İZMİR KÂTİP ÇELEBİ ÜNİVERSİTESİ
2010
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms

Submitted to the Graduate School of Natural and Applied Sciences
in partial fulfillment of the requirements for the degree of

Master of Science

in Electrical & Electronics Engineering

by
Emre ÇANCIOĞLU

ORCID 0000-0002-9918-4668

June, 2022

This is to certify that we have read the thesis **Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms** submitted by **Emre ÇANCIOĞLU**, and it has been judged to be successful, in scope and in quality, at the defense exam and accepted by our jury as a MASTER'S THESIS.

APPROVED BY:

Advisor:	**Assoc. Prof. Savaş ŞAHİN** İzmir Kâtip Çelebi University	……………..
Co-advisor:	**Assoc. Prof. Yalçın İŞLER** İzmir Kâtip Çelebi University	……………..

Committee Members:

	Assoc. Prof. Mehmet Emre ÇEK Dokuz Eylül University	……………..
	Asst. Prof. Mustafa Berkant SELEK Ege University	……………..
	Asst. Prof. Kamil ÇETİN İzmir Kâtip Çelebi University	……………..

Date of Defense: June 13, 2022

Declaration of Authorship

I, **Emre ÇANCIOĞLU**, declare that this thesis titled **Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms** and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for the Master's degree at this university.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. This thesis is entirely my own work, with the exception of such quotations.
- I have acknowledged all major sources of assistance.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signature:

____

Date: 13.06.2022

____

Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms

Abstract

This thesis presents diagnose and detect faults of Tennessee Eastman Process (TEP) with machine learning algorithms via Poincaré plots-based feature extraction and statistically analysis-based feature selection. The IEEEDataPort online dataset, obtained from a big plant that contains nonlinear processes from various chemical units, is utilized in this thesis. It contains measures from 52 process points in TEP with 20 dissimilar malfunction types. In this study, raw dataset and dataset that applied feature extraction and feature selection was used. Poincaré plot was applied to the dataset for feature extraction so that four commonly used nonlinear features for every measurement point were calculated. After that, among these features, the features that show a statistically significant difference (alpha = 0.05) between failure types were selected. The machine learning tools such as Decision Tree, Discriminant Analysis, Naive Bayes, k-Nearest Neighbors, Support Vector Machine, and Ensemble Learning algorithms were utilized to classify the fault types from both datasets. The maximum classifier accuracies were 89.5% for the whole feature dataset using the Subspace Discriminant Algorithm of the Ensemble Learning Classifier method and 93.5% for the selected features only using the Linear Discriminant Analysis during this study. These performances could be comprehendible among the results achieved in similar studies.

Keywords: Tennessee Eastman Process, Fault Detection and Diagnosis, Poincaré Plot Based Feature Extraction, One-way ANOVA Based Feature Selection, Machine Learning Tools.

Çeşitli Makine Öğrenimi Algoritmalarını Kullanarak Tennessee Eastman Sürecinde Poincaré Grafik Tabanlı Hata Tespiti

Öz

Bu tez, Poincaré grafiğine dayalı öznitelik çıkarımı ve istatistiksel olarak analize dayalı öznitelik seçimi yoluyla makine öğrenme algoritmaları ile Tennessee Eastman Süreci (TEP) hatalarını teşhis ve tespit etmeyi sunar. Bu çalışmada, çeşitli kimyasal birimlerden doğrusal olmayan süreçleri içeren bir prosesten elde edilen IEEEDataPort çevrimiçi veri seti kullanılmıştır. 20 farklı arıza tipi ile TEP'de 52 proses noktasından alınan önlemleri içerir. Bu çalışmada, öznitelik çıkarımı ve öznitelik seçimi uygulanan ham veri seti ve veri seti kullanılmıştır. Her ölçüm noktası için yaygın olarak kullanılan dört doğrusal olmayan özellik hesaplanacak şekilde, özellik çıkarımı için veri kümesine Poincaré grafiği uygulandı. Daha sonra bu öznitelikler arasından hata türleri arasında istatistiksel olarak anlamlı fark (alfa = 0.05) gösteren öznitelikler seçilmiştir. Her iki veri setinden de hata türlerini sınıflandırmak için Karar Ağacı, Diskriminant Analizi, Naive Bayes, k-En Yakın Komşular, Destek Vektör Makinesi ve Topluluk Learning algoritmaları gibi makine öğrenme araçları kullanılmıştır. Topluluk Learning Sınıflandırıcı yönteminin Altuzay Ayırım Algoritması kullanılarak tüm özellik veri kümesi için maksimum sınıflandırıcı doğruluğu %89,5 ve bu çalışma sırasında yalnızca Doğrusal Ayırım Analizi kullanılarak seçilen öznitelikler için %93,5 olmuştur. Bu performanslar, benzer çalışmalarda elde edilen sonuçlar arasında anlaşılabilir bir sonuç olabilir.

Anahtar Kelimeler: Tennessee Eastman Süreci, Arıza Tespiti ve Teşhisi, Poincaré Çizimi Tabanlı Özellik Çıkarma, Tek Yönlü ANOVA Tabanlı Özellik Seçimi, Makine Öğrenimi Araçları.

To my family…

Acknowledgment

Firstly, I would like to express my sincere gratitude to my supervisor, Assoc. Prof. Savaş ŞAHİN and co-supervisor Assoc. Prof. Yalçın İŞLER, for their confidence, patience, valuable suggestions, and encouragement. Thank you for giving tirelessly of your time, ideas, expertise, and helping me to achieve my goal.

I would like to extend my sincere thanks to my committee members Assoc. Prof. Mehmet Emre ÇEK, Asst. Prof. Mustafa Berkant SELEK, and Asst. Prof. Kamil ÇETİN, for the benefit of their collective wisdom, helpful comments to guide me through the master's program.

I would like to offer my special thanks to my parents for their endless support and constant love throughout my life. I have never been able to find suitable words that describe their unwavering support and belief in me.

Finally, I am deeply grateful to my colleagues, Özgür DENİZ, and Alaattin YILDIRIM for their encouragement, assistance, and tolerance.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

TEP	Tennessee Eastman Process
IoT	Internet of Things
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
WPE	Wavelet Packet Energy
CNN	Convolutional Neural Network
SVM	Support Vector Machine
SCADA	Supervisory Control and Data Acquisition
ANOVA	Analysis of Variance
FFT	Fast Fourier Transform
WA	Wavelet Analysis
ANN	Artificial Neural Network
LMS	Large Memory Storage
CART	Classification and Regression Trees
CHAID	Chi-square Automatic Interaction Detector
LDF	Linear Discriminant Function
QDF	Quadratic Discriminant Function

Chapter 1

Introduction

In this thesis, diagnose predictions and upcoming fault detections of Tennessee Eastman Process (TEP) is fulfilled with machine learning algorithms by using both Poincaré plots-based feature extraction and statistical analysis-based feature selection. TEP is complex-process-control-system containing nonlinear chemical units such as separators, strippers, reactors and products [1]. The use of maintenance technologies is important to continue uninterruptedly mass production process with minimum loss for industries. These maintenance methods are made of three sub-classifications such as periodic, predictive and after failures [2]. Periodic maintenance ensures that the machine parts should be examined after a certain period duration and it might be replaced if necessary. Predictive maintenance, unlike periodic maintenance, aims prediction of fault detection via monitoring the conditions acquired plant process data. After failure method is to fix something such as equipment, motor, sensor that has malfunctioned. One of the maintenance requirements is to extend the remaining life of the machine element or to prolong the failure time. Therefore, the implemented maintenance policies are correctly expected to reduce service and negative interruption consequences [3], and the mass production becomes faster and human-induced errors minimized [4]. Nowadays, Industry 4.0 and internet of things technology (IoT) have been used for predictive maintenances including machine learning (ML), deep learning (DL), and artificial intelligence (AI) [5, 6]. In order to develop predictive maintenance methods, machine-learning ones might be used with the feature extraction, and feature selection stages [7].

As for the feature extraction, it is to minimize the number of features by generating features from the obtaining data or existing ones [8]. Ding X. et al. [9] developed a feature extraction method for fault detection in shaft bearings by combining wavelet packet energy (WPE) and deep convolutional neural network (CNN) based on energy wave feature learning. WPE first transformed it into a 2-D image according to the brightness and then classified it with the deep CNN. As a result of testing the proposed method on 6 data sets and 7 different features, it was seen that the success rate was between around 96-99%. Simona M. et al. [10] developed Poincare variables for feature extractions of Wi-Fi protocols such as IEEE 802.11n and IEEE 802.11ac.

When it is come to feature selection, its aim is to find the minimal data set by choosing the necessary, important and meaningful features in the data set [11]. Senoussi H. et al. [12] proposed feature selection methods in TEP with four feature selection methods including Correlation Based Feature Selection, Fast Correlation Based Feature Selection, Minimum Redundancy Maximum Relevance and Interacting Features Based Feature Selection. Yangtao X et al. [13] published that the linear kernel Support Vector Machine (SVM) Recursive Feature Selection has been applied to fault diagnosis. Şahin Ş. and İşler Y. [14] were performed the ways to perform supervisory control and data acquisition (SCADA) and robotics experiments at a reasonable cost in control and automation courses, and they quantitatively evaluated the presented experiment results using one-way Analysis of Variance (ANOVA) test. Şahin Ş. and İşler Y. [15] designed test and experimental setups to show how an economically viable SCADA system could be set up, and examined the students' performances. A quantitative evaluation was made using the one-way ANOVA test on the exam results of the students and a comparison was made with respect to previous years.

As for fault detection methods, they might be divided into three classes such as model-based, information-based and signal-based fault detection [16]. In the model-based fault detection, a model of the system is composed and the fault of the system might be detected by changing its parameters [17,18]. The information-based fault detection method is based on the analysis of a large amount of real and/or modeled system data. The signal-based fault detection method can be explained with analyzing and evaluating the signals obtained from the system via Fast Fourier Transform (FFT) and Wavelet Analysis (WA) [19]. ML and DL algorithms are used for fault detection and fault diagnosis in various applications such as bearing faults, motor faults, control panel faults etc. [20-22]. Haidong S. et al. [23] proposed deep CNN using the data of electric locomotive roller bearing’s fault diagnosis. After the comparison other deep

Artificial Neural Network (ANN) performances, the proposed algorithm had a success rate of 97.43% in a shorter time. Wen, et al. [24] presented DL method using sparse auto-encoder for fault diagnosis. In this method, by applying penalty points to neurons in the hidden layer with the help of a three-layer sparse auto-encoder, it showed a higher performance than traditional auto-encoders. Guo L. et al. [25] proposed an intelligent deep CNN for machine failures with unlabeled data. They formed with two modules as situation recognition and field adaptation. To learn features from raw vibration data in the state recognition module, a one-dimensional CNN was created and a health state classifier was designed. They showed that the method they presented performed 32.1% more accurate classification compared to traditional methods. He M. et al [26] designed a neural network using Large Memory Storage (LMS) structure to perform diagnostics on big data. In the presented structure, the signals were formerly pre-processed by the Short-time Fourier transform method, the designed neural network was latterly trained and used for fault diagnosis. They achieved the classification accuracies of 80-88% for the CNN and 96-98% for the LMS networks. In other studies, fault detection for electrical motors using machine learning techniques is getting popular [27]. Motor current signal analysis is one of the used methods for diagnosing motor errors. Since induction motors are symmetrical electromagnetic systems, this situation causes eccentricity to occur between the rotor and stator, resulting in errors such as winding, bearing, and rotor [28, 29]. Most of the proposed approaches to detect bearing failures use vibration signals. Although vibration signals are powerful in identifying this error type, sensor selection and localization affect the performance of the detection and diagnostic. Therefore, these errors create certain changes in the stator current. In this proposed method, the effect of bearing failure on both current and vibration signals is investigated [30, 31]. In another interesting study, Taştimur C. et al. [32] used images of the obtained vibration signals. They divided the signal data into 100 segments and 400 pixels wide. They classified the bearing failures in four different categories using CNN from these images' segments. They tested the model they presented with 400 training data, 60 validation data, and 100 test data. They reported an accuracy of 100% with this limited data set. As for the principal component analysis (PCA) based fault detection, Ammar A. S. et al. [33] and Aldrich, C et al. [34] nonlinear dynamical PCA based feed forward neural network is applied for fault detection in TEP.

In this thesis, dataset of the TEP complex-process-control-system containing nonlinear chemical units such as separators, strippers, reactors and products used in order to analyze the proposed fault detection techniques. Poincare Plot measurements (i.e., Standard Deviation 1 (SD1) and Standard Deviation 2 (SD2)) values, which have achieved successful results especially in biomedical applications [35-37], were used for the first time in such an application for feature extraction from the raw data set. One-way ANOVA method [38, 39], which is used in fault detection and diagnosis and with successful results, was used in the thesis to find the optimal data set with the statistical-based feature selection method that will help effective learning. As a result of this feature selection method, since it was determined that the SD1 measurements of the sensors 24, 26, 32, 37, 39, 40, and 41 did not show a statistically significant difference, these features were removed from the data set, and a new optimal data set was created. Different L algorithm such as Decision Tree, Discriminant, Naive Bayes Algorithms, SVM, k-Nearest Neighbors (k-NN) and Ensemble Learning algorithms are applied for both raw dataset and optimal dataset. Then, the accuracy of both datasets for different methods compared to find an optimal technique for fault detection in TEP. The best accuracies for proposed methods are as 93.5% and 89.5% by using Linear Discriminant Analysis and Subspace Discriminant that is the method of Ensemble Learning.

The rest of the parts of this thesis are organized as follows: In Chapter 2, background on the dataset, Poincare Measures, one-way ANOVA, and classification methods. Chapter 3 presents the results and discussion of proposed methods. The conclusions are given in Chapter 4.

Chapter 2

Methods

This section provides background information on thesis research. General information about the main components of the TEP and the dataset is given. The feature extraction method, which is based on Poincare plot measurements, is explained. The one-way ANOVA test method, as a statistical-based feature selection method, is described. Then, the classification methods such as Decision Trees, Discriminant Analysis, Naive Bayes Classification, SVM, k-NN, and Ensemble Learning are explained. In addition, within the scope of this research, the parts of the thesis related to the application are explained.

