5 for FTS database. Similar to local learning, a table of prediction results were created, and RMSE and R^2 were calculated from this table.

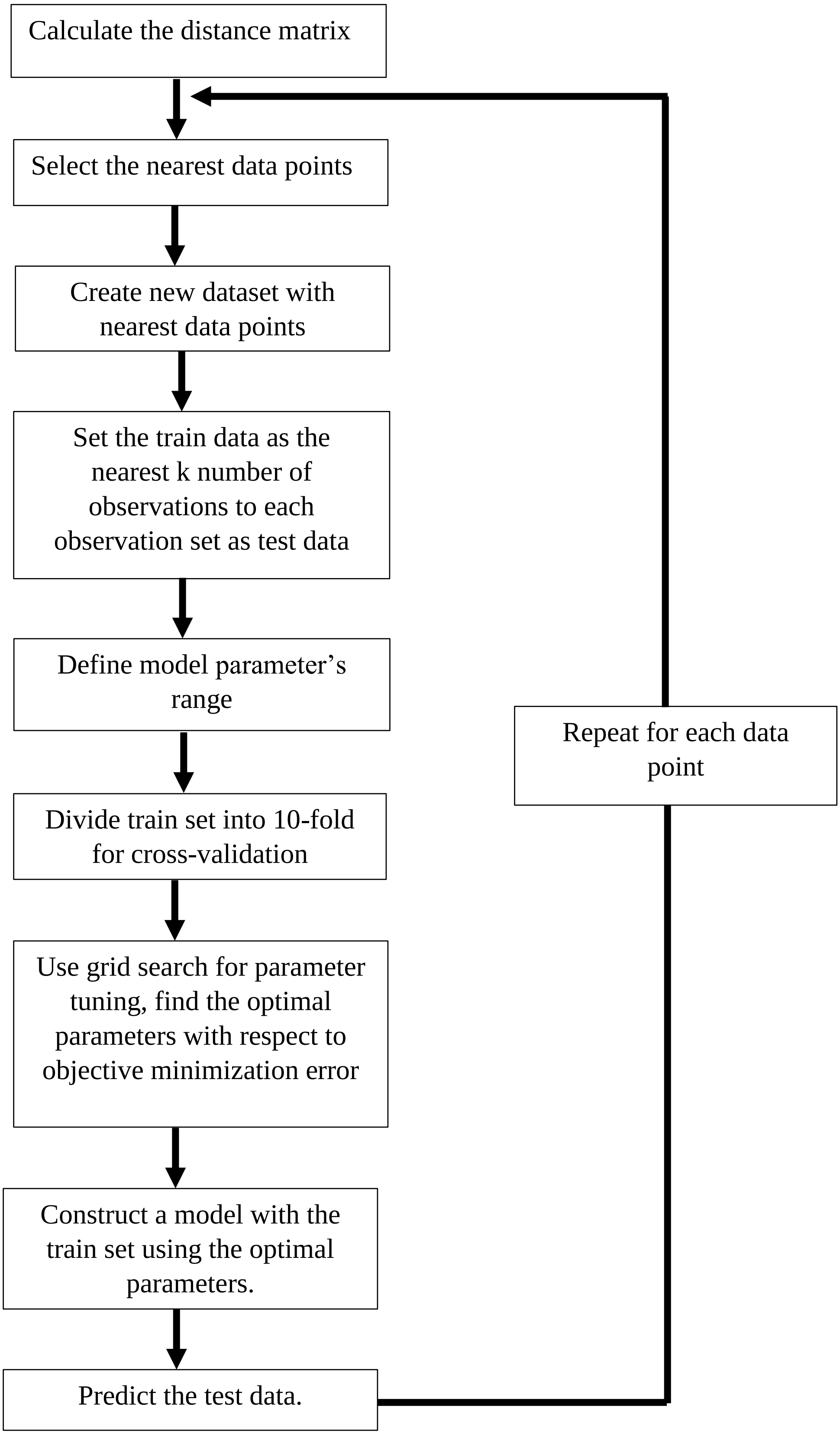


Figure 3.1. Gradient boosting with local learning construction process

3.2.2. Artificial Neural Network

R environment provides several packages for artificial neural network (ANN) implementations. "AMORE", "neural net", "monmlp", "qrnn", "frbf", “nnet” and “RSNNS” are some of the successful ANN packages. “nnet” was used to model the datasets with ANN algorithm. The process of constructing and validating ANN is very similar to gradient boosting models. Hence, the same process diagram in Figure 3.1 was followed.

Parameters are the number of units in the hidden layer (size) and the regularization parameter to avoid over-fitting (decay). Size and decay values were set between 5-7 and 0.1-0.5 accordingly.

The rest of the procedure was the same with gradient boosting model. Parameters were optimized by “caret::train” function and used to train the complete training set. These procedures repeated for each data point. Test data was predicted by using the obtained model. Table of predicted results of each data point and best-fit parameters was created. Parameters from this table were averaged, and averaged value was used in global learning part.

3.2.3 Random Forest

“randomForest” package (Liaw and Wiener, 2002) was used to model the datasets with random forest algorithm. “randomForest” function was used to grow the forest. The process is like GBM and ANN. The same procedure in Figure 3.1 was followed.

Random forest parameters, ‘mtry’, ‘ntree’ and ‘nodesize’, are set between 1-10, 50-500 and 1-5 accordingly. Other procedures are same with GBM and ANN.

4. RESULTS AND DISCUSSION

In this section, results are discussed in two parts. First, local and global learning algorithms were compared by using selective CO oxidation, water gas splitting, and steam reforming data prediction results. Secondly, the results of lazy learning algorithms on FTS database were evaluated and elaborated.

4.1. Comparison of Local and Global Learning

To compare the global and lazy learning algorithms, selective CO oxidation, water gas shift reaction and steam reforming databases were modeled using both global and lazy learning techniques.

In global learning applications, leave one out cross validation technique was used for evaluating the performance of the models built on the train data. Out of m observations, (m-1) train data was used to evaluate performance for the left out one data observation point. This process was iterated over the whole dataset until there were m predicted values built on (m-1) train data.

In lazy learning applications, to predict and evaluated one observation, 200 observations were chosen; they were the nearest neighboring observations out of the m number of observations in the euclidean distance. The closest observations were trained and evaluated based on its predictive accuracy on the one observation. This process was iterated on the m number of observations.

To compare the model performance, RMSE and R^2 values were computed by using equation 3.2 and equation 3.3 respectively.

4.1.1. Comparison of Local and Global Learning in Selective CO Oxidation Dataset

In selective CO oxidation dataset, there are 1337 data points (observations) and 63 independent variables, which include active metal, preparation method, calcination conditions, support and promoter type, operational and other conditions. Conversion of CO was used as output, and prediction results were compared.

First, gradient boosting model was evaluated. Parameters of GBM (interaction depth, shrinkage, number of minimum observation in node and number of trees) were tuned by using hyper-parameter grid search.

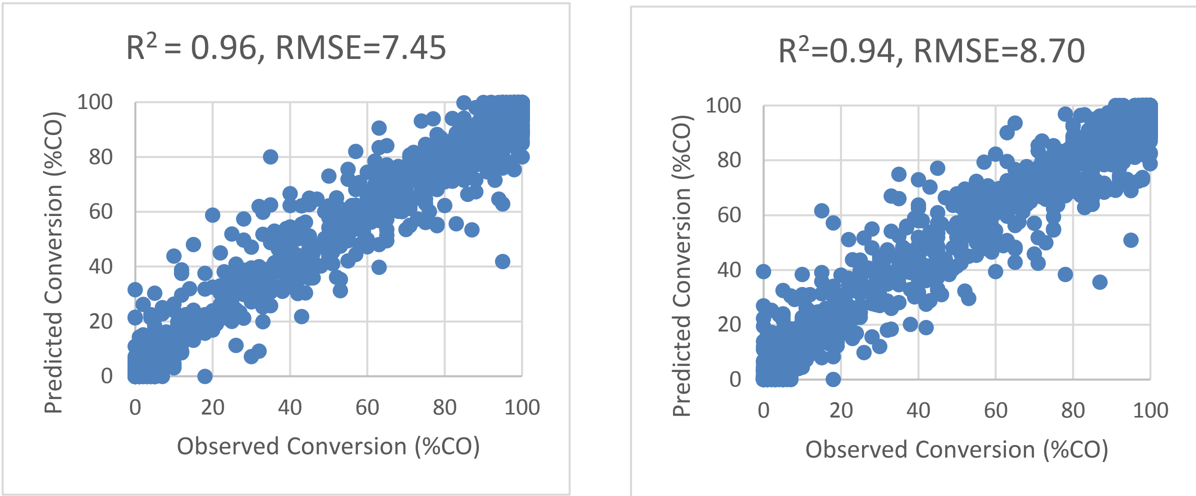


Figure 4.1. Observed vs. predicted CO conversion values for selective CO oxidation data with gradient boosting model by using (a) local learning algorithms (b) global learning algorithms

Figure 4.1a and 4.1b show the lazy learning and global learning results of selective CO oxidation database with gradient boosting modeling respectively. RMSE and corresponding R^2 are quite satisfactory for both algorithm training, although lazy learning results are slightly better than the global learning both in terms of RMSE and R^2.

