RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA
RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Électrotechnique

End-of-study project dissertation

for obtaining the State Engineer's degree in "Électrotechnique"

Comparative Analysis of Machine Learning Methods for Power Transformer Oil Diagnosis Using Dissolved Gas Analysis

BOUKHARI Imed-Eddine

Under the supervision of: **Dr. BENMAHAMED Youcef** ENP

Presented publicly 23/06/2024

Jury's composition:

President:	Pr. HELLAL Abdelhafid	ENP
Supervisor:	Dr. BENMAHAMED Youcef	ENP
Examiner:	Pr. TEGUAR Madjid	ENP

ENP 2024

Électrotechnique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Alger Algérie.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

Département d'Électrotechnique

End-of-study project dissertation

for obtaining the State Engineer's degree in "Électrotechnique"

Comparative Analysis of Machine Learning Methods for Power Transformer Oil Diagnosis Using Dissolved Gas Analysis

BOUKHARI Imed-Eddine

Under the supervision of: **Dr. BENMAHAMED Youcef** ENP

Presented publicly 23/06/2024

Jury's composition:

President:	Pr. HELLAL Abdelhafid	ENP
Supervisor:	Dr. BENMAHAMED Youcef	ENP
Examiner:	Pr. TEGUAR Madjid	ENP

ENP 2024

Électrotechnique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Alger Algérie.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Électrotechnique

Mémoire de projet de fin d'études

Pour l'obtention du diplôme d'ingénieur d'état en Électrotechnique

Analyse comparative des méthodes d'apprentissage automatique pour le diagnostic de l'huile des transformateurs de puissance en utilisant l'analyse des gaz dissous

BOUKHARI Imed-Eddine

Sous la direction de : **Dr. BENMAHAMED Youcef** ENP

Présenté et soutenu publiquement le (23/06/2024)

Composition du jury :

Président :	Pr. HELLAL Abdelhafid	ENP
Promoteur :	Dr. BENMAHAMED Youcef	ENP
Examinateur :	Pr. TEGUAR Madjid	ENP

ENP 2024

Électrotechnique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Alger Algérie.

ملخص

يركز هذا العمل على تشخيص حالة زيت محولات الطاقة من خلال تحليل الغازات المذابة والتي تشمل C_2H_4 ، C_2H_2، CH_4، H_2 و C_2H_6.

لهذا الغرض، تم تطوير العديد من خوارزميات التعلم الآلي. تم النظر في ثمانية متجهات إدخال و استخدام أربعة تقنيات معالجة مسبقة. تحتوي قاعدة البيانات المستخدمة على 666 عينة، تم اختيار 506 منها للتدريب و160 للاختبار. أخذا بالمعايير الدولية مثل IEC و IEEE، تم اعتبار ستة عيوب كهربائية وحرارية وهي PD، D1، D2، T1، T2 و T3. تم تحقيق أفضل معدل تشخيص بنسبة 99.375 % بإستعمال شجرة قرار خاصة.

الكلمات المفتاحية: محول طاقة، زيت عازل، تشخيص، تحليل الغازات المذابة، عيوب كهربائية وحرارية، التعلم الآلي.

Resumé

Ce travail se concentre sur le diagnostic de l'état de l'huile des transformateurs de puissance par l'analyse des gaz dissous comprenant H_2, CH_4, C_2H_2, C_2H_4 et C_2H_6. À cette fin, de nombreux algorithmes d'apprentissage automatique ont été développés. Huit vecteurs d'entrée ont été considérés et plusieurs techniques de prétraitement ont été utilisées. La base de données utilisée contient 666 échantillons, dont 506 sont sélectionnés pour l'entraînement et 160 pour le test. Inspiré par des normes internationales telles que IEC et IEEE, six défauts électriques et thermiques ont été considérés, à savoir PD, D1, D2, T1, T2 et T3. Le meilleur taux de diagnostic est de 99,375 % a été obtenu en utilisant un arbre de décision personnalisé.

Mots clés: Transformateur de puissance, huile isolante, diagnostic, analyse des gaz dissous défauts électrique et thermiques, Apprentissage automatique.

Abstract

This work focuses on diagnosing the condition of power transformer oil through dissolved gas analysis consisting of H_2, CH_4, C_2H_2, C_2H_4, and C_2H_6. For this purpose, many machine learning algorithms have been developed. eight input vectors have been considered, and several pre-processing techniques were used. The database used contains 666 samples, of which 506 are selected for training and 160 for testing. Inspired by international standards such as IEC and IEEE, six electrical and thermal faults have been considered, namely PD, D1, D2, T1, T2, and T3. The best diagnostic rate of 99.375% was achieved using a custom-built decision tree.

Keywords: Power transformer, insulating oil, diagnosis, dissolved gas analysis, electrical and thermal faults, Machine Learning.

Acknowledgements

Over the past five years, I have dedicated the majority of my time to my school, often extending well beyond regular hours. This period has been profoundly shaping, marked by numerous highs and lows. It was during this time that I encountered remarkable individuals, became part of inspiring clubs and groups, received invaluable guidance from those who came before me, and did my best to contribute positively to my peers' lives.

I wish to express my deepest gratitude and affection to my family members. Their unwavering support, even when faced with my late-night returns—often after spending the entire day at school—has been nothing short of extraordinary. They have made me feel welcomed, even when I most often arrived home late.

Following my family, I am immensely grateful to my friends within the school community. Without them, reaching this milestone would not have been possible. A heartfelt thank you goes to the VIC ENP, a club that significantly impacted my life. My appreciation extends to Ascem and the Vniverse friends, as well as my high school friends.

Among these friendships, a special mention goes to my friends from the Bouraoui Land, And those from the EEA (Sbigha and twin tower), including Nassim, Said, Rosso and Mehdi, whose assistance during my PFE was invaluable.

I would like to express my profound gratitude to my supervisor, Dr. Benmahamed, for his guidance, support, and innovative ideas throughout this journey. I also extend my sincere thanks to Pr. Hellal for accepting to be the president of the jury. Additionally, I am grateful to all the professors who have contributed to my education over the past five years at ENP, especially Pr. Teguar for examining my dissertation. My thanks also go to Pr. Adnane, Pr. Mahdi, and Pr. Boughrara.

Imed.

Contents

List of Tables

List of Figures

General Introduction

0.1 Introduction

Power transformers are crucial and essential components in the transmission and distribution chain of electrical energy. These electrical equipment, which are quite expensive, must operate correctly for many years. The lifespan of a power transformer depends closely on the lifespan of its insulation system, generally consisting of a traditional solid component (such as paper, printed cardboard, etc.) and a dielectric fluid [1].

Immersed power transformers use insulating oil as the dielectric fluid due to its good physio-chemical and electrical properties and low cost. Besides insulation, this oil helps dissipate the heat generated by the magnetic circuit and winding's. During operation, the insulating oil is subjected to various constraints, including electrical, thermal, and chemical stresses. These constraints lead to the progressive degradation of the insulating oil and eventually result in the transformer being taken offline if not analyzed in time [2, 3]. Hence, various oil analysis techniques are proposed to diagnose the internal condition of power transformers. The most popular ones are physio-chemical analyses [1, 4, 5] and dissolved gas analysis (DGA) [3, 5].

Monitoring to Detect the initial degradation of an insulation, which could lead to its total breakdown, enables us to foresee an incident, and thus possibly avoid a widespread power cut and the shutdown of factories and other important socio-economic structures.

DGA(Dissolved Gas Analysis) is a widely used diagnostic technique based on interpreting the concentrations of dissolved gases in the oil. Under the influence of electrical and thermal stresses, the oil decomposes, releasing gases in small quantities [3, 5]. These gases are separated and quantified using gas chromatography [5]. DGA can be performed in the laboratory on samples taken (offline mode) which is the oldest and most used method, There is also the online mode by introducing sensors into transformers in service. [6].

The five main gases resulting from the oil decomposition are hydrogen (H_2), methane (CH_4), acetylene (C_2H_2), ethylene (C_2H_4), and ethane (C_2H_6). The proportions of the concentrations of these gases in a sample can determine the type of faults, In addition, the decomposition of insulation paper produces carbon monoxide (CO) and carbon dioxide (CO_2).[7]. According to IEC 60599-2008 [8] and IEEE Standard C57.104-2019 [9], six electrical and thermal faults exist, consisting of partial discharges (PD), low-energy discharges (D1), high-energy discharges (D2), thermal faults for T $<$ 300°C (T1), thermal faults for T from 300°C to 700°C (T2), and finally, thermal faults for T $>$ 700°C (T3).

In order to interpret the results of DGA of transformer oil, various conventional techniques have been developed [10]. Detailed in the second chapter of this present dissertation, these techniques include:

- Methods using individual gas concentrations in ppm such as the IEEE Standard C57.104-2019 method, also known as Total Dissolved Combustible Gas (TDCG), or in percentages such as the Key Gas Method (KGM) [9];
- Methods of gas concentration ratios, such as those of Dornenburg [11], Rogers [12], and IEC 60599-2008 [8];
- Graphic methods of Duval using percentage concentration ratios, such as the triangle with its seven versions (1 to 7) [13] and the pentagon with its two variants (I and II) [14].

Conventional methods have been the subject of several research works and are still relevant. Syafruddin and Nugroho [15] applied TDCG, KGM, Dornenburg's and Rogers' ratios, as well as Duval's triangle to determine faults in transformer oil. The faults detected were of a thermal nature. Benmahamed [7] used Duval's triangle, Duval's pentagon I, Rogers' ratios, and IEC 60599 ratios. Duval's triangle provided the best diagnostic result, with a success rate of 90%. Furthermore, Dukam et al [16] proposed an intermediate diagnostic method between Duval's triangle and Duval's pentagon I. It is a four-dimensional simplexe which have ten triangular faces. The proposed technique successfully competed with conventional ones, achieving the best detection rate of 93%.

In order to improve transformer oil diagnosis through DGA, many techniques integrate traditional methods with those based on artificial intelligence and metaheuristics. Benmahamed et al [7, 17] developed two algorithms to improve the classification rate of Duval pentagon I. The first (Duval pentagon-SVM-PSO) consists of a combination of Duval's pentagon and Support Vector Machines (SVM) that reached an accuracy of 98%, with parameters optimized by the Particle Swarm Optimization (PSO) technique. The second (Duval pentagon-KNN) combines Duval's pentagon with the k-Nearest Neighbors (kNN) algorithm. The first algorithm achieved a success rate of 98% compared to 92% for the second. In another research work, Benmahamed et al [18] applied Naive Bayes (NB), to diagnose transformer oil insulation by DGA. The Naive Bayes algorithm provided a relatively low accuracy rate of 86%. Furthermore, Benmahamed et al [19] developed two classifiers. The first is Gaussian, and the second (SVM-Bat) uses Support Vector Machines (SVM), with parameters optimized by the Bat algorithm. The success rate of SVM-Bat is 93.75% compared to 69.37% for Gaussian. Also, Guerbas and el in [20] used neural networks with PSO to acheive a 99.375% Accuracy, The best so far. Finally, Kherif et al [21] developed an algorithm combining KNN with the decision tree principle. An accuracy rate exceeding 93% was obtained, demonstrating the effectiveness of the proposed algorithm and because of it's effectiveness we will see later in chapter 4 how we used this principle to achieve a 99.375% accuracy.

This dissertation aims to explore different machine learning (ML) algorithm's for the fault diagnosis of power transformers through Dissolved Gas Analysis (DGA) and to compare its effectiveness with other conventional and AI-based diagnostic methods.

For this purpose, We started by using SVM and Ensemble learning methods to get better insights about this methods achieving an accuracy rate of 96.625% and in order to get better results, A decision tree principle with ML algorithm's is developed. We opted for a decision tree combined with Extreme Gradient Boosting (XGBoost), and Random Forest (RF) which at the end gave us a 99.375% accuracy. A total of 666 sets of sample data are considered, with three-quarters used for the learning process (506 samples) and the remaining quarter used for testing (160 samples). The six classes of faults (PD, D1, D2, T1, T2, and T3) recommended by IEC 60599-2008 and IEEE Standard C57.104-2008 are adopted [8, 9]. A comparative study is conducted among the different algorithms used by us and those from other papers. Thus, the present dissertation is structured around four distinct chapters.

The first chapter focuses on generalities about insulating oils for power transformers. We first present the technology of such transformers by showing their locations in electrical installations, their classification and constitution, as well as their sources of failures. We then discuss the different components of the insulation system, namely the insulating part (paper and cardboard) and the liquid part (mainly composed of insulating oils). We reveal the composition, types, and roles of each component. The different electrical, physical, and chemical properties of insulating oils, as well as the influence of the main aging factors, are also mentioned. We conclude this chapter by illustrating the diagnosis of transformer insulating oils by dissolved gas analysis.

In the second chapter, we present the main conventional (traditional) diagnostic techniques following the three major categories mentioned earlier. The first, using concentrations of dissolved gases, in ppm or percentages individually, concerns the method of IEEE Standard C57.104-2019 or TDCG (concentrations in ppm) and the Key Gas Method (KGM) (concentrations in percentages) [9]. The second, employing concentration ratios, includes Dornenburg's, Rogers', and IEC 60599-2008 methods [8]. The last category, based on gas concentration ratios, includes the two graphical representations of Duval, namely the triangle and the pentagon with their variants. These representations actually use the percentage concentration ratios of dissolved gases in the insulating oil. The advantages and disadvantages of conventional methods are also revealed.

Chapter three is dedicated to the theoretical aspects of the machine learning algorithms used in our study, specifically Support Vector Machines (SVM) and Ensemble Learning methods (EL). We begin by introducing these algorithms, delving into the mathematical foundations and theoretical principles that underpin them. First, we present Support Vector Machines (SVM). The mathematical formulation is thoroughly discussed to provide a comprehensive understanding of SVM's capabilities and limitations. Next, we focus on Ensemble Learning methods. We emphasize two main types of ensemble learning: Bagging and Boosting. For Bagging methods, we concentrate on Random Forest and then Extra-Tree, explaining its structure and the process of building multiple decision trees. For Boosting methods, We highlight Extreme Gradient Boosting (XGBoost) and its iterative approach to model building.

The fourth and final chapter presents the results obtained from using the four machine learning algorithms discussed in Chapter III. We begin by detailing the tools and software used, followed by a description of our input vectors and the preprocessing techniques applied to our data. Subsequently, we discuss and interpret all the results obtained. Based on these results, we select three algorithms, two input vectors, and two preprocessing techniques, which are then combined using a decision tree principle, achieving an accuracy of 99.375%. Additionally, we present a final table comparing our work with other studies on the same test data.

In conclusion, our dissertation underscores the effectiveness of our AI-based approach in fault diagnosis, utilizing Support Vector Machines (SVM), Ensemble Learning methods, and the decision tree principle. Our study demonstrates significant enhancements in accuracy.

Chapter 1

Insulating oils for transformers

1.1 Introduction

The escalating need for electrical power necessitates the deployment of high-quality transformers catering to both transmission and distribution systems. Insulating oils play a pivotal role as cooling and insulating agents within transformers, with a myriad of oil types engineered to ensure optimal insulation. Key considerations include stability against oxidation and thermal fluctuations. Mineral oil emerges as the predominant choice among these oils. This chapter offers an in-depth exploration of the primary properties of insulating oils, accounting for the diverse operational constraints they encounter. It commences with a concise introduction to power transformers, followed by an examination of their malfunctions and causes, and the essential properties of insulating oils. Additionally, It talk about the aging processes of these oils and the diagnostic methods used to ensure transformer reliability And it concludes with the different types of faults that can occur in power transformers.

1.2 Power transformers

A *power transformer* is a static AC electrical machine **(with no moving parts, and therefore no mechanical losses)** featuring a single, closed ferromagnetic circuit [22]. The cores are wound with concentric, tightly coupled coils, with no electrical connections between them. The transformer transmits electrical power electromagnetically, at a fixed frequency, from a primary circuit on the receiver side to a secondary circuit on the generator. Generally speaking, the voltage and current magnitudes change, but the power remains stable. The IEC divides power transformers in oil into three categories according to apparent power [23].

- Distribution transformer: up to 2,500 kVA three-phase or 833 kVA single-phase.
- Medium-power transformer: from 2.5 MVA up to 100 MVA three-phase and 33.3 MVA single-phase.
- High-power transformer: three-phase over 100 MVA, single-phase over 100 MVA.

Figure 1.1 illustrates the main parts of a power transformer.

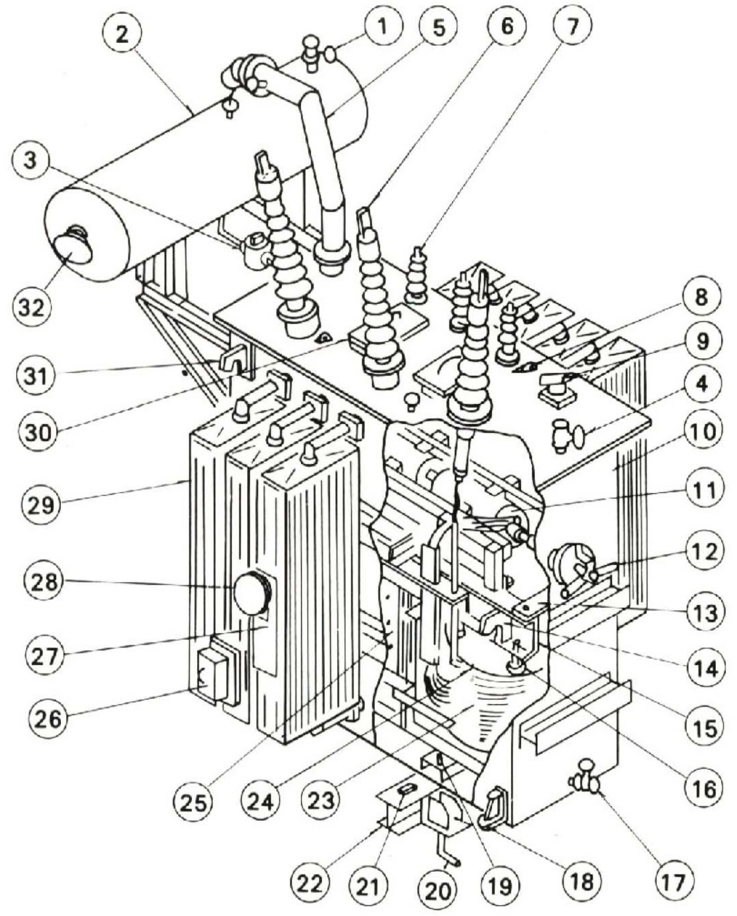


Figure 1.1: View of a power transformer [24]

The Table 1.3 have the power transformer parts as enumerated in Figure 1.1

Table 1.1: Power transformer parts [24]

N		**N**		**N**	
1.	Oil filter valve	12.	Tap changer handle	23.	Coil
2.	Conservator	13.	Fastener for core and coil	24.	Coil pressure plate
3.	Buchholz relay	14.	Lifting hook for core and coil	25.	Core
4.	Oil filter valve	15.	End frame	26.	Terminal box
5.	Pressure-relief vent	16.	Coil pressure bolt	27.	Rating plate
6.	High-voltage bushing	17.	Oil drain valve	28.	Dial thermometer
7.	Low-voltage bushing	18.	Jacking boss	29.	Radiator
8.	Suspension lug	19.	Stopper	30.	Manhole
9.	B C T Terminal	20.	Foundation bolt	31.	Lifting hook
10.	Tank	21.	Grounding terminal	32.	oil level gauge.
11.	De-energized tap changer	22.	Skid base		

1.3 Malfunctions, causes and statistics

Power transformer failures are one of the most frequent causes of long-term power supply interruptions, with serious repercussions on electrical system reliability. Transformer failure can occur due to a variety of causes and conditions. they can be defined as [23]:

- A transformer failure followed by forced unavailability of service (winding damage, faulty tap changer, etc.);
- A problem requiring the transformer to be disconnected for repair, or extensive on-site maintenance (excessive gas production, presence of moisture, etc.).

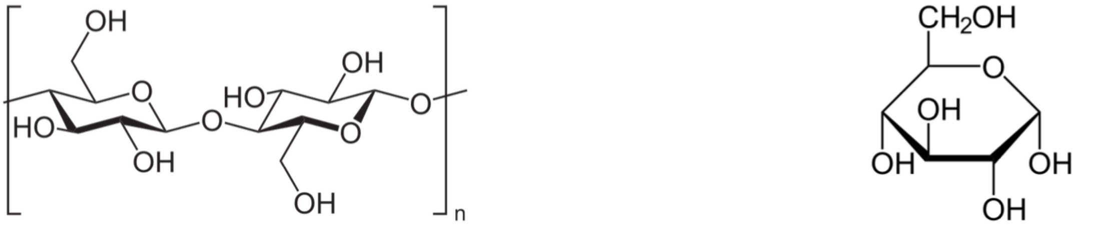
Faults can occur in a transformer in a variety of ways: electrical, thermal or mechanical, internal or external. Table 1.2 lists some of them.

Table 1.2: Causes of malfunctions of power transformers [23]

Internal Causes	External Causes
- Deterioration of insulation - Loss of winding tightening - Overheating - Solid contamination in insulating oil - Partial discharge - Design and manufacturing defects	- Overvoltages from switching or atmospheric Overload

1.4 Solid insulation of power transformers

Paper and cardboard have been utilized in electrical machines for over a century, serving as solid insulation between winding's and between winding's and the core. Derived from cellulose, extracted from wood or cotton fiber, both materials contain approximately 40 to 50% cellulose [25], Cellulose, comprising long chains of glucose rings connected by oxygen bridges, forms the structural basis of paper and cardboard. With a dielectric constant twice that of oil, paper and cardboard are strategically employed in regions of high electric field to reinforce insulation within the oil. Through a process known as oil impregnation, wherein the paper undergoes vacuum drying to reduce moisture content to 0.5%, followed by the introduction of dried and degassed insulating oil, the insulation system attains exceptional dielectric and mechanical properties. Moreover, cardboard in power transformers serves additional functions beyond electrical insulation, providing mechanical support for winding's and facilitating coolant circulation channels [26]. Hence, paper and cardboard play a crucial role in enhancing oil insulation within high electric field areas of transformers, Cellulose and glucose structures are illustrated in Figure 1.2.



(a) Cellulose Structure

(b) Glucose structure

Figure 1.2: Glucose and cellulose structures [27, 28]

1.5 Liquid insulation for power transformers

The liquid insulation of transformers is achieved through insulating liquids primarily composed of mineral oils, vegetable oils, and synthetic oils. Mineral oils account for 90-95% of the insulating liquid market, primarily due to their low cost and favorable physio-chemical properties [29].

1.5.1 Oil Role

Transformer oil, in particular, serves dual purposes as both an insulating medium and a coolant, playing a crucial role in preventing insulation breakdowns and ensuring the seamless operation of transformers within electrical power systems. The oil inside power transformers primarily serves three roles [30]:

- Electrical insulation;
- Ensuring the dissipation of heat produced by losses at the winding level towards cooling devices (external radiators);
- Slowing down the degradation of solid insulation.

1.5.2 Oil degradation

The prolonged use of oil under various stresses—electrical, thermal, and chemical—generally leads to its degradation. Aging is the phenomenon resulting in a slow and irreversible change in the material's properties [30]. The main types of aging of transformer oil are thermal aging, electro-chemical aging, and electrical aging. The change in properties significantly affects insulation performance and leads to a reduction in its lifespan.

Thermal aging corresponds to repeated or continuous long-term cyclic heating at relatively high temperatures. **Electro-chemical aging** is mainly characterized by the consequences, under the influence of the electric field, of the long-term action of certain chemicals accidentally introduced into the oil or resulting from its own degradation. **Electrical aging**, on the other hand, is mainly characterized by the long-term action of partial discharges or any ionization phenomenon due to the electric field. It is characterized by an increase in dielectric losses and a decrease in cross-resistance.

1.5.3 Categories of insulating oils

Insulating oils are classified according to their origins. Three types are distinguished: vegetable oils (or natural esters), mineral oils (based on refined petroleum products), and synthetic oils. This classification is linked to their chronological order of appearance in power transformers applications. Indeed, vegetable oils were the first to be used, followed by mineral oils, and then synthetic liquids were subsequently manufactured to address the shortcomings of mineral oil in certain applications.

The Table 1.3 below indicates the typical performance characteristics of dielectric liquids used in immersed transformers.

Table 1.3: Performance characteristics of dielectric liquids used in immersed transformers [29]

Properties	**Mineral Oil**	**Silicone Oil**	**Synthetic Esters**	**Vegetable Oils**	**Test Methods**
Dielectric Strength (kV)	30-85	35-60	45-70	82-97	IEC 60156
Relative Permittivity at $25°C$	2.1-2.5	2.6-2.9	3-3.5	3.1-3.3	IEC 60247
Viscosity at $0°C, mm^2/s$	< 76	$81 - 92$	$26 - 50$	$43 - 77$	ISO 3104
Viscosity at $40°C, mm^2/s$	$3 - 16$	$35 - 40$	$14 - 29$	$16 - 37$	
Viscosity at $100°C, mm^2/s$	$2 - 2.5$	$15 - 17$	$4 - 6$	$4 - 8$	
Pour Point ($°C$)	-30 to -60	-50 to -60	-40 to -50	-19 to -33	ISO 3016
Flash Point ($°C$)	$100 - 170$	$300 - 310$	$250 - 270$	$315 - 328$	ISO 2592 (I)
Fire Point ($°C$)	$110 - 185$	$340 - 350$	$300 - 310$	$350 - 360$	ISO 2592 (I)
Density at $20°C, kg/dm^3$	$0.83 - 0.89$	$0.96 - 1.10$	$0.90 - 1.00$	$0.87 - 0.92$	ISO 3675
Specific Gravity at $20°C$	$1.6 - 2.0$	1.5	$1.8 - 2.3$	$1.5 - 2.1$	ASTM E 1269
Thermal Conductivity ($W/m \cdot K$)	$0.11 - 0.16$	0.15	0.15	$0.16 - 0.17$	(DCS)
Coefficient of Expansion ($10^{-4} \cdot K^{-1}$)	$7 - 9$	10	$6.5 - 10$	$5.5 - 5.9$	ASTM D1903

1.5.3.1 Mineral oils

Mineral oil, a hydrocarbon-based insulating fluid derived from petroleum refining, constitutes the predominant choice for insulation in power transformers. Its widespread usage stems from com- mendable dielectric properties and efficient thermal conductivity, facilitating effective dissipation of heat generated during transformer operation. Despite its prevalent adoption, mineral oil is en- cumbered by inherent drawbacks, notably its susceptibility to combustion, which poses significant fire hazards, and its potential environmental impact due to its flammability and propensity for pollution [8].

1.5.3.2 Synthetic and silicon oils

Although mineral oils play a significant role among liquids for electro-technical applications, synthetic oils are used whenever specific properties are required [31]. This is particularly the case when it comes to improving the fire resistance of transformers located near the population (distribution and traction). These synthetic liquids are all obtained from various petrochemical processes. There are three main types intended for transformers:

- Pentaerythritol esters, also known as synthetic esters or organic esters (as opposed to natural esters or vegetable oils),
- Silicones Table 1.3,
- Chlorinated hydrocarbons (halogenated), such as PCB's (Polychlorinated biphenyl).

1.5.3.3 Vegetable oils

Although primarily used in capacitors, vegetable oils, due to the environmental enthusiasm, have been the subject of numerous studies [32] to be developed for usage in power transformer.

1.5.4 Usage

1.5.4.1 Mineral Oils

Mineral oil is the most widely used insulating liquid in electrical equipment due to its excellent dielectric and heat transfer properties, its compatibility with cellulose insulators, and its low cost (the latter can contain between 40,000 and 80,000 liters of oil!). It is primarily for this techno-economic reason that this oil is also the most commonly used in power transformers [8].

1.5.4.2 Synthetic esters

Tetraesters are used for filling "fire-resistant" distribution transformers. Their high flash point (> 300°C) compared to mineral oils is the primary characteristic of these products. This is why they are found in equipment close to the population, in buildings, tunnels, etc. However, since their use is relatively recent (about fifteen years), there is not enough data available to fully understand their long-term behavior, considering that the normal lifespan of a transformer is at least 25 years.

Primarily due to their higher cost (about four times more expensive than mineral oils), tetraesters have not been widely adopted in power transformers. The synthetic esters intended for transformers, the most well-known and commercially available of which are Midel 7131 manufactured by M&I Materials, or Envirotemp 200 manufactured by Cooper Power Systems[31].

1.5.4.3 Silicons oil or PDMS

PDMS(Polydimethylsiloxane) is primarily used for filling distribution and traction transformers, where some level of fire resistance is desired. Their use is less common than organic esters due to their very high cost (eight times more expensive than mineral oil) and the difficulty in disposing of them after use as they are not biodegradable at all. However, they are widely used for traction transformers (installed on trains) because they have good thermal stability, withstanding temperatures of 125 to 150°C.

The most well-known silicone oils are Rhodorsil 604V50 manufactured by Rhodia and 561 by Dow Corning [33].

1.5.4.4 Vegetable oil

As said earlier in Section 1.4.3.3, this type of oil is not commonly used in transformers. However, especially with the growing interest in biodegradable products in recent years, numerous studies have been conducted to replace mineral oil with vegetable oils for transformer impregnation. Examples include Biotemp oil marketed by ABB and Envirotemp FR3 marketed by Cooper Power Systems [32].

1.6 Characteristics of transformer insulating oils

The choice of an insulating oil is based on a large number of properties, grouped into three categories: electrical, physical and chemical [3], the Table 1.4 summarizes industry standards and related publications concerning standardized test procedures for transformer oil characteristics. These tests are grouped into routine tests, supplementary tests and special investigation tests [34].

Table 1.4: Transformer oil characteristic test standards [34]

Test Group	Test Name	Test Standard
Routine Tests	Color Determination	ISO 2049
	Breakdown Voltage	IEC 60156
	Water Content	IEC 60814
	Acidity (Neutralization Value)	IEC 62021
	Dielectric Loss Factor	IEC 60247
	Inhibitor Content	IEC 60666
	Oxidation	IEC 61125 method C
Additional Tests	Interfacial Tension	ISO 6295
	Particles (Number of Particles)	IEC 6097
Special Investigative Tests	Oxidation Stability	IEC 61125
	Flash Point	ISO 2719
	Compatibility	IEC 61125
	Pour Point	ISO 3016
	Density	ISO 3675
	Viscosity	ISO3104
	Polychlorinated Biphenyls (PCB)	IEC 61619
	Corrosive Sulfur	DIN 51353

1.6.1 Electrical properties

The electrical properties of liquids depend on their formation. The study of electrical properties is a multidisciplinary field, involving chemistry, electronics and mechanics [3].

1.6.1.1 Dielectric strength and breakdown voltage

Dielectric strength corresponds to the maximum electric field (breakdown voltage over the inter-electrode distance) that can be applied without a disruptive discharge. The breakdown voltage of a volume of insulating liquid is measured between two electrodes whose nature, shape and distance apart are specified in IEC 60156-1995 [35]. Dielectric strength follow this formula:

$$E_c = \frac{U_c}{e} \tag{1.1}$$

E_c is the dielectric strength (kV/mm), U_c is the breakdown voltage (kV), e is the distance between the two inter-electrode (mm).
Dielectric strength varies based on several parameters, including electrode shape and the distance between them, which ranged from a few millimeters to a few centimeters [36]. A small amount of moisture in the oils significantly reduces dielectric strength [7]. Breakdown voltage of oil improves with temperature Figure 1.3, as water solubility increases with rising temperature. Initially, it increases with the number of breakdowns, then stabilizes, and finally decreases. This is due to moisture in the oil, which dissipates through breakdowns (heating and evaporation of

water) or gas bubble elimination. Numerous breakdowns degrade the oil, potentially rendering it unusable [7].