2.1 Tennessee Eastman Process

TEP is a process which model was developed by Vogel and Downs with the Eastman Chemical Company [40]. The model has become a benchmark issue for academia and industry's testing, control, and optimization strategies [34]. Several simulators have been established for this issue either in MATLAB [41] or in other coding languages [42]. The practicability of this simulator stems from the fact that it represents an actual functioning plant that involves different typical engineering units such as several controlled and manipulated variables and also disturbances [40]. The process simulated within the TEP consists of 5 units including condensers, reactors, separators, strippers, and compressors. In addition, the simulation contains several chemical reactions that turn into 8 different components in the simulation: 4 reactants (A, C, D, and E), 2 products (G and H), a byproduct (F), and an inert (B). Figure 2.1 shows the process flow diagram of the TEP that represents the plant operation in detail. Moreover, it should be pointed out that the process simulation in an open loop is unstable due to the presence of reactions in the reactor. So, there have been several control schemes implemented to equalize the process. The most popular Piping & Instrumentation Diagrams (P&ID) of TEP was developed by Ricker [43]. P&ID [44, 45] represents the piping and related components such as; PI: Pressure Indicator, PC: Pressure Controller, FI: Flow Indicator, and FC: Flow Controller,

<

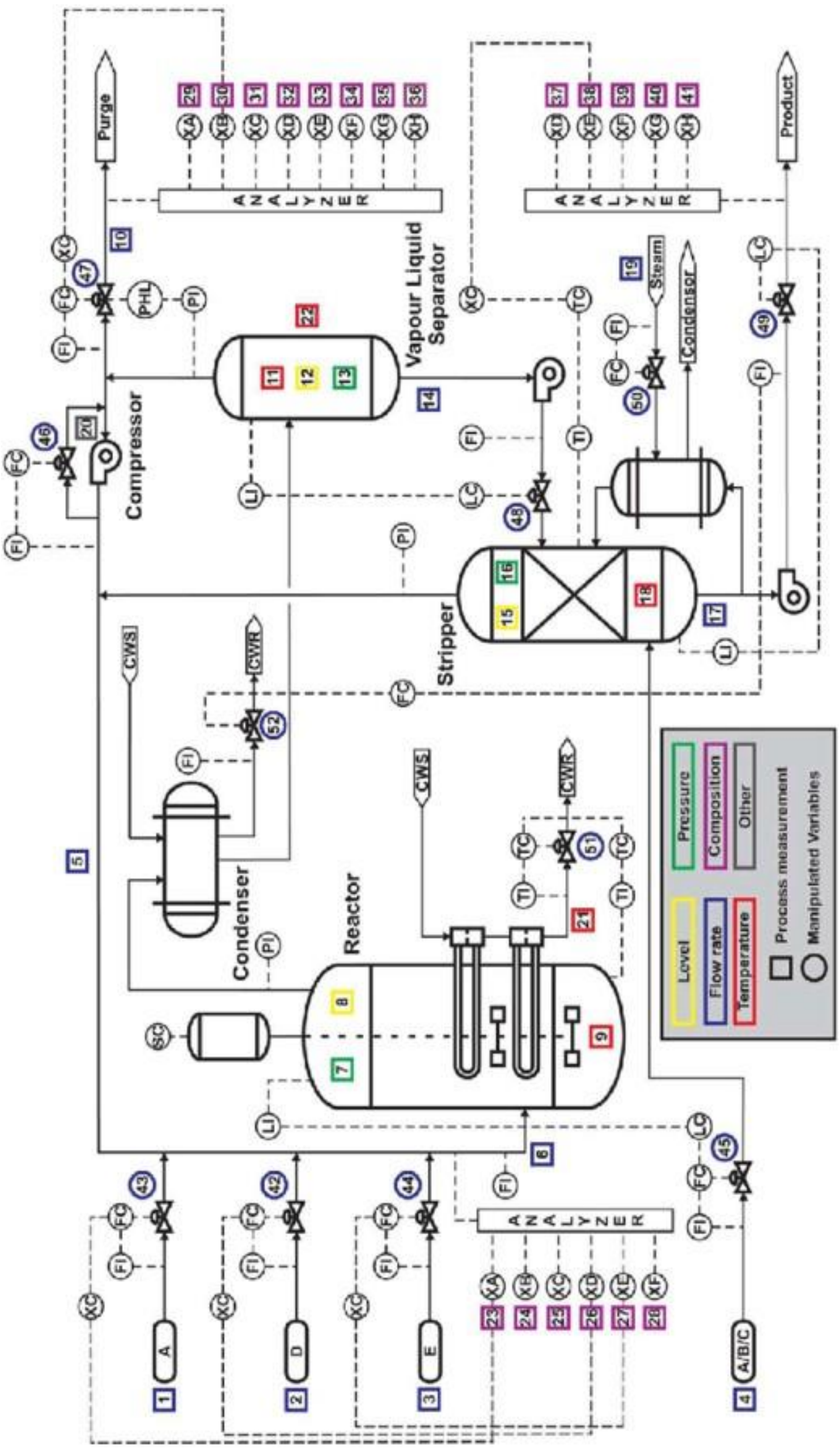

Figure 2.1: P&ID of TEP [43].

TEP process contains 53 total measurements composed of 22 continuous process variables (Table 2.1), 12 manipulated variables (Table 2.3), and 19 sampled analyzer process variables (Table 2.2).

Table 2.1: Continuous process measurements in TEP

No	Label	Description	Unit
1	xmeas_1	A Feed (Steam 1)	kscmh
2	xmeas_2	D Feed (Steam 2)	kg/hr
3	xmeas_3	E Feed (Steam 3)	kg/hr
4	xmeas_4	A and C Feed (Steam 4)	kscmh
5	xmeas_5	Recycle Flow (Steam 8)	kscmh
6	xmeas_6	Reactor Feedrate (Steam 6)	kscmh
7	xmeas_7	Reactor Pressure	kPa gauge
8	xmeas_8	Reactor Level	%
9	xmeas_9	Reactor Temperature	Deg C
10	xmeas_10	Purge Rate (Steam 9)	kscmh
11	xmeas_11	Product Separator Temperature	Deg C
12	xmeas_12	Product Separator Level	&
13	xmeas_13	Product Separator Pressure	kPa gauge
14	xmeas_14	Product Separator Underflow (Steam 10)	m^3/hr
15	xmeas_15	Stripper Level	%
16	xmeas_16	Stripper Pressure	kPa gauge
17	xmeas_17	Stripper Underflow (Steam 11)	m^3/hr
18	xmeas_18	Stripper Temperature	Deg C
19	xmeas_19	Stripper Stream Flow	kg/hr
20	xmeas_20	Compressor Work	kW
21	xmeas_21	Reactor Cooling Water Outlet Temperature	Deg C
22	xmeas_22	Separator Cooling Water Outlet Temperature	Deg C

Table 2.2: Sample process measurements in TEP

No	Label	Description
1	xmeas_23	Reactor Feed Analysis of A (Stream 6)
2	xmeas_24	Reactor Feed Analysis of B (Stream 6)
3	xmeas_25	Reactor Feed Analysis of C (Stream 6)
4	xmeas_26	Reactor Feed Analysis of D (Stream 6)
5	xmeas_27	Reactor Feed Analysis of E (Stream 6)
6	xmeas_28	Reactor Feed Analysis of F (Stream 6)
7	xmeas_29	Purge Gas Analysis of A (Stream 9)
8	xmeas_30	Purge Gas Analysis of B (Stream 9)
9	xmeas_31	Purge Gas Analysis of C (Stream 9)
10	xmeas_32	Purge Gas Analysis of D (Stream 9)
11	xmeas_33	Purge Gas Analysis of E (Stream 9)
12	xmeas_34	Purge Gas Analysis of F (Stream 9)
13	xmeas_35	Purge Gas Analysis of G (Stream 9)
14	xmeas_36	Purge Gas Analysis of H (Stream 9)
15	xmeas_37	Product Analysis of D (Stream 11)
16	xmeas_38	Product Analysis of E (Stream 11)
17	xmeas_39	Product Analysis of F (Stream 11)
18	xmeas_40	Product Analysis of G (Stream 11)
19	xmeas_41	Product Analysis of H (Stream 11)

Table 2.3: Manipulated variables in TEP

No	Label	Description
1	xmv_1	D Feed Flow (Steam 2)
2	xmv_2	E Feed Flow (Steam 3)
3	xmv_3	A Feed Flow (Steam 1)
4	xmv_4	A and C Feed Flow (Steam 4)
5	xmv_5	Compressor Recycle Valve
6	xmv_6	Purge Valve (Steam 9)
7	xmv_7	Separator Pot Liquid Flow (Steam 10)
8	xmv_8	Stripper Liquid Product Flow (Steam 11)
9	xmv_9	Stripper Steam Valve
10	xmv_10	Reactor Cooling Water Flow
11	xmv_11	Condenser Cooling Water Flow
12	xmv_12	Agitator Speed

Variables in the dataset are named "*Faulty_Free_Training*", "*Faulty_Free_Testing*", "*Faulty_Testing*" and "*Faulty_Training*" corresponding to different data files. Each data file contains 55 columns of data. The "*faultNumber*" column defines as 1 to 20 types of faults in the dataset while the "0" represents no failure. The "*Sample*" column gives the sequence number of the measurement taken in the training data set between 1 and 500 (between 1 and 960 in the test data set). Starting with "*xmeas_1*" and ending with "*xmv_11*" (4 to 55) gives the values observed from the sensors, which correspond to 25 hours of sensor data in the training dataset and 48 hours in the test dataset, which are sampled every 3 minutes. Thus, the dataset is so widely used to compare algorithms for detecting abnormal situations containing erroneous and error-free data files.

The process contains 20 previously programmed faulty scenarios, which are described in Table 2.4. In 2015, Ricker release a revision to his original control P&ID for the

TEP that brought updates in algorithms, process measurements, and disturbances [43]. The labels considered in Tables 2.1 - 2.4 are the labels used in the TEP.

Table 2.4: Process disturbances in TEP

No & Fault ID	Label	Type	Description
1	idv_1	Step	A/C Feed Ratio, B Composition Constant
2	idv_2	Step	B Composition, A/C Ratio Constant
3	idv_3	Step	D Feed Temperature
4	idv_4	Step	Reactor Cooling Water Inlet Temperature
5	idv_5	Step	Condenser Cooling Water Inlet Temperature
6	idv_6	Step	A Feed Loss
7	idv_7	Step	C Header Pressure Loss-Reduced Availability
8	idv_8	Random	A, B, C Feed Composition
9	idv_9	Random	D Feed Temperature
10	idv_10	Random	C Feed Temperature
11	idv_11	Random	Reactor Cooling Water Inlet Temperature
12	idv_12	Random	Condenser Cooling Water Inlet Temperature
13	idv_13	Drift	Reactor Kinetics
14	idv_14	Stiction	Reactor Cooling Water Valve
15	idv_15	Stiction	Condenser Cooling Water Valve
16	idv_16	Random	Deviations of Heat Transfer Within Stripper
17	idv_17	Random	Deviations of Heat Transfer Within Reactor
18	idv_18	Random	Deviations of Heat Transfer Within Condenser
19	idv_19	Stiction	Recycle Valve of Compressor, Underflow Separator, Underflow Stripper and Steam Valve Stripper
20	idv_20	Random	Unknown

There are different measurement points, system variables, and errors which are inevitable and detectable challenges for users and designers, so TEP has been prevailingly used as an information for process monitors and fault detections [46-48].

2.2 Poincare Plot Measures

The Poincare plot measures, a method taken from nonlinear dynamics of the data. It is a graph of every single data on the x-axis against the subsequent data on the y-axis (Figure 2.2). The shape of the distribution is utilized to characterize the dynamics of the time series [49]. The plot provides basic information as well as specific information [50-51]. The Poincare plot is becoming a preferred method because of its proved ability as a cardiac dysfunction and drawing an ellipse to the Poincare plot is becoming popular method [49,51]. The standard deviation of the distance of the data determines the width SD_1 and length SD_2 [51]. These variables can be calculated as follows:

$$SD_1 = \sqrt{\frac{1}{2}SDSD^2} = std(\frac{x_{i+1}-x_i}{\sqrt{2}}) \tag{2.1}$$

$$SD_2 = \sqrt{2(SD)^2 - \frac{1}{2}SDSD^2} = std(\frac{x_{i+1}+x_i}{\sqrt{2}}) \tag{2.2}$$

where $SDSD$ and SD are standard deviation of consecutive differences and the standard deviation of the data where l=1 respectively. The product ($SD_1 . SD_2$) and the ratio (SD_1 / SD_2) can be determined to specify the relationships between these components. (SD_1 / SD_2) is assumed to be a signal of the balance between the vagal activities and sympathetic [50]. In most of the study, the conventional value of lag (l) is 1 53, 54], but a few studies used distinct values from 1 to 10 [55, 56]. The results for lagged plots are transferred to higher dimensional plots [57].

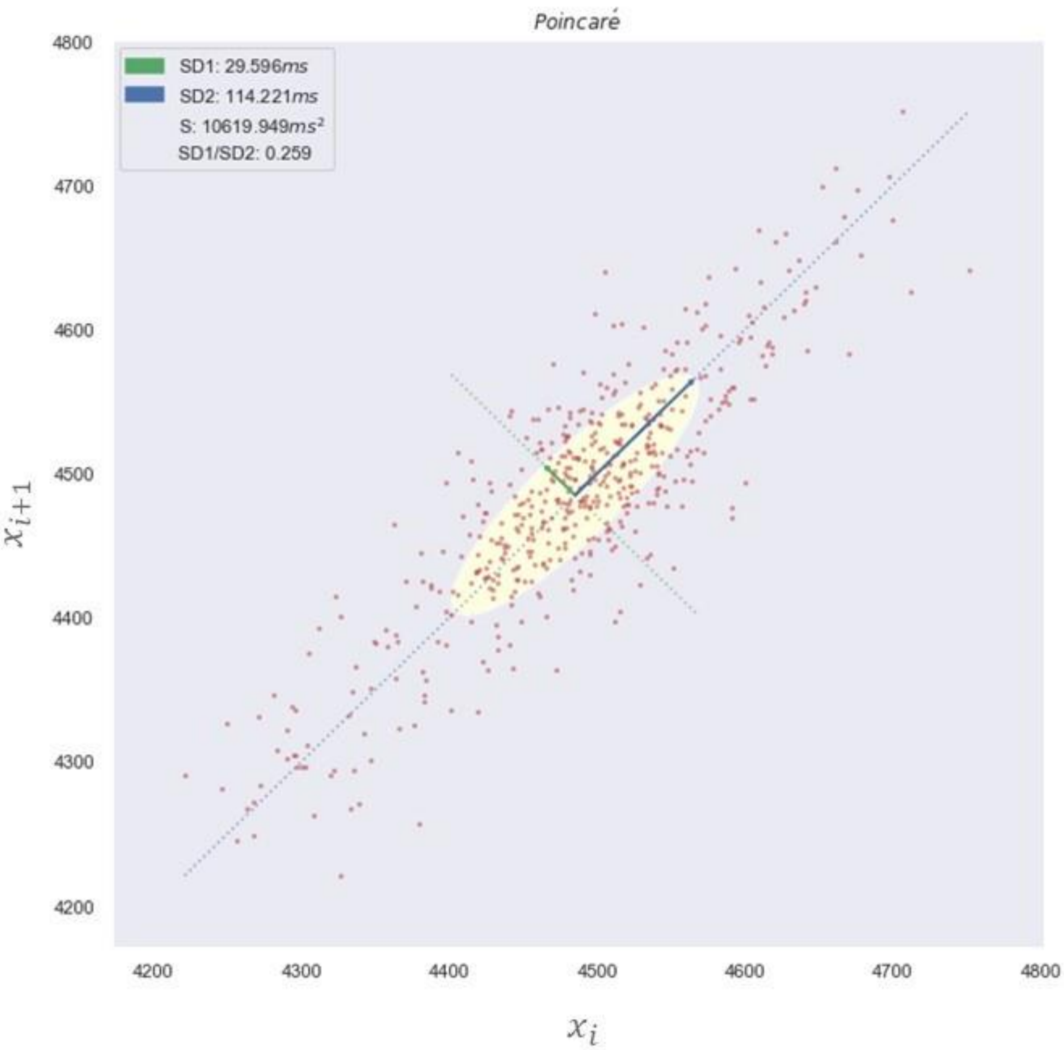

Figure 2.2: Poincare plot and the measures of SD_1 and SD_2 for *simulationRun*=1, *faultNumber*=1 of *xmeas_3* variable in the dataset

2.3 One-way ANOVA Test

One-way ANOVA is a statistical method that is concerned with comparing the means of several data and it is used to determine whether there is a statistically significant difference between the means of independent groups and it is determined by investigating the variances. It can be thought of as an appendage of the t-test for two independent data to more than two groups [38, 39]. While performing an ANOVA, the subsequent assumptions are required: The investigations are independent of one another. The investigation of groups might have satisfied as a normal distribution. The population variances in every groups are the same.