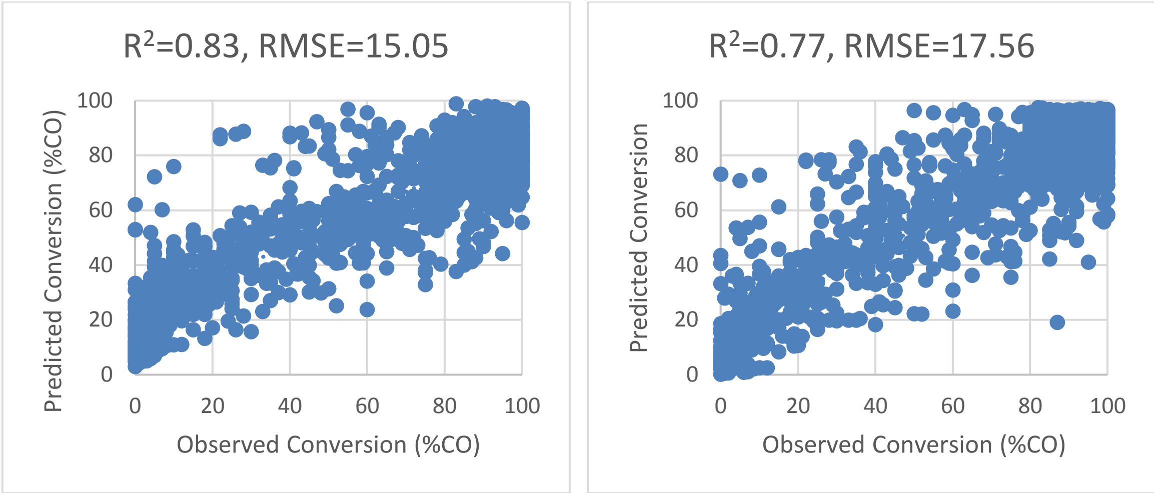


Figure 4.2. Observed vs. predicted CO conversion values for selective CO oxidation data with artificial neural networks by using (a) local learning algorithms (b) global learning algorithms

Figure 4.2a and 4.2b show the lazy learning and global learning results of selective CO oxidation database with artificial neural networks respectively. In comparison to GBM, ANN lower performance probably because of ANNs, in general, requires a larger amount of data to and have a high tendency to over-fit on train data unless regularized appropriately.

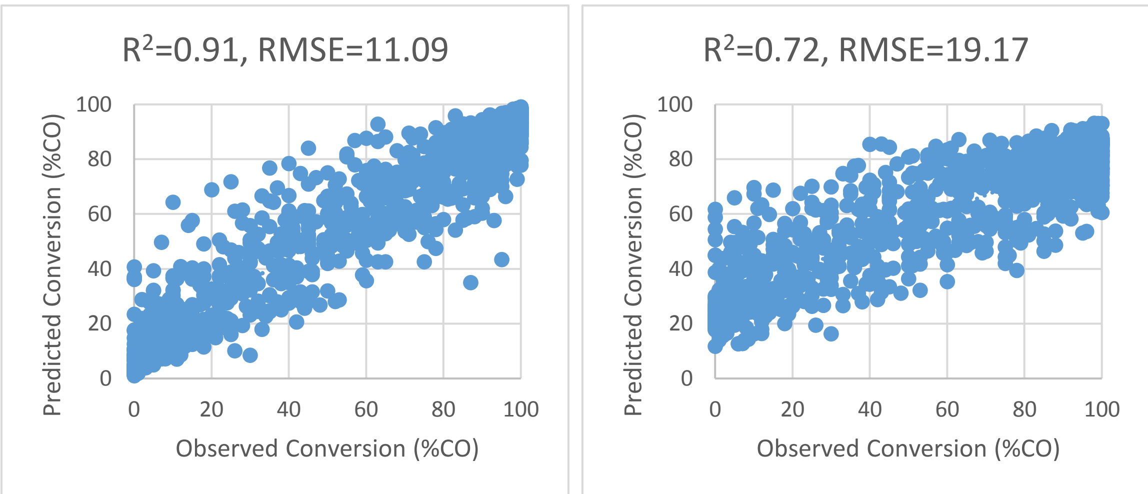


Figure 4.3. Observed vs. predicted CO conversion values for selective CO oxidation data with random forest by using (a) local learning algorithms (b) global learning algorithms

Figure 4.3a and 4.3b show the lazy learning and global learning results of selective CO oxidation database with random forest model respectively. Random Forest model has outperformed ANN but falls short of GBM in terms of model R^2 and RMSE in lazy learning and global learning respectively.

As the main results from this part, the lazy learning outperforms global learning in each different model, and GBM is better than other two methods.

4.1.2. Comparison of Local and Global Learning in Water Gas Shift Reaction Dataset

In Water Gas Shift (WGS) dataset, there are 4360 data points and 80 independent variables which include a base metal type and weight, catalyst preparation method, calcination conditions, support type, promoter type and weight and operation conditions. Conversion of CO was used as output, the prediction.

Local Learning **Global Learning**

(a) **(b)**

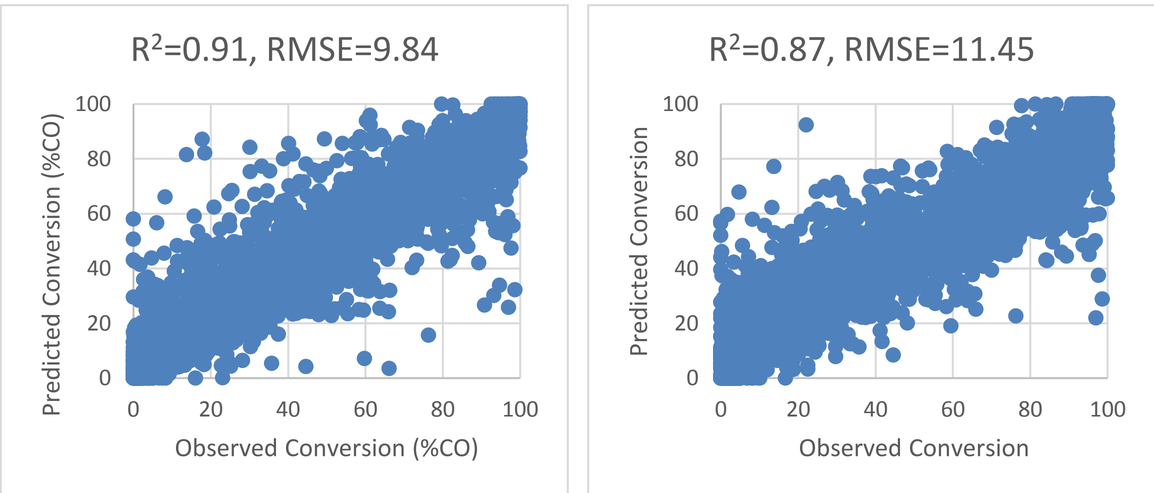
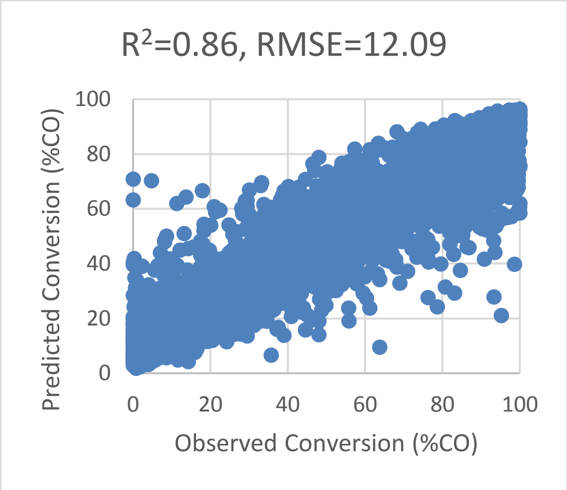