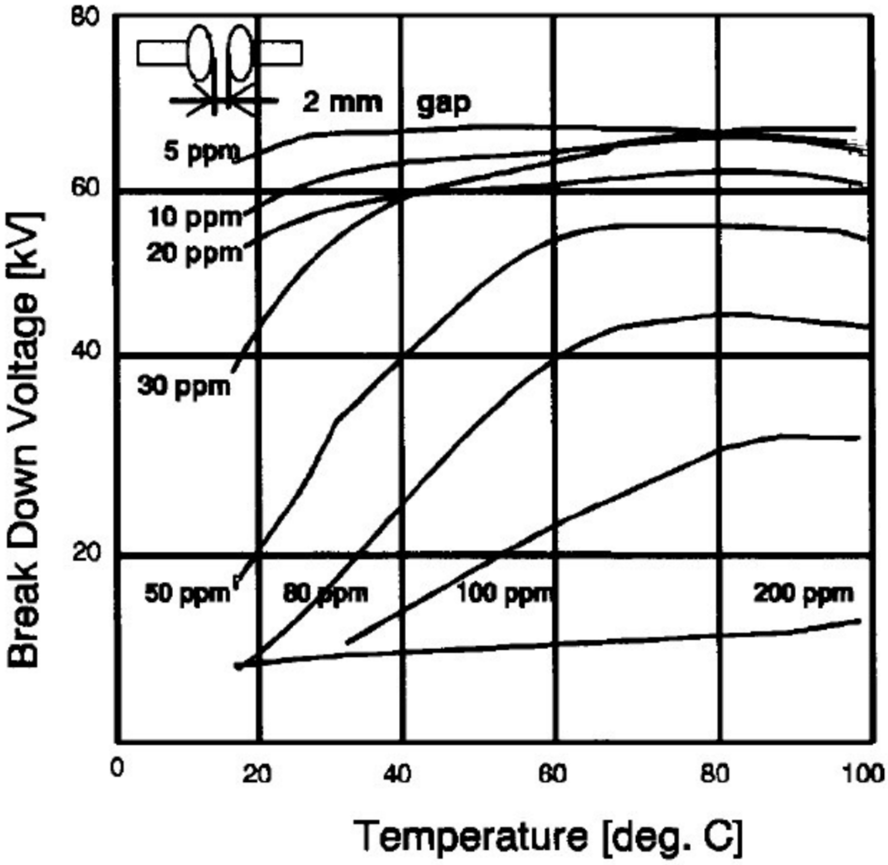


Figure 1.3: Effect of temperature on the breakdown voltages of an insulating oil with different water contents [37]

1.6.1.2 Relative Permittivity

Permittivity is the fundamental macroscopic constant that characterizes a dielectric. It expresses the possibility of polarization under the action of an electric field. It is defined by the following relationship[38].

$$\varepsilon = \varepsilon_0 \varepsilon_r \tag{1.2}$$

ε is the absolute permittivity of oil (F/m), ε_0 is the Vacuum permittivity ($\varepsilon_0 = \frac{1}{36\pi 10^9} = 8.85pF/m$) and ε_r is the oil relative permittivity (No unit).
The relative permittivity ε_r of an insulator is defined as the ration between the capacitance of a capacitor filled with this insulator (C_r) and the capacitance of the same capacitor in a vacuum (C_0) [38].

$$\varepsilon_r = \frac{C_r}{C_0} \tag{1.3}$$

1.6.1.3 Dielectric loss factor

The dielectric loss factor, or dissipation factor ($\tan\delta$), is a dimensionless value directly related to resistivity and permittivity in alternating voltage. The smaller the δ is, the smaller the dielectric losses. The higher the dielectric temperature is, the higher the dielectric losses [7].

1.6.1.4 Conductivity / Resistivity

The conductivity of an insulating liquid is due to the presence of free charges that move under the effect of an electric field, thus generating a conductive current. A good oil should have the lowest possible conductivity. In other words, it must have very high resistance. Insulating oils have a conductivity of between 10^{-11} to 10^{-13} (S/m) [7].

1.6.1.5 Partial discharges

Partial discharges are caused by the presence of gas bubbles or other impurities (moisture, particles, etc.) in the oil. Under the action of the electric field, these impurities can be the site of localized discharges. Gases that may be present in transformer oil include hydrogen (H_2) and light hydrocarbons such as methane (CH_4), as well as smaller quantities of ethane (C_2H_6), ethylene (C_2H_4) and ethylene (C_2H_4). acetylene (C_2H_2), as well as carbon monoxide (CO) and carbon dioxide[39].

1.6.1.6 Gassing effect

Gassing (mm^3/mm) is the rate of change of gas volume during 120 min of voltage application $(10kV$ at $80°C)$. If the volume increases, the oil will have positive gassing. otherwise, the oil will have negative gassing [7].

1.6.2 Physical properties

1.6.2.1 Viscosity

Viscosity is a very important factor in heat transfer. r. The more viscous the liquid, the more difficult it is to circulate it through the radiators to cool the active parts. The viscosity of hydrocarbons is correlated to their molecular weight; the lighter the product, the lower its viscosity. It is expressed by this relation [40]:

$$CST = C_t - \frac{b}{t} \tag{1.4}$$

CST is viscosity in centi-stokes, C_t the viscosity constant, t the flow time in seconds and b the viscosity coefficient.

The viscosity index is a characteristic giving the variation in viscosity of an oil as a function of temperature. The lower the variation in viscosity with temperature, the better the oil is [7].

1.6.2.2 Flow point

The use of electrical devices outdoors requires knowing the viscosity of liquid insulators at low temperatures (in winter the temperature can reach $-25°C$, and even $-60°C$).

Flow point is defined as the maximum temperature below which circulation of liquid cannot be established, therefore it is the temperature at which liquids freeze. It is determined according to the ISO 3016 standard [41].

1.6.2.3 Flash point

The proper operation of electrical equipment requires a sufficiently high flash point (min $135°C$), measured in accordance with ISO 2719 (Pensky-Martens method in isolation). The flash point is the minimum temperature for which the concentration of vapors emitted is sufficient to produce a fire blast on contact with a flame or a hot spot [42].

1.6.2.4 Fire point

This is the temperature at which steam ignites and sustains combustion for at least 5 seconds when the ignition source is removed [7].

1.6.2.5 Thermal capacity

Heat capacity (or specific heat) is defined as the quantity of heat necessary to increase the temperature of one kilogram of a substance by one Kelvin. Heat capacity increases with temperature and decreases with density. It is expressed in joules per kilogram kelvin and can be calculated by the following relationship[29].

$$C_p = \frac{1684 - 3,39T}{\sqrt{\rho_{15}}} \tag{1.5}$$

C_p is the specific heat $(J/kg.K)$, T the temperature (K) and ρ_{15} the density at $15°C$. The heat capacity of liquids at $20°C$ varies from 1000 to 2300 $J/(kg.K)$.

1.6.2.6 Thermal conductivity

Thermal conductivity (λ) is inversely proportional to density and decreases with temperature. It expresses the thermal flux flowing in steady state under the effect of a thermal gradient between two isotherms of the liquid. It is between 0.11 and 0.14 $W/(K.m)$. Thermal conductivity is a very important parameter because it determines the ability of the oil to dissipate heat [43]. Thermal conductivity decreases with increasing temperature Figure 1.4 [44].

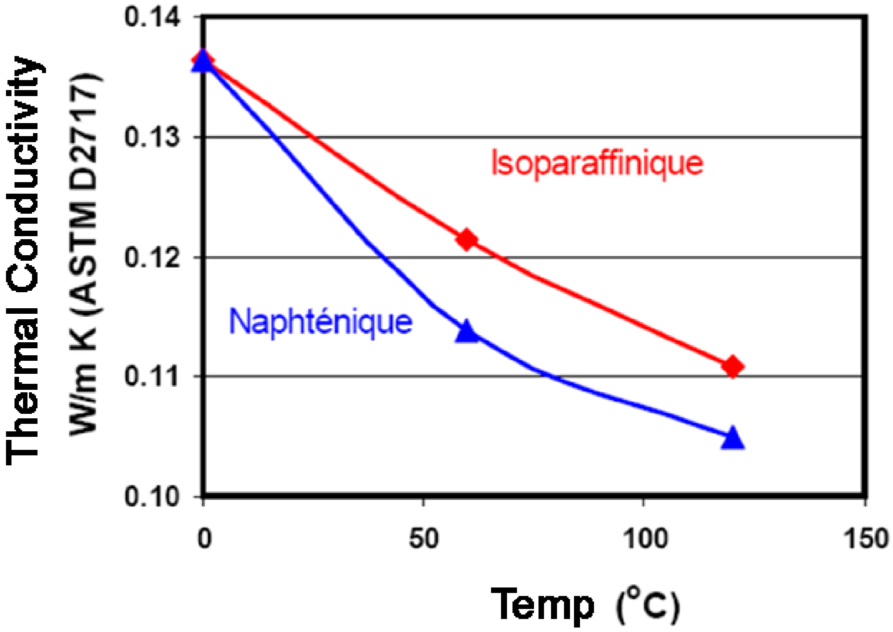


Figure 1.4: Thermal conductivity varying with temperature [45]

1.6.2.7 Color

The color of new oil is an indication of its degree of refinement. In fact, highly refined oils are white; A variation in color in deliveries of new oil may indicate a change in the origin of the oil or in the manufacture, although this is not absolutely certain. In the case of used oil, a dark color indicates its deterioration [46]. In fact, optical properties are of particularly great interest, because they can characterize the constitution of a liquid [7].

1.6.2.8 Density and coefficient of expansion

The density ρ_θ is an essential characteristic of a product. It depends on its chemical composition. The density is defined by the ratio of a mass m of liquid to its volume, measured at the temperature θ, it is expressed in kilogram per cubic meter [7].
According to the ISO 3675 standard, the density is determined at $20°C$; temperature at which it can be between 800 and 2000 kg/m^3. The best oils have a density lower than that of water 1000 kg/m^3. Thus, water does not rise to the surface of the oil in the event of an emulsion and therefore does not lead to the appearance of electric arcs. The density decreases when the temperature increases, it is expressed by the following relation [29]:

$$\rho_\theta = \rho_{20}\left(1 - \alpha v[\theta - 20]\right) \tag{1.6}$$

ρ_θ is the density of the oil at temperature θ, ρ_{20} is the density of the oil at $20°C$ (kg/m^3), θ the oil temperature in $°C$ and αv the volume expansion coefficient (K^{-1}). The volume expansion coefficient varies from 5.10^4 to 10.10^{-4} K^{-1} and it is determined according to the standard ASTM D 1903-03 [47].

1.6.2.9 Heat transfer capability

The release of heat caused by losses in the transformer requires the development of a cooling system. This consists of a circulation circuit for a fluid which must be insulating. Two types of fluid can be used, namely air and oil. This transports heat better and improves the dielectric strength of the cardboard by impregnating it [7].

1.6.3 Chemical properties

1.6.3.1 Water content

The water content of an oil must be as low as possible. When the oil is new, the water content is less than 100 ppm (1 ppm = 100 mg/kg) according to standard IEC 60814-1997 [48]. The latter defines the water content as being:

$$w(\mu g/g) = \frac{m}{M} \tag{1.7}$$

m is the quantity of water tit-rated in micro-grams μg and M is the mass of insulating liquid in grams (g).

1.6.3.2 Acidity

Acidity corresponds to the concentrations of free organic and inorganic acids present in the oil and is expressed in milligrams of potassium hydroxide (KOH) necessary to neutralize the total free acid contained in one gram of oil. Acids generally arise from oil breakdown, oxidation products, and external atmospheric sources. The presence of these acids in the oil harms the insulation system and causes corrosion of the expansion tank. For new oil, it is around $0.005(mg_{KOH}/g)$. It is measured by calorimetric titration (ISO 6618) [49], or by potentiometric titration (IEC 62021-1-2003)[50].

1.6.3.3 Oxidation

Oil oxidation causes acidity and the formation of sludge. This can be avoided by using oils with high oxidation stability minimizing sludge deposition and maximizing the life of the insulation. Oxidation stability is measured according to method C of IEC 61125 [51].

Oxidation has harmful consequences on the proper functioning of transformers because it leads, according to [7]:

- An increase in viscosity, resulting in a reduction in the ability to eliminate heat release;
- An increase in dielectric losses;
- An increase in corrosiveness towards cellulose;
- An increase in corrosiveness towards metal parts.

To reduce the effect of oxidation, antioxidant additives can be incorporated into the liquid insulation to reduce or delay its degradation [7].

1.7 Aging of insulating oils

The main agents present in transformers that can degrade insulating oils (and cellulose) are oxygen and humidity. Oxygen is present in dissolved form in the oil and moisture is present in the paper (even after drying, moisture remains, greater than 0.2% by mass). These sources of aging are all the more present in the case of power transformers, which are called "breathing" (like most in Europe), in contact with the atmosphere via a preservative and a desiccator [52]. Oxygen and humidity cause oil degradation by oxidation and hydrolysis, respectively. In addition to these two “natural” aging factors, there are two other significant catalysts, which are temperature (pyrolysis) and the electric field. Finally, the presence of the construction materials of the device, such as iron, copper or cellulose, also act as aging catalysts [53].

1.7.1 Oxidation of mineral oil

The oxidation of mineral oil results mainly from the simultaneous action of molecular oxygen and temperature in the presence of the materials of construction of the devices (copper, iron, cellulose, etc.). The action of humidity and the electric field is still poorly understood on the oxidation phenomenon [53].

Oxidation constitutes the predominant factor in the aging of liquid insulation, and more particularly in that of mineral oils. This refers to a set of complex and slow chemical reactions during which hydrocarbons react with dissolved oxygen [54]. Oils, like all natural products, oxidize giving degradation products which are first characterized by a change in color and odor, linked to the formation of light acidic compounds, then by an appearance of deposits and aggressive residues, linked to heavy acids, and accompanied, in general, by an increase in viscosity [53].

It is generally accepted that oxidation proceeds according to a mechanism of chain reactions initiated by free radicals (molecules or fragments of molecules in an excited electronic state which have a free valence, and which are chemically very reactive and electrically neutral). Such reactions are called auto-catalytic, that is to say that the products formed by oxidation serve as catalysts [53].

The oxidation of a mineral oil leads to the formation of an ester and water. Initially, soluble decomposition products (related to naphthenic and paraffinic compounds) are formed, such as aldehydes, ketones and carboxylic acids. Then, through polymerization, insoluble deposits (linked to aromatic compounds) are formed, also called sludge, which increase the viscosity of the oil and can block the cooling circuits [29].

The oxidation of mineral oils increases with temperature and oxygen concentration. The reaction rate doubles approximately every 8 to $10\,°C$, starting at $60\,°C$ [55].

Nikolay Markovich Emanuel [56] showed that the rate of progress of the reaction follows a law of the type:

$$\omega = K[R^0][O_2] \tag{1.8}$$

ω is the rate of oxygen consumption, K the reaction constant, $[R^0]$ the concentration of hydrocarbon radicals, $[O_2]$ the concentration of oxygen in the liquid phase.

1.7.2 Humidity

Moisture is the “number one enemy” of electrical insulators, in the sense that unlike them, it conducts electricity. This phenomenon is all the more pronounced in insulating oils because the latter have low water solubility. We often speak of the phenomenon of hydrolysis to define the influence of water on the aging of oil [57].
Before filling the transformer, the oil is dried to achieve a low moisture content (or water content). From this moment on, this moisture content will continually increase for different reasons. The two main sources of increased water content in oil are the penetration of moisture from the atmosphere and the degradation of cellulose. Moisture ingress from the atmosphere comes from the fact that power transformers are often breathable. The degradation of cellulose provides water because although the latter is dried, there always remains at least 0.2% by mass of humidity within it [52].
Water can be found in oil in dissolved form, in dispersed form (water droplets) or in emulsion (appearance of two immiscible phases). The solubility W_s of water in oil is given in mg/kg or ppm (parts per million). It depends on the condition of the oil, the temperature and the type of oil. The variation of the solubility of water in oil is expressed by [58]:

$$W_s = W_{oil} e^{\left(-\frac{B}{T}\right)} \tag{1.9}$$

W_s is the solubility of water in oil (mg/kg or ppm), T is the temperature of the oil (K), W_{oil} and B are constants specific to the liquid.
The higher the temperature, the greater the solubility of water in the oil. This variation in solubility with temperature can prove problematic. Indeed, when the device is hot, water can dissolve in greater quantities, and when the device cools, the solubility of the water decreasing again can allow free water to appear [58]. Thus, it is always appropriate to determine the solubility of the oil at the same temperature at which the oil was taken. The standardization committee for electrotechnical fluids has established a correction factor which makes it possible to reduce the value of the water content of a mineral oil taken at a certain temperature ($> 20\,°C$) to the temperature of $20\,°C$ to have a criterion of comparison [52].

On contact with a humid atmosphere, water gradually dissolves in the oil. This is why so-called “breathing” transformers are always equipped with a desiccator to dry the air. The water content of an oil is often expressed as a percentage of saturation at a given temperature, also called relative humidity [53].

1.7.3 Partial discharge

under the effect of electrical field and local inhomogeneities (gas, humidity, particles), micro-discharges can appear which decompose the oil by generating gases. In the long term, these discharges can become harmful if they tend to attach to the impregnated solid insulators and reach a sufficient level to deteriorate them [53].

1.7.4 Temperature

As we saw previously, temperature acts as a catalyst for oxidation. It is also a factor which gradually degrades insulating oils. We often speak of thermal stability to define the behavior of the product at high temperature, ignoring any other constraints such as oxidation, hydrolysis or partial discharges. The effect of temperature on the aging of an oil is often defined as pyrolysis. Under normal conditions of use of transformers (temperature between 80 and $100\,°C$), the stability of existing insulating liquids is largely sufficient. Temperature therefore mainly acts as a catalyst and not as a trigger. However, there may be hot spots in certain parts of the transformer that degrade the oil [53].

1.8 Diagnosis of transformer oil by dissolved gas analysis

Power transformers are extremely expensive devices. Evaluating the condition of a device based on analyzes carried out on the oil it contains is much less expensive compared to the cost of interrupting the supply of electrical energy following a transformer breakdown and the replacement of the latter. In-service monitoring of large transformers is therefore systematic. This monitoring also exists on small transformers, but on a more punctual basis, in particular to try to explain certain malfunctions [53].

1.8.1 Insulator oil decomposition

The decomposition of transformer oil also results in the formation of solid compounds and the formation of gases. These gases are usually generated by typical anomalies such as aging of insulation, the formation of electric arcs, partial discharges, locally high temperatures or, more generally, imperfect cooling [9].

These gases, soluble in oil as indicated, are an important indicator of faults arising in a transformer. Constant monitoring and evaluation of gas formation makes it possible to react promptly, avoid greater damage and significantly increase the lifespan of transformers [9]. Dissolved gas analysis, which involves looking for gases in the oil, is one of the most reliable power transformer diagnostic methods used [2].

1.8.2 Dissolved gaz analysis

Dissolved Gaz Analysis (DGA) introduced more than forty years ago, is one of the most used techniques for transformer diagnosis thanks to the fact that it is non-destructive and can be used for real-time monitoring. To perform an analysis, this technique requires only a small quantity of insulating oil and does not require transformer service interruption. The main gases produced are methane (CH_4), ethane (C_2H_6), ethylene (C_2H_4), acetylene (C_2H_2), hydrogen (H_2), monoxide (CO) and carbon dioxide (CO_2)) [14].

1.8.3 Principle of detection and measurement of dissolved gas

To extract and quantify these gases, Gas Chromatography (GC) is applied. GC is a method of separating gaseous compounds or compounds capable of being vaporized by heating without decomposition. This method is carried out using the chromatograph. The results obtained by the latter indicate the proportions of the concentrations of gases dissolved in the oil in ppm [2]. The chromatography operation is shown on Figure 1.5, a representative oil sample is extracted and introduced into the DGA system. There, the dissolved gases trapped within the oil are separated through a gas extraction process. These separated gases then travel through a gas chromatography column where they are divided based on their unique properties. Finally, a detector identifies and measures each gas component, allowing the analyze of the concentrations and diagnose potential transformer faults [59].

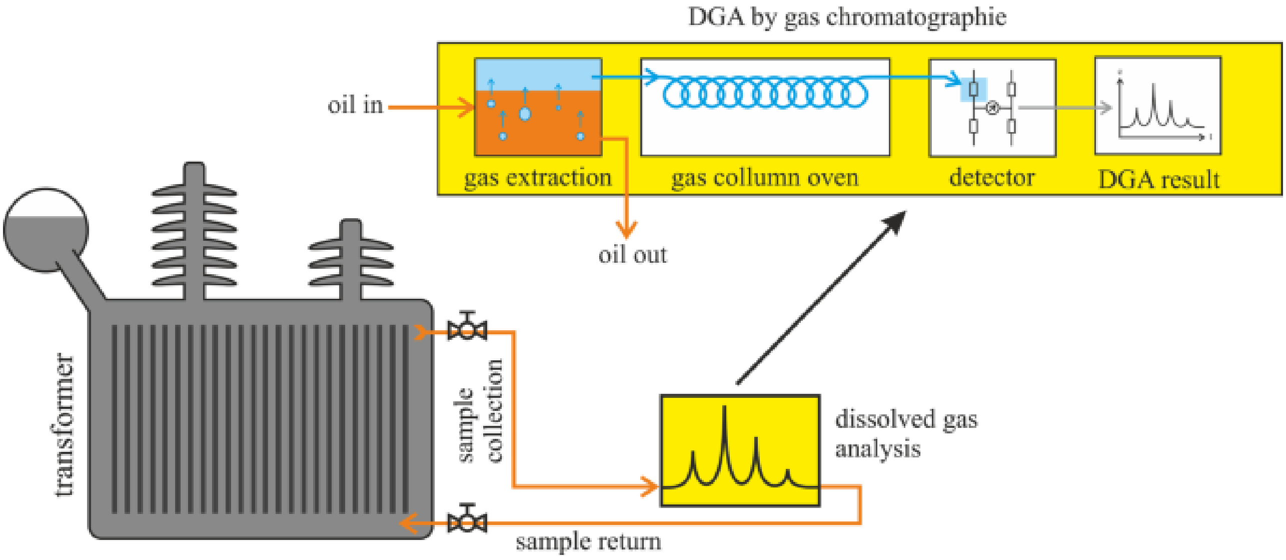


Figure 1.5: The use of DGA on power transformer oil [59]

1.8.4 Type of power transformer faults

Standards IEC 60599-2008 [8] and IEEE standard C57.104-2019 [9] classify transformer faults detectable by dissolved gas analysis into two categories: electrical fault and thermal fault. These two main categories can be further classified into six types of transformer faults, based on the magnitudes of the fault energy [60]:

- Partial discharges (PD);
- Low energy discharges (D1);
- High energy discharges (D2);
- Thermal faults with temperatures below 300 $°C$ (T1);
- Thermal faults with temperatures between 300 $°C$ and 700 $°C$ (T2);
- Thermal faults with temperatures above 700 $°C$ (T3).

1.8.4.1 Partial discharges

A partial discharge is a very localized, low-intensity electrical discharge that occurs between two separate conductors. Partial discharges appear in the form of short-duration pulses which are often accompanied by the emission of sound, light, heat and chemical reactions. Sources of partial discharges include voids and cracks in solid insulation, floating components such as water drops and air bubbles, and corona caused by sharp edges and corners of insulation solid, winding's or tank [61].

Partial discharge is the type of discharge that only partially fills the insulation gap between the electrodes. The discharge can occur completely inside the transformer insulation or near

the conductors. The partial discharge around an electrode in a gaseous medium is called cold plasma discharge or "corona effect", while others such as that occurring in a transformer liquid is commonly called "streamer" [62].
Partial discharges, known as one of the most determining reasons for the degradation of insulators, which could lead to breakdown when they accumulate and propagate completely between two electrodes. To avoid costly transformer failures, it is extremely important to monitor partial discharge activities for early detection of fault inception in transformers [60]. Usually, this type of fault is characterized by the production of hydrogen and methane [62].

1.8.4.2 Arc or spark discharges

After decades of study, it is now generally accepted that breakdown occurs after the streamers fully propagate through the electrode gap. When the energy of dielectric breakdown is limited, it acts as small arcs called “sparking faults”. Compared to partial discharge faults, sparking faults generate much more gas during the fault which can be critical for the operation of the transformer [60].
Arc discharges generate very high temperatures (above $5000\,°C$) and a large amount of gas, mainly acetylene and hydrogen. This type of fault is very dangerous and, if left unchecked, can cause excessive pressure in the transformer tank, even leading to an explosion [61].

1.8.4.3 Thermal faults

Thermal faults result from overheated conductors, short circuits, overheated winding’s due to eddy currents, loose connections and insufficient cooling. Localized overheating is known as hot spots. The temperature of a hot spot on a metal surface can reach $1500\,°C$, causing local heating of the surrounding oil, leading to the generation of hydrocarbon gases [61]. Different types of fault gas will be formed in different temperature ranges. Therefore, fault gases could be used to diagnose the transformer fault temperature [60].

1.9 Conclusion

It emerges from this first chapter that the choice of a liquid insulator is not based solely on these electrical, physical or chemical properties, but also depends on its resistance to aging and its capacity to evacuate heat. It appears that the main factors responsible for the aging of an insulating oil are oxygen, temperature and humidity. This humidity comes mainly from the atmosphere and cellulose. The combined action of temperature and oxygen causes the insulating oil to oxidize. These overriding factors may over time lead to the appearance of various defects.

Chapter 2

Conventional diagnostic methods

2.1 Introduction

The oil used in power transformers is regarded as the memory of events that have occurred on this electrical equipment. Such events are detected by analyzing and interpreting dissolved gases, based on their concentrations and proportions, in the sampled oil. In fact, any abnormal operation of the transformer generates dissolved gases in the oil.

Dissolved gas analysis (DGA) is an important measure in transformer life-cycle management. Correct interpretation of dissolved gas analysis can provide valuable information on the health of transformers. In fact, it can be used to identify and interpret faults (thermal and electrical) and the source of their creation, with a view to preventing malfunctions at an early stage.

This chapter is devoted to the presentation of the most popular conventional methods, which we divide into three main categories.

- The first category uses dissolved gas concentrations on an individual basis, assigning each gas one or more defects. Gas concentrations can be expressed in ppm for the Individual gases method [5] IEEE Standard C57.104-2019 method, also known as the Total Dissolved Combustible Gas (TDCG) [9], or in percentages for the Key Gas Method (KGM) [5, 7].
- The second category uses concentration ratios. These include the Dornenburg method, the Rogers method and IEC 60599-1978 [11, 12][63].
- The third category is based on graphical representations, such as Duval triangle (with its seven versions) [13] and the pentagon (with its two variants I and II) [14].

2.1.1 Methods that uses gas concentration

2.1.1.1 Individual gases method

Generally speaking, several types of combustible and non-combustible gases can be extracted from oil. The main combustible gases are hydrogen (H_2), methane (CH_4), ethane (C_2H_6), ethylene (C_2H_4), acetylene (C_2H_2) and carbon monoxide (CO). The main non-combustible gases are carbon dioxide ($CO2$), Nitrogen (N_2) and oxygen (O_2). Table 2.1 shows the main combustible and non-combustible gases and their corresponding defects. From this table, each type of gas is generally linked to a single defect, with the exception of ethylene (C_2H_4) and ethane (C_2H_6), which are linked to two separate defects [5].

Table 2.1: Main gases and associated faults [5]

Gas	Associated Faults
Acetylene ($\mathrm{C_2H_2}$)	High-energy electrical fault (electrical arcs, sparks)
Ethylene ($\mathrm{C_2H_4}$)	Thermal fault characterized by local overheating; Thermal decomposition of oil
Ethane ($\mathrm{C_2H_6}$)	Low-energy thermal fault; Thermal decomposition of oil
Hydrogen ($\mathrm{H_2}$)	Low-energy electrical fault (low discharges, corona effect)
Methane ($\mathrm{CH_4}$)	Partial discharges (arcs) or thermal decomposition of oil
Carbon monoxide (CO)	Thermal decomposition (due to overheating) of cellulose (paper)
Carbon dioxide ($\mathrm{CO_2}$)	Thermal decomposition (due to overheating) of cellulose (paper)
Nitrogen ($\mathrm{N_2}$)	Sample defect
Oxygen ($\mathrm{O_2}$)	Leakage fault

2.1.1.2 IEEE Standard C57.104 or TDGC method

TDGC (Total dissolved gaz concentration) is based on the use of concentrations, in ppm, of all combustible gases. These consist of the gases resulting from the decomposition of the oil, namely hydrogen (H_2), methane (CH_4), acetylene (C_2H_2), ethylene (C_2H_4) and ethane (C_2H_6), and those due to the decomposition of the insulation paper, consisting of carbon monoxide (CO) and carbon dioxide (CO_2) [9].

For an easy and correct application of this method, it is necessary to have the previous history of dissolved gases determining the normal operating state of the transformer. Consequently, it can be difficult to determine whether a transformer is operating normally in the absence of previous dissolved gas history. In these circumstances, four conditions (levels) have been established to classify the risks associated with transformers and are presented in Table 2.2.

Table 2.2: Conditions for dissolved gas concentration according to IEEE standard C57.104-2019 [9]

State	Key Gas Concentration Limits [$\mu g/L(ppm)$]							
	$\mathbf{H_2}$	$\mathbf{CH_4}$	C_2H_2	C_2H_4	C_2H_6	**CO**	$\mathbf{CO_2}$	TDGC
Condition 1	100	120	1	50	65	350	2500	720
Condition 2	$101-700$	$121-400$	$2-9$	$51-100$	$66-100$	$351-570$	$2500-4000$	$721-1920$
Condition 3	700– 1800	401– 1000	10– 35	$101-200$	$101-150$	571– 1400	4001– 10000	1921– 4630
Condition 4	> 1800	> 1000	> 35	> 200	> 150	> 1400	> 10000	> 4630

The conditions interpretation is as follows [9]:

1. The transformer is in good working order as long as the TDCG is below the level indicated in Table 2.2. However, any gas whose concentration exceeds the values indicated requires special attention.
2. A TDGC value within the range implies that the dissolved gas value is higher than normal. This may result in a fault. Thus, any combustible gas exceeding the specific level requires further investigation.
3. A TDGC value within the range indicates a high level of insulation decomposition (paper and/or oil). This situation may give rise to a strong possibility of a fault. Indeed, any combustible gas exceeding the specific level requires daily analysis.
4. A TDGC exceeding the required level indicates excessive decomposition of the insulation (paper and/or oil). Continued operation of the transformer could lead to transformer failure. The transformer must be shut down immediately.

2.1.1.3 Key gases method

The KGM uses different concentrations of key gases in percentages to detect defects. This method can be seen as a modification of the TDGC [64]. KGM determines defect types from the typical or predominant gases released into the oil. In fact, the key gases and their quantities are closely related to the nature of the defect.

Table 2.3 summarizes the main gases and defects generated when applying KGM. The defects considered are those listed in IEC 60599-1999 [65]. The four main faults are as follows:

- Thermal fault due to oil overheating;
- Thermal fault due to cellulose overheating;
- Electrical fault due to corona or partial discharge;
- Electrical fault due to arcing.