ANOVA is the typically used advanced research method in the economic literature and business [58]. This method is useful in revealing information in interpreting experimental results and specifying the influence of some factors on other parameters [59]. The original ideas of ANOVA were improved by the Sir Ronald A. Fisher [60] in his book which is namely “Statistical Methods for Research Workers”. Much of the early studies in this area are related to agricultural tests [59].

Comparison tests are divided into two groups according to equal or different variance approaches. These two groups are shown in the Table 2.5 below.

Table 2.5: Groups according to the variance method

No	Equal Variance	Different Variance
1	Fishcer’s Significant Difference Test	Tamhane Test
2	Bonferroni Test	Dunnet T3 Test
3	Tukey HSD Test	Games-Howell Test
4	Scheffe Test	Dunnet-C Test
5	Duncan Test	-
6	Dunnet Test	-
7	Waller-Duncan Test	-

2.4 Classification

Classification is the process of estimating the class of data points that are given. Classes are occasionally called labels, targets, or categories. The aim of classification is approximating a function of mapping (f) from input (X) to discrete output (Y). For instance, classifying given a handwritten character can be identified as a classification issue. This is multi-class classification because there are more than two classes as all letters are in the alphabet. A classifier utilizes some training data to figure out how given data relates to the class. In this case, all letters have to be used as the training data. When the classifier is trained successfully, it can be used to estimate a handwritten character. The classification belongs to the category of supervised learning.

There are plenty of applications in classification like diagnosis, target marketing [59, 60], and also plenty of classification software existing now however it is not feasible to finalize which one is superior to the other. It depends on the available data set and the application. For instance, if the categories are linearly separable, the linear classifiers such as Logistic Regression, Fisher's Linear Discriminant can outperform complicated models.

Classification methods such as Decision Tree, Discriminant Analysis, Naive Bayes Classification, SVM, k-NN, Ensemble Learning techniques are explained.

2.4.1 Decision Tree

Decision tree analysis is positioned as a predictive model tool used in many fields [112, 113]. Decision trees are created using an algorithmic method for decomposing data sets according to different situations. Decision trees can be found in forms suitable for multivariate or multi-effect analyses. All forms of multivariate analysis allow to predict, describe, explain or classify as targets. These multivariate analyzes have an important place in today's technique because almost all specific outcomes that determine success are based on different factors. Figure 2.3 shows an example of a decision tree with connected nodes.

Nodes can be categorized as decision nodes and leaf nodes. If one of the decision nodes takes place as the initial state, it is called the root node. Decision nodes work to direct the path flows that create the rules. Decision nodes provide a branch to subsequent nodes, while leaf nodes indicate the end of a consequential path. Relationships in decision trees are non-linear and often interactive. It is not possible in regression analyses to reveal these patterns in the data set without prior modeling, but classification trees reveal these patterns without the need for predetermination.

Decision trees are machine learning structures that stand out with their flowchart-like structures, enable the collection of results to terminal units, that is leaves, by performing a test on each branch, and each branch reflects a classification model from roots to leaves.

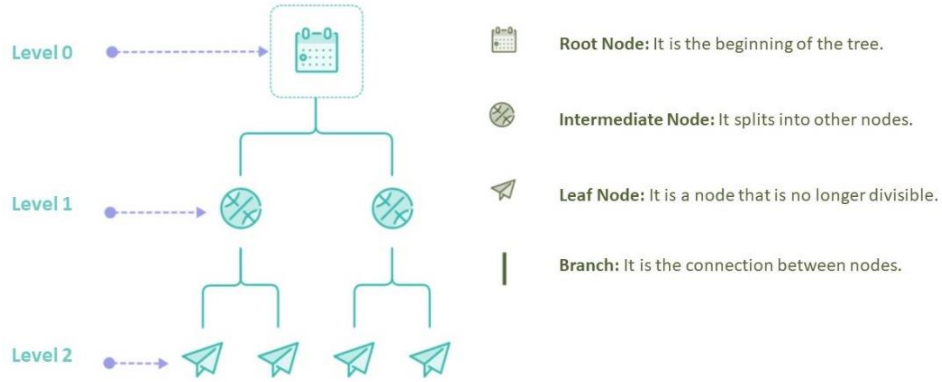

Figure 2.3: Decision tree diagram and its components.

One of the first models of analysis studies based on decision trees was found by J. Ross Quinlan [63] in his book. The first algorithm he developed was recorded as Iterative Dichotomiser 3 (ID3). This algorithm is based on the creation of the smallest and most efficient decision tree based on Occam's razor principle. Quinlan continued his studies and developed this model, firstly it evolved into C4.5, C5.0 algorithms, respectively. To briefly mention other important decision trees; Classification and Regression Trees (CART) [64], Chi-square Automatic Interaction Detector (CHAID) [65], Multivariate Adaptive Regression Splines (MARS) [66], Random Forest [67] are important decision trees in the literature. In decision analysis, decision trees are created depending on the use of data, usually with a supervised learning method. The focus is on ensuring that the trees created represent both the dataset used and the data to come later. In this thesis, different types of decision tree classifier types were used and difference of the types are shown in Table 2.6.

Table 2.6: Decision Tree classifier types and maximum number of splits

No	Classifier Type	Max. Number of Splits
1	Fine Tree	4
2	Medium Tree	20
3	Coarse Tree	100

2.4.2 Discriminant Analysis

Discriminant analysis is a classifier technique that creates functions that will allow the variables in the data set to be divided into two or more real groups, taking into account the p-items of the units or observations, and ensuring that these units are optimally assigned to their real groups. For example, suppose there are three diseases that have similar characteristics, such as A, B, and C. Each disease will have a value according to variables. It is possible to create functions that determine its properties. With the help of these functions, it can be determined which group feature a new disease observation vector has, that is, which disease diagnosis may have, and its assignment to the right group can be done by discriminant analysis [68].

By minimizing the possibility of misclassification, the researcher will want to divide the observations into the groups they belong to or identify the groups from which these observations were drawn [69]. Here, it is aimed to maximize the difference between the means of the groups to be determined. The basic assumptions of the discriminant analysis method are that the variables are normally distributed and that the groups have a common variance-covariance matrix. The discriminant function that satisfies these assumptions is called the Linear Discriminant Function (LDF). If these assumptions are not met, alternative functions can be used. If the data are normally distributed but the variance-covariance matrices of the groups are different, the function used is defined as the Quadratic Discriminant Function (QDF).

The functions of discriminant analysis can be grouped under two main headings. The first of these is the assignment of any observation (variable) whose audience is unknown to the appropriate audience (group). The second function is to provide functions that can be used in the future, due to this function, discriminant analysis differs from cluster analysis and approaches multivariate regression analysis. In discriminant analysis, the number of clusters (groups) is known, this number does not change during the analysis and the researcher is asked to classify the observations into these clusters. In this thesis, different types of discriminant analysis classifier types were used and difference of the types are shown in Table 2.7.

Table 2.7: Discriminant Analysis classifier types and maximum number of splits

No	Classifier Type	Boundaries Type
1	Linear Discriminant	Linear
2	Quadratic Discriminant	Nonlinear

2.4.3 Naive Bayes Classification

Naive Bayes Classification is an algorithm that predicts the probability that the available data belong to the determined classes by applying Bayes theorem under strong independence assumption [70]. In this method, classification is done based on Bayesian probability. Bayesian probability is a generalized version of the conditional probability for k discrete events. The Naive Bayes algorithm allows to predict new and unlabeled observations with the same manner by making use of the feature information of observations based on the conditional probability function [71].

The Naive Bayes formula is as in (2.3). $P(c)$ represents the previous probability value of the target, $P(x|c)$ the probability value of the parameter relative to the target, $P(c|x)$ the next probability value of the target relative to the parameter, $P(x)$ represents the previous probability value of the parameter. The $P(x)$ value is calculated using the data in the training dataset (Eq. 2.3). In this thesis, two types of Naive Bayes classifier types were used and difference of the types are listed in Table 2.8.

$$P(c|x) = \frac{P(X|C)P(c)}{P(x)} \tag{2.3}$$

Table 2.8: Naive Bayes classifier types and parameters for model flexibility

No	Classifier Type	Parameters for Model Flexibility
1	Gaussian Naive Bayes	Cannot change parameters
2	Kernel Naive Bayes	Kernel Type and Support Settings

2.4.4 Support Vector Machine

SVM is a popular topic in the learning area [72, 73]. In the late 1990s, the neural network approaches suffered issues with producing models, generalization. It was developed by Vladimir Vapnik [74] and gained popularity due to its features. It is first been introduced as a technique for solving classification issues. However, due to its attractive properties, it has expanded into the field of regression analysis.

SVM is a supervised learning method that can use given sample to solve certain issues by attempting to turn them into linearly separable problems [75, 76]. The SVM is input data called training data sets linked to outputs to classify new observations to one of the two classes by creating a separating hyperplane [76]. This study uses six dissimilar kernels: Linear, Quadratic, Cubic, Fine Gaussian, Medium Gaussian, and Coarse Gaussian. In this thesis, different types of SVM classifier types were used and difference of the types are shown in Table 2.9.

Table 2.9: SVM classifier types and maximum number of neighbors

No	Classifier Type	Max. Number of Neighbors
1	Linear SVM	1
2	Quadratic SVM	10

3	Cubic SVM	100
4	Fine Gaussian SVM	10
5	Medium Gaussian SVM	10
6	Coarse Gaussian SVM	10

2.4.5 k-Nearest Neighbors

k-NN is one of the ML algorithms. k-NN is used in pattern recognition and statistical estimation for regression [77] and classification [78-81]. For both regression and classification, the input consists of the nearest training examples in given data space and the output depends on regression or classification. There are several measures for calculation of distance like Chebyshev, Euclidean Squared, and Euclidean. Among all these Euclidean is the most popular method to measure the distance between two points. It is calculated as:

$$d\,(x, y) = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2} \tag{2.4}$$

where $d\,(x, y)$ is the distance of Euclidean between the unknown point, x, and the training point, y. Other metrics, such as the distance of Manhattan, the coefficient of cosine, or the distance of Lagrange have been used [82].

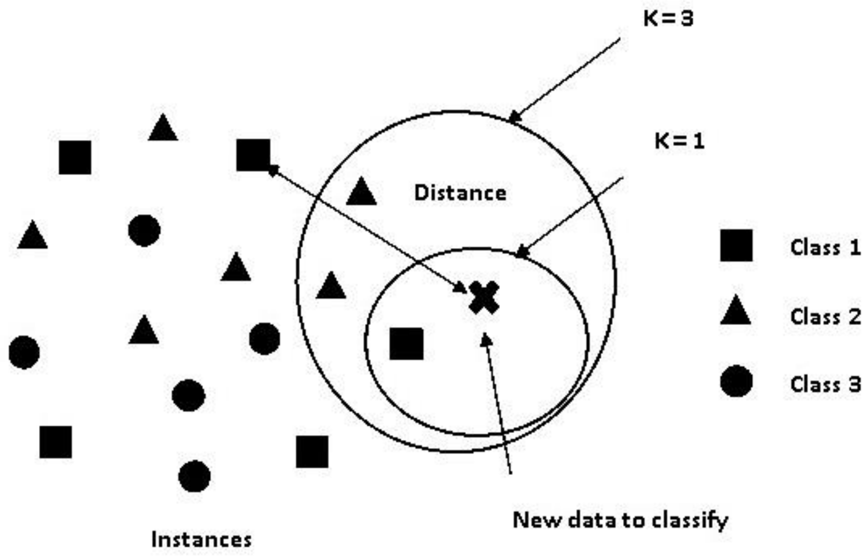

Figure 2.4: An example k-NN

There are plenty of approaches for obtaining the appropriate k [83, 84]. The most used is to test different values of k by using cross-validation [82] and keep the k giving the lowest classification error rate. In this thesis, different types of decision tree classifier types were used and difference of the types are listed in Table 2.10. In cosine, cubic and weighted k-NN classifier types are applied by using cosine, cubic and weighted distance metric respectively.

Table 2.10: k-NN classifier types and maximum number of neighbors

No	Classifier Type	Max. Number of Neighbors
1	Fine k-NN	1
2	Medium k-NN	10
3	Coarse k-NN	100
4	Cosine k-NN	10
5	Cubic k-NN	10
6	Weighted k-NN	10

2.4.6 Ensemble Learning

Ensemble learning creates multiple classifier models, unlike the use of a single classifier model that occurs with classical machine learning algorithms. The evaluation process is based on the logic of interpreting and presenting the results from all classifier models [85,86]. Bagging technique is a popular ensemble learning approach applied in various real problem scenarios such as intrusion detection, spam classification, credit scoring, etc. [87]. With this technique, the dataset is divided into parts and each part is modeled as a separate training set with base classifiers. The test takes place on all models. The classification result is obtained by analyzing the classification results collected from the models.

If the classification is performed on a numerical value as a result of classification, Eq. 2.5 is used and the classification result is produced by taking the average of the numerical output values from the multiple classification models. If it occurs on categorical values, it is carried out according to Eq. 2.6 and the most classified category among the categorical values from each classifier model is accepted as the result [85].

n ∈ [1, N] and M_{n}: The dataset created with a random piece from the main dataset.

$$M(x) = \frac{1}{N} + \sum_{n=1}^{N} (M_{\mathrm{n}}(X)) \qquad 2.5$$

$$M(x) = argmax_{\mathrm{p}}(M_{\mathrm{n}}(X) = p) \qquad 2.6$$

The working principle of the bagging technique is shown in Figure 2.5. The main dataset is randomly divided by the specified number of parameters. Classifier models are trained on each part of the dataset. The classification result collected from all models is interpreted according to Formula 2.5 or Formula 2.6.

The Boosting technique is an ensemble learning technique approach developed to develop the performance of learning algorithm. As shown in Figure 2.5, each iteration is weighted on the dataset to decrease the error of the weak learning model. In this way, it is aimed to strengthen weak models. Reinforcement works by continually running a particular weak learning method on distributions over the training data and then combining the weak generated classifiers into a single composite classifier. Ensemble learning classifier types that were used in this thesis are listed in Table 2.11.

