jamk

Detection of Anomalies in Electric Vehicle Charging Sessions Data

A case study for Electric Vehicle Charge Detail Record Data

Sami Myllykoski

Master's thesis
May 2024
Master's Degree Programme in Artificial Intelligence and Data Analytics

jamk | Jyväskylän ammattikorkeakoulu
University of Applied Sciences

jamk

Description

Myllykoski, Sami

Detection of Anomalies in Electric Vehicle Charging Sessions Data

Jyväskylä: Jamk University of Applied Sciences, May 2024, 86 pages.

Degree Programme in Artificial Intelligence and Data Analytics. Master’s thesis.

Permission for open access publication: Yes

Language of publication: English

Abstract

Anomalies, or outliers, in data are deviations from expected patterns and can result from errors, rare events, system glitches, or fraudulent activities. Understanding anomalies is crucial across industries, as they can signal fraud, malfunctions, or unforeseen events. This research study explores various anomaly detection techniques, from supervised to unsupervised learning methods, to enhance data analysis skills across domains.

The study categorizes different techniques into subgroups based on their underlying approaches, aiming to offer a structured overview. It focuses on anomaly detection in data, anticipating to enhance methodological understanding and demonstrate cross domain applicability.

Anomaly detection is a notable problem investigated across numerous research domains and application fields. While some techniques are tailored for specific application domains, others are more generalized. The study seeks to offer a thorough understanding of the techniques landscape in anomaly detection in Electric Vehicle Charging Sessions Data.

The study enhance understanding of the various methological directions within anomaly detection and clustering. It illustrates what techniques was developed in one domain can be deployed to other domains where it was not initially intended.

The evaluation metrics for classification models have achieved an accuracy of over 99% with binary classification algorithms. This high accuracy rate is encouraging and presents a significant potential to automate anomaly detection in production systems. This automation can help mitigate business risks substantially.

Keywords/tags (subjects)

Anomaly Detection, Machine Learning, Neural Network, Artificial Intelligence

jamk

Kuvailulehti

Myllykoski, Sami

Sähköajoneuvojen lataussessioiden datassa esiintyvien poikkeamien havaitseminen

Jyväskylä: Jyväskylän ammattikorkeakoulu, Toukokuu 2024, 86 sivua.

Koulutusohjelma: Tekoäly ja data-analytiikka. Insinööri (ylempi AMK), tietotekniikka

Verkkojulkaisulupa myönnetty: Kyllä

Julkaisun kieli: englanti

Tiivistelmä

Poikkeavuudet eli poikkeamat datassa ovat odotetuista malleista poikkeavia ja voivat johtua virheistä, harvinaisista tapahtumista, järjestelmävioista tai petollisesta toiminnasta. Poikkeamien ymmärtäminen on tärkeää eri toimialoilla, sillä ne voivat viitata petoksiin, toimintahäiriöihin tai ennakoimattomiin tapahtumiin. Tämä tutkimus käsittelee erilaisia poikkeamien havaitsemistekniikoita, aina ohjatusta oppimisesta valvomattomiin oppimismenetelmiin, parantaakseen datan analysointitaitoja eri aloilla.

Tutkimus jaottelee eri tekniikat alakategorioihin niiden taustalla olevien lähestymistapojen perusteella, pyrkien tarjoamaan jäsennellyn yleiskuvan. Keskittyy datassa esiintyvien poikkeamien havaitsemiseen, tavoitteenaan parantaa menetelmällistä ymmärrystä ja osoittaa monialainen sovellettavuus.

Poikkeamien havaitseminen on merkittävä ongelma, jota on tutkittu monilla eri tutkimusaloilla ja sovellusalueilla. Jotkin tekniikat on räätälöity tiettyihin sovellusalueisiin, kun taas toiset ovat yleisempiä. Tutkimuksen tavoitteena on tarjota perusteellinen ymmärrys poikkeamien havaitsemistekniikoiden kentästä sähköajoneuvojen lataussessioiden datassa.

Tutkimus syventää ymmärrystä eri menetelmällisistä suunnista poikkeamien havaitsemisessa ja klusteroinnissa. Sekä osoittaa, että yhdessä sovellusalueessa kehitettyjä tekniikoita voidaan soveltaa myös muille alueille, joihin niitä ei alun perin ollut tarkoitettu.

Luokittelumallien arviointimenetelmät saavuttivat yli 99 % tarkkuuden binääriluokittelualgoritmeilla. Korkea tarkkuusaste on rohkaiseva ja tarjoaa merkittävän potentiaalin automatisoida poikkeamien havaitsemista tuotantojärjestelmissä. Automaatiolla voi merkittävästi auttaa vähentämään liiketoimintariskejä.

Avainsanat

Poikkeamien havaitseminen, Koneoppiminen, Neuroverkko, Tekoäly

Contents

Figures

Tables

Abbreviations

AI	Artificial Intelligence
AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
CDR	Charging Details Records
CM	Confusion Matrix
CPMS	Charging Point Management System
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
EDA	Exploratory Data Analytics
GMM	Gaussian Mixture Model
FN	False Negatives
FP	False Positives
IDE	Integrated Development Environment
IDS	Intrusion Detection Systems
OCA	Open Charge Alliance

OCPP	Open Charge Point Protocol
OCPI	Open Charge Point Interface
PCA	Principle Component Analysis
RF	Random Forest
ROC	Receiver Operating Characteristic
ROC AUC	Receiver Operating Characteristic Area Under Curve

1 Introduction

Escalating this energy demand are Artificial Intelligence (AI) models. Huge, popular models like ChatGPT signal a trend of large-scale AI, boosting some forecasts that predict data centers could draw up to 21% of the world's electricity supply by 2030 (Foy, 2023). AI is rapidly advancing, presenting the opportunity to revolutionize numerous aspects of our lives and work. AI holds the potential to transform an broad range of industries, inventing manner for new avenues and innovations. Significant strides have been made in machine learning and deep learning techniques, leading to transformations across various sectors such as healthcare, finance, and transportation. “By 2035, Artificial Intelligence (AI) has the power to increase productivity by 40 percent or more, these benefits mean that AI has the potential to boost profitability an average of 38 percent by 2035.” (Purdy et al., 2017).

Anomaly detection in electric vehicle charging data is a multidimensional challenge that necessitates a combination of domain expertise, data preprocessing, and the application of suitable anomaly detection techniques. As the adoption of electric vehicles continues to grow, effective anomaly detection becomes increasingly important for maintaining the reliability, the risk mitigation and efficiency of charging infrastructure for the electric vehicle drivers.

In this research study has many key benefits and potential business savings with it. What aim to enhance our ability to proactively address anomalies, reduce operational risks, and optimize decision-making processes. The potential benefits of early anomaly detection, cost savings, and enhanced decision-making make this investment a strategic imperative. The choice between unsupervised and supervised machine learning algorithms for anomaly detection relies on having a labeled dataset available and the specific requirements of the application. There are many financial justification studies, here are some early once and the latest one for their financial benefits. “Google and DeepMind as AI study in 2016 has been shown across data centers lines up to 40% noise can be reduced.” (Google & DeepMind 2016). AI is one of the most desired digital transformation possibilities for businesses in the near future.

This business case is essential for developing an AI pilot with potential solutions and is a way to facilitate ongoing learning for both humans and AI.

Understanding the charging behavior of electric vehicle drivers is crucial for optimizing the charging infrastructure. anomaly detection can reveal patterns such as unusual charging times, excessive power consumption, or unexpected charging durations, providing insights into user behavior. McKinsey Global Institute, the Next Digital Frontier, states that “Successful AI transformations require elements similar to those found in successful digital and analytics transformations”. The elements are: use cases/sources of value, data ecosystems, techniques & tools, workflow integration and open culture & organization (Bughin J et al., 2016). Anomaly detection helps identify unauthorized access or potential fraudulent activities, such as someone trying to manipulate charging transactions or gain unauthorized access to charging stations.

2 Artificial intelligence

Machines learning falls under the umbrella of AI. There are various approaches to implementing machines learning. The primary methods for applying machine learning to data are supervised learning, unsupervised learning, and reinforcement learning (Sivula, 2021). Du-Harpur et al. (2020) wrote that machine learning is a method of instructing an algorithm to draw judgments from provided data. A machine learning model is a computational framework which represents a problem or system, engineered to discern patterns, relationships, and associations from data. It functions as a mathematical and algorithmic construct capable of predicting outcomes, classifying data, or making decisions based on input data. In machines learning, anomaly detection means the way of finding out the patterns of normal data and then detecting anomaly data that do not have the same pattern (Hawkins, 1980).

2.1 Machine learning

The greater number of instances the algorithm receives, the more precise the resultant model becomes. Given the unpredictable nature of life or business, and the possibility of skewed, incomplete, or inaccurate data, developing an algorithm that achieves high accuracy can be challenging (Yang et al., 2023). In crafting a dataset for a specific use case, it is essential to contemplate how to address occurrences external to the offered dataset that may impact the events within the dataset. Such factors can occasionally lead to inaccuracies in the algorithm beyond the testing phase. This involves choosing which data to include and which features to exclude to avoid biases or inaccuracies (Yu, 2004).

In many cases, the initial step is recognizing a issue that requires a resolution (Sivula, 2021). Nevertheless, this can also be interchangeably employed for recognizing patterns within data and shaping a problem that can be solved using the available records. Once the first matter is acknowledged and depicted, the subsequent procedure involves sourcing data that would be utilized to devise a solution (Sivula, 2021). Typically, machines learning is generally applied to solve a variety of problems that are challenging to solve

through conventional methods or those that necessitate the analysis of substantial amounts of data.

Common challenges encountered in machine learning include inadequate data availability, data biases, and issues such as inaccuracies, incompleteness, or faults within the data itself (Yang et al., 2023). In many instances, gathering data from the beginning is a time-consuming process, often spanning months or even years, contingent on the complexity of the requirements. The machine learning process often operates similarly across various scenarios, adhering to a series of steps aimed at achieving the desired outcome (Sivula, 2021).

Figure 1 show the categorizing of various machine learning algorithms into different categories, which can be customized and applied in various ways within the different use cases. Nassif et al. (2021) proposed several classification algorithms could be used for anomaly detection, among the most popular are support vector machines, neural network, naive bayes, decision trees, ensemble methods and k-nearest neighbor.

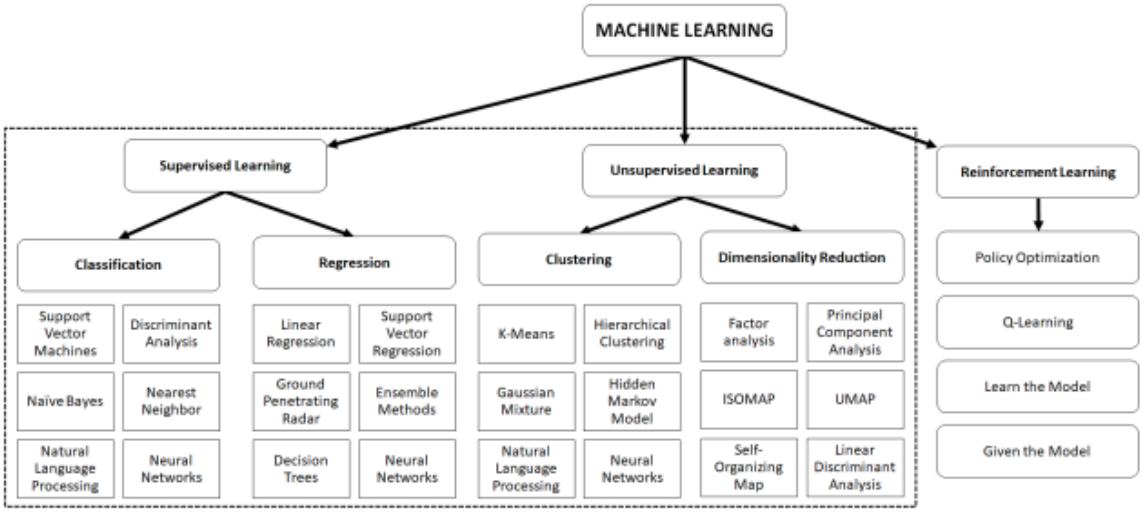


Figure 1: Summary of machine learning algorithms (Adapted from Durga et al., 2019)

In such cases, rather than learning the underlying structure in the data, the machines learning algorithm ends up memorizing the specific details of the dataset. If labeled anomaly data is available and the effort is feasible, supervised learning can be a powerful tool.

However, unsupervised methods are often preferred in scenarios where obtaining labeled anomalies is challenging or where adaptability to changing patterns is crucial. Hybrid approaches that combine elements of both unsupervised and supervised techniques are also explored in certain cases, leveraging the strengths of each paradigm.

2.2 Unsupervised Machine learning

Supervised methods may provide more interpretability as they explicitly learn from labeled data, while unsupervised methods may operate as "black boxes" in terms of anomaly identification. In the unsupervised category, you can find models like Isolation Forests, One-Class Support Vector Machines, and Clustering techniques (e.g., Density-Based Spatial Clustering of Applications with Noise (DBSCAN)) can be applied for batch detection, don't require labeled anomalies during training. Unsupervised machine learning techniques are characterized by their ability to handle unlabeled data (Doshi et al., 2022). In these instances, the data lacks labels and is employed to uncover patterns. In the book Doshi et al. (2022) explains how, unlike in supervised learning, unsupervised learning does not use on labeled data and instead works with data that lacks clear classifications or annotations. In such scenarios, the data is unlabeled and is employed to address challenges of greater complexity. These algorithms do not require labeled anomalies during training and can detect deviations from normal patterns effectively. In unsupervised anomaly detection, it's essential for normal data to be more prevalent than anomalous data (Bhattacharyya et al., 2014). Consider the cost and effort involved in labeling anomalies. If the effort is substantial or ongoing, unsupervised methods might be more practical

The clustering type of learning technique is utilized to aggregate same data nodes together based on their attributes or features. Clustering (Jain et al., 1988; Tan et al. 2005) involves grouping similar data nodes into clusters. Clustering is primarily an unsupervised technique operate under semi-supervised clustering (Basu et al., 2004) has also been explored lately. The clustering based anomaly detection techniques can be grouped into three categories (Chandola et al., 2009).

Compared to supervised learning, while in supervised learning algorithms are educated on tagged data. Unsupervised learning algorithms operate with untagged data, meaning there are no predefined categories or classes. This process reveals the inherent structure and groupings within the data, providing valuable insights that may not be immediately apparent from simply observing the data itself (Theissler, 2021).

The primary objective of clustering is to split a dataset into separate batches, or clusters, data instances within the identical cluster display higher uniformity to each other than to those in separate clusters. This process aims to disclose natural schemes or characteristics in the data without relying on predefined labels. Depending on the objectives of the model, unsupervised learning could be employed to data batch in various manners (Géron, 2017; Salian, 2018).

Types of Clustering Algorithms:

- Partitioning Methods: These algorithms segment the data into a preassigned amount of clusters. K-means clustering is an unsupervised clustering method that relies on distance-based calculations. It organizes data dots into a predetermined number of clusters or batchs, where dots that are nearby one another in the attribute space are assigned to the same cluster.

- Hierarchical Methods: These algorithms create a tree-like hierarchical decomposition of the dataset, where clusters at one level of the hierarchy

are established by joining or fragmenting clusters at the previous level. Agglomerative and divisive clustering exemplify hierarchical methods.

- Density-Based Methods: These algorithms identify clusters as areas of high data point density divided by regions of sparse. DBSCAN is one of a widely recognized density-based technique.

- Model-Based Methods: These algorithms operate under the assumption that algorithms is expecting the input was created based on a combination of probability distributions. They aim to find the factors of these distributions to best explain the data. The Gaussian mixture model (GMM) stands in manner of a prominent model-based clustering algorithm.

Evaluation of Clustering: Unlike supervised learning where we have predefined metrics like accuracy, assessing the effectiveness of clustering algorithms can be more intuitive due to the absence of ground truth labels. However, some common methods for evaluating clustering include:

- Internal Evaluation: Metrics like silhouette score, Davies–Bouldin index, and Calinski-Harabasz index measure the quality of clustering based on inner criteria such as cohesion, separation, and compactness of clusters. Liu et al., (2010) shows a suite of 11 widely used internal validation measures, these measures represent a good coverage of the validation measures available in different fields, such as machine learning.

- External Evaluation: When ground truth labels are accessible, external assessment criterias such as Adjusted Rand Index or Fowlkes-Mallows index can be used to compare the clustering results with the true labels.

The primary goal of a neural network is to uncover relationships among properties within the data. It comprises a selection of learning algorithms that mitigate the functioning of the human brain or aiming to replicate some of the

cognitive functions of the human brain in artificial schemes. In this context, a "neuron" in a neural network is a mathematical function designed to gather and categorize information based on a predefined architecture.

The points are structured and organized into linear sequences called layers (Alaloul et al., 2020).

Neural networks are utilized in situations where the interconnection between the entry attributes (X) and the outcome label (Y) is non-linear. Utilizing randomly initialized weights and biases, the network employs differentiable non-linear functions, typically continuous in nature, to predict the output variable, can be referred to as y-predicted.

In neural networks, there are typically comprises three primary layers: the Input Layer, Hidden Layer(s), and Output Layer like Figure 2 below. Please refer to the diagram provided for a visual representation of these layers. Deep learning advances us layer by layer, connection by connection, propelling us toward a outlook where machines grasp a deep knowledge of our intricacies, this revolutionizes our interactions with technology and unlocks new frontiers of innovation (Meriem, 2018).

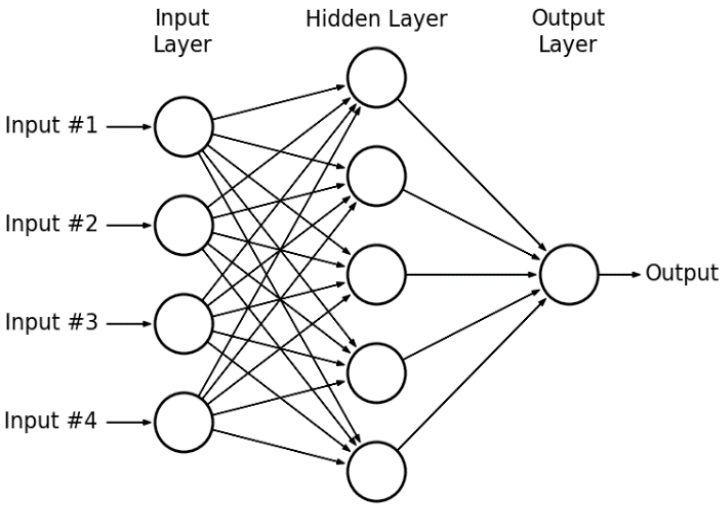


Figure 2: Neural Network example (Adapted from Manzini, 2017)

The predicted value undergoes many numerous iterations rounds, known as epochs, through the computation and minimization of error loss. Error loss is serving as a mathematical measure of the deviation when the assumed value and the real output value serve as guides for refining the model. This holds significance for selecting architecture, fine-tuning deep learning models, tuning hyperparameters, and optimizing them (Dasaradh, 2020).

In a classification technique, the objective is to foresee whether something belongs to a particular class or not. Essentially, it involves determining whether an observation falls into one category or another. In the scenario where you're predicting whether a person has diabetes or not, there are two possibilities: either the person has diabetes or they don't. This type of classification problem, where there are only two possible results, is mentioned to as binary classification.

2.3 Supervised Machine learning

Machine learning classification utilizes mathematically proven algorithms to conduct analytical tasks that would require humans significantly more time to accomplish. In the book, Doshi et al. (2022) the explanation of supervised learning underscores its reliance on data with clearly defined, measurable metrics, which serve as goalposts guiding the learning process. With the right algorithms and a well-trained model, classification programs can achieve levels of accuracy beyond what humans could achieve.