Table 2.3: Diagnostic criteria by KGM [66]

Key Gas	Faults	Detected Gases	Percentage of Dominant Gases
C_2H_4	Oil Overheating	In addition to ethylene and methane, decomposition products also include small amounts of hydrogen and ethane. Moreover, traces of acetylene may be detected during severe faults or those involving electrical contacts.	C_2H_4 : 63% C_2H_6 : 20%
CO	Cellulose Overheating	Cellulose overheating generates large quantities of carbon monoxide and dioxide. If the fault involves an oil-impregnated structure, methane and ethylene will also be formed.	CO : 92%
H_2	Partial Discharges (Corona Effect)	Low-energy electrical discharges develop hydrogen and methane, and small amounts of ethane and ethylene. Additionally, discharges in cellulose release comparable amounts of carbon monoxide and dioxide.	H_2 : 85% CH_4 : 13%
H_2 and C_2H_2	Electrical Arcs	Electrical arcs cause the formation of significant amounts of hydrogen and acetylene, and small amounts of methane and ethylene. If electrical arcs touch cellulose, there may also be formation of CO and CO_2.	H_2 : 60% C_2H_2 : 30%

Note that single gas methods seem straightforward. However, they are characterized by a low fault classification rate. Indeed, they are not considered reliable diagnostic tools for power transformers.

2.1.2 Methods that uses gas concentration ratio's

Gas concentration ratios were first used by Dornenburg, then confirmed by Rogers and the European community in particular, and the scientific community in general. Such methods use six main ratios (R1 to R6) of key gases in power transformer oil fault diagnosis [5, 7]. These ratios consist of :

$$R_1 = \frac{CH_4}{H_2} \quad R_2 = \frac{C_2H_2}{C_2H_4} \quad R_3 = \frac{C_2H_2}{CH_4} \quad R_4 = \frac{C_2H_6}{C_2H_2} \quad R_5 = \frac{C_2H_4}{C_2H_6} \quad R_6 = \frac{C_2H_6}{CH_6}$$

2.1.2.1 Dornenburg method

Dornenburg ratio method (DRM) This method is one of the first techniques introduced for diagnosing power transformer oils, with a view to interpreting the results of dissolved gas analysis (DGA). This method can only be valid if the concentration of each gas is above the permissible limit (L1) reported in Table 2.4 [11].

Table 2.4: Dissolved gas concentration (L1) for DRM [9]

Gases	Dornenberg(L1) (ppm)
H_2	100
CH_4	120
CO	350
C_2H_2	35
C_2H_4	50
C_2H_6	65

Once dissolved gas levels are sufficient, the ratios R1 to R4 are calculated. By comparing the gas concentration values with those of the admissible limits (L1), the fault, if any, can be detected. In fact, three types of defects are considered, namely:

- Thermal decomposition;
- Low-energy partial discharges or corona;
- High-energy electric arcs.

These fault types are determined according to the range of variation of each ratio (R1 to R4), as shown in table Table 2.5.

Table 2.5: Dornenberg method suggested fault diagnosis [11]

Suggested Fault Diagnosis	Ratio 1	Ratio 2	Ratio 3	Ratio 4
Thermal Decomposition	> 1	< 0.75	< 0.3	> 0.4
Partial Discharge (Low-intensity PD)	< 0.1	Not significant	< 0.3	> 0.4
Arcing (High-intensity PD)	0.1 to 1	> 0.75	> 0.3	< 0.4

Based on IEEE Standard C57.104-2019 [9], the Dornenburg ratio method can be applied, for any oil sample, according to the flow chart in Figure 2.1.

The Huge drawback of the DRM diagnosis is that it cannot be conducted if the key gas concentration of $H_2, CO, CH_4, C_2H_2, C_2H_4$, and C_2H_6 exceeds the relevant L1, check Table 2.4 [9]. Also, it does not cover all possible faults in power transformers, as it considers only three faults: thermal decomposition, low-energy partial discharges or corona, and high-energy electric arcs. As a result, its interpretation is very limited and inadequate.

2.1.2.2 Rogers method

With a view to improving on the Dornenburg method, Rogers ratio method (RRM) initially developed an investigation based on dissolved gas analysis using four ratios R_1, R_2, R_5 and R_6, before deleting the R_6 ratio, which he felt was not sufficiently useful in identifying defects [12].

The methodology and procedure of Rogers' methods are similar to those of Dornenburg. However, Rogers' methods can be applied even when gas concentrations (in ppm) do not exceed the threshold limit values reported in Table 2.4 for the dornenburg method.

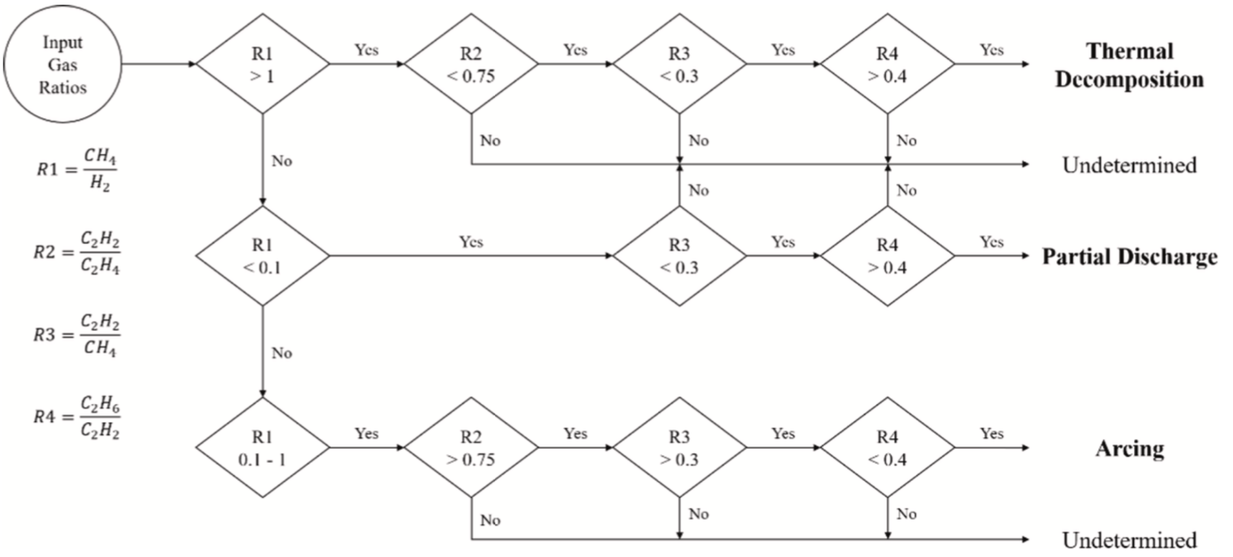


Figure 2.1: Flowchart for DRM fault classification [10]

Rogers proposed and exploited the three ratios to generate three codes, namely 0, 1 and 2. Each code corresponds to a range of variation of each ratio. The ranges of the three ratios and their codes are shown in Table 2.6.

Table 2.6: Ranges of the three ratios and their codes according to the Rogers method [12]

Gas Ratio	Range	Code
$R_1 = \frac{CH_4}{H_2}$	$R_1 < 0.1$	1
	$0.1 \leq R_1 \leq 1$	0
	$R_1 > 1$	2
$R_2 = \frac{C_2H_2}{C_2H_4}$	$R_2 < 0.1$	0
	$0.1 \leq R_2 \leq 3$	1
	$R_2 > 3$	2
$R_5 = \frac{C_2H_4}{C_2H_6}$	$R_5 < 1$	0
	$1 \leq R_5 \leq 3$	1
	$R_5 > 3$	2

The combination of codes from the three reports can be linked to an interpretation giving a specific fault diagnosis, as shown in Table 2.7. The latter also shows the six defects considered by Rogers' method [12].

Table 2.7: Codes and faults following RRM [12]

Case	Code			Type of Fault
	$\mathbf{R_1}$	$\mathbf{R_2}$	$\mathbf{R_5}$	
1	0	0	0	No Fault
2	1	0	0	Partial Discharges with Low Energy
3	1	1	0	Partial Discharges with High Energy
4	0	$1 \sim 2$	$1 \sim 2$	Electrical Discharge with Low Energy
5	0	1	2	Electrical Discharge with High Energy
6	0	0	1	Overheating with temperature $T < 150°C$
7	2	0	0	Overheating with temperature $150°C < T < 300°C$
8	2	0	1	Overheating with temperature $300°C < T < 700°C$
9	2	0	2	Overheating with temperature $T > 700°C$

Table 2.7 clearly shows the disadvantage of Rogers' three-ratio method. In fact, only 9 cases can be interpreted out of all possible combinations. The others, such as (1,1,1) and (1,0,0) of the ratios R_1, R_2 and R_5, cannot be interpreted at all. Furthermore, similar to the Dornenburg method, Rogers' ratios may be insignificant, leading to misinterpretation. This limits the use of this method.

The flow chart illustrating the application of Rogers' three-ratio method to analyze and identify defects in an oil sample is shown in Figure 2.2.

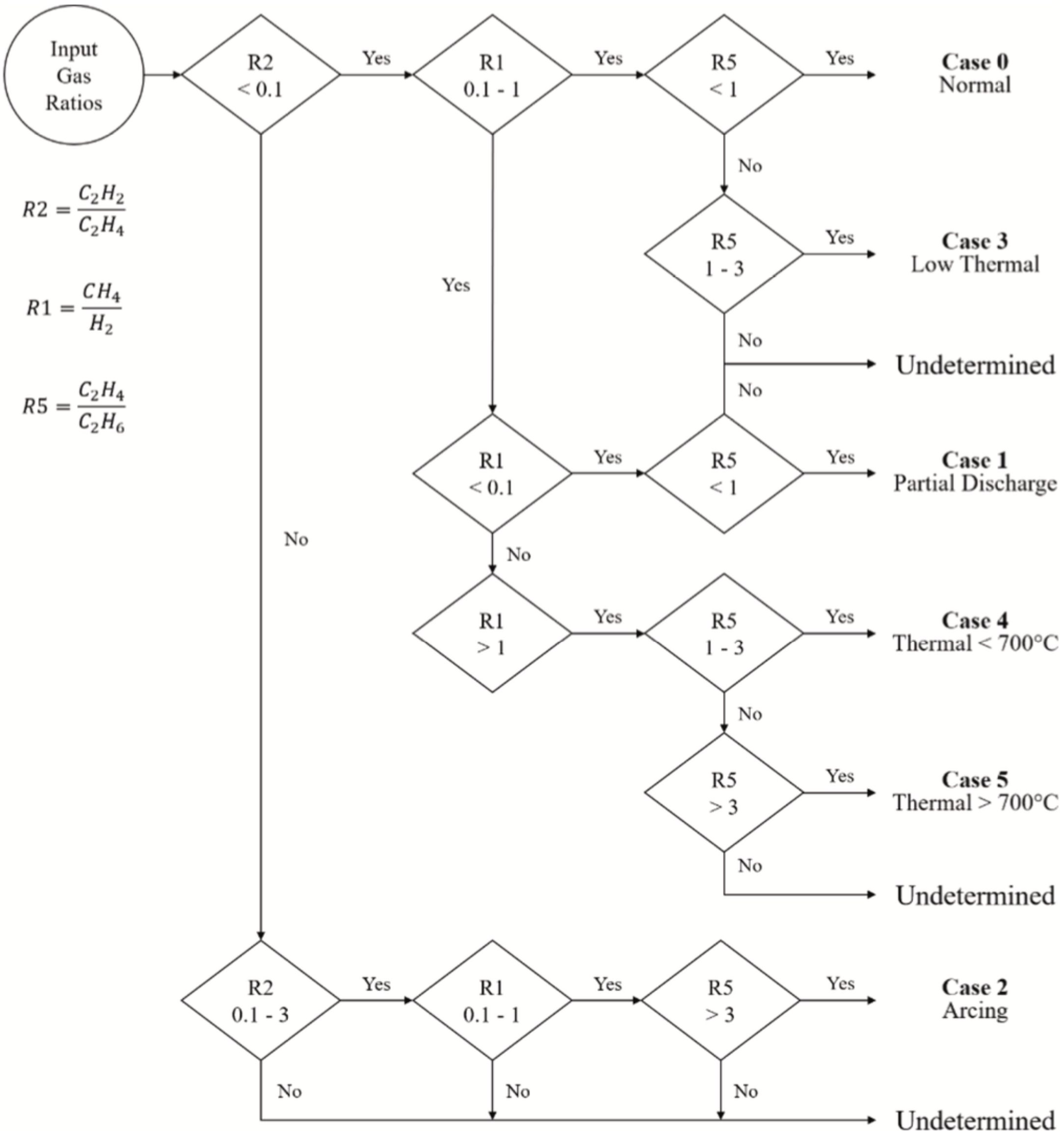


Figure 2.2: Roger ratio method flow chart [10]

2.1.2.3 IEC 60599 Method

The IEC 60599 method is used as a guide for interpreting the analysis of dissolved gases in the oil of transformers in service. In order to proceed with the diagnosis, it is necessary to ensure that the measured concentrations of dissolved gases are significant, in order to be able to interpret the results. Low concentrations can lead to misinterpretation [8, 63].

The IEC 60599 method uses Roger's three ratios (R_1, R_2 and R_5) and codes (0, 1 and 2). In addition, it has standardized its six faults as follows: partial discharge (PD), low-energy electrical discharge (D1), high-energy electrical discharge (D2), thermal fault T<300°C (T1), thermal fault 300°C< T<700°C (T2)and thermal fault T>700°C (T3) [8, 63].

The interpretation of the different defect types, after dissolved gas analysis, is presented in Table 2.8.

Table 2.8: Interpretation of DGA by IEC 60599 Method [8]

Case	**Characteristic Defect**	R_1	R_2	R_5
PD	Partial discharges	NS[(1)]	< 0.1	< 0.2
D1	Low-energy discharges	> 1	$0.1 - 0.5$	> 1
D2	High-energy discharges	$0.6 - 2.5$	$0.1 - 1$	> 2
T1	Thermal defect (T < 300°C)	NS[(1)]	> 0.1	< 1
T2	Thermal defect (300°C < T < 700°C)	< 0.1	> 1	$1 - 4$
T3	Thermal defect (T > 700°C)	0.2[(2)]	> 1	> 4

[(1)] **NS**= Not Significant
[(2)] an increasing C_2H_2 value may indicate that the hot spot is at a temperature above 1000°C

As with Rogers' method, there are only 9 interpretable cases among all possible combinations. This proves that this method and the ratio's method in general are limited. Check the Figure 2.3 for the IEC method flow chart.

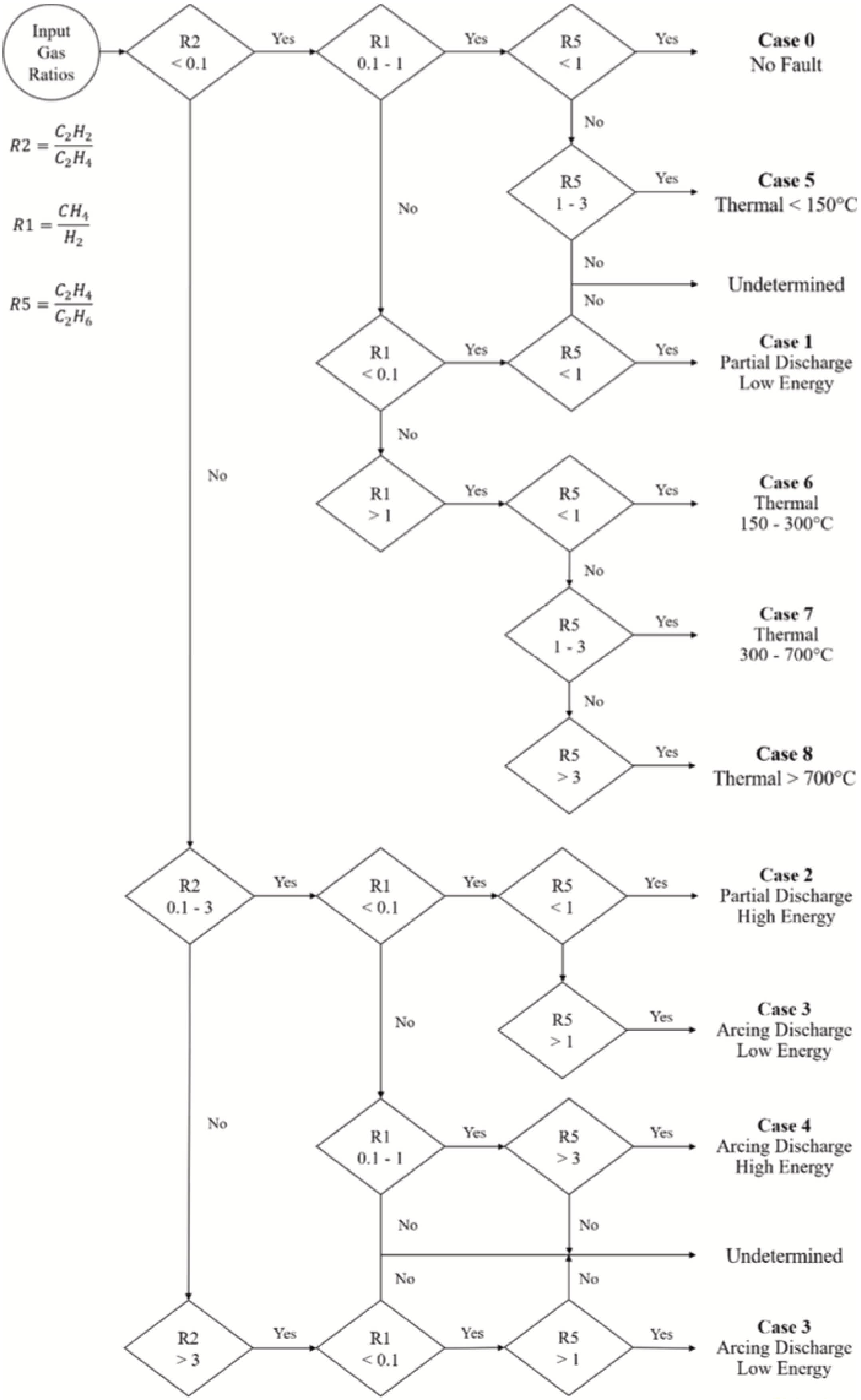

Input Gas Ratios
$R2 = \frac{C_2H_2}{C_2H_4}$
$R1 = \frac{CH_4}{H_2}$
$R5 = \frac{C_2H_4}{C_2H_6}$
R2 < 0.1
R1 0.1 - 1
R5 < 1
Case 0 No Fault
R5 1 - 3
Case 5 Thermal < 150°C
Undetermined
R1 < 0.1
R5 < 1
Case 1 Partial Discharge Low Energy
R1 > 1
R5 < 1
Case 6 Thermal 150 - 300°C
R5 1 - 3
Case 7 Thermal 300 - 700°C
R5 > 3
Case 8 Thermal > 700°C
R2 0.1 - 3
R1 < 0.1
R5 < 1
Case 2 Partial Discharge High Energy
R5 > 1
Case 3 Arcing Discharge Low Energy
R1 0.1 - 1
R5 > 3
Case 4 Arcing Discharge High Energy
Undetermined
R2 > 3
R1 < 0.1
R5 > 1
Case 3 Arcing Discharge Low Energy
Yes
No

Figure 2.3: IEC method flow chart [10]

2.2 Graphical methods

Instead of classifying the fault based on the type of gasses produced directly, Offering a quantitative approach, Graphical methods involve visual interpretation of gas concentration patterns for identifying fault conditions use the gazes ratio's, Each method utilizes a distinct mapping technique to represent fault types within the graphical space, providing a qualitative analysis. It was first introduced by Duval with his triangle's [13]. Duval also developed a pentagon representation [14].

2.2.1 Duval triangle's

M. Duval developed a triangular (graphical) representation based on the use of three gases. Seven versions of the triangle were developed by M. Duval, each version applied to a well-defined oil type or case [13, 14]. Table 2.9 illustrates the combinations of gases used in each triangle version.

Table 2.9: Gases used in each Duval triangle [13, 67]

	H_2	CH_4	C_2H_2	C_2H_4	C_2H_6
Triangle 1		✓	✓	✓	
Triangle 2		✓	✓	✓	
Triangle 3		✓	✓	✓	
Triangle 4	✓	✓		✓	
Triangle 5		✓	✓	✓	
Triangle 6	✓	✓	✓		
Triangle 7		✓		✓	✓

Triangles 3, 6 and 7 have been developed for vegetable oils and silicone oils. In the following, In order not to make the manuscript too long. we only present the versions developed for mineral oils. i.e. 1, 2, 4 and 5.

For a given sample and Duval triangles, the calculation steps, detailed in [13], can be summarized as follows:

1. Calculation of gas percentages.
2. Plot each gas percentage on its respective axis.
3. Calculation of the coordinates of the percentages of the three gases.
4. Calculate the area of the triangle created.
5. Calculate the coordinates of the new triangle center of mass.
6. Determine the defect by locating this center of mass in the defect zone.

2.2.1.1 Duval triangle 1 :

it was developed in 1974 for diagnosing transformers and cables with mineral oil insulation. Duval's Triangle 1 uses CH_4, C_2H_2 and C_2H_4 gases, and the faults detected are: DP, D1, D2, T1, T2, T3 and DT.

DT: mixed electrical and thermal faults.

The Duval triangle I representation is illustrated in Figure 2.4.

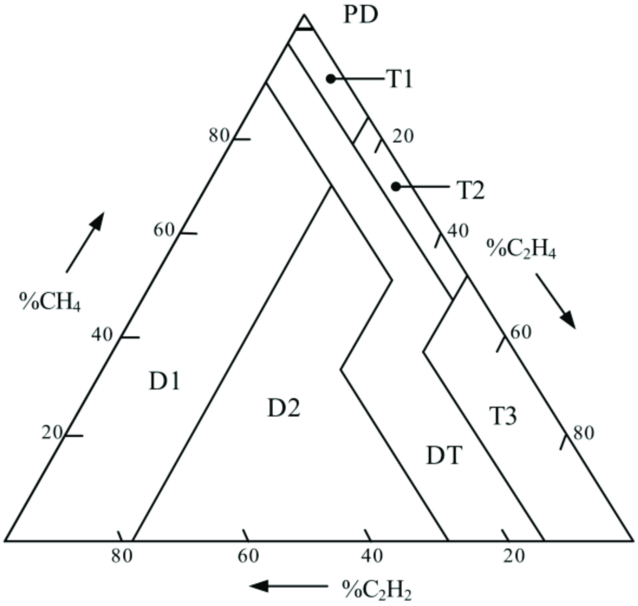


Figure 2.4: Duval triangle 1 [13]

2.2.1.2 Duval triangle 2 :

This version applicable in the case of tap changers (auto-transformers) filled with mineral oil. The gases used are CH_4, C_2H_2 and C_2H_4. The six zones that define this triangle are : D1, T2, T3, N (normal operation), X1 (T3 or T2 fault in progress, or D2 severe abnormal arc) and X3 (D1 abnormal arc or thermal fault in progress).

The Duval triangle I representation is illustrated in Figure 2.5.

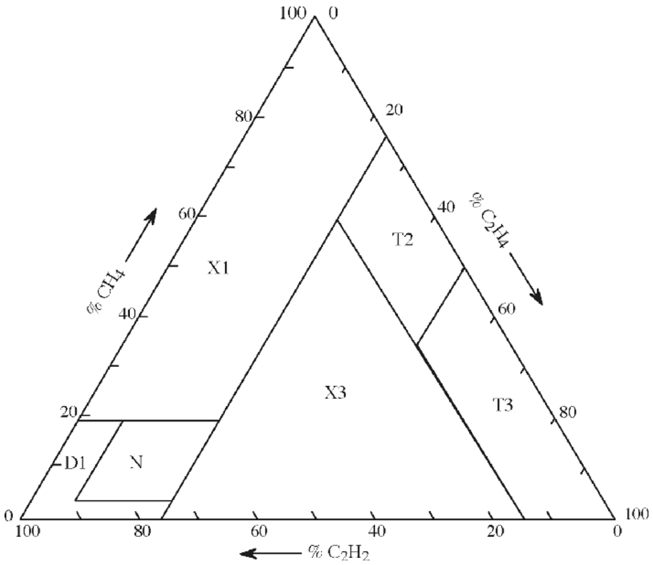


Figure 2.5: Duval triangle 2 [13]

2.2.1.3 Duval triangle 4

This version has been developed specifically for low-energy or temperature defects (DP, T1 and T2) using H_2, C_2H_4 and C_2H_6 gazes. In order to obtain more information on these defects in mineral oils, it is characterized by five zones : DP, S (stray gassing), C (hot spot with paper carbonization at T > 300°C), O (overheating T < 250°C) and the N/D zone (not determined).

The Duval triangle I representation is illustrated in Figure 2.6.

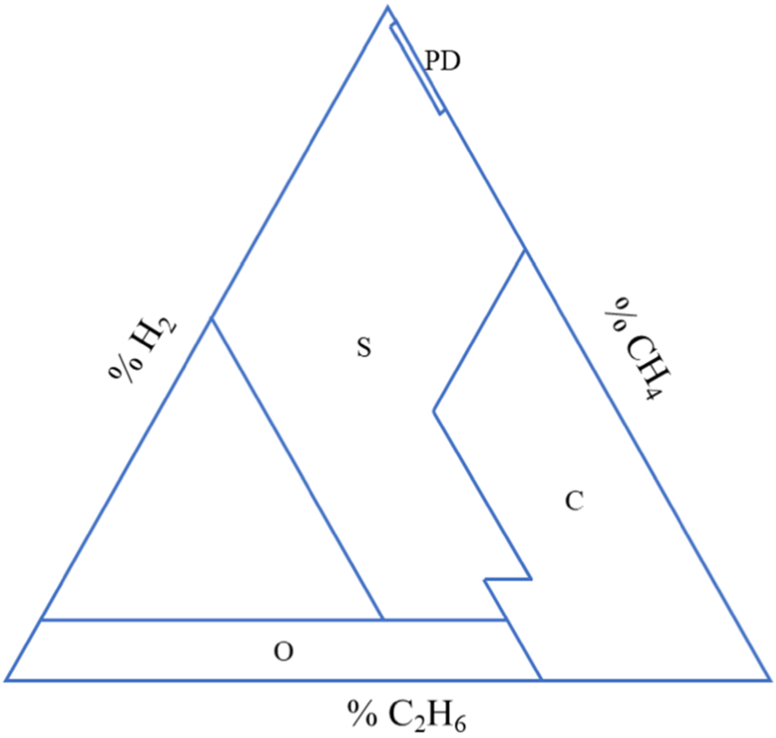


Figure 2.6: Duval triangle 4 [13]

2.2.1.4 Duval triangle 5

This version employs gases (CH_4, C_2H_4 and C_2H_6) which are formed more specifically for thermal defects causing overheating of mineral oil and carbonization of paper. The seven zones that form triangle 5 which are: PD, T2, T3, and S, C, O, and the N/D zone. the shape of this zones shown in Figure 2.7.

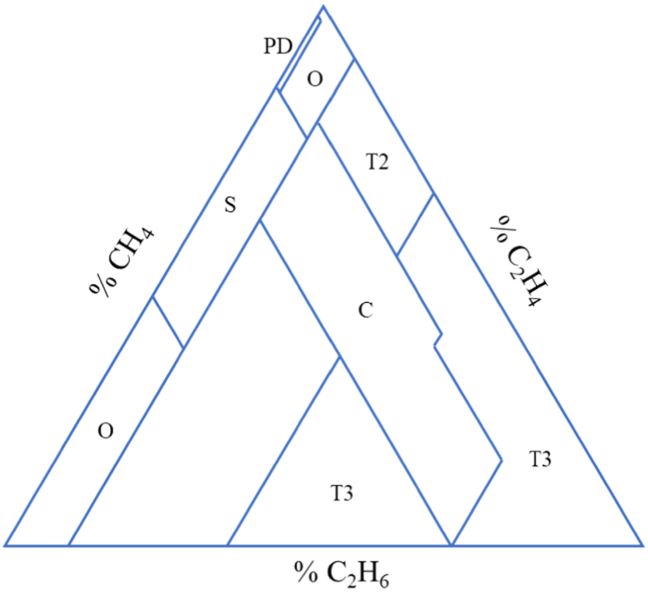


Figure 2.7: Duval triangle 5 [13]

2.2.2 Duval pentagon

This method was developed in 2014 by M. Duval [14]. Like the triangle method, Duval's pentagone is a graphical representation for visualizing different faults. Two types of pentagon can be distinguished : Pentagon I and Pentagon II.
In the case of Pentagon I, in addition for the six electrical and thermal faults that exists (PD, D1, D2, T1, T2 and T3), Duval has included a seventh fault named "Stray gassing" [14], designated by the indication S and standardized by IEC in 2015 (IEC 60599) [68].

The pentagon II defects are: DP, D1, D2, thermal defect T3-H in oil only, thermal defects T3-C, T2-C and T1-C with paper carbonization (C), oil overheating T1-O $<$ 250°C (O) and stray gassin S (Table II.11). The oil decomposition gases of interest are hydrogen (H_2), methane (CH_4), acetylene (C_2H_2), ethylene (C_2H_4), and ethane (C_2H_6). Duval's pentagon I and II method uses percentage concentrations of these gases. This method is based on the principle of separating fault zones by hyperplanes [69]. Each type of Duval pentagon can identify seven defects, the differences between these two types of pentagon are summarized in Table 2.10 and Figure 2.8.

Table 2.10: Comparison of fault types in the two versions of the duval pentagon [70]

Duval's Pentagon I	Duval's Pentagon II
PD: Partial discharges	PD: Partial discharges
D1: Low-energy discharges	D1: Low-energy discharges
D2: High-energy discharges	D2: High-energy discharges
T3: Thermal faults > 700°C	T3-H: Thermal fault T3-H in oil only
T2: Thermal faults 300°C to 700°C	C: Thermal faults T3-C, T2-C, and T1-C with carbonization of paper
T1: Thermal faults < 300°C	O: Overheating T1-O < 250°C
S: Stray gassing	S: Stray gassing

For a given sample and Duval pentagon, the calculation steps, detailed in [14], can be summarized as follows:

1. Calculation of gas percentages.
2. Plot each gas percentage on its respective axis.
3. Calculation of the coordinates of the percentages of the five gases.
4. Calculate the area of the irregular polygon.
5. Calculate the coordinates of the irregular pentagon's center of mass.
6. Determine the defect by locating this center of mass in the defect zone.

Figure 2.8 shows the exact location of each of the seven zones in the Duval pentagon, with each zone corresponding to a specific defect [14].