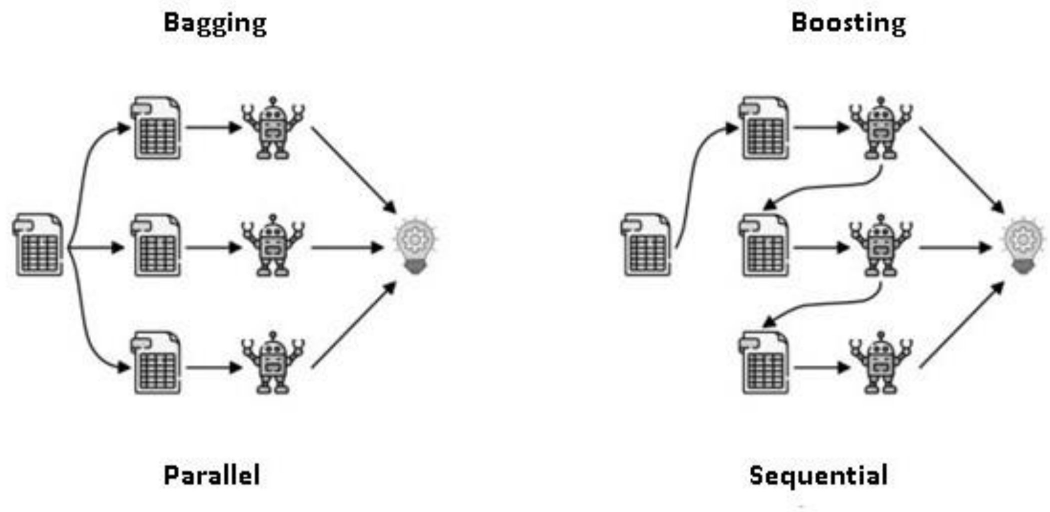

Figure 2.5. Bagging and boosting type of ensemble learning method [40]

Table 2.11: Ensemble Learning classifier types

No	Classification Type	Ensemble Method
1	Boosted Trees	AdaBoost with Decision Tree Learners
2	Bagged Trees	Random Forest Bag with Decision Tree Learners
3	Subspace Discriminant	Subspace with Discriminant Learners
4	Subspace k-NN	Subspace with Nearest Neighbor Learners
5	RUSBoosted Trees	RUSBoost with Decision Tree Learners

2.5 Implementation

As for the preprocessing, SD_1 and SD_2 values were extracted from the raw dataset by using Poincare plot measurements. After that, one-way ANOVA test was used to find out which of the Poincare measures in this dataset differed between error types at the 5% significance level. As a result of this test, the features that should be removed from the data set were determined and a preprocessed dataset was obtained besides the raw dataset. After extraction and selection part of the study, classifier algorithms were run to classify 20 different faults for both dataset by using three different Decision Tree algorithms (Fine Tree, Medium Tree, and Coarse Tree), two different Discriminant algorithms (Linear Discriminant and Quadratic Discriminant), two different Naive Bayes algorithms (Gaussian Naive Bayes and Kernel Naive Bayes), six different SVM algorithms (Linear SVM, Quadratic SVM, Cubic SVM, Fine Gauss SVM, Medium Gauss SVM, and Coarse Gauss SVM), six different k-NN algorithms (Fine k-NN, Medium k-NN), Coarse k-NN, Cosines k-NN, Cubic k-NN, and Weighted k-NN) and five different Ensemble Learning algorithms (Boosted Trees, Bagged Trees, Subspace Discriminant, Subspace k-NN, and RUSBboosted Trees).

The performances of all classifiers were compared by calculating the ratio of the number of correctly classified errors to the total number of errors. Figure 2.6 summarizes the operation of the proposed fault detection system.

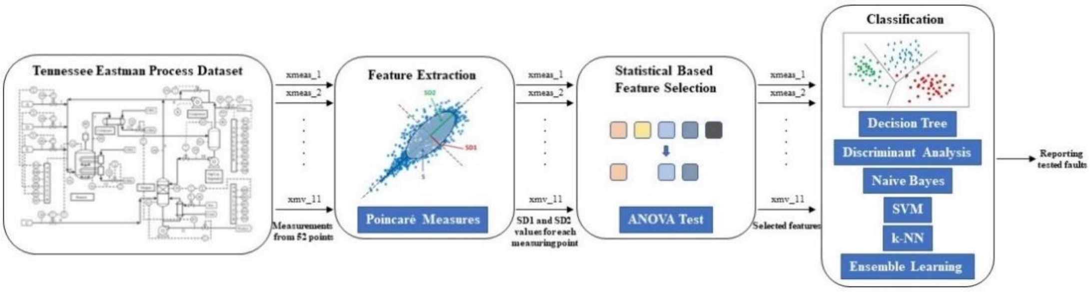

Figure 2.6: Proposed diagnostic system diagram

Chapter 3

Results and Discussion

In this thesis, Poincaré plots were applied to the dataset for feature extraction so that four commonly used nonlinear features (i.e., SD_1 and SD_2 values) for every measurement point were calculated. After that, among these features, the features that show a statistically significant difference of 5% between failure types were selected by using one-way ANOVA test. As a result of this test, since it was determined that the SD_1 measurements of the sensors 24, 26, 32, 37, 39, 40, and 41 did not show a statistically significant difference, these features were removed from the data set, and a new data set was created. After all the preprocessing processes, the classification stage was applied to classify 20 different faults for both datasets by using three different Decision Tree algorithms, two different Discriminant algorithms, two different Naive Bayes algorithms, six different SVM algorithms, six different k-NN algorithms, and five different Ensemble Learning algorithms. According to the results with and without feature selection, the decision tree algorithm gave the classifier accuracies listed below in Table 3.1.

Table 3.1: Decision Tree classifier results with and without feature selection

No	Classification Method	Without Feature Selection			With Feature Selection		
		Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Fine Tree	76.5%	~3000 obs/sec	3.7655 sec	74.5%	~17000 obs/sec	0.5718 sec
2	Medium Tree	76.5%	~12000 obs/sec	0.4074 sec	74.5%	~18000 obs/sec	0.1429 sec
3	Coarse Tree	25.0%	~16000 obs/sec	0.1721 sec	24.5%	~17000 obs/sec	0.1238 sec

Decision matrix of Decision Tree classifier for all classification method are shown in Figure 3.1 – 3.3. According to Table 3.1, Fine and Medium Decision Tree classification method is more successful than coarse tree for Decision Tree algorithm. When number of splits are increasing, the time for training is decreasing but the accuracy percentage is also decreasing. Although the feature selection applied in all cases reduces the training time, it also reduces the accuracy.

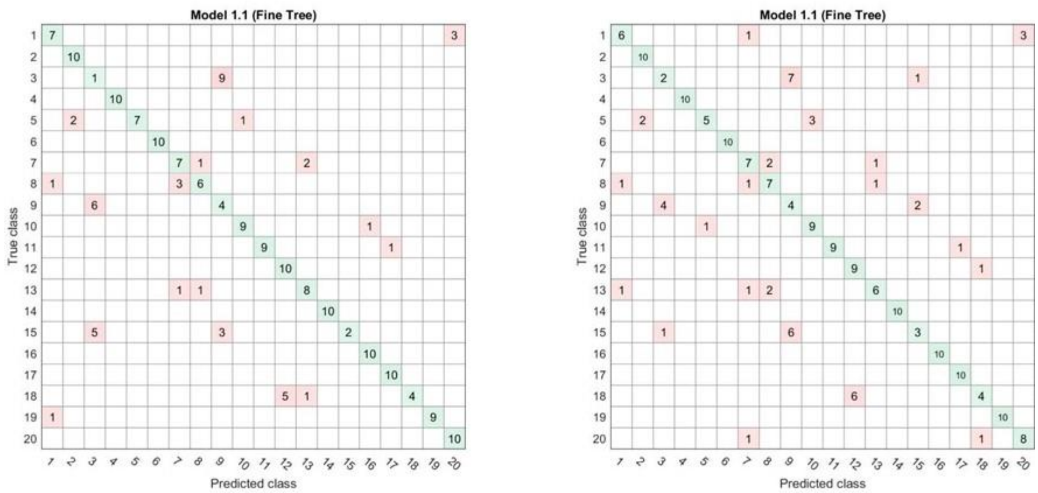

Figure 3.1: Decision matrix for Fine Tree classifier with and without feature selection

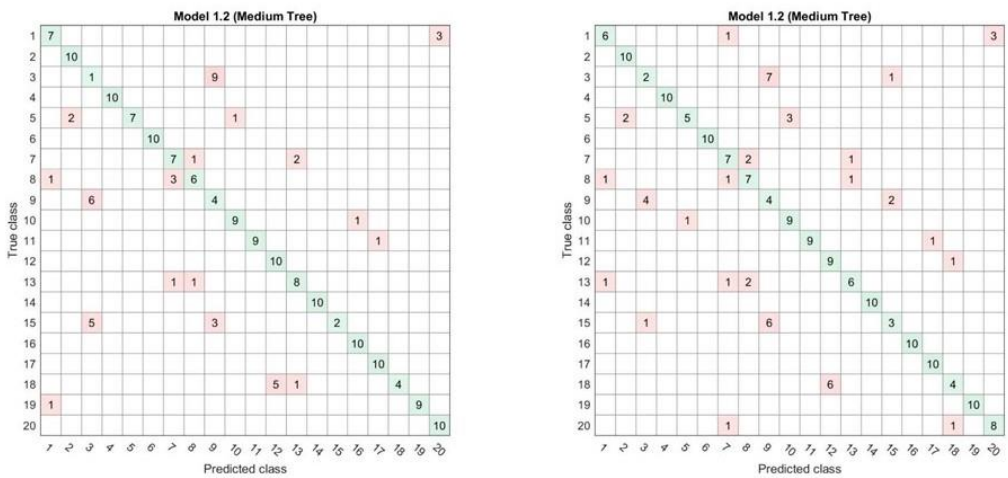

Figure 3.2: Decision matrix for Medium Tree classifier with and without feature selection

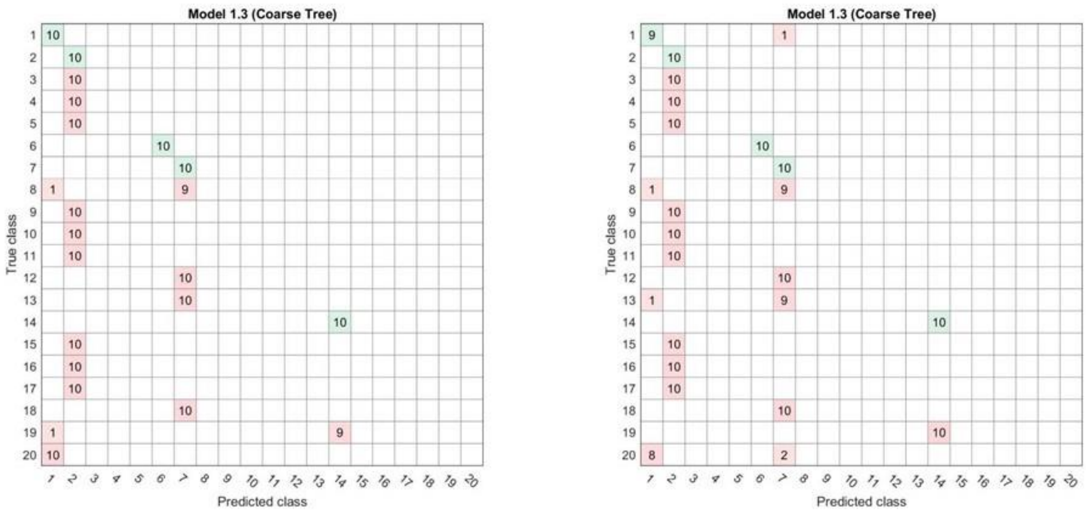

Figure 3.3: Decision matrix for Coarse Tree classifier with and without feature selection

According to the results with and without feature selection, the algorithm gave the Discriminant Analysis classifier accuracies listed below in Table 3.2.

Table 3.2: Discriminant Analysis results with and without feature selection

No	Classification Method	Without Feature Selection			With Feature Selection		
		Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Linear Discriminant	90.5%	~4400 obs/sec	0.7896 sec	93.5%	~11000 obs/sec	0.2055 sec
2	Quadratic Discriminant	Failed	Failed	Failed	Failed	Failed	Failed

Decision matrix of Discriminant Analysis classifier for Linear Discriminant are shown in Figure 3.4. The relevant decision matrix table for Quadratic Discriminant could not be extracted because the learning was unsuccessful. According to Table 3.2, Linear Discriminant classification method is more successful than Decision Tree method. The feature selection is useful for training time and accuracy.

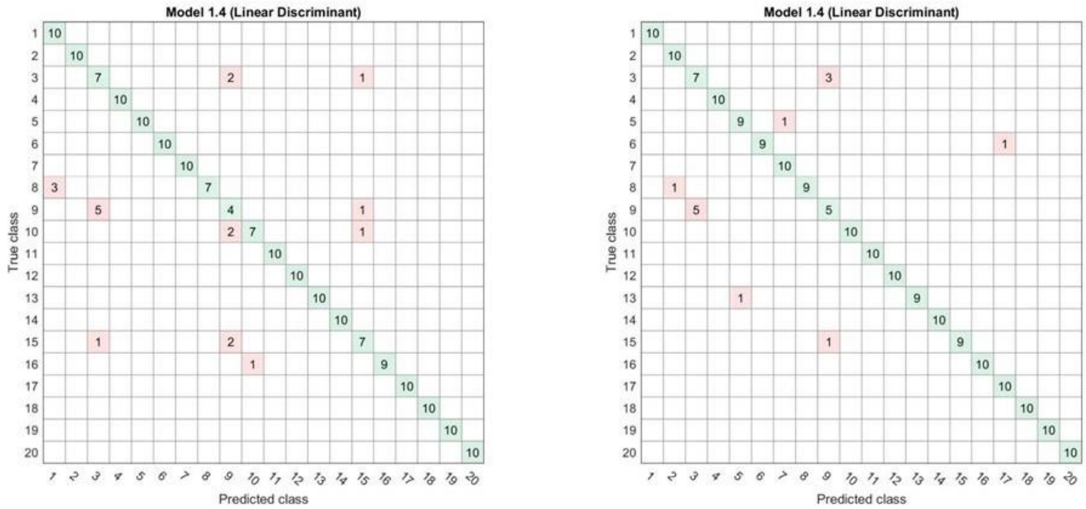

Figure 3.4: Decision matrix for Linear Discriminant classifier with and without feature selection

According to the results with and without feature selection, the algorithm gave the Naive Bayes classifier accuracies listed below in Table 3.3.

Table 3.3: Naive Bayes classifier results with and without feature selection

		Without Feature Selection			With Feature Selection		
No	Classification Method	Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Gaussian Naive Bayes	77.0%	~3400 obs/sec	1.1045 sec	72.0%	~5500 obs/sec	0.5653 sec
2	Kernel Naive Bayes	64.5%	~140 obs/sec	26.016 sec	68.0%	~130 obs/sec	25.695 sec

Decision matrix of Naive Bayes classifier for all classification method are shown in Figure 3.5 – 3.6. According to Gaussian Naive Bayes classification method in Table 3.3, the way that is without feature selection is more successful than the way that is with feature selection. Nevertheless, the way that is with feature selection is more successful than the way that is without feature selection for Kernel Naive Bayes classification method. But Linear Discriminant classification method is still the better way for this study.