Supervised machine learning algorithms are typically not directly used for anomaly detection because they require labeled training data with both normal and anomalous instances. anomaly detection is frequently categorized within the domain of unsupervised learning, where the algorithm learns from normal patterns missing explicit labels for anomalies. There are certain technique in which supervised machine learning can be adapted or utilised indirectly for anomaly detection.

If labeled anomalies are readily available, supervised learning may be a viable option. However, acquiring labeled anomalies can pose a challenge in numerous real-world scenarios.

In supervised learning, the output can manifest as either a classification or a infinite value. Utilizing a supervised learning model results in an algorithm capable of predicting the precise label or value for a previously unseen feature, thereby adding value to the process. Doshi et al. (2022) and Hoang et al. (2023) outlines supervised learning as applicable when the training data includes both input parameters and an outcome attributes, with the desired outcome being explicitly labeled.

In machine learning, classification algorithms exploit input training data to estimate the probability that future data points will fall into predefined categories. A conventional application of classification is sorting the informations into either "spam" or "non-spam" categories.

The classification models are classify dataset into established groups or classes based on input features. When appraising the efficiency of classification models, several metrics are commonly used, consisting of accuracy metrics. It's vital to be aware of results that while employing micro-averaging, precision, recall, and F1-score all equal accuracy (Grandini et al., 2020).

Here is more about each of these metrics:

Accuracy being one of the most straightforward metrics, it offers the rate of successful predicted occurrences relative to the total occurrences in the dataset. While accuracy is a beneficial metric, it may not suffice for skewed datasets where the batches are unevenly represented.

Precision illustrates the degree of successful predicted positive occurrences out of all occurrences that were anticipated to be correct once, therefore It concen-

trates on the correctness of predictions. Precision is a crucial metric when the impact of false positives (FP) is significant.

Recall appraises a model's capability to perceive all pertinent observations false negatives (FN) within a dataset. Recall is important when the impact of FN is significant, and you want to ensure that you're collecting as many relevant instances as possible.

The F1 score, also called a coherence mean of precision and recall, provides a one indicator that achieves a stability among precision and recall. The F1 score achieves the optimal value while it is 1, meaning utmost precision and recall, and its lowest value while it is 0. It's a valuable method for assessing a model's performance, particularly relevant in situations where dealing with imbalanced class distribution or when the costs linked with untrue positives and untrue negatives differ.

Above metrics gives different insights into the performance of a classification classifier. Depending on the particular issue and the associated costs of various types of errors, one may prioritize certain metrics over others. For example, in medical diagnosis, recall might be more critical to minimize FN, while in spam detection, precision might be more important to minimize false positives.

Therefore, selecting appropriate evaluation metrics for classification models requires careful consideration of the problem's context and requirements. To simplify it, the value we predict in classification problems is not a numerical value (such as salary, height), but categorical values such as sick/healthy, rainy/sunny, successful/unsuccessful, positive/negative (Gültekin, 2023).

2.4 Algorithms

At the heart of machines learning are three key ingredients: an algorithm, the learning method, and the resultant model. The algorithm represents the methodology for addressing issue, while learning involves utilizing the algorithm to the data, such as making predictions, the model on the other hand

is the outcome of learning an algorithm with the data (Mattmann et al., 2020). An algorithm is a finite set of manual performed in a specific sequence to accomplish a defined task. According to Sivula (2021) these actions lead to the learning and evaluation phase, then testing can determine if there is sufficient data and if the chosen algorithm is suitable.

In situations where the selected choices for the algorithm struggles to extrapolate effectively from the observed data, it may experience overfitting to the provided dataset. Xue (2019) elaborates on the potential consequence of algorithm training, highlighting that one possible outcome is the creation of an overfitted model. In contrast to an overfitted model, which has been exposed to the training data excessively, an underfitted model has not undergone sufficient training. Jabbar et al. (2015) clarify how underfitting represents the flip side of poorly instructed algorithms. Underfitting model is overly simplistic and does not adequately capture the complexities within the training data. This leads to high errors on the training set itself. In the article Ding et al. (2020) elaborates on the procedure where machine learning engineers analyze the case, select, and test the algorithms best suited for it, then fine-tune them to optimize the benefits.

2.5 Features engineering

Extract relevant features from the data that may help in identifying anomalies. Hyper-parameter tuning involves scrutinizing essential aspects of the machines learning model during training phases. A critical aspect of fine-tuning is determining how many times the model is trained to employ the learning data, referred to as epochs.

Repeating the iteration as needed is an option, but a more prevalent and cost-effective strategy involves concentrating on revising and enhancing the algorithm through further training as new data becomes accessible. This stage is referred to as feature engineering, aiming to optimize both the algorithm's performance and computational efficiency (Mattmann et al., 2020).

It is typical prevent overfitting by segregating the data into training and testing utilities. The standard recommendation is to allocate 70 percent of the data to the training set and reserve the remaining 30 percent for the test set. The classification accuracy is evidently contingent upon the quality of the training set (Zhu & Wu, 2004).

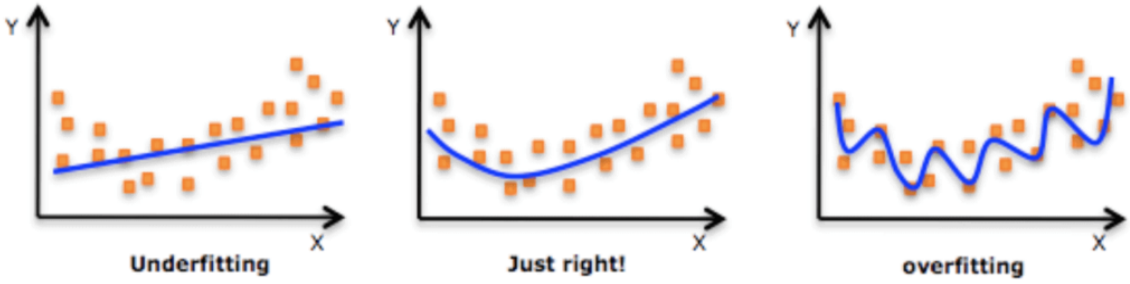


Figure 3: Different fitting types (Adapted from Datarobot, 2024)

Underfitting is another phenomenon to be aware of in machine learning. It occurs when the model generated during the learning phase fails to capture the correlations present in the training set. Fortunately, underfitting is typically easier to detect compared to overfitting Figure 3. Signs of underfitting manifest as poor performance metrics immediately, indicating when the model fails to effectively adapt into the training data.

Overfitting may occur because of a lack of data, inadequate preprocessing, or excessive training epochs, while underfitting is primarily caused by oversimplifying the model (Jason, 2023).

GridSearchCV is a method for conducting hyperparameter tuning to identify the optimal values for a given model. One of reason to using it is that to mitigate overfitting. The efficiency of a model is extensively affected by the hyperparameter values. Since the best values for hyperparameters cannot be known in advance, ideally, it should explore all possible values to determine the optimal ones. However, manually testing all combinations would be prolonged and laborious.

Therefore, it was employed GridSearchCV to automate the hyperparameter tuning process, not with all classifiers but neural network where it is more demanding manual work.

The technique knows as Stratified K-fold cross-validation what operates similarly to standard k-fold cross-validation in Figure 4, but with a focus on stratified sampling instead of random sampling.

This method might be computationally expensive but efficiently utilizes avail able data without significant waste (Scikit-Learn, 2023).

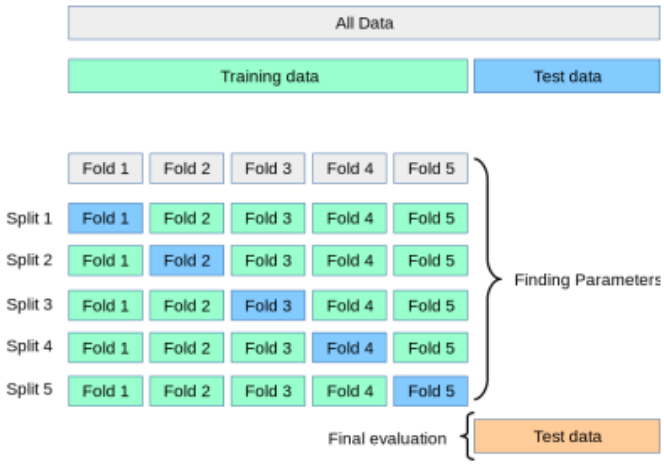


Figure 4: Visualization of k-fold on training data
(Adapted from Scikit-Learn, 2023)

2.6 Anomaly detection

In the literature on anomaly detection, anomalies are often categorized into varied groups based on their characteristics and the context in which they occur. The types of anomalies provide a framework for understanding and categorizing abnormal patterns in data, enabling researchers and practitioners to create tailored anomaly detection methods and algorithms for different application domains. A thorough grasp of the several anomaly types present in datasets holds significance for several reasons. Having a fundamental grasp of anomalies, including their various types and defining characteristics, is

essential in fields such as statistics, data science, machine learning, analytics, and cognitive systems. Table 1 shows the selection of approaches and applications encompassed by different surveys (Chandola et al., 2009) and the various number of survey articles mentioned below.

Table 1: Comparison of different surveys to other related survey articles (Adapted from Chandola et al., 2009)

		1	2	3	4	5	6	7	8
Techniques	Classification Based	√	√	√	√		√		
	Clustering Based	√	√	√			√		
	Nearest Neighbor Based	√	√	√			√		√
	Statistical	√	√	√		√	√	√	√
	Information Theoretic	√							
	Spectral	√							
Applications	Cyber-Intrusion Detection	√					√		
	Fraud Detection	√							
	Medical Anomaly Detection	√							
	Industrial Damage Detection	√							
	Image Processing	√							
	Textual Anomaly Detection	√							
	Sensor Networks	√							

Anomaly detection is a technique applied in data analysis to pinpoint patterns, events, or observations that deviate substantially from the anticipated behavior within a dataset. These anomalies, also referred to as outliers, novelties, or exceptions, may indicate errors, fraud, or interesting phenomena in the data.

Reporting anomalies is a critical aspect of any anomaly detection technique, typically falling into two main categories (Chandola et al., 2009). The example of outlier considers each example of test data that allocate points according to the scoring method. In such techniques, the result is listing the anomalies based on their respective scores. In the labeling technique, each test instance is assigned a particular label, either "normal" or "anomalous." Scoring-based anomaly detection methods enable analysts to set a domain-specific threshold to identify appropriate anomalies, whereas techniques providing binary labels lack direct control over this selection process.

In evaluating each anomaly detection algorithm, in case of supervised or unsupervised, it's essential to assess its effectiveness. Since anomalies are typically rare compared to normal data points, using accuracy as an evaluation metric isn't suitable. If a model simply labels everything as non-anomalous,

it could gain a high accuracy score, but fail to capture any anomalies, which is undesirable. Instead, our goal is to determine how many anomalies were detected and how many were missed. This requires metrics like precision, recall, F1-score, and the receiver operating characteristic area under curve (ROC AUC) to offer a comprehensive assessment of the algorithm's proficiency in detecting anomalies.

Indeed, one useful metric for evaluating anomaly detection algorithms is to compute the confusion matrix of values. The confusion matrix offers a compact overview of prediction outcomes in a classification problem, as depicted in Figure 5. It divides the number of accurate and inaccurate predictions based on specified classes, enabling a detailed assessment of the algorithm's proficiency.

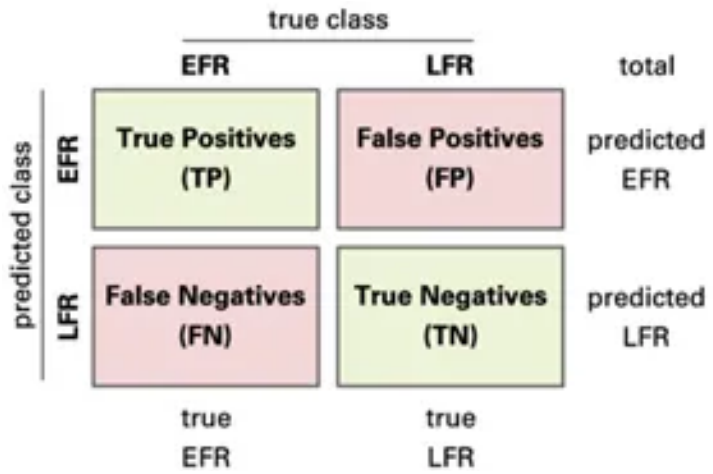


Figure 5: The confusion matrix (Adapted from Bittrich, 2019)

In anomaly detection, a true positive arises when the classifier properly predicts anomalous data as anomalous. Conversely, a true negative is when the model properly detects anomalous data as anomalous.

Alternatively, a false positive transpires when the classifier mistakenly tags non-anomalous data as anomalous, and a false negative transpires when the classifier mistakenly classifies anomalous data as non-anomalous.

These distinctions are crucial for understanding the proficiency of the anomaly detection algorithm.

Anomaly detection finds applications in various domains, including finance (fraud detection), cybersecurity, industrial monitoring, healthcare (disease outbreak detection), and many others, where identifying unusual patterns or events is vital for determination and resolution. Purpose is apply the trained model to new or hidden data to spot instances that deviate significantly from the learned patterns. The deviations are considered anomalies.

Individual anomaly detection algorithms often have limitations in detecting all types of anomalies and can vary in performance. Moreover, the complexity of algorithms does not necessarily correlate with superior performance compared to relatively simple ones.

2.7 Typology of anomalies

The typology aids researchers in understanding which algorithms can detect specific types of anomalies to what extent, thus promoting transparency in algorithm selection. These fundamental and data-centric dimensions naturally yield 3 broad groups, 9 basic types, and 63 subtypes of anomalies Foorthuis R, (2021). The typology offers a solid framework, enabling researchers to systematically analyze the expertise of different algorithms to identify various types of anomalies and to what extent. Moreover, despite some 250 years of publications on the topic, no comprehensive and concrete overviews of the different types of anomalies have hitherto been published Foorthuis R, (2021). While the typology does not directly enhance the clarity of the algorithms themselves, it facilitates a precise comprehension of the categories of anomalies and attributes of them.

This understanding is derived without delving into detailed formulas and algorithms, providing valuable insights into the anomaly detection process.

Reference	G/S	DC?	Classes of anomalies	Explicit classificatory dimensions
[6, 69, 70]	G	Y	Extreme value ano, rare class ano, simple mixed data ano, multidimensional numerical ano, multidimensional rare class ano, multidimensional mixed data ano	Types of data, cardinality of relationship
[2, cf. 31]	G	N	Extreme genuine member, contaminant	None
[34]	G	Y	Fringelier, distant outlier	None
[52]	G	Y	Strongest outlier, weak outlier, trivial outlier	Attribute subspace
[132]	G	Y	white crow, in-disguise anomaly	None
[5, 133]	G	Y	Weak outlier, strong outlier	None
[96]	G	N	Procedural error, extraordinary event, extraordinary observation, unique value combination	None
[136]	G	N	Data error, normal variance, data from other distributions, distributional assumption	None
[7]	G	N	Point anomaly, contextual anomaly, collective anomaly	None
[184]	G	Y	Known distribution ano, sparse distribution ano, local density-based ano, global density-based ano, rare instance ano, burst ano, deviant sequence ano, trend ano, irregularity ano	None
[182]	G	Y	Trivial outlier, non-trivial outlier	None
[3, 143]	S	N	Outlier, high-leverage point, influential point	None
[138, cf. 141]	S	Y	Additive outlier, temporary change, level shift, innovational outlier	None
[187]	S	Y	Isolated outlier, patch outlier, level shift	None
[142]	S	Y	Isolated outlier, shift outlier, amplitude outlier, shape outlier	None
[233]	S	Y	Trend anomaly, seasonality anomaly	None
[314]	S	Y/N	Outlier, spike, stuck-at, high-noise (plus several non-data-centric anomalies)	None
[281]	S	Y	Various spatio-temporal change patterns	Temporal, spatial, raster/vector
[20]	S	Y	Deviant vertex, deviant edge, deviant subgraph	None
[205]	S	Y	Near-star, near-clique, heavy vicinity, dominant edge	None
[125]	S	Y	Insertion, update and deletion anomaly	Based on database CRUD functions
[60]	S	Y	Foreign-symbol, foreign n-gram, rare n-gram	None
[118, cf. 146]	S	Y	Positional outlier, angular outlier	None

G/S refers to a general (broad and usually abstract) versus specific way to distinguish between classes of anomalies. *DC* stands for data-centric, meaning the anomalies can be distinguished by analyzing the dataset, without a reference to or dependency on external factors (such as unknown real-world events or arbitrary analyst decisions)

Figure 6: Summarizes the anomaly classes acknowledged in the extant literature (Adapted from Foorthuis R, 2021)

The typologies mentioned above, as summarized in Figure 6 Foorthuis R, (2021), are either overlap broad and abstract, lacking clarity and specificity regarding anomaly types, or they present clearly defined types that are specifically applicable to certain purposes, like time series analysis or regression modeling.

The typology describes the dimensional structure of data, each type provides a fundamental aspect of the data's nature to differentiate between different types of anomalies. It pertains to the classification of the characteristics (i.e., attributes) along in the outlier nature of an abnormal event, requiring proper handling during analysis to facilitate detection.

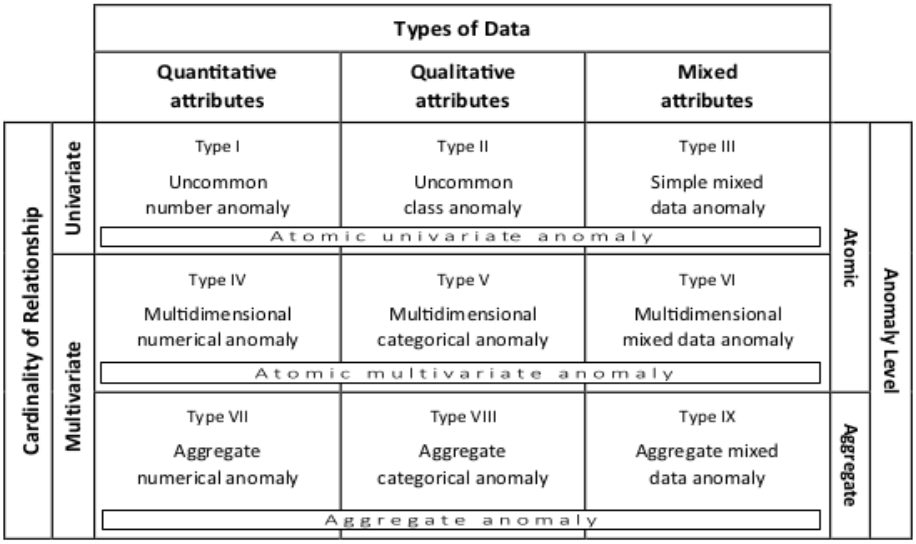


Figure 7: The framework for the topology of anomalies (Adapted from Foorthuis R, 2021)

The typology's framework, as depicted in Figure 7, comprises three main dimensions, namely data type, cardinality of relationship and anomaly level, each of which represents a classificatory principle that describes a key characteristic of the nature of data Foorthuis R, (2021).

The first dimension indicates to the data types utilized in characterizing the characteristics of occurrences. It applies to the data properties accountable for the outlier nature of a given outlier type. The second dimension refers to the cardinality of relationships, illustrating what different attributes are interconnected when describing anomalous behavior.

The third dimension pertains to the level of anomaly, which distinguishes among atomic anomalies (individual low-level cases or data points) and aggregate anomalies (groups or collective structures) at Figure 7. It is therefore advised that researchers use the typology to provide clear insight into the functional capabilities of their anomaly detection algorithms by explicitly stating which anomaly type(s) can be detected Foorthuis R, (2018).

3 Research Objective

It's indeed a fascinating topic because it involves the utilization of machines learning and neural network models, which can be either unsupervised or supervised in architecture. The models are rather well known, coming after the breakthrough in machines learning analysis became popular in 2016. Machines learning have been the most used model for linear or classification datasets for a long time. The research problem focuses on anomalies in charging session datasets as it is the second leading bad customer experience of charging session.