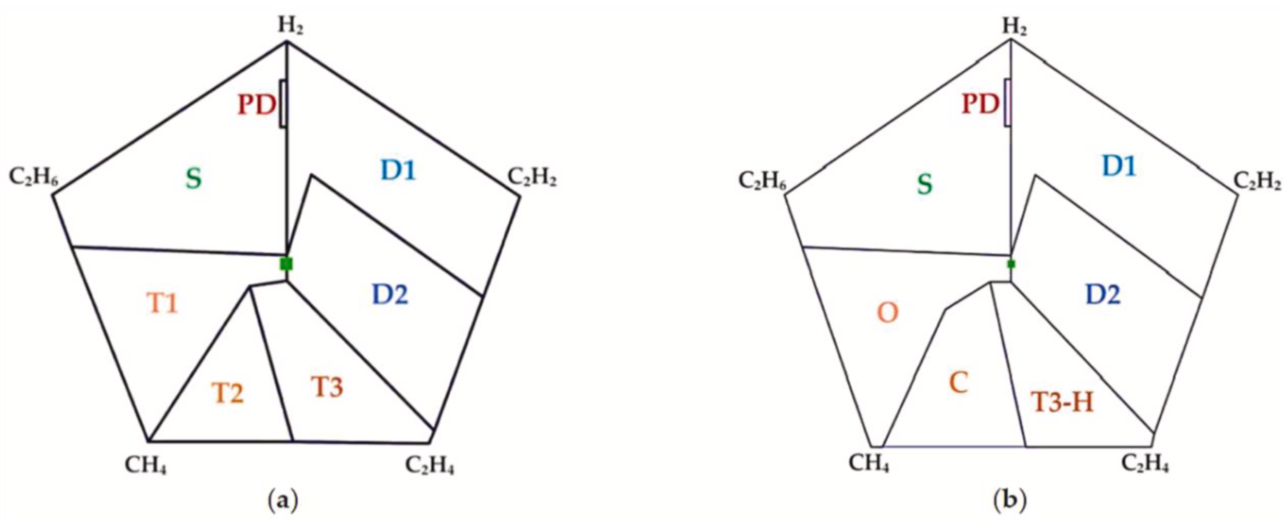


Figure 2.8: Duval Pentagon to represent fault areas: (a) Pent I; (b) Pent II [70]

2.3 Conclusion

In this chapter, we have presented the most popular conventional methods for analyzing dissolved gases in power transformer oil. Despite the simplicity of individual gas methods (TDGC and KGM), their defect detection rates remain modest. As a result, they are not widely accepted by the scientific community. On the other hand, ratio methods (Dornenburg, Rogers and IEC 60599) have been adopted, as their classification rates are acceptable. However, the use of these methods is limited by their disadvantages (requirement of significant gas concentrations, presentation of non-interpretable cases, etc.). Duval's graphical methods (triangle and pentagon) offer appreciable classification rates. These methods are still reliable and relevant today.

To overcome the shortcomings of conventional methods and/or improve their fault detection rates, it is imperative to develop more accurate diagnostic approaches based on machine learning algorithm's. With this in mind, this method's have been explored, comparing different classification algorithms such as ensemble learning methods and SVM. These AI-based approaches offer significant improvements in accuracy and reliability for fault detection by leveraging advanced data processing techniques and adaptive learning capabilities. To address the limitations of conventional methods and enhance fault detection rates, it is essential to develop more accurate diagnostic approaches based on machine learning algorithms. In this context, we have explored various methods, comparing different classification algorithms including ensemble learning methods, support vector machines (SVM) and using those algo's with a decision tree principle. These AI-based approaches provide substantial improvements in accuracy and reliability for fault detection by utilizing advanced data processing techniques and adaptive learning capabilities.

Chapter 3

Classification techniques

3.1 Introduction :

Classification tasks play a pivotal role within the world of Artificial Intelligence (AI), demonstrating versatility across numerous domains due to their capability to process a wide array of datasets. The landscape of classification is characterized by a continuous evolution of algorithms, each aimed at improving accuracy and efficiency.

In this chapter, we explore the theoretical foundations of two key classification algorithms: Support Vector Machines (SVM) and ensemble learning methods. We will begin by discussing tree-based methods, focusing on decision trees, which serve as the foundation for many ensemble learning models. Following this, we will delve into ensemble methods, dividing them into bagging methods (such as Random Forest and Extra-Trees) and boosting methods (including Gradient Boosting and XGBoost).

3.2 Support vector machine

Support Vector Machines (SVM) are supervised statistical learning techniques. These techniques are designed to solve discrimination and regression problems involving high-dimensional data. SVM classifiers sometimes supplantes neural networks and other learning techniques [17]. SVMs were first introduced by V. Vapnik in 1995 [19]. They are widely used in statistical learning and have been very successful in almost every field in which they have been applied. They have been used, for example, in medical diagnostics (MRI image sequence analysis, cardiac arrhythmia detection, cancer risk assessment, etc.), pattern recognition (image and speech processing), and so on [71, 72].

SVM classifiers were first developed for classification problems involving two groups or classes (also known as binary classification) [73], before being extended to multiple or multi-class classification [74]. SVM classifiers generate a quadratic optimization problem [75]. Solving the quadratic optimization problem means finding the separation hyperplane between two classes with maximum margin. The data may be linearly or non-linearly separable.

3.2.1 Binary SVM

3.2.1.1 Linearly separable DATA

For a dataset $T = \{x_k, y_k\}_k^m$, where x_k is the input vector, $y_k \in \{1, -1\}$ denotes the class label of x_k, and m is the total number of data points, the decision function $f(x)$ of SVM can be written as follows [73]:

$$f(x_k) =< wx_k > +b = \sum_{k=1}^{m} w_k x_k + b \tag{3.1}$$

w is the weight vector and b the polarization term (to determine the separation position of the hyperplane).

$< wx_k >$ denotes the dot product between the weight vector w and the feature vector x_k The dot product essentially measures the similarity between w and x_k indicating how much x_k aligns with the learned pattern represented by w.

A definitely separating hyperplane satisfies the following constraints:

- $f(x_k) \geq 1$ if $y_k = 1 \quad \rightarrow \quad x_k$ belongs to class 1
- $f(x_k) \leq -1$ if $y_k = -1 \quad \rightarrow \quad x_k$ belongs to class - 1

These constraints can be merged as follows:

$$y_k\,(< w \cdot x_k > +b) \geq 1 \quad \text{pour} \quad k = 1, 2, \ldots, m \tag{3.2}$$

As shown in Figure 3.1, the position of the separation hyperplane is defined by w and b. The maximum margin (corresponding to the maximum distance between the hyperplane and the nearest data) is evaluated by $\frac{2}{||w||}$

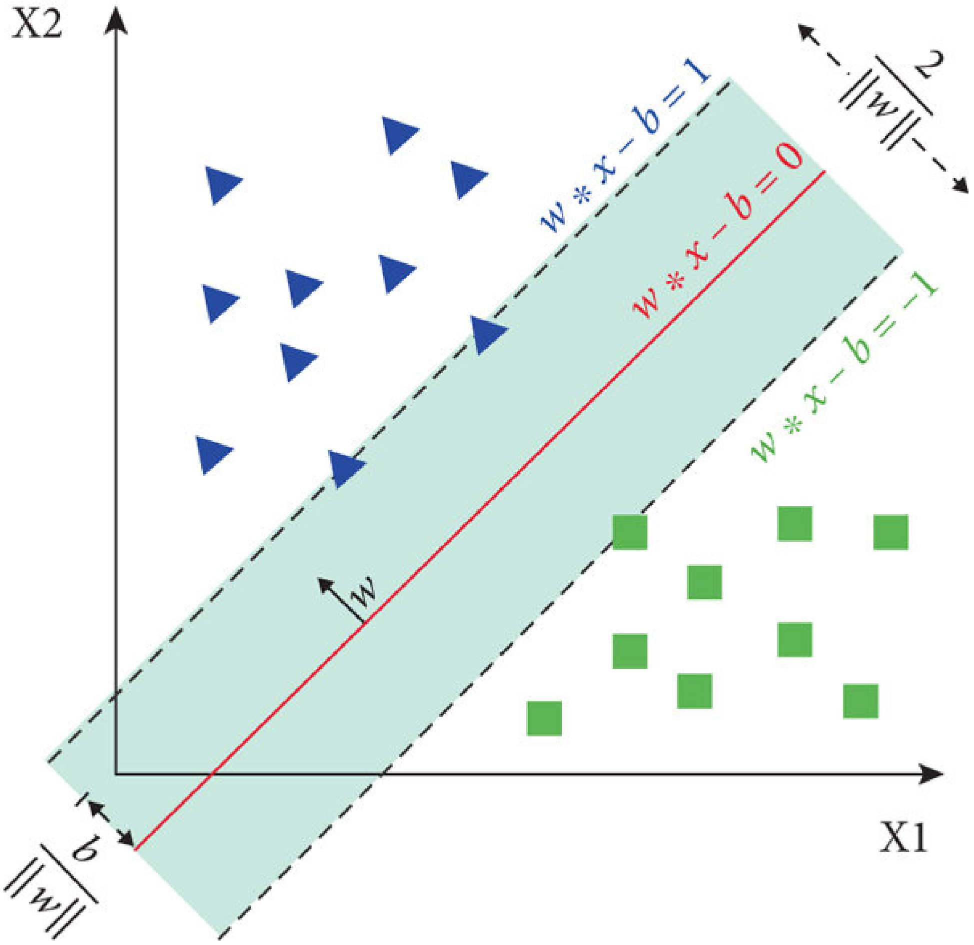


Figure 3.1: linear separation between two classes in SVM [76]

The separation hyper-plan should satisfy the next constraints :

$$\begin{cases} y_k\,(< w \cdot x_k > +b) \geq 1 \quad k = 1, 2, \ldots, m \\ \min \frac{1}{2}\|w\|^2 \end{cases} \tag{3.3}$$

$\|w\|^2 = w^T w$
The Lagrange principle is used to find the optimal solution for the hyperplane. Lagrange transforms Equation 3.3 into :

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{k=1}^{m} \alpha_k [y_k(< wx_k > +b - 1] \tag{3.4}$$

$$\begin{cases} \dfrac{\partial L(w, b, \alpha)}{\partial w} = 0 \\ \dfrac{\partial L(w, b, \alpha)}{\partial b} = 0 \end{cases} \tag{3.5}$$

The resolution of Equation 3.5 gives:

$$\begin{cases} w = \sum_{k=1}^{m} \alpha_k \cdot x_k \cdot y_k \\ \sum_{k=1}^{m} \alpha_k \cdot y_k = 0 \end{cases} \tag{3.6}$$

a problem like this could be solved using the double representation of wolfe [77]:

$$\begin{cases} \text{maximize } L(w, b, a) \\ \text{subject to } w = \sum\limits_{k=1}^{m} a_k \cdot x_k \cdot y_k \\ \sum\limits_{k=1}^{m} a_k \cdot x_k = 0 \\ \forall k, a_k \geq 0 \end{cases} \tag{3.7}$$

This system of equations is equivalent to:

$$\begin{cases} \text{maximize } \sum\limits_{k=1}^{m} a_k - \frac{1}{2} \sum\limits_{k,j} a_k a_j y_k y_j < x_k, x_j > \\ \text{subject to } \forall k, a_k \geq 0, \text{ and } \sum\limits_{k=1}^{m} a_k \cdot x_k = 0 \end{cases} \tag{3.8}$$

3.2.1.2 Non linearly separable data

In the case of non-linearly separable classes Figure 3.2, we introduce the margin of error.

the margin of error (ζ_k) into the hyperplane Equation 3.9, as follows:

$$\begin{cases} y_k \,(< w \cdot x_k > +b) \geq 1 - \zeta_k \quad k = 1, 2, \ldots, m \\ \forall k, \quad 0 < \zeta_\mathrm{k} < 1 \end{cases} \tag{3.9}$$

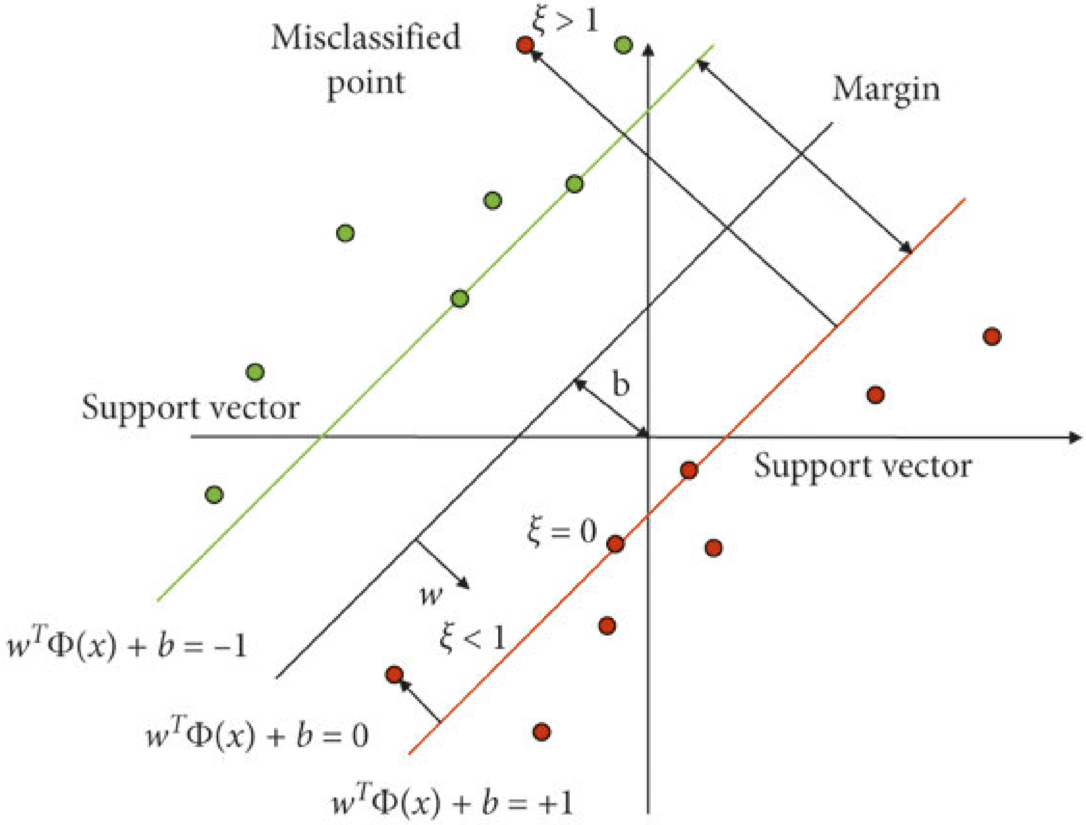


Figure 3.2: Non linear separation between two classes in SVM [78]

”C” is the margin parameter or regularization constraint (the cost attributed to a misclassified sample). The optimization problem becomes :

$$\begin{cases} \text{minimze } \frac{1}{2}\|w\|^2 + C\sum\limits_{k=1}^{m}\zeta_k \\ \text{subject to } y_k(w.x_k + b) \geq 1 - \zeta_k \\ \forall k, 0 < \zeta_k < 1 \qquad k = 1, 2, ..., m \end{cases} \tag{3.10}$$

As a result, in the non-linearly separable case, SVM cannot find any separable hyperplane. The trick is to find a nonlinear transformation ϕ that represents the original points in a much higher-dimensional space where it will be easier to find a linear separator, as shown in Figure 3.3.

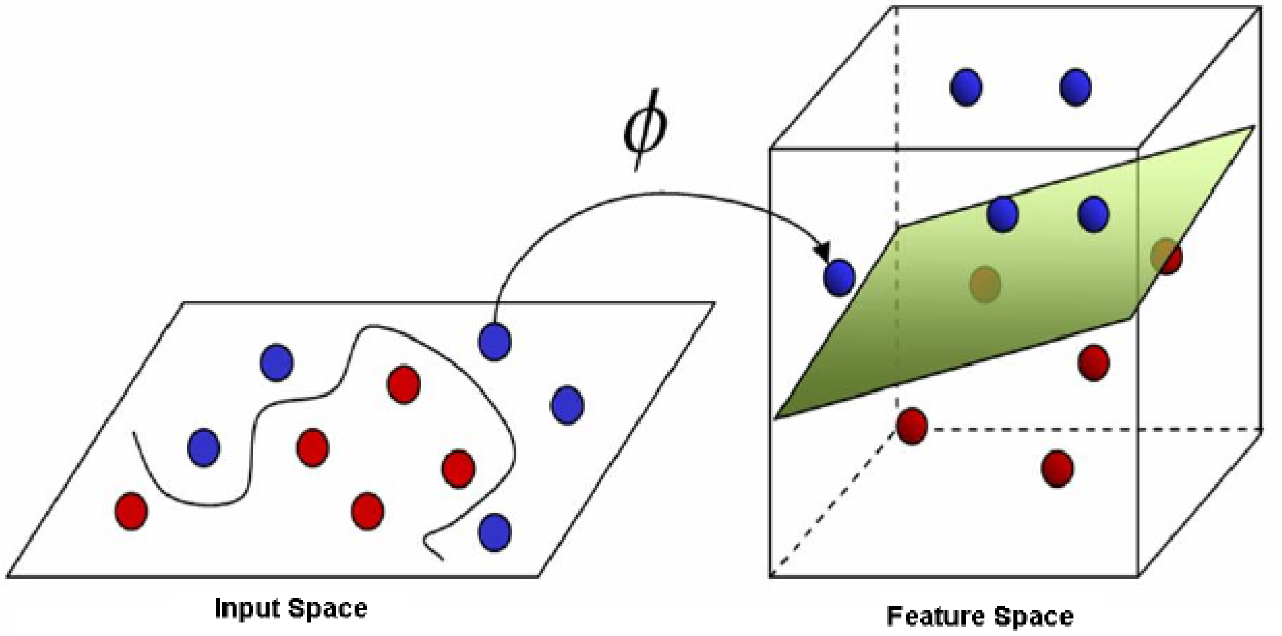


Figure 3.3: Space changing [79]

This idea was introduced by Boser and al [80]. The aim of this idea is to place the observations in a higher-dimensional Hilbert T space. The resulting T space is called the feature space or transformed space. Applying the Hermitian transformation to the system of Equation 3.8, the optimization problem becomes :

$$\begin{cases} \text{maximize } \sum\limits_{k=1}^{m} \alpha_k - \frac{1}{2} \sum\limits_{k,j} \alpha_k \alpha_j y_k y_j < \phi(x_k), \phi(x_j) > \\ \text{subject to } \forall k, 0 \geq \alpha_k \geq C, \text{ and } \sum\limits_{k=1}^{m} \alpha_k x_k = 0 \end{cases} \tag{3.11}$$

It remains to determine the transformation ϕ. However, this transformation could be compromised if the dimension of the space of $\phi(x)$ is very large. To surpass this problem, Aizerman [81] introduced a method known as the **"kernel trick"**. Under certain assumptions about ϕ, the scalar product $< \phi(x_i), \phi(x_j) >$ can be computed using a symmetric function k (kernel). The kernel k must satisfy Mercer's conditions [82] (k must be symmetric and positive semi-definite).

$$k(x_i, x_j) =< \phi(x_i), \phi(x_j) > \tag{3.12}$$

All that needs to be done is to replace the scalar product by the non-linear kernel function. As a result, SVM classifiers can find a linear separating hyperplane. The system Equation 3.11 is then transformed into the following form:

$$\begin{cases} \text{maximize } \sum\limits_{k=1}^{m} \alpha_k - \frac{1}{2} \sum\limits_{k,j} \alpha_k \alpha_j y_k y_j k(x_k, x_j) \\ \text{subject to } \forall k, 0 \geq \alpha_k \geq C, \text{ and } \sum\limits_{k=1}^{m} \alpha_k x_k = 0 \end{cases} \tag{3.13}$$

In practice, a few families of configurable kernel functions are commonly used. It is simply up to the user to carry out tests to determine which one is best suited to his or her application. In the literature, we find the following kernel functions [83]:

- Linear function: $k(x, y) = x.y'$

- Polynomial function: $k(x,y) = (c + x.y^{'})^{d}$
- Radial basis function (RBF) : $K(x,y) = \exp - \frac{\|x - y^{'}\|}{2\sigma^2}$
- The sigmoid function: $k(x,y) = tanh(\alpha_0(x.y^{'}) + \beta_0)$

3.2.2 SVM for multi-classes classification

In practice, most classification problems contain more than two classes. There are several approaches (strategies) to solving the multi-class problem, consisting of "all-together", "one-against-one" and "one-against-all". Comparing these three strategies, it was found that "one-against-one" and "one-against-all" produces better accuracy and requires fewer support vectors, and consequently less computation time [84]. As a result, this two strategy's were adopted in the present study. In the following, we present the two learning process.

3.2.2.1 One-versus-one

The one-versus-one (OvO) strategy was first associated with SVM by Mayoraz and Alpaydin [84]. This strategy constructs $N\frac{(N-1)}{2}$ classifiers (N being the number of classes) where each is formed by data from two classes. Therefore, we need to find $N\frac{(N-1)}{2}$ decision functions. The solution to the multi-class problem is to reformulate it so as to obtain a binary (two-class) classification. Each classifier separates the data into two categories; the first comprises a single class, while the other data are merged to form the second class. Figure 3.4 illustrates the different scenarios that result, for a 4-class problem.

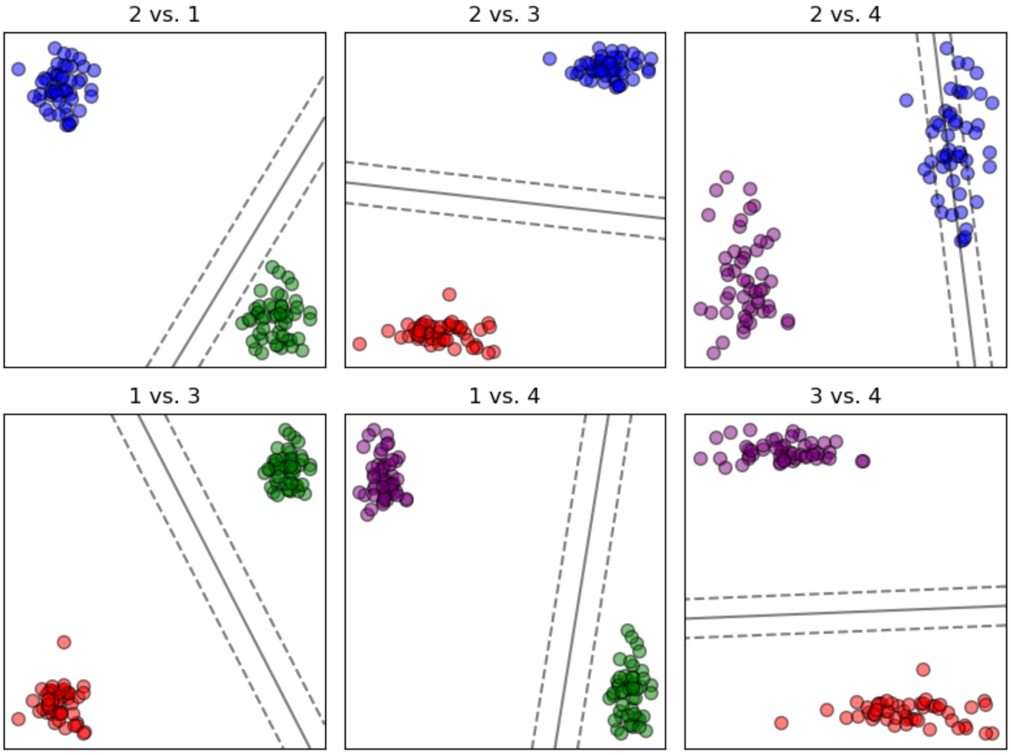


Figure 3.4: one versus one strategy for a 4 classes classification Problem

3.2.2.2 One-versus-all

The one-versus-all (OvA) strategy, also known as one-versus-rest (OvR), is a common approach in multi-class classification. In OvA, each class is treated as a separate binary classification problem, with one classifier trained to distinguish that class from all others. This results in N classifiers for N classes, where N is the number of classes. During prediction, the class assigned to an input data point is determined by the classifier with the highest confidence score. This strategy simplifies the classification task by breaking it down into multiple binary tasks, which are often easier to solve. For illustration, refer to Figure 3.5.

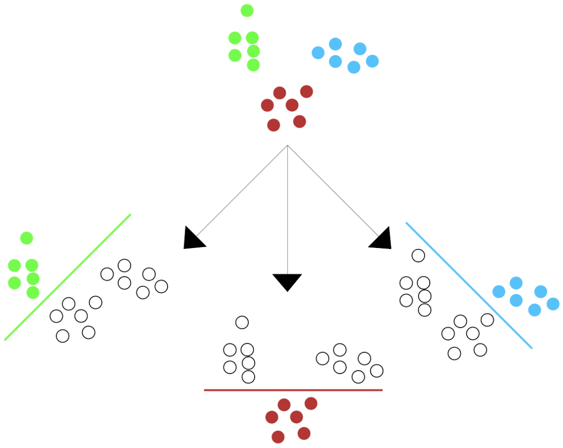


Figure 3.5: one versus all strategy for a 3 classes classification Problem

3.2.3 Advantages and disadvantages of SVM

The Support Vector Machine (SVM) algorithm addresses problems through a convex function, featuring a single global optimum that corresponds to minimizing structural risk. This approach aims to identify the hypothesis with superior generalization capabilities within a given hypothesis space. A crucial aspect of this process is the selection of the kernel function, which often includes adjustable parameters. However, finding the optimal values for these parameters does not equal searching for the minimum of a convex function. Instead, determining the ideal parameters for the kernel function, known as model selection, is essential [85, 86]. Choosing an inappropriate kernel function or its parameter values can result in over-fitting. Additionally, the duration of the SVM algorithm's training phase is significant, especially concerning model selection and when dealing with large training datasets.

3.3 Tree-Based methods

Tree-based methods use a sequence of if-then rules to partition the feature input space into regions, with each region representing a class or an average of the values in that region. Figure 3.6 illustrates an example of a simple tree structure where the first if-then rule asks if the first feature X_1 is less than or equal to the threshold t_1. If true, the next feature is compared to a new threshold value t_2; if also true, we end up in region R_1. If not true, feature X_1 is compared against a new threshold t_3. This principle is used in building Classification and Regression Trees (CART) [87], commonly known as decision trees.

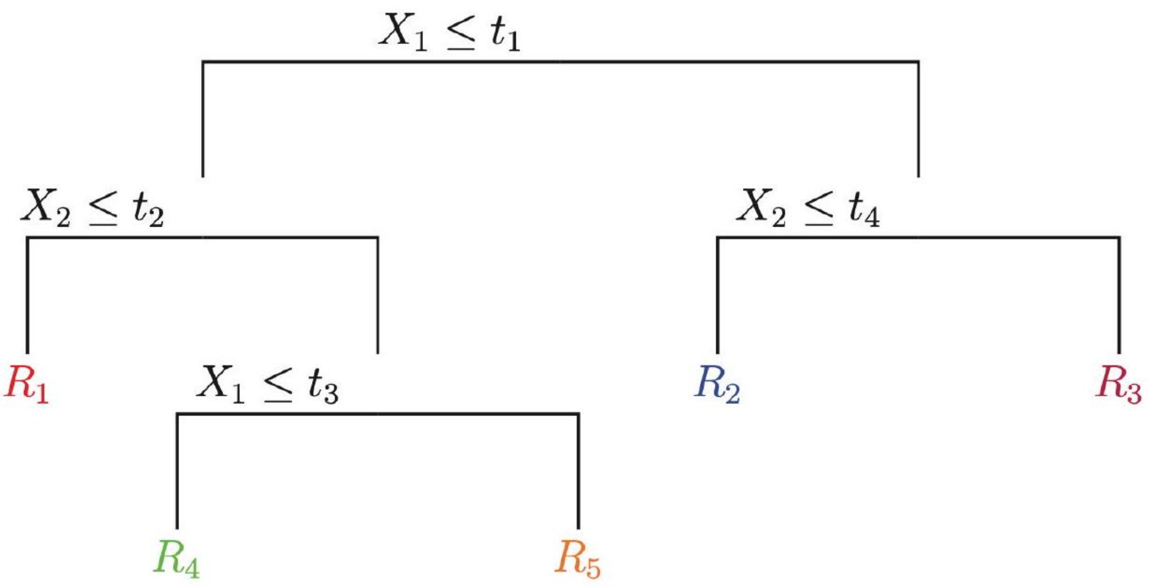


Figure 3.6: Example of a simple tree structure [87]

More formally, consider the classification setting with two classes, C_1 and C_2, given the data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^k$ and $y_i \in \{C_1, C_2\}$. The feature space $\mathcal{X}$ is partitioned into J regions $R_1, \ldots, R_J$. The model can be written as:

$$\hat{y}_i = f(\mathbf{x}_i) = \mathbb{E}[y_i \mid \mathbf{x}_i] = \sum_{j=1}^{J} b_j \mathbb{I}\{\mathbf{x}_i \in R_j\}, \tag{3.14}$$

where b_j is the most common class in the j-th region, determined by a majority vote. Specifically,

$$\hat{b}_{j,c} = \frac{1}{N_j} \sum_{x_i \in R_j} \mathbb{I}\{y_i = c\}, \quad c \in \{C_1, C_2\}, \tag{3.15}$$

where N_j is the total number of data points in region j. A new observation $\mathbf{x}_{i+1} \in R_j$ is assigned to the class with the most votes:

$$\hat{y}(\mathbf{x}_{i+1}) = \arg\max_c \hat{b}_{j,c}. \tag{3.16}$$

3.3.0.1 Decision tree construction

Here is a step-by-step process for building a Decision Tree for classification [88]:

- **Tree Initialization:** Start with the root node containing the entire dataset $\mathcal{D}$.
- **Node Splitting:** For each node, select the feature that best separates the data based on a certain criterion, such as Information Gain (Equation 3.17) or Gini Impurity (Equation 3.18).

$$\mathrm{IG}(s, t) = \mathrm{H}(t) - \sum_{i=1}^{k} \frac{|t_i|}{|t|} \mathrm{H}(t_i), \tag{3.17}$$

where $\mathrm{H}(t)$ is the entropy of node t, and t_i are the child nodes resulting from the split s.

$$\mathrm{GI}(t) = 1 - \sum_{i=1}^{C} p_i^2, \tag{3.18}$$

where p_i is the probability of an instance being classified into class i, and C is the number of classes ($C = 2$ for binary classification).

- **Split Execution:** Split the node into two child nodes based on the selected feature and split point. Assign instances to the appropriate child node.
- **Repeat the Process:** Recursively apply the splitting process to each child node until the stopping criteria are met.
- **Stopping Criteria:** Define conditions to stop splitting, such as reaching a maximum depth or having a minimum number of instances per node.
- **Leaf Node Prediction:** When a node cannot be split further, it becomes a leaf node. Each leaf node represents a class prediction, with the majority class of instances in the leaf node being assigned as the prediction.

One way of improving the performance is by aggregating the prediction of multiple trees (Bagging). One method for ensembling CART's is through Random Forest or Extra-Tree. One way of improving the performance is by summing up the prediction of multiple trees and where each tree is called a weak learner (Boosting). One method for ensembling weak learners is through Gradient Boosting.

3.3.1 Ensemble learning

Ensemble learning (EL) is a powerful technique in machine learning that involves combining multiple models to solve a particular problem. The core idea is to leverage the strengths of various individual models, often referred to as base learners, to create a more robust and accurate predictive model Figure 3.7. By aggregating the predictions of these base learners, ensemble methods aim to reduce the risk of over-fitting and improve generalization to new, unseen data. There are various ensemble methods, each with unique approaches to combining models, including bagging and boosting [89].