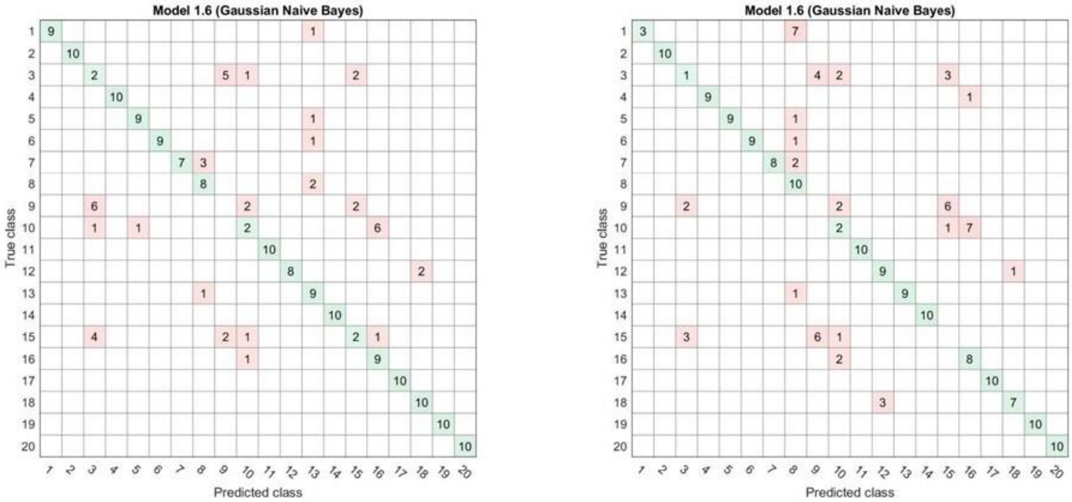

Figure 3.5: Decision matrix for Gaussian Naive Bayes classifier with and without feature selection

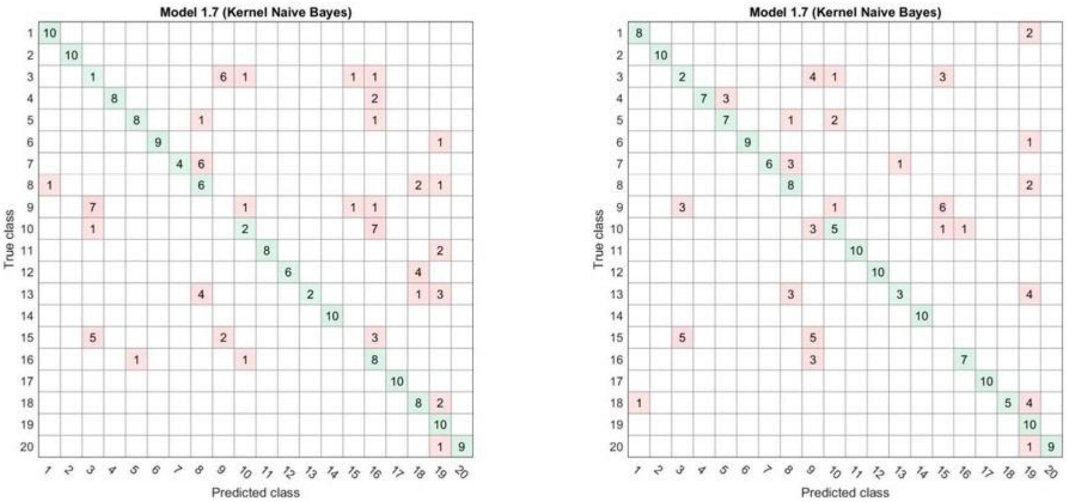

Figure 3.6: Decision matrix for Kernel Naive Bayes classifier with and without feature selection

According to the results with and without feature selection, the algorithm gave the SVM classifier accuracies listed below in Table 3.4.

Table 3.4: SVM classifier results with and without feature selection

No	Classification Method	Without Feature Selection			With Feature Selection		
		Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Linear SVM	51.0%	~590 obs/sec	6.4595 sec	54.5%	~370 obs/sec	6.608 sec
2	Quadratic SVM	56.0%	~750 obs/sec	5.0216 sec	54.5%	~710 obs/sec	6.3595 sec
3	Qubic SVM	49.0%	~840 obs/sec	4.8289 sec	46.0%	~420 obs/sec	7.3898 sec
4	Fine Gaussian SVM	6.0%	~790 obs/sec	4.127 sec	6.5%	~910 obs/sec	4.0549 sec
5	Medium Gaussian SVM	64.0%	~840 obs/sec	4.0879 sec	71.0%	~900 obs/sec	3.9831 sec
6	Coarse Gaussian SVM	45.0%	~860 obs/sec	4.1042 sec	51.0%	~920 obs/sec	3.9377 sec

Decision matrix of SVM classifier for all classification method are shown in Figure 3.7 – 3.12. According to Quadratic and Qubic SVM classification method in Table 3.4, the way that is without feature selection is more successful than the way that is with feature selection.

Nevertheless, the way that is with feature selection is more successful than the way that is without feature selection for Linear, Fine Gaussian, Medium Gaussian and Coarse Gaussian classification method. But Linear Discriminant classification method is still the better way for this study.

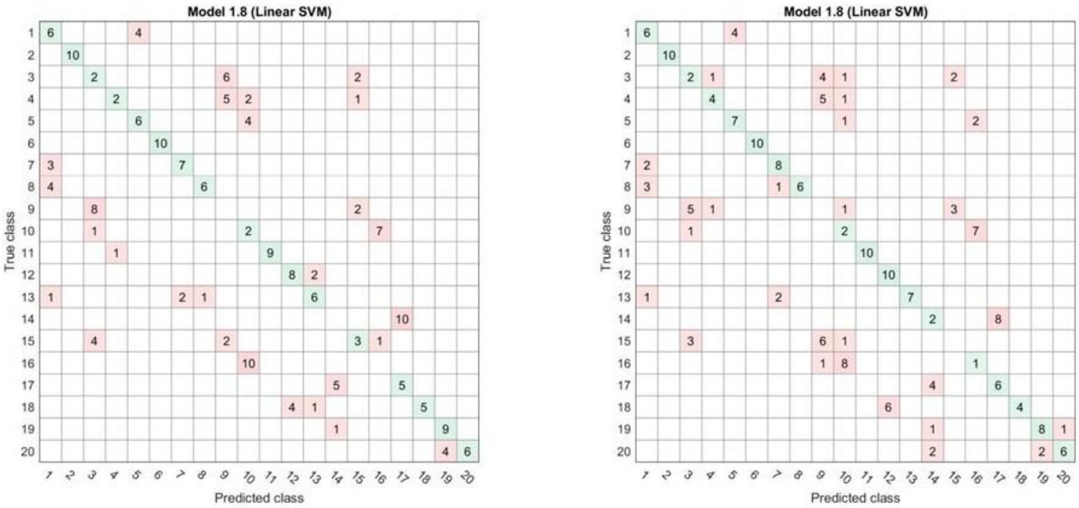

Figure 3.7: Decision matrix for Linear SVM classifier with and without feature selection

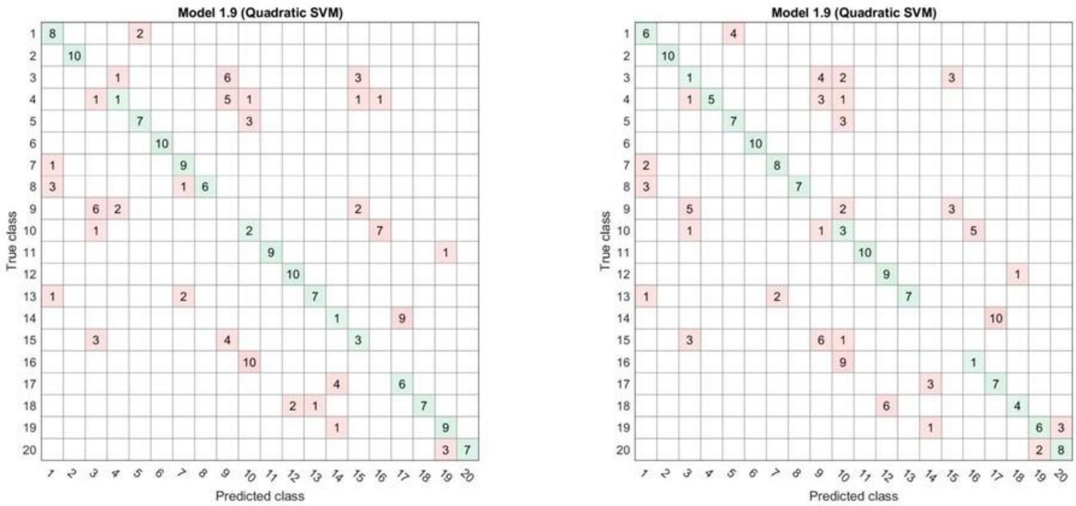

Figure 3.8: Decision matrix for Quadratic SVM classifier with and without feature selection

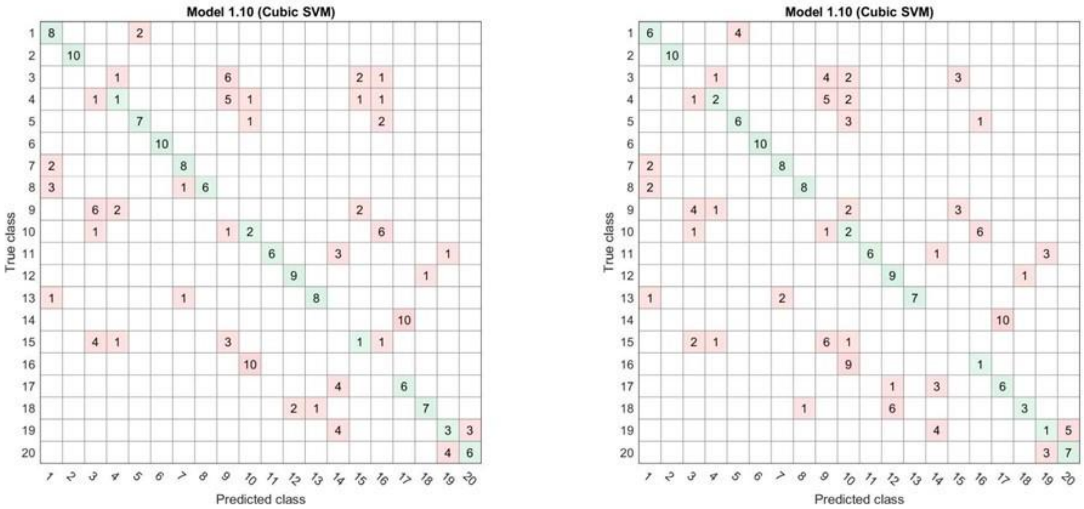

Figure 3.9: Decision matrix for Qubic SVM classifier with and without feature selection

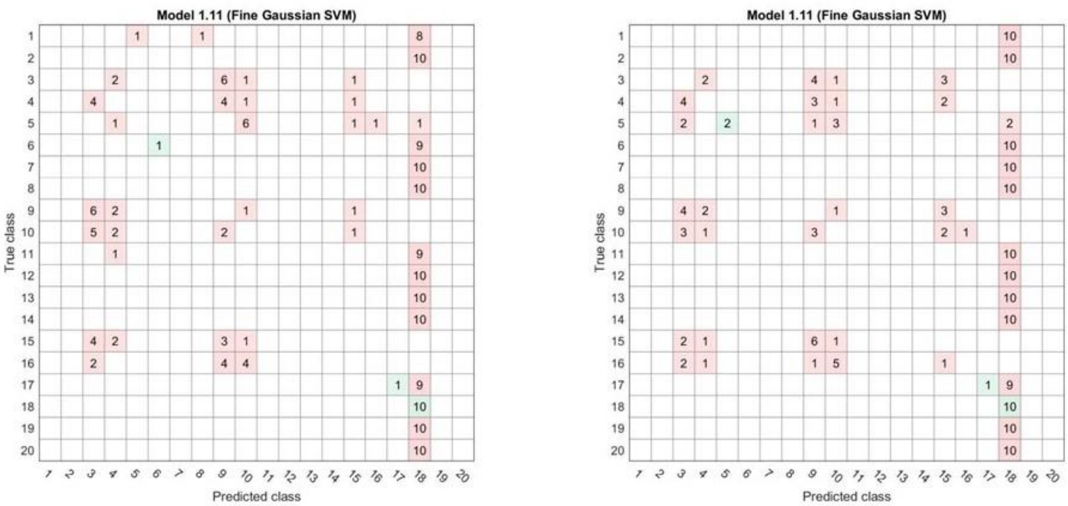

Figure 3.10: Decision matrix for Fine Gaussian SVM classifier with and without feature selection

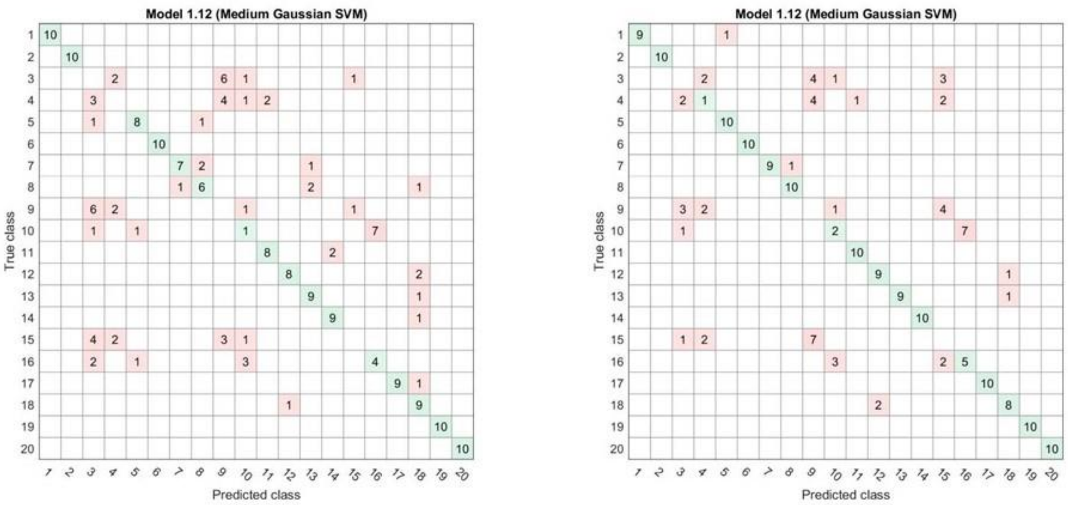

Figure 3.11: Decision matrix for Medium Gaussian SVM classifier with and without feature selection

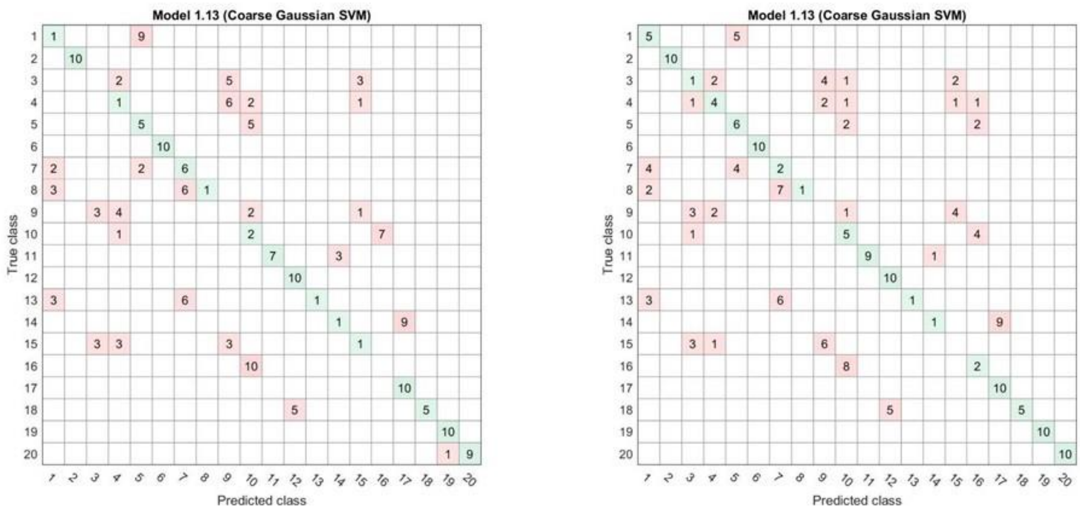

Figure 3.12: Decision matrix for Coarse Gaussian SVM classifier with and without feature selection

According to the results with and without feature selection, the algorithm gave the k-NN classifier accuracies listed below in Table 3.5.