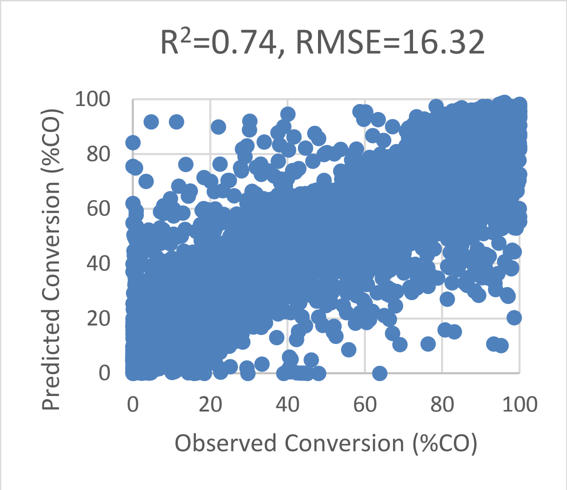
Figure 4.4. Observed vs. predicted CO conversion values for water gas reaction data with (a) Local learning and GBM (b) Global learning GBM (c) Local learning ANN (d) Global learning ANN (e) Local learning Random Forest (f) Global learning Random Forest

Local Learning **Global Learning**

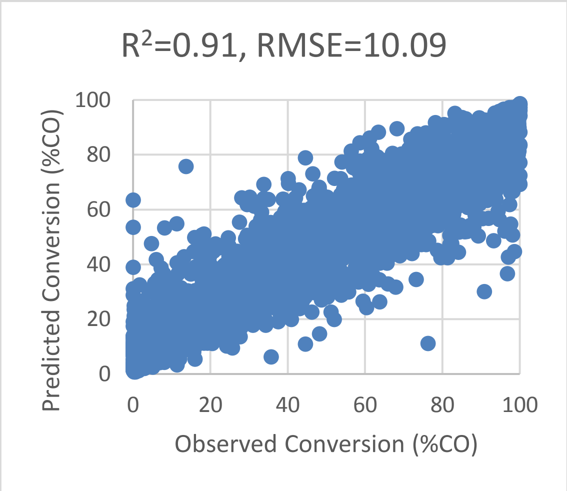
(c)



(d)



(e)



(f)

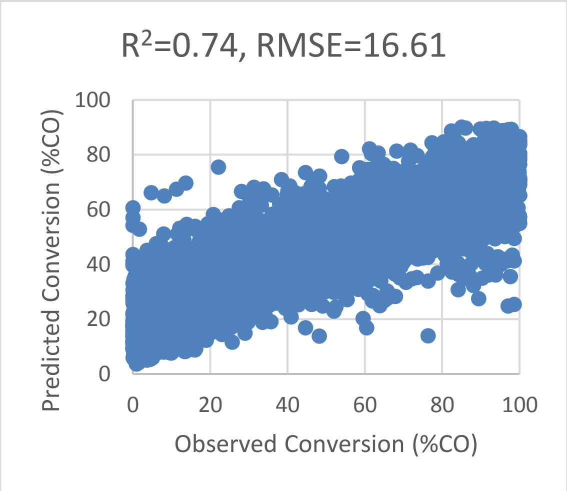


Figure 4.4. Observed vs. predicted CO conversion values for water gas reaction data with (a) Local learning and GBM (b) Global learning GBM (c) Local learning ANN (d) Global learning ANN (e) Local learning Random Forest (f) Global learning Random Forest (cont.)

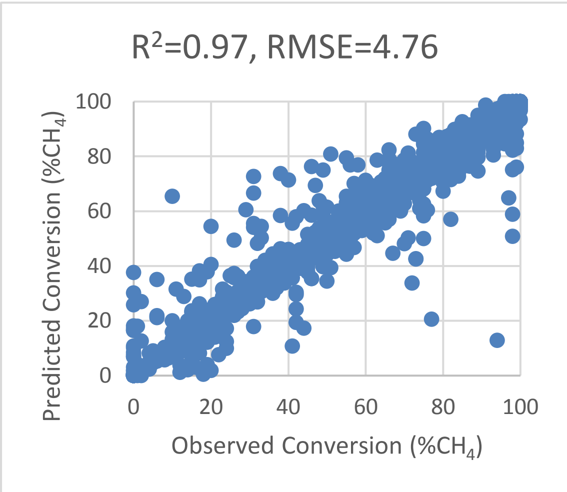
Figure 4.4a-f show the comparison of lazy and global learning methods on wager gas shift reaction database. It can be observed that lazy learning technique also outperforms global learning approaches for water gas shift data compared, and again GBM is better than ANN and Random Forest model.

4.1.3. Comparison of Local and Global Learning in Steam Reforming Dataset

In Steam Reforming dataset, there are 3426 data points and 48 independent variables which include a base metal type and weight, catalyst preparation method, calcination conditions, reduction conditions, supports (molar ratios), other parameters, reactor types. Conversion of CH_4 was used as the output for prediction.

Local Learning **Global Learning**

(a)



(b)

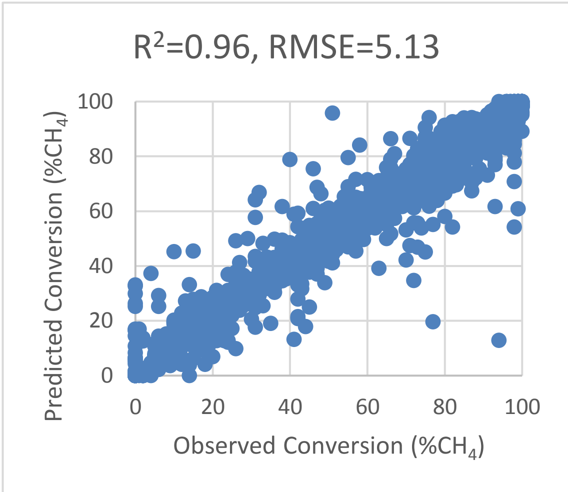
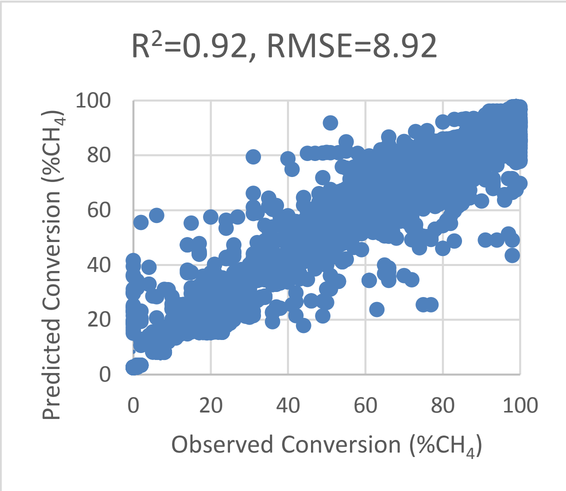


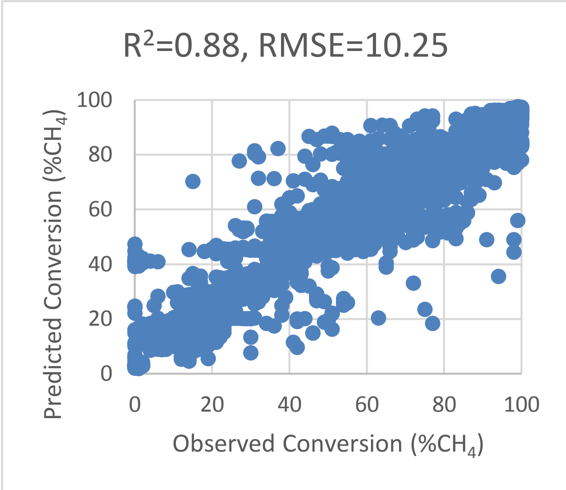
Figure 4.5. Observed vs. predicted CH4 conversion values for steam reforming data with (a) Local learning and GBM (b) Global learning GBM (c) Local learning ANN (d) Global learning ANN (e) Local learning Random Forest (f) Global learning Random Forest

Local Learning **Global Learning**

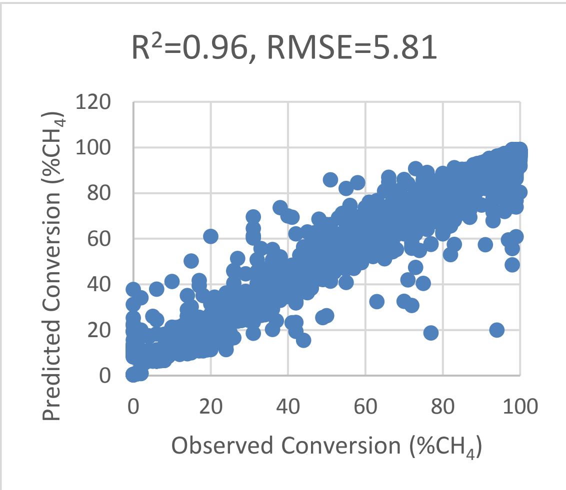
(c)