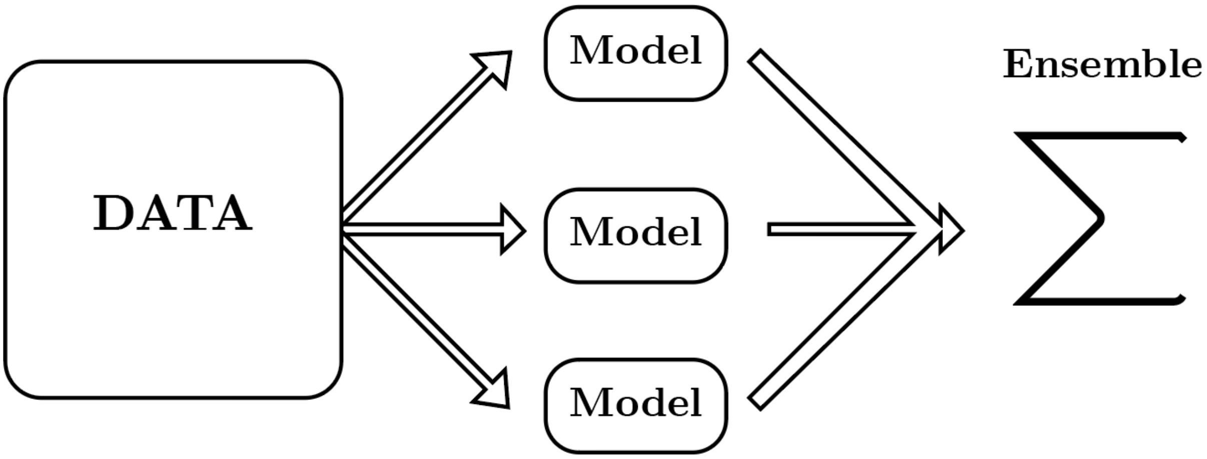


Figure 3.7: Ensemble learning model

The primary distinctions between Bagging and Boosting, The two ensemble learning methods [90], are outlined below:

- **Combining Predictions :** Bagging combines predictions from the same type of models, whereas Boosting combines predictions from different types of models.
- **Model Building Process:** Bagging builds models independently, whereas Boosting influences new models based on the performance of previously built models.
- **Training Data Subsets:** Bagging selects different training data subsets using row sampling with replacement and random sampling methods from the entire training dataset. Boosting, however, ensures every new subset contains elements that were misclassified by previous models.
- **Parallelism vs. Sequential-ism:** In Bagging, base classifiers are trained in parallel, whereas in Boosting, they are trained sequentially.

3.3.1.1 Bagging

In ensemble algorithms, Bagging (short for Bootstrap aggregating) methods build several instances of estimators on random subsets of the original training set (bootstrapping) and then aggregate their individual predictions to form a final prediction. These methods reduce the variance of a base estimator, such as a decision tree, by introducing randomization into its construction and then creating an ensemble. Bagging methods are a simple way to improve performance over a single model without needing to adapt the underlying algorithm. By reducing over-fitting, bagging methods work best with strong and complex models, such as fully developed decision trees, in contrast to boosting methods, which typically work best with weak models, like shallow or pruned decision trees [91]. Bagging involves the two steps mentioned earlier [92].

1. **Bootstrapping:** Bootstrapping is a statistical technique of sampling a data set with replacement. The resulting new data set is called a *bootstrap*. It is a random sampling method that is used to derive samples from the data using the replacement procedure like in Figure 3.8. In this method, first, random data samples are fed to the primary model, and then a base learning algorithm is run on the samples to complete the learning process. bootstraps are generated to simulate different training data set from the same distribution and thus introduce some diversity between the models

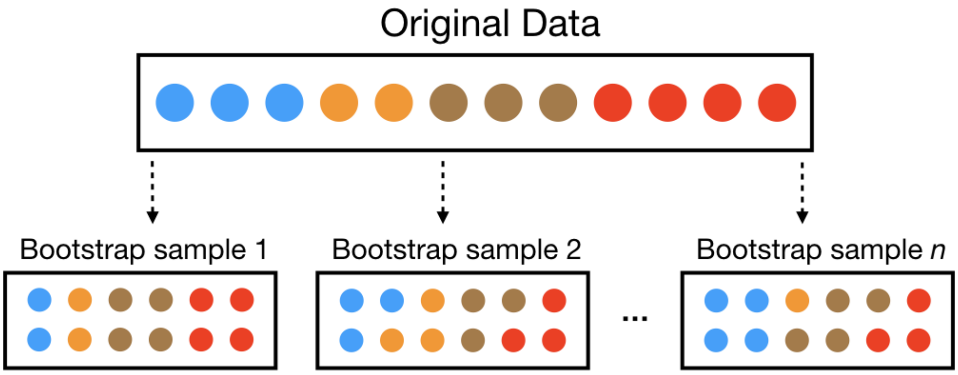


Figure 3.8: Bootstraping from original data

2. **Aggregation:** This is a step that involves the process of combining the output of all base models and, based on their output, predicting an aggregate result with greater accuracy and reduced variance.

We will discuss one of the most common bagging methods, which has provided great accuracy for our problem: the Random Forest algorithm.

3.3.1.1.1 Random forest Random Forest (RF) is an ensemble classifier rooted in decision trees. It is an integration of tree predictors, where each tree depends on values from a randomly sampled vector, common to every tree in the forest. RF can be used to classify categorical response variables or to perform regression on continuous ones. Several studies have reported the application of RF on power transformer condition monitoring and diagnostic [93].

Mathematically, A Random Forest consists of T decision trees. For each tree t, A bootstrap sample $\mathcal{D}_t$ is drawn from the original dataset $\mathcal{D}$ with replacement. A random forest can be built following this steps which are quite similar to the way of building decision trees [91]:

- **Tree Initialization :** Start with the root node containing the entire bootstrap sample $\mathcal{D}_i$.
- **Node Splitting:** For each node, perform the following steps :
    1. **Feature Subset Selection :** Randomly select a subset of m features from the total number of features p (where m is typically $\sqrt{p}$ for classification tasks). This randomness helps to create diverse trees and reduces correlation between trees.
    2. **Best Split Determination :** For each feature in the selected subset, calculate the best split point that maximizes the Information Gain (IG) 3.17 or minimizes the Gini Impurity (GI) 3.18. The best split is chosen based on the criterion used, When all criterions are meet and the trees are completed.
    3. **Split Execution:** Split the node into two child nodes using the best feature and split point determined. Assign instances to the appropriate child node based on the split condition.
- **Repeat the Process :** The splitting process is recursively applied to each child node. This continues until the stopping criteria are met (e.g., a maximum depth is reached, a minimum number of instances per node is met, or no further information gain is possible.) When a node cannot be split further (due to stopping criteria), it becomes a leaf node. Each leaf node represents a class prediction. For classification tasks, the class of the majority of instances in the leaf node is assigned to it.
- **Aggregating Trees in Random Forest :** Once all trees in the forest are built, the Random Forest makes predictions by aggregating the predictions of individual trees, The number of tree's is adjustable in order to obtain the highest accuracy possible. For classification tasks, the final class prediction is determined by majority voting among the trees.

The purpose for the two sources of randomness (bootstrapping and Feature subset selection) is to decrease the variance of the forest estimator. Indeed, individual decision trees typically exhibit high variance and tend to over-fit. The injected randomness in forests yields decision trees with somewhat decoupled prediction errors. By taking an average of those predictions as illustrated in Figure 3.9, some errors can cancel out. Random forests achieve a reduced variance by combining diverse trees, sometimes at the cost of a slight increase in bias. In practice, The variance reduction is often significant, Yielding an overall better model [91].

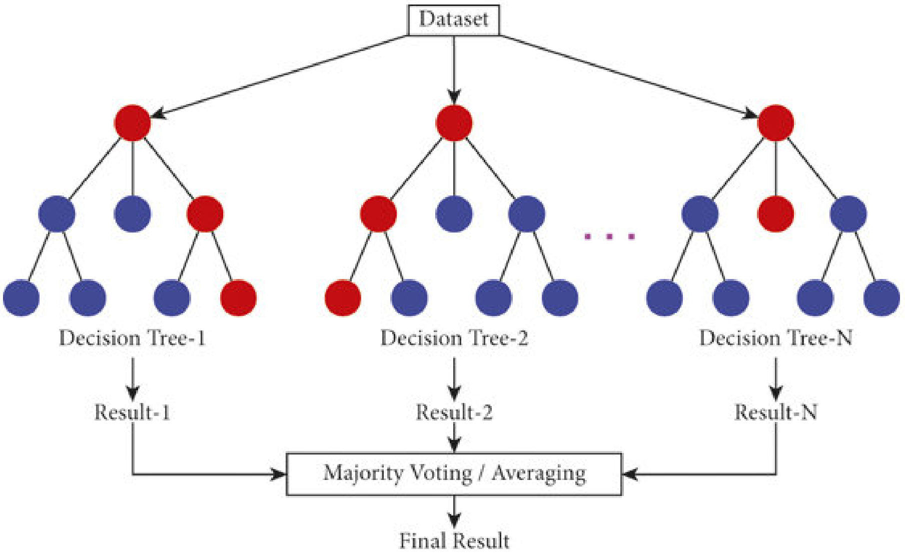


Figure 3.9: Random forest principle [94]

3.3.1.1.2 Extra-Tree The Extremely Randomized Trees Classifier (Extra-Trees Classifier) is an ensemble learning technique akin to Random Forests but with a distinction in how randomness is applied. In Extra Trees, the process of split computation differs by taking an additional step. Similar to Random Forests, a random subset of candidate features is utilized. However, instead of seeking the most discriminating thresholds, thresholds are randomly drawn for each candidate feature. Among these randomly generated thresholds, the best one is selected as the splitting rule. This approach often results in a further reduction of model variance, albeit with a slightly greater increase in bias [95]. In random forests, bootstrap samples are used by default while in extra-trees we use the whole dataset. The Table 3.1 highlights the main differences between the two methods.

Table 3.1: Comparison between extra-trees and random forest algorithms

Feature	**Extra Trees Classifier**	**Random Forest Classifier**
Data Sampling	Uses the entire original training data for splitting at each node.	Uses bootstrapping to create diversified training sets for each tree.
Feature Splitting	Randomly selects the best split point among a random subset of features at each node.	Selects the best split point by considering a random subset of features at each node.
Overall Bias-Variance Trade-off	Generally lower variance but potentially higher bias compared to Random Forest.	Generally lower bias but potentially higher variance compared to Extra Trees.

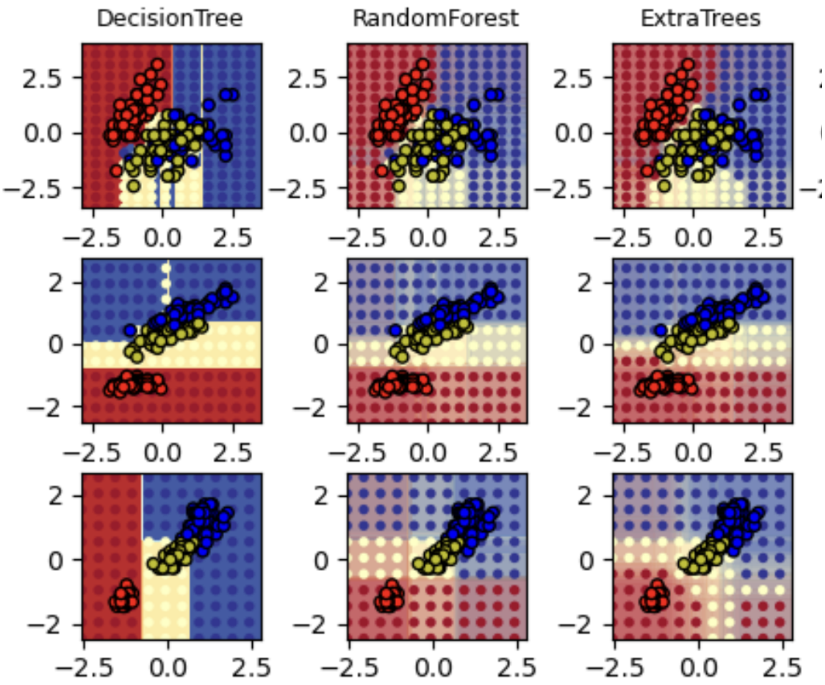


Figure 3.10: comparaison between DT's, RF and Extra classifier on a subset of the Iris dataset [91]

3.3.1.2 Boosting

Boosting is an ensemble method that enables each member to learn from the preceding member's mistakes and make better predictions for the future. Unlike in bagging method, all base learners (weak) are arranged in a sequential format so that they can learn from the mistakes of their preceding learner. Hence, in this way, all weak learners get turned into strong learners and make a better predictive model with significantly improved performance [96]. The general process for boosting is shown in Figure 3.11.

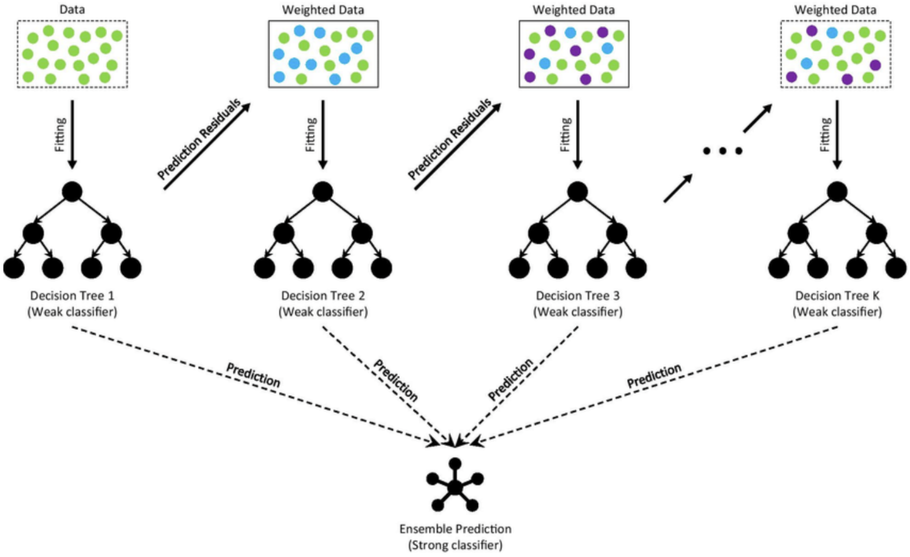


Figure 3.11: Boosting algorithm principle [96]

Our research primarily revolves around using XGBoost, a highly regarded boosting technique. While we recognize and have explored other methodologies like AdaBoost, CatBoost, and Gradient Boosting, our primary emphasis is on XGBoost due to its widespread adoption and proven effectiveness across diverse domains [97]. Although XGBoost and Gradient Boosting are not technically identical, we will describe XGBoost after defining the gradient boost framework for the sake of simplicity. This approach will help in understanding the unique contributions and enhancements introduced by XGBoost by building upon the foundational principles of gradient boosting [96].

3.3.1.2.1 Gradient Boosting The principle of boosting was originally derived in 1990 by Schapire [98]. He showed that any weak learner or base learner could be efficiently 'boosted' into a strong learning algorithm. This means that boosting works by repeatedly running a weak learner on different versions of the training data and combining the classifier into a single strong classifier. The model can be written in the form [97]:

$$f(\mathbf{x}) = \sum_{m=1}^{M} \beta_m \phi\left(\mathbf{x}; \mathbf{a}_m\right) \tag{3.19}$$

The function $\phi\left(\mathbf{x}; \mathbf{a}_m\right)$ is the m-th generic weak learner, each typically parameterized by $\mathbf{a}_m$ but not necessary, and $\beta_m \in \mathbb{R}$ can be seen as a weight of the weak learner that measures how important has the m_{th} weak learner been in the final boosted model. When using decision trees, we obtain what has been referred to as Gradient Boosting Decision Trees (GBDT).

The objective is to find the classifier that is optimal in some sense according to a metric that measures the 'loss', i.e., the following optimization problem:

$$\min_{f} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f\left(\mathbf{x}_i\right)\right) \tag{3.20}$$

Where $\mathcal{L}(y, \hat{y})$ is an arbitrary loss function, and f is given by equation 3.19. It is possible to derive a solution, known as Gradient Boosting, proposed by Friedman [99].

Solving the optimal f is hard, and the solution builds on a numerical optimization scheme that takes the form

$$\hat{y}_i = f\left(\mathbf{x}_i\right) = \sum_{m=0}^{M} f_m\left(\mathbf{x}_i\right) \tag{3.21}$$

with an initial guess $f_0(\mathbf{x})$ and $\{f_m(\mathbf{x})\}_{m=1}^{M}$ are incremental functions ('boosts'). Then with the assumption of a parameterized form of f as in equation 3.19, The minimized estimated loss is

$$\left(\beta_m, \mathbf{a}_m\right) = \arg\min_{\beta, \mathbf{a}} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f_{m-1}\left(\mathbf{x}_i\right) + \beta\phi\left(\mathbf{x}_i; \mathbf{a}\right)\right) \tag{3.22}$$

Then set

$$f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \beta_m \phi\left(\mathbf{x}; \mathbf{a}_m\right) \tag{3.23}$$

The minimization problem of equation 3.22 for arbitrary loss function is infeasible; therefore, this is solved stage-wise using gradient descent. Let the gradient be denoted $\mathbf{g}_m = \{g_m(\mathbf{x}_i)\}_{i=1}^N$ and is the best steepest-decent step direction in the functional space f_{m-1}. The gradient takes the form

$$g_{im} = \left[\frac{\partial \mathcal{L}(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}\right]_{f=f_{m-1}} \tag{3.24}$$

and then the step taken is

$$f_m = f_{m-1} - \rho_m \mathbf{g}_m \tag{3.25}$$

ρ_m Is the step length which in turn can be calculated with the aid of line search, a method of finding a local minimum of the following objective function

$$\rho_m = \arg\min_{\rho} \sum_{i=1}^{N} \mathcal{L}(y_i, f_{m-1}(\mathbf{x}_i) + \rho\phi(\mathbf{x}_i; \mathbf{a}_m)) \tag{3.26}$$

However, the gradient is only defined at the data points. Therefore, we do not learn a function that can generalize. To address this, the algorithm is modified by fitting a weak learner to approximate the negative gradient $-g_m(\mathbf{x})$ using a least-squares function minimization constraint

$$\mathbf{a}_m = \arg\min_{\mathbf{a}} \sum_{i=1}^{N} [-g_{im} - \phi(\mathbf{x}_i; \mathbf{a})]^2 \tag{3.27}$$

3.3.1.2.2 Extreme Gradient Boost (XGBoost) is an optimized and scalable implementation of the Gradient Boosting presented in Section 3.3.1.2.1. It is publicly available as an open-source library for large-scale and distributed computing environments. XGBoost has several advanced features that make it a powerful tool for predictive modeling, including support for missing values, parallel processing, early stopping to prevent over-fitting, and regularization to control over-fitting. In summary, Gradient Boosting is a general concept, and XGBoost is an optimized implementation of Gradient Boosting.

In Section 3.3.1.2.1, the basis function or weak learner $\phi(\cdot)$ was not explicitly determined when we defined the gradient boosting concept. In the XGBoost implementation, the weak learner is a CART discussed earlier in Section 3.3 where the data is recursively split based on the value of different features, to minimizing impurity in equations 3.18 and 3.17. Let the tree ensemble model be made up of K trees

$$\hat{y}_i = f(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i) \tag{3.28}$$

$f_k(\mathbf{x}_i) = \phi\left(\mathbf{x}_i; \{b_j, R_j\}_{j=1}^J\right) = \sum_{j=1}^J b_j \mathbb{I}(\mathbf{x} \in R_j)$, hence the weak learner is a parameterized CART tree with coefficients, $b_j \in \mathbb{R}$ and boundary regions/splitting variables R_j. Then the goal is the same as stated in Equation 3.20, find the function (sum of trees) that optimizes according to some objective function. In the setting of XGBoost, the objective has an added regularization term

$$\mathcal{L} = \sum_{i=1}^{N} L\left(y_i, f\left(\mathbf{x}_i\right)\right) + \sum_{k=1}^{K} \Omega\left(f_k\right) \tag{3.29}$$

So that the objective becomes the sum of training loss and the complexity of the trees. The regularization term

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda\|b\|^2 \tag{3.30}$$

$\Omega(f)$ is a contribution of the number of leaf nodes (regions) T scaled by a hyperparameter γ and the L2-norm of leaf weights, b scaled by a hyperparameter λ. As previously shown, this optimization problem is infeasible and relies on an additive approach, so for each step t, the goal is to solve Equation 3.22, which with the added regularization term, becomes

$$\mathcal{L}_t = \sum_{i=1}^{N} L\left(y_i, \hat{y}_i^{(t-1)}\left(\mathbf{x}_i\right) + f_t\left(\mathbf{x}_i\right)\right) + \Omega\left(f_t\right) \tag{3.31}$$

The goal is to minimize the objective function with the proposed gradient boosting algorithm, which requires the calculation of the gradient. Chen's derivation in [100] takes the second-order Taylor expansion at the previous prediction y_i^{t-1}, hence the objective at step t becomes

$$\mathcal{L}_t = \sum_{i=1}^{n} \left[g_i f_t\left(\mathbf{x}_i\right) + \frac{1}{2} h_i f_t^2\left(\mathbf{x}_i\right)\right] + \Omega\left(f_t\right) \tag{3.32}$$

Where $g_i = \partial_{\hat{y}_i}(t-1)\mathcal{L}_t\left(y_i, \hat{y}_i^{(t-1)}\right)$ and $h_i = \partial_{\hat{y}_i}^2(t-1)\mathcal{L}_t\left(y_i, \hat{y}_i^{(t-1)}\right)$ are the first and second derivatives of the loss function. To connect these derivatives to the tree structure, define $I_j = \{i \mid q\left(\mathbf{x}_i\right) = j\}$ as all sample points that belong to leaf node j. Here q represents the structure of one unique tree that maps a sample to a leaf. Then let $f(\mathbf{x}) = b_{q(\mathbf{x})}$ represent the score vector $b \in \mathbb{R}$ of each leaf node given the structure $q : \mathbb{R}^d \to \{1, 2, \ldots, T\}$. With this notation, Equation 3.32 becomes

$$\mathcal{L}_t = \sum_{j=1}^{T_t} \left[G_j b_{t,j} + \frac{1}{2}\left(H_j + \lambda\right) b_{t,j}^2\right] + \gamma T_t \tag{3.33}$$

where $G_j = \sum_{i \in I_j} g_j$ and $H_j = \sum_{i \in I_j} h_i$. In contrast to the previous, it was stated that b_j was set to the average number of samples corresponding to that leaf node. Instead, we derive the optimal value by assuming a fixed tree structure, $q(\mathbf{x})$. Since $G_j b_{t,j} + \frac{1}{2}\left(H_j + \lambda\right) b_{t,j}^2$ is convex w.r.t $b_{t,j}$. Taking the derivative leads to the optimal value of leaf j at step t,

$$b_{t,j}^* = -\frac{G_j}{H_j + \lambda} \tag{3.34}$$

Then the objective function becomes

$$\mathcal{L}_t(q) = -\frac{1}{2}\sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T_t \tag{3.35}$$

$\mathcal{L}_t(q)$ can be used to evaluate the quality of tree structure. It is similar to how decision trees are evaluated, but it works for a broader range of objective functions [100].

Usually, it is not possible to consider every possible tree structure. Instead, a greedy algorithm is used. It starts from a tree with a depth of 0 and adds branches to the tree one at a time. After splitting the data into two sets, the loss reduction is calculated using I_L and I_R as the instance sets for the left and right nodes, respectively. $I = I_L \cup I_R$ represents the entire instance set. The loss reduction after the split is then calculated by

$$\text{Gain } = \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} - \gamma \tag{3.36}$$

where G and H are the gradients and Hessians (second-order derivatives) of the loss function, respectively, R and L refers for the right and left node.

To evaluate the optimal split rule for a node, go through all the available features. For each feature, sort the instances based on their feature values. Then, use a linear scan to determine the optimal split according to the Gain for that feature. Finally, choose the best split, i.e., the one with the highest Gain among all the features.

Finally, the hyper-parameters γ and λ should be optimized when used in a practical setting.

3.4 Conclusion

In this chapter, we presented the essential theoretical foundations for understanding the principles of supervised learning methods, specifically SVM and tree-based methods, which are the foundations of ensemble learning. We discussed bagging (Random Forest and Extra Tree) and boosters (Gradient Boost and Extreme Gradient Boost). In Chapter 4, we will explore the application of these algorithms to diagnosing power transformers oils, their use separately, and how we implemented a custom-built decision tree.

Chapter 4

Application of SVM, Ensemble learning methods and decision tree principle in fault diagnosis

4.1 Introduction

The aim of this chapter is to identify fault types based on dissolved gas analysis of power transformers oils. Leveraging the intelligent methods discussed in Chapter 3, we will use 8 input vectors and 4 preprocessing techniques for the data. hyperparameters tuning for all algorithms will be conducted using Optuna. We will compare the results with conventional methods and introduce our hybrid approach, which integrates a decision tree principle with XGBoost and Random Forest algorithms. At the end, we will evaluate and compare the performance of these machine learning methods against other works.

4.2 Software, libraries, and frameworks

4.2.1 Python

Python is an interpreted, multi-paradigm, and cross-platform programming language. It encourages structured imperative, functional, and object-oriented programming. It features strong dynamic typing, automatic memory management through garbage collection, and an exception handling system. Due to its simplicity and readability, Python has become a popular choice for both beginners and experienced developers.

The Python programming language was created in 1989 by Guido van Rossum in the Netherlands. The name Python is a tribute to the television series Monty Python's Flying Circus, of which G. van Rossum is a fan. The first public version of this language was released in 1991. Since then, Python has evolved significantly and is now widely used in various fields, including web development, data science, artificial intelligence, scientific computing, and more. The language's extensive standard library and vibrant community contribute to its versatility and widespread adoption.

4.2.2 Scikit-learn

Scikit-learn is a free Python library designed for machine learning. It is developed by numerous contributors, particularly in the academic world. Scikit-learn provides simple and efficient tools for data mining and data analysis, making it accessible for both beginners and professionals in data science.

It offers a wide range of ready-to-use algorithm libraries within its framework. These libraries are available to data scientists for various tasks such as classification, regression, clustering, and dimensionality reduction. Scikit-learn includes functions for estimating decision trees, random forests, logistic regressions, and support vector machines (SVM). Additionally, it provides utilities for model selection, evaluation, and preprocessing, facilitating the entire machine learning workflow from start to finish.

Scikit-learn is designed to work seamlessly with other free Python libraries, notably NumPy and Pandas. This interoperability allows for easy integration and manipulation of data, enhancing the overall efficiency of the data analysis process. The combination of Scikit-learn's comprehensive functionality and ease of use has made it a cornerstone in the toolkit of data scientists and machine learning practitioners.

4.2.3 XGBoost

XGBoost, or Extreme Gradient Boosting, is an open-source library designed for gradient boosting algorithms. Developed by Tianqi Chen, it is widely used for supervised learning tasks, particularly in machine learning competitions and industry applications.

XGBoost is known for its efficiency, speed, and accuracy in handling large datasets. It provides implementations for various machine learning algorithms, including regression, classification, and ranking. Its algorithm is based on the principles of gradient boosting, enhancing the performance of weak learners through iterative learning, The library offers a range of features, such as parallelization, regularization, and tree pruning, to optimize model performance.

4.2.4 Optuna

Optuna is an automatic hyperparameter optimization software framework, particularly designed for machine learning and deep learning. It provides a powerful and flexible platform to efficiently search for the best hyperparameters to maximize model performance. Optuna achieves this by using techniques such as Bayesian optimization, which intelligently narrows down the search space based on previous trials.

The hyperparameter tuning process with Optuna involves defining an objective function that evaluates model performance with given hyperparameters. Optuna then uses this function to explore and exploit the hyperparameter space, finding the optimal combination of parameters to maximize accuracy. This process is as follows :

- **Define the Objective Function:** This function trains the model with a given set of hyperparameters and returns a performance metric, such as accuracy.
- **Search Space Definition:** Specify the range of hyperparameters to be optimized, including their types (e.g., integers, floats) and possible values.
- **Optimization Process:** Optuna uses a sampler, such as Tree-structured Parzen Estimator (TPE) or Bayesian optimization, to sample hyperparameters and evaluate the objective function.
- **evaluation and Pruning:** During the optimization process, trials that do not show promise based on intermediate results can be pruned to save computation time.
- **Best Hyperparameters:** After a predefined number of trials or a stopping criterion, Optuna identifies the hyperparameters that resulted in the best performance.

In our study, we employed Optuna to tune hyperparameters across all algorithms, including SVM, XGBoost, Random Forestn and Extra-Tree. By exploring the hyperparameter space, Optuna helped us identify the optimal parameters that maximized the accuracy of our models. This automated approach enhance models performance and significantly reduced the time and effort required for manual hyper-parameter tuning.

4.3 DGA input vector's

Eight different input vectors were considered, following the methodology used by Kherif et al. (2021) [21]. These vectors were designed to effectively represent the concentrations and ratios of dissolved gases in order to achieve higher accuracy.

- **Vector 1:** Since the database contains the concentrations of the five gases in parts per million or ppm, each sample **X** is represented as follows:

$$\mathrm{X} = [H_2, \mathrm{CH_4}, \mathrm{C_2H_2}, \mathrm{C_2H_4}, \mathrm{C_2H_6}]$$

- **Vector 2:** The IEC Ratios method is used to produce the following input vector containing three ratios of the dissolved gases given by:

$$\mathrm{X} = \left[\frac{\mathrm{CH_4}}{\mathrm{H_2}}, \frac{\mathrm{C_2H_2}}{\mathrm{C_2H_4}}, \frac{\mathrm{C_2H_4}}{\mathrm{C_2H_6}}\right]$$

- **Vector 3:** Roger's four-ratio method has been selected in this case to transform each sample to the following one:

$$\mathrm{X} = \left[\frac{\mathrm{CH_4}}{\mathrm{H_2}}, \frac{\mathrm{C_2H_2}}{\mathrm{C_2H_4}}, \frac{\mathrm{C_2H_4}}{\mathrm{C_2H_6}}, \frac{\mathrm{C_2H_6}}{\mathrm{CH_4}}\right]$$

- **Vector 4:** Dornenburg's method is also investigated in this study. In this method, the input consists of four ratios computed as a function of the dissolved gases in ppm as follows:

$$\mathrm{X} = \left[\frac{\mathrm{CH_4}}{\mathrm{H_2}}, \frac{\mathrm{C_2H_2}}{\mathrm{C_2H_4}}, \frac{\mathrm{C_2H_4}}{\mathrm{C_2H_6}}, \frac{\mathrm{C_2H_2}}{\mathrm{CH_4}}\right]$$

- **Vector 5:** Duval triangle is a graphical method that uses only the concentration of three gases ($\mathrm{CH_4}, \mathrm{C_2H_2}$, and $\mathrm{C_2H_4}$) to produce the input vector as follows:

$$\mathrm{X} = [C_x, C_y]$$

  where the components C_x and C_y are computed by:

$$C_x = \frac{x_{CH_4} + x_{C_2H_4} + x_{C_2H_2}}{3}$$

  and

$$C_y = \frac{y_{CH_4} + y_{C_2H_4} + y_{C_2H_2}}{3}$$

  A Python script was used to map the gas concentrations into duval triangle and then retrieve the x and y coordinates, as the result is shown in Figure 4.1.