Table 3.5: k-NN classifier results with and without feature selection

No	Classification Method	Without Feature Selection			With Feature Selection		
		Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Fine k-NN	53.5%	~4600 obs/sec	0.5097 sec	56.0%	~13000 obs/sec	0.1401 sec
2	Medium k-NN	59.0%	~12000 obs/sec	0.1852 sec	63.0%	~13000 obs/sec	0.1199 sec
3	Coarse k-NN	5.0%	~11000 obs/sec	0.1332 sec	5.0%	~13000 obs/sec	0.1231 sec
4	Cosine k-NN	59.0%	~9700 obs/sec	0.1409 sec	65.0%	~13000 obs/sec	0.1232 sec
5	Cubic k-NN	57.5%	~5000 obs/sec	0.2550 sec	56.5%	~5600 obs/sec	0.2134 sec
6	Weighted k-NN	55.0%	~11000 obs/sec	0.1231 sec	61.0%	~13000 obs/sec	0.1184 sec

Decision matrix of k-NN classifier for all classification method are shown in Figure 3.13 – 3.18. According to cubic k-NN classification method in Table 3.5, the way that is without feature selection is more successful than the way that is with feature selection. Nevertheless, the way that is with feature selection is more successful than the way that is without feature selection for the rest of the k-NN classification method. But linear discriminant classification method is still the better way for this study.

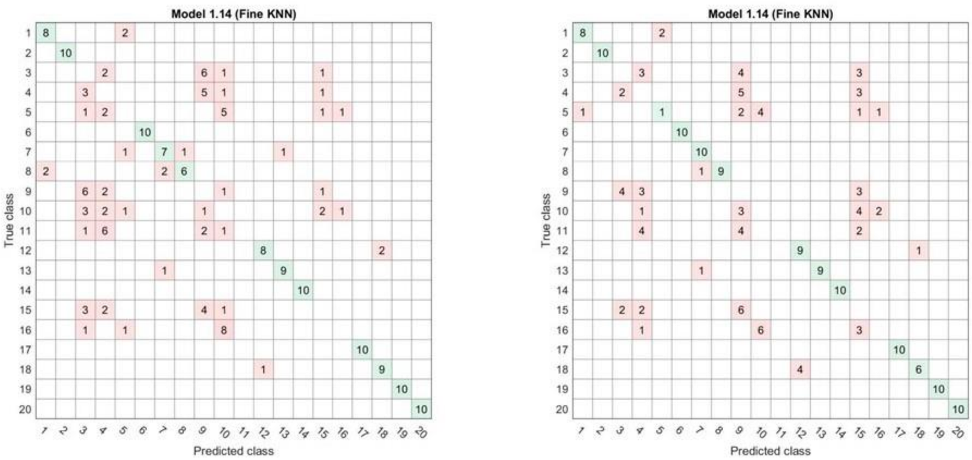

Figure 3.13: Decision matrix for Fine k-NN classifier with and without feature selection

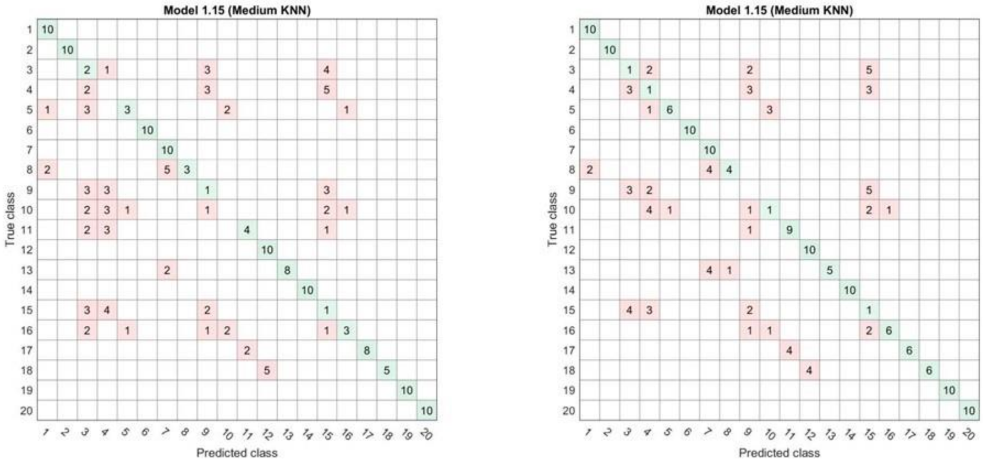

Figure 3.14: Decision matrix for Medium k-NN classifier with and without feature selection

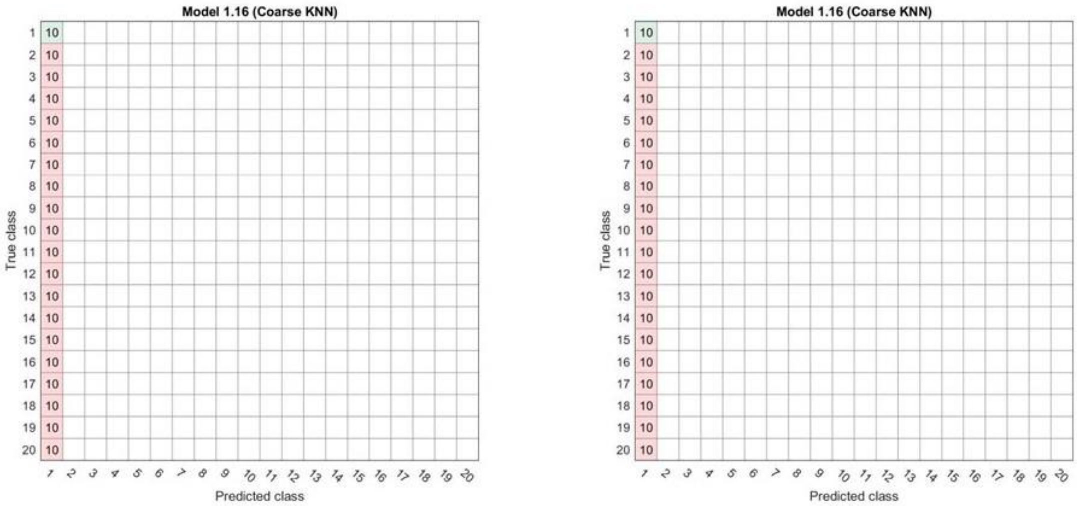

Figure 3.15: Decision matrix for Coarse k-NN classifier with and without feature selection

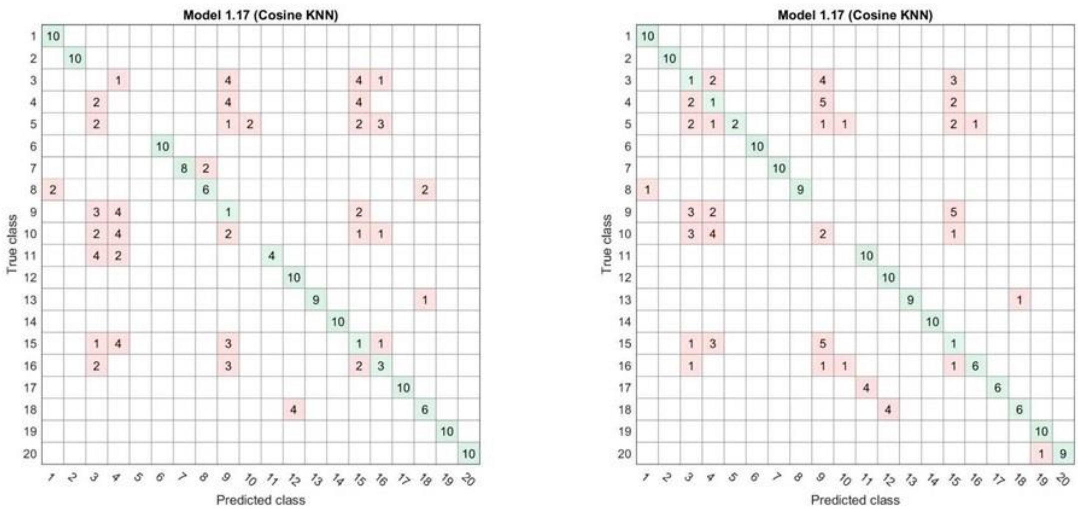

Figure 3.16: Decision matrix for Cosine k-NN classifier with and without feature selection

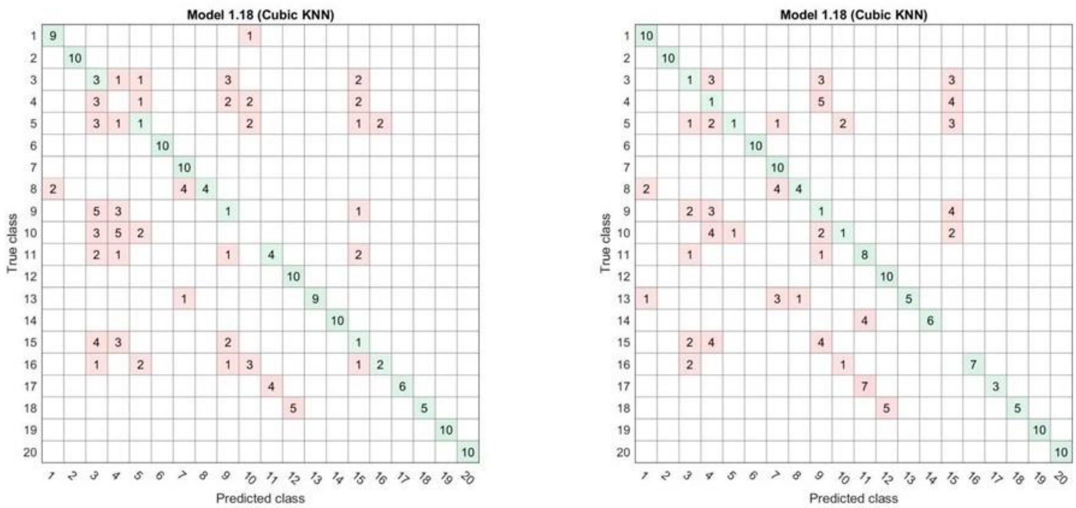

Figure 3.17: Decision matrix for Cubic k-NN classifier with and without feature selection

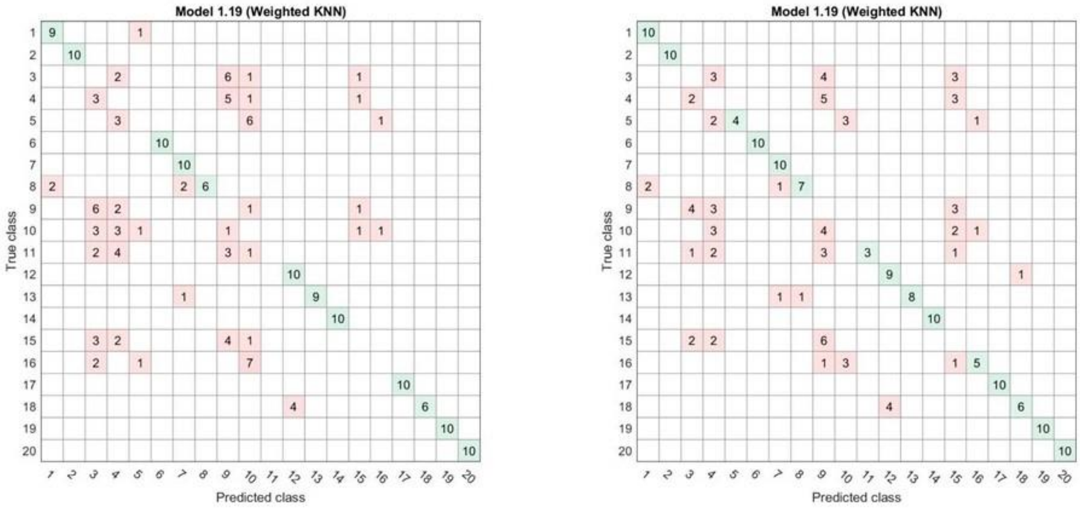

Figure 3.18: Decision matrix for Weighted k-NN classifier with and without feature selection

According to the results with and without feature selection, the algorithm gave the Ensemble Learning classifier accuracies listed below in Table 3.6.

Table 3.6: Ensemble Learning classifier results with and without feature selection

No	Classification Method	Without Feature Selection			With Feature Selection		
		Accuracy	Prediction Speed	Training Time	Accuracy	Prediction Speed	Training Time
1	Boosted Trees	81.5%	~2300 obs/sec	2.3082 sec	79.0%	~3200 obs/sec	1.5173 sec
2	Bagged Trees	70.5%	~2800 obs/sec	1.1705 sec	70.0%	~2300 obs/sec	0.9184 sec
3	Subspace Discriminant	89.5%	~1200 obs/sec	1.4247 sec	89.5%	~1100 obs/sec	1.1595 sec
4	Subspace k-NN	51.5%	~1300 obs/sec	0.9344 sec	53.5%	~1300 obs/sec	0.8997 sec
5	RUSBoosted Trees	76.5%	~3300 obs/sec	1.7989 sec	73.0%	~2900 obs/sec	1.6482 sec

Decision matrix of Ensemble Learning classifier for all classification method are shown in Figure 3.19 – 3.23. According to Boosted Tree, Bagged Tree and RUSBoosted Trees for Ensemble Learning classification method in Table 3.6, the way that is without feature selection is more successful than the way that is with feature selection. Nevertheless, the way that is with feature selection is more successful than the way that is without feature selection for Subspace k-NN classification method. There is no difference of accuracy percentage for Subspace Discriminant but the training time is decreasing with feature selection. But Linear Discriminant classification method is still the better way for this thesis.