(d)



(e)



(f)

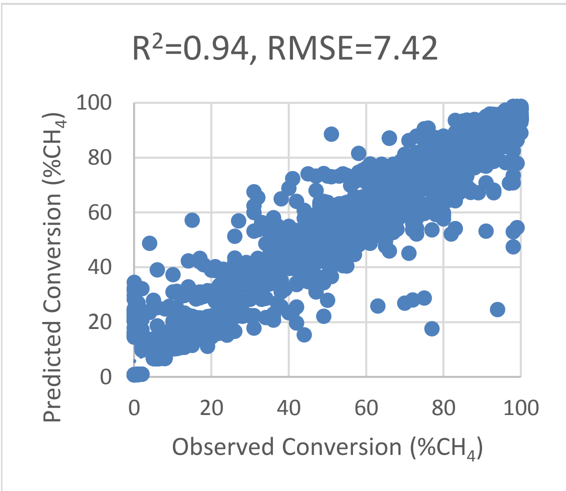


Figure 4.5. Observed vs. predicted CH4 conversion values for steam reforming data with (a) Local learning and GBM (b) Global learning GBM (c) Local learning ANN (d) Global learning ANN (e) Local learning Random Forest (f) Global learning Random Forest (cont.)

Comparison of lazy and global learning techniques on steam reforming database can be seen in Figure 4.5a-f. In steam reforming database, lazy learning technique works better than the global models, however, the difference is less significant for this case Gradient boosting model is also slightly better than random forest while ANN has the largest deviation from the actual data set.

4.2. Local Learning Algorithm on Fischer-Tropsch Synthesis

As it was evident from CO oxidation, water gas shift reaction and the steam reforming models, the local learning approach seems to better fit these kind problems and data sets considering that similar results were obtained for three datasets.

In the following sections, local learning techniques were applied on FTS database testing for various mini-batch sizes of shortest distance to each data point, and prediction results are compared in terms of R^2 and RMSE values. Finally, these local learning results are compared with global learning results for each method.

The batch size of the data is 2432 observations with 100 variables. Iteration was done over random mini-batches of observations sizing from 100 to 500. If an upwards trend for R^2 and RMSE were observed while increasing mini-batch size, larger sized random batches were tried to search for the optimal result.

GBM was applied to FTS data base with different numbers of query data and results are shown in Figure 4.6. The local learning model with 700 query data performs better than the other mini-batches models although their R^2 and RMSE are quite close to each other. R^2 value is observed as 0.87 while RMSE value, which is the ability of a network to predict the unseen data was obtained as 8.6. Both RMSE and corresponding R^2 values can be considered as satisfactory for such a complex process like FT.

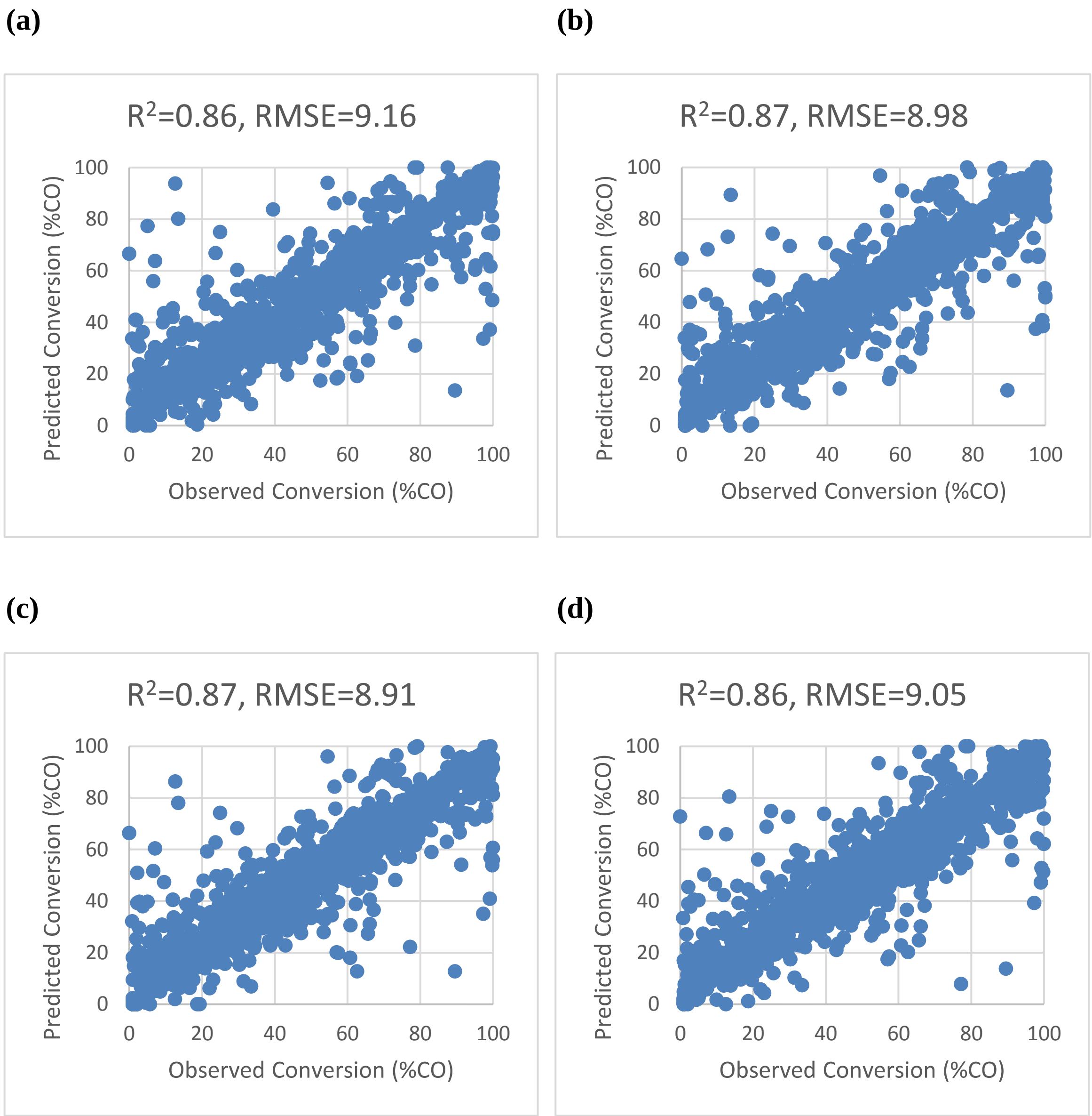
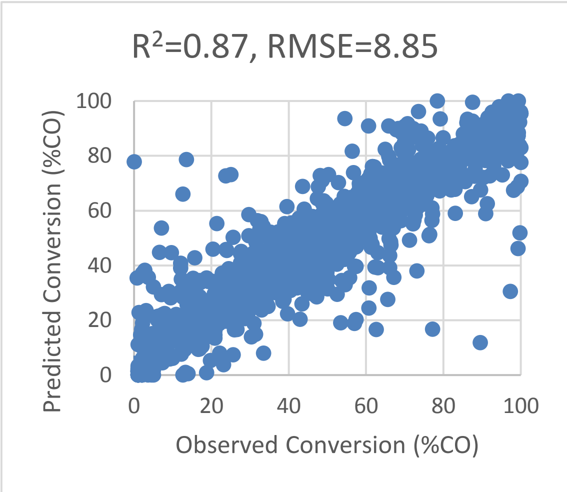
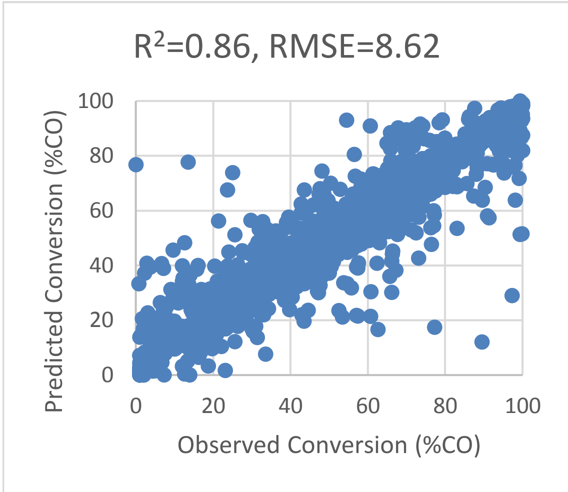