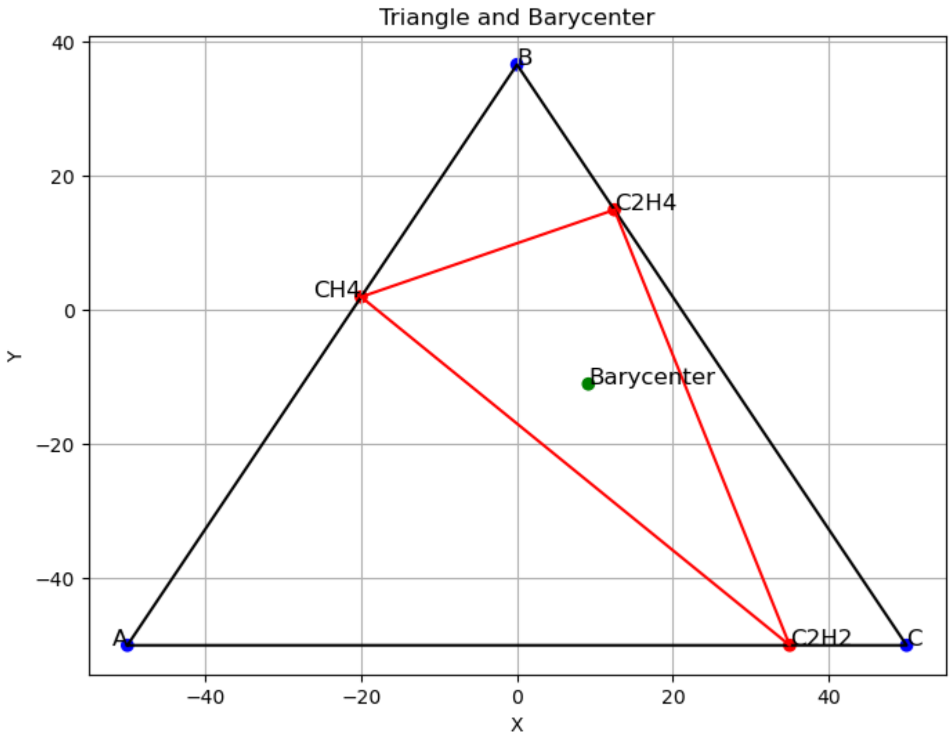


Figure 4.1: Example of mapping points into duval triangle

- **Vector 6:** Duval pentagon is a graphical method that uses the concentration of all five gases ($CH_4, C_2H_2, C_2H_4, C_2H_6, H_2$)) to produce the input vector as follows::

$$X = [C_{px}, C_{py}]$$

where the components C_{px} and C_{py} are computed by :

$$C_x = \frac{x_{H_2} + x_{CH_4} + x_{C_2H_2} + x_{C_2H_4} + x_{C_2H_6}}{5}$$

and

$$C_y = \frac{y_{H_2} + y_{CH_4} + y_{C_2H_2} + y_{C_2H_4} + y_{C_2H_6}}{5}$$

A Python script was also used to map the gas concentrations into duval pentagon and then retrieve the x and y coordinates, as the result is shown in Figure 4.2.

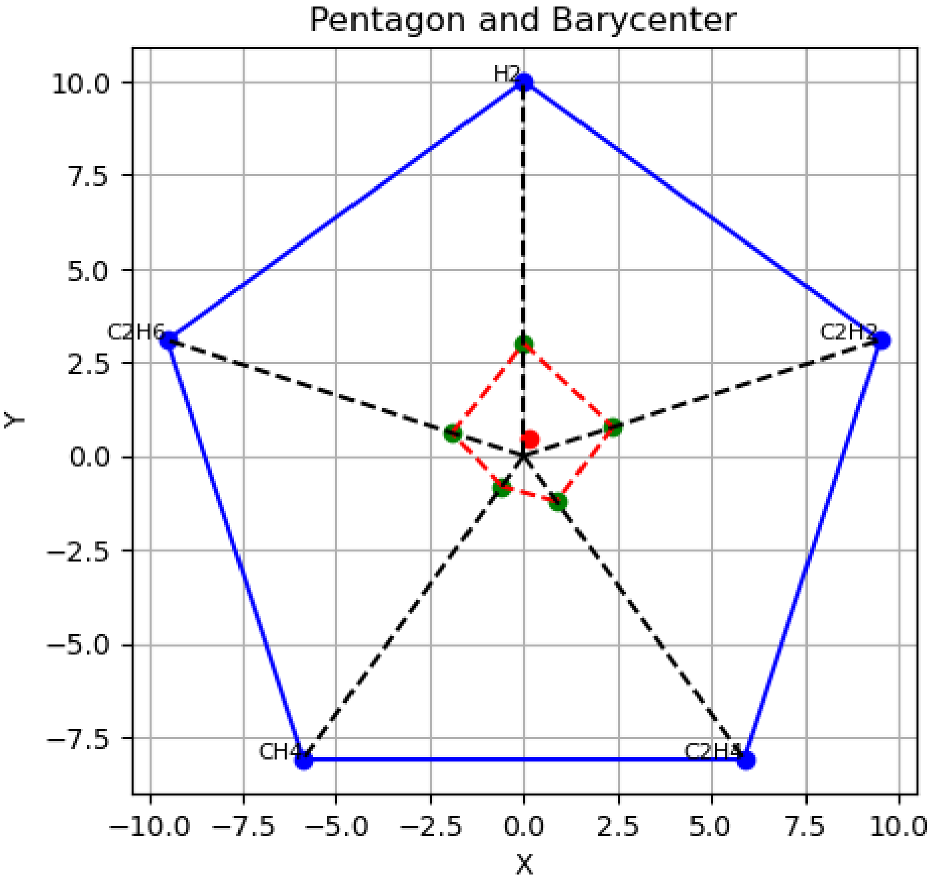


Figure 4.2: Example of mapping points into duval Pentagon

- **Vector 7:** This combination consists of an input vector of five ratios which are the mixture of Roger's and Dornenburg's methods given by:

$$\mathrm{X} = \left[\frac{\mathrm{CH_4}}{H_2}, \frac{\mathrm{C_2H_2}}{\mathrm{C_2H_4}}, \frac{\mathrm{C_2H_4}}{\mathrm{C_2H_6}}, \frac{\mathrm{C_2H_2}}{\mathrm{CH_4}}, \frac{C_2H_6}{CH_4}\right] \tag{4.1}$$

- **Vector 8:** The input vector, in this case, has four components given as follows:

$$\mathrm{X} = [C_x, C_y, C_{px}, C_{py}]$$

According to Duval's triangle and pentagon, this vector has been computed using the concentration of five gases in percentages. C_x and C_y are calculated according to the triangle method, while C_{px} and C_{py} employ the pentagon technique.

4.4 Preprocessing techniques

Preprocessing involves preparing and cleaning raw data before using it in a machine learning algorithm. In this dissertation, four preprocessing techniques were employed: NormalizeScaler, MinMaxScaler, RobustScaler, and StandardScaler [101].

4.4.1 NormalizeScaler

For a dataset with N rows and C columns represented by a matrix X, the NormalizeScaler function scales each feature (column) by dividing each element by the sum of its respective row, computing the normalized matrix X' as:

$$X'_{ij} = \frac{X_{ij}}{\sum\limits_{k=1}^{C} X_{ik}}$$

where X_{ij} represents the element at row i and column j of matrix X, and $\sum\limits_{k=1}^{C} X_{ik}$ is the sum of the elements in row i.

4.4.2 MinMaxScaler

MinMaxScaler scales each feature to a given range (usually [0, 1]) by subtracting the minimum and dividing by the range:

$$X'_{ij} = \frac{X_{ij} - X_{\mathrm{min}_j}}{X_{\mathrm{max}_j} - X_{\mathrm{min}_j}}$$

where X_{min_j} and X_{max_j} are the minimum and maximum values of column j, respectively.

Both NormalizeScaler and MinMaxScaler are effective for datasets with strictly positive values because they preserve the relative relationships between the data points and ensure that all features are within a specific range.

4.4.3 RobustScaler

RobustScaler scales features using statistics that are robust to outliers. It subtracts the median and divides by the interquartile range:

$$X'_{ij} = \frac{X_{ij} - \mathrm{median}(X_j)}{\mathrm{IQR}(X_j)}$$

where $\mathrm{median}(X_j)$ is the median and $\mathrm{IQR}(X_j)$ is the interquartile range of column j.

4.4.3.1 Interquartile Range (IQR)

The interquartile range (IQR) is a measure of statistical dispersion, representing the range between the first quartile (Q1) and the third quartile (Q3) of a dataset. In the context of a specific column j in a dataset, the interquartile range ($\mathrm{IQR}(X_j)$) provides insight into the spread of the data within that column. Q1 and Q3 and the IQR are represnted in Figure 4.3.

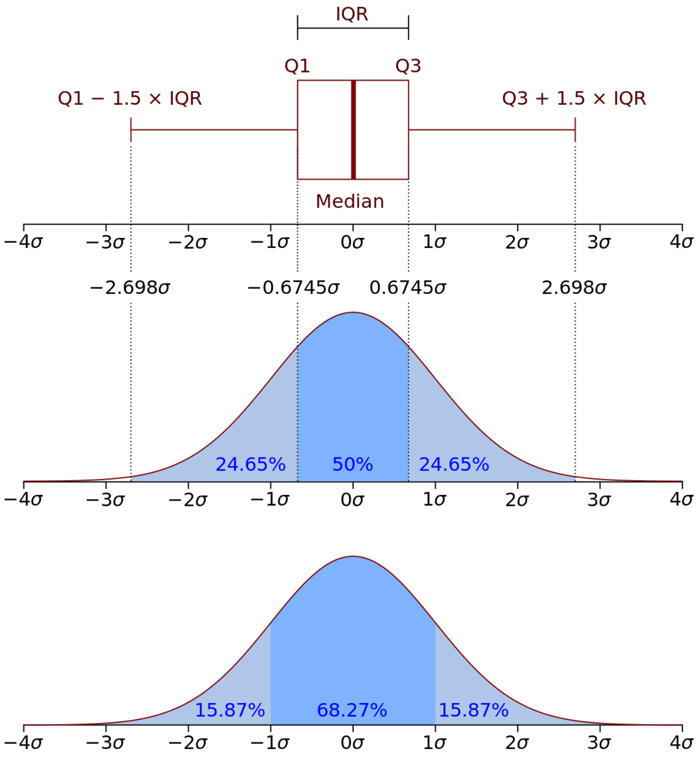


Figure 4.3: Q1, Q3 and the interquartile range [102]

- **Sorting the Data:** First, you need to sort the values in column j from smallest to largest.
- **Identifying Quartiles:**
  - **First Quartile (Q1):** This is the value below which 25% of the data falls. It is the median of the lower half of the data.
  - **Third Quartile (Q3):** This is the value below which 75% of the data falls. It is the median of the upper half of the data.
- **Calculating IQR:** Once you have identified Q1 and Q3, the interquartile range ($\mathrm{IQR}(X_j)$) is computed as:

$$\mathrm{IQR}(X_j) = Q3 - Q1$$

The interquartile range is a useful measure of spread because it is not affected by extreme values or outliers in the data, making it more robust than other measures like the range or standard deviation. It provides a measure of the middle 50% of the data, giving insight into the variability of the dataset without being overly influenced by extreme values. This makes it particularly useful for scaling data in a way that is robust to outliers, as is done in the RobustScaler preprocessing technique.

4.4.4 StandardScaler

StandardScaler standardizes features by removing the mean and scaling to unit variance:

$$X'_{ij} = \frac{X_{ij} - \text{mean}(X_j)}{\text{std}(X_j)}$$

where $\text{mean}(X_j)$ is the mean and $\text{std}(X_j)$ is the standard deviation of column j.

RobustScaler and StandardScaler are suitable for datasets with negative values because they are less influenced by outliers and ensure that the data distribution is centered around zero with a standard deviation of 1, which is desirable for many machine learning algorithms.

4.4.5 Why preprocessing is important

Preprocessing is a very important step in the data analysis pipeline, especially in the context of classification methods. Here are the key reasons why preprocessing is important [103]:

- **Improves data quality:** preprocessing removes noise and irrelevant features, reducing the risk of misleading patterns in the data and improving the overall quality of the dataset.
- **Enhances model accuracy:** By normalizing or scaling the data, preprocessing ensures that features contribute equally to the model, preventing dominance by features with larger magnitudes and leading to more accurate predictions.
- **reduces overfitting:** By selecting relevant features and normalizing data, preprocessing can help in reducing overfitting, where the model learns noise instead of the actual pattern.
- **Speeds up training:** Well-preprocessed data can significantly reduce the computational load and training time, as the model can converge faster when the data is scaled and cleaned properly.
- **Improves interpretability:** Preprocessed data can make it easier to understand the influence of each feature on the classification outcomes, leading to better insights and interpretability of the model.

4.4.6 Train and test data

Our dataset contains 666 samples and like most classification tasks our classes are unbalanced, The total data distribution can be summarized in Table 4.1:

Table 4.1: Distribution of classes in the dataset used

Classes	Train	Test
PD	58	16
D1	70	26
D2	118	42
T1	95	32
T2	62	16
T3	103	28
Total	506	160

The histogram in Figure 4.4 illustrates the significant imbalance in the dataset, with certain fault types, such as D2, being much more frequent than others like T2 and PD. This imbalance can affect the performance and reliability of classification models, as they may become biased towards the more prevalent classes. Therefore, it is important to apply careful considerations and techniques when developing and evaluating our machine learning models to mitigate this issue.

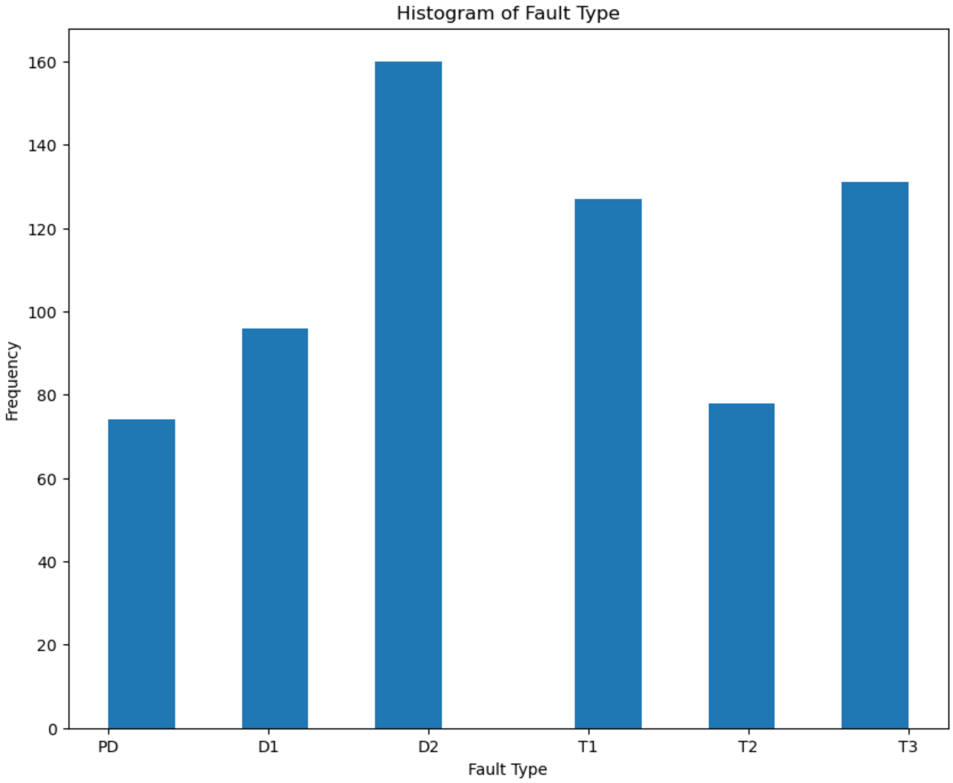


Figure 4.4: Distribution of classes over all dataset

4.4.7 Evaluating the model :

4.4.7.1 Confusion Matrix

The confusion matrix is a summary of predictions which results in a classification problem. Correct and incorrect predictions are highlighted and distributed by class. The results are then compared with the actual values [104].

This matrix allows understanding how the classification model is confused when making predictions. This not only allows knowing what errors are made but especially the type of errors made. Users can analyze them to determine which results indicate how errors are made. We will have a graphical representation of true and false positives and negatives in the form of a matrix as shown in Figure 4.5.

We will thus have the true classes represented in columns and the classes predicted by our models arranged in rows.

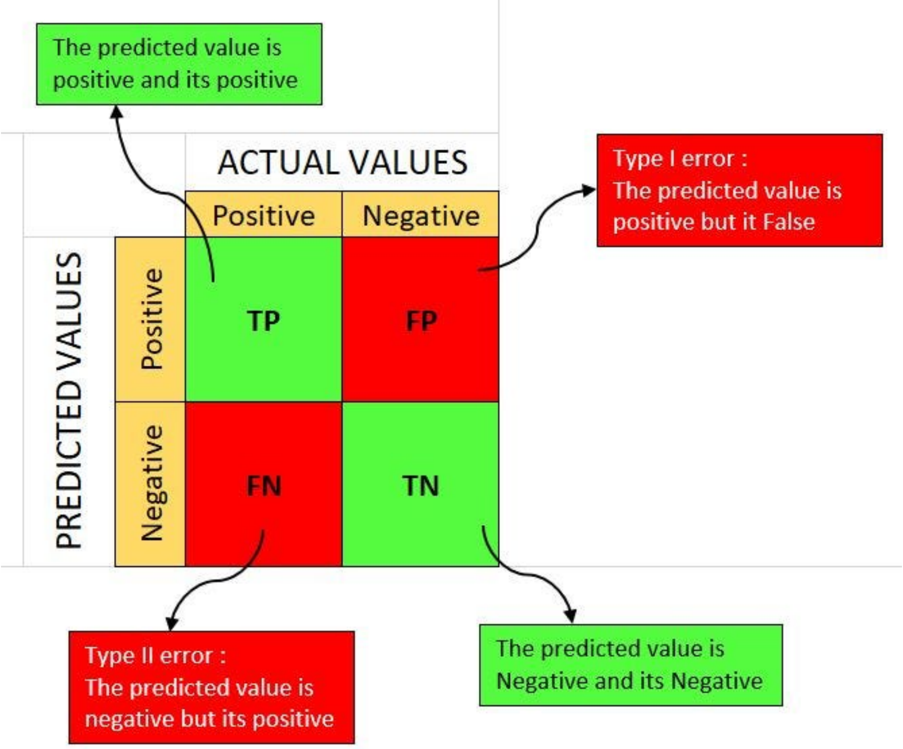

The predicted value is positive and its positive
ACTUAL VALUES
Positive
Negative
Type I error : The predicted value is positive but it False
PREDICTED VALUES
Positive
Negative
TP
FP
FN
TN
Type II error : The predicted value is negative but its positive
The predicted value is Negative and its Negative

Figure 4.5: Confusion matrix [104]

This matrix allows for quick and effective observation of the quality of classification and the classes for which the model's distinction is most difficult.

4.4.7.2 Accuracy

Accuracy is the most commonly used metric for evaluating artificial intelligence models. It is calculated as the ratio of correctly predicted instances to the total instances in the dataset. The formula for accuracy is:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

This metric provides a straightforward measure of the model's overall performance.

4.5 Applying ML algorithms to power transformer DGA Data

This section is the summary for our work, we delve into the practical application of the four machine learning algorithms discussed in chapter 3 on identifying fault types based on dissolved gas analysis data of power transformer oil. We go over these three steps for each algorithm :

1. **Data preparation:**

   Utilizing the eight-entry vectors and various preprocessing techniques discussed in Sections 4.3 and 4.4, we prepare the data for feeding into the models. This step is crucial for ensuring that the input data is clean, consistent, and appropriately formatted for the machine learning algorithms.

2. **Objective function definition for Optuna:** We define the objective function for Optuna, specifying the ranges for the parameters used during the training phase. This step is essential for optimizing the performance of the machine learning models by fine-tuning their hyperparameters.

3. **Performance evaluation:** We assess the accuracy of all combinations of datasets and compare the results to identify the model with the highest accuracy. Additionally, we plot the confusion matrices for these models. This comparative analysis helps us understand the effectiveness of each algorithm under different conditions and select the most suitable model for our application.

By following these steps, we aim to get the max out of SVM, Random Forest, Extra-Tree and XGBoost algorithms in accurately diagnosing faults in power transformers based on DGA data. Through many testing and comparison, we try to demonstrate the potential of these advanced machine learning techniques in enhancing the reliability and efficiency of fault diagnosis in power transformers.

4.5.1 Objective Functions Definition

The following code snippets summarize the range and categories of the parameters used for our three algorithms. These algorithms and their parameters are discussed in detail in Chapter 3:

4.5.1.1 SVM objective function

```
def objective(trial):
 # Define the search space for hyperparameters for the SVM
  c = trial.suggest_float('C', 0.01, 20)
  gamma = trial.suggest_float('gamma', 0.01, 20)
  kernel = trial.suggest_categorical('kernel', ['linear', 'poly', 'rbf', 'sigmoid'])
  degree = trial.suggest_int('degree', 1, 4)
  decision_function = trial.suggest_categorical('decision_function_shape', ['ovr','ovo'])
  random_state = trial.suggest_int('random_state', 1, 100)
```

4.5.1.2 RF objective function

```
Define the objective function for Random Forest
f objective(trial):
  param_space = {
  'n_estimators': trial.suggest_int('n_estimators', 10,60),
  'criterion': trial.suggest_categorical('criterion', ['gini','entropy']),
  'max_depth': trial.suggest_int('max_depth', 2, 20),
  'min_samples_split': trial.suggest_int('min_samples_split', 2, 20),
  'min_samples_leaf': trial.suggest_int('min_samples_leaf', 3, 20),
  'max_features': trial.suggest_float('max_features', 0.1,1),
  'random_state': trial.suggest_int('random_state', 1,100),}
```

4.5.1.3 Extra-Tree objective function

```
  # Define the objective function for Extra-Tree
  def objective(trial):
      param_space = {
          'n_estimators': trial.suggest_int('n_estimators', 10, 60),
          'criterion': trial.suggest_categorical('criterion', ['gini', 'entropy']),
          'max_depth': trial.suggest_int('max_depth', 2, 20),
          'max_features': trial.suggest_float('max_features', 0.1,1),
          'min_samples_split': trial.suggest_int('min_samples_split', 2, 20),
          'min_samples_leaf': trial.suggest_int('min_samples_leaf', 3, 20),
          'random_state': trial.suggest_int('random_state', 1, 100),}
```

Extra-Trees (Extremely Randomized Trees) and Random Forest differ mainly in their splitting criteria and bootstrapping approach as discussed in their comparison in Table 3.1.

4.5.1.4 XGBoost objective function

```
Define the objective function for XGBoost
f objective(trial):
  # Define the parameter space to search over
  param_space = {
      'n_estimators': trial.suggest_int('n_estimators', 1,50),
      'max_depth': trial.suggest_int('max_depth', 2,10),
      'min_child_weight': trial.suggest_int('min_child_weight', 2,10),
      'gamma': trial.suggest_float('gamma', 1, 10),
      'subsample': trial.suggest_float('subsample', 0.1, 1.0),
      'colsample_bytree': trial.suggest_float('colsample_bytree', 0.1,1),
      'max_leaves': trial.suggest_int('max_leaves', 1,20),
      'lambda': trial.suggest_float('lambda', 0.5, 3),
      'alpha': trial.suggest_float('alpha', 0.5,  1),
      'learning_rate': trial.suggest_float('learning_rate', 0.1, 0.9),
      'random_state': trial.suggest_int('random_state', 1,100),}
```

4.5.1.4.1 Parameters explained:

- **C, Gamma, kernel and degree** are SVM parameters, Explained in details in section 3.2

- **n_estimators**: The number of boosting rounds or trees to build. Increasing this parameter can improve accuracy but also increases computation time.

- **max_depth**: The maximum depth of each tree. Higher values allow the model to capture more complex patterns but can also lead to overfitting.

- **min_child_weight**: The minimum sum of instance weight (hessian) needed in a child. This parameter controls overfitting by preventing the model from learning from overly specific patterns in the training data. Higher values make the algorithm more conservative by avoiding the creation of nodes that correspond to fewer samples.

- **min_samples_split**: The minimum number of samples required to split an internal node. This parameter helps control overfitting by ensuring that each split occurs only if there are enough samples to justify the creation of a new branch in the tree.

- **min_samples_leaf**: The minimum number of samples required to be at a leaf node. This parameter prevents the model from learning overly specific patterns by ensuring that each leaf node represents a sufficient number of samples.

- **criterion**: The function used to measure the quality of a split. Common criteria include *gini* for the Gini impurity and *entropy* for the information gain. This parameter determines how the decision tree algorithm splits nodes to maximize information gain or minimize impurity.

- **gamma**: The minimum loss reduction required to make a further partition on a leaf node of the tree. It serves as a regularization parameter and makes the algorithm more conservative by only allowing splits that significantly reduce the loss.

- **subsample**: The fraction of samples to be used for fitting individual base learners. This parameter introduces randomness into the model building process, helping to prevent overfitting.

- **colsample_bytree**: The fraction of features to be used for building each tree. Like *subsample*, this parameter introduces randomness and helps to prevent overfitting by ensuring that the model does not rely too heavily on any one feature.

- **max_leaves**: The maximum number of leaves in a tree. This parameter controls the complexity of the tree and works in conjunction with *max_depth* to prevent the model from becoming too complex.

- **lambda**: The L2 regularization term on weights. Increasing this value makes the algorithm more conservative and helps prevent overfitting by penalizing large weights.

- **alpha**: The L1 regularization term on weights. Similar to *lambda*, this parameter helps control the model complexity and prevents overfitting by adding a penalty for large weights.

- **learning_rate**: The step size shrinkage used in updates to prevent overfitting. Smaller values make the model more robust to overfitting by shrinking the impact of each individual tree, though this requires more boosting rounds to achieve the same level of performance.

- **random_state**: The seed used by the random number generator. Setting this parameter ensures the reproducibility of the model training process by generating the same sequence of random numbers.

4.5.1.5 Acccuracy evaluation

The diagnostic accuracy of the four algorithm implementations with eight input vectors and the four preprocessing techniques discussed in Sections 4.3 and 4.4 is summarized in the following tables:

Table 4.2: Accuracy evaluation for SVM algorithm

SVM	ppm	IEC	Roger's	Doen	Triangle	Pentagon	Mixed	Tri-Pent
NormalizeScaler	93.75%	70.625%	78.125%	73.75%	76.875%	77.5%	78.75%	89.375%
MinMaxScaler	60%	25.5%	26.875%	27.5%	78.75%	88.75%	27.5%	**95%**
StandardScaler	83.125%	27.5%	25.5%	27.5%	79.375%	88.125%	26.875%	**95%**
RobustScaler	76.875%	70.625%	78.125	73.125%	76.875	78.125%	78.75%	94.375%

Table 4.3: Accuracy evaluation for RF algorithm

RF	ppm	IEC	Roger's	Doen	Triangle	Pentagon	Mixed	Tri-Pent
NormalizeScaler	**95.625%**	76.875%	79.375%	80%	80%	80.625%	80.625%	91.875%
MinMaxScaler	94.375%	89.375%	89.375%	91.25%	80.625%	88.125%	90%	91.875%
StandardScaler	94.375%	88.75%	89.375%	91.25%	81.25%	88.75%	93.125%	91.25%
RobustScaler	95%	88.125%	89.375%	91.25%	80%	88.75%	91.25%	93.75%

Table 4.4: Accuracy evaluation for Extra-Tree algorithm

Extra-Tree	ppm	IEC	Roger's	Doen	Triangle	Pentagon	Mixed	Tri-Pent
NormalizeScaler	95%	73.75%	76.25%	73.75%	78.75%	79.375%	78.125%	**95.625%**
MinMaxScaler	89.375%	70.625%	62.25%	76.25%	79.375%	86.25%	82.5%	90%
StandardScaler	90.625%	68.75%	68.75%	71.25%	78.125%	85.625%	80.625%	90.625%
RobustScaler	90%	73.75%	75%	73.125%	77.5%	84.375%	74.375%	90.625%

Table 4.5: Accuracy evaluation for XGB algorithm

XGB	ppm	IEC	Roger's	Doen	Triangle	Pentagon	Mixed	Tri-Pent
NormalizeScaler	**95.62%**	72.5%	73.125%	81.875%	78.75%	80%	81.25%	94.375%
MinMaxScaler	86.875%	91.25%	88.125%	91.25%	87.5%	87.5%	87.5%	86.875%
StandardScaler	88.75%	86.25%	89.375%	91.25%	77.5%	87.5%	91.25%	87.5%
RobustScaler	88.75%	87.5%	88.75%	90.625%	77.5%	90%	90%	87.5%

4.5.2 Results interpretation

1. **Impact of input vectors on accuracy:**

   - Different input vectors have a significant impact on the performance of the algorithms. Vectors such as 'Tri-Pent' and 'ppm' consistently gave higher accuracy in most algorithms and preprocessing techniques.
   - The 'Tri-Pent' vector, combining features from both Duval triangle and pentagon methods, achieved some of the highest diagnostic accuracy scores especially with RF and XGBoost.
   - In contrast, vectors based on ratios such as 'IEC' and 'Roger's' generally resulted in lower accuracy, particularly for SVM, due to the sensitivity of SVM to feature scaling and the potential non-linearity introduced by ratios.
   - The black lines on each bar on Figure 4.6 represent the confidence intervals (standard deviation or standard error) of the accuracy measurements, indicating the variability or uncertainty around the mean accuracy.
   - Figure 4.6 provides a comparison of accuracy across all algorithms and input vectors, illustrating the impact of different input vectors on overall performance.

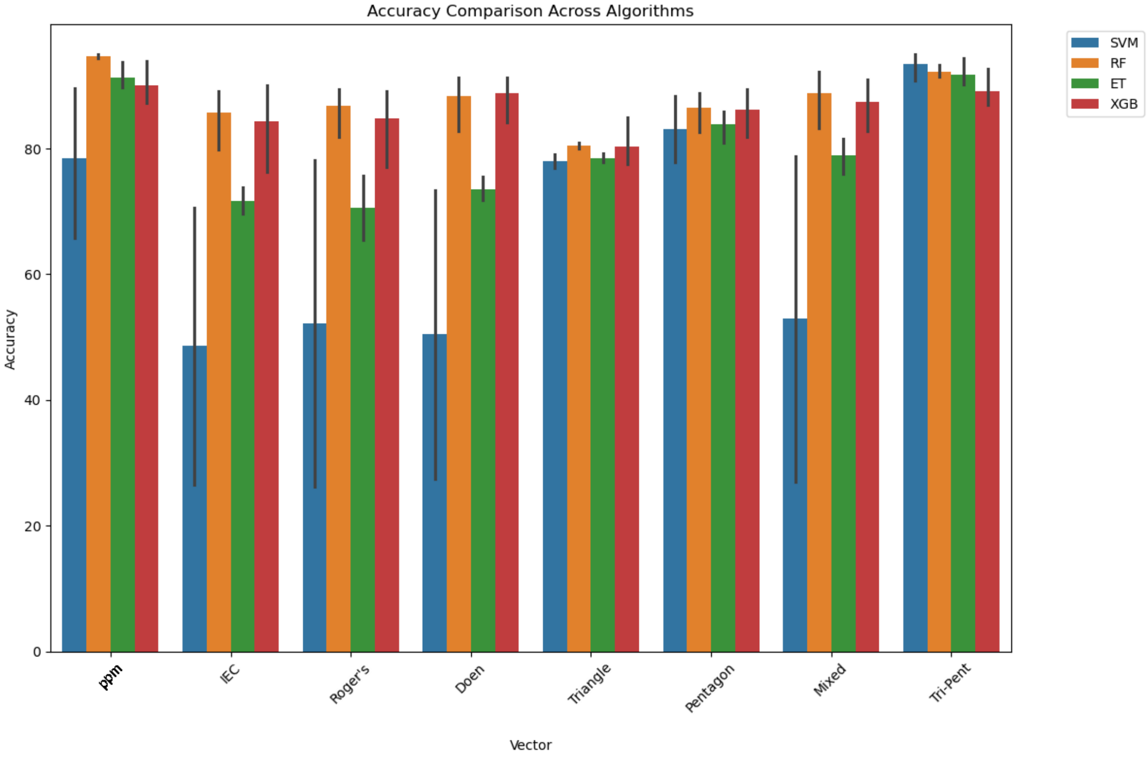


Figure 4.6: Accuracy comparison across algorithms and input vectors

2. **Impact of preprocessing techniques on accuracy:**

   - Preprocessing techniques generally improve model performance by transforming the data, which helps algorithms converge more efficiently and make more accurate predictions.
   - Standardization and normalization ensure that the features are on a similar scale, which is important for algorithms like SVM that are sensitive to feature scales.
   - Robust Scaling, by using the interquartile range, reduces the influence of outliers and generally leads to more stable models. However, in our study, it did not achieve the highest accuracy compared to other preprocessing techniques.
   - Figure 4.10 shows the average accuracy of each preprocessing technique across all vectors and algorithms. It is evident that:
     - **NormalizeScaler** achieved consistently high accuracy for all algorithms, particularly for RF and XGB, indicating its effectiveness in standardizing the data for these models.
     - **MinMaxScaler** resulted in a notable improvement in accuracy for RF and XGB, but it performed poorly for SVM, demonstrating that this scaler might be better suited for certain algorithms.
     - **StandardScaler** showed strong performance for RF and XGB, but less so for SVM, highlighting the variability in how different algorithms respond to this scaler.
     - **RobustScaler** provided stable but not the highest accuracy across most algorithms, suggesting that while it is effective in handling outliers, it may not always lead to the best accuracy.