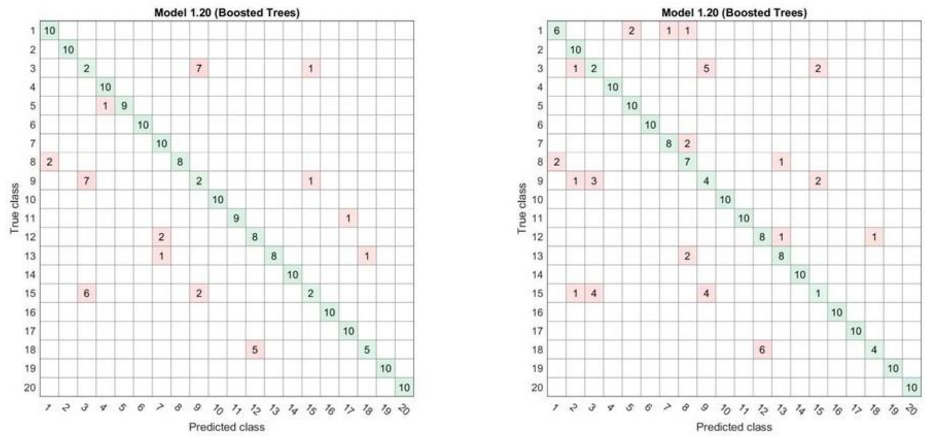

Figure 3.19: Decision matrix for Boosted Trees classifier with and without feature selection

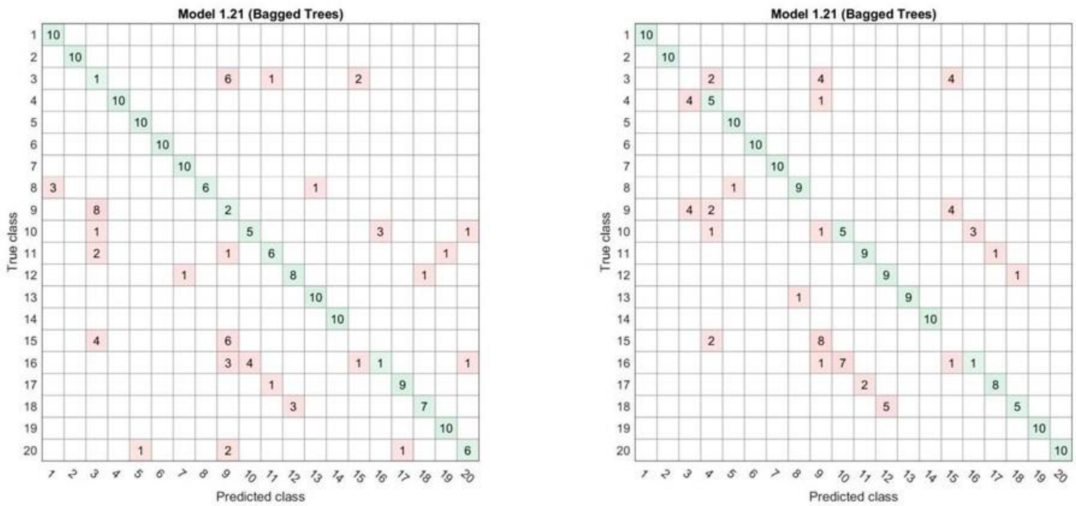

Figure 3.20: Decision matrix for Bagged Trees classifier with and without feature selection

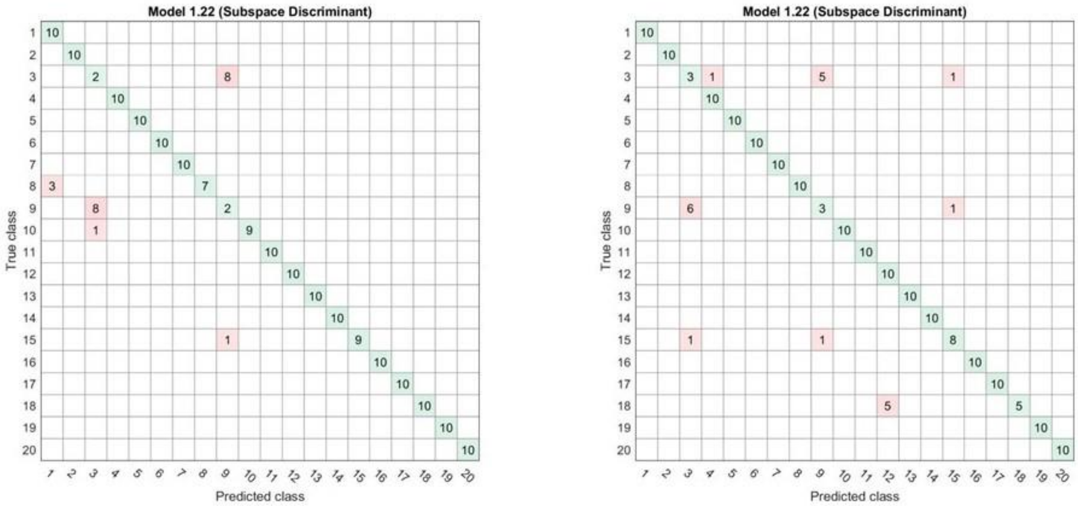

Figure 3.21: Decision matrix for Subspace Discriminant classifier with and without feature selection

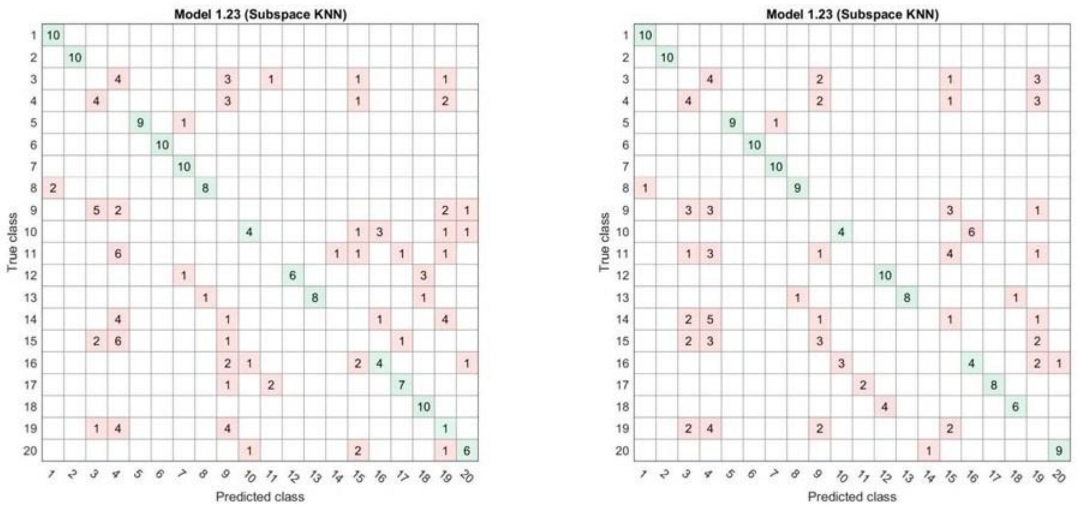

Figure 3.22: Decision matrix for Subspace k-NN classifier with and without feature selection

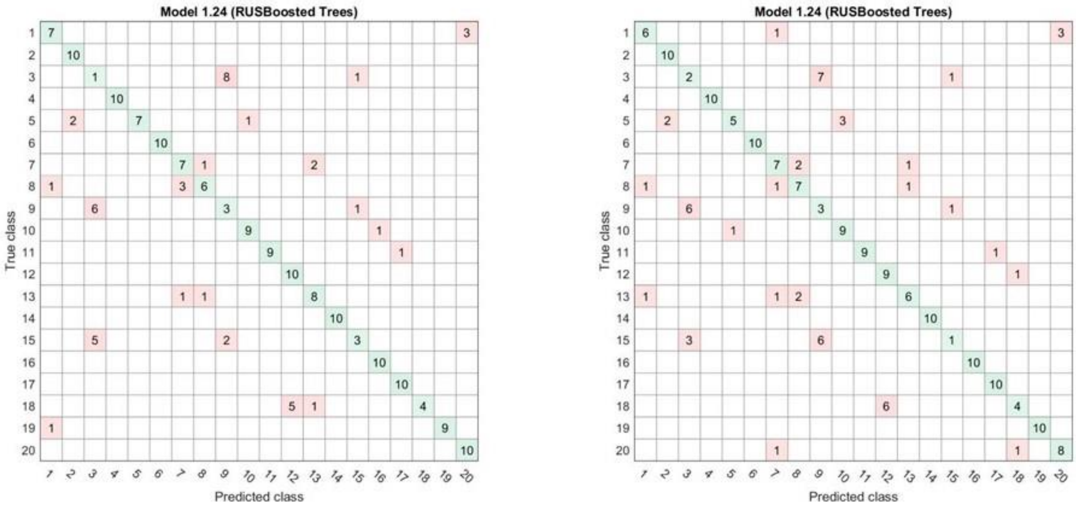

Figure 3.23: Decision matrix for RUSBoosted Trees classifier with and without feature selection

According to table for all applied classification methods, Linear Discriminant Analysis and Subspace Discriminant Analysis are the best with 93.5%, 89.5% of accuracies and 0.2055 sec, 1.1595 sec of training time respectively in this study. In another study, classifier accuracies were listed in Table 3.7.

Table 3.7: Accuracies of fault detection in other studies

No	Classification Method	Accuracy
1	Fuzzy/Bayesian [88]	88.52%
2	PCA [88]	67,21%
3	SVM [88]	41,3%
4	Hierarchical Neural Network [89]	73.0%
5	Artificial Neural Network [90]	91.18%
6	Auto Encoder & Long-Short Term Memory [91]	91.9%
7	Deep Stacking Network and Sparse Stacked Autoencoders [92]	83.2%

Chapter 4

Conclusion

The size of the data created with Industry 4.0 has increased and it has become difficult to process and make sense of this data by human hands. Changing maintenance approaches have made the use of ML methods popular so that production processes can continue without interruption. In the thesis study, preprocessing dataset and classification of 20 different faults were carried out with different ML methods on TEP, which is the online data set of IEEEDataPort that is obtained from a plant that contains nonlinear processes from various chemical units such as condenser, reactors, separators, strippers, and compressors.

As for the preprocessing part, Poincare Plot measures that are proven in the field of biomedical application are used in the fault classification method in order to extract new features from raw dataset. The one-way ANOVA test was used to find out which of the Poincare measures differed between faults at the 5% significance level. As a result of this test, the features that should be removed from the data set were determined and a preprocessed dataset was obtained besides the raw dataset. As for the training ML models, the algorithms such as Decision Tree, Discriminant Analysis, Naive Bayes, k-Nearest Neighbors, Support Vector Machine, and Ensemble Learning algorithms were utilized to classify the fault types from both raw and preprocessed datasets. The model accuracies are compared and the maximum classifier accuracies were 89.5% for the whole feature dataset using the Subspace Discriminant Algorithm of the Ensemble Learning Classifier method and 93.5% for the selected features only using the Linear Discriminant Analysis during this study. These performances could be comprehendible among the results achieved in similar studies.

This thesis study reveals how accurate fault detection is by using preprocessing parts and ML models with predictive maintenance method of machines used in industry.

Although the data set used in testing the applied methods is the data obtained from the real process environment, it is known that the real-life data have similar characteristics. As a result, it is obvious that ML approaches will provide serious benefits in maintenance work on the data obtained in the industry 4.0 environment. It is aimed that the thesis study, together with other studies in the literature, will contribute to the studies to be made to use ML models in the predictive maintenance approach, and that the maintenance approaches used in the industry will be less costly in this direction. Achieving more successful results with different ML and DL approaches and optimization algorithms creates motivation for future studies.

References

[1] Bathelt, A., Ricker, N. L., & Jelali, M. (2015). Revision of the Tennessee Eastman process model. IFAC-PapersOnLine, 48(8), 309-314.

[2] Li Z, & Wang K. Industry 4.0 – Potentials for Predictive Maintenance. International Workshop of Advanced Manufacturing and Automation (IWAMA), 2016.

[3] Endrenyi, J., Aboresheid, S., Allan, R. N., Anders, G. J., Asgarpoor, S., Billinton, R., ... & Grigg, C. (2001). The present status of maintenance strategies and the impact of maintenance on reliability. IEEE Transactions on Power Systems, 16(4), 638-646.

[4] Munirathinam, S. (2020). Industry 4.0: Industrial Internet of Things (IIOT). Advances in Computers, 117(1), 129-164.

[5] Rojko, A. (2017). Industry 4.0 concept: background and overview, International Journal of Interactive Mobile Technologies, 11(5), 77–90.

[6] Marino, R., Wisultschew, C., Otero, A., Lanza-Gutierrez, J. M., Portilla, J., & de la Torre, E. (2020). A machine-learning-based distributed system for fault diagnosis with scalable detection quality in industrial IoT. IEEE Internet of Things Journal, 8(6), 4339-4352.

[7] Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. D. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 137, 106024.

[8] Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. Journal of Applied Science and Technology Trends, 1(2), 56-70.

[9] Ding, X., & He, Q., (2017). Energy-Fluctuated Multiscale Feature Learning with Deep ConvNet for Intelligent Spindle Bearing Fault Diagnosis, IEEE Transactions on Instrumentation and Measurement, 66 (8), 1926 - 1935.

[10] Miclaus, S., Sarbu, A., & Bechet, P. (2021). Using Poincare Plots for Feature Extraction of the Dynamics of Electromagnetic Field Exposures when Using Different Protocols of Wi-Fi Communications. In 2021 8th International Conference On Wireless Communication And Sensor Networks (pp. 32-38).

[11] Yu, K., Guo, X., Liu, L., Li, J., Wang, H., Ling, Z., & Wu, X. (2020). Causality-based feature selection: Methods and evaluations. ACM Computing Surveys (CSUR), 53(5), 1-36.

[12] Chebel-Morello, B., Malinowski, S., & Senoussi, H. (2016). Feature selection for fault detection systems: application to the Tennessee Eastman process. Applied Intelligence, 44(1), 111-122.

[13] Xue, Y., Zhang, L., Wang, B., Zhang, Z., & Li, F. (2018). Nonlinear feature selection using Gaussian kernel SVM-RFE for fault diagnosis. Applied Intelligence, 48(10), 3306-3331.

[14] Sahin, S., & Isler, Y. (2013). Microcontroller-based robotics and SCADA experiments. IEEE Transactions on Education, 56(4), 424-429.

[15] Sahin, S., Olmez, M., & Isler, Y. (2009). Microcontroller-based experimental setup and experiments for SCADA education. IEEE Transactions on Education, 53(3), 437-444.

[16] Serneels V., & Pierre J. (2005). Bootstrap confidence intervals for trilinear partial least squares regression. Analytica Chimica Acta 544, 153-158.

[17] Isermann, R. (2005). Model-based fault-detection and diagnosis–status and applications. Annual Reviews in control, 29(1), 71-85.

[18] Isermann, R., & Balle, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. Control Engineering Practice, 5(5), 709-719.

[19] Yu, J., Chen, J., & Rashid, M. M. (2013). Multiway independent component analysis mixture model and mutual information-based fault detection and diagnosis approach of multiphase batch processes. AIChE Journal, 59(8), 2761-2779.

[20] Esakimuthu Pandarakone, S., Mizuno, Y., & Nakamura, H. (2019). A comparative study between machine learning algorithm and artificial intelligence neural network in detecting minor bearing fault of induction motors. Energies, 12(11), 2105.

[21] Liu, X., Pei, D., Lodewijks, G., Zhao, Z., & Mei, J. (2020). Acoustic signal-based fault detection on belt conveyor idlers using machine learning. Advanced Powder Technology, 31(7), 2689-2698.

[22] Kankar, P. K., Sharma, S. C., & Harsha, S. P. (2011). Fault diagnosis of ball bearings using machine learning methods. Expert Systems with Applications, 38(3), 1876-1886.

[23] Shao, H., Jiang, H., Zhang, H., & Liang, T. (2018). Electric Locomotive Bearing Fault Diagnosis Using a Novel Convolutional Deep Belief Network. IEEE Transactions on Industrial Electronics, 65(3), 2727–2736.

[24] Wen, L., Gao, L., & Li, X., (2019). A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49 (1), 136-144.

[25] Guo, L., Lei, Y., Xing, S., Yan, T., & Li, N. (2019). Deep Convolutional Transfer Learning Network: A New Method for Intelligent Fault Diagnosis of Machines with Unlabeled Data, IEEE Transactions on Industrial Electronics, 66 (9), 7316-7325.

[26] He, M., & He, D. (2017). Deep learning based approach for bearing fault diagnosis. IEEE Transactions on Industry Applications, 53(3), 3057-3065.

[27] Kumar, P., & Hati, A. S. (2021). Review on machine learning algorithm-based fault detection in induction motors. Archives of Computational Methods in Engineering, 28(3), 1929-1940.