3.1 Research methods and the methodology

The research question is related to how these models can be used to perform anomaly classification analysis and how the results compare to those obtained using machines learning or neural network models. Another aspect of the research is to explore whether pre-training models on the different dataset gives different results than using the supervised or unsupervised models.

The method used was not easy to identify, but it appears to be a controlled experiment or focus group dataset conducted in a train and test environments. The methodology used is action research, where a problem is identified and various labeled solutions are available for evaluation.

The main experiments were used a target Charging Details Records (CDR) dataset that included tabular dataset, what totally consists of 39 features. The dataset was separated into training and validation sets, and the experimental data was evaluated utilizing the test set. However, no further information was provided regarding the collection of the dataset. It would be seen that the data is trustworthy and proper when it is used the customer charging sessions invoices.

The research method used was a case study, which is a good research approach in applied science. Priya (2021) presents case study as a research strategy. I personally like the case study method because it delves in-depth into how and why things happen.

It's a very practical approach to investigating problems from both sides of the research. Based on Yin (2009), a case study is a detailed investigation that explores a current phenomenon thoroughly within its real-life context, particularly when the distinctions between the phenomenon and its context are not clearly defined.

Choosing a method and methodology is essential in answering the question of 'what'. It is related to what we are researching, as it is the backbone of the tool that needs to be defined in the early stages of developing the thesis. It's an approach to collecting data for the thesis.

The methodology is employed to address the question of 'how'. In the use of a case study, it typically leads to action research or phenomenology, as both methodologies are suitable for examining a problem and its various solutions. The methodology describes how to analyze the data using the selected method. It encompasses choices regarding the study's overarching structure, the methods chosen for research, techniques for gathering data, and procedures for analyzing the data collected.

There are two paradigms in research: quantitative and qualitative. The qualitative paradigm is more focused on exploring a question through methods such often involving in-depth analysis of non-numerical data, without a set hypothesis. The quantitative paradigm, on the other hand, involves questions that contain a hypothesis or predict something, The gathering and examination of numerical data are conducted to address research questions or examine hypotheses.

When using data in a case study, the data credibility needs to be assessed in all stages if it is being used as a quantitative question. Quantitative research focuses on facts, large samples, and the formulation and testing of hypotheses. Qualitative research, on the other hand, focuses on understanding the meaning and gaining a holistic understanding of each unique situation.

Sample research questions are designed to discover the information that wants to be addressed. Qualitative research can help with an exploratory approach, while quantitative research focuses on correlations, trends, and relationships. The research question in this case was more quantitative, focused on the behavior and experience of charging session dataset.

In this thesis was utilized the case study methodology. It was noted that a research methodology closely related to case studies is action research, which seemed like a confusing rationale because the case study method primarily answers the question "what," while the methodology addresses the question "how." This approach of merging the two topics together seemed during the research study.

Alternatively, combining the case study method with phenomenology as the methodology may work well since the analyzed data is based on lived or prior experiences. Based on the research question of exploring the behavior and experience of charging data related to learning techniques, In this thesis appears to be may be more quantitative in nature, rather than fitting into both of quantitative or qualitative paradigms. Here is not qualitative elements, like observation, subjective and purposeful data.

3.2 Research ethic

Research ethics consist of a set of written responsibilities, principles, and guidelines that govern the conduct of research in a proper way. The purpose of research ethics is to secure that the research is conducted in a responsible, transparent, and trustworthy manner. It also encompasses issues such as disregard, misconduct, conflicts of interest, and the responsible handling,

presenting, and storage of data. Adherence to ethical principles is essential to the credibility and integrity of research, and is required by various regulatory bodies. In this case company fully covers European regulation General Data Protection Regulation (GDPR) supposes organizations to pay attention to privacy (European Commission, 2018).

There are some aspects what might be related to ethical principles, confidentiality, and data security. The data relates to individuals whom are using the case company services, therefore it has been anonymized to ensure that no individual can be recognised from the dataset. All personally identifiable information is represented by anonymous identification numbers. Charging session data typically does not include any privately sensitive information or people-related data.

Instead, it primarily comprises business-critical information related to charging activities, such as usage statistics, session behavior, and energy consumption what might related to data protection. However, even though the data may not contain personal information, it's important to adhere to ethical principles and privacy regulations when handling and analyzing this type of data. Indeed, researchers are increasingly focusing on these challenges because individuals are becoming more conscious of their human dignity, particularly concerning their self identity and data (Fabiano, 2019). Customer specific informations such as credit card numbers, the birth dates, and government ID numbers are required to process transactions, but these were not processed during this case study.

4 Background

4.1 Case company

Virta’s digital electric vehicle charging platform is used by over 1,000 private and public companies and organisations in retail, hotel, real estate, parking, petrol retail, automotive, and energy industries. These customers operate over 75,000 chargers in 35 countries, forming the “Powered by Virta” network. Through roaming, electric vehicle drivers can access over 350,000 charging points in over 65 countries (Virta, 2024).

4.2 The electric vehicle charging protocols

Protocols play a crucial role in facilitating effective communication within electric vehicle charging ecosystem. They are essential for ensuring interoperability, standardization, and secure communication among various components. These protocols are following well-know open-source principle Figure 8 below (Tridens Technology, 2023).

The standard was established by the Open Charge Alliance (OCA) for electric vehicle infrastructure market, and it has become a vital requirement to ensure interoperability among electric vehicle charging manufacturers, charging network operators, and software providers (OCA, 2024).

The Open Charge Point Protocol (OCPP) serves as an electric vehicle charging communication protocol connecting an electric vehicle charging point with a charging point management system (CPMS). This protocol, available for free use, has gained widespread adoption among various vendors on a global scale.

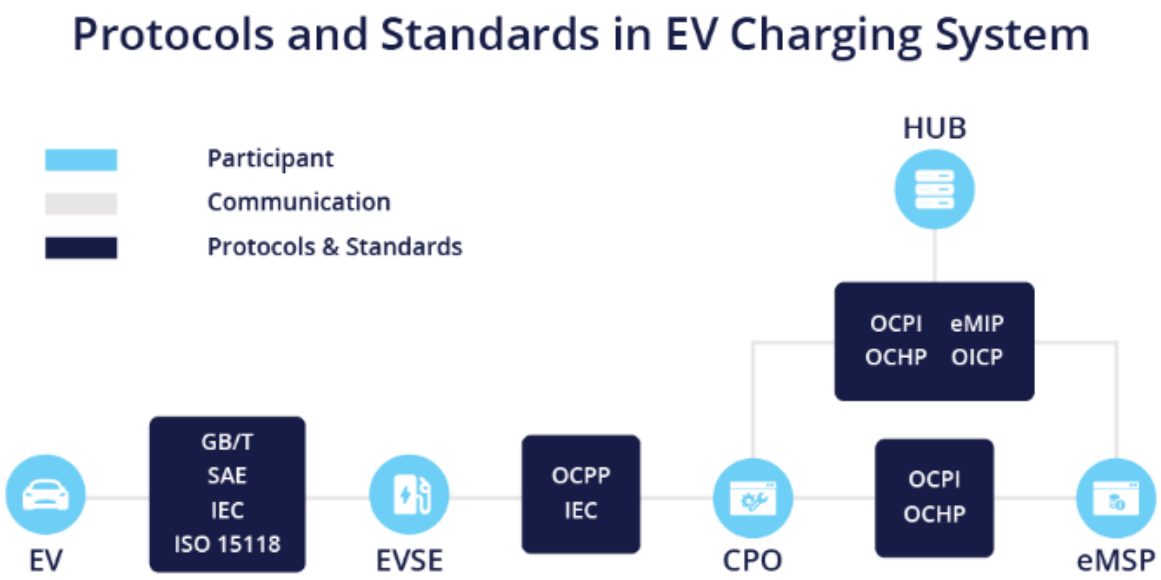


Figure 8: The picture of protocols and standards in the electric vehicle charging system (Tridens Technology, 2023)

The data from these charging sessions are essential for accurately billing the charging session to an electric vehicle driver. Charge sessions involves recording data in CDRs, often facilitated through the use of CDRs in the Open Charge Point Interface (OCPI). OCPP focuses on the real-time interaction and control of charging infrastructure, ensuring interoperability between different vendors' charging stations and backend systems. CDRs primarily serve as a means to track and document charging activities, enabling accurate billing for energy consumed, analyzing usage patterns, optimizing charging infrastructure, and ensuring compliance with regulations.

CDRs are records that contain detailed information about the usage incurred during a charge session. CDRs typically include details such as the start time, end time, duration, source, destination, type of service, and any additional pertinent information related to the activity.

The CDR is the sole billing-relevant module, when CDRs are transmitted from the charging point operator to e-mobility service provider after the charging session has concluded. Although there is no requirement to send CDRs in real-time, it is seen as good practice to send them as soon as possible (OCPI, 2021).

When a charge session occurs, the backend system generates a corresponding CDR to log the specifics of the session. These records are then used for various purposes:

Billing: CDRs are crucial for accurately billing customers based on their usage of services. They provide the necessary data to calculate charges according to predefined rates or pricing plans.

Analytical Purposes: CDRs are also valuable for analyzing usage patterns, trends, and network performance. Telecom companies, for instance, may analyze CDRs to optimize network resources, identify potential fraud, or tailor services to better meet customer needs.

Legal Compliance: CDRs may also be used for regulatory compliance purposes, such as meeting requirements for data retention or providing evidence in legal proceedings.

Charge sessions involve the usage of services that incur charges, and CDRs are the detailed records of that usage, which are essential for billing, analysis, and compliance purposes.

In some cases, electric vehicle charging sessions are paid for through a postpaid billing system what is provided by the charging service provider. It means that users are not required to pay for their charging sessions upfront but instead receive a bill for their usage at a later date.

In the context of electric vehicle charging session, a postpaid bill would be issued to the customer after they have charged their vehicle, detailing the

amount of electricity consumed and the corresponding charges incurred. The customer would then typically have a specified period to pay the bill, typically on a monthly basis. The charging service provider tracks the usage during each charging session and then compiles this information into a bill that is sent to the user at regular intervals.

Postpaid billing offers convenience for users, as they can charge their vehicles without needing to manage payments for each individual session. It also allows for easier tracking of charging expenses over time. Postpaid billing differs from prepaid or one-time payments, where customers pay in advance or just after charge session was provided for a certain amount of service. In postpaid billing, customers use the service first and are billed later based on their usage.

5 Case Experiment

5.1 Experimental approach to case study

In most cases, a mere data rows is insufficient to train a machine for tasks such as classification or prediction. Herman et al. (2013) describe data science briefly as the art of turning data into actions. The required amount of data depend on the task's complexity, but there is a general belief that an ample number of samples representing diverse situations is crucial. Marr (2017), states that “Highlights three core areas in that data has the most value for business: better decision making, improved operations, and the possibility to monetize data”. More data is generally considered better. However, it's important to factor in the learning and runtime for larger datasets, as this aspect often contributes to increased infrastructure costs. This is a critical juncture, as inadequate or inaccurate data can result in flawed machine learning models (Yang et al., 2023).

“Building a data strategy often starts with the creation of one or more data use cases based on the most pressing business issues”, (Fleckenstein et al., 2018). This research study dataset was originally recorded within a company's backend databases and subsequently extracted for additional data processing. Comprehensive data analysis was conducted during the preprocessing phase. Various supervised and unsupervised model selections were employed. However, more optimized fine-tuning of hyperparameters was not pursued. The models were executed on the entire dataset features. In the case of the unsupervised model, it underwent 100 or 200 epochs without any early stopping or other model tuning mechanisms.

The research study was utilized a combination of a controlled experiment and a case study, when the dataset was preprocessed to remove duplicate data rows using manual work. The research questions were more aligned with a quantitative research paradigm, and the data samples were collected with care. There were focused on facts and formulated hypotheses with supervised and unsupervised models and it was tested.

For this reseach study was utilize the Visual Studio (VS) Code Integrated Development Environment (IDE), a freely available collaborative interface offered by Microsoft. Leveraging VS Code enables us to create and run Python code within an interactive environment. In here was exclusively utilize only Python libraries.

In the book Leekha (2021) outlines several rationales for selecting Python as the preferred programming language, given the prevailing trend of leveraging Python for machine learning applications, its significance is notably heightened. The dataset was analyzed using Python, as it is a high-level language renowned for its dynamic typing capabilities. Python was selected as the preferred coding language due to its widespread usage in machine learning applications (Leekha, 2021). Python is a versatile, general-purpose language equipped with numerous libraries, making it a staple in both business and academic projects worldwide. Python was chosen as the foundation of the project because most machine learning tools are built on it, making it a dependable baseline coding language (Leekha, 2021). With extensive documentation, guides, tutorials, and a vibrant community, Python offers robust support and fosters rapid development and problem-solving. Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

5.2 Workflow of data analytics

Here is provide a textual description of each step which can be seen as a guide to see visual representations and performance results of datasets. It is quite similar than Cross-Industry standard process for data mining (CRISP-DM) (Chapman et al., 2000). Figure 9 according to Martinez-Plumed (2021) the standard CRISP-DM process typically includes the following stages: business understanding, data understanding, data preparation, modeling, evaluation, and deployment.

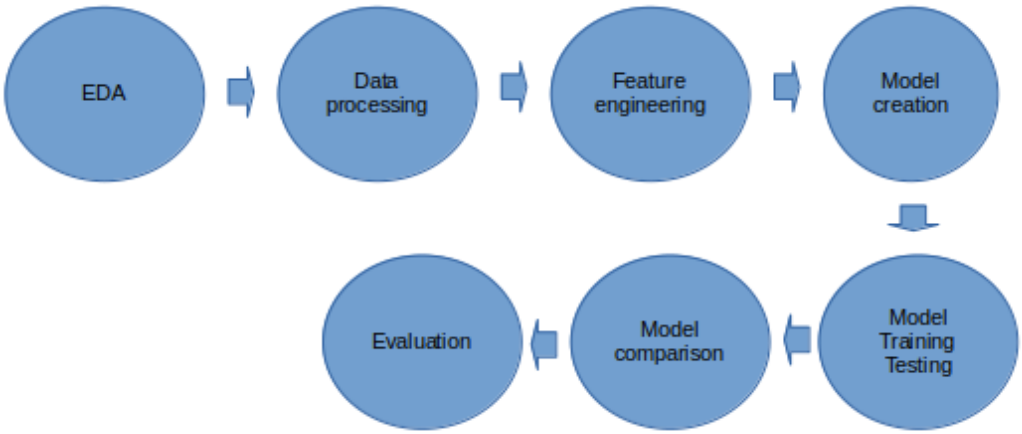


Figure 9: Visual representations of dataset processing

Exploratory Data Analysis (EDA) create visualizations to discover null values, abnormalities, and any patterns in the dataset. Use library tools like Matplotlib, Pandas, or NumPy in Python to generate visualizations.

Feature preprocessing represent your data in a more meaningful way by creating new features derived from existing ones. Visualize feature distributions before and after engineering to understand the transformation. Use techniques like one-hot encoding, binning, scaling, etc., and visualize the effects on the data.

Feature engineering with Principle Component Analysis (PCA) was exploited to converting a large number of attributes into a more manageable one, whilst preserving maturity of the insight from the original dataset. This was undertaken with the aim of simplifying data exploration and visualization of feature importance scores obtained from various techniques. Additionally, correlation matrices were plotted to identify highly correlated features and evaluate the performance of different classifiers. PCA is a dimensionality reduction and machine learning method used to simplify a large data set into a smaller set while still maintaining significant patterns and trends (Jain et al. 2022).

Model Creation visualize the performance of different models employing different gauges such as accuracy, precision, recall, etc., on training and validation data by creating visualizations like bar charts, line plots. Plot confusion and classification matrixes to understand the model's ability as the training data size increases. Visualize decision boundaries for classification models.

Evaluation will focuses on model gauges such as accuracy and other gauges. However, in the actual evaluation phase, the emphasis shifts towards gaining a broader understanding of whether the solution aligns with original business benefit. A pre-decision is made regarding whether the case study proceeds to the deployment stage or if further iterations are necessary to enhance the models performance by hyperparameters before it.

5.3 Exploratory Data Analysis

EDA is an analysis approach that identifies general patterns in the data, these patterns include outliers and features of the data that might be unexpected (EPA, 2024). EDA serves as a critical initial stage in any data analysis process, as illustrated in Table 2. In accordance with them, the principal focus of data assessment is to amplify determination throughout the organization, thereby directly contributing to the success of the business (Provost et al., 2013A).

Table 2: Data types

features	types	nunique	missing
f1	int64	2	0
f2	int64	4	0
f3	int64	7605	0
f4	float64	69170	0
f5	object	1	0
f6	object	77	0
f7	object	12	0
f8	object	2758	39
f9	object	2	652
f10	object	4	0
f11	object	3	0
f12	object	4	0
f13	object	4	0
f14	object	9112	753
f15	float64	7610	1149
f16	float64	7724	1149
f17	object	11898	35171
f18	object	2	652
f19	object	2	0
f20	float64	10	0
f21	object	25	351
f22	float64	31424	0
f23	float64	26564	0
f24	float64	9224	0
f25	object	9	0
f26	object	9	16
f27	float64	10166	0
f28	object	3	301698
f29	object	13	19793
f30	object	72	14458
f31	object	129	0
f32	datetime64[ns]	2550	0
f33	int64	3314	0
f34	datetime64[ns]	1051	10717
f35	int64	27	0
f36	int64	2	0
f37	float64	2	0
f38	object	7	0
f39	object	12	0

EDA presents a lot of helpful insights about the data. EDA holds significant importance in the realm of data science and machine learning as it offers effective means to delve into unfamiliar datasets using various analysis techniques (Milo et al., 2020). The utilized data originates from the backend dataset system of a Finnish company known as Liikennevirta Oy. The dataset comprises 39 columns in total, denoted as Table 2 above. Each row represents a transaction made by an electric vehicle driver, with various features derived from the CDRs dataset.

The dataset includes information on the types, uniqueness, and missing values per feature, as outlined Table 2. Features with the highest number of missing values were mitigated, while the remaining ones were replaced with zero values to facilitate further analysis.

Hence, the outcomes of findings from practical data analyses might prove inaccurate and unpredictable. Hence, a profound comprehension of anomalies is necessary to ascertain whether the identified cases genuinely qualify as anomalies. It is particularly important for unsupervised outlier techniques, when them are frequently applied without known labeled data. As Shoresh et al., (2012) explains that once cognition on row-level operations is gathered, it would be utilized to generate illustrations that provide enhanced advantages to the operator, making outliers and anomalies easier to identify. In the book Myatt et al. (2014) delves into the topic of data exploration, examining its intricacies and significance.

5.4 Data preprocessing

Sivula (2021) then depicts the third phase subsequent to problem identification and data acquisition as preprocessing the data. This process is undertaken to render the data suitable for use in machines learning applications.

To ensure the security of the implementation and the ethical handling of data, compliance with the General Data Protection Regulation (GDPR) was essential (European Parliament & Council, 2016) and the upcoming European Union (EU) standard for AI solutions (European Commission, 2021), several meetings were conducted with the information security officers and the technology team.

Research by Yang et al. (2023) integrates an analysis phase into the preprocessing stage. This step is taken to further refine and optimize the data, narrowing it down to a subset that aligns with the intended purpose. There were created following two separate datasets, one numerical data as float64 type with 12 dimensions and other categorical data as object type with 17 dimensions Figure 10. Prior to analysis, standardization of the categorical

features was deemed necessary, necessitating the application of the OneHotEncoder technique, resulting in the creation of new columns and expanding the dataset to 383. Subsequently, the tabular and standardized categorical datasets were merged into a unified dataset of dimensions 395. Machines learning algorithms often struggle to perform well when if input attributes exhibit varying scales. To address this issue, it was employed features normalization, which involves rescaling all numerical attributes by the Standard Scaler.