Figure 4.6. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by local learning and GBM methods with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data (f) 600 query data (g) 700 query data (h) 800 query data (i) 900 query data

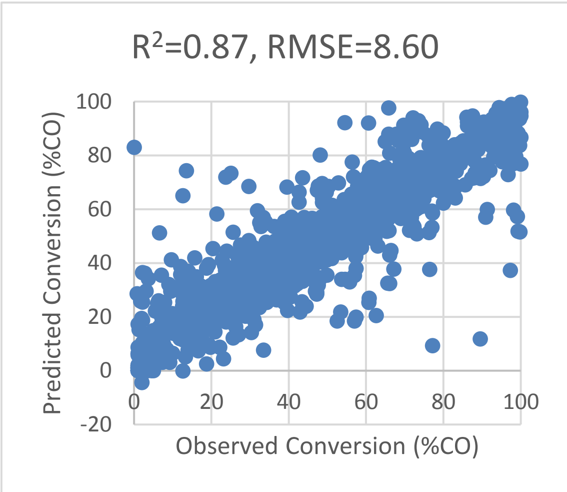
(e)



(f)



(g)



(h)

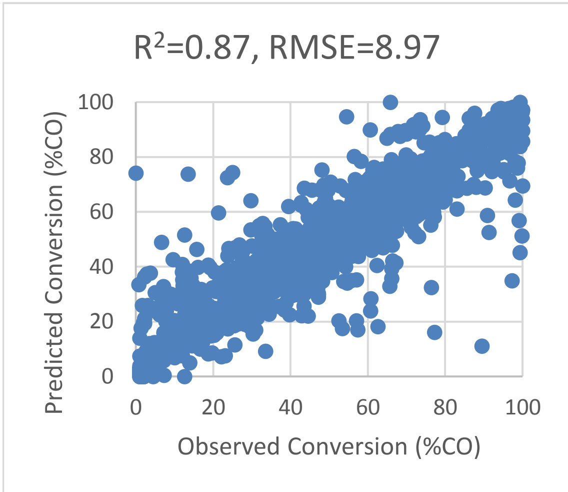


Figure 4.6. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by local learning and GBM methods with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data (f) 600 query data (g) 700 query data (h) 800 query data (i) 900 query data (cont.)

(i)

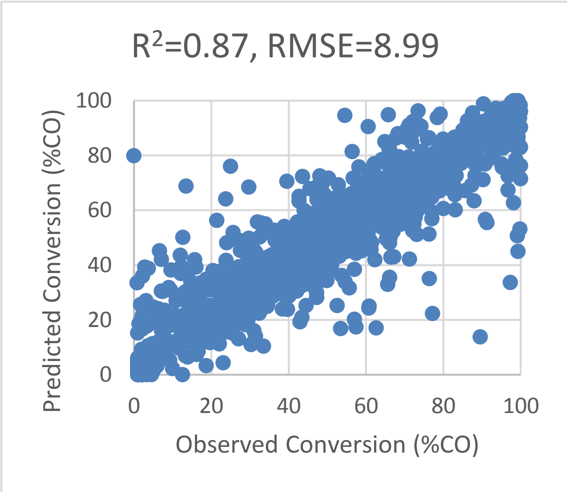


Figure 4.6. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by local learning and GBM methods with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data (f) 600 query data (g) 700 query data (h) 800 query data (i) 900 query data (cont.)

To compare with the results presented above, a global learning method was also developed for FTS database with gradient boosting method. As figure 4.7 shows, R^2 value was determined as 0.84 while RMSE value was computed as 9.5. It can be inferred from these results that the local learning application produces better results for FTS database if GBM was used; however, the improvement is not substantial. It is very rare that the training data is evenly or randomly distributed in the feature space for both local areas and global areas. Local learning approach locally adjusts the learning model to the local variation providing performance breakthroughs in many complex problems in comparison to global learning approaches where the variation is not evenly distributed (Bottou and Vapnik, 1992). The variation in the local areas in the training set did not significantly differ than the variation in FTS database global area; it was expected that local learning's error rate and performance would only slightly outperform the global error rate and performance. Local learning outperforms global learning significantly in datasets where the variations of local area significantly differ.

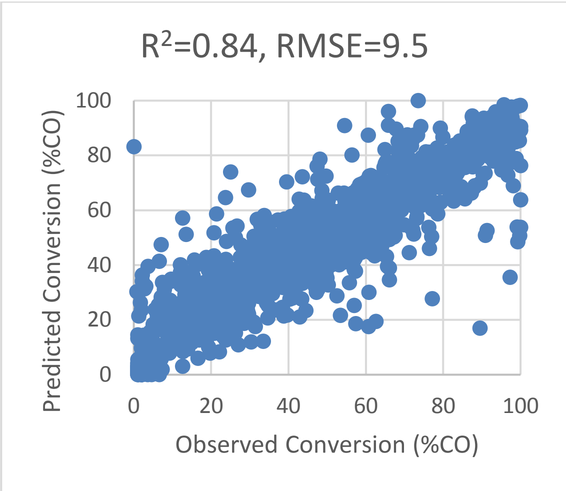


Figure 4.7. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by global learning and GBM methods

In addition to GBM, artificial neural networks were applied on FTS database and results are shown in Figure 4.8. R^2 ranges between 0.68 and 0.75 while RMSE takes values between 12 and 14. As the number of query points increases the R^2 and RMSE are in a steady range. An insight generated from application and tuning of the model is that GBM's results are not affected by the number of mini-batches. Unlike ANN where the batch size of similar observations affects the model accuracy due to increasing variance. Comparing to GBM, it can be observed that ANN shows relatively weak performance on FTS database as in the CO oxidation, water gas shift, and steam reforming case. In Figure 4.9, global learning with ANN modeling applied on FTS database. R^2 and RMSE values are observed as 0.63 and 14 respectively. Similar to GBM, NN performs slightly better on FTS database with local learning method than global learning method.

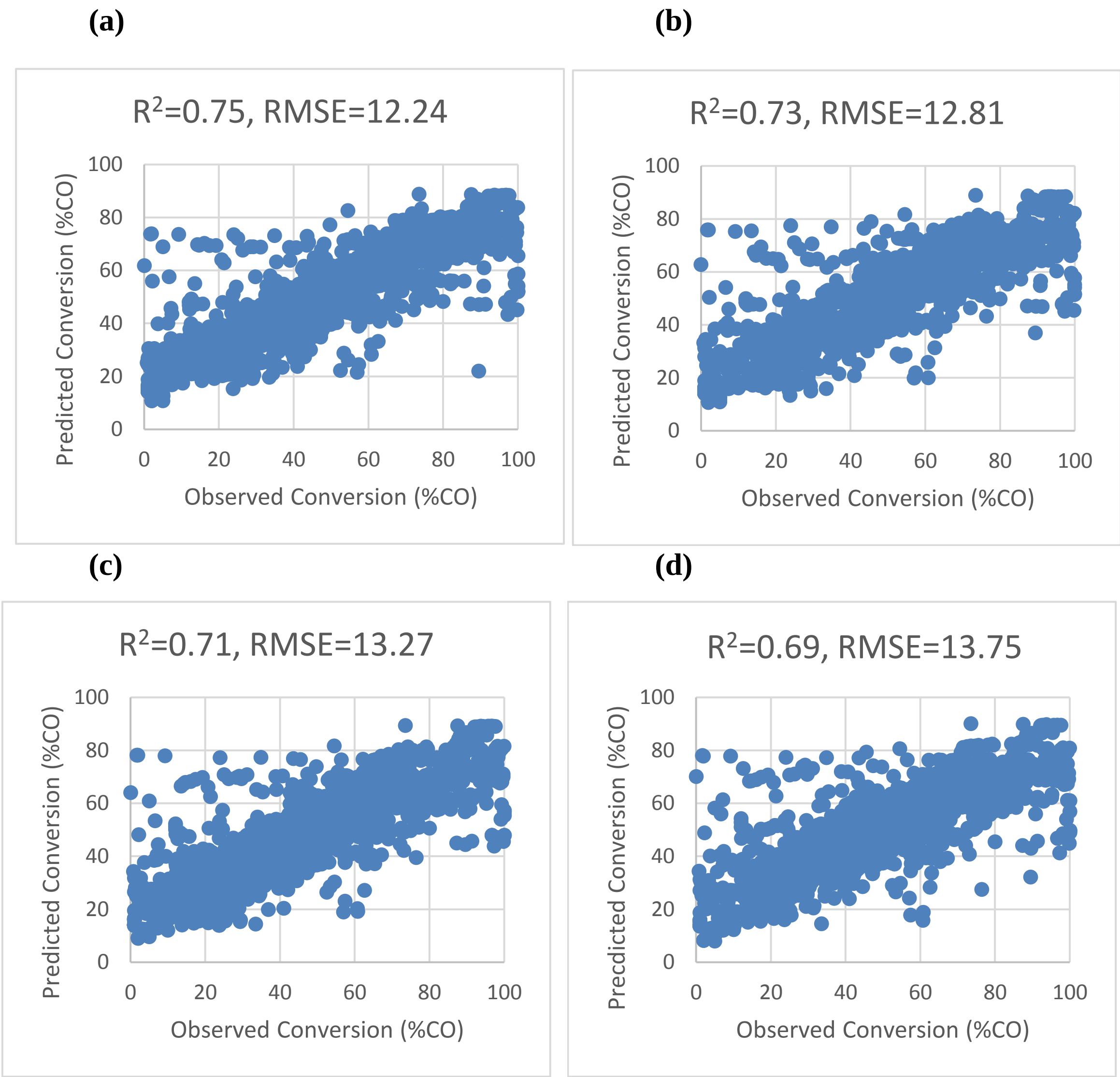


Figure 4.8. Observed vs. predicted CO conversion values for FTS by local learning and Artificial Neural Network with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data