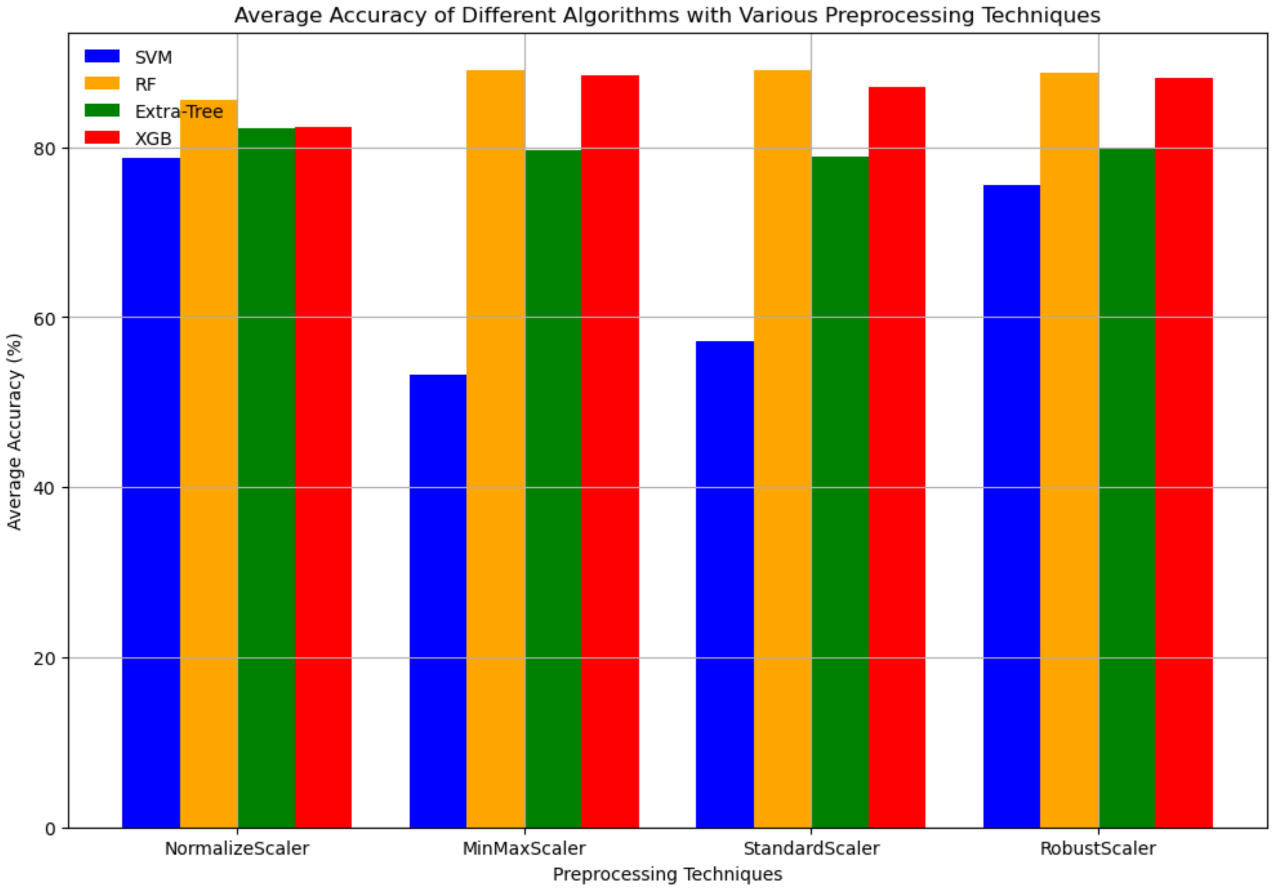


Figure 4.7: Average accuracy for prepossessing techniques

3. **Performance of different methods:**

   - The primary goal of testing different methods (SVM, Tree based methods (Random Forest, Extra Trees, and XGBoost), preprocessing techniques and input vectors is to identify the best combination that maximizes the diagnostic accuracy.
   - RF and XGBoost Algorithms generally perform better across most input vectors and preprocessing techniques compared to SVM and Extra Trees. This is evident from the high accuracy scores observed in the bar plot comparison (Figure 4.6).
   - The box plot of accuracy distribution (Figure 4.8) shows that RF and XGBoost had the best overall performance when considering the various input vectors and preprocessing techniques.
   - SVM algorithm exhibited higher variance in accuracy, indicating its performance is more dependent on the choice of preprocessing and input vectors.
   - RF and XGBoost demonstrated more consistent performance, highlighting their robustness across different preprocessing techniques and vectors.

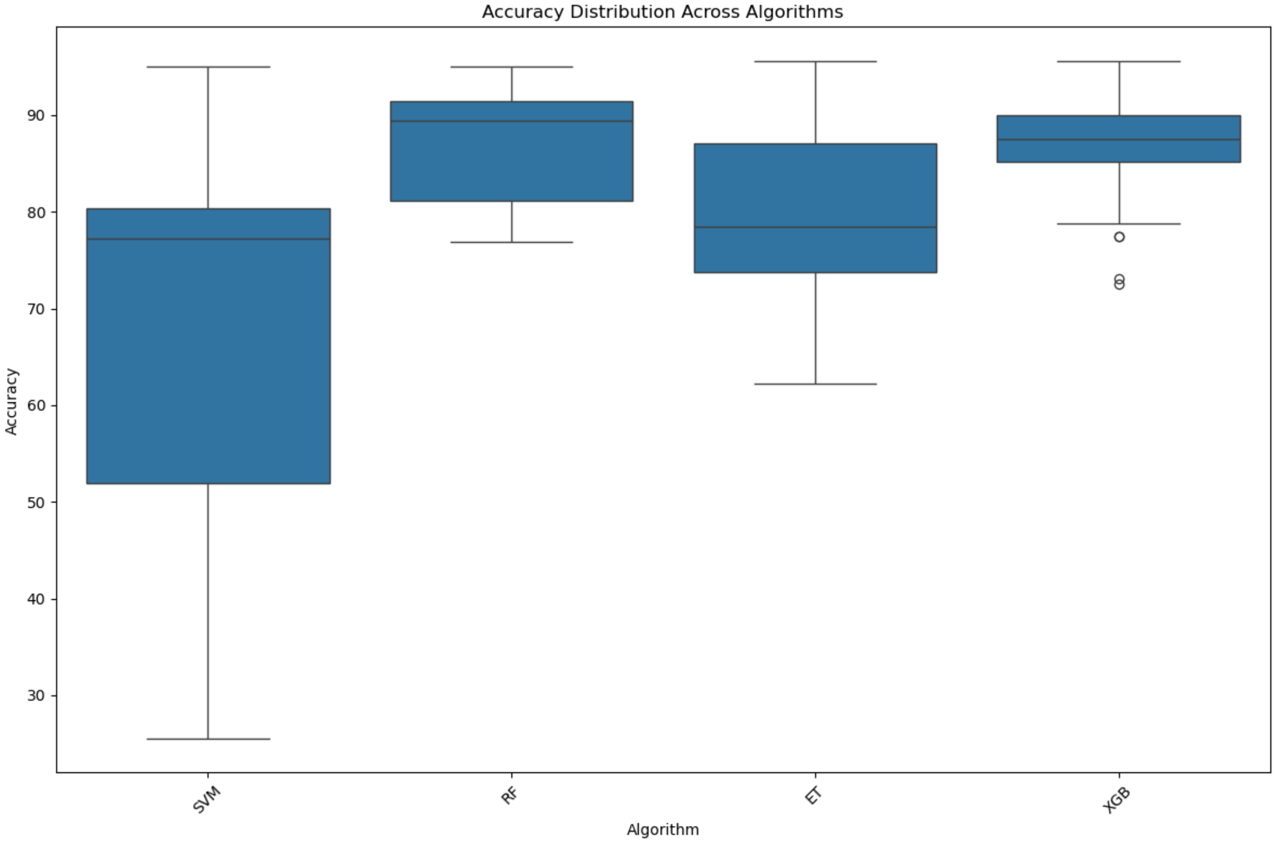


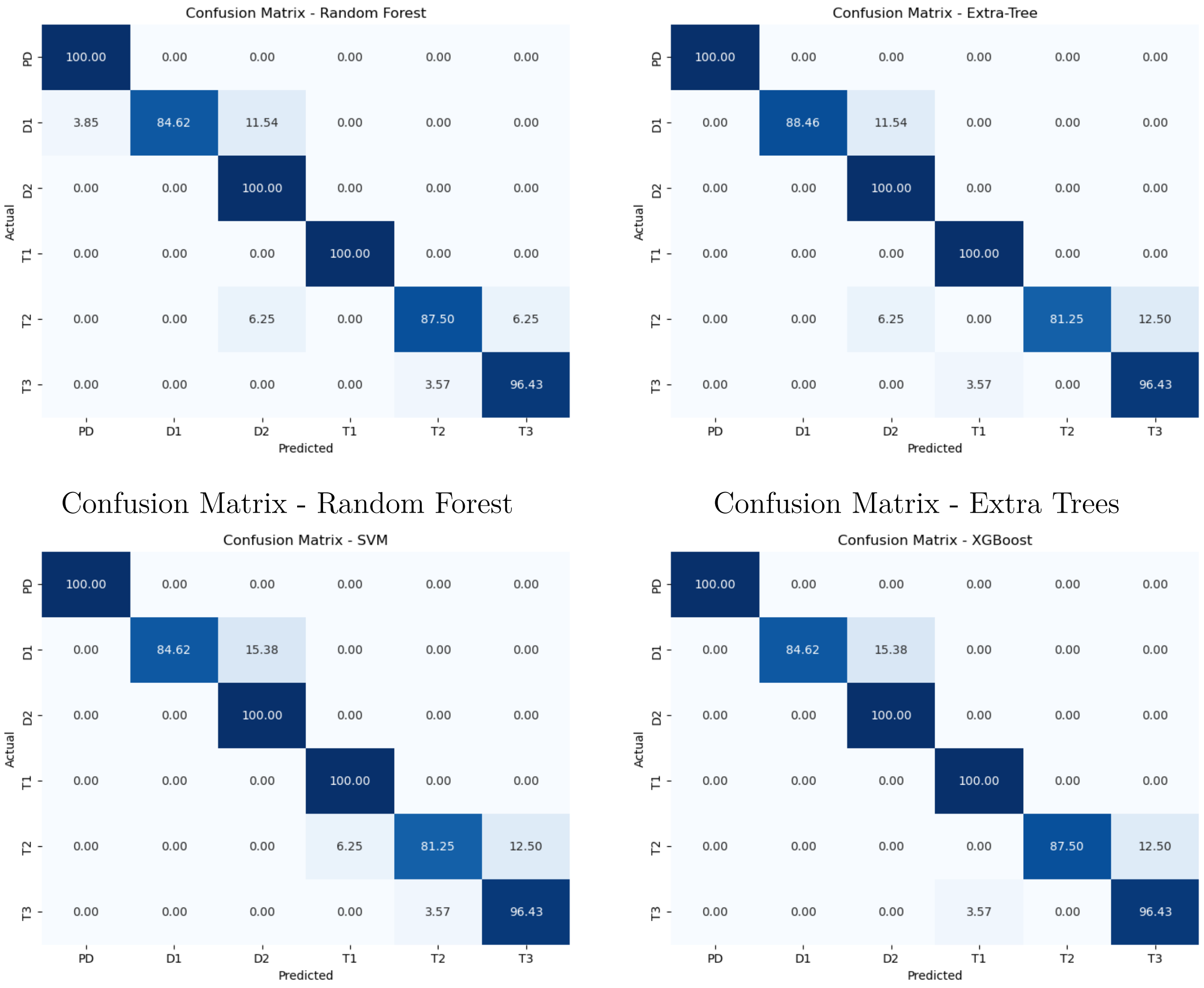
Figure 4.8: Accuracy distribution across algorithms

4. **Confusion Matrices analysis:**

   - For the fault types PD, D2, and T1, all samples have been correctly classified across all algorithms, with 16 out of 16 for PD, 42 out of 42 for D2, and 32 out of 32 for T1.
   - For the fault type T3, one sample was consistently misclassified as T2 across all methods, resulting in 27 out of 28 correct classifications.

- For the fault type T2, the RF and XGBoost algorithms misclassified two samples, resulting in 14 out of 16 correct classifications. In contrast, the Extra-Tree and SVM algorithms misclassified three samples, resulting in 13 out of 16 correct classifications.
- For the fault type D1, the confusion matrices for the best-performing models illustrate common misclassifications. The Extra-Tree algorithm misclassified 3 samples, while the other algorithms each misclassified 4 samples.
- Fault type D1 consistently shows lower accuracy across all methods, indicating that it is often confused with other fault types. This similarity in misclassification patterns suggests that stacking these models would not provide significant benefits. Therefore, ensemble stacking was not considered a primary strategy, as the models tend to make similar errors, reducing the potential gain from combining them.
- Figure 4.9 shows the confusion matrices for the **best performing** Random forest, Extra Trees, SVM, and XGBoost models respectively, highlighting the consistent struggle with fault type D1.

Note that the codes 0, 1, 2, 3, 4, 5 represent PD, D1, D2, T1, T2, T3 respectively.



Confusion Matrix - Random Forest

Confusion Matrix - Extra Trees

Confusion Matrix - SVM

Confusion Matrix - XGBoost

Figure 4.9: Confusion Matrices for different models (random forest, extra trees, SVM, XGBoost)

4.6 Accuracy improvement with decision tree principle

Inspired by the work of Kherif et al. (2021) [21], we opted for a custom-built decision tree to enhance the accuracy of fault classification in transformer data. The strategy for constructing this decision tree is outlined below.

4.6.1 Decision tree construction strategy

The process begins by dividing the faults into two primary categories: thermal faults and electrical faults. This initial categorization simplifies the subsequent classification steps by narrowing down the fault types at each level. For both electrical and thermal faults, we explored all possible nodes to find the optimal way to classify these faults. This approach ensures that each type of fault is isolated in a structured manner, allowing for more precise identification at each level of the decision tree.

Given that we have six faults (PD, D1, D2, T1, T2, T3) that can be further classified in up to three levels, we considered various nodes at each division step. Electrical faults (PD, D1, D2) were split in all possible ways: ({PD,D1,D2}) ,({PD}, {D1, D2}), ({D1}, {PD, D2}), and ({D2}, {PD, D1}). Similarly, thermal faults (T1, T2, T3) were split into : ({T1,T2,T3}), ({T1}, {T2, T3}), ({T2}, {T1, T3}), and ({T3}, {T1, T2}).

In total, we constructed up to eight different decision trees, each representing a unique combination of splits at each level:

Total possible nodes = (4 different electrical faults nodes) + (4 different thermal faults nodes) = 8

4.6.2 Input vectors, preprocessing Techniques, and algorithms

4.6.2.1 Input vectors and preprocessing techniques

For the input vectors, we prioritized the ppm and the triangle-pentagon dataset, as these achieved the highest accuracy across all algorithms. Preprocessing was limited to StandardScaler and NormalizeScaler, selected based on their superior performance in prior analyses. This approach was adopted to efficiently manage the extensive testing required, involving 8 nodes, 2 input vectors (ppm and Tri-Pent datasets), 2 preprocessing techniques (StandardScaler and NormalizeScaler) and 2 ML methods (RF and XGB). Consequently, we will be testing 64 models. This is in contrast to the 128 tests previously conducted in Section 4.5.1.5. This strategy facilitated improved accuracy within a shorter timeframe. Figure 4.11 provides a rationale for the exclusive use of these preprocessing techniques.

The bar plot compares the differences in accuracy between NormalizeScaler/StandardScaler and MinMax/RobustScaler for RF and XGB models on "ppm" and "Tri-Pent" vectors. Positive values indicate that NormalizeScaler and StandardScaler generally outperform MinMaxScaler and RobustScaler. This suggests that NormalizeScaler and StandardScaler provide more consistent and superior performance, justifying their selection for preprocessing. These observations support the conclusion that NormalizeScaler and StandardScaler are preferable choices for preprocessing compared to MinMaxScaler and RobustScaler specifically for the ppm and triangle-pentagon datasets.

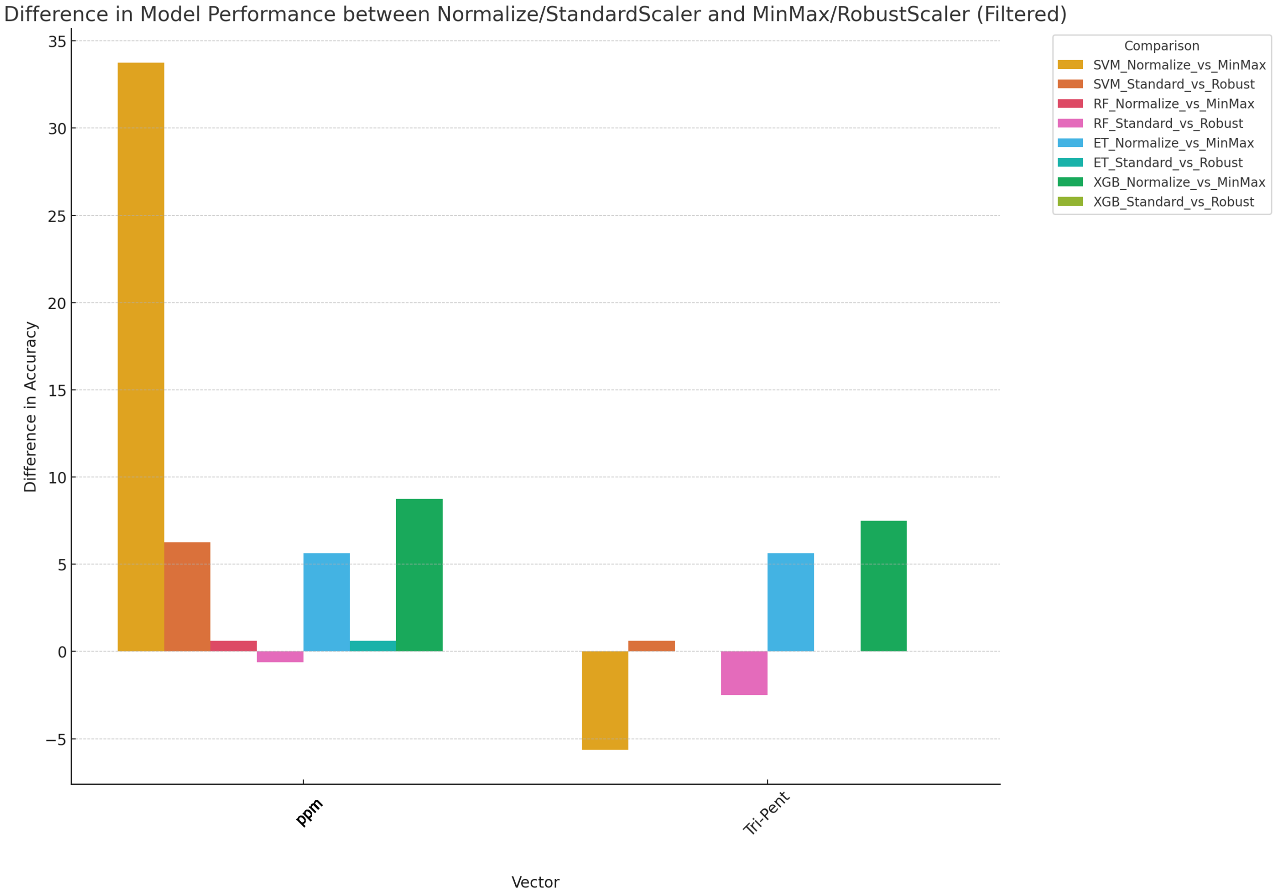


Figure 4.10: Difference in performance when using Normalize/StandardScaler and MinMax/RobustScaler

4.6.2.2 Algorithms used

After constructing this detailed decision tree and in order to optimize the search space, the Extra-Trees method was dropped because, at its core, it is very similar to Random Forest. The heatmap in Figure 4.11 illustrates that RF generally performed better in every single combination except for one spot. Given this consistent superiority, RF was selected over Extra-Trees for further analysis.

SVM exhibited higher variance in accuracy, indicating its performance is more dependent on the choice of preprocessing and input vectors. Based on this analysis and the box plot shown in Figure 4.8, SVM was also dropped. Only XGBoost and RF were retained for further analysis due to their more consistent and higher accuracy distributions.

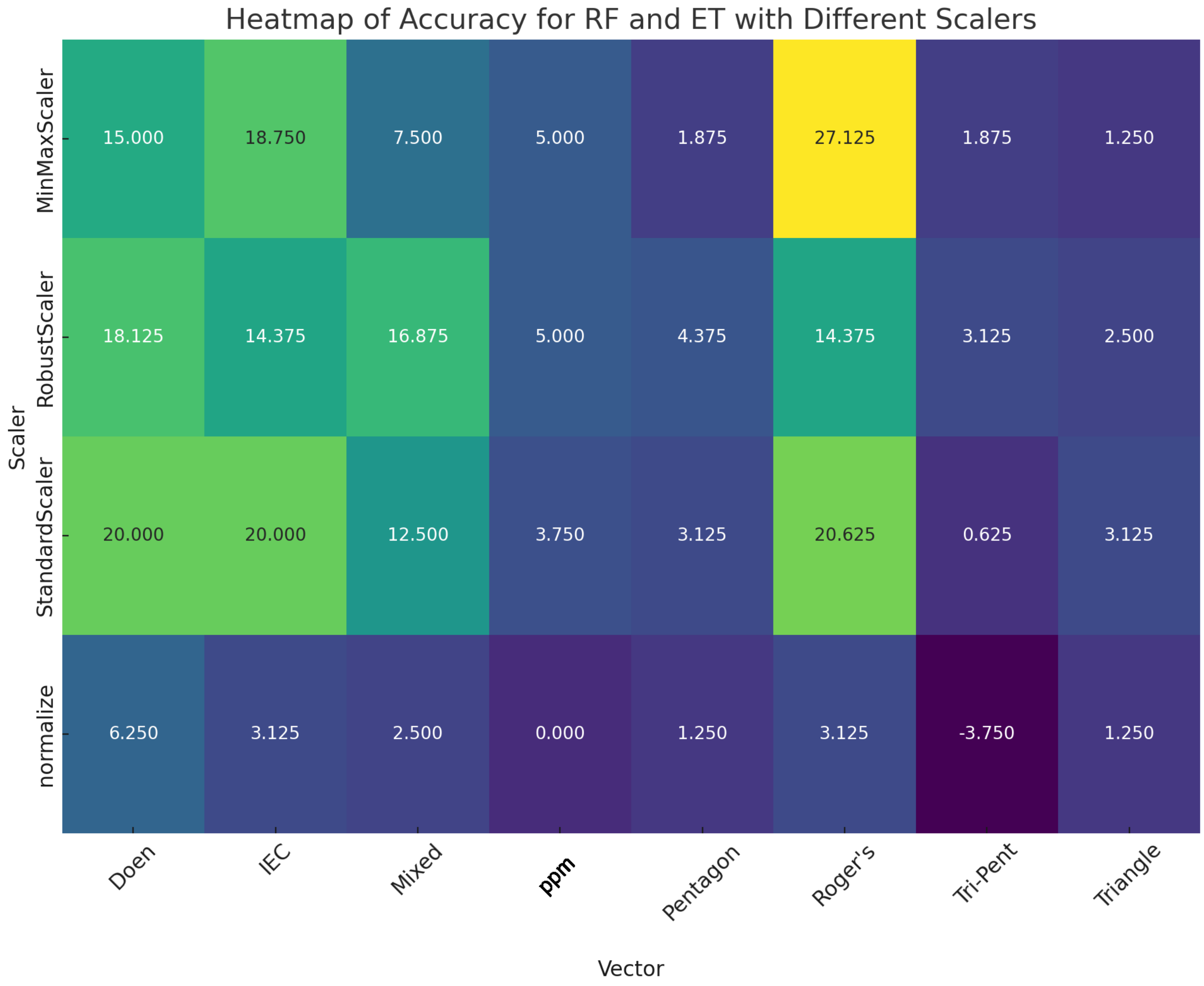


Figure 4.11: Heatmap of accuracy differences between RF and ET with different scalers across various vectors

4.6.3 Building the decision tree

We will build the decision tree level by level, starting by splitting faults into thermal and electrical faults and then going deeper as explained in Section 4.6.1.

4.6.3.1 Level 1: Identifying thermal and electrical faults

At this level, we categorize the faults into two types: D (electrical) and T (thermal). Training at this level was straightforward, as all input vectors, preprocessing techniques, and algorithms achieved perfect accuracy (100%) as shown in Table 4.6. This result is expected since tree-based methods often perform better in binary classification compared to multi-class classification due to the lower computational complexity and reduced potential for errors in binary classifiers [105].

Table 4.6: Accuracy of different models for Level 1 fault classification (thermal vs. electrical)

Preprocessor	**NormalizeScaler**		**StandardScaler**	
Input Vector	**ppm**	**Tri-penta**	**ppm**	**Tri-penta**
RF	100%	100%	100%	100%
XGB	100%	100%	100%	100%

4.6.3.2 Level 2: Separated electrical and thermal faults nodes

4.6.3.2.1 Electrical fault nodes : Testing the four different trees possible for electrical faults (PD, D1, D2) and all possible division ways PD, D1, D2, D1, PD, D2, and D2, PD, D1, we found that the combination PD, D1, D2 achieved 100% accuracy, as shown in Table 4.7.

Table 4.7: Accuracy of different models for various splits of electrical faults

	Accuracy		
Split	**XGB**	**RF**	**(Input vector, preprocessing)**
{PD,D1,D2}	95.23%	95.23%	(ppm, NormalizeScaler)
	95.23%	95.23%	(Tri-penta, StandardScaler)
{PD,{D1,D2}}	**100%**	**100%**	(ppm, NormalizeScaler)
	100%	**100%**	(Tri-penta, StandardScaler)
{D1,{PD,D2}}	**95.23%**	94.07%	(ppm, NormalizeScaler)
	85.7%	86.9%	(Tri-penta, StandardScaler)
{D2,{PD,D1}}	97.6%	**98.8%**	(ppm, NormalizeScaler)
	95.23%	92.8%	(Tri-penta, StandardScaler)

4.6.3.2.2 Thermal fault nodes : Testing the four different trees possible for thermal faults (T1, T2, T3) and all possible divisions (T1, T2, T3, T2, T1, T3, and T3, T1, T2), we found that the combination T1, T2, T3 reached 100% accuracy, as shown in Table 4.8.

Table 4.8: Accuracy of different models on different nodes for thermal faults

	Accuracy		
Split	**XGB**	**RF**	**(Input vector, preprocessing)**
{T1,T2,T3}	**97.63%**	**97.63%**	(ppm, NormalizeScaler)
	93.42%	96.05%	(Tri-penta, StandardScaler)
{T1,{T2,T3}}	**100%**	**100%**	(ppm, NormalizeScaler)
	100%	**100%**	(Tri-penta, StandardScaler)
{T2,{T1,T3}}	**96.05%**	96.05%	(ppm, NormalizeScaler)
	85.52%	96.05%	(Tri-penta, StandardScaler)
{T3,{T1,T2}}	97.36%	**98.68%**	(ppm, NormalizeScaler)
	97.36%	**98.68%**	(Tri-penta, StandardScaler)

4.6.3.3 Level 3: Further faults classification

4.6.3.3.1 Electrical fault side : Given the best accuracy achieved with the node {PD}, {D1, D2}, we proceeded with it at the next level. The results are shown in Table 4.9:

Table 4.9: Accuracy of different models on different nodes for the D_1, D_2 node

{D1,D2}	**NormalizeScaler**	**StandardScaler**
Input Vector	**ppm**	**Tri-penta**
RF	97.07%	**98.52%**
XGB	95.58%	94.17%

4.6.3.3.2 Thermal fault side Using the same strategy as for the electrical faults, we achieved 100% accuracy for the thermal faults, as shown in Table 4.10:

Table 4.10: Accuracy of different Models on different nodes for the T_2, T_3 node

{T2,T3}	**NormalizeScaler**	**StandardScaler**
Input Vector	**ppm**	**Tri-penta**
RF	95.45%	97.72%
XGB	93.81%	**100%**

Despite extensive experimentation with all mentioned algorithms and additional methods, the highest achievable accuracy was 98.52%. While perfect accuracy of 100% might suggest overfitting, the current accuracy level is considered excellent. Achieving 100% accuracy in this case could indeed be an indication of overfitting, especially given the limitations of the dataset. Without access to additional data, it is challenging to definitively determine the model's performance beyond our test data. Therefore, while the model demonstrates strong performance on the available test data, caution is warranted in interpreting these results, as there may be a risk of overfitting. Nonetheless, the results are promising and indicate a high level of accuracy.