It is typical option of processing the data to divide it into a training set and a test set. It could be trained the model employing the training set and then assess its performance utilizing the test set. However, if it will be only evaluated the model once, there may be uncertain whether the good results are due to chance or actual effectiveness. Therefore it was aim to assess the model multiple times to enhance confidence in its design. The random_state is a parameter in train_test_split that dictates the random number generator for shuffling data prior to splitting. Setting it to None, it do not allows for different randomizations with each run. While setting it to a specific numerical value then it ensures consistent randomization across runs. If random sampling is employed to separate the dataset into the training and test sets for a binary classification problem, there's a risk of imbalanced class distributions between the two sets. For instance, if all occurrences of the positive class end up in the test set while the training set holds only occurrences of the negative class or opposite, the model may not learn effectively, leading to poor accuracy scores due to an inability to generalize well to unseen data. It was used the Stratified sampling techniques with PyTorch model, which aim to maintain class balance in both training and validation sets.

Furthermore, to ensure the coherence and robustness of the subsequent analytical procedures, all features underwent scaling using the Standard Scaler method, resulting in a standardized dataset of the same dimensions 395 Figure 10. Simultaneously, the target variable was isolated into its distinct dataset of dimensions one.

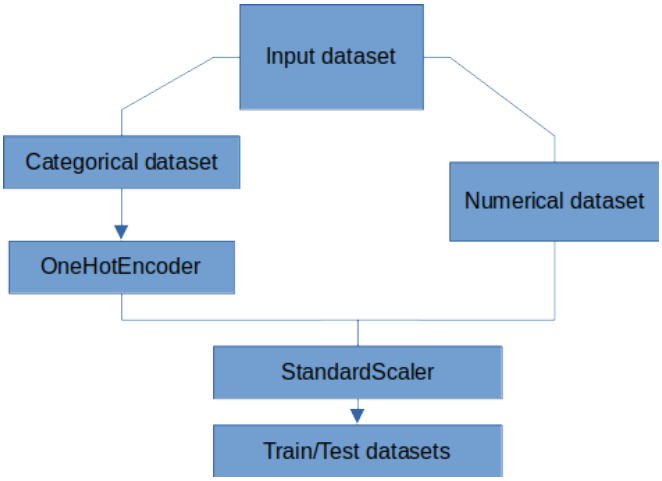


Figure 10: Dataset structure

Yang et al. (2023) highlights the impact of data quantity on the quality of machine learning models. Performing preprocessing is a pivotal aspect of the process, as it may unveil issues that could impede subsequent steps.
Since data serves as the fundamental root of insight for the recipe, it must be in a suitable state to render the evolving machines learning model workable for applications in back of the testing phase. Inadequate data can drive to the production of similar results by the evolving model. In the journal Yang et al. (2023) also covers categories of inaccurate or incomplete data that need to be considered. During dataset preprocessing and under further investigation comprises that remaining dataset contains a relatively small proportion of missing values compared to the total values, which isn't significant overall. Figure 11. There is still one feature what is a significant portion of all feature values are missing. However, for other features, the missing values only represent a very small portion of all missing data.

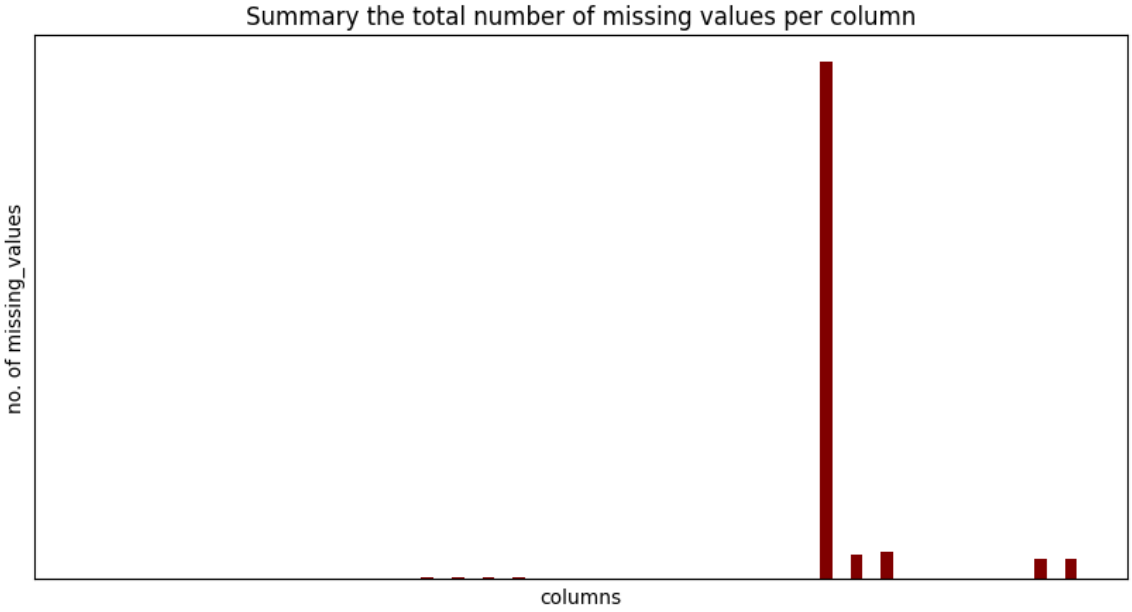


Figure 11: Total number of missing values per feature

You can see below the normal transactions versus anomalous transactions class distribution on a bar graph Figure 12. and separate them into their own datasets, the first needs to calculate the fractions of each type of transaction. Then, was created the bar graph and separate the transactions into their respective datasets.

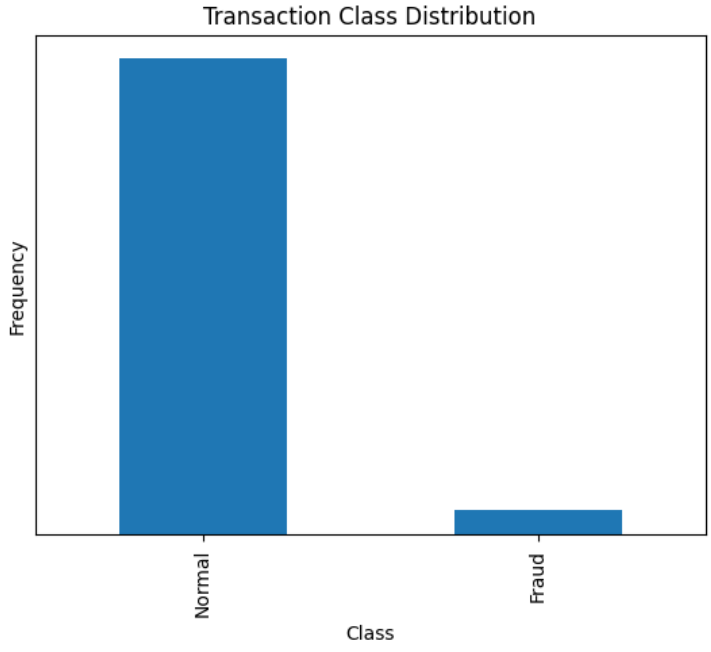


Figure 12: Transaction class distribution

Damaged or inconsistent data can be addressed or rectified to some extent while EDA phase, but there are limits to what can be done (Yang et al., 2023). Whole dataset consists reasonable amount of anomalous data while also providing separate datasets for evaluating its performance on both non-anomalous and anomalous examples. In their article Wang et al. (2020) elucidates strategies for addressing corruptions in power systems.

Upon selecting a suitable model that has undergone testing, training, and evaluation, yielding satisfactory results, the initial iteration of the process concludes. Provost et al. (2013) argues that data science encompasses ethics, workflows, and techniques aimed at comprehending phenomena through the computerized analysis of data. Following the preprocessing steps, the dataset was segregated into separate training and testing sets. The training features dataset comprised 70% percent of entries, paired with a corresponding target dataset of dimension. Simultaneously, the testing features dataset contained 30% percent of entries, along with its respective target dataset of dimension.

In their journal Shoresh et al. (2012) explain that data exploration involves a process aimed at comprehending the events or patterns present in the data. Typically, it implies the involvement of individuals the once are acquainted with the data. It provides information to guide the individual, group, or team conducting the appraisal regarding the data's limitations, functioning, and the most effective approach to tailor it to the use case. In an ideal scenario, subject matter experts can offer insights on the relevance of specific fields and rows, distinguishing between what is valuable and what is extraneous.

5.5 Feature Engineering

Feature engineering normally involves the technique of verifying, tweaking, and evolving raw data into features what explains enough variance and are suitable for use in algorithms learning. Regarding the data required by the learning algorithm, it must be sufficiently useful and accurate to effectively represent the schemes and occurrences present in the substance. This stage is frequently referred to as the hyperparameter tuning phase, which can be facili-

tated using approaches like Cartesian grid search (Ding et al. 2020). This study primarily emphasizes PCA to ensure that the selected features cover enough variance to run learning models effectively.

5.5.1 Principal Component Analysis

PCA (Banerjee, 2022) described it as a statistical procedure that employs a technique to amend a set of associated attributes into a set of irrelevant attributes. Alternatively referred to as PCA, is a parametric technique used for investigating the connections among variables. The PCA is a multivariate statistical technique first developed by Pearson (1901) and employed in different research areas. PCA compresses the data and generates a new, smaller subset of dimensions by identifying the principle components of the data points (Jolliffe et al., 2016). During preprocessing phase where noticed that there were quite many features included to existing dataset. The objective of the methodology is to analyse a dataset and extract “the important information [...], to represent it as a set of new orthogonal variables called principal components” (Abdi et al., 2010).

PCA assists us in concentrate the most significant features within a dataset by generating new features, known as principal components. The spread of large datasets can be shrunk by the statistical PCA method while still preserving important relationships between the dimensions in order to make it easier to analyze and to visualize large datasets (Glen, 2023). The new components cover the most of the variability on existing the original data. It is a linear transformation obtained by maximizing the variance in the data in order to retain as much statistical information as possible using the least number of dimensions (Jolliffe et at., 2016). It simplifies data by converting a collection of observations, potentially correlated, into sets of linearly uncorrelated values called principal components.

PCA can be exceptionally valuable, particularly when dealing with numerous predicting features, ranging from hundreds to even thousands. Therefore, if the primary objective is to obtain the most optimal performing model, even at the

cost of sacrificing the interpretation of feature importance, then experimenting with PCA could prove beneficial. In other words, it's the method to decrease the number of features in a dataset while retaining as much critical information as feasible (GeeksforGeeks, 2023).

Using PCA for all feature dataset what it gave about 250 of the PCA instead of the initial 395 features, what can achieve a total explained variance of 95% Figure 13. below.

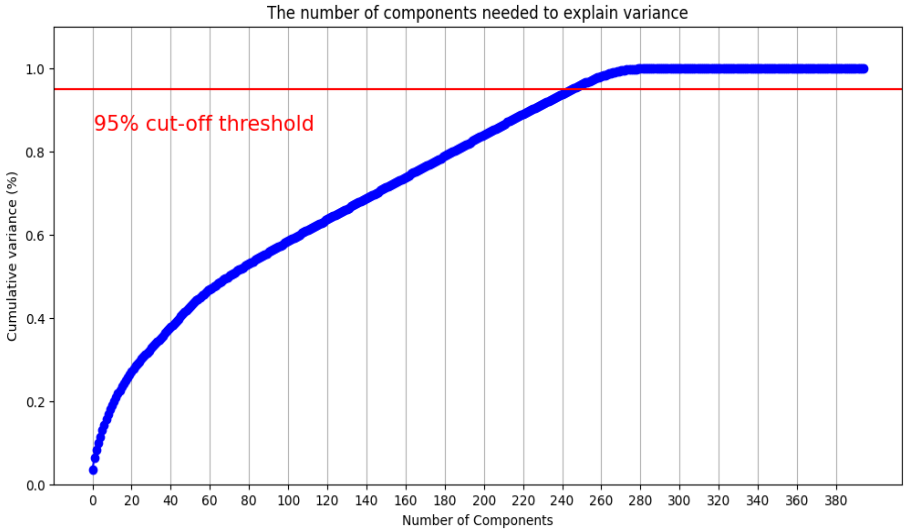


Figure 13: The number of components to explain variance in all data

The scatter plot of the two-component PCA visualizes patterns of anomalies and non-anomalies, as depicted in Figure 14 below. The total explained variance in this plot is 6.4%, and it is represented in two dimensions.

The initial component contains the highest variance in the original data, followed by the second component, which contains the second-highest variance, and so forth (Biswal, 2023). Notably, the anomalies, denoted by the blue class, exhibit distinctiveness, while some overlap is observed among the non-anomalies. Scaling this data would likely yield different results, potentially leading to separated data clusters. Dimensionality reduction helps filter out these unimportant variables. Reducing the dimensionality of data makes it simpler to visualize, facilitating improved insights and pattern recognition (Ewerton, 2022).

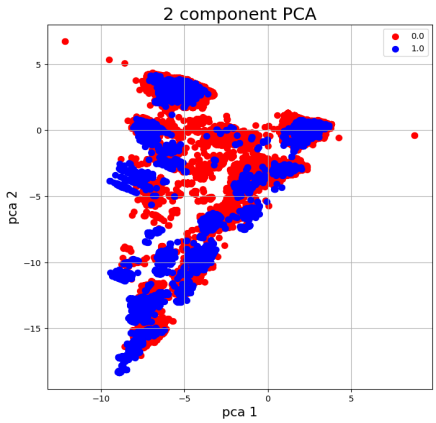


Figure 14: Two component PCA for all dataset

The higher explained variance ratio, the more important the principal component is in explaining the variance of the data.

PCA components = two
Total explained variance = 6.41%
[0.03513907 0.0289619]

Here, we have an array where 3.5% of the fluctuation is described by the first principal component (PC1) and the 2.9% is explained by PC2. Together, they explain 6.41 % of the fluctuation of the data. The explained fluctuation, or eigenvalue, in PCA indicates the proportion of fluctuation that can be assigned to each of the principal components [13.87995414 11.43997108].

The explained variance in PCA helps us understand how much information is retained after dimensionality reduction. It represents the proportion of the original data's diversity captured by each principal component.

5.5.2 Categorical data

Categorical variables categorize data into distinct groups or types. Occurrences of categorical variables are race, sex, age group, and educational level (Yale (2024). Using PCA only for categorical dataset what gave about same 240 features of the PCA instead of the initial 17 features what was transferred by One-

HotEncoder technique to 383 features, what can achieve a total explained variance of 95% Figure 15. below.

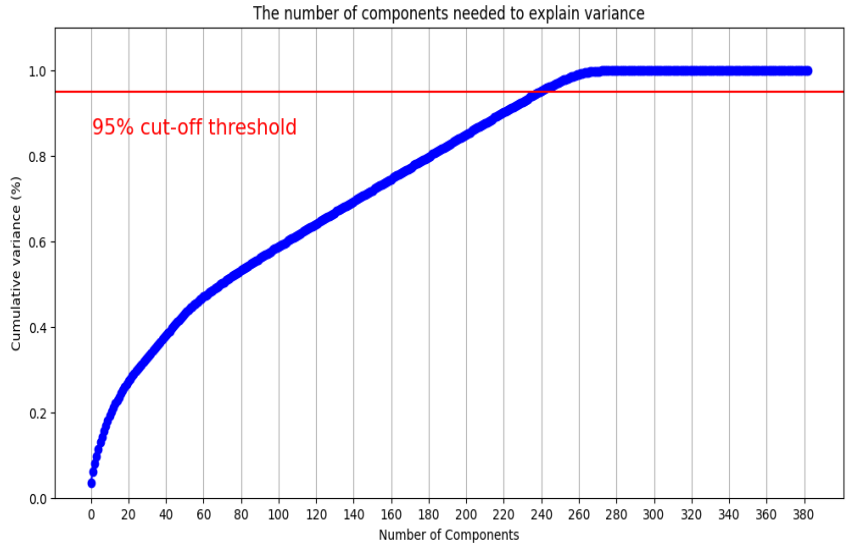


Figure 15: The number of components to explain variance in categorical data.

The scatter plot of the two-component PCA visualizes patterns of anomalies and non-anomalies, as depicted in Figure 16 below. The total explained variance in this plot is 6.23%, and it is represented in two dimensions.

Notably, the entire plot appears to be inverted compared to the entirety of the dataset. Similar patterns persist, indicating that scaling the dataset could potentially result in separate data clusters. The anomalies represented by the blue class are dispersed across the entire scale of the plot Figure 16.

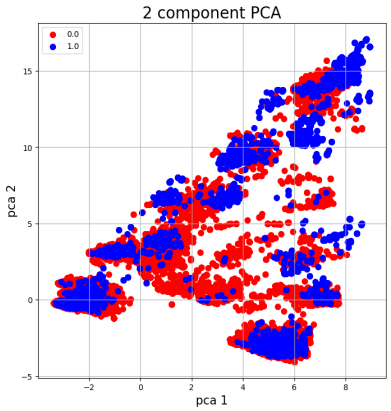


Figure 16: Two component PCA in categorical data.

5.5.3 Numerical data

Using PCA only for numerical dataset what gave seven features of the PCA instead of the initial 12 features what can achieve a total explained variance of 95% figure 17. below.

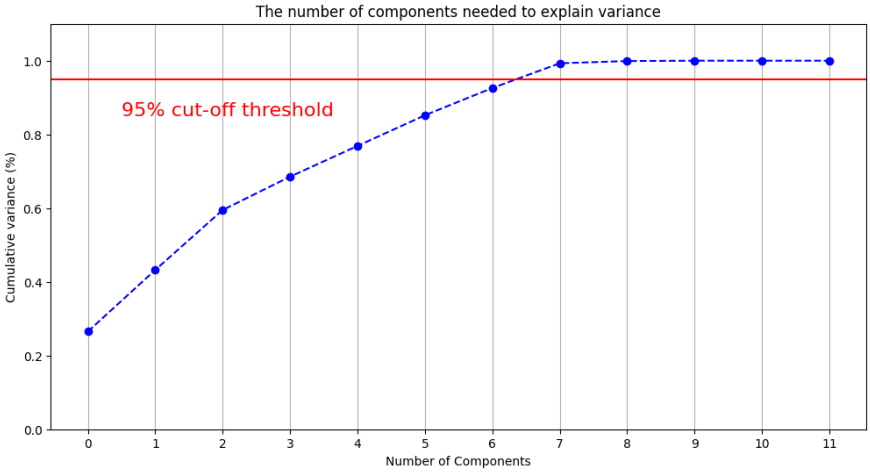


Figure 17: The number of components to explain variance in numerical data.

The scatter plot of the two-component PCA illustrates patterns of anomalies and non-anomalies, as showcased in Figure 18 below. The total explained variance in this plot is 43.21%, which significantly surpasses what both the all and categorical datasets were able to convey.

Notably, there are two distinct clusters of datasets, which exhibit a similar spread of anomalies. This suggests that scaling the dataset could potentially result in two separate data clusters. The anomalies, represented by the blue class, are dispersed across the scale of the plot encompassing these two clusters. This observation implies the presence of two separate groups of features that heavily influence the components Figure 18.

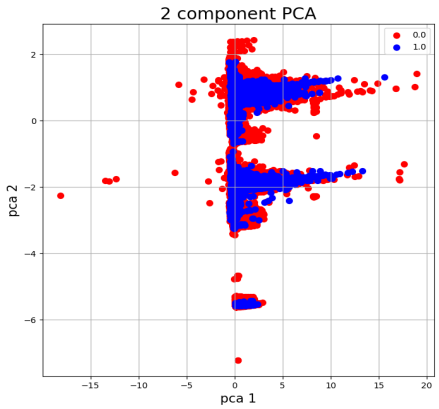


Figure 18: Two component PCA in tabular data

Plotting two PCA components on a dataset reveals certain drawbacks. One issue is that this reduction in dimensionality can result in a new set of variables that capture too little variance of the dataset. This limitation becomes evident when considering the number of features needed to achieve a total explained variance of 95%. Consequently, the transformation leads to a loss of interpretability of the original features, making it more challenging to discern the relationship between the input variables and the outcome of interest in the classification task. Additionally, if the most important features for classification are not adequately captured by the two principal components, it can negatively impact the performance of the classifier.