(e)

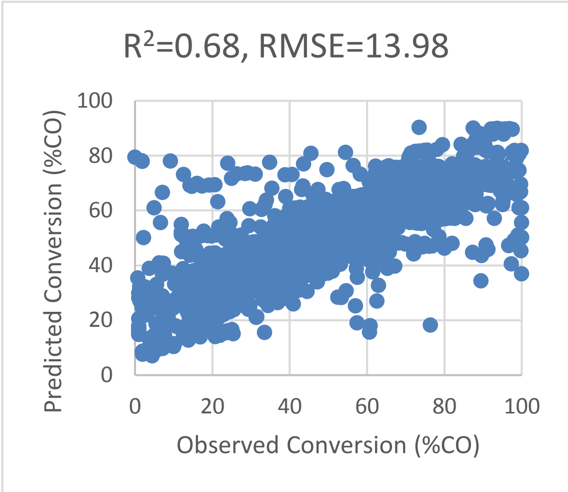


Figure 4.8. Observed vs. predicted CO conversion values for FTS by local learning and Artificial Neural Network with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data (cont.)

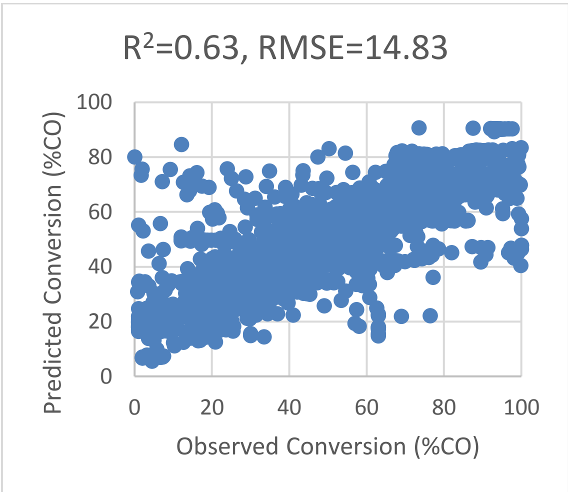


Figure 4.9. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by global learning and ANN methods

As the number of nearest neighbor query data increase, in other words, if more points are included with increasing distance or dissimilarity, the R^2, and RMSE of the developed model decreases, the model variance increases. The results on ANN depict a similar pattern of such increase in variance and therefore a decreased inaccuracy with the growing the number of observations in the data set.

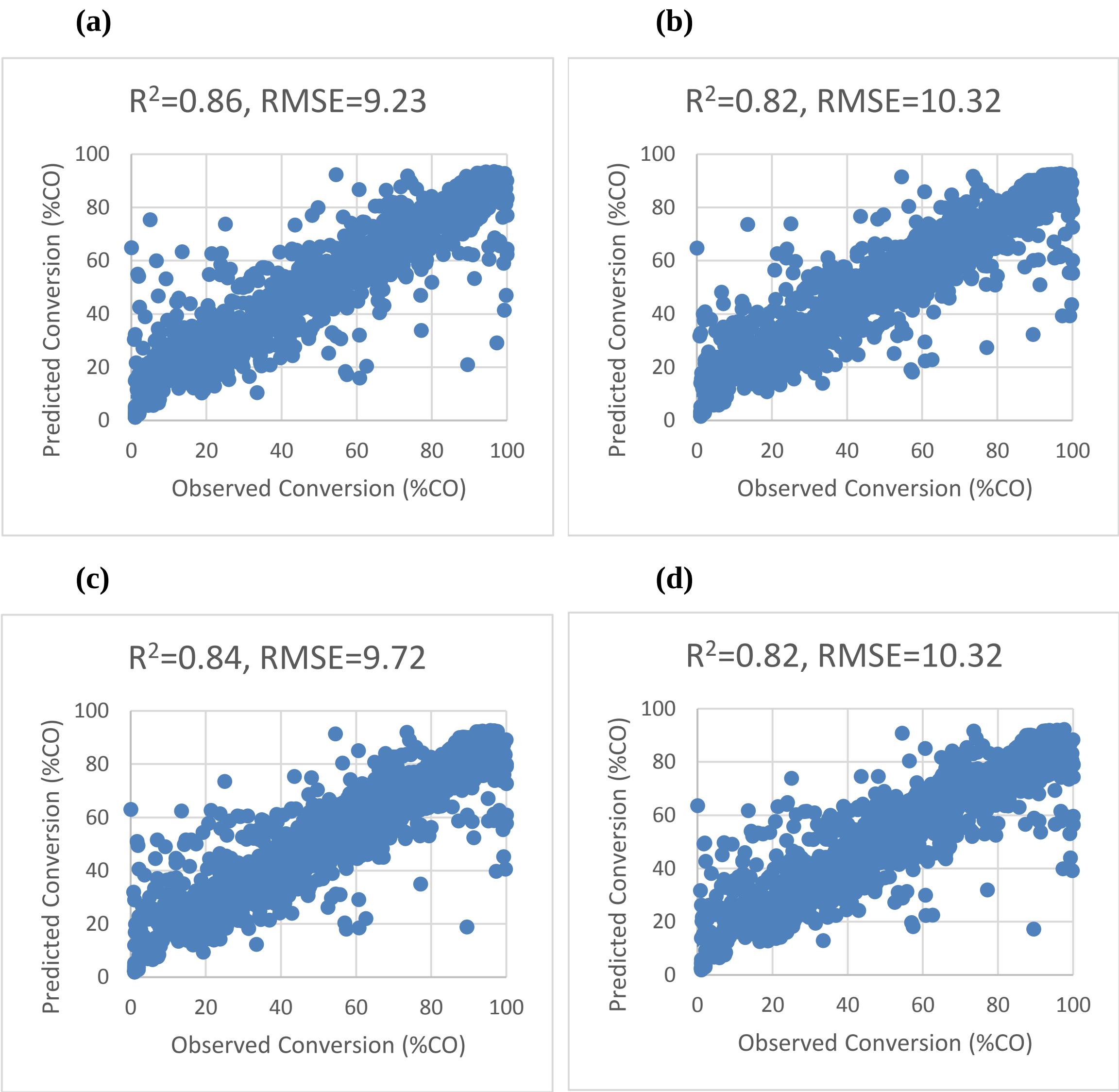


Figure 4.10. Observed vs. predicted CO conversion values for FTS by local learning and Random Forest with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data

(e)

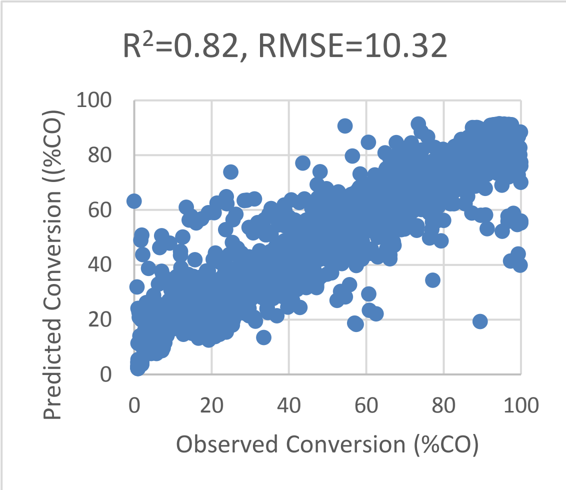


Figure 4.10. Observed vs. predicted CO conversion values for FTS by local learning and Random Forest with (a) 100 query data (b) 200 query data (c) 300 query data (d) 400 query data (e) 500 query data (cont.)