4.6.4 Final decision tree

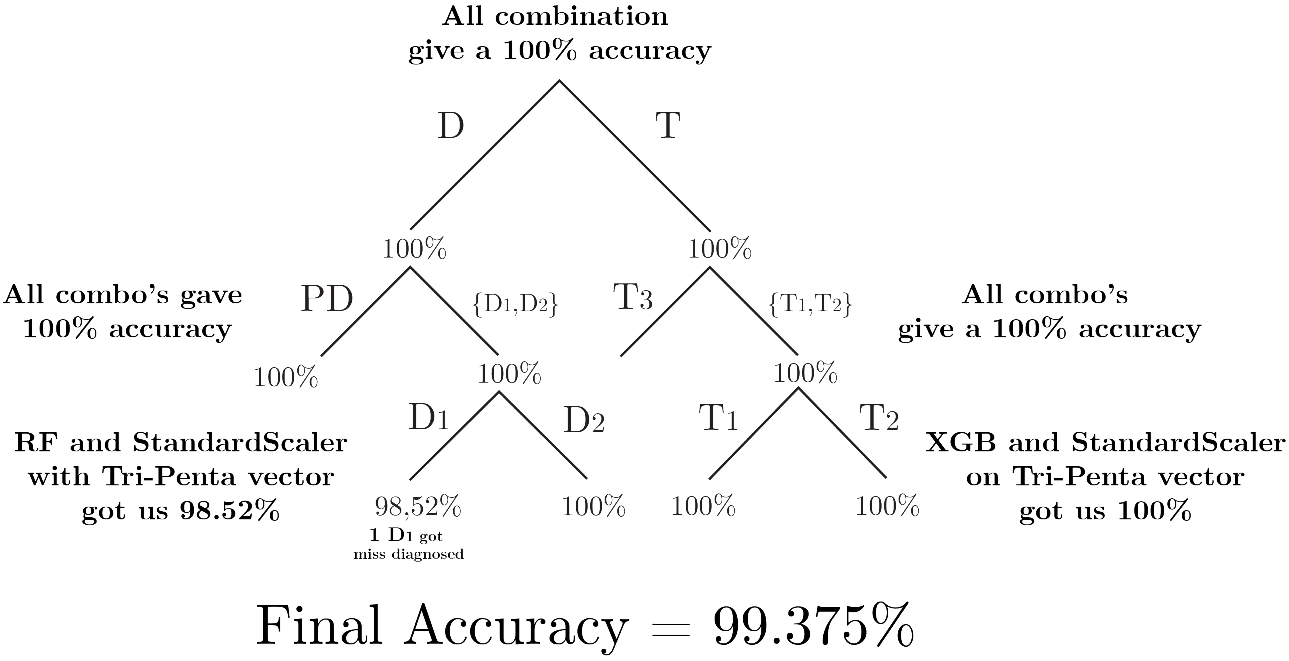

All combination give a 100% accuracy
D
T
100%
100%
All combo's gave 100% accuracy
PD
{D1,D2}
T3
{T1,T2}
All combo's give a 100% accuracy
100%
100%
100%
D1
D2
T1
T2
RF and StandardScaler with Tri-Penta vector got us 98.52%
XGB and StandardScaler on Tri-Penta vector got us 100%
98,52%
1 D1 got miss diagnosed
100%
100%
100%
Final Accuracy = 99.375%

Figure 4.12: The most accurate decision tree combination

As shown in Figure 4.12, we can see that the model misses only one D1 sample. This supports the analysis of the confusion matrices in Figure 4.9 as they all struggled with the D1 faults.

4.7 Comparative analysis of our approach and alternative methods

It wouldn't be fair to compare our work with other studies that used different test data sets, as the results would not be directly comparable. Therefore, to ensure a fair and accurate comparison, All methods compared are on the same set of 160 test data points. The tables 4.11 and 4.12 below summarizes the accuracy's of different algorithms and approaces, including the conventional methods discussed in Chapter 2. This comprehensive comparison highlights the performance of our approach's in relation to traditional diagnostic techniques as well as various machine learning models.

Table 4.11: Performance of conventional diagnostic methods on test data

	ACT	Duval Triangle	IEC Code-60599	Rogers 4 Ratios	Modified IEC Code	Modified Rogers 4 Ratios
PD	16	93.75	43.75	37.5	8.5	75
D1	26	76.92	26.92	0	76.92	84.61
D2	42	85.71	66.66	73.80	90.47	88.09
T1	32	65.62	53.125	90.62	93.75	93.75
T2	16	50	68.75	25	93.75	93.75
T3	28	100	82.14	57.14	89.28	89.28
All	**160**	**80**	**58.12**	**53.75**	**88.75**	**88.12**

Table 4.12: Comparison of our approaches with other machine learning techniques

	KNN	ANN-PSO	RF	ET	XGB	SVM	DT (KNN)	DT (RF & XGB)
PD	100	100	100	100	100	100	100	100
D1	80.77	98.52	84.62	88.46	84.62	84.62	80.77	98.52
D2	97.62	100	100	100	100	100	97.62	100
T1	95.18	100	100	100	100	100	98.93	100
T2	81.25	100	87.5	81.25	87.5	81.25	81.25	100
T3	96.43	100	96.43	96.43	96.43	96.43	96.43	100
All	**91.875**	**99.375**	**95.625**	**95.625**	**95.526**	**95**	**92.5**	**99.375**

The accuracy's presented in the tables highlight the effectiveness of each method in diagnosing the test data. It is important to interpret these results in the context of the specific characteristics and complexities of the test data used. Variations in accuracy can be attributed to the different strengths and weaknesses of each algorithm, as well as their ability to handle the specific nuances of the data. Our hybrid approach, which combines multiple techniques, demonstrates a significant improvement in accuracy in general matching the ANN-PSO (Particle Swarm Optimization), underscoring the potential benefits of leveraging ensemble methods in diagnostic applications.

4.8 conclusion

In this chapter, we aimed to improve the accuracy of power transformer oil diagnosis based on dissolved gas analysis data using machine learning algorithms. By employing Support Vector Machines and ensemble learning methods, including Random Forest , Extra-Trees, and XGBoost, we used various input vectors and preprocessing techniques. Our initial experiments achieved an accuracy of 95.62% with RF, ET, and XGB and the confusion matrix analysis revealed common miss-classification patterns among these models, particularly with D1 faults.

To enhance accuracy further, we developed a custom decision tree strategy. This strategy involved narrowing down the best-performing input vectors, preprocessing techniques, and algorithms based on our initial findings. By constructing a detailed decision tree and leveraging the optimal combinations, we achieved a diagnostic accuracy of 99.375%, with only one misclassified sample (a D1 fault diagnosed as a D2 fault).

Our results highlight the potential of a hybrid approach that combines multiple machine learning algorithms with a well-structured decision tree. This approach demonstrates significant improvements in diagnostic accuracy of power transformers oil, underscoring the benefits of leveraging ensemble methods and customized strategies in fault diagnosis applications.

General Conclusion

4.9 General Conclusion

The lifespan of power transformers is closely linked to the condition of the insulating oil used. This oil degrades due to faults resulting from overheating, electrical arcs, low or high energy discharges, and other factors. Such degradation can lead to transformer failures or malfunctions. Early detection of these faults is a critical step in preventing such failures. Therefore, in this study, we focused on the application and improvement of fault diagnosis using dissolved gas analysis technique, considered one of the fastest and most economical techniques widely used.

We began this dissertation with extensive bibliographic research, divided into two chapters. The first chapter is devoted to general information about power transformer oils. Through this chapter, we learned that such transformers are essential components in the operation of electrical networks. Their lifespans essentially depend on their insulation systems. Degradation of these systems can lead to significant economic losses due to the high repair costs and the unavailability of electrical service they cause. Special attention was given to insulating oil.

The dielectric properties of the oil degrade under the influence of contamination, moisture, and aging. Aging is primarily caused by oxidation (presence of oxygen), hydrolysis, increased temperature, and the presence of gaseous inclusions. Characterized by a high electric field, these inclusions can host micro-discharges that decompose the oil, generating gases.

Various oil analyses can diagnose the internal state of the power transformer. The most popular analyzes are the electrical, physical, and chemical properties of the oil, and dissolved gas analysis, which interprets their proportions in the oil under electrical and thermal stress. This latter analysis is advantageous due to its low cost relative to that of interruption and repair. Moreover, it uses small amounts of gas and is non-destructive. Furthermore, it can be applied to transformers in service (online mode) or in the laboratory on collected samples (offline mode).

The second chapter is dedicated to presenting the three most common categories of conventional methods. The first uses the concentrations of dissolved gases individually, linking each gas to one or two faults. Gas concentrations can be expressed in ppm (IEEE Standard C57.104 or TDCG) or in percentages (key gas method). Individual gas methods are simple to apply. However, they are not very reliable given their modest fault classification rates.

The second category includes the Dornenburg, Rogers, and IEC 60599 methods. These methods use the ratios of gas concentrations. Given their acceptable classification rates, these methods are fairly reliable. However, they require significant gas concentrations to be valid. Additionally, they can present uninterpretable cases, which limits their use.

The third category comprises graphical methods (triangle, pentagon, and their variants) by Duval, which use the percentages of concentration ratios. These methods are robust and reliable. Although their classification rates are appreciable, they could be improved to obtain a more refined diagnosis.

In order to overcome the limitations of individual gas methods and concentration ratios, and to further improve the classification rate of graphical methods, the use of more powerful diagnostic systems such as artificial intelligence techniques (machine learning) is essential. In the third chapter, we presented different machine learning algorithms including Support Vector Machines (SVM), and another branch of machine learning called ensemble learning where we used Random Forest, Extra Trees, and Extreme Gradient Boosting (XGBoost).

In Chapter 4, we applied the discussed intelligent methods to identify fault types based on dissolved gas analysis of power transformer oil. We utilized an 8 input vector and various preprocessing techniques for the data. Hyper-parameter tuning for all algorithms was conducted using Optuna. We introduced our hybrid approach, which integrates a decision tree principle with XGBoost and random forest algorithms and finally compared the results with conventional methods and other AI based works.

The eight different input vectors considered were designed to effectively represent the concentrations and ratios of dissolved gases in order to achieve higher accuracy. These vectors included individual gas concentrations (in ppm), ratios as per IEC, Rogers, and Dornenburg methods, graphical coordinates from Duval triangle and pentagon, and combinations of these two last approaches.

Preprocessing techniques such as Normalize, MinMaxScaler, RobustScaler, and StandardScaler were employed to prepare and clean the data. These techniques are important for improving data quality, enhancing model accuracy, reducing overfitting, speeding up training, and improving interpretability.

By employing different input vectors and preprocessing techniques, we achieved an accuracy of 95.62% with Random Forest, Extra Trees, and XGBoost. However, the confusion matrix analysis revealed common misclassification patterns among these models, particularly with D1 faults.

In order to enhance the accuracy, our approach also involved the use of a custom built decision tree. Inspired by the previous work, to categorize faults into thermal and electrical types initially and then further classify them. This custom decision tree, combined with the best-performing input vectors and preprocessing techniques, achieved an accuracy of 99.375% (159/160), with only one misclassified sample.

In conclusion, our study demonstrates the potential of a hybrid approach that combines multiple machine learning techniques with a well-structured decision tree in enhancing the reliability and establish an efficiency model of fault diagnosis in power transformers oil. This approach shows significant improvements in diagnostic accuracy compared to traditional methods and other machine learning models results, underscoring the benefits of leveraging ensemble methods and customized strategies in practical applications.

Bibliography

1. Sanchez, J. *Aide au diagnostic de défauts des transformateurs de puissance* Spécialité: Génie Electrique. Thèse de Doctorat (Université de Grenoble, Grenoble, France, Juin 2011).
2. Betie, A. *Impacts de la qualité du système d'isolation sur la condition et l'efficacité des transformateurs de puissance* Thèse de Doctorat (Université du Québec à Chicoutimi, Chicoutimi, Canada, Sept. 2015).
3. Benamar, R. *Influence du nombre de claquages sur les propriétés de l'huile de transformateur* Mémoire de Magister (Ecole Nationale Polytechnique, Alger, Algérie, Juin 2008).
4. Abdi, S. E. *Influence du vieillissement thermique sur les propriétés de l'huile de transformateur* Thèse de Doctorat (Ecole Nationale Polytechnique, Alger, Algérie, Juin 2012).
5. Eke, S. *Stratégie d'évaluation de l'état des transformateurs: Esquisse de solutions pour la gestion intégrée des transformateurs vieillissants* Spécialité: Génie Electrique. Thèse de Doctorat (Ecole Centrale de Lyon, Lyon, France, Juin 2018).
6. Liang, Y., Wang, Z. & Liu, Y. *Power transformer DGA data processing and alarming tool for on-line monitors* in *2009 IEEE/PES Power Systems Conference and Exposition* (Seattle, WA, 15–18 March 2009), 1–8.
7. Benmahamed, Y. *Application de méthodes intelligentes pour le diagnostic de l'huile de transformateur par analyse des gaz dissous* Thèse de Doctorat (Ecole Nationale Polytechnique, Alger, Algérie, Mai 2019).
8. International Electrotechnical Commission. *IEC 60599-2008, "Mineral oil-impregnated electrical equipment in service-guide to the interpretation of dissolved and free gases analysis"* (2007).
9. IEEE Standards Coordinating Committee. *IEEE Guide for the Interpretation of Gases Generated in Mineral Oil-Immersed Transformers* Standard C57.104-2019. New York, NY, USA, 2019.
10. Ali, M. S., Abu Bakar, A. H., Omar, A., Abdul Jaafar, A. S. & Mohamed, S. H. Conventional methods of dissolved gas analysis using oil-immersed power transformer for fault diagnosis: A review. *Electric Power Systems Research* **216,** 109064. ISSN: 0378-7796. https://www.sciencedirect.com/science/article/pii/S0378779622011130 (2023).
11. Dornenburg, E. & Strittmatter, W. Monitoring oil-cooled transformers by gas analysis. *Brown Boveri Review* **61,** 238–247 (1974).
12. Rogers, R. R. IEEE and IEC codes to interpret incipient faults in transformers using gas in oil analysis. *IEEE Transactions on Electrical Insulation* **13,** 348–354 (Oct. 1978).

13. Duval, M. The Duval triangle for load tap changers, non-mineral oils and low temperature faults in transformers. *IEEE Electrical Insulation Magazine* **24,** 22–29 (2008).

14. Duval, M. & Lamarre, L. The Duval Pentagon- A new Complementary Tool for the Interpretation of Dissolved Gas Analysis in Transformers. *IEEE Electrical Insulation Magazine* **30,** 9–12 (Nov. 2014).

15. Syafruddin, H. & Nugroho, H. P. *Dissolved gas analysis (DGA) for diagnosis of fault in oil-immersed power transformers : A case study* in *4rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)* (Medan, Indonesia, Mar. 2020), 57–62.

16. Dukarm, J., Draper, Z. & Piotrowski, T. Diagnostic simplexes for dissolved-gas analysis. *Energies* **13,** 1–16 (Dec. 2020).

17. Benmahamed, Y., Teguar, M. & Boubakeur, A. Application of SVM and KNN to Duval pentagon 1 for transformer oil diagnosis. *IEEE Transactions on Dielectrics and Electrical Insulation* **24,** 3443–3451 (Dec. 2017).

18. Benmahamed, Y., Kemari, Y., Teguar, M. & Boubakeur, A. *Diagnosis of power transformer oil using knn and naïve bayes classifiers* in *Second International Conference on Dielectrics (ICD 2018 Conference), fully sponsored by IEEE Dielectrics and Electrical Insulation Society* (Budapest, Hungary, July 2018), 1–4.

19. Benmahamed, Y., Kherif, O., Teguar, M., Boubakeur, A. & Ghoneim, S. S. M. Accuracy improvement of transformer faults diagnostic based on DGA data using SVM-BA classifier. *Energies* **14,** 2970 (May 2021).

20. Guerbas, F., Benmahamed, Y., Teguar, Y., Dahmani, R. A., Teguar, M., Ali, E., Bajaj, M., Dost Mohammadi, S. A. & Ghoneim, S. S. M. Neural networks and particle swarm for transformer oil diagnosis by dissolved gas analysis. *Scientific Reports* **14,** 9271. ISSN: 2045-2322. https://doi.org/10.1038/s41598-024-60071-0 (2024).

21. Kherif, O., Benmahamed, Y., Teguar, M., Boubakeur, A. & Ghoneim, S. S. M. Accuracy improvement of power transformer faults diagnostic using KNN classifier with decision tree principle. *IEEE Access* **9,** 81693–81701 (June 2021).

22. Ibtiouen, R. *TRANSFORMATEURS MONOPHASES* (l'ENP Alger, 2020-2021).

23. International Electrotechnical Commission. *IEC 60076-1-2011, "Transformateurs de puissance - Partie 1: Généralités"* (Apr. 2011).

24. Engineering Zone. *Different Parts of Transformer* Accessed: 2024-06-06. 2015. https://engineeringszone.blogspot.com/2015/10/different-parts-of-transformer.html.

25. Bouaicha, A. *Application de techniques modernes au diagnostic des transformateurs de puissance* PhD thesis (Université du Québec à Chicoutimi, Chicoutimi, Canada, 2015).

26. Schenk, A. *Surveillance continue des transformateurs de puissance par réseau de neurones* PhD thesis (Ecole Polytechnique Fédérale de Lausanne, Lausanne, Suisse, 2001).

27. contributors, W. *Glucose* Accessed: 2024-06-20. 2024. https://en.wikipedia.org/wiki/Glucose.

28. contributors, W. *Cellulose* Accessed: 2024-06-20. 2024. https://en.wikipedia.org/wiki/Cellulose.

29. Berger, N. Liquides isolants en électrotechnique – Caractéristiques des produits. *Techniques de l'ingénieur* **Traité de Génie Electrique, D2471,** 20 (Nov. 2002).

30. Fontes, D., Padilla, E. & Filho, E. B. Nanofluids Based on Oil Flowing Inside an Electrical Transformer Simplified (Nov. 2014).

31. Perrier, C. & Beroual, A. Experimental Investigations on Insulating Liquids for Power Transformers: Mineral, Ester, and Silicone Oils. *IEEE Electrical Insulation Magazine* **25,** 6–13 (Nov. 2009).

32. Oommen, T. V. Vegetable Oils for Liquid-Filled Transformers. *IEEE Magazine* **18,** 6–11 (Jan. 2002).

33. Rhodia. *Rhodorsil 604V50 Silicone Oil* Product datasheet. Available online: https://www.silitech.ch/wp-content/uploads/fiche_technique_f/198.pdf. Year.

34. Wiklund, B. P. P. Properties of Mineral Insulating Oils in Service. *CIGRE,* 65–73 (1 2010).

35. Commission Electrotechnique Internationale. *CEI 60156-1995, "Liquides isolants – Mesure de la tension de claquage à fréquence industrielle - Méthode d'essai"* (Août 1995).

36. Wiklund, B. P. P. Properties of Mineral Insulating Oils in Service. *CIGRE,* 65–73 (1 2010).

37. Wang, Z. *Evaluation of the Dielectric Capability of Ester Based Oils for Power Transformers* PhD thesis (Department of Electrical and Electronic Engineering, June 2008).

38. Vuarchex, P. J. Huiles et liquides isolants. *Technique d'Ingénieur* **D230, D231, D232** (1986).

39. CIGRE. *Recent Developments in DGA Interpretation* Technical Brochure 296 (CIGRE, 2006).

40. Commission Electrotechnique Internationale. *CEI 61203-1992, "Esters organiques de synthèse à usage électrique – Guide de maintenance des ester pour transformateurs dans les matériels"* (Décembre 1992).

41. Organisation internationale de normalisation. *ISO 3016, "Produits pétroliers - Détermination du Point d'écoulement"* (Août 1994).

42. Organisation internationale de normalisation. *ISO 2719, "Détermination du point d'éclair - Méthode Pensky-Martens en vase clos"* (Nov. 2002).

43. Organisation internationale de normalisation. *ISO 3016, "Produits pétroliers - Détermination du Point d'écoulement"* (Août 1994).

44. Organisation internationale de normalisation. *ISO 2719, "Détermination du point d'éclair - Méthode Pensky-Martens en vase clos"* (Nov. 2002).

45. Fofana, I. *Isolants Liquides en Haute Tension* in *Atelier CNHT 2016* (2016).

46. Abdi, S., Harid, N., Saffidine, L., Boubakeur, A. & Haddad, A. The Correlation of Transformer Oil Electrical Properties with Water Content using a Regression Approach. *Energies* **14,** 1–14 (Apr. 2021).

47. ASTM International. *ASTM D 1903-03, "Standard Test Method for Coefficient of Thermal Expansion of Electrical Insulating Liquids of Petroleum Origin, and Askarels"* (Oct. 2003).

48. Commission Electrotechnique Internationale. *CEI 60814-1997, "Isolants liquides — Cartons et papiers imprégnés d'huile — Détermination de la teneur en eau par titrage coulométrique de Karl Fischer automatique"* (Août 1997).

49. International Organization for Standardization. *ISO 6618, "Produits pétroliers et lubrifiants – Détermination de l'indice d'acide ou de l'indice de base – Méthode par titrage en présence d'un indicateur coloré"* (Février 1997).

50. International Electrotechnical Commission. *IEC 62021-1-2003, "Insulating Liquids - Determination of Acidity -Part 1: Automatic Potentiometric Titration"* (Sept. 2003).

51. International Electrotechnical Commission. *IEC 61125, "Insulating Liquids – Test Methods for Oxidation Stability Test Method for Evaluating the Oxidation Stability of Insulating Liquids in the Delivered State"* (Jan. 2018).

52. Wasserberg, V., Borsi, H. & Gockenbach, E. *A Novel System for the Prolongation of the Lifetime of Power Transformers by Reduced Oxidation and Aging* in *IEEE Conference Record of the 2004 IEEE International Symposium on Electrical Insulation (ISEI)* (Indianapolis, IN, USA, 19–22 September 2004), 233–236.

53. Perrier, C. *Etude des huiles et des mélanges à base d'huile minérale pour transformateurs de puissance – recherche d'un mélange optimal* Spécialité: Génie Electrique. Thèse de Doctorat (Ecole Centrale de Lyon, Avril 2005).

54. Prabhashankar, V. & Badkas, D. J. Mechanism of Oxidation of Transformers Oils. *Journal of the Institute of Petroleum* **47,** 201–211 (June 1961).

55. Shtern, E. N., Kuznetsova, S. S., Karpukina, N. A. & Mogileva, T. S. Oil Oxidation Temperature Coefficient. *Soviet Power Engineering* (Jan. 1981).

56. Emanuel, N. M., Denisov, E. T. & Maizus, Z. K. *Liquid Phase Oxidation of Hydrocarbons* (Plenum Press, New York, USA, 1967).

57. Sierota, A. & Rungis, J. Electrical Insulating Oils – Part 1: Characterization and Pretreatment of New Transformer Oils. *IEEE Electrical Insulation Magazine* **11,** 8–20 (1995).

58. Comité d'Electrotechnique Internationale. *IEC 60422-2005, "Lignes Directives de la Maintenance et de la Surveillance des Huiles Minérales Isolantes en Service dans les Matériels Électriques"* (Oct. 2005).

59. Riedmann, C. & et al. Online Dissolved Gas Analysis Used for Transformers – Possibilities, Experiences, and Limitations. *e & i Elektrotechnik und Informationstechnik* **139,** 1–10 (Feb. 2022).

60. Li, S. *Study of Dissolved Gas Analysis under Electrical and Thermal Stresses for Natural Esters used in Power Transformers* PhD thesis (The University of Manchester, Dec. 2012).

61. Gómez, J. A. *Experimental Investigations on the Dissolved Gas Analysis Method (DGA) through Simulation of Electrical and Thermal Faults in Transformer Oil* PhD thesis (Institute fur Angewandte Analytische Chemie, Universität Duisburg-Essen, 2014).

62. Senoussaoui, M. E. A. *Contributions des techniques intelligentes au diagnostic industriel des transformateurs de puissance* Thèse de Doctorat (Université Djillali Liabbes de Sidi Bel Abbes, Jan. 2019).

63. International Electrotechnical Commission. *IEC 60599-1978, "Interpretation of the analysis of gases in transformers and other oil-filled electrical equipment in service"* (1978).

64. Syafruddin, H. & Nugroho, H. P. *Dissolved Gas Analysis (DGA) for Diagnosis of Fault in Oil-Immersed Power Transformers: A Case Study* in *2020 4th International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)* (2020), 57–62.

65. International Electrotechnical Commission. *Mineral oil-impregnated electrical equipment in service - Guide to the interpretation of dissolved and free gases analysis* Technical Report 60599-1999 (International Electrotechnical Commission, Mar. 1999).

66. Wang, Z. *Artificial intelligence applications in the diagnosis of power transformer incipient faults* PhD thesis (Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA, Aug. 2000).

67. Duval, M. New techniques for dissolved gas-in-oil analysis. *IEEE Electrical Insulation Magazine* **19,** 6–15 (Mar. 2003).

68. *Mineral oil-filled electrical equipment in service – Guidance on the interpretation of dissolved and free gases analysis* International Electrotechnical Commission, 2015.

69. Dornenburg, E. & Strittmatter, W. Monitoring oil-cooled transformers by gas analysis. *Brown Boveri Review* **61,** 238–247 (1974).

70. Aciu, A.-M., Nicola, C., Nicola, M. & Nitu, M. C. Complementary Analysis for DGA Based on Duval Methods and Furan Compounds Using Artificial Neural Networks. *Energies* **14,** 588 (Jan. 2021).

71. Bendiabdallah, M. H. *L'utilisation des systèmes intelligents pour le diagnostic médical* PhD thesis (Université d'Oran 1 Ahmed Ben Bella).

72. Barigou, F. *Contribution à la catégorisation de textes et à l'extraction d'information* PhD thesis (Université d'Oran 1 Ahmed Ben Bella, 2012).

73. Cortes, C. & Vapnik, V. Support-vector network. *Machine Learning* **20,** 273–297 (1995).

74. Hsu, C.-W. & Lin, C.-J. A Comparison of Methods for Multiclass Support Vector Machines. *IEEE Transactions on Neural Networks* **13,** 415–425 (2002).

75. Chang, C.-C., Hsu, C.-W. & Lin, C.-J. The Analysis of Decomposition Methods for Support Vector Machines. *IEEE Transactions on Neural Networks* **11,** 1003–1008 (2000).

76. Tapak, L., Afshar, S., Afrasiabi, M., Ghasemi, M. & Alirezaei, P. Application of Genetic Algorithm-Based Support Vector Machine in Identification of Gene Expression Signatures for Psoriasis Classification: A Hybrid Model. *BioMed Research International* **2021,** 1–10 (Sept. 2021).

77. Theodoridis, S. & Koutroumbas, K. *Pattern Recognition* 2nd. Chap. 3 (Academic Press, Elsevier, 2003).

78. Hussain, L., Awan, I., Aziz, W., Saeed, S., Ali, A., Zeeshan, F. & Kwak, K. Detecting Congestive Heart Failure by Extracting Multimodal Features and Employing Machine Learning Techniques. *BioMed Research International* **2020,** 1–19 (Feb. 2020).

79. Ahmadi, S. & Khosrowjerdi, M. Fault detection Automation in Distributed Control Systems using Data-driven methods: SVM and KNN (Aug. 2021).

80. Boser, A., Guyon, I. & Vapnik, V. N. *A training algorithm for optimal margin classifiers* in *Fifth Annual Workshop on Computational Learning Theory* (Pittsburgh, 1992), 144–152.

81. Aizerman, A., Braverman, E. M. & Rozoner, L. I. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control* **25,** 821–837 (1964).

82. Mercer, J. Functions of positive and negative type and their connection with the theory of integral equations. *Philos. Trans. Roy. Soc. London* **209,** 415–446 (1909).

83. Rakotomamonjy, A. & Sudar, F. *SVM Variable selection with application to pedestrian detection* in *Congrès de Reconnaissance de Formes et d'Intelligence Artificielle* **4** (Toulouse, France, 2004).

84. Mayoraz, E. & Alpaydin, E. *Support vector machines for multi-class classification* tech. rep. 98-06 (IDIAP, 1998).

85. Platt, J. in *Advances in Kernel Methods-Support Vector Learning* 185–208 (MIT Press, 1999).

86. Duan, K. B. & Keerthi, S. *Which is the best multiclass SVM method: an empirical study* in *Springer-Verlag, Berlin Heidelberg* (2005), 278–285.

87. Murphy, K. P. *Machine Learning: A Probabilistic Perspective* ISBN: 9780262018029. https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr_1_2?ie=UTF8&qid=1336857747&sr=8-2 (MIT Press, Cambridge, Mass., 2013).

88. Spiceworks. *What is Decision Tree?* Accessed on June 8, 2024. https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-decision-tree/.

89. Mian, Z., Deng, X., Dong, X., Tian, Y., Cao, T., Chen, K. & Jaber, T. A. A literature review of fault diagnosis based on ensemble learning. *Engineering Applications of Artificial Intelligence* **127,** 107357. ISSN: 0952-1976. https://www.sciencedirect.com/science/article/pii/S0952197623015415 (2024).

90. GeeksforGeeks. *Bagging vs Boosting in Machine Learning* Visited: 2024-06-04. 2024. https://www.geeksforgeeks.org/bagging-vs-boosting-in-machine-learning/.

91. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. *Forest of randomized trees* Accessed: 2024-06-07. 2024.

92. Narassiguin, A. *Apprentissage Ensembliste, Étude comparative et Améliorations via Sélection Dynamique* 2018LYSE1075. PhD thesis (2018). http://www.theses.fr/2018LYSE1075/document.

93. Sutikno, H., Prasojo, R. A., Abu-Siada, A., *et al.* Machine learning based multi-method interpretation to enhance dissolved gas analysis for power transformer fault diagnosis. *Heliyon* (2024).

94. Khan, M. Y., Qayoom, A., Nizami, M., Siddiqui, M. S., Wasi, S. & Syed, K.-U.-R. R. Automated Prediction of Good Dictionary EXamples (GDEX): A Comprehensive Experiment with Distant Supervision, Machine Learning, and Word Embedding-Based Deep Learning Techniques. *Complexity* (Sept. 2021).

95. Geurts, P., Ernst, D. & Wehenkel, L. Extremely randomized trees. *Machine Learning* **63,** 3–42 (2006).

96. Dernsjö, A. & Blom, E. *A Gradient Boosting Tree Approach for Behavioural Credit Scoring with Bayesian Optimization and Explainable AI* (Degree Project in Mathematical Statistics, Second Cycle, 30 Credits, 2023).

97. Nielsen, D. Tree Boosting With XGBoost - Why Does XGBoost Win “Every” Machine Learning Competition? *Online publication* (2016).

98. Schapire, R. E. The strength of weak learnability. *Machine Learning* **5,** 197–227. ISSN: 1573-0565. https://doi.org/10.1007/BF00116037 (1990).

99. Analytics, P. An Introduction to Decision Tree Theory. *Annals of online.* Accessed on June 8, 2024. https://www.precision-analytics.ca/articles/an-introduction-to-decision-tree-theory/ (Accessed 2024).

100. Chen, T. & Guestrin, C. *XGBoost: A Scalable Tree Boosting System* in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, San Francisco, California, USA, 2016), 785–794. ISBN: 9781450342322. https://doi.org/10.1145/2939672.2939785.

101. scikit-learn developers. *Preprocessing data in scikit-learn* Accessed: 2024-06-12 (2023). https://scikit-learn.org/stable/api/sklearn.preprocessing.html.

102. Wikipedia contributors. *Interquartile range — Wikipedia, The Free Encyclopedia* Accessed: 2024-06-19. 2024. https://en.wikipedia.org/wiki/Interquartile_range.

103. Inc., S. I. *What are Neural Networks?* http://www.faqs.org/faqs/ai-faq/neural-nets/part2/section-16.html. Accessed: 2024-06-10. 2003.

104. Vidhya, A. *What is a Confusion Matrix?* Medium. Date Unavailable. https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5.

105. Brownlee, J. *One-vs-Rest and One-vs-One for Multi-Class Classification* Accessed: 2024-06-12. 2020. https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/.