Another important observation is that PCA assumes linear relationships between variables, which may not always hold true, especially in classification and clustering datasets. If the relationships within the data are non-linear, PCA may fail to effectively capture the underlying structure, resulting in suboptimal outcomes in the classification task.

Moreover, if the objective of the classification task is to interpret the importance of specific original features, applying PCA beforehand may impede this interpretation. This is because PCA transforms the original features into new components, making it more challenging to directly relate the transformed components to the original features and their importance in the classification process.

Ultimately, the decision to utilize PCA before classification should be made after carefully considering the particular attributes of the dataset and the objectives of the analysis. It's crucial to assess whether the benefits of dimensionality reduction outweigh the potential loss of interpretability and assuming linear connections among variables. The decision-making process should be informed by a complete understanding of the dataset and the purposes of the analysis.

In this research study, PCA components were not employed with any supervised or unsupervised classifier models. They were solely used during the visualization process, specifically for plotting two-dimensional scatter plots. Dimensionality mitigation techniques are utilized to address these challenges and obtain the most pertinent information from the data while decreasing its dimensionality (Han, 2012). PCA is indeed a robust approach that can effectively determine the number of features required to cover 95% of the `fluctuation` in the data.

5.6 Supervised Learning analytics

Evaluation typically involves an iterative process characterized by a feedback loop between the results obtained and the model itself. At the core of this process lies the crucial task of selecting an appropriate metric. Numerous metrics exist for evaluating classification models metrics, and the optimal choice depends on various components such as the obstacles at hand, properties of the data, and the expectations from the model. The evaluation metrics for evaluating the proficiency of classification models are key part of analytics. The classification is an vital element of machines learning, involving the duty of unknown data points to predefined classes. In binary classification obstacles, where there are two classes labeled as 0 and 1, the objective is to ascertain which class the given data point belongs to.

A Confusion Matrix (CM) is a tabular depiction illustrating the classification consequences of a binary classifier in categorizing actual and predicted outcomes. It evaluates the true positives, true negatives, FP, and FN for each classification. It provides a short summary of both accurate and not accurate predictions

made by models, along with the distribution among different classes. It enables the computation of several metrics, consisting of accuracy, precision, recall, and F1-score, which appraises the effectiveness of model's proficiency.

The best performing classifier results can be seen at Figure 19 below, observing that Random Forest (RF) has low variance and high accuracy are suitable for further analysis, while Linear Discriminant Analysis displays low accuracy and high variance on training dataset. The data distribution significantly impacts accuracy. A high accuracy result suggests that the model is making a substantial proportion of correct predictions, while a low accuracy result indicates that the model is making a significant number of incorrect predictions.

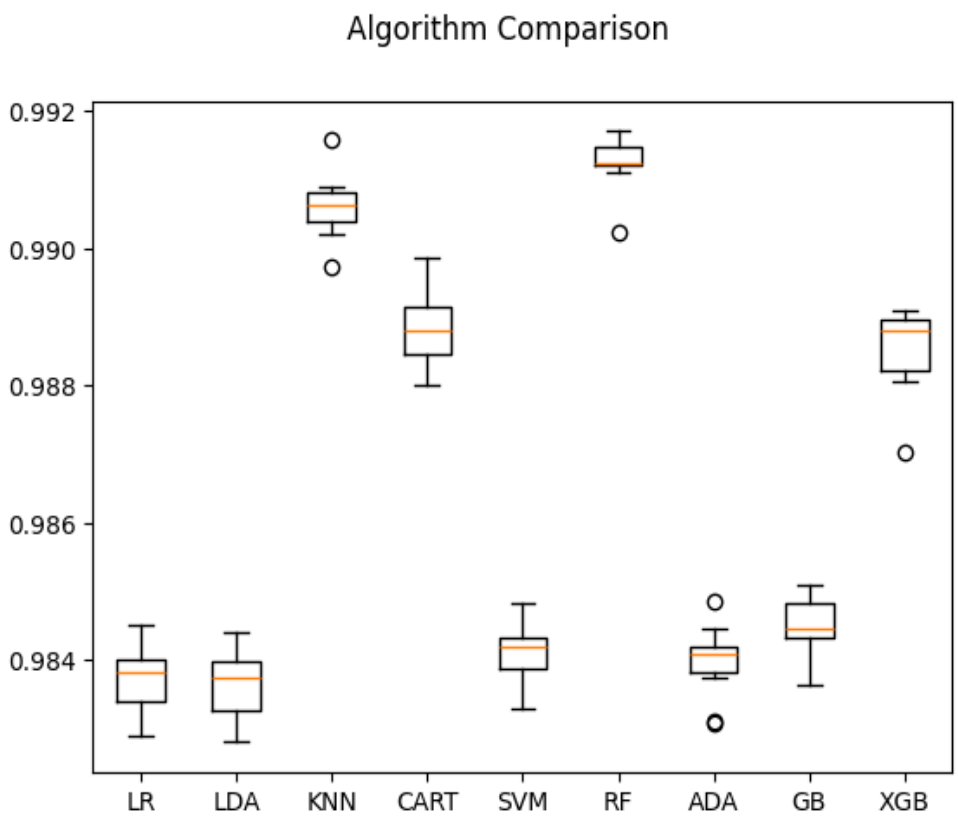


Figure 19: Linear Discriminant Analysis displays

CM with various performance metrics for a RF Classifier can draw several conclusions based Table 4 metrics.

The accuracy score of the classifier is indicating that there were correctly classified approximately 99,15% of the total illustrations in the dataset.

The precision of the classifier indicates that when it anticipates a positive result is correct approximately 99.73% of the time. In other words, out of all illustrations anticipated as positive, were actually positive.

The recall of the classifier is indicating that it properly identifies approximately 82.95% of all actual positive instances, were properly recognized by the classifier.

The F1 score, calculated as the harmonic mean is mix of precision and recall, it is 89.04%. This metric provides an equitable evaluation of precision and recall, take into assessment both FP and FN.

RF classifier performs very well with the provided dataset, achieving high accuracy, precision, recall, and F1 score. However, it's also crucial to take into account the particular context and requirements of the classification problem to determine if these performance metrics are satisfactory.

Additionally, further analysis, such as feature importance, model tuning, and evaluation on unseen data, may be necessary for a complete assessment of the classifier's performance. The feature importance is more complex analysis when PCA was not employed to cut the features into smaller group of features.

Based on below Table 3 output, it can be inferred that the top 20 features such as there are highly predictive of anomalies as totally 66,24%. Additionally, there were several features with importance close to zero when there was totally 395 features, suggesting that they may not significantly contribute to the model and could potentially be excluded. It can be used the feature importance variable to see feature importance scores. Where can see that the most important feature is f1 and least important feature is f21 on list below.

Table 3: Plot Gradient Boosting Classifier
top 20 feature importance.

f1	0.1045
f2	0.0770
f3	0.0653
f4	0.0517
f5	0.0423
f6	0.0385
f7	0.0322
f8	0.0277
f9	0.0262
f10	0.0254
f11	0.0242
f12	0.0233
f13	0.0227
f14	0.0214
f15	0.0162
f16	0.0147
f17	0.0129
f18	0.0129
f19	0.0121
f20	0.0114
f21	0.6624

Below is a solid dataset and foundation on how to interpret and utilize a confusion matrix for classification algorithms in all machines learning with all feature dataset Table 4. The models listed below were ranked based on their performance scores, although there were additional models considered noteworthy but ultimately dropped from further analysis. The matrix scores well understanding which areas where the model has gone wrong, offering direction for correction and it is powerful and frequently employed tool for appraising the effectiveness of a classification model in machine learning.

Table 4: All data scores using different classifiers

--- Decision Tree Classifier ---
Confusion Matrix:
Accuracy Score:0.9891
Precision:0.8823
Recall:0.8986
F1 score:0.8904

--- Random Forest Classifier ---
Confusion Matrix:
Accuracy Score:0.9915
Precision:0.9973
Recall:0.8295
F1 score:0.9057

--- LogisticRegression ---
Confusion Matrix:
Accuracy Score:0.9835
Precision:0.9853
Recall:0.6761
F1 score:0.8019

--- KNeighbors Classifier ---
Confusion Matrix:
Accuracy Score:0.9911
Precision:0.9310
Recall:0.8852
F1 score:0.9075

--- AdaBoostClassifier ---
Confusion Matrix:
Accuracy Score:0.9838
Precision:0.9920
Recall:0.6776
F1 score:0.8052

--- GradientBoostingClassifier ---
Confusion Matrix:
Accuracy Score:0.9842
Precision:0.9979
Recall:0.6824
F1 score:0.8105

--- XGBClassifier ---
Confusion Matrix:
Accuracy Score:0.9884
Precision:0.9890
Recall:0.7743
F1 score:0.8686

Nonetheless, accuracy can be deceptive, particularly for uneven datasets where one class comprises significantly more samples. In such cases, the overall model accuracy tends to be skewed, reflecting its capability in identifying the majority class rather than the specific class of interest. Thus, the accuracy metric may not provide informative insights, particularly if the objective is to effectively detect spam.

Therefore it is a reason to look at other results at same time. RF was not the best performing classifier on all binary class dataset areas Table 4, within the recall 82,95% results, where some instances that were anticipated to belong to the actual positive class were categorized as the false class by the classifier. For some reason, the model is producing more of these mis-classifications compared to a few other models, despite having one of the highest absolute totals for true values. Hence, it was termed recall as the model's prediction of "negative" was incorrect when Decision Tree was give a bit better results with recall 89,86%. This results warrants further investigation to understand the source of the skew and its underlying reasons.

Additionally, it's crucial to assess the potential cost impact on the business if the number of false is high, potentially necessitating manual verification by personnel before proceeding with false negative results.

Likewise, within precision result samples were expected to belong to the true negative category but were misallocated as "positive" by the classifier. The Gradient Boosting model yielded much less samples classified as false positives, which represents a slight enhancement assessed to the precision of the RF model. It refers to a sample that is actually from the negative category but is incorrectly classified as the positive category. These instances are termed "False Positives." It can be seen precision results that RF precision results was 99,73% when Gradient Boosting was 99,79% results what is a bit more correct than RF.

Evaluating RF model closely involves analyzing these false positive and false negative distinct numbers from the matrix to gain deeper insights into its performance to compared to other better performing models. The false positive result achieved is indeed impressive, surpassing the performance of other models in the dataset.

A classification report gives a textual overview presenting key metrics for every class within a machine learning model at Table 5. Typically, it comprises precision, recall, F1-score, and support values for each class, alongside the weighted average of these metrics across all classes.

It provides an more breakdown of your model's performance across individual classes, revealing how it manages the harmony between precision and recall. Additionally, it displays the percentage of instances for each class, offering insights into class imbalances or dataset sizes. If you look at RF classification reports Table 5 and focus on more class 1.0 results what is anomaly detection of target feature. It can be seen that recall results, it not detecting all positive instances on this class which actually belong to it correctly where the biggest difference compared to sharpness is FN instances.

One potential reason could be the insufficient amount of this class data, which may have limited the model's proficiency to learn and expand adequately from that class.

Table 5: Classification reports

```
==============================
Model: XGBoost
Classification Report:
              precision    recall  f1-score
         0.0       0.99      1.00      0.99
         1.0       0.99      0.77      0.87
    accuracy                           0.99
   macro avg       0.99      0.89      0.93
weighted avg       0.99      0.99      0.99
==============================
Model: RandomForest
Classification Report:
              precision    recall  f1-score
         0.0       0.99      1.00      1.00
         1.0       1.00      0.83      0.91
    accuracy                           0.99
   macro avg       0.99      0.91      0.95
weighted avg       0.99      0.99      0.99
==============================
Model: GradientBoosting
Classification Report:
              precision    recall  f1-score
         0.0       0.98      1.00      0.99
         1.0       1.00      0.68      0.81
    accuracy                           0.98
   macro avg       0.99      0.84      0.90
weighted avg       0.98      0.98      0.98
==============================
Model: KNeighbors
Classification Report:
              precision    recall  f1-score
         0.0       0.99      1.00      1.00
         1.0       0.93      0.89      0.91
    accuracy                           0.99
   macro avg       0.96      0.94      0.95
weighted avg       0.99      0.99      0.99
==============================
Model: Neural Network
Classification Report:
              precision    recall  f1-score
         0.0       0.99      1.00      0.99
         1.0       0.97      0.72      0.83
    accuracy                           0.99
   macro avg       0.98      0.86      0.91
weighted avg       0.99      0.99      0.98
==============================
```

The primary distinction between CM and a classification report lies in their presentation and focus. CM visually displays the actual and predicted counts for each class, providing an intuitive overview. Otherwise, a classification report presents calculated metrics for each class in a numerical and analytical

format. While CM aids in identifying error types and sources, a classification report assists in evaluating the quality and reliability of the model's proficiency.

A classification report is better suited for tasks like measuring precision, recall, F1-score, and support for each class; assessing the agreement between precision and recall; evaluating the weighted average of metrics across all classes; or reporting results using tables, numbers, or text.

Precision relates to the correct positive predictions impact to the total positive predictions. A high precision grade means that the model effectively identifies positive instances with accuracy, whereas a low precision grade implies that the model makes an excessive number of FP predictions.

Recall indicates the portion of expected positive observations through of all the correct positive observations. It is also mentioned to as delicacy index or true positive rate, is a performance gauge that evaluates the ratio of positive samples appropriately identified by a binary classification model among of all the actual positive instances.

The F1 score is a weighted combination of precision and recall. It is a performance gauge that joins precision and recall to furnish a full appraisal of a binary classification model.

When exploring the distribution and frequency of predictions and outcomes, detecting misclassifications between classes, reviewing the performance of various models or algorithms, a confusion matrix is the ideal tool.

Figure 20 below illustrating concepts in PCA two dimensions is straightforward with all different visualized classifier results. Typically, a scatter plot with x- and y-axes suffices for visual representation in two dimensions. Dimensionality reduction techniques like PCA become valuable in such scenarios. It allows to reduce the dimensionality to two, facilitating visualization. However, it's crucial to note that PCA is sensitive to scale, resulting in variations in colors and

behaviors across different models. Then is good to remember that below Figures 20 explain 6.41 % of the variance of the dataset. It is apparent how different classifiers are identifying anomalies and why the performance scores may not be noteworthy Figure 20, proves well that RF was the best performing classifier.

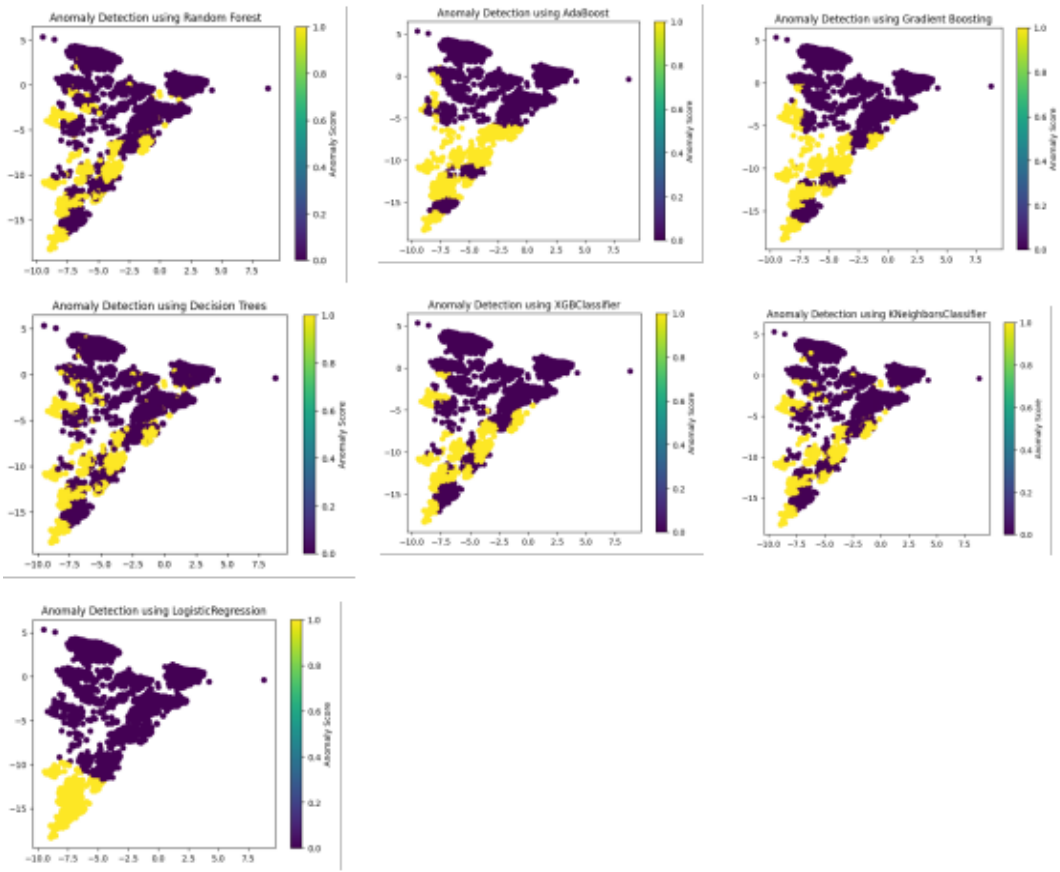


Figure 20: All classifiers for anomaly detection and visualized results.

In this dataset Table 6 below, comprised solely of only numerical features, the results notably reduced when the number of features was shrank to 12 from the original total of 395 features. Accuracy is not anymore same level than with all dataset features. RF model was showing recall results Table 6, what indicate almost double amount of false negative samples than all features dataset. Likewise, within precision false samples were expected to belong to the negative class but were misassigned as "positive" by the classifier what is even four times more samples than all features dataset.

Table 6: Numerical data scores using different classifier.

--- Decision Tree Classifier ---	--- Random Forest Classifier ---	--- LogisticRegression ---
Confusion Matrix:	Confusion Matrix:	Confusion Matrix:
Accuracy Score:0.9759	Accuracy Score:0.9853	Accuracy Score:0.9817
Precision:0.7477	Precision:0.9868	Precision:0.9662
Recall:0.7746	Recall:0.7117	Recall:0.6526
F1 score:0.7609	F1 score:0.8270	F1 score:0.7790
--- KNeighbors Classifier ---	--- AdaBoostClassifier ---	--- GradientBoostingClassifier ---
Confusion Matrix:	Confusion Matrix:	Confusion Matrix:
Accuracy Score:0.9828	Accuracy Score:0.9837	Accuracy Score:0.9839
Precision:0.9376	Precision:0.9962	Precision:0.9975
Recall:0.6986	Recall:0.6729	Recall:0.6756
F1 score:0.8007	F1 score:0.8032	F1 score:0.8056
--- XGBClassifier ---		
Confusion Matrix:		
Accuracy Score:0.9846		
Precision:0.9876		
Recall:0.6974		
F1 score:0.8175		

In the dataset show in Table 7, which consists entirely of categorical features, reducing the categorical of features to 383 from the original total of 395 did not result in significant improvements. In fact, accuracy and precision were slightly lower compared to the dataset with numerical features. RF model was showing recall 82,02% results Table 7, what indicates almost same amount of false negative samples than all features dataset. Likewise, within precision result tells that some samples expected to belong to the negative class but were misassigned as "positive" by the classifier what is even four hundred times more samples than all features dataset.

Table 7: Categorical data scores using different classifiers.