Finally, random forest was applied to FTS database using the local learning algorithms. Figure 4.10 showed the plots of experimental vs. predicted CO conversions with different numbers of query data with random forest. R^2 takes values between 0.82 and 0.86 while RMSE takes values between 9 and 10. 100 query data give the best results on random forest. Random forest performs better over the ANN models.

Global learning also applied to RF modeling and results are shown in Figure 4.11. R^2 and RMSE values are calculated as 0.77 and 11.65 respectively. These results indicate that global learning applications also show relatively weak performance on FTS database with Random Forest.

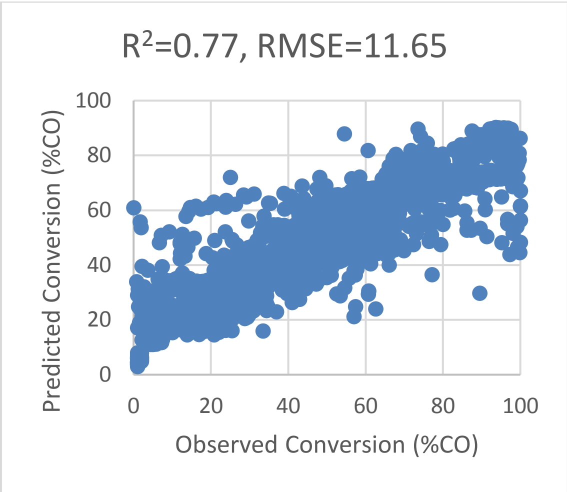


Figure 4.11. Observed vs. predicted CO conversion values for Fischer-tropsch synthesis by global learning and Random Forest methods

In choosing the best model for the chemical reaction dataset where the dataset's predictor features and observations are of dissimilar quantity compositions, an iterative train-test approach was used in finding the model with a low error for the datasets. In the FTS Problem, using FTS Database, the Gradient Boosted Model (GBM) performed better in both on the train and test sets among the three main models: Gradient Boosted Model, Random Forest and Neural Network in the experiment.

ANN did not perform well within the classical machine learning approaches compared in the scope of this research as good as the tree ensemble models for several reasons. First, the Neural Network is a very flexible and complex model but to foster its flexibility and power the model has to be trained for prolonged durations, requiring high computation power, furthermore with large amounts of input data. If there is not sufficient data the model will over fit. Thus, it will develop a high variance model. One method to decrease the variance would be to increase the number of observation however that is out of the scope of this research.

The tree ensemble methods perform relatively better because these models are by nature easy to train and difficult to over fit. These models have both the flexibility and complexity to overcome bias/variance trade-off. Random Forest uses fully grown decision trees which are by low nature bias but high variance models, the RF model tackles the error minimization by generating a forest of trees and using a majority voting approach to decrease the model's variance but cannot reduce bias for the chemical reaction dataset. Each tree in the forest of ensemble random forest model, outputs a prediction score where the total scores are ensemble to get a final score.

It is not easy to train all the trees at once. Instead, using an additive strategy: fix what the model has learned (one of the predictor trees has learned), and add a new tree once at a time to fix the prediction. Gradient boosting is not different than Random Forests in terms of trees and the ensemble structure, but the method of training is different for Gradient Boosting. Gradient Boosting is based on shallow trees as small as decision stumps, and it reduces the error by reducing the bias of the whole model by adding a new learner to improve the already trained learner by trying to find an optimal linear combination of tress. In the chemical reaction experiment's topology, the low number of observations and the non-linearity of the optimal decision boundary can be handled extremely well with the GBM. In problem settings where the number of observations is low, sparsity aware models such as the Gradient Boosted Model can outperform and scale solving chemical reaction problems.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusions

The aim of this study was to extract knowledge from experimental data using local learning tools to get a better knowledge about the Fischer-Tropsch synthesis and compare the results with global learning.

First, local and global learning tools with different modeling techniques (gradient boosting model, artificial neural network, and random forest) were applied to simpler datasets such as selective CO oxidation, water gas shift reaction and steam reforming to compare their abilities in such an analysis. Local learning found to be much more successful and used for the analysis of FTS database, which contained 4755 data points with 140 variables reviewed from 249 research articles. All the computational work was carried out in R 3.3.2 environment.

Gradient boosting, artificial neural network and random forest applied on FTS database both with local and global learning tools. Leave-one-out cross-validation was used for evaluating in training phase for the best model. In gradient boosting analysis, a number of trees, interaction depth, shrinkage and minimum observation in node were tuned and optimum model with tuned parameters was developed. For local learning, query data took values between 100 and 900. The optimum result obtained with 700 query data R^2 and RMSE values are observed as 0.87 and 8.6 accordingly. When global learning with gradient boosting model applied on FTS R^2 and RMSE values are observed as 0.84 and 9.5 accordingly.

In artificial neural network analysis, a number of units in the hidden layer and the regularization parameter were tuned in the train function. Best result obtained with 100 query data R^2 and RMSE values observed as 0.75 and 12 respectively. With global learning, on the other hand, R^2 and RMSE values of 0.63 and 14 were observed respectively.

Random forest was the last algorithm used for modeling the FTS database. In random forest, mtry, number of trees and node size are tuned. With local learning applications, R^2 and RMSE calculated as 0.86 and 9.23 respectively with 100 query data. When global learning applied on FTS with random forest modeling R^2 was decreased to 0.77 and RMSE was increased to 11.65.

To conclude, local learning applications can be successfully used to model and extract knowledge from the FTS publications in the literature. It was proven that for FTS database, local learning techniques give best results in terms of both R^2 and RMSE with gradient boosting model.

5.2. Recommendations

Based on the results obtained in this work, the recommendations for the future studies can be stated as follows;

- Further research to measure the discrepancy between local learning and global learning can be carried out in more chemical reaction datasets where distribution is different in local areas.

- Analysis can be carried out using other Gradient Boosting tools such as XGBoost and Microsoft’s LightGBM to outperform regular Ensemble Models.

REFERENCES

Adib, H., Haghbakhsh, R., Saidi, M., Takassi, M. A., Sharifi, F., Koolivand, M., . . . Keshtkari, S. (2013). Modeling and Optimization of Fischer-Tropsch Synthesis in the Presence of Co (III)/Al2O3 Catalyst Using Artificial Neural Networks and Genetic Algorithm. *Journal of Natural Gas Science and Engineering, 10,* 14-24.

Alpaydin, E. (2010). *Introduction to Machine Learning.* Cambridge, Massachusetts, USA: The MIT Press.

Avgouropoulos, G., Ioannides, T., Papadopoulou, C., Batista, J., Hocevar, S., & Metralis, H. K. (2002). A comparative study of Pt/γ-Al 2 O 3, Au/α-Fe 2 O 3 and CuO–CeO 2 catalysts for the selective oxidation of carbon monoxide in excess hydrogen. *Catalysis Today, 75*, 157-167.

Azzam, K. G., Babich, I. V., Seshan, K., and Lefferts, L. (2007). Bifunctional catalysts for single-stage water–gas shift reaction in fuel cell applications.: part 1. Effect of the support on the reaction sequence. *Journal of Catalysis,* 153-162.

Bartolomew, C. H. and Farrauto, R. J. (2006)). Hydrogen production and synthesis gas reactions. *Fundamentals of Industrial Catalytic Processes, Second Edition*, 339-486.

Baysal, M., Günay, M.E. and Yıldırım, R. (2017). Decision tree analysis of past publications on catalytic steam reforming to develop heuristics for high performance: A statistical review. *International Journal of Hydrogen Energy*, 243-254.

Bengaard, H. S., Nørskov, J. K., Sehested, J., Clausen, B. S., Nielsen, L. P., Molenbroek, A. M., & Rostrup-Nielsen, J. R. (2002). Steam reforming and graphite formation on Ni catalysts. *Journal of Catalysis,* 365-384.

Bontempi, G., Birattari, M., and Bersini, H. (1999). Lazy learning for local modelling and control design. *International Journal of Control, 72*, 643-658.

Bottou, L. and Vapnik, V. (1992). Local learning algorithms. *Neural computation,* 888-900.

Breiman, L. (2001). Random forests. *Machine learning*, 5-32.

Burnak, B. (2015). *Analysis of Published Data in the Literature to Extract Knowledge for Fischer-Tropsch Syntehis.* İstanbul: Bogazici University.

Callan, R. (1999). *The Essence of Neural Networks.* New York: Prentice Hall Europe.

Choudhary, T. V. and Goodman, D. W. (2002). Oxidation catalysis by supported gold nano-clusters. *Topics in Catalysis, 21,* 25-34.