[28] F. Dalvand, A. Kalantar, & M. S. Safizadeh, “A novel bearing condition monitoring method in induction motors based on instantaneous frequency of motor voltage,” IEEE Trans. Ind. Electron., vol. 63, no. 1, pp. 364–376, Jan. 2016.

[29] Elbouchikhi, E., Choqueuse, V., Amirat, Y., Benbouzid, M. E. H., & Turri, S. (2017). An efficient Hilbert–Huang transform-based bearing faults detection in induction machines. IEEE Transactions on Energy Conversion, 32(2), 401-413.

[30] Aydin, I., & Kaner, S. (2020). A new hybrid diagnosis of bearing faults based on time-frequency images and sparse representation. Traitement du Signal, Vol. 37, No. 6, pp. 907-918.

[31] Tang, H., Lu, S., Qian, G., Ding, J., Liu, Y., & Wang, Q. (2020). IoT-Based Signal Enhancement and Compression Method for Efficient Motor Bearing Fault Diagnosis. IEEE Sensors Journal, 21(2), 1820-1828.

[32] Tastimur, C., Karakose, M., Aydin, I., & Akin, E. (2018). Defect Diagnosis of Rolling Element Bearing using Deep Learning. 2018 International Conference on Artificial Intelligence and Data Processing (IDAP).

[33] Taqvi, S. A. A., Zabiri, H., Tufa, L. D., Uddin, F., Fatima, S. A., & Maulud, A. S. (2021). A Review on Data-Driven Learning Approaches for Fault Detection and Diagnosis in Chemical Processes. ChemBioEng Reviews, 8(3), 239-259.

[34] Krishnannair, S., & Aldrich, C. (2017). Fault detection in the Tennessee Eastman benchmark process with nonlinear singular spectrum analysis. IFAC-PapersOnLine, 50(1), 8005-8010.

[35] Golińska, A. K. (2013). Poincaré plots in analysis of selected biomedical signals. Studies in logic, grammar and rhetoric, 35(1), 117-127.

[36] Brennan, M., Palaniswami, M., & Kamen, P. (2001). Do existing measures of Poincare plot geometry reflect nonlinear features of heart rate variability? IEEE transactions on biomedical engineering, 48(11), 1342-1347.

[37] Karmakar, C. K., Khandoker, A. H., Voss, A., & Palaniswami, M. (2011). Sensitivity of temporal heart rate variability in Poincaré plot to changes in parasympathetic nervous system activity. Biomedical Engineering Online, 10(1), 1-14.

[38] Zanoli, S. M., Astolfi, G., & Barboni, L. (2010). Principal component analysis based on anova test for multishaft centrifugal compressor fault detection and diagnosis. IFAC Proceedings Volumes, 43(21), 90-96.

[39] Andrianajaina, T., RAZAFIMAHEFA, D., Héraud, N., & Sambatra, E. J. R. (2020). ANOVA fault detection method in photovoltaic system. In 2020 7th International Conference on Control, Decision and Information Technologies (CoDIT) (Vol. 1, pp. 37-41). IEEE.

[40] Tian, Z., & Hoo, K. A. (2005). Multiple Model-Based Control of the Tennessee−Eastman Process. Industrial & engineering chemistry research, 44(9), 3187-3202.

[41] Bathelt, A., Ricker, N. L., & Jelali, M. (2015). Revision of the Tennessee Eastman process model. IFAC-PapersOnLine, 48(8), 309-314.

[42] Räsänen, P. (2020). Event-driven approach for Tennessee Eastman process.

[43] Ricker, N. L. (1996). Decentralized control of the Tennessee Eastman challenge process. Journal of Process Control, 6(4), 205-221.

[44] Kang, S. O., Lee, E. B., & Baek, H. K. (2019). A digitization and conversion tool for imaged drawings to intelligent piping and instrumentation diagrams (P&ID). Energies, 12(13), 2593.

[45] Koltun, G., Kolter, M., & Vogel-Heuser, B. (2018). Automated generation of modular PLC control software from P&ID diagrams in process industry. In 2018 IEEE International Systems Engineering Symposium (ISSE) (pp. 1-8). IEEE.

[46] Larsson, T., Hestetun, K., Hovland, E., & Skogestad, S. (2001). Self-optimizing control of a large-scale plant: The Tennessee Eastman process. Industrial & engineering chemistry research, 40(22), 4889-4901.

[47] Lau, C. K., Ghosh, K., Hussain, M. A., & Hassan, C. C. (2013). Fault diagnosis of Tennessee Eastman process with multi-scale PCA and ANFIS. Chemometrics and Intelligent Laboratory Systems, 120, 1-14.

[48] Rato, T. J., & Reis, M. S. (2013). Fault detection in the Tennessee Eastman benchmark process using dynamic principal components analysis based on decorrelated residuals (DPCA-DR). Chemometrics and Intelligent Laboratory Systems, 125, 101-108.

[49] Marciano, F., Migaux, M. L., Acanfora, D., Furgi, G., & Rengo, F. (1994). Quantification of Poincare'maps for the evaluation of heart rate variability. In Computers in Cardiology 1994 (pp. 577-580). IEEE.

[50] Brennan, M., Palaniswami, M., & Kamen, P. (2001). Do existing measures of Poincare plot geometry reflect nonlinear features of heart rate variability. IEEE transactions on biomedical engineering, 48(11), 1342-1347.

[51] Smith, A. L., Reynolds, K. J., & Owen, H. (2007). Correlated Poincaré indices for measuring heart rate variability. Australasian physical & engineering sciences in medicine, 30(4), 336-341.

[52] Brennan, M., Palaniswami, M., & Kamen, P. (2001). Do existing measures of Poincare plot geometry reflect nonlinear features of heart rate variability. IEEE transactions on biomedical engineering, 48(11), 1342-1347

[53] Stein, P. K., Le, Q., Domitrovich, P. P., & Cast Investigators. (2008). Development of more erratic heart rate patterns is associated with mortality post–myocardial infarction. Journal of Electro cardiology, 41(2), 110-115.

[54] Capurro, A., Malta, C. P., Diambra, L., Contreras, P., & Migliaro, E. R. (2007). Respiratory modulation of heart beat-to-beat interval in diabetic patients. Physica A: Statistical Mechanics and its Applications, 378(2), 336-344.

[55] Thakre, T. P., & Smith, M. L. (2006). Loss of lag-response curvilinearity of indices of heart rate variability in congestive heart failure. BMC cardiovascular disorders, 6(1), 1-10.

[56] Brennan, M., Palaniswami, M., & Kamen, P. (2001). Do existing measures of Poincare plot geometry reflect nonlinear features of heart rate variability. IEEE Transactions on Biomedical Engineering, 48(11), 1342-1347.

[57] Montgomery, D. C., & Runger, G. C. (2010). Applied statistics and probability for engineers. John Wiley & Sons

[58] Saleem, M. A. (2012). The impact of socio-economic factors on small business success. Geografia, 8(1).

[59] Fisher, R. A. (1992). Statistical methods for research workers. In Breakthroughs in statistics (pp. 66-70). Springer, New York, NY.

[60] Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. Emerging Artificial Intelligence Applications in Computer Engineering, 160(1), 3-24.

[61] Kaparthi, S., & Bumblauskas, D. (2020). Designing predictive maintenance systems using decision tree-based machine learning techniques. International Journal of Quality & Reliability Management.

[62] Hill, D. A., Delaney, L. M., & Roncal, S. (1997). A chi-square automatic interaction detection (CHAID) analysis of factors determining trauma outcomes. Journal of Trauma and Acute Care Surgery, 42(1), 62-66.

[63] Quinlan, J. R. (1986). Induction of decision trees. Machine learning, 1(1), 81-106.

[64] Breiman, L. (1984). Classification and Regression Trees. New York: Routledge.

[65] Hill, D. A., Delaney, L. M., & Roncal, S. (1997). A chi-square automatic interaction detection (CHAID) analysis of factors determining trauma outcomes. Journal of Trauma and Acute Care Surgery, 42(1), 62-66.

[66] Friedman, J. H. (1991). Multivariate adaptive regression splines. The annals of statistics, 19(1), 1-67.

[67] Rigatti, S. J. (2017). Random forest. Journal of Insurance Medicine, 47(1), 31-39.

[68] Klecka, W. R., Iversen, G. R., & Klecka, W. R. (1980). Discriminant analysis (Vol. 19). Sage.

[69] Johnson, R.A. & Wichern, D.W. (2002). Applied Multivariate Statistical Analysis. Prentice Hall, New Jersey.

[70] Kökver, Y., Barişçi, N., Çiftçi, A., & Ekmekçi, Y. (2014). Hipertansiyona Etki Eden Faktörlerin Veri Madenciliği Yöntemleriyle İncelenmesi. Engineering Sciences, 9(2), 15-25.

[71] Nazir, S., Khan, S., Khan, H. U., Ali, S., García-Magariño, I., Atan, R. B., & Nawaz, M. (2020). A comprehensive analysis of healthcare big data management, analytics and scientific programming. IEEE Access, 8, 95714-95733.

[72] Ghaffari Razin, S. R., & Hooshangi, N. (2022). Modeling and prediction of the ionospheric total electron content time series using support vector machine in 2007-2018. Journal of the Earth and Space Physics, 48(1), 173-187.

[73] Karthika, P., Barani Sundaram, B., Kedir, T., Sorsa, T. T., Sreenivas, N., Mishra, M. K., & Thirumoorthy, D. (2021). Steganalysis for Images Security Classification in Machine Learning Using SVM. In Computational Vision and Bio-Inspired Computing (pp. 855-867). Springer, Singapore.

[74] Vapnik, V. The Nature of Statistical Learning Theory. NY: Springer-Verlag. 1995.

[75] Bottou, L., & Chih-Jen L.. National Taiwan University, Department of Computer Science, “Support Vector Machine Solvers.” (2009).

[76] Pradhan, B. (2013). A Comparative Study on The Predictive Ability of the Decision Tree, Support Vector Machine and Neuro-Fuzzy Models in Landslide Susceptibility Mapping Using GIS. Computers & Geosciences. 51: 350-365.

[77] N. Cristianini & J. Shawe-Taylor, An introduction to support vector machines: and other kernel-based learning methods. Cambridge University Press, 2000.

[78] Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. In OTM Confederated International Conferences" On the Move to Meaningful Internet Systems" (pp. 986-996). Springer, Berlin, Heidelberg.

[79] Wu, Y., Ianakiev, K., & Govindaraju, V. (2002). Improved k-nearest neighbor classification. Pattern recognition, 35(10), 2311-2318.

[80] Kowalski, B. R., & Bender, C. F. (1972). k-Nearest Neighbor Classification Rule (pattern recognition) applied to nuclear magnetic resonance spectral interpretation. Analytical Chemistry, 44(8), 1405-1411.

[81] Hwang, W. J., & Wen, K. W. (1998). Fast kNN classification algorithm based on partial distance search. Electronics Letters, 34(21), 2062-2063.

[82] Bhuvaneswari, P., & Therese, A. B. (2015). Detection of cancer in lung with k-nn classification using genetic algorithm. Procedia Materials Science, 10, 433-440.

[83] Davison AC, & Hinkley DV (1997). Bootstrap methods and their application. Cambridge University Press, Cambridge

[84] Diciccio TJ, & Romano JP (1988). A review of bootstrap confidence intervals. Journal of the Royal Statistical Society. Series B (Methodological) 50:338

[85] Breiman, L. (1996). Bagging predictors. Machine learning, 24(2), 123-140.

[86] Zhang, S., Yu, Q., He, H., Zhu, F., Wu, P., Gu, L., & Jiang, S. (2020). iDHS-DSAMS: Identifying DNase I hypersensitive sites based on the dinucleotide property matrix and ensemble bagged tree. Genomics, 112(2), 1282-1289.

[87] Agarwal, S., & Chowdary, C. R. (2020). A-Stacking and A-Bagging: Adaptive versions of ensemble learning algorithms for spoof fingerprint detection. Expert Systems with Applications, 146, 113160.

[88] D’Angelo, M. F., Palhares, R. M., Camargos Filho, M. C., Maia, R. D., Mendes, J. B., & Ekel, P. Y. (2016). A new fault classification approach applied to Tennessee Eastman benchmark process. Applied Soft Computing, 49, 676-686.

[89] Eslamloueyan, R. (2011). Designing a hierarchical neural network based on fuzzy clustering for fault diagnosis of the Tennessee–Eastman process. Applied soft computing, 11(1), 1407-1415.

[90] Heo, S., & Lee, J. H. (2018). Fault detection and classification using artificial neural networks. IFAC-PapersOnLine, 51(18), 470-475.

[91] Park, P., Marco, P. D., Shin, H., & Bang, J. (2019). Fault detection and diagnosis using combined autoencoder and long short-term memory network. Sensors, 19(21), 4612.

[92] Chadha, G. S., & Schwung, A. (2017). Comparison of deep neural network architectures for fault detection in Tennessee Eastman process. In 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (pp. 1-8). IEEE.

Appendices

Appendix A

Publications from the Thesis

Conference Papers

1. Fault Detection and Diagnosis on Process Control Systems Using Ensemble Learning Algorithms from Poincare Plot Measures (HORA 2021 & European Journal of Science and Technology Special Issue 26, pp. 30-34, July 2021) (DOI 10.31590/ejosat. 952761)

2. Fault Detection and Diagnosis on Process Control Systems Using k-Nearest Neighbors from Poincare Plot Measures (Book of Abstracts, p. 34, 3rd International Conference of Applied Sciences, Engineering and Mathematics (IBU-ICASEM 2021), June 3-5, Skopje/North Macedonia, 2021)

Republic of Turkey

İzmir Kâtip Çelebi University

Graduate School of Natural and Applied Sciences

Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms

Department of Electrical & Electronics Engineering

Master's Thesis

Emre ÇANCIOĞLU

ORCID 0000-0002-9918-4668

Thesis Advisor: Assoc. Prof. Savaş ŞAHİN

June 2022

CANCIOGLU

Poincaré Plot-Based Fault Detection on Tennessee Eastman Process Using Various Machine Learning Algorithms

MASTER’ S THESIS

2022

Curriculum Vitae

Name Surname: Emre ÇANCIOĞLU

Education:

2014–2019 İzmir Kâtip Çelebi University, Electrical & Electronics Engineering

2019–2022 İzmir Kâtip Çelebi University, Electrical & Electronics Engineering

Work Experience:

2017 – 2018 Dimes A.Ş.

2018 – 2018 Man Türkiye A.Ş.

2018 – Present Doğuş Vana San. ve Tic. A.Ş.

Publications:

1. Fault Detection and Diagnosis on Process Control Systems Using Ensemble Learning Algorithms from Poincare Plot Measures (HORA 2021 & European Journal of Science and Technology, Special Issue 26, pp. 30-34, July 2021) (DOI 10.31590/ejosat. 952761)
2. Fault Detection and Diagnosis on Process Control Systems Using k-Nearest Neighbors from Poincare Plot Measures (Book of Abstracts, p. 34, 3rd International Conference of Applied Sciences, Engineering and Mathematics (IBU-ICASEM 2021), June 3-5, Skopje/North Macedonia, 2021)
3. Heart Sounds Analysis and Classification Based on Long-Short Term Memory (Journal of Intelligent Systems with Applications 3(1), pp. 25-28, 2020) (DOI: 10.54856/jiswa.202005104)
4. Design and Implementation of Digital Filters for ECG Data Based on Field Programmable Gate Array and MATLAB (Journal of Intelligent Systems with Applications 3(1), pp. 17-19, 2020) (DOI: 10.54856/jiswa.202005102)