--- Decision Tree Classifier --- Confusion Matrix: Accuracy Score:0.9865 Precision:0.9005 Recall:0.8178 F1 score:0.8572	--- Random Forest Classifier --- Confusion Matrix: Accuracy Score:0.9873 Precision:0.9135 Recall:0.8202 F1 score:0.8644	--- LogisticRegression --- Confusion Matrix: Accuracy Score:0.9835 Precision:0.9862 Recall:0.6758 F1 score:0.8020
--- KNeighbors Classifier --- Confusion Matrix: Accuracy Score:0.9851 Precision:0.8828 Recall:0.8064 F1 score:0.8429	--- AdaBoostClassifier --- Confusion Matrix: Accuracy Score:0.9835 Precision:0.9875 Recall:0.6747 F1 score:0.8016	--- GradientBoostingClassifier --- Confusion Matrix: Accuracy Score:0.9837 Precision:0.9862 Recall:0.6791 F1 score:0.8043
--- XGBClassifier --- Confusion Matrix: Accuracy Score:0.9851 Precision:0.9744 Recall:0.7166 F1 score:0.8258		

Evaluation metrics play a decisive role in appraising the proficiency of different classification models. With various metrics at our disposal, the selection of the most suitable one(s) depend on several elements including the specific challenge being addressed, the implications of FP and FN, and the degree of class imbalance in the dataset. It's imperative to choose metrics that align with the objectives and boundaries of the task at hand, securing a comprehensive appraisal of the model's proficiency.

Accuracy is indeed the almost commonly used appraisal metric, but its reliability can be compromised when dealing with imbalanced classes. In such cases, precision and recall offer valuable insights, especially when the cost of incorrect positives and incorrect negatives varies. Precision prioritizes minimizing incorrect positives, whereas recall focuses on minimizing incorrect negatives. The F1-score, a harmonic mean of precision and recall, offers a neutral judgment which considers two metrics as one value.

Determining the right evaluation metrics is crucial for developing and refining classification models. It's crucial to select the metrics that are well-suited to the specific problem being addressed and offer a thorough evaluation of model performance. By carefully considering the objectives and nuances of the problem, it can ensure that choice of evaluation metrics provides meaningful insights into the effectiveness of the classification model.

5.7 Unsupervised Learning analytics

There are numerous unsupervised learning techniques available, including neural networks, reinforcement learning, and clustering. The option of algorithm varies on the properties and structure of your data. Unsupervised learning applications can be broadly sorted into two main sections: clustering and association, see at Table 1 comparison of different surveys to other related survey articles (Chandola et al., 2009).

The objective of association is to uncover and understand representative rules within the dataset, such as the observation that customers who purchase product A also often buy product B. Association analysis can aid in the development of effective cross-selling, upselling, or observing recommendation strategies, as well as in discovering product groups, pricing, and marketing promotion opportunities. It is important to remind that association analysis alone does not provide guidance on how to split on customers according to their attributes or behaviors. To uncover hidden relationships between variables, association analysis is employed.

Clustering applications aim to identify inherent groupings within the data, like distinguishing customer segments based on their purchasing behavior. The clustering differs from classification in that it doesn't include assigning predefined labels or predicting values. Clustering, on the other hand, encompasses sorting data nodes together based on the consistency of their attributes. The aim is to uncover inherent trends and interactions within the data. Clustering is especially beneficial when engaging with unlabeled data and pursuing to uncover hidden structures within it.

Neural network are notably effective when a non-linear interactions exists among the input features and the output variable. The networks utilize randomly initialized weights and biases along with differentiable non-linear functions, which are continuous in nature. Through this mechanism, the network predicts the output variable, often denoted as predicted value.

5.7.1 Clustering

If you're uncertain about which features to utilize for your machine unsupervised learning model, clustering can reveal patterns that help identify significant aspects within the data. Clustering is particularly beneficial for investigating unfamiliar datasets. While it may require some experimentation to determine the most effective clustering algorithm, the insights gained are invaluable. It may uncover connections and relationships that were previously unforeseen. The computational complexity of an anomaly detection technique is a crucial factor, particularly while employed to real-world domains, techniques such as clustering-based methods may have high training times, testing is typically rapid. (Chandola et al., 2009).

Clustering is one of an unsupervised algorithm, therefore it doesn't require a target feature for training. Instead, it learns directly from the data and the relationships between features, grouping similar points into clusters based on spatial distance. It's vital to note that all features should be on the same scale before applying a clustering technique. The cluster grouping can be done by selecting the best 'k' value using like elbow method. K-means clustering is useful, like (Patel, 2023) wrote that It can be sensitive to the initial guess, outliers can impact the results therefore it assumes round clusters and prior expertise of the number of clusters is necessary. This section will delve into the five most renowned and significant clustering algorithms. They are k-means, Mean-Shift, DBSCAN, Gaussian mixture, and Hierarchical clustering (Rosidi, 2023; Kumar et al., 2021).

K-means is a clustering recipe that is typically applied. It operates as a centroid-based methodology and represents the simplest form of unsupervised

learning algorithm. K-means iterates through all the data points, implying that classifying data points may be time-consuming, especially with large datasets. Regardless, a constraint of k-means is that it supposes data to be circular in shape. Its distance calculation method relies on a circular path, causing it to struggle with correctly clustering non-circular data distributions. Therefore k-means is linear in m, the number of points and is efficient as well as simple provided that k, the number of cluster is significantly less than m (Firdaus et al., 2015).

The Figure 21 suggests an Distortion score elbow point at k=4, but k=2 also appears to be a Silhouette score elbow point. Therefore, it is clear based Silhouette results which value should be chosen as the elbow point for optimal clustering is k=2 Table 8 below. Although the Silhouette score reaches its maximum at k=2 (0.18), this alone may not be sufficient to determine the optimal value of k for clustering. When examining a PCA visualization showcasing four clusters at Figure 22, it's essential to consider other factors and evaluation metrics to make a more informed decision. In this scenario, it's notable that two clusters are larger while the other two are smaller, suggesting potential differences in data distribution and cluster characteristics that may warrant further investigation.

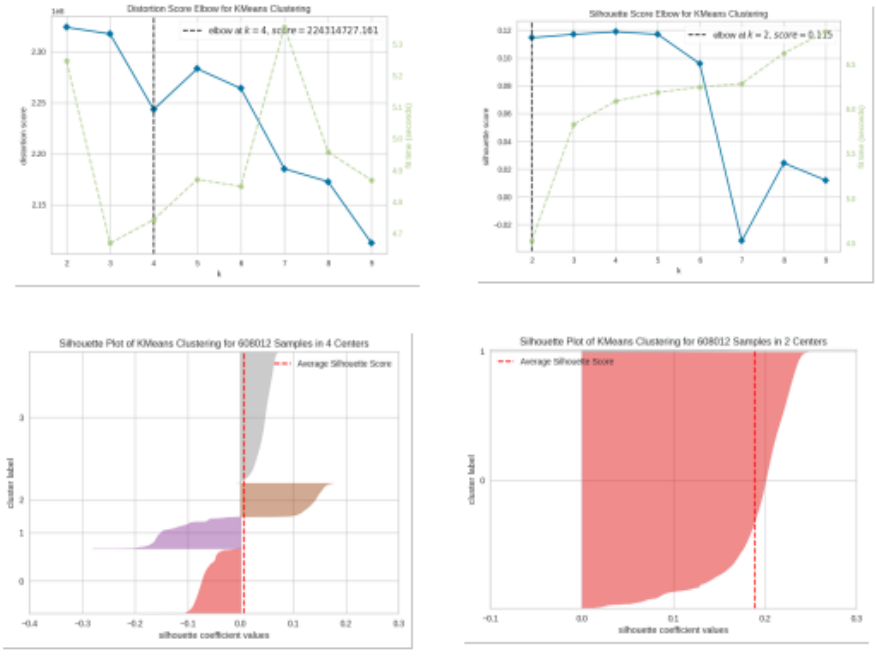


Figure 21: K-means clustering metrics

Customer segmentation involves categorizing customers into groups based on common behaviors, enabling companies to customize marketing strategies for each cluster more efficiently. In this dataset, segmentation extends beyond customers to include the end drivers' behaviors. Understanding how drivers utilize existing services can provide valuable insights for optimizing operations and enhancing service offerings.

Segmentation is typically done based on similarities in behavior and habits among the drivers. Clustering is a valuable tool for identifying various driver segments using CDR datasets, which contain real-time data encompassing theirs behaviors Table 8 has identified for two clusters and Table 9 for four clusters.

Table 8: Customer segmentation for two clusters with all features dataset

	2	3	4	5	7	10	11
clusters							
0	693.4885	16.4976	53.6185	19.1204	24.4891	394.6621	0.0988
1	44.4167	7.6246	49.1314	14.3921	518.2600	981.4167	0.0000

With drivers segmentation for four clusters in place Table 9, the company could now tailor its marketing efforts to each segmented group appropriately. This includes displaying different advertisements to different driver segments based on their unique characteristics and preferences what seems to be quite different like gross feature point of view.

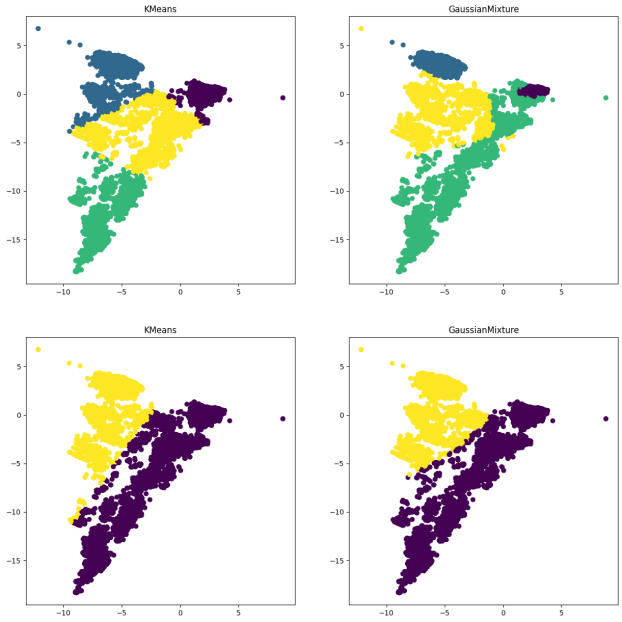
Table 9: Customer segmentation for four clusters with all dataset.

	2	3	4	5	7	10	11
clusters							
0	306.7365	19.1220	40.0600	8.3555	85.2247	259.2551	0.0599
1	96.5512	24.7785	39.9712	13.9348	155.5680	84.6581	0.1814
2	280.8137	14.8193	60.0132	24.2571	4.0785	500.7851	0.1187
3	3335.2149	19.6941	47.6895	14.1013	11.8645	136.9280	0.0738

After evaluating various clustering algorithms, clustering scores show significantly greater disparities compared to k-means and others as a result, these algorithms are ranked lower in here the table below. Note that the value of might heavily influence the cost of the algorithm, since a value too large might raise the cost of a neighborhood query to linear complexity (Firdaus et al., 2015). Continuing to assess the attributes of binary classification utilizing k-means through a classification report, the results were standing a very low compared to the metrics of all other machines learning models. In the classification results was seen that it is behaving completely opposite way with negative and positive samples. Similarly, model performance means that there is more samples expected to belong to negative class but were misclassified as "positive" by the classifier, representing much more times higher than that of the RF model. Clustering may not be the most effective model for identifying anomalies, but it serves as a valuable approach for grouping different behaviors within a dataset Figure 22 below. Enhancing classification accuracy of induced models is known as the main issue of noise detection techniques (Sluban, 2010).

There is possibility to shrink the number of all features before training models. To achieve this, it can be employed the PCA technique for feature reduction. Initially, it was generated optimal number of features per datasets to advise in determining the ideal number of components for PCA. At above bottom of Figure 22 can be seen PCA clustering by two components. There are two major clusters and two smaller once what is well in line with Table 8 and Table 9 results with details. It can be seen how these two different clustering model are defining 2 and 4 clusters. Yang (2024) was applying PCA on traffic data to reduce dimensionality can be implemented as extracting 3 important periods (morning, noon, evening) from totally 21 working hours in Taipei metro station’s traffic.

Figure 22: K-means and Gaussian mixture for 2 and 4 clusters



Therefore, the appropriate order of clustering algorithms would be:

1. K-means
2. Gaussian mixture
3. Hierarchical
4. Mean-Shift
5. DBSCAN

GMM computes the feasibility of a data point pertaining to a particular Gaussian distribution, determining the cluster to which it belongs. Unlike k-means, GMM doesn't require data to have a circular shape to function effectively. GMM offers soft boundaries, meaning data points can belong to multiple clusters simultaneously with varying degrees of certainty. For instance, a data point might have a 60% likelihood of belonging to cluster one and a 40% likelihood of pertaining to cluster two.

GMM necessitates the user to discover the count of components (clusters) prior to training the model. In this regard, one can utilize metrics such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC) to assist

in making this decision. When a subset comprising 10% of the training data was utilized to see below results Figure 23 because of the lengthy computing time required with all feature dataset.

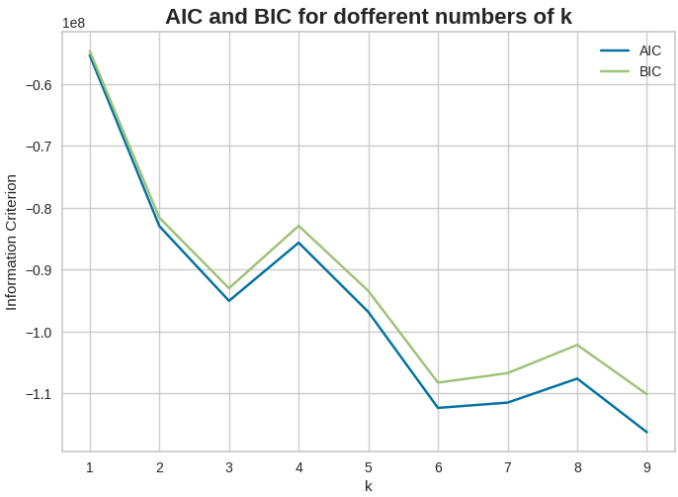


Figure 23: BIC and AIC have minimum at k = 9 and maximium k=1.

Here was utilized the Silhouette score to identify the optimal value of k, and then use this value to create and train GMM 10% of training data below Figure 24. See what number of cluster is the best Silhouette score indicates a higher score and it indicated the best fit with k = 2.

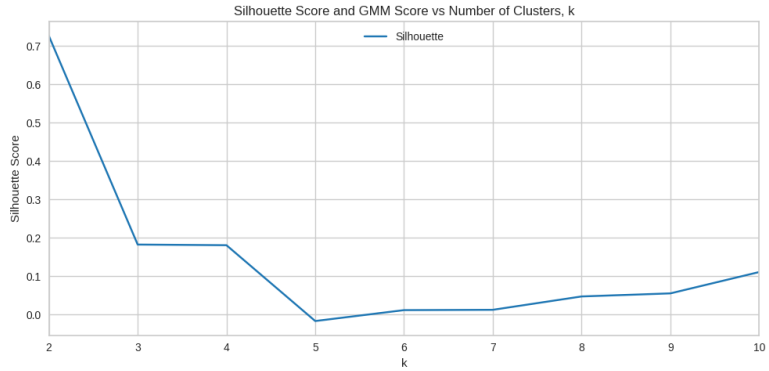


Figure 24: Silhouette score.

Mean-Shift is a valuable algorithm, especially for tasks involving images and computer vision processing. When using Mean-Shift, selecting the optimal bandwidth (radius) value is crucial for its performance.

This can be achieved using the estimate_bandwidth() function. Mean-Shift is mainly model which is iteratively shifting data points towards the mode, which signifies the high-density area within a region. It's worth noting that Mean-Shift is a hierarchical clustering algorithm, and its effectiveness depends on appropriately chosen bandwidth values. In this study was used 16.52 bandwidth (radius). It took several days to complete the modeling process using Mean-Shift. After running for more than four days without producing any results, the decision was made to stop the kernel. When a subset comprising 10% of the training data was utilized, the analysis yielded results indicating an estimated number of clusters of 210.

Agglomerative hierarchical clustering stands out as the most typical form of hierarchical clustering algorithm. Its purpose is to cluster objects based on their similarity to one another. This process involves creating a dendrogram, which aids in determining the optimal cluster count required for hierarchical clustering. Due to the computational complexity IDE gave MemoryError unable to allocate 1.34 TiB for an array with shape and data type float64 therefore did not dried it anymore. Agglomerative hierarchical clustering algorithms are computationally and storage-intensive. The finality of all merges can present challenges, especially with noisy, high-dimensional data like document data (Firdaus et al., 2015). When a subset consisting of 10% of the training data was used, the results are displayed in Figure 25 below.

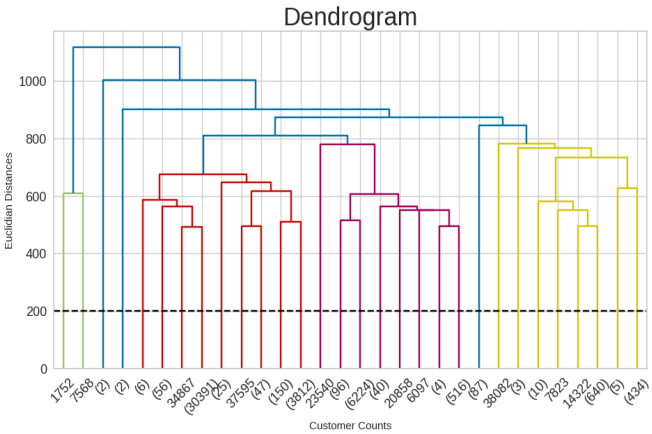


Figure 25: Dendrogram Clustering.

DBSCAN is a highly efficient clustering algorithm designed to classify datasets according to the density distribution of data points. It excels in identifying outliers within datasets characterized by significant differences between high and low-density regions. On the other hand, the algorithm suffers when there are too many dimensions or when the data clusters are similar in density which makes it harder to DBSCAN to distinguish the clusters (D.Dey, 2023). The Kernel crashed while executing code due to the error in cell. The IDE prompts for a code review in the cell(s) to identify potential causes of the failure; consequently, it has not been completed. DBSCAN can be expensive when the computation of nearest neighbors requires computing all pairwise proximity, as is usually the case for high dimensional data (Singh et, 2011). When a subset comprising 10% of the training data was utilized, number of clusters found by DBSCAN: 208 clusters, the maximum distance between two points was referred to as epsilon (eps) was set 10. Points that defined within an eps distance of each other are regarded as part of the same cluster.

5.7.2 Neural Network for binary classification

In Keras, Sequential API is a model type that facilitates the creation of neural network in a sequential layer-by-layer fashion where each layer are sequential stacked. In this case, we have 200 epochs, meaning the model will iterate the complete training dataset 200 times during training process. The Keras library furnishes metrics utilized to assess the values of each model (JavaTpoint, 2023). The model was created with the layers of neural network using the Sequential API. the Sequential API is simpler and more straightforward, suitable for models with a linear stack of layers. After constructing the model, it needs to be compiled, which involves defining how the model's performance should be measured and how it should improve during training. This involves specifying the loss function to quantify the deviation between predicted and actual values, as well as the optimizer, which dictates how the model's weights were enhanced after on the calculated loss.

The model need to be fitted, it is trained on a dataset to learn schemes and associations within the data. This process involves providing the model with input

data and related target values, iterating through epochs (cycles of training), and adapting the model's weights using the specified optimizer to minimize the loss function and enhance performance. It was obtaining the required output, with more than 99% of the data appearing to be correctly. The train model's accuracy was 99,42%, it suggests that the model's performance is a very good even without fine-tuning hyper parameters, indicating that it has learned meaningful patterns in the dataset. In such cases, training the model for a longer duration may help improve its performance by enabling it to learn more intricate schemes and associations within the data. This can be achieved by boosting the number of epochs while training or adjusting other hyperparameters such as the learning rate of the optimizer. However, it's crucial to observe the model's proficiency over training phase by the separate validation data in order to avoid overfitting, where the model becomes adept at succeeding in the data while fights to extend to novel, hidden one.