Cios, K. J., Pedrycz, W., Swiniarski, R. W. and Kurgan, L. A. (2007). *Data mining: a knowledge discovery approach.* Springer Science & Business Media.

Dalai, A. K. and Davis, B. H. (2008). Fischer-Tropsch Synthesis: A Review of Water Effects on the Performances of Unsupported and Supported Co Catalysts. *Applied Catalysis A: General, 348,* 1-15.

Elith, J., Leathwick, J. R. and Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology,* 802-813.

Francesconi, J. A., Mussati, M. C., Mato, R. O. and Aguirre, P. A. (2007). Analysis of the energy efficiency of an integrated ethanol processor for PEM fuel cell systems. *Journal of Power Sources, 167*, 151-161.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 367-378.

Gallucci, F., Comite, A., Capannelli, G. and Basile, A. (2006). Steam reforming of methane in a membrane reactor: an industrial case study. *Industrial & engineering chemistry research*, 2994-3000.

Gallucci, F., Paturzo, L., Fama, A. and Basile, A. (2004). Experimental study of the methane steam reforming reaction in a dense Pd/Ag membrane reactor. *Industrial & engineering chemistry research*, 928-933.

Galvis, H. M., Koeken, A. C., Bitter, J. H., Davidian, T., Ruitenbeek, M., Dugulan, A. I. and de Jong, K. P. (2013). Effects of Sodium and Sulfur on Catalytic Performance of Supported Iron Catalysts for the Fischer-Tropsch Synthesis of Lower Olefins. *Journal of Catalysis*, 22-30.

Giudici, P. and Castelo, R. (2003). Improving Markov chain Monte Carlo model search for data mining. *Machine learning, 50*, 127-158.

Guyon, I., Vapnik, V. N., Boser, S. E., Bottou, L. and Solla, S. A. (1992). Structural risk minimization for character recognition. *In Advances in neural information processing systems*, 471-479.

Guyon, I., Vapnik, V., Boser, B., Bottou, L. and Solla, S. A. (1992). Structural risk minimization for character recognition. *In Advances in neural information processing systems*, 471-479.

Günay, M. E. and Yıldırım, R. (2010). Analysis of selective CO oxidation over promoted Pt/Al 2 O 3 catalysts using modular neural networks: Combining preparation and operational variables. *Applied Catalysis A: General*, 174-180.

Günay, M. E. and Yıldırım, R. (2011). Neural network analysis of selective CO oxidation over copper-based catalysts for knowledge extraction from published data in the literature. *Industrial & Engineering Chemistry Research*, 12488-12500.

Hattori, T. and Kito, S. (1995). Neural Network as a Tool for Catalyst Development. *Catalysis Today, 23*, 347-355.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013). *An introduction to statistical learning.* New York: Springer.

Jones, G., Jakobsen, J. G., Shim, S. S., Kleis, J., Andersson, M. P., Rossmeisl, J. and Rostrup-Nielsen, J. R. (2008). First principles calculations and experimental insight into methane steam reforming over transition metal catalysts. *Journal of Catalysis*, 147-160.

Khajeh-Hosseini-Dalasm, N., Ahadian, S., Fushinobu, K., Okazaki, K. and Kawazoe, Y. (2011). Prediction and analysis of the cathode catalyst layer performance of proton exchange membrane fuel cells using artificial neural network and statistical methods. *Journal of power sources, 196*, 3750-3756.

Kwack, S. H., Bae, J. W., Park, M. J., Kim, S. M., Ha, K. S. and Jun, K. W. (2011). Reaction Modeling on the Phosphorous-Treated Ru/Co/Zr/SiO2 Fischer-Tropsch Catalyst with the Estimation of Kinetic Parameters and Hydrocarbon Distribution. *Fuel, 90*, 1383-1394.

Li, X. (2005). Diversification and localization of energy systems for sustainable development and energy security. *Energy Policy*, 2237-2243.

Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. *R News*, 18-22.

Lögdberg, S., Tristantini, D., Borg, Ø., Ilver, L., Gevert, B., Järås, S. and Holmen, A. (2009.). Hydrocarbon Production Via Fischer-Tropsch Synthesis from H2-Poor Syngas over Different Fe-Co/γ-Al2O3 Bimetallic Catalysts. *Applied Catalysis B: Environmental*, *89*, 167-182.

Lu, Y. and Lee, T. (2007). Influence of the Feed Gas Composition on the Fischer-Tropsch Synthesis in Commercial Operations. *Journal of Natural Gas Chemistry*, *16*, 329-341.

Manasilp, A. and Gulari, E. (2002). Selective CO oxidation over Pt/alumina catalysts for fuel cell applications. *Applied Catalysis B: Environmental*, *37*, 17-25.

Mirzaei, A. A., Vahid, S. and Torshizi, H. O. (2013). Effect of Support and Promoter on the Catalytic Performance and Structural Properties of the Fe-Co-Ni Catalysts for Co Hydrogenation. *Journal of Natural Gas Science and Engineering*, 106-117.

Nurunnabi, M., Murata, K., Okabe, K., Inaba, M. and Takahara, I. (2008). Performance and Characterization of Ru/Al2O3 and Ru/SiO2 Catalysts Modified with Mn for Fischer-Tropsch Synthesis. *Applied Catalysis A: General*, 203-211.

Odabasi, C., Gunay, M. E. and Yildirim, R. (2014). Knowledge extraction for water gas shift reaction over noble metal catalysts from publications in the literature between 2002 and 2012. *International Journal of Hydrogen Energy*, 5733-5746.

Rokach, L. and Maimon, O. (2008). *Data mining with Decision Trees: Theory and Applications.* Hackensack, NJ, USA: World Scientific.

Romero, C. and Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005 . *Expert Systems with Applications, 33*, 135-146.

Rostrup-Nielsen, J. R. (1993). Production of synthesis gas. *Catalysis today*, 305-324.

Röper, M. (1983). "Fischer-Tropsch synthesis." Catalysis in C1 Chemistry. *Springer Netherlands*, 41-88.

Rud, O. P. (2001). *Data mining cookbook: modeling data for marketing, risk, and customer relationship managemen.* John Wiley & Sons.

Sharma, B. K., Sharma, M. P., Kumar, S., Tendulkar, S. B. and Tambe, S. S. (1998). Fischer-Tropsch Synthesis with Co/SiO2–Al2O3 Catalyst and Steady-State Modeling Using Artificial Neural Networks. *Fuel, 77*, 1763-1768.

Shimura, K., Miyazawa, T., Hanaoka, T. and Hirata, S. (2014). Preparation of Co/Al2O3 Catalyst for Fischer-Tropsch Synthesis: Combination of Impregnation Method and Homogeneous Precipitation Method. *Applied Catalysis A: General, 475*, 1-9.

Son, I. H. and Lane, A. M. (2001). Promotion of Pt/γ-Al2O3 by Ce for preferential oxidation of CO in H2. *Catalysis letters*, *76*, 151-154.

Song, C. (2002). Fuel processing for low-temperature and high-temperature fuel cells Challenges and opportunities for sustainable development in the 21st centry. *Catalysis Today, 77*, 17-49.

Tanksale, A., Beltramini , J. and Lu, G. (2010). A review of catalytic hydrogen production processes rom biomass. *Renew Sustain Energy Rev*, 166-82.

Trépanier, M., Dalai, A. K. and Abatzoglou, N. (2010). Synthesis of CNT-Supported Cobalt Nanoparticle Catalysts Using a Microemulsion Technique: Role of Nanoparticle Size on Reducibility, Activity and Selectivity in Fischer-Tropsch Reactions. *Applied Catalysis A: General*, *374*, 79-86.

Tufféry, S. (2011). *Data mining and statistics for decision making* (Vol. 2). Chichester: Wiley.

Vapnik, V. (1992). Principles of risk minimization for learning theory. *In Advances in neural information processing systems*, 831-838.

Widmer, G. and Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 69-101.

Zhai, Q., Xie, S., Fan, W., Zhang, Q., Wang, Y., Deng, W. and Wang, Y. (2013). Photocatalytic conversion of carbon dioxide with water into methane: platinum and copper (I) oxide co-catalysts with a core–shell structure. *Angewandte Chemie, 125*, 5888-5891.

Zhang, Q., Deng, W. and Wang, Y. (2013.). Recent Advances in Understanding the Key Catalyst Factors for Fischer-Tropsch Synthesis. *Journal of Energy Chemistry, 22*, 27-38.

Zhang, Q., Kang, J. and Wang, Y. (2010). Development of novel catalysts for Fischer–Tropsch synthesis: tuning the product selectivity. *ChemCatChem, 2*, 1030-1058.