Here are few reason for Sequential API. Simplicity, perfect for constructing straightforward, feed forward networks arranged in a sequential manner. User-Friendly, intuitive and readily comprehensible, making it excellent for newcomers. Rapid Prototyping, allows for swift prototyping of basic models. Sequential API has some drawbacks, it may not be optimal for complex with multiple models, like featuring branching or multiple inputs and outputs.

GridSearchCV hyperparameter tuning was used by process results Table 11 based on train dataset at before model was run throughout. It provides predefined values for hyperparameters by specifying a dictionary containing each hyperparameter along with the values it can take. It exhaustively tries all combinations of the values provided in the dictionary and testing the model for each assortment using the cross-validation method. To mitigate overfitting, it is essential to utilize separate datasets for training and evaluating the model. It can be seen Figure 26 that there is a small risk for overfitting even it was utilised both datasets.

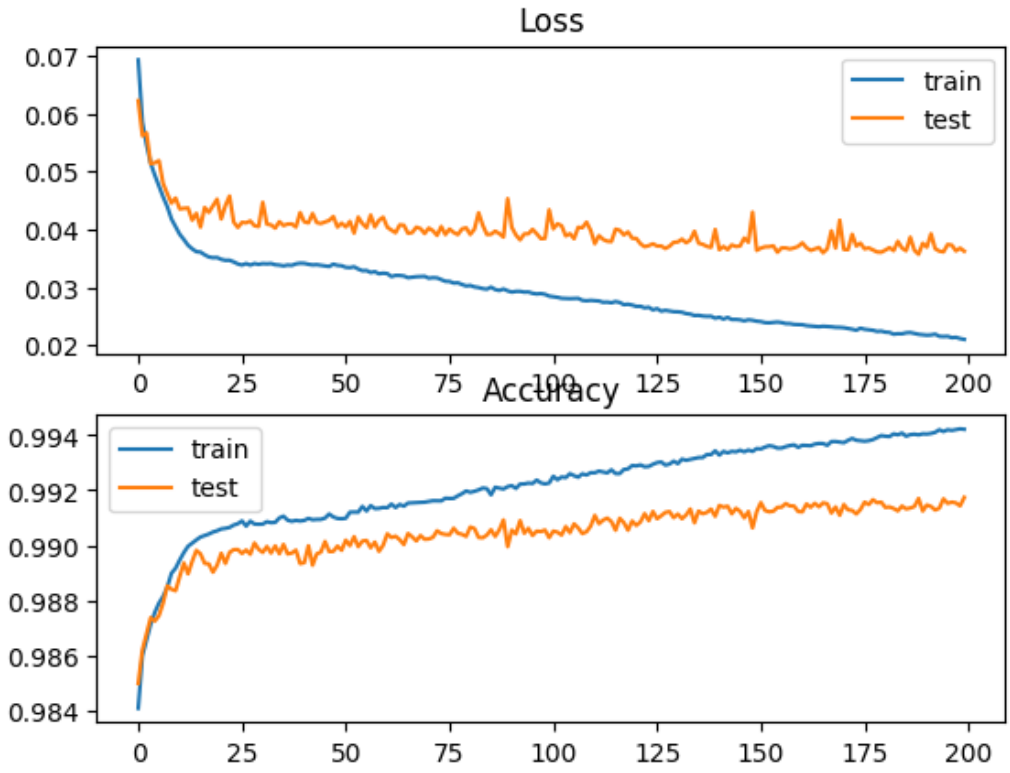
```
{ neurons = [24, 48, 98, 198, 395]
dropout_rate = [0.0, 0.1, 0.2, 0.3]
init_mode = ['uniform', 'lecun_uniform', 'normal', 'zero', 'glorot_normal',
'glorot_uniform', 'he_normal', 'he_uniform']
optimizer = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax',
'Nadam']
activation = ['softplus', 'softsign', 'relu', 'tanh', 'sigmoid', 'hard_sigmoid', 'lin-
ear'] }
Best: 0.9854 using {'model__activation': 'tanh', 'model__dropout_rate': 0.0,
'model__init_mode': 'he_uniform', 'model__neurons': 395, 'model__optimizer':
'RMSprop'}
```

Table 10: GridSearchCV hyperparameter values

Figure 26 displays two line plots: one depicting the learning curves of the loss on the train and test sets, and the other illustrating the accuracy of the classification on the train and test sets.

The plots imply that the model has achieved a beneficial fit for the obstacle. The train accuracy was improving well to 99,42%. The test accuracy was reaching at 99,17% at end of model.

It is not a clear overfit model because the test accuracy increase well by same way than the train accuracy then the test accuracy starts to and continues until the end of model. It seems that test dataset has a bit different content that train dataset, when examining the shape of learning curves.



Epoch 200/200

13301/13301 - 53s - train_loss: 0.0210 - train_accuracy: 0.9942 - val_loss: 0.0363 - val_accuracy: 0.9917 - 53s/epoch – 4ms/step.

Figure 26: Neural network learning curves

At Figure 26 graphs, it's evident that initially, both training and test accuracy are improving a very fast. Above curves are showing a great result while the model is fitting and adapting well.

However, with continued training, the accuracy improves more, and the cross-entropy loss do same. It comes a early point where the training accuracy surpasses the test accuracy, and despite further improvements in training accuracy, the test accuracy was not improve as well than training data or even almost stayed flat. This signifies that the training model fitted well, rendering it is suitable for use case.

The validation accuracy score of neural network is indicating that 99,17% of the total samples in test dataset. The precision of neural network indicated an predicted positive outcome, what was 95,17% of the time. The recall of neural network identified 87,75% of actual positive samples. The F1 score is the harmonic mean of precision and recall what was 91,31%. neural network was better performing than any supervised machine learning classifiers on all binary class dataset areas Table 4. As a result, the recall outcomes indicated predicted samples was much less false compared to the RF model. Table 12 is also focusing on more class 1.0 results what is anomaly detection of target feature. It can be seen that recall results, it not detecting all positive instances on this class which actually belong to it correctly where the biggest difference compared to precision is FN instances. The precision result means that there is much more false samples than what RF was predicted the number of samples. Neural network model results of samples what belong to the false class out of all predicted positive samples that the model classified as positive, which is called as precision.

Recall results are much better than RF one, while the result was not successfully predicted as the positive class out of all true samples, what actually belong to the positive class is referred to as recall or true positive rate. Cohens kappa value nearby 1 implies a high level of consensus, suggesting that the model or rater is highly reliable and accurate.

Table 11: Neural Network TensorFlow Keras metrics

Model: TensorFlow Keras
Classification Report:

	precision	recall	f1-score
0.0	0.99	1.00	1.00
1.0	0.95	0.89	0.92
accuracy			0.99
macro avg	0.97	0.94	0.96
weighted avg	0.99	0.99	0.99

Model performance:
Accuracy: 0.9923
Precision: 0.9545
Recall: 0.8860
F1 score: 0.9190
Cohens kappa: 0.9149
ROC AUC: 0.9419

In PyTorch is part of deep leaning, is a model type that facilitates the creation of neural networks in binary classification problems. It stands out as one of the leading libraries for deep learning, widely recognized and utilized within the field. PyTorch, an open-source Python library for deep learning, is created and managed by the Facebook AI lab. In PyTorch, tensors (torch.Tensor) are employed to manage and manipulate rectangular arrays of numerical data. Tensors resemble NumPy arrays but offer the advantage of graphics processing unit (GPU) acceleration. The torch.nn package is utilized to construct neural network within PyTorch. PyTorch's compatibility with the most widely used Python libraries further enhances its appeal among researchers and practitioners. Its popularity and vibrant community render PyTorch a preferred option for numerous deep learning initiatives.

In this research, the model will loop through the entire training dataset 100 times during training. Before further processing, it's recommended to convert data into PyTorch tensors initially. It is advised because PyTorch typically operates using 32-bit floating points, whereas NumPy defaults to 64-bit floating points. By converting to PyTorch tensors, potential issues stemming from implicit conversion can be circumvented.

It involves partitioning a larger dataset into segments, exploiting one segment as the test set while combining the remaining k-1 segments as the training set. This process is repeated k times, resulting in k different combinations like Figure 4. Subsequently, the experiment is iterated k times, and the average outcome is calculated. It serves to evaluate the model's design rather than a specific training instance. This is achieved by retraining the model with the same design using different training sets.

A model can be conceptualized as a sequence of layers, with a Sequential model in PyTorch being created by listing out these layers. Fully connected layers, also known as dense layers (Rastogi, 2023), are defined utilizing the Linear class in PyTorch, essentially performing matrix multiplication operations. The batch size is constrained by the memory capacity of the system, and the computational workload is directly proportional to the batch size. Experimentation through trial and error is often necessary to resolve the optimal results for the quantity of epochs and the bundle portion.

The evaluation of the model demonstrates a classification accuracy of approximately about 99.13% and 99.10% respectively for both the wide and deep models. A high accuracy score signifies that the both models are making a significant ratio of correct predictions. The wide model is more advanced than the deep model, in the sense that the mean accuracy is higher and its standard deviation is a bit higher. The two models were not having same amount of parameters. The deep model has more parameters what can not be seen better accuracy results. Even the deep model boasts 50% more parameters than the wide model, totaling 469,656 compared to 313,631, it follows that the running time

of the deep model will also be roughly 50% or more longer compared to the wide model.

Table 13 within the recall 87,43% result displayed in table, means that the false negative samples can be identified well what is much better then RF model was able to make. Consequently, the precision outcomes indicated 95,08% results what is a bit more false positive compared to the RF model. Table 13 primarily focuses on class 1.0 results, which pertain to anomaly detection of the target feature. It is evident from the recall results that not all positive instances in this class are being correctly detected, with the main difference compared to sharpness being the occurrence of false negative instances. Precision results indicate that the RF model was predicting less samples, while the PyTorch model predicted more samples, actually belonging to the negative class out of all the samples that were predicted to be of the positive once by the model, which is called precision. Table 13 all performance results looks overall rather same than TensorFlow Keras Table 12, only a small difference but no any major once.

Table 12: PyTorch metrics

```
==================================
Model:   PyTorch Wide
Classification Report:
              precision    recall  f1-score
         0.0      0.99      1.00      1.00
         1.0      0.95      0.87      0.91
    accuracy                           0.99
   macro avg  0.97      0.94      0.95
weighted avg 0.99      0.99      0.99
==================================
Model Wide performance:
Accuracy: 0.9916
Precision: 0.9508
Recall: 0.8743
F1 score: 0.9110
Cohens kappa: 0.9066
ROC AUC: 0.9360
==================================
Model accuracy:
Wide: 99.13% (+/- 0.03%)
Deep: 99.10% (+/- 0.02%)
Retrain a wide model: Final model accuracy: 99.16%
```

The Receiver Operating Characteristic (ROC) curve diagrams the true positive rate against the false positive rate of the model across various boundary or conditions Figure 27 with PyTorch model. The ROC curve consistently begins from the lower left corner and progresses towards the upper right corner. The closer the curve approaches the upper left corner, the higher the performance of your model. It's possible that by running more epochs, false positive results could potentially be improved accordingly.

Experimenting with different hyperparameters, including the number of epochs, can help optimize model performance. Both neural network models' final performance ROC AUC values are presented in Table 12 and Table 13 for comparison. Below curve clearly indicates an improving trend, affirming the effectiveness of the model.

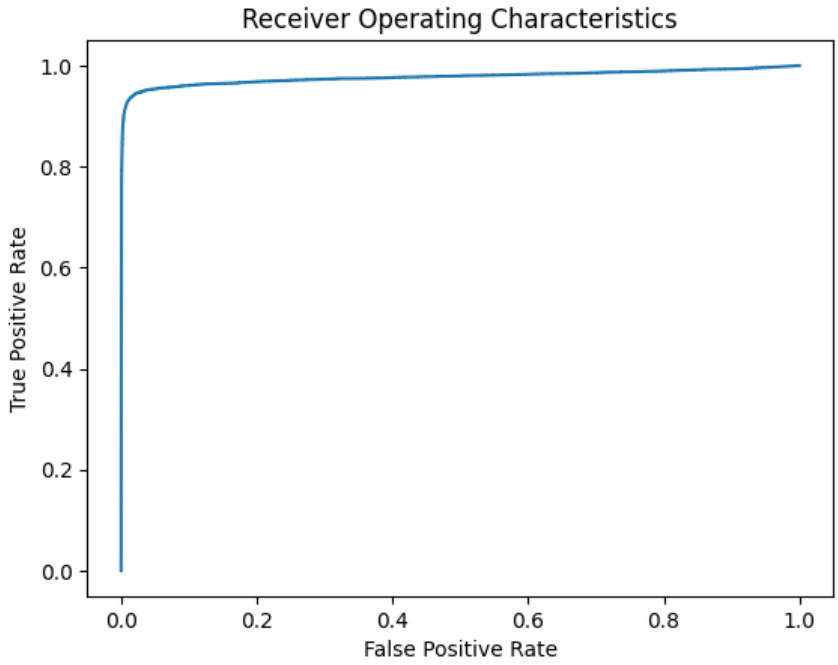


Figure 27: ROC curve

Neural networks are stochastic algorithms, it is meaning that running the same algorithm on identical data can result in training different models with varying each time the code is executed. Rather than being a flaw, this is a typical characteristic. The fluctuation in the model's performance necessitates fitting it multiple times and computing the average accuracy scores to obtain a reasonable approximation of its performance. There are possible to randomize and shuffle data at before model is computed what has direct impact on the results. Both neural network models exhibit a notable area for improvement, particularly in reducing false positives, which could be achieved through fine-tuning hyperparameters.

6 Additional future development

Building a classifier can be relatively straightforward with a few steps, but it outlines the preferred steps to properly construct your classifier. It begins with data cleaning and some basic feature engineering, followed by pipeline construction, model selection, model training and evaluation, and ultimately model persistence. There are still two additional steps remaining, the feature selection and the hyperparameter tuning, have been omitted here but it need to be covered in later. Depending on the complexity of data, stepping into feature selection and hyperparameter tuning can maximize your model's performance.

Typically, it relies on statistical formulas derived from extensive datasets. However, the utilization of neural network transforms the algorithms into a human-centric approach to managing input, employing artificial neurons. Neural network offer greater flexibility and performance for more complex classification problems with nonlinear relationships in the data. Method plays crucial roles in the field of machines learning, providing powerful tools for classification tasks across various domains. Classification tasks in machines learning involve predicting categorical labels for input data based on patterns learned from labeled training examples. It comprises associated nodes organized into layers, encompassing input, hidden, and output layers. Neural networks learn complex patterns and correlations in during backpropagation, errors are propagated backward through the network to fine-tune the weights and biases (Kostadinov, 2019), a crucial step in refining the model's performance.

Neural network can capture complex nonlinear relationships in the data and are highly adaptable to various types of classification tasks. Neural network are computationally intensive, demand large amounts of data for training, and are prone to overfitting if not properly regularized. Neural network have also been able to achieve outstanding performance, particularly Keras or PyTorch, can be effective in capturing complex patterns and identifying anomalies.

Understanding the distinctions between these types of classification problems is crucial for selecting appropriate algorithms, evaluation metrics, and modeling techniques to address specific application requirements effectively.

Increasing the depth of neural network by adding more layers can help capture complex consistencies and relationships in the data. However, adding too many layers may lead to overfitting, so it's essential to monitor performance on a validation set. Increasing the quantity of neurons in each layer can increase the model's effectiveness to acquire sophisticated consistencies in the data, so it's crucial to strike a balance.

Experimenting with number of activation functions (e.g., ReLU, sigmoid, tanh) can affect how information flows through neural network and may improve model performance. Choosing appropriate activation functions dependent on the challenge territory and the attributes of the data is important.

Optimization algorithms like Adam, Stochastic Gradient Descent, RMSprop, and others play a crucial role in updating the model's weights during training. Different optimization algorithms have different convergence properties and can affect training speed and performance. Experimenting with various enhancement algorithms can help find the one that works best for the specific problem.

Fine-tuning hyperparameters is making changes the following subsets, like learning rate, batch size, number of epochs, and hyperparameter can considerably effect model performance. Systematic hyperparameter tuning is utilizing ready-made practices like Grid search, Random search, or Bayesian where fine-tuning will be assisting in the optimum synthesis of regularization strength.

By systematically experimenting with these methods and parameters, unsupervised machine learning practitioners can iteratively enhance the performance of their models and achieve better results on their target tasks.

The dataset exhibits a low occurrence of missing values out of the total dataset, which is relatively minimal compared to the overall data was presented. However, there were few specific features stands out with approximately high of its values missing, signifying a substantial proportion of missing data within that feature characteristic. It's important to pay attention to missing data when developing features for applications because addressing this issue can significantly enhance data analysis later on.

Mattmann & Penberthy (2020) present a fourth category of meta-learning and Doshi et al. (2022) introduces semi-supervised learning, semi-supervised learning integrates elements of both supervised and unsupervised learning. In semi-supervised learning, the dataset is processed with a limited subset of tagged data, on the contrary the majority of the data remains untagged. A small subset of labeled data is utilized as training data for the larger dataset (Potrimba, 2022). Historical and real-time charging session data can be employed for a multitude of purposes and can be utilized in various combinations of learning models tailored to different use cases. Like Kern et al. (2023) propose the Intrusion Detection Systems (IDS) design based on a hybrid of detection- and regression models, whereby different detection model designs are considered (using supervised classification, semi-supervised novelty detection, as well as an ensemble of both). When initiating a charging session, data on electric vehicle battery status and charging point power is available, allowing CPMS to develop a regression model to track the relationship curve between these variables over time. Rosenstatter et al. (2021) consider the use of anomaly detection for Vehicle to Everything (V2X)-related interactions.

The computational complexity and power during the training phase came up with the clustering based techniques as one of key aspect. It might also impact on selecting other anomaly detection techniques on both the selected technique employed and the point of detection technique utilized within supervised or unsupervised machines learning models. Zanero wrote (2018) a security vulnerability in a cyber-physical system can potentially enable cyber-attacks that threaten physical functions.

Computer power holds a decisive role in securing the detection of anomalies in electric vehicle charging sessions, especially in areas of investigation such as like IDS and Vehicle to Grid power transfer. Satadru and Munmun (2020) were see it as cyber-physical component therefore proposes a method for detection of cyber-attacks against electric vehicle batteries, such as denial-of-charge attacks or overcharging of the battery, via an IDS in charging points what potentially can start a fire.

7 Results

Typically, machines learning models require a larger dataset to achieve good results. However, in this research study, a large dataset was needed even the data had already been pre-trained. Although machines learning have been the dominant technique for a long time. It was demonstrated that, for the specific issue under consideration, a straightforward machines learning method, specifically Random Forest, proves highly adept at identifying anomalies with minimal computational effort.

The findings and observations were relatively straightforward when comparing these two different models. The research study was concise and focused strongly on core datasets, making it easy to draw conclusions about the current state of machine learning and unsupervised model performance on charging session data. The recommendation can be described as follows: as computing power increases, it becomes possible to perform more powerful analyses. Although machines learning have been dominant in the tabular data analysis market, one difference with it is that it focuses more on long-range dependencies rather than local variations.

The background and previous studies have shown positive results in using machines learning and neural network models. It was a reason for inspiring to integrate all dataset as one into the field of tabular and categorical dataset analysis. Detecting anomalies in charging session data is pivotal for fraud detection and risk mitigation. In this research study, have showcased how neural network models can be leveraged to construct the anomaly detection further. By training the model on training data and it can recognize the reconstruction errors, potential fraudulent transactions can be identified. This methodology furnishes charging industry with a robust mechanism to fortify their operations and mitigate financial fraud risks. Additionally, the model can be customized and optimized further to cater to specific needs and enhance detection accuracy.

The charging industry manages a big volumes of data on a daily basis, encompassing transactions, customer interactions, energy data, and various other

more. Anomalies within this data can incur significant costs, potentially resulting in financial losses and tarnishing an organization's reputation. It is imperative to detect these anomalies to uphold financial integrity and security.

Another interesting insight came up from using clustering techniques was the ability to access driver behaviors. Through clustering, four main clusters were identified within the entire dataset, providing valuable insights into the diverse behaviors exhibited by drivers. After comparing these clustering models, it opted to utilize k-means as the primary model. Then, we partitioned the data into four clusters even results were better for two, as this number was deemed suitable for discerning customer behaviors effectively. However, it's crucial to note that each of these clusters possesses distinct characteristics.

The subsequent step involves identifying the clusters and determining the main features associated with each cluster group. This analysis helps in understanding the unique characteristics and behaviors represented by each cluster. Absolutely, clustering algorithms like k-means clustering should always be employed alongside other techniques and domain knowledge to derive meaningful insights from the data. Combining these methods allows for a more complete grasp of the underlying narrative and structures present in the dataset.

8 Conclusion

This research study on anomaly detection should enable readers to grasp the rationale behind employing specific anomaly detection techniques while also offering a comparative analysis of various methods. However, existing research has been conducted in an structured manner, not lacking a definition of anomalies. It was not complicates the task of establishing comprehension of the anomaly detection problem. A potential avenue for future research would involve consolidating the presumptions created by various approaches regarding normal and anomalous behavior within neural network what is subset of machines learning frameworks, but also consider different dataset. This approach could facilitate a more systematic and cohesive understanding of anomaly detection methodologies or even more.

It is crucial to meticulously select the appropriate metric(s) for appraising the performance of a classification models. Neural network topology with multiple layers provides greater opportunities for the network to extract essential features and combine them in meaningful nonlinear ways. This increased depth allows for more sophisticated learning and representation of complex relationships within the charging data. This research study proves the characteristics of binary classification and classifies them into these categories relationship between machine learning, neural network and deep learning. It involves straightforward conversion to binary format without optimizing the quantization function. On the other hand, optimized binarization techniques focus on minimizing quantization error, enhancing the loss function, and reducing gradient error to improve neural network performance to detecting even better anomalies in the charging data.

Neural network can also be utilized for clustering tasks, allowing the grouping of identical data nodes together. It is particularly useful for tasks such as identifying patterns in driver behavior or clustering similar images. Neural network are highly versatile and can be applied to a diverse array of machines learning tasks beyond just classification, demonstrating their broad utility in various domains. Analyzing the available dataset to comprehend driver behaviors and

identifying the main, most important features per driver cluster forms the foundational aspect of data science. This process enables us to uncover insights that are crucial for understanding customer segments and tailoring strategies accordingly.

References

Abdi, H. and Williams, L.J. (2010) ‘Principal component analysis: Principal component analysis’, Wiley Interdisciplinary Reviews: Computational Statistics, do2(4), pp. 433–459. Retrieved at: https://doi.org/10.1002/wics.101

Alaloul, Wesam Salah & Qureshi, Abdul Hannan 2020. Data Processing Using Artificial Neural Networks. IntechOpen. Retrieved 31.10.2023, from https://doi.org/10.5772/intechopen.91935.

Aurélien Géron 2017. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media

Banerjee, A. (2022) Principal Component Analysis (PCA) in Feature Engineering, Retrieved from https://medium.com/geekculture/principal-component-analysis-pca-in-feature-engineering-472afa39c27d

Basu, S., Bilenko, M., and Mooney, R. J. 2004. A probabilistic framework for semi-supervised clustering. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM Press, New York, NY, USA, 59–68

Bhattacharyya, D., & Kalita, J. (2014). Network anomaly detection: a machine learning perspective. CRC Press.

Biswal, A. (2023, November 7). What is Principal Component Analysis?

Bittrich, S., Kaden, M., Leberecht C., Kaiser, F., Villmann. T., Labudde, D., (2019), BioData Mining, Application of an interpretable classification model on Early Folding Residues during protein folding, Retrieved from https://biodatamining.biomedcentral.com/articles/10.1186/s13040-018-0188-2

Bughin, J; Hazan, E; Ramaswamy, S; Chui, M; Allas, T; Dahlström, P; Henke, N; Trench,M. (2016). Artificial Intelligence: The Next Digital Frontier. McKinsey Global Institute. Discussion paper. Retrieved from URL: https://www.mckinsey.com/~/media/McKinsey/Industries/Advanced%20Electronics/Our%20Insights/How%20artificial%20intelligence%20can%20deliver%20real%20value%20to%20companies/MGI-Artificial-Intelligence-Discussion-paper.ashx

Chandola, V.; Banerjee, A.; Kumar, V. (2009). "Anomaly detection: A survey"

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. R., & Wirth, R. (2000).

CRISP-DM 1.0: Step-by-step data mining guide. Retrieved: 2024 Feb 12, from /paper/CRISP-DM-1.0%3A-Step-by-stepdata-mining-guide-Chapman-Clinton/54bad20bbc7938991bf34f86dde0babfbd2d5a72

Datarobot, Underfitting, What does Underfitting Mean?, Retrieved: 2024 April 20, from https://www.datarobot.com/wiki/underfitting

Dasaradh, S.K. 2020. A Gentle Introduction To Math Behind Neural Networks. Retrieved 31.10.2023, from https://towardsdatascience.com/introduction-to-math-behind-neural-networkse8b60dbbdeba.

Ding, K., Lev, B., Peng, X., Sun, T., & Vasarhelyi, M. A. (2020). Machine learning improves accounting estimates: Evidence from insurance payments. Review of accounting studies, 25(3), 1098-1134. Retrieved from https://doi.org/10.1007/s11142-020-09546-9

Dey D., “DBSCAN Clustering in ML | Density based clustering,” Geeks for Geeks, 23 May 2023. [Online]. Retrieved 24 Jan 2024, from: https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/,

DOMO, Sixth edition of DOMO’s report Data never sleeps 6.0, Retrieved: 2024 March 18, from https://www.domo.com/solution/data-never-sleeps-6 (2020)

Doshi, R., Hiran, K., Jain, R., Lakhwani, K. (2022). Machine learning: Master supervised and unsupervised learning algorithms with real examples. BPB Publications.

Durga, S.N., & Rani, K.U. (2019). A perspective overview on machine learning algorithms. International Conference On Computational And Bio Engineering. Cham: Springer, pp. 353-364

Du-Harpur, X., Watt, F., Luscombe, N., & Lynch, M. (2020). What is AI? Applications of artificial intelligence to dermatology. British journal of dermatology (1951), 183(3), 423-430. Retrieved from https://doi.org/10.1111/bjd.18880

Fabiano, N. (2019). Ethics and the Protection of Personal Data. Systemics, Cybernetics and Informatics, 17(2), pp. 58-64.

Firdaus, Sabhia and Md. Uddin, Ashraf, A Survey on Clustering Algorithms and Complexity Analysis, IJCSI International Journal of Computer Science Issues, Volume 12, Issue 2, March 2015

Fleckenstein, M. & Fellows, L. (2018). Modern Data Strategy. Springer International Publishing. Retrieved from DOI: https://doi.org/10.1007/978-3-319-68993-7

Foy K., Communications & Community Outreach Office, AI models are devouring energy, MIT Lincoln Laboratory, Retrieved from https://www.ll.mit.edu/news/ai-models-are-devouring-energy-tools-reduce-consumption-are-here-if-data-centers-will-adopt, Sept 22, 2023

Foorthuis R, (2018), A Typology of Data Anomalies, UWV, La Guardiaweg 116, 1040 HG Amsterdam, The Netherlands, ralph.foorthuis@uwv.nl, Retrieved from https://arxiv.org/pdf/2107.01615.pdf

Foorthuis R, (2021), On the nature and types of anomalies: a review of deviations in data. International Journal of Data Science and Analytics (2021) 12:297–331, Retrieved from https://doi.org/10.1007/s41060-021-00265-1

GeeksforGeeks, Introduction to Dimensionality Reduction (2023), URL: https://www.geeksforgeeks.org/dimensionality-reduction/, Retrieved 24 April 2024.

Gilles Van Kriekinge * , Cedric De Cauwer , Nikolaos Sapountzoglou , Thierry Coosemans and Maarten Messagie, Electric Vehicle Charging Sessions Generator Based on Clustered Driver Behaviors (2023), World Electr. Veh. J. 2023, 14(2), 37; Retrieved from https://doi.org/10.3390/wevj14020037

Glen S., “ “Z-Score: Definition, Formula and Calculation”,” StatisticsHowTo.com, [Online]. Retrieved 19 July 2023 from: https://www.statisticshowto.com/probability-and-statistics/z-score/. .

Google. (2016). deepmind.com. Retrieved 24 Feb 2024, from URL: https://www.deepmind.com/blog/deepmind-ai-reduces-google-data-centre-cooling-bill-by-40

Grandini, M., Bagli, E., & Visani, G. (2020). Metrics for Multi-Class Classification: An Overview (arXiv:2008.05756). ArXiv. Retrieved from https://doi.org/10.48550/arXiv.2008.05756

EPA United States Environmental Protection Agency (2024), Exploratory Data Analysis, Retrieved 12 Feb 2024, from https://www.epa.gov/caddis/exploratory-data-analysis

European Commission (2021, April 21). Proposal for a regulation of the European parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts. EUR-Lex. Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:52021PC0206 59

European Commission (2023, June 20). Regulatory framework proposal on artificial intelligence. European Commission digital strategy. Retrieved from https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai

European Commission. (2018). Ethics and data protection. Retrieved from https://ec.europa.eu/info/sites/info/files_5._h2020_ethics_and_data_protection_0.pdf

European Parliament & Council (2016, April 27). Regulation (EU) 2016/679 of the European Parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). EUR-Lex. Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32016R0679

Evesto (2022), Retrieved 4 March 2024, from https://www.evesto.com/key-protocols-that-govern-electric-vehicle-ev-charging/#:~:text=Open%20Charge%20Point%20Protocol%20(OCPP,1%20being%20the%20latest

Ewerton Cristhian Lima de Oliveira, Santana Kauê, Junior Paulo Sergio Taube, Lima Anderson Júnior Claudomiro de Souza de Sales. “Biological Membrane-Penetrating Peptides: Computational Prediction and Applications.” Frontiers in Cellular and Infection Microbiology, 2022: 12.

Hawkins, D. M. (1980). Identification of outliers (Vol. 11). Springer

Hazal Gültekin, Classification vs Regression Dec 2, 2023, https://medium.com/@hazallgultekin/classification-vs-regression-e15c619604f7

Herman, M., Rivera, S., Mills, S., Sullivan, J., Guerra, P., Cosmas, A., & Kim, M. (2013). The field guide to data science. Booz Allen Hamilton.

Hoang, D., & Wiegratz, K. (2023). Machine learning methods in finance: Recent applications and prospects. European financial management: the journal of the European Financial Management Association, 29(5), 1657-1701. Retrieved from https://doi.org/10.1111/eufm.12408

Isha Salian 2018. SuperVize Me: What’s the Difference Between Supervised, Unsupervised, Semi- Supervised and Reinforcement Learning? Retrieved 06.06.2022, from https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/.

Jabbar, H., & Khan, R. Z. (2015). Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). Computer Science, Communication and Instrumentation Devices, 70(10.3850), 978-981.

Jain, A. K. and Dubes, R. C. 1988. Algorithms for Clustering Data. Prentice-Hall, Inc

Jason, B. 2019. Overfitting and Underfitting With Machine Learning Algorithms. Retrieved: 2023 May 25, from https://machinelearningmastery.com/overfittingand-underfitting-with-machine-learning-algorithms/

JavaTpoint, Overfitting in Machine Learning. Retrieved: 2024 Jan 25, from https://www.javatpoint.com/overfitting-in-machine-learning

Jiawei Han, Micheline Kamber, Jian Pei. “3 - Data Preprocessing.” In Data Mining (Third Edition), by Morgan Kaufmann, 83-124. 2012.

Jolliffe I. T. and J. Cadima, “Principal component analysis: a review and recent developments,” Philosophical Transactions: Physical Sciences and Engineering, 2016.

Kern D., Krauß C. Darmstadt University of Applied Sciences, Germany and Hollick M., Technical University of Darmstadt, Germany. Detection of Anomalies in Electric Vehicle Charging Sessions, Retrieved from DOI: https://doi.org/10.1145/3627106.3627127 Dec, 2023.

Kostadinov S., (2019) Understanding Backpropagation Algorith, https://towards-datascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd
Kumar V, Tan P-N., Steinbach M., Karpatne A., Introduction to Data Mining (Second Edition), University of Minnesota Cluster Analysis: Basic Concepts and Algorithms, Retrieved from https://www-users.cse.umn.edu/~kumar001/dmbook/ch7_clustering.pdf, 24 Mar, 2021

Manzini N., Programming and Design, Single hidden layer neural network, Retrieved from https://www.nicolamanzini.com/single-hidden-layer-neural-network/, 19 Nov 2017

Mattmann, C. A., & Penberthy, S. (2020). Machine Learning with TensorFlow (Second edition.). Manning.

Marr, B. (2017). Data Strategy : How to Profit From a World of Big Data, Analytics and the Internet of Things. 1st Edition. Kogan Page

Martinez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez-Orallo, J., Kull, M., Lachiche, N., Flach, P. (2021). CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories. IEEE transactions on knowledge and data engineering, 33(8), 3048-3061. Retrieved from https://doi.org/10.1109/TKDE.2019.2962680

Meriem Bahi, Mohamed Batouche. “Deep Learning for Ligand-Based Virtual Screening in Drug Discovery.” 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS). 2018. 1-5.

Milo, T., & Somech, A. (2020). Automating exploratory data analysis via machine learning: An overview. Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, pp. 2617-2622.

Minal Jain 2022, An Overview of Principal Component Analysis, Retrieved from https://heartbeat.comet.ml/an-overview-of-principal-component-analysis-part-1-b6476fd78bf7

Myatt, G. J., & Johnson, W. P. (2014). Making sense of data I: A practical guide to exploratory data analysis and data mining (Second edition.). Wiley.

Nassif A. B., Talib M. A., Nasir Q., and Dakalbab F. M. 2021. Machine learning for anomaly detection: A systematic review. Ieee Access 9 (2021), 78658–78700.

OCA Open Charge Assosiation standard, Connecting the EV Charging Industry, Retrieved: 2024 March 18, from https://openchargealliance.org/

OCPI 2.2.1, Open Charge Point Interface OCA 2021, https://github.com/ocpi & https://ocpi-protocol.or, document version 2.2.1, 2021-10-06 Retrieved from https://evroaming.org/app/uploads/2021/11/OCPI-2.2.1.pdf

Leekha, G. (2021). Learn AI with Python. BPB Publications.

Liu Y., Li Z., Xiong H., Gao X., Wu J., (2010), Understanding of Internal Clustering Validation Measures, School of Economics and Management, University of Science and Technology Beijing, China, School of Economics and Management, Beihang University, China. wujj@buaa.edu.cn

Patel, K (2023). Understanding the Limitations of K-Means Clustering. https://medium.com/@kadambaripatel79/understanding-the-limitations-of-k-means-clustering-1fb5335f7859

Pearson K., (1901) ‘On lines and planes of closest fit to systems of points in space’. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2:559-572.

Potrimba, P. (2022, December 16). roboflow. Retrieved from What is Semi-Supervised Learning? A Guide for Beginners: Retrieved from https://blog.roboflow.com/

Priya, A. (2021). Case Study Methodology of Qualitative Research: Key Attributes and Navigating the Conundrums in Its Application. Sociological Bulletin, 70(1), 94-110. Retrieved from https://doi.org/10.1177/0038022920970318

Provost, F. & Fawcett, T (2013A). Data Science and its Relationship to Big Data and Data-driven Decision Making. Big Data Journal, Vol. 1, No. 1, March 2013

Purdy, M. and Daugherty, P. (2017). How AI Boosts Industry Profits and InnovationBy. Retrieved from URL: https://www.accenture.com/fr-fr/_acnmedia/36dc7f76eab444cab6a7f44017cc3997.pdf

Rastogi V., (2023). Fully Connected Layer, Retrieved from https://medium.com/@vaibhav1403/fully-connected-layer-f13275337c7c

Rosenstatter T., Olovsson T., and Almgren M., 2021. V2C: A Trust-Based Vehicle to Cloud Anomaly Detection Framework for Automotive Systems. In Proceedings of the 16th International Conference on Availability, Reliability and Security. 1–10.

Rosidi N., KDnuggets Market Trends & SQL Content Specialist on May 11, 2023 in Machine Learning, Clustering in machine learning with Python: algorithms, evaluation metrics, real-life applications, and more. Retrieved from https://www.kdnuggets.com/2023/05/clustering-scikitlearn-tutorial-unsupervised-learning.html

Satadru D. and Munmun K., 2020. Cybersecurity of Plug-In Electric Vehicles: Cyberattack Detection During Charging. IEEE Transactions on Industrial Electronics (2020).

Scikit-Learn 1.4.2, (2023), 3.1 Cross-validation: Evaluating estimator performance. Retrieved 2024 April 25, from https://scikit-learn.org/stable/modules/cross_validation.html

Singh, Shalini S., and N. C. Chauhan. "K-means v/s K-medoids: A Comparative Study." National Conference on Recent Trends in Engineering & Technology. 2011.

Sivula, A. (2021). Exploring Machine Learning in Higher Education Industry: Student Performance Prediction [Master's thesis, Jamk University of applied sciences]. Theseus. Retrieved from https://urn.fi/URN:NBN:fi:amk-202105189164

Shoresh, N., & Wong, B. (2012). Data exploration. Nature methods, 9(1), 5. Retrieved from https://doi.org/10.1038/nmeth.1829

Sluban B, Gamberger D, Lavra N (2010) Advances in class noise detection. Front Artif Intell Appl 2151:1105–1106

Steven M. Holland, Univ. of Georgia: Principal Components Analysis (PCA), Retrieved from http://stratigrafia.org/8370/lecturenotes/principalComponents.html

Taylor & Francis 2(11). Dublin Philosophical Magazine and Journal of Science. Available at: 559–572, Retrieved from DOI :10.1080/14786440109462720.

Theissler A., Pérez-Velázquez J., Kettelgerdes M., Elger G. “Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry.” Reliability Engineering & System Safety 215 (2021): 107864

Tridens Technology (2023), OCPP Protocol Explained (OCPP 1.6 and OCPP 2.0.1) Retrieved from https://tridenstechnology.com/ocpp-protocol/ , Accessed 2 Feb 2024.

Virta, Charge the World with us, A global forerunner in smart EV Charging solutions, Retrieved: 2024 April 22, from https://www.virta.global/company2

Wang, Q., & Bu, S. (2020). Deep learning enhanced situation awareness for high renewable-penetrated power systems with multiple data corruptions. IET renewable power generation, 14(7), 1134-1142. Retrieved from https://doi.org/10.1049/iet-rpg.2019.1015

Yale, Categorical Data: two-way tables, bar graphs, segmented bar graphs, (2024), Retrieved from http://www.stat.yale.edu/Courses/1997-98/101/catdat.htm

Yang B., (2024). PCA & K-Means for Traffic Data in Python. Retrieved from https://towardsdatascience.com/pca-k-means-for-traffic-data-in-python-a0ec66ab4789

Yang, B., Nazari, R., Elmo, D., Stead, D., & Eberhardt, E. (2023). Data preparation for machine learning in rock engineering. IOP conference series. Earth and environmental science, 1124(1), 12072. Retrieved from https://doi.org/10.1088/1755-1315/1124/1/012072

Yin, R. (2009). Case study research: design and methods (4th edition). Sage Publications Inc: California.

Yu, L.L. (2004). Efficient feature Selection via Analysis of Relevance and Redungdancy. Journal of Machine Learning Research, 5, pp. 1205-1224.

Zanero S. 2018. When cyber got real: Challenges in securing cyber-physical systems. In 2018 IEEE SENSORS. IEEE, 1–4.

Zhu X, Wu X (2004) Class noise vs. attribute noise: a quantitative study of their impacts. Artif Intell Rev 223:177